AUGMENTED INTELLIGENCE:

DEEP LEARNING, MACHINE LEARNING, COGNITIVE COMPUTING, EDUCATIONAL DATA MINING

Editors:
Om Praksh Jena
Alok Ranjan Tripathy
Brojo Kishore Mishra
Ahmed A. Elngar

Bentham Books

Advances in Computing Communications and Informatics

(Volume 3)

Augmented Intelligence: Deep Learning, Machine Learning, Cognitive Computing, and Educational Data Mining

Edited by

Om Praksh Jena & Alok Ranjan Tripathy
Department of Computer Science
Ravenshaw University
Odisha, India

Brojo Kishore Mishra
Department of CSE, GIET University
Gunupur, Odisha 765022, India

&

Ahmed A. Elngar
Faculty of Computers & Artificial Intelligence
Department of Computer Science
Beni-Suef University, Beni-Suef, Egypt

Advances in Computing Communications and Informatics

Volume # 3

Augmented Intelligence: Deep Learning, Machine Learning, Cognitive Computing,

Educational Data Mining

Series Editors: Pradeep Kumar Singh, Bharat Bhargava & Wei-Chiang Hong

Volume Editors: Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra & Ahmed A. Elngar

ISSN (Online): 2737-5730

ISSN (Print): 2737-5722

ISBN (Online): 978-981-5040-40-1

ISBN (Print): 978-981-5040-41-8

ISBN (Paperback): 978-981-5040-42-5

First published in 2022.

need for a court order if at any point you breach any terms of this License Agreement. In no event will any delay or failure by Bentham Science Publishers in enforcing your compliance with this License Agreement constitute a waiver of any of its rights.

3. You acknowledge that you have read this License Agreement, and agree to be bound by its terms and conditions. To the extent that any other terms and conditions presented on any website of Bentham Science Publishers conflict with, or are inconsistent with, the terms and conditions set out in this License Agreement, you acknowledge that the terms and conditions set out in this License Agreement shall prevail.

Bentham Science Publishers Pte. Ltd.
80 Robinson Road #02-00
Singapore 068898
Singapore
Email: subscriptions@benthamscience.net

CONTENTS

PREFACE

Augmented intelligence is an alternate approach to artificial intelligence (AI), which emphasizes on AI's assistive role, underlining the fact that cognitive technology is intended to improve human intelligence instead of replacing it. It enhances human skills of reasoning in a robotic system or software, including expectancy, educational Mining, and problem solving, recollection & sequencing, and decision-making capabilities. With the collaboration of Machine Learning, Deep Learning and Cognitive Computing it can be highly implemented technologies in the near future. It can overcome the human limitations that hinder effective research. This AI layer complements knowledge workers' efforts and allows them to essentially automate or drastically reduce the time devoted to low-value tasks, while focusing more on higher impact issues. Augmented Intelligence can eliminate time-consuming tasks, freeing up teams to focus on higher value analysis.

There are many research problems which are NP-Complete. Being a research fellow, one has to provide an optimal solution to the hard-to-solve problems in different sectors of optimization problems. For such needs, this book will give helpful insight while identifying problem statements and then the idea to develop time-optimal strategies for such unresolved issues. This book is also helpful to write articles on various issues which involve Augmented Intelligence based on ML, DL, Cognitive Computing and its various applications in real life scenario. They can grab the opportunity from this book to devolve some new algorithms, models, patents based on Augmented Intelligence to design more robust, efficient, accurate models.

This book will be able to define the model which supports Knowledge Discovery, educational model design, self-learned system, logical reasoning, virtual learning, social network analysis, *etc*.

Om Prakash Jena & Alok Ranjan Tripathy
Department of Computer Science
Ravenshaw University
Odisha
India

Brojo Kishore Mishra
Department of CSE
GIET University, Gunupur
Odisha 765022
India

&

Ahmed A. Elngar
Faculty of Computers & Artificial Intelligence
Department of Computer Science
Beni-Suef University, Beni-Suef
Egypt

List of Contributors

Ahmad Aljabr	College of Computing and Informatics, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia
Ali Aloss	Department of Computer Science and Engineering, Siksha'O'Anusandhan (Deemed to be university), Bhubaneswar, Odisha, India
Alexandre G.M. de Freitas	Centro de Estudos em Leite, Departamento de Tecnologia Rural e Animal, Universidade Estadual do Sudoeste da Bahia, Itapetinga, Brazil
Aju D.	Vellore Institute of Technology, Vellore, Tamil Nadu 632014, India
Arin Paul	School of Electronics Engineering, Vellore Institute of Technology, Tamil Nadu, India
Barnali Sahu	Department of Computer Science and Engineering, Siksha'O'Anusandhan (Deemed to be university), Bhubaneswar, Odisha, India
Bárbara E.A. de Magalhães	Instituto de Química, Universidade Federal da Bahia, Salvador, Brazil
Balabhim Sankrappa Biradar	Professor, Department of Library and Information Science, Kuvempu University, Shankaraghatta, Karnataka, India
B.U. Kannappanavar	Librarian, Sahayadri College, Shivamogga, Karnataka, India
Carlos Ankora	Vellore Institute of Technology, Vellore, Tamil Nadu 632014, India
Jagjit Singh Dhatterwal	Department of Computer Sciences & Applications, PDM University, Bahadurgarh, Haryana, India
Kailash Kumar	College of Computing and Informatics, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia
Kuldeep Singh Kaswan	School of Computing Science and Engineering, Galgotias University, Greater Noida, India
Lucas A.C. Minho	Departamento de Química, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil
Makram Soui	College of Computing and Informatics, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia
Mallikarjun Kappi	Department of Library and Information Science, Kuvempu University, Shankaraghatta, Karnataka, India
Madhu S.	Department of Studies in Library and Information Science, Kuvempu University, Shankarghatta, Shimoga Dist, Karnataka 577451, India
Nada Namani Zitouni	University of Gabes, Zrig Eddakhlania, Tunisia
Om Prakash Jena	Department of Computer Science, Ravenshaw University, Cattuck, Odisha, India
Priyanka Mishra	Indian Institute of Information Technology Kota, MNIT Campus, Jaipur, India
Ritayan Patra	School of Electronics Engineering, Vellore Institute of Technology, Tamil Nadu, India
Salima Smiti	University of Manouba, Manouba, Tunisia

Sayak Banerjee School of Electronics Engineering, Vellore Institute of Technology, Tamil Nadu, India

Sumit Kumar Jindal School of Electronics Engineering, Vellore Institute of Technology, Tamil Nadu, India

CHAPTER 1

Integrating Educational Data Mining in Augmented Reality Virtual Learning Environment

Carlos Ankora[1] and **Aju D.**[1,*]

[1] *Vellore Institute of Technology, Vellore, Tamil Nadu 632014, India*

Abstract: Virtual learning environments have become an essential tool, incorporated in learning activities in educational institutions and individuals' homes, especially during the COVID-19 pandemic. Digital devices provide the platform for the learning environment, but learning sometimes becomes passive and boring. Augmented reality provides learners with the needed motivation, engagement, thereby boosting the learner's activity within the virtual learning environment. It augments the traditional learning materials with 3D objects, animations, audio and visual elements, which offer better interactivity for a rich learning experience.

This study aims to give an overview of the development of an augmented reality system to provide a virtual learning environment that delivers a more engaging and motivating lesson, story and experience. The study incorporates Scrum methodology, an agile software development practice that uses small increments called sprints to develop the virtual learning environment in several usable modules. The study also discusses the software tools, Blender and Unity 3D, to develop 3D models and the augmented reality modules for the virtual learning environment. The system uses image targets as markers to project 3D objects to augment the images from the traditional learning materials and offer a better visual experience. The development incorporates features of Educational Data Mining to optimise users' learning styles and learning experiences. This chapter will demonstrate augmented reality technologies to implement a virtual learning environment that will offer an interactive and engaging learning experience.

Keywords: Augmented Reality, Agile Development, Educational Data Mining, Virtual Learning Environment.

1.1. INTRODUCTION

The educational system has gone through several technological incorporations over the years. Educational institutions have transformed the teaching and learning process, resulting in progressive consequences [1].

* **Corresponding author Aju D:** Vellore Institute of Technology, Vellore, Tamil Nadu 632014, India; Tel: +919150477551; E-mail: daju@vit.ac.in

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

Despite its limitations, the traditional classroom remains an essential aspect of education. Both instructors and learners still value the face-to-face model of teaching.

Virtual learning environment have become an essential tool incorporated in learning activities in educational institutions and individuals' homes, especially during the COVID-19 pandemic. Schools were closed down, and the management of these institutions resorted to virtual class sessions to continue teaching when it did not seem like the pandemic would soon be over.

Some institutions utilised live teaching sessions; students log in to online video conferencing and collaborative systems to watch and listen to their instructors delivering lectures. These systems provide features for questions *via* chat or using the device's microphone, just like students raising hands to ask questions in class. Others instead use recorded class sessions (audio or video) and make them available online for students to watch online or download to their devices and watch offline. Such systems provide discussion forums to encourage peer discussions with the instructor to moderate the questions and comments posted.

Digital devices provide the platform for the learning environment, but learning sometimes becomes passive and boring. Students usually watch or listen to these sessions while getting distracted and find themselves doing other activities alongside. Augmented reality provides learners with the needed motivation, engagement, thereby boosting the learner's activity within the virtual learning environment. It augments the traditional learning materials with 3D objects, animations, audio and visual elements, which offer better interactivity for a rich teaching and learning experience for both instructors and learners.

1.2. VIRTUAL LEARNING ENVIRONMENTS (VLE)

Virtual learning environments usually comprise resources and tools used within a network-connected computer systems environment. A wired or wireless network shares information among the connected systems. The information includes text, image, video and audio files shared among various stakeholders in the virtual learning environments, running on the internet or an institutional intranet. Such networks also allow communication between students and their instructors within the virtual learning environments. The communication could be synchronous, occurring simultaneously from both nodes, or asynchronous, occurring at different times based on the time schedules of the users. Virtual learning environments provide the platform within the educational infrastructure to accommodate all these functionalities [2].

The educational sector has embraced virtual learning environments to promote information and content sharing and communication among students and instructors [1]. Studies have shown that peer interaction and collaborative learning activities are essential for a virtual learning environment in education. A virtual learning environment provides the platform for students to collaborate individually or in groups to learn and work on activities and projects without being physically around each other. Virtual learning environments deliver content in various formats and allow students to apply different strategies in the learning process. Students can use learning materials based on preference and suitability. They can revisit previous content and control the pace of learning at their convenience [2].

Learning environments come with a focus on the delivered content. Virtual learning environments are built-in with features, activities and content to support the engagement and interactivity of learners and create an immersed state within the VLE [3].

The development and implementation of a virtual learning environment should focus on its content and the presentation to the students. Students have varied learning styles; some prefer reading text, others listen to audio files, and others watch video demonstrations. A properly structured virtual learning environment will provide the necessary resources for students to access learning materials in the format they are convenient with, and they should have the option of switching between different formats of the same content [4].

Several studies have recommended features and functionalities that should be present in a virtual learning environment [4 - 7]. An implemented system might not have all these features (Fig. **1**) deployed. However, all of these have unique functionalities that will make a virtual learning environment provide an outstanding teaching and learning experience for both instructors and students and all other necessary stakeholders.

1.3. AUGMENTED REALITY (AR)

Augmented reality technology offers a richer augmentation to information by including virtual objects implanted into images; the virtual objects come in various forms – text, images, audio, video, animation, 3D models. Augmented reality gives an illusion of the objects within the same physical environment [8 - 10]. Augmented reality uses virtual objects displayed in a real-world physical environment to provide an enhanced experience.

Some studies on augmented reality evidence the enriched teaching and learning experience AR provides to its users [11].

The implementation of augmented reality in education has allowed instructors and students to go beyond the static presentation of learning materials and media to enhance them with virtual 3D objects, giving room for further exploration of learning materials accessed by the students [1]. Augmented reality creates an immersive learning environment giving students a different experience in the learning process.

AR technology mainly runs on a computing device, mainly a smartphone, which uses the camera to pick images from the real-world environment while displaying the additional virtual 3D images to provide extra information to the user.

Fig. (1.1). Features of Virtual Learning Environment.

1.3.1. Elements of AR in a VLE

The main elements of an augmented reality implementation in a virtual learning environment include a target marker, an augmented reality engine, an optical sensing device and a display. A smart mobile device represents the primary device; it is easier to use its display and the camera as an optical sensing device.

An image target that contains the marker of the image to be detected is prepared with distinct characteristics to make it easy to detect. The camera of the mobile phone device scans the physical environment and picks the image from the target marker. The system transmits the detected information to the AR engine to compute the position and angles of the detected marker.

A match to the marker generates the object shown on the mobile device's display. The associated object could be textual information or a graphic image that provides extra information about the image target. A 3D model, an audio or video clip, or an animation could be embedded to enrich the visual experience and engagement [12].

1.3.2. Target Markers used in AR Applications

AR applications use an object as a target to create a 3D space [9]. They survey some types of targets – markers, coded markers, images, multi targets, text recognition, simple shapes, object recognition.

The simple marker has a border easily identified as a target for rendering. The marker uses less processing power and is usually effective when in operation. This coded marker causes constraints within the border, but the patterns could vary. Each group of patterns generates varying virtual objects. An image could be utilised in place of the marker; it prevents the generation of various markers linked with the virtual objects.

The images should be clear with distinct outlines and colour contrast to make it easier with the image pattern analysis and matching for pattern identification.

The text could be used as a target for virtual objects; it identifies the text in particular font styles and sizes. Simple 2D or 3D shapes are used in AR targets by identifying the edge dimensions and texture patterns. Objects are also used as AR targets; 3D objects could be identified and used within a 3D space. Some targets combine different object modes for identification. The AR device picks up the target related to the object to be identified and displayed in a 3D space.

1.3.3. Hardware Platforms/Peripherals

Several devices and peripherals render AR applications; handheld mobile devices (smartphones, tablet personal computers), wearable smart glasses, head-mounted displays, laptop computers, and personal digital assistants [8, 13, 14]. The most commonly used among these are mobile devices [9].

Most mobile devices are equipped with cameras, usually front-facing and rear-facing. The rear-facing camera picks up the video signals from the physical environment by aiming the camera to the desired target. The device identifies the marker and renders the virtual object into the 3D space based on the camera and marker's position and angle. The rendering requires a lot of graphics and CPU processing, which most good smartphones possess these days. The mobile devices also contain GPS sensors mainly used for navigational purposes. Some AR applications use the marker's location to determine the resultant object displayed on the screen. This feature is also helpful because mobile devices are built to make it easier to carry around the real-world environment, which is valuable to AR applications, detecting targets around the environment.

1.3.4. AR Applications in Education

The educational sector has seen some of the mainstream implementations of augmented reality technology. AR applications are used for skill training, classroom learning, and home learning (Fig. **2**). AR technology provides virtual examples to learning materials and includes gaming elements to increase learner's engagement and interactions. Trainers use augmented reality applications to compliment workshop skill practices, especially in situations where the learners cannot access equipment readily and where safety issues are very much of a concern [8, 13, 15, 16].

1.4. DEVELOPMENT OF METHODOLOGY FOR AR IN VLE

Advancements in technology necessitate organisations and institutions to stay up-to-date in technological implementations at the workplace. These trends include hardware, software, and communication network installations; educational institutions are also expected to follow the trends. Implementation of software systems improves operations; new systems replace existing systems while existing systems are redesigned and updated to meet new system and operational requirements.

The designing and development of systems for virtual learning environments consider the context, cultural resources, and socio-cultural features of the learning

environments. In a study [7], two critical aspects must also be considered when designing smart learning environments – involvement of the users in the design process and the inclusion of functionality to accept feedback from system users.

The system design process has moved from the traditional approach where developers meet users at the early stages of system development to take a list of requirements, which becomes a checklist for the User Acceptance Test (UAT). It has become necessary that users and other stakeholders in a system being developed be involved in the whole design and development process. Feedback from users should run a complete cycle on the lifetime of the software, from the inception of the design through development and maintenance stages till the software is no more in use. Continuous feedback from software users keeps the system current and satisfies all relevant requirements. Software update becomes a recurring process over the system's lifetime.

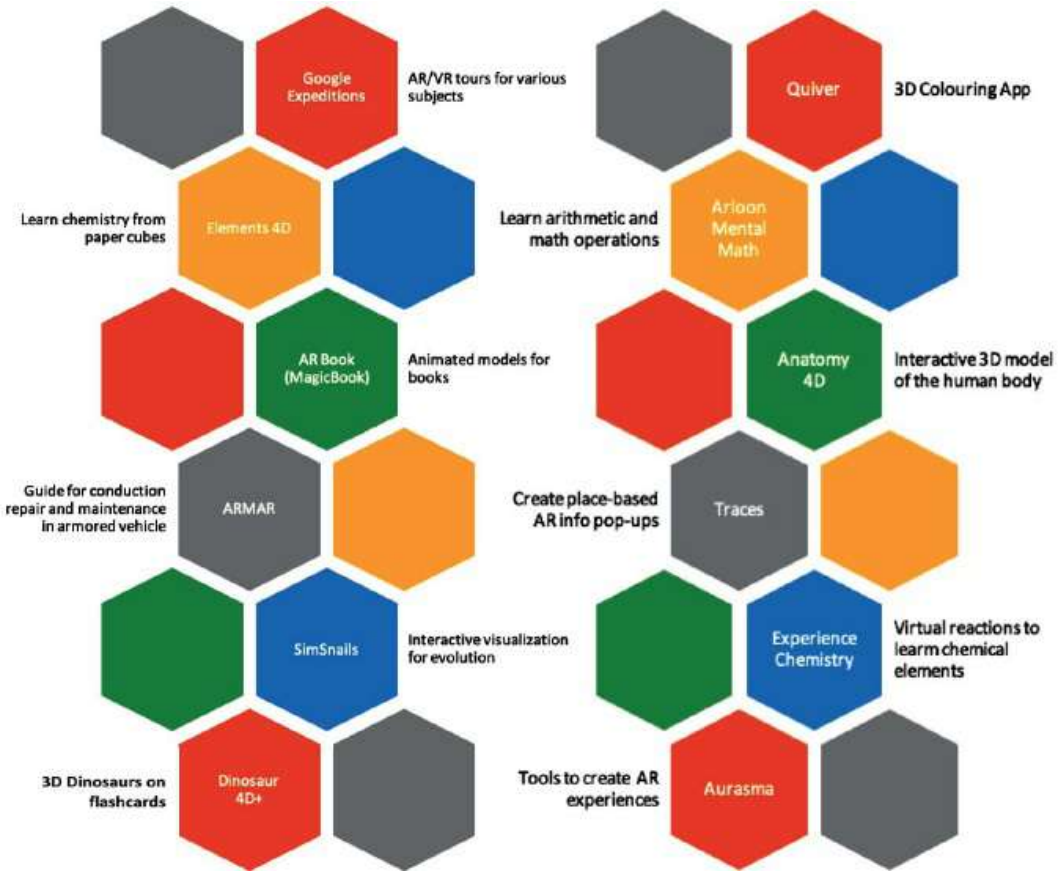

Fig. (1.2). AR applications used in education.

1.4.1. Agile Methodology

The Agile Software Development Methodology is one of the user/customer-focused methodologies that organises the whole development process into small increments called sprints or iterations. At the end of each iteration, a product (minimum viable product) is presented to the customer. The Agile values and principles emphasise interactions and collaborations among all team members, of which the user is a significant participant in the development process. The user gives continuous feedback, which results in updates in requirements and new requirements [17 - 19].

1.4.2. Developing AR-VLE using Agile Scrum

The Scrum methodology is the most popular Agile Development practice. Scrum uses sprints/iterations of 1-4 weeks to develop systems in small increments, delivering a product after each sprint. Augmented Reality systems could be split into modules, each module (or several modules depending on the amount of work required) worked on during a sprint and deployed afterwards.

The team (scrum team) during the development of an augmented reality system for a virtual learning environment will clarify the requirements and choose the technology stack to be used in the development of the system. The process is part of the product and sprint backlog in preparation for the start of the sprint. During the sprint session, the team creates the 3D models and image targets to be used in the AR engine. They also work on the prototypes for the module and develop the AR module to be deployed on a smart mobile device. The module is demonstrated, and a review session is held to take feedback from the various stakeholders. The feedback leads to updates to meet new requirements, which are considered during the subsequent sprint, and new modules are developed (Fig. **3**).

1.5. AR SYSTEM DEVELOPMENT FOR VLE

Several software tools can be used to develop augmented reality modules for a virtual learning environment. The development is broken down into two sections – designing the 3D objects/models and developing the AR module with the AR engine.

1.5.1. Designing the 3D Models

There are lots of 3D modelling software available, some paid and some for free (Table **1**). The software runs on various platforms – Windows, macOS, Linux and

the online versions running on the browser. Each of them offers certain unique features and user experiences to meet the needs of different designers.

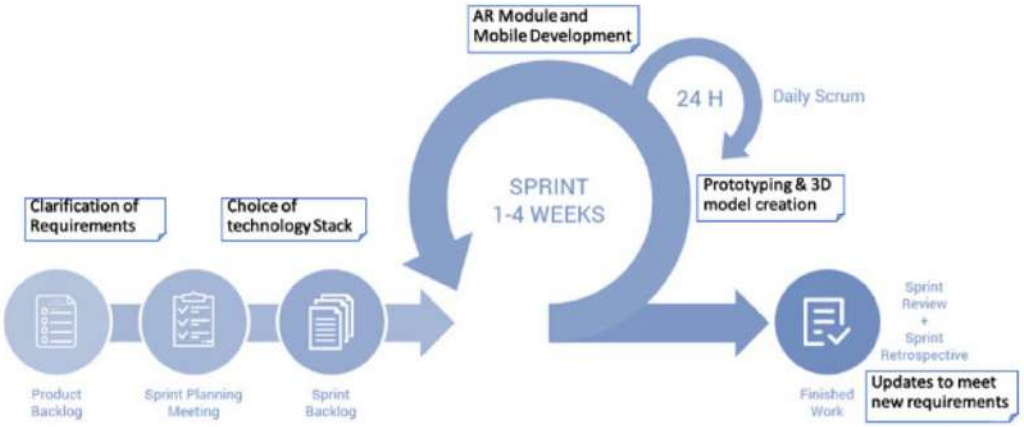

Fig. (1.3). Developing an AR application using Agile Scrum.

Table 1.1. Examples of 3D Modelling Software.

Paid	Free
Autodesk Maya	Blender
Houdini	Daz Studio
Cinema 4D	SketchUp
ZBrush	Sculptris
Rhinoceros	Vue

With modelling, lighting, rendering and texturing features, Autodesk Maya is highly seen as an industry standard. Houdini uses a node-based procedural production format, which is quite distinctive from Autodesk Maya. Users are offered a good amount of control over the Houdini tools. Cinema 4D is easy to learn; it is stable and boasts volumetric modelling. It is helpful in motion graphics, visualisation and illustration. One of the tools suitable for 3D printing is ZBrush, a sculpting and modelling tool which creates UV maps and paint textures. Rhinoceros has learnable features with a good rendering engine for complex animations.

Vue has features that create incredible 3D landscapes. It also contains camera lens distortions, depth of field and anti-aliasing strategies. Sculptris produces excellent models and usually integrates with other 3D software. It is mainly used for sculpting. One of the famous 3D modellers that run in the browser is SketchUp,

which allows importing 3D models into the project. The models are from the manufacturer and are also user-generated. Daz Studio specialises in creating digital art using virtual people, vehicles, people, the environment and other assets and accessories. These tools contain unique features that designers can use based on the projects being worked on, the platform being used, and the budget.

BLENDER

For demonstration, Blender is used to design the 3D models for this study. Blender is a free and open-source software that runs on most operating systems. It comes along with the main features expected in a modelling software – 3D modelling, UV wrapping, raster graphics editing, sculpting, animating, motion graphics and rendering. It has a solid rendering engine. Blender is an open source with a community of developers who continuously improve the software and its functionality. The figures (Figs. **4-6**) demonstrate how Blender designs 3D models of falling cubes.

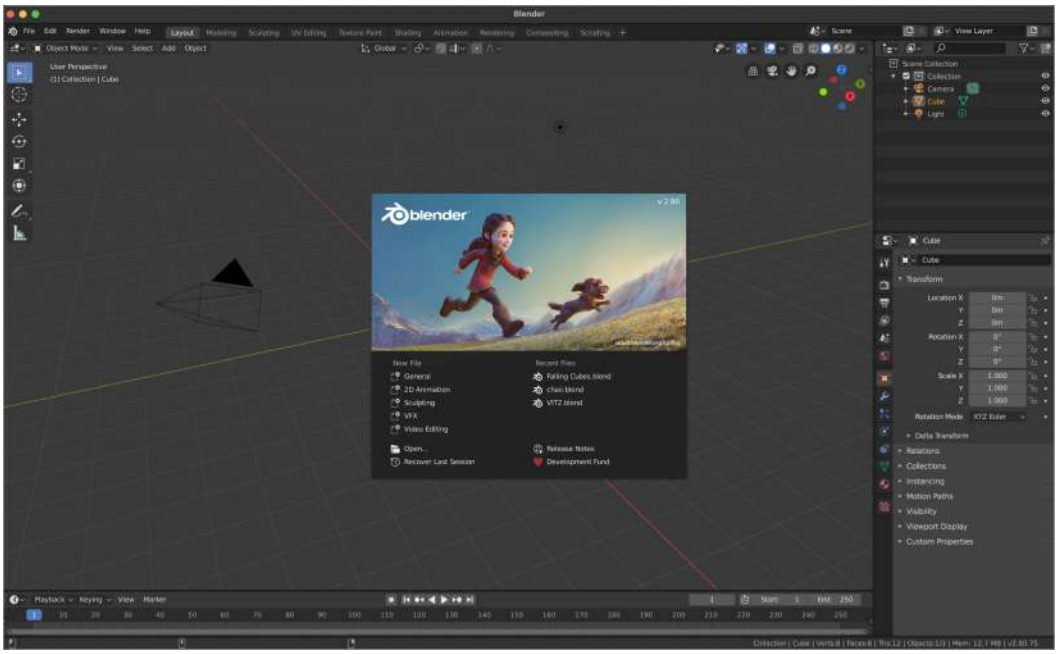

Fig. (1.4). The Interface of Blender.

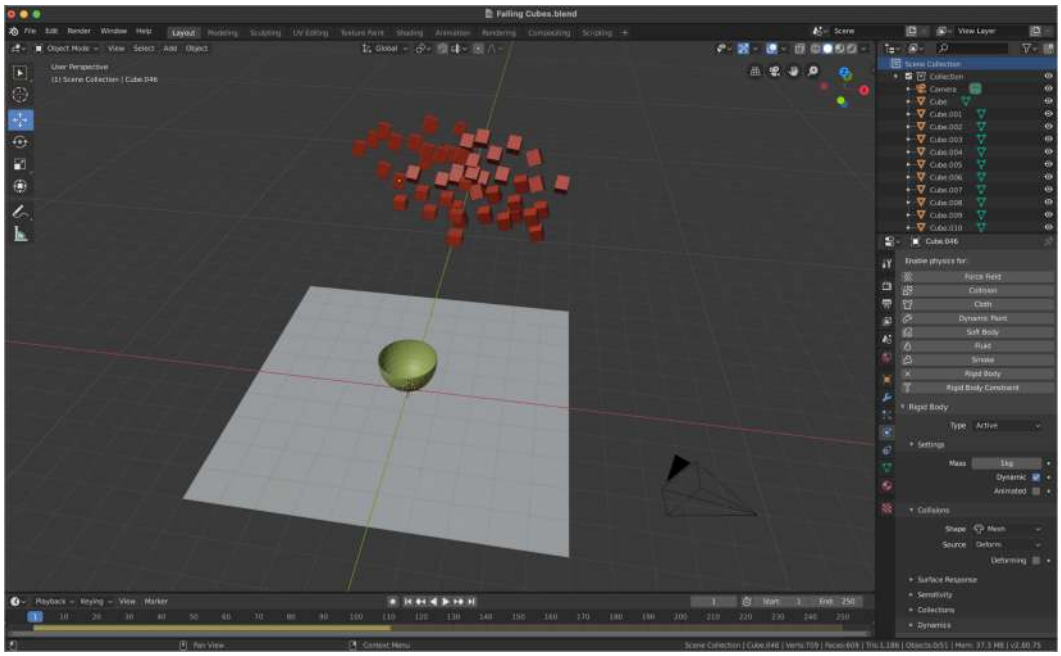

Fig. (1.5). A design of 3D cubes hanging at the top using Blender.

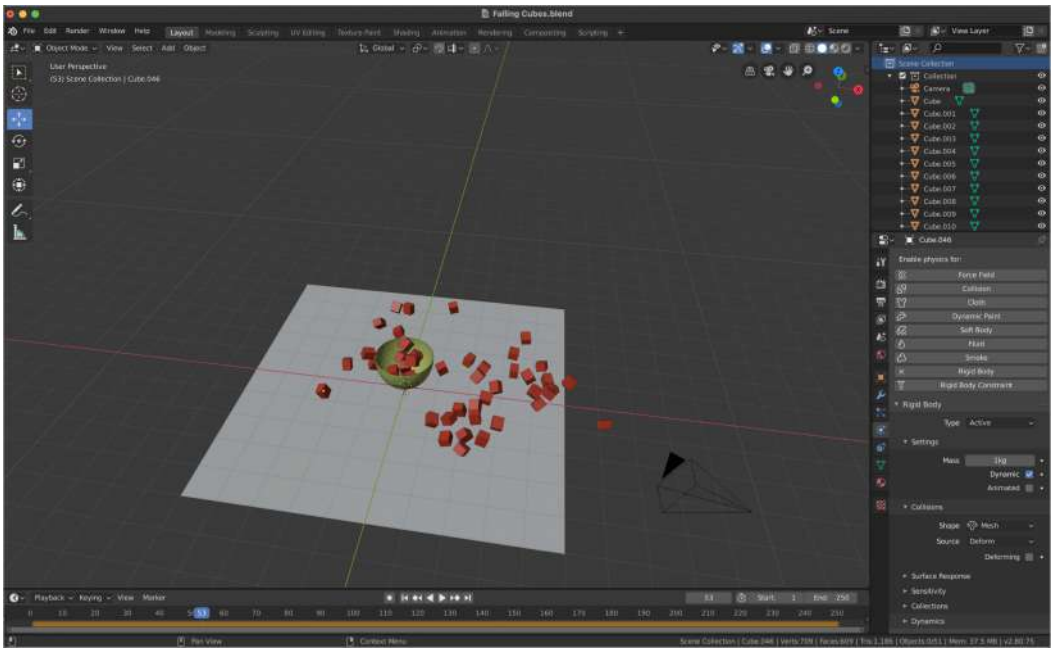

Fig. (1.6). Animated Design of 3D cubes falling using Blender.

1.5.2. Developing the AR Modules

Augmented reality developers and professionals use development software and game engines such as ARKit, ARCore, Unity, Vuforia Engine, HP Reveal, and Amazon Sumerian. The software integrates with various devices, especially smartphones, tablets, and headsets.

UNITY 3D

For demonstration, Unity 3D and the Vuforia Engine are used to develop the augmented reality modules for this study. Unity is a game engine used to create 2D, 3D, augmented reality and virtual reality games and applications across several platforms and operating systems. Unity integrates well with C# to write scripts that bring added functionality and enhancement to the Unity scenes. Unity also has an asset store where users create their models and assets and make them available to other creators for sale or for free.

Unity integrates with the Vuforia augmented reality software development kit to create image targets that are used as markers to detect images and the position and angles of the objects in correspondence with the objects in the real-world physical environment. The figures (Figs. **7-10**) demonstrate how Unity designs an AR module showing extra book information using the book cover as an image target.

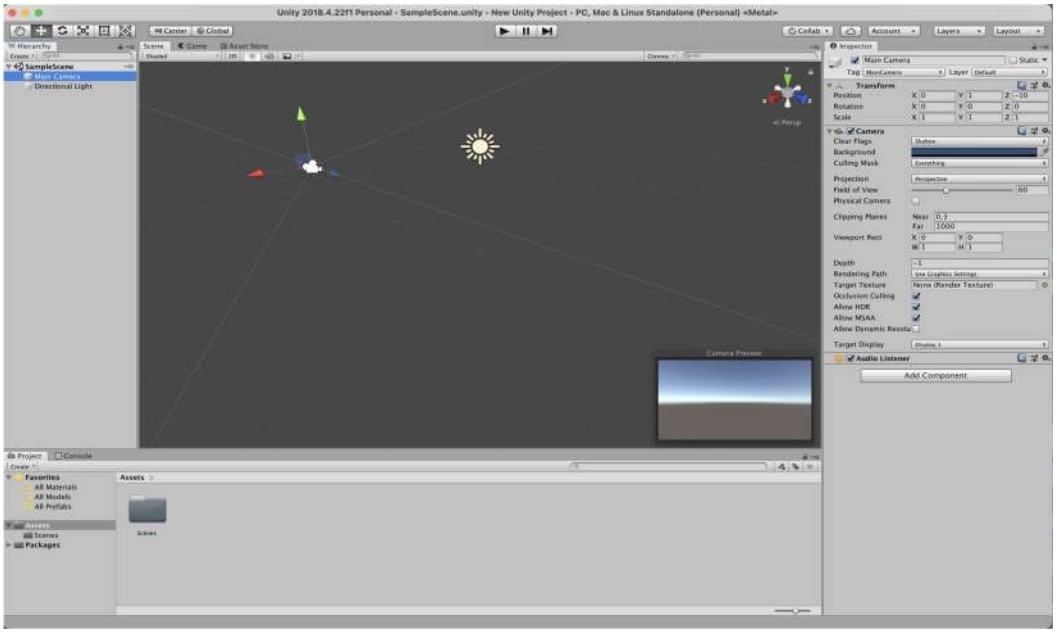

Fig. (1.7). The Interface of Unity.

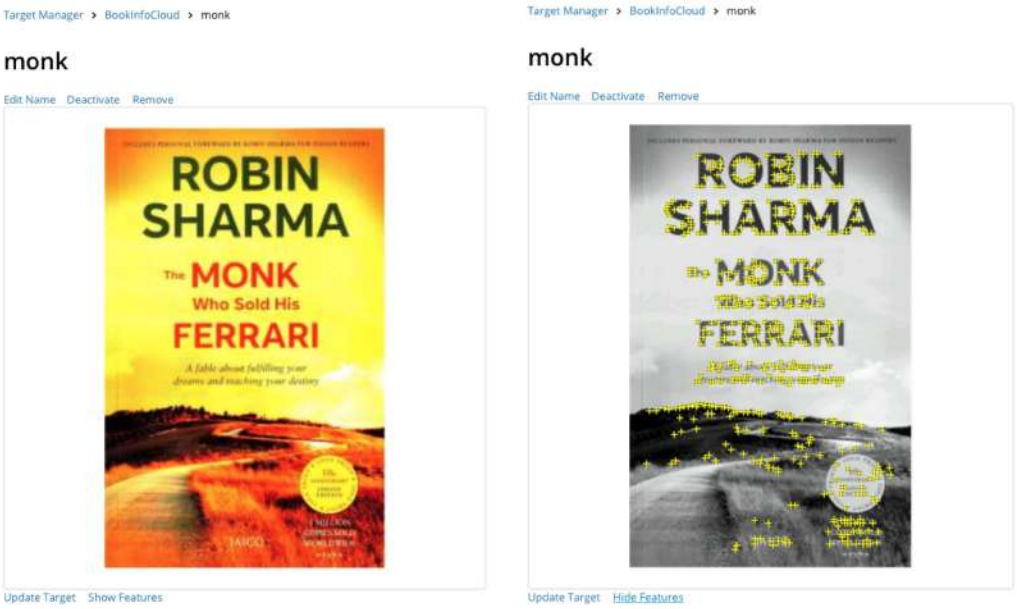

Fig. (1.8). Image Target in Vuforia showing marker features.

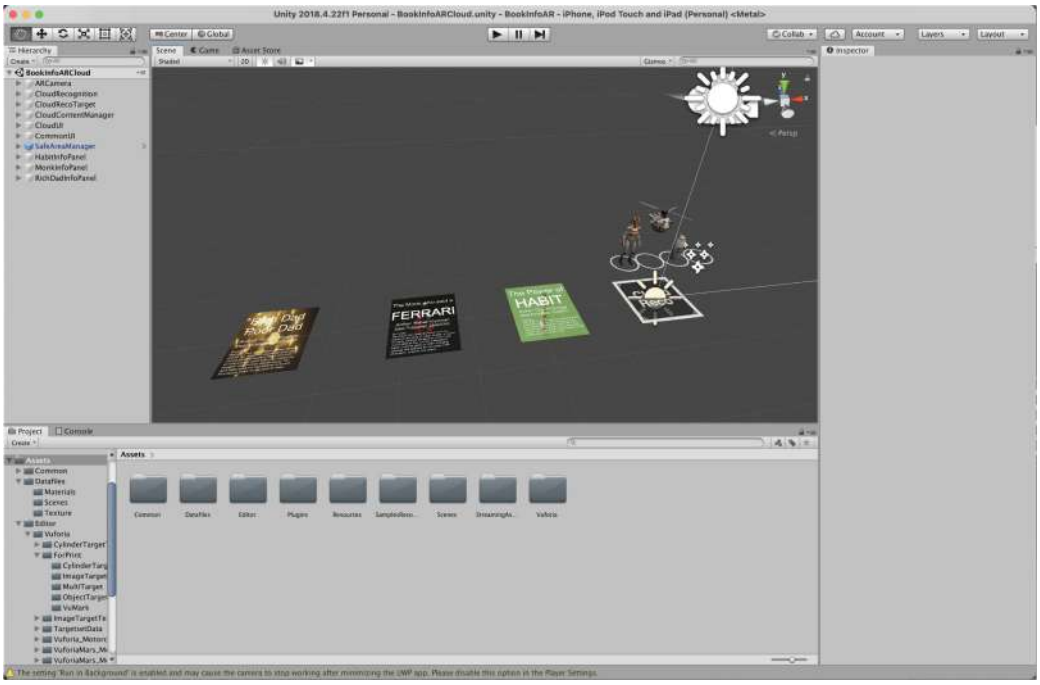

Fig. (1.9). Developing an AR module scene in Unity.

Fig. (1.10). Testing of a scene using the image target in Unity.

1.6. EDUCATIONAL DATA MINING IN AR-VLE

In general, data mining involves collecting data from various sources – databases, flat files, data warehouses and performing analysis, making statistical predictions and inferences. A review study [20] identified four stages in data mining projects to analyse and identify patterns in datasets. The process begins with the data cleaning and filtering phase; the necessary variables are then identified and selected. The data then obtains the relevant knowledge from the dataset and provides added meaning by interpreting and appraising the derived information.

Educational Data Mining mainly involves gathering data related to the activities of students within a learning environment and making inferences to identify ways to improve learning and other related activities. Such data is collected from various sources in a virtual learning environment (Fig. **11**).

In an augmented reality virtual learning environment, students with similar profiles could be categorised and given specialised themes in the learning environment. Lesson materials will be provided in the format that each student group will identify and associate with, but the main content remains the same irrespective of the format.

The system will also have functionalities for feedback and evaluations to collect students' suggestions on their learning performance and the instructor's performance (which could be anonymous) and system.

Fig. (1.11). Data Sources in VLE.

Each student enrolled in the system will have a learning path to track their progression. The learning path will be made available before a new learning session is initiated. It shows which sessions have been completed and which ones are yet to be started.

The system should have personalisation features that allow students to retake specific completed sessions when they feel the preferred accomplishment level was not attained and want to improve.

Based on the comprehensive data available, the system would have several recommendation functionalities to help students make learning choices. Recommendations would be made based on student profiles, test scores and other monitored activities on the system. Students could switch between different learning formats, retake sessions or complete trials and self-tests [21].

CONCLUSION

This chapter has presented an overview of designing and developing an augmented reality system in a virtual learning environment to increase engagement to learning materials and interactions among students and instructors. The development incorporates the agile methodology that breaks down the system development process into small increments called sprints to build usable modules. The methodology is also customer-centric, which plays a vital role in developing augmented reality systems and smart learning environments to involve the user in the whole development process and provide continuous user feedback even after the system has been deployed and in use. The software tools used in designing 3D models and developing augmented reality modules are also discussed. The chapter ends with the techniques of educational data mining that could be integrated into the augmented reality virtual learning environment.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] L.B. Kiat, M.B. Ali, N.D.A. Halim, and H.B. Ibrahim, "Augmented Reality, Virtual Learning Environment and Mobile Learning in education: A comparison", *2016 IEEE Conf. e-Learning, e-Management e-Services, IC3e 2016,* 2017pp. 23-28 [http://dx.doi.org/10.1109/IC3e.2016.8009034]

[2] A.C. Beluce, and K.L. De Oliveira, "Students' motivation for learning in virtual learning environments", *Paideia,* vol. 25, no. 60, pp. 105-113, 2015. [http://dx.doi.org/10.1590/1982-43272560201513]

[3] F. Ke, S. Lee, and X. Xu, "Teaching training in a mixed-reality integrated learning environment", *Comput. Human Behav.,* vol. 62, pp. 212-220, 2016.
[http://dx.doi.org/10.1016/j.chb.2016.03.094]

[4] F. Annansingh, "Mind the gap: Cognitive active learning in virtual learning environment perception of instructors and students", *Educ. Inf. Technol.,* vol. 24, no. 6, pp. 3669-3688, 2019.
[http://dx.doi.org/10.1007/s10639-019-09949-5]

[5] G-J. Hwang, H-C. Chu, C. Yin, and H. Ogata, "Transforming the educational settings: innovative designs and applications of learning technologies and learning environments", *Interact. Learn. Environ.,* vol. 23, no. 2, pp. 127-129, 2015.
[http://dx.doi.org/10.1080/10494820.2014.998863]

[6] Z-T. Zhu, M-H. Yu, and P. Riezebos, "A research framework of smart education", *Smart Learn. Environ.,* vol. 3, no. 1, p. 4, 2016.
[http://dx.doi.org/10.1186/s40561-016-0026-2]

[7] B. Gros, "The design of smart educational environments", *Smart Learn. Environ.,* vol. 3, no. 1, p. 15, 2016.
[http://dx.doi.org/10.1186/s40561-016-0039-x]

[8] O. GÜLER and İ. YÜCEDAĞ, "Developing an CNC lathe augmented reality application for industrial maintanance training", *2018 2nd Int. Symp. Multidiscip. Stud. Innov. Technol,* 2018

[9] J. Linowes, and K. Babilinski, *Augmented reality for Developers.* Packt Publishing Ltd.: Birmingham, UK, 2017.

[10] E. Cruz, "An augmented reality application for improving shopping experience in large retail stores", *Virtual Real. (Walth. Cross),* no. 0123456789, pp. 1-11, 2018.
[http://dx.doi.org/10.1007/s10055-018-0338-3]

[11] H-K. Wu, S.W-Y. Lee, H-Y. Chang, and J-C. Liang, "Current status, opportunities and challenges of augmented reality in education", *Comput. Educ.,* vol. 62, pp. 41-49, 2013.
[http://dx.doi.org/10.1016/j.compedu.2012.10.024]

[12] P. Reuksupasompon, M. Aruncharathorn, and S. Vittayakorn, "AR Development for Room Design", *Proceeding 2018 15th Int. Jt. Conf. Comput. Sci. Softw. Eng. JCSSE 2018,* 2018pp. 1-6
[http://dx.doi.org/10.1109/JCSSE.2018.8457343]

[13] Y. Yang, Q. Liu, L. Wu, S. Xu, S. Yu, and N. Zhang, "Design and Development of Mobile Augmented Reality for Mathematical Experiments", *2019 Int. Symp. Educ. Technol.,* 2019pp. 139-143
[http://dx.doi.org/10.1109/ISET.2019.00037]

[14] D.H. Youm, S.H. Seo, and J.Y. Kim, "Design and development methodologies of Kkongalmon, a location-based augmented reality game using mobile geographic information", *EURASIP J. Image Video Process.,* vol. 2019, no. 1, pp. 1-11, 2019.
[http://dx.doi.org/10.1186/s13640-018-0395-2]

[15] A. Protopsaltis, M. Mentzelopoulos, J. Ferguson, and K. Kaloyan, "“Quiz Cube: An AR mobile learning application,” Proc. - 11th Int", *Work. Semant. Soc. Media Adapt. Pers. SMAP,* vol. 2016, pp. 151-155, 2016.
[http://dx.doi.org/10.1109/SMAP.2016.7753401]

[16] J.M. Mota, I. Ruiz-Rube, J.M. Dodero, and I. Arnedillo-Sánchez, "Augmented reality mobile app development for all", *Comput. Electr. Eng.,* vol. 65, pp. 250-260, 2018.
[http://dx.doi.org/10.1016/j.compeleceng.2017.08.025]

[17] R. Hoda, N. Salleh, and J. Grundy, "The Rise and Evolution of Agile Software Development", *IEEE Softw.,* vol. 35, no. 5, pp. 58-63, 2018.
[http://dx.doi.org/10.1109/MS.2018.290111318]

[18] F. Mattioli, D. Caetano, A. Cardoso, and E. Lamounier, "On the Agile Development of Virtual Reality

Systems", *Int. Conf. Softw. Eng. Res. Pract.,* 2015pp. 10-16

[19] J. Shore, and S. Warden, The Art of Agile Development. 2007.

[20] A.V. Manjarres, L.G.M. Sandoval, and M.J.S. Suárez, "Data mining techniques applied in educational environments: Literature review", *Digit. Educ. Rev,* no. 33, pp. 235-266, 2018. [http://dx.doi.org/10.1344/der.2018.33.235-266]

[21] P. Ducange, R. Pecori, L. Sarti, and M. Vecchio, "Educational big data mining: How to enhance virtual learning environments", *Adv. Intell. Syst. Comput.,* vol. 527, pp. 681-690, 2017. [http://dx.doi.org/10.1007/978-3-319-47364-2_66]

CHAPTER 2

Brain and Computer Interface

Kuldeep Singh Kaswan[1,*] and **Jagjit Singh Dhatterwal**[2]

[1] *School of Computing Science and Engineering, Galgotias University, Greater Noida, India*

[2] *Department of Computer Sciences & Applications, PDM University, Bahadurgarh, Haryana, India*

Abstract: Brain-computer interfaces (BCIs) are defined as the science and technology of devices and systems responding to neural processes in the brain that generate motor movements and to cognitive processes (*e.g*., memory) that modify motor movements. Advances in neuroscience, computational technology, component miniaturization, the biocompatibility of materials, and sensor technology have led to the much-improved feasibility of useful BCIs. Brain-Computer Interface can be developed by engineers, neuroscientists, physical scientists, and behavioral and social scientists as a team effort. A study on brain computers (BCI) discusses how the brain and external systems interact. In intrusive systems, electrodes are implanted in the cortex; in non-invasive systems, they are mounted on the scalp and use electroencephalography or electrocorticography to monitor neuronal activity. The BCI systems can be generally ranked based on the location of the electrodes used for detecting and measuring neurons in the brain. This WTEC report was intended to compile and reveal to government decision-makers and the scientific community the information on global developments and patterns in BCI research. The design of hardware, device architecture, functional electrical stimulation, non-invasive systems of communication, academic and industrial cognitive and emotional neuroprosthesis has been discussed in this chapter. The purpose of the present chapter is to review the current sensor technologies used for invasive and non-invasive BCI approaches throughout North America, Europe, and Asia. We have visited and/or interacted with key laboratories with expertise in these areas. Although not completely comprehensive, this chapter gives an overview of the major sensor technologies being developed for potential BCI applications.

Keywords: Cortex, CNS tissue, ECoG strips, Electrode cap, Electrocorticography, Emotional neuro-prostheses, EEG signals, Etching technique, Geometric electrodes, Invasive, Integrated chip, Micro-heater, Multi-electrodes, Non-invasive, Neurosurgical technology, Neuronal map, Polyimide microelectrode, Sensor technology, Sensor Technology, Silicon-based Electrodes.

[*] **Corresponding author Kuldeep Singh Kaswan:** School of Computing Science and Engineering, Galgotias University, Greater Noida, India; Tel: +919467247612; E-mail: kaswankuldeep@gmail.com

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

2.1. INTRODUCTION

This chapter offers a description of the sensors in the brain-computer interface technology (BCI) data set. We divide sensor technology into two fundamental categories for this chapter. First of all, we speak about "invasive" technologies, including brain implantation surgical techniques involving mostly the multi-electrode recording of microelectrode arrays inserted directly into the brain to test single-cell action potential. The key subject of this chapter is a big growth market for sensor technology [1]. We should point out that this technology has not been approved for the trials on human beings [2]. Furthermore, observations of subduing or epidural stripes of electrode arrays that monitor cortical potential [3], which is somewhat similar to EEG recordings on the skull surface, will be addressed, as this is the main use of these intrusive electrodes for human epilepsy surgery [4]. However, the development of other BCI applications may be improved. Secondly, the "nonspecific" technologies are discussed, involving mainly "Wet" silver (Ag) or gold (Au) multi-electrode EEG registration arrays of pull electrodes that are mounted on the surface of the crane to monitor EEG operation. These electrodes come from a variety of sources online, but relatively little development has occurred in this field [5]. We warn that "non-invasive," scalp-invasive, BCI-technology applications are often used acutely and can become more invasive for people at home or at work. Further advancement of technologies in this field will be addressed briefly.

2.2. WHAT IS BCI?

The term "BCI" has been popularized and was first published by UCLA professor Jacques Vidal. Vidal is well known in peer-reviewed arts as the scientist of BCIs [6]. A brain-computer interface is an effective communication interaction between the external device and the central nervous system, also known as a mind-machine interface [7], that circumvents the use of encapsulation. This travels through the cortex straight to the device instead of from the cortex to the finger on the keyboard by the musculoskeletal mechanism [8].

Brain interface devices shown in these times need deliberate thinking, although potential implementations will work smoothly [9]. sThe old method to BCI, which required the implantation of a mechanical device in the brain, and it seemed to monitor as a genetic component of the body, is being replaced by current, in-depth information on non-invasive BCI. BCIs are built to increase the sensory-motor or human cognitive processes, support them or restore them. This incorporates electrical engineering, information science, biomedical and neurosurgical technology [10].

2.3. BCI SENSOR WORLD OVERVIEW

The bulk of European BCI sciences include “non-invasive” sensor technology [11], *i.e*., multi-electrode recording from EEG electrode arrays on the skull floor [12, 13]. This sensor technology has seen very little development and needs to be greatly improved [14]. In Europe, several BCI sites are capable of delivering sensing devices, which may help advance the production of “invasive” sensor technology [15].

Brain-machine interfaces (BMI) [18], while they are designed to solve the same problem, are substantially different from BCIs, expressing the purpose of a subject in robotic orders [16]. BCI acts in a manner that is understood to equate with actions but is diffusing and unspecified with the macroscopic brain function (mostly EEG) [17]. BCI has already achieved progress and is ready for patients to use, building on accessible EEG research and machine learning techniques details [19, 20].

2.4. HISTORY OF IMPLANTABLE ELECTRODES

CNS history of implantation of electrodes dates back to Hess' early work with original feline implants in the 1930s' [21]. In the 1970s, however, implantable electrodes saw a more recent adaptation. The development of 50 μm tungsten microwire arrays was recorded by Selman and Bach in the early 1970s in the electrophysiological records and the early 1980ies by Chapin & Woodward (1986). In essence, many laboratories use this type of technology for more frequent single-unit recordings and many animal BCI applications. The accurate monitoring and functionality of individual wires for electrode recording sites are some of the difficulties with multi-wire arrays [22].

Wise and Angell developed the principle of the use of integrated chip (IC) technology to improve the microelectrodes from 1970 to 1975 [23]. In the next several years, various articles were written; Bement Standard wire electrode works were ground breaking in the 1980s, varying in diameter, with a duration exposed of up to 1 mm, from 13 to 200 μm [24]. Fig. (**2.1**) provides an illustration of a high-density array of embedded micro disk for recording up to 128 wires in spontaneously movement mice. Wire electrode is widely used by rats, primates, animals, and late monitoring purposes [25].

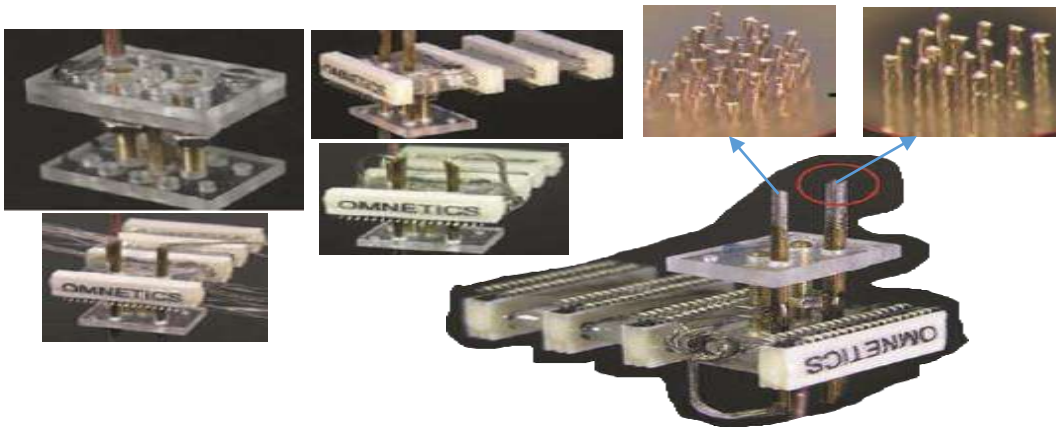

Fig. (2.1). Creating a Microdrive for Mice for a High-Density Ensemble.

For some factors, conventional microelectrodes of the wire type are still commonly used [26]. They can mainly be ordered from different manufacturers or manufactured from available commercial Substances [27]. Secondly, very small microelectrodes can be produced [28]. Thirdly, they are set up on the field. But the drawbacks are conventional wire microelectrodes. As they are handmade, large heterogeneity may be accomplished between individual geometric microelectrodes [29]. Ground surfaces can be influenced by defects in the cutting tip and the connection between the metal and the isolated substance that can result in altering reaction characteristics. Due to the supplies and resources needed as well as process of manufacturing, many laboratories have trouble integrating reproductive surface electrodes [30].

2.5. TYPES OF MICROELECTRODES

2.5.1. Mass-Fabricated Microelectrodes

Mass manufacturing of microelectrodes includes photolithographic methods used in the microcircuit industry there was a mistake [31, 32]. Structures of 5–10μm can now be recorded routinely and areas of 0,1-4μm will be developed in the future using photolithographs [33]. This is competing or exceeding one of the highest accuracies intra electrode shapes. For model features of 50–100μm, the least expensive graphics techniques, which do not require quite specific microelectronic features. Several microelectrodes are also available prototypes will simultaneously be modeled on the same substratum to simultaneously create a greater number of microelectrodes, which lowers manufacturing costs [34]. Besides, micromachining techniques can be used to create microelectrodes in very

well spatial relationships with numerous sites that can be utilized to document layered intellectual frameworks [35]. The micro-electrodes will respond to the configuration of the brain [36]. The enhanced output of mice can be reached by the manufacture of microelectrodes by specialists from the semiconductor industry while eliminating the expense of building internal production facilities [37]. The microelectrodes (*e.g.*, Thin Film Technologies, Inc.).

2.5.2. Silicon-Based Microelectrodes

Silicon was the very first substratum used to create multisite related semi-conductor microelectrodes and several studies of microelectrodes have been reported for brain monitoring and stimulating of neural tissue [38]. One of the attractive features of silicon as a substratum is the method of chemical etching [39]. Without the need for laser processing or sawing, independent microelectrodes may be produced from a single substratum at a time [40]. It is necessary to create small functions, such as canals in the substratum. The grading can create very thin microelectrodes to measure the thickness of the substratum [50]. Thin substrates have been recorded as thin as 6–15μm [40].

Doping can change the semiconductor characteristics of silicone. Silicious is also highly compliant with onboard circuits [41]. Silicon has various characteristics which have made it commonly used to shape microelectronic arrays [42]. Pictures of some of BCI's major contributions to microelectrode technology focused on silicone are seen in Fig. (**2.2**) of the "Center for Neural Communication Technology at the University of Michigan". There are two of the latest prototypes for rats and nonhuman primates for BCI applications [43].

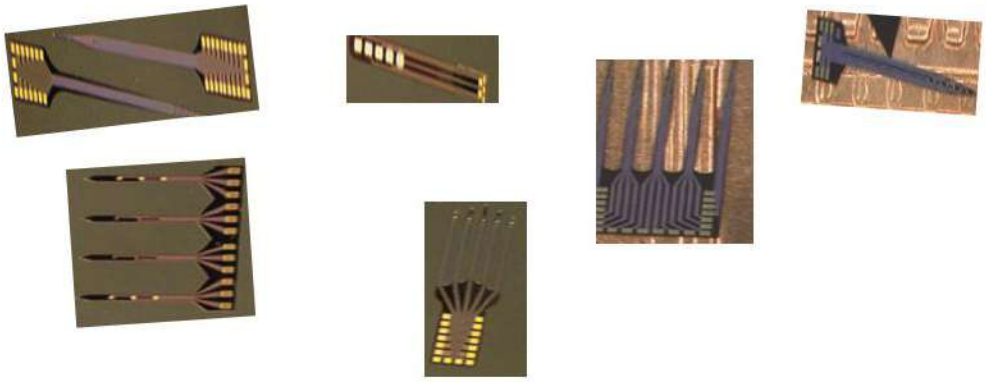

Fig. (2.2). Silicon-based microelectrode array photomicrographs developed at Michigan University.

Doping can change the semiconductor characteristics of silicone. Also, there are many flexible architectures that this method reveals in this grouping of

microelectronics. Silicon has one of the huge perks as a base medium is the choice of chemical grazing procedures. The thickness and shape of the microelectrode may be adjusted through grafting [44]. It is used for diluting the substrate with an isotropic etchant (10% hydrofluoric acid, 90% nitric acid) [45]. The etching of the silicon substrate is isolated by an etch of ethylene, diamine, and pyrocatechol water (EDP) [46]. The intended type of microelectrode can be described by a layer of silicon nitride modeled on the silicon wafer [47]. Silicon nitride prohibits the substrate from reacting. Also, a graft can be avoided by injecting the substrate with boron selectively [48].

These are also the largest production potential of microelectrodes in the European Union, which is highly competitive with technology developed in the United States and Asia [49]. The indicative models are shown in Fig. (**2.3**). New systems with semiconductor electrodes placed can be embedded in the sensors. Holes were inserted into the substratum for the microelectrode securement and even stronger incorporation into the outer membrane brain space [50]. During electro physiological records, several channels for the transport of chemically / medicines have been graded into the substratum of the silicone stoned (see Fig. **2.4**).

Fig. (2.3). "Examples (Top-left) of silicon-based ACREO arrays; micrograph of different site; (Bottom) series of the Datatypes binary formats schema.

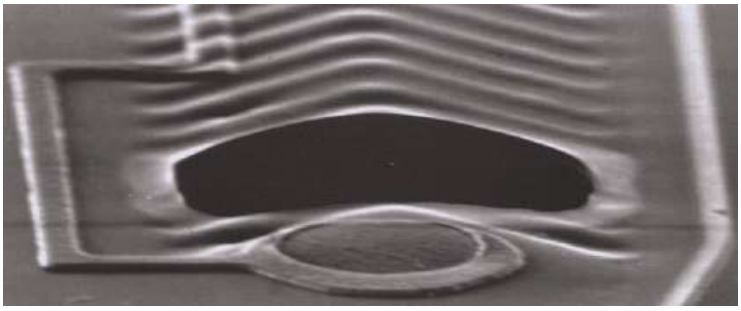

Fig. (2.4). SEM of microchannels for the transmission of chemical substances into CNS tissue on a silicone-based microelectrode. Michigan Sound images supported by the NIH/NCRR Resource Hub at the University of Michigan; replicated with authorization of the Encyclopedia of Sensing.

Microelectrode arrays contain combined Ag/AgCl comparative electrodes. Micro-drives for local adaptation after installation in the microelectrode design. An optimized polysilicon microheater was designed [51]. Using VLSI chips in the silicon substrate (see Fig. **2.5** with embedded amplification) can result in the electrode modulation and signal method. Microelectrodes basis on silicone allowed the manufacture of "hybrid" designs of the microelectrode.

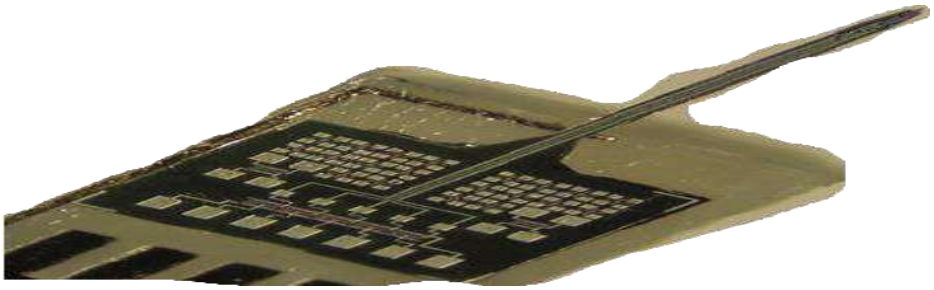

Fig. (2.5). "A silicone-based electrophysiological image with an amplifying chip is seen for lectrophysiological capture (photograph provided by Sung June Kim of Inter-University Semiconductor Research Center at Seoul National University, Korea; reprinted with permission from Encyclopedia of Sensors".

An interface for prosthesis has been created for the pioneering work of Norman, Donoghue, and coworkers to be set up in electrophysiological arrays at 100 tracking sites.

These designs are still used for human uses and comprise the very first BCI [52]. vaccinated, non-human, and human primate research arrays of microelectrodes. Independent "shafts" of the microelectrodes stretch 1, 5 mm of the 10 to 10 mm substratum [53]. The tips of the shaft are metalized by pt on silicon to lead down the shaft. The silicon conveyor is isolated with silicone and glass nitride. An SEM of a "Utah" electrode is shown in Fig. (**2.6**). Several planar silicones multi shank

micro-samples as seen in Fig. (**2.7**), can be used to create identical three-dimensional microelectrode arrays [54]. Flat microelectrode arrays were used to map neuronal connectivity for the brain-slice recordings [55].

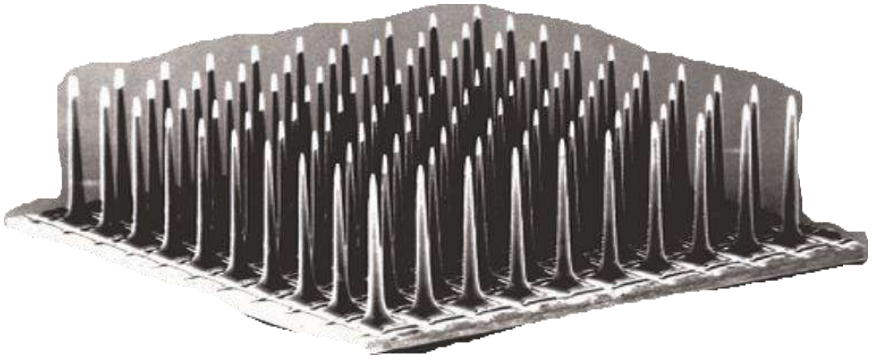

Fig. (2.6). "SEM for the optical prosthesis from Utah Electrode Array (UEA).

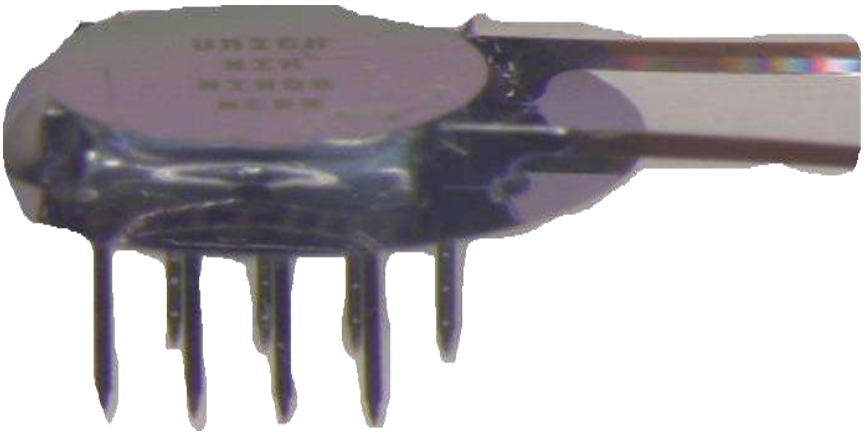

Fig. (2.7). "A multichannel photomicrograph generated with several silicon-based micro electric electrodes.

2.5.3. Ceramic-Based Microelectrodes

Specific microelectrodes should be sliced Manually from the wafer as the pottery does not agree with standard etching techniques [56]. Laser work is the most flexible way of cutting bulk wafers' microelectrodes that allow the production of complex types [57]. But laser machining can create rough edges due to stepping off the laser which can cause possible issues with microelectrode insertion [58]. A diamond saw, which glitters when it slices, will render several smoother microelectrode boundaries, thereby preventing excessive tissue injury. The intact brain biology must be tested with minimum CNS tissue injury. It is harder to form complicated forms by using a diamond saw because sciaws are usually cut into clear lines [59]. Image 1 of Fig. (**2.8**) is a picture of a complicated laser machining

microelectrode design. Image 2 of Fig. (**2.8b**) is a basic microelectrode ceramic substrate created by a diamond saw operated by a PC. Image 3 of Fig. (**2.8**) represents an expanded smoother edge of this microelectrode. The use of lasers excimer can have cleaner edges than typical laser processing. The polishing of the ceramic substratum will accomplish thinner microelectrodes [60].

Fig. (2.8). (**1**) image of the complicated microelectrode shape of a ceramic substratum, cut by laser processing; (**2**) the less complex microelectrode form created by the diamond screen operated by computing; (**3**) a smoother edge amplification of the microelectrode of Sensors.

Fig. (**2.9**) reveals ceramic-mounted microelectrodes media coated with a thickness of about 38 and 51μm with a tip diameter of 60μm. Apply ion reflector deposition to the isolating alumina layer. These 20 to 80μm platinum recording sites with a range of 200μm were used *in vivo* recording for up to 24 weeks. Several Pt microelectrodes have been produced at four and five sites on ceramic substrates. Fig. (**2.10**) shows the versatility of lithographic techniques [61]. Usually, the capturing sites are grouped or linearly isolated, side-by-side. On a linear basis device similar to that mentioned previously, two recent designs configure the microelectrodes for 50μm / 50μm for each. The newest versions have large 50 to 100 Pt sites and 50 to 150μm to decide if greater sites will report better potential behaviors or lower CH detection limits [62].

Fig. (2.9). Using an ion beam-aided deposition, the photomicrograph of a ceramic-based microelectrode built on thinner substratum alumina is added.

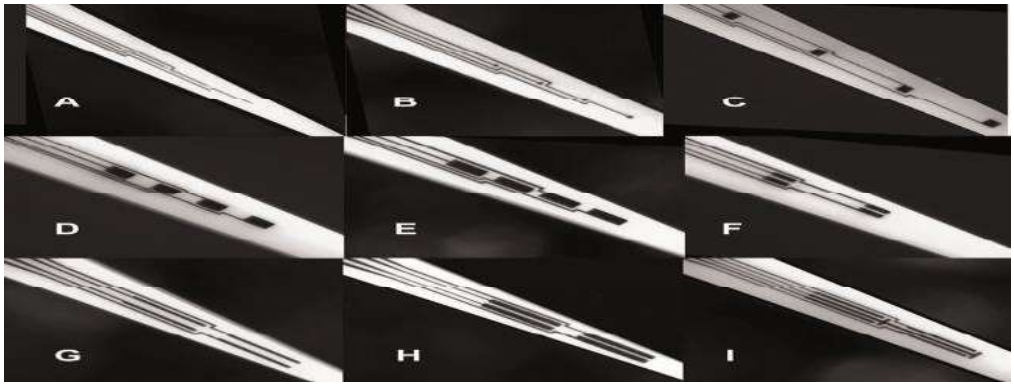

Fig. (2.10). Photographs of many microelectrode projects based on ceramics.

Fig. (**2.11**) displays several prototypes for "conformal" 8-site microelectrodes, currently being produced in rats and monkeys for various brain regions. The electrodes can be selected according to brain area(s) and form of interest recording. To better understand thin transmembrane domains, for *e.g.*, on the cerebral cortex, or the pyramid cells of the hippocampus Purkinje cells, two or three grab sites are useful at the end of the microelectrode. In the field of concern of the brain, a large concentrate of monitoring sites at the tip can be obtained with several steps.

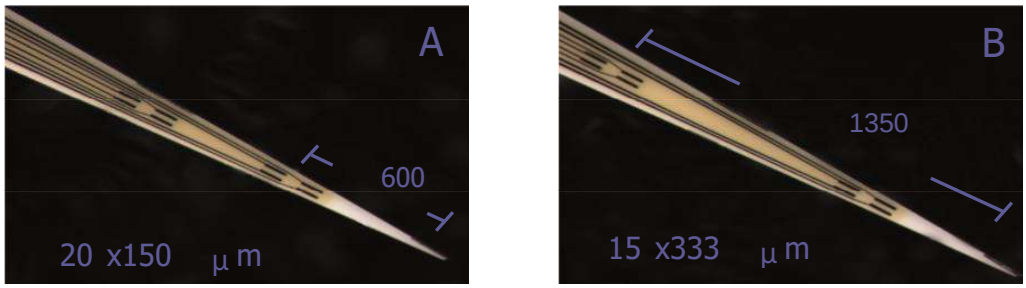

Fig. (2.11). Layouts of "conformal" ceramic microelectrodes with 8 recording points.

2.5.4. Polyimide Microelectrode

Microelectrodes less than 20µm thick were produced [64]. is polyimide particularly functionally flexible as a substratum? The image microgram of a 3-D multi-shank microelectrode for intracortical implantation is shown in Fig. (**2.12**).

Fig. (2.12). Image of an Intracortical Implantation polyimide-based microelectrode array.

While polyimide flexibility may make implantation complicated, in some cases a versatile microelectrode can lead to less tissue damage. Guiding incisions are also required in neural tissue to prevent a microelectrode shaft from hanging on the implantation of microelectrode (Fig. **2.13**).

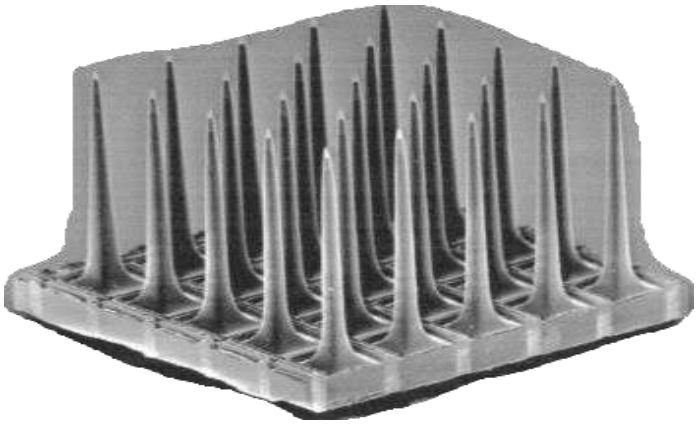

Fig. (2.13). "Extending of multiple recording locations with a permeability trough in a polyimide microelectrode to protect the tissue of the microelectrode.

2.5.5. Microelectrodes Connectors

Microelectrode is a significant issue in the manufacture of microelectrodes in conjunction with recording equipment. The microelectrode is also secured to a "paddle" or a PCB-holder [63]. Drives from the pads on the microelectrode are electrically connected to the operator to brakes on the plug. Metal lines usually pass on pins or some other connector in the length of the holder. It may also be inserted into sockets using dual-inline pins or zero-inserting power sockets (ZIF). with electronic equipment.

2.5.6. ECoG Strip Electrodes

The use of electrocorticogram (ECoG) recordings for BCI is the field of research. The Technology from EEG clinical recordings was developed in the 1930s and 1950s, through Jasper and Penfeld's work. Firstly, in thousands of people Technology protection has been extensively checked at least. Second, ECoG's spatial resolution is higher than EEG (a tenth of millimeters *vs.* centimeters) and current electrode prototypes (Fig. **2.14**) provide more temporal resolution than directly penetrated electrode data. Thirdly, the impulses from the brain surface display higher amplitudes for a larger bandwidth [64]. Fourth, in the United States and Europe patients undergoing epilepsy therapy occupy a significant testbed for the investigation of BCI technique. Finally, these validated innovations may have enhanced *in vivo* long-term reliability, but they are still in the range between 4 and 64 recording positions [65].

Fig. (2.14). ECoG subdural epidural recording microgrid.

Fig. (**2.15**) presents various Ad-Tech ECoG stream electrodes.

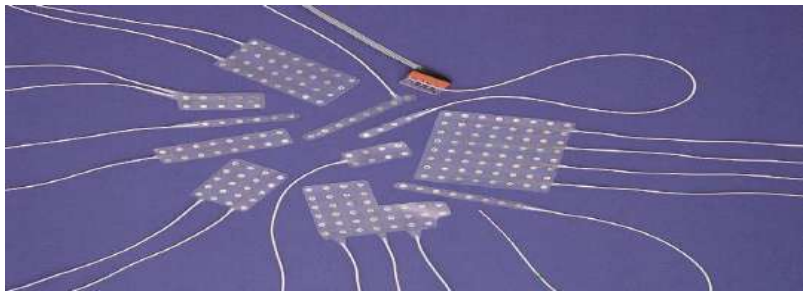

Fig. (2.15). Recorder electrode strip stripe detectors from 4 to 64 ECoG positions "(Reprinted with permission from Ad-Tech Medical Instruments)".

2.6. BCI EEG SENSORS

Almost all BCI studies require the use of skull-fixed Ag or Au disk electrodes with non-invasive sensors to allow the installation of the EEG electrodes using a certain type of head cap setup. Over the last two decades, there have been little advances in developing these instruments to easily and conveniently position them on the skull of a BCI patient. Head caps have been designed to support the calculation and positioning of the "International 10–20 grid system" EEG electrodes from 64 to 256. There are several head cap suppliers and electrodes which include G. Tech (Guger Technological OEG), Grass Technologies, BioSemi, etc. In the field of wet electrodes recording, as seen in Fig. (**2.16**) is One of the strongest caps' sources for a range of BCI technologies. Its special head cap for the configuration of EEG electrodes offers some of the industry's best signals to rumors. The electrode cap configuration takes extra time to fit electrodes but achieves outstanding signal-to-noise features. For other devices and amps as well as other vendors of such instruments, this extremely robust style can be used.

Fig. (2.16). Head Cap System for EEG.

The method of fitting people with EEG electron head caps, however, takes time, involves checking for their impedance by individual electrodes and leads to a painful or practical interface for regular use of BCI. "dry electrodes" need to be created that can be used without the requisite planning for the present designs. To increase the and leads to a painful or practical interface for regular use of BCI, active electrode architectures (such as those marketed by BioSemi) are also required.

2.7. MODELING AND SIGNAL PROCESSING OF BMI/BCI TECHNIQUES OF MULTI MICRO ELECTRODE ARRAY

Interfaces for here, brain robots display spike info structures. Multi microelectrode sequence data are used in two main methods: one class includes spikes; a second class uses binning spike rate estimations, *i.e*., the number of spikes over a while (the bin). BMI analyzes have binned frameworks controlled.

2.7.1. Binned Conceptual Data Models

BMI experiment framework provides strong computational methods and techniques for the analysis of the signal, used to extract optimized data models. Both the BMI feedback and the required answer have been synchronously generated by the researcher (hand position).

Precise and automatic spike identification and trials remain a continuous focus of study there was a mistake Subject to the point function is correctly extracted, the decoding model is the neuronal spike firing functionality. The problem of decoding can be converted into a device recognition paradigm where a linear or non-linear parametric device is specifically learned from the data obtained for proximity. In Fig. (**2.17**) control theory and signal processing models are widely researched, so there is a range of approaches that can be used.

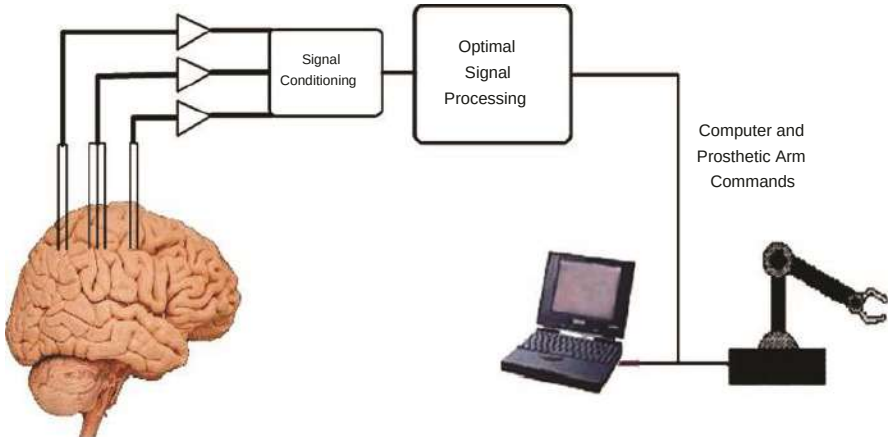

Fig. (2.17). System identification framework.

The particulate filter system eliminates both the Kalman filter's constraints (linearity and Gaussian assumption) but complicates computer algorithms considerably. The particle system utilizes the sequential reverse approximation at - phase as its counterpart. But since the model is usually unclosed, the posterior solution must be calculated by playing with it.

2.7.2. EEG/ECoG Recordings

In certain ways, EEG and ECoG signal processing are identical. The signal is a field in both situations arising from broad neuron activity. Signals can only be observed in both cases by synchronizing all of these neurons. The asynchronous operation should be discontinued to reflect the composition of synchronous neuronal activity. Two stages in BCI signal processing, the extraction and the conversion of features.

2.8. HARDWARE IMPLEMENTATION

2.8.1. Paralysis Patients Restoring Movement

Paralysis disease, affecting people internationally, affects several problems, include shock, strokes, infections & inflammatory disease. In the cortex, spinal cord, vertebral nerves, and muscles the main injury may be expressed. Generally speaking, "paralysis" means extreme or full engine operation failure whereas "paresis" is a comparatively small loss. A big concern is extreme paralysis, not only by sacrificing the capacity of patients to live normally but also by the massive cost of hospital maintenance. *E.g.*, in patients with TC, nearly all-volunteer motor functions are lost under the throat, but also somato-sensation, *i.e.*, their feelings of contact, discomfort, temperature, and the location of their limbs are lost. Extreme lateral amyotrophy (ELS) is further compounded by the fact that patients may lose their entire body's engine function.

2.8.2. EEG-Based Brain-Computer Interfaces

In current usage, there are two distinct types of BCIs, one being the "non-invasive" B CII, an electronic recordings device that uses scalp electrode contacts to capture electroencephalographic signals (EEG) from the cerebral cortex of the persons. The person typically needs biofeedback approaches to learn how to control a cursor on a computer screen using his own "brain waves." Use the BCI to guide the machine cursor to select alphabet letters after a lot of effort, the cursor's control becomes sufficiently precise to spell words. This BCI strategy has been incredibly popular and appears to be primed for an outstanding short-term solution to have some regulation over external equipment for serious SCI patients. The use of electronic techniques to recover the motor control in SCI is also shown. However, there are limitations in that subjects need significant preparation and rigorous focus. In comparison, the non-invasive BCI solution is limited by the number of freedom degrees regulated by EEG recordings (currently one to two).

2.8.3. Direct Brain-Computer Interfaces

The use of embedded Multielectrode collections to track directly within the brain from the engine control circuit is an alternative solution to EEG control BCI. In the past eight years, the viability of this direct BCI method was shown, starting with an animal and moving currently to humans (Hochberg *et al.*, 2006). This approach is used to collect the brain's engine orders since multi-Electrode arrays record from neuron populations in the engine cortex of the subject. The reason that the collections specifically touch on the underlying functions of the processing brain makes it possible for the registered signals to direct the driving of a BCI with the inner motor system [39].

2.8.4. Recording, Extracting, and Decoding Neural Motor Commands

The capability of a multi-electrode recording device to detect expected motion by sampling huge communities of neurons in motor cortices depends on all direct BCIs. The action potentials of each electrode are registered by multiple nearby neurons, each supplying detailed information about the expected course of travel. The same electrodes also record the amplitude of local range capacities aim of providing Details regarding the acceleration speed or strength. When synaptic impulses are captured in the brain, all of this information is automatically amplified, filtered, discriminated against, and transferred to a computer. This knowledge is then converted into a robot or machine cursor managed output format [49].

2.8.5. Predict Limb Movement Kinematics use of Multivariate Regression Analysis (MRA)

We considered MRA to be a robust method for transforming multi-neuron recordings into a continuous approximation of limb activity in a subject or a robot arm control. Although several approaches have been introduced to this issue, most multi-neuron managed devices still use MRA variants that produce linear models that estimate a variety of related variables (output) in an entity (entry) variables array. MRA is the method of mathematics choose between autonomous (*e.g.*, population neurons of the drive system) and a predictor variable to describe the linear mathematics association among (*e.g.*, a motor output function). MRA gives a linear structural equation in its simplest forms

$$\text{“}Y = a + b_1X_1 + b_2X_2 + ... + b_m X_m\text{”}$$

Where X is independent variable, Y is dependent variable, a is an offset and b is a weighting coefficient for independent and is a counter-set. The relationship is conveniently determined based on a sample data set by a linear less than a square

approximation, which measures a line along the data points seen to minimize the sum of the squared differences in the points shown along this line. This measure also results in the decision coefficient (R^2), which evaluates the superior predictive performance (in a range from 0 to 1).

2.8.6. Monkey Brain Motor Commands of Extraction

Next step was a description of the potential for neural knowledge incapable of encoding movements in many directions using Multi-Neuro Engine cortex recordings of primates. Monkeys were not only used to recognize the human cortex motor but also to create more space for the electrode implantation in their larger brains. Fig. (**2.18**) illustrates how the monkey took a handle and pushed it right and left for juice prizes randomly. Fig. (**2.18**) is used to construct this experiment. Multi-neuron records were concurrently obtained from electrode arrays 32-channel inserted in several regions of the brain, including the motor, premotor, somatosensory, and parietal brain cortices.

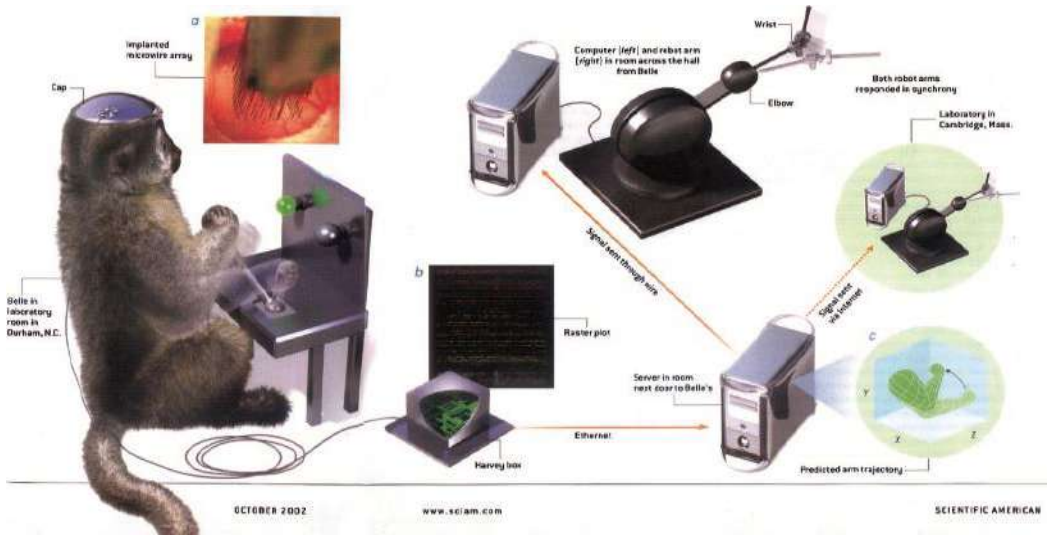

Fig. (2.18). Monkey Brain Motor Commands of Extraction.

Whereas in these recordings, the consistency and quantity of useful coding information differ from one sensorimotor to another, both areas Contributing to the legislation at least some useful information. This is similar to earlier findings, that cortical neurons are typically well balanced and can also engage in an experimental environment of a large range.

The broad tuning is the explanation for many effective multi-electrode recordings because the researcher will rely on obtaining valuable signals from the majority of

neurons captured. In the previous literature records, this was supposed to be broadly related to the planned course of activity of cortical motor neurons. For both naive and over-compressed owl primates, that was real. As a result, before movements in two separate ways, the same sample of Neurons exhibited intensified firing.

For 18 months we kept our cortical documents safe and workable, which was the first major finding of the possible application of this method in developing brain-controlled prosthetic instruments.

2.8.7. Biofeedback Changes Coding of Robot Arm Movement

Only then did most scientists confirm that the robot arm action disintegrates from a real arm after transitioning from arm power to robot brain management. Although the subject is only compensated for the exact operation It is less reliable with the time of the robot arm to adjust the actual arm.

In the meantime, the accuracy of monkeys normally goes on improving. In reality, the exact coding of a neuronal path was discovered by Carmena *et al.* (2003). This dissociation, though, tends to rely totally on the experimental background since the subjects usually use their actual weapons after returning to their cages. Since, animal subjects are clear about enhancing their control exactness, it forces one to conclude that human persons will also learn to alter the characteristics of their neurons.

2.9. BRAIN CONTROL OF MULTIPLE-OUTPUT FUNCTIONS

The next step will be to explore the use of this technique to monitor the various consequences. Brain control of gestures has already been certainly seen. Furthermore, the model can simulate all the Correlations forces of 0.9 since these linear models are based on data gathered from animals move toward two or even more forces. If this linear pattern consists of animal data moving against two or more different forces of force *i.e*., steady spring, the model shall therefore be conditioned will reliably simulate all forces (Fig. **2.19**).

The blue marks represent the animals and redwork. Marks created by neuronal behavior of the animal processed with a weight matrix built using a model of regression. The weights were calculated in conjunction with the desired variable of importance for the neuronal activity in previous experiments, in this case, the work. Note that the projected trajectories (red) have a high association between actual trajectories (blue).

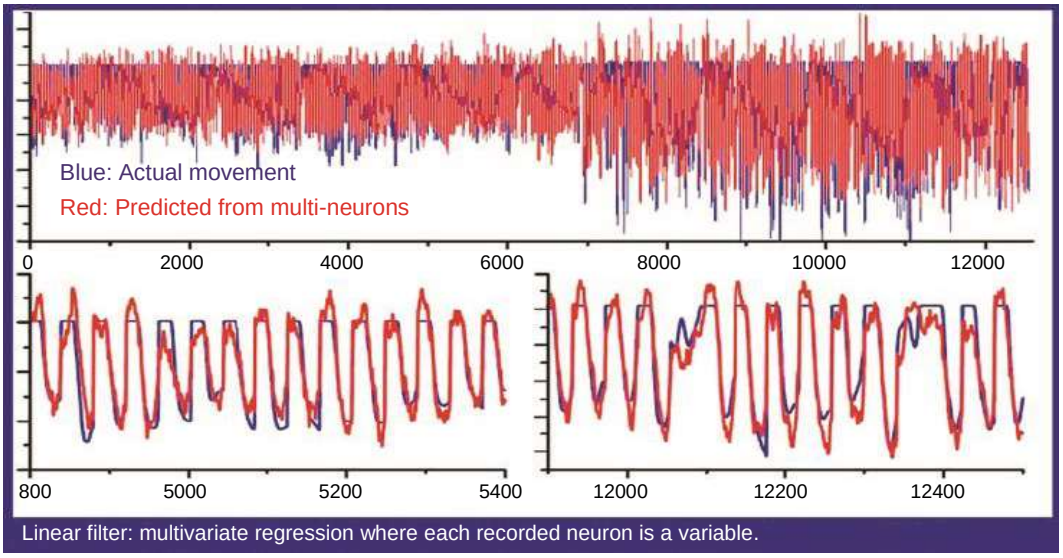

Fig. (2.19). An experiment that taught an animal to take motion while gripping a robotic manipulandum that generated two separate consistent forces, either a 3 g or a 10 g force.

2.10. BIOMIMETIC ROBOT RESEARCH

A variety of important robotics have been developed, including the actuators and the sensors are not just proprioceptive, touch, and visual. The ARTS robot is made up of 25 DOFs, 2 visual sensors, 39 sensors, and 135 image stabilization. and has a head, arm, and hand. It contains a multinetwork architecture that is biologically inspired and incorporates progressive and functional learning for motion analysis. A 5-digit biomimetic hand prosthesis is their latest biomechatronic cyber hand (Fig. **2.20**) which includes a motor plus embedded biomimetic strength and proprioceptive sensors for each digit. Although, the numbers are under control, conformity with the hand and the integrated loop control offer an artificial hand excellent gripping feature. This instrument is meant for use by amputees and is thus operated by an interface with the nerves in the remaining arm of the patient.

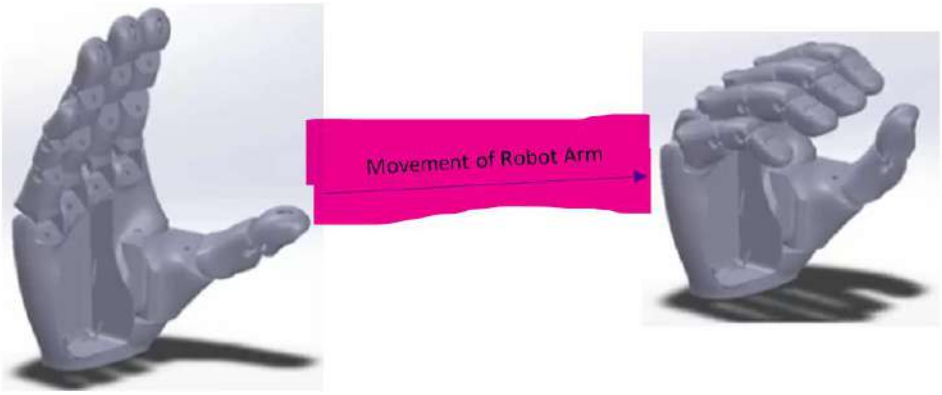

Fig. (2.20). Robot hand system.

2.10.1. The Rationale for Biomimetic Hand Prostheses

Typical hand-prothesis has until recently been made up of metal hooks operated by a myoelectric interface. It is therefore essential to upgrade these prosthesis hands for practical and esthetic reasons. The cyber hand [45] is thus motivated by a knowledge of hand and finger kinesiology (Fig. **2.21**). Since the creation of a robot with all the features of actual human beings would be difficult, the challenge is to build modifications incorporating similar and less complicated roles.

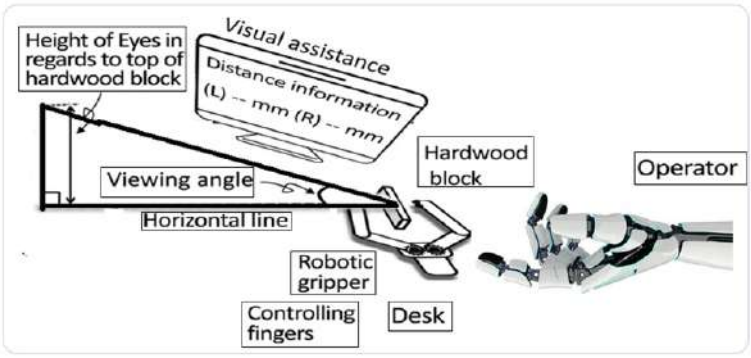

Fig. (2.21). Manual and automated kinesiology during manipulation.

2.10.2. Research Approach to Bio Mechatronics at SSSA

Professor Dario emphasizes the importance of using anatomy, physiology, and neuroscience to support the development of robotics. This laboratory, as shown in Fig. (**2.22**), advocates a biomechanical approach that goes beyond basic mechanics, rather than a whole framework that typically takes the hand into account.

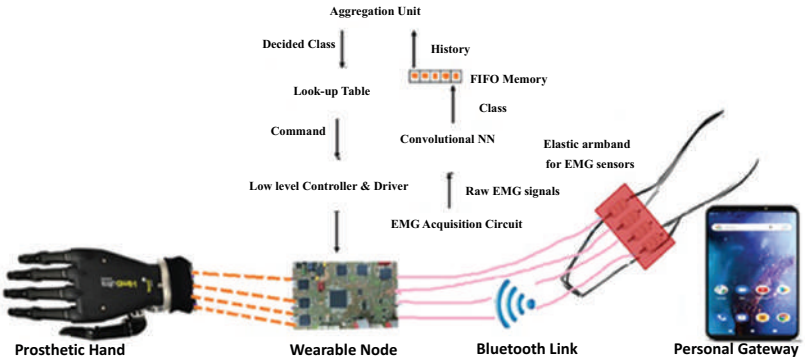

Fig. (2.22). Biomechatronic approach to duplicating the natural hand.

2.10.3. Using Direct BCIs to Control Biomimetic Robotic Prostheses

Although SSSA focuses on regulating hand prosthetics through the use of signals taken from the peripheral nerves, the use of direct BCIs can also be envisaged, allowing for a prosthesis directly from the brain to be regulated. Our latest BCI monkey studies use a two-fold brain interface that involves an engine prosthesis (activated by neural motor particular records) and a tactile/proprioceptive neuro transparent neuro-prostheses by stimulating electrode clusters.

Liberty degrees. The creation of high-DOF protheses robots, such as Cyber hand, enables this possibility only in parallel. It is important to remember that work in several fields contributes to synergistic success around the board. The promotes study that makes use of the research between research focused on neuroscience and the technical paradigm of continuous production and testing of experimental technological devices, based on hypotheses.

CONCLUSION

Most of Europe's BCI sciences include “non-invasive” sensor technology, which means multi-electrode recordings from EEG electrode arrays mounted on the skull surface. This sensor technology has seen little development till now and requires considerable progress in future. Many European sites cooperate or use paradigms built in the US even in the case of non-invasive technology (Wadsworth Center, Albany, NY). Because of large population, now China is emphasis on the low-cost, noninvasive BCI innovations to enhance healthcare services in the countary. Furthermore, Japan concentrates on noninvasive BCI technology focused on EEGs. In China and Japan, however, fast economic growth and scientific expenditures are pushing BCI in Asia. There are also strong signs that BCI sensor technology invasive in China is available for interest and equipment. Asia has production facilities and resources to enable new, invasive

creation of the BCI sensor that will contend with or surpass US efforts in five to 10 years from now.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] T. Akin, B. Ziaie, S.A. Nikles, and K. Najafi, "A modular micromachined high-density connector system for biomedical applications", *IEEE Trans. Biomed. Eng.,* vol. 46, no. 4, pp. 471-480, 1999. [http://dx.doi.org/10.1109/10.752944] [PMID: 10217885]

[2] D.J. Anderson, K. Najafi, S.J. Tanghe, D.A. Evans, K.L. Levy, J.F. Hetke, X.L. Xue, J.J. Zappia, and K.D. Wise, "Batch-fabricated thin-film electrodes for stimulation of the central auditory system", *IEEE Trans. Biomed. Eng.,* vol. 36, no. 7, pp. 693-704, 1989. [http://dx.doi.org/10.1109/10.32101] [PMID: 2744793]

[3] Q. Bai, and K.D. Wise, "Single-unit neural recording with active microelectrode arrays", *IEEE Trans. Biomed. Eng.,* vol. 48, no. 8, pp. 911-920, 2001. [http://dx.doi.org/10.1109/10.936367] [PMID: 11499528]

[4] Q. Bai, K.D. Wise, and D.J. Anderson, "A high-yield microassembly structure for three-dimensional microelectrode arrays", *IEEE Trans. Biomed. Eng.,* vol. 47, no. 3, pp. 281-289, 2000. [http://dx.doi.org/10.1109/10.827288] [PMID: 10743769]

[5] A.A. Baumeister, "Serendipity and the cerebral localization of pleasure", *J. Hist. Neurosci.,* vol. 15, no. 2, pp. 92-98, 2006. [http://dx.doi.org/10.1080/09647040500274879] [PMID: 16608738]

[6] S.L. BeMent, K.D. Wise, D.J. Anderson, K. Najafi, and K.L. Drake, "Solid-state electrodes for multichannel multiplexed intracortical neuronal recording", *IEEE Trans. Biomed. Eng.,* vol. 33, no. 2, pp. 230-241, 1986. [http://dx.doi.org/10.1109/TBME.1986.325895] [PMID: 3957372]

[7] D.A. Borkholder, J. Bao, N.I. Maluf, E.R. Perl, and G.T. Kovacs, "Microelectrode arrays for stimulation of neural slice preparations", *J. Neurosci. Methods,* vol. 77, no. 1, pp. 61-66, 1997. [http://dx.doi.org/10.1016/S0165-0270(97)00112-X] [PMID: 9402558]

[8] A. Bragin, J. Hetke, C.L. Wilson, D.J. Anderson, J. Engel Jr, and G. Buzsáki, "Multiple site silicon-based probes for chronic recordings in freely moving rats: implantation, recording and histological verification", *J. Neurosci. Methods,* vol. 98, no. 1, pp. 77-82, 2000. [http://dx.doi.org/10.1016/S0165-0270(00)00193-X] [PMID: 10837874]

[9] A. Branner, R.B. Stein, E. Fernandez, Y. Aoyagi, and R.A. Normann, "Long-term stimulation and recording with a penetrating microelectrode array in cat sciatic nerve", *IEEE Trans. Biomed. Eng.,* vol. 51, no. 1, pp. 146-157, 2004. [http://dx.doi.org/10.1109/TBME.2003.820321] [PMID: 14723504]

[10] J.J. Burmeister, and G.A. Gerhardt, "Self-referencing ceramic-based multisite microelectrodes for the detection and elimination of interferences from the measurement of L-glutamate and other analytes", *Anal. Chem.,* vol. 73, no. 5, pp. 1037-1042, 2001.
[http://dx.doi.org/10.1021/ac0010429] [PMID: 11289414]

[11] Burmeister J.J., and Gerhardt G.A., 2006. Neurochemical arrays. In Encyclopedia of sensors, Vol. 6, Eds. C. Grimes, E. Dickey, and M.V. Pishko. Stevenson Ranch, CA: American Scientific Publishers, 525.

[12] J.J. Burmeister, K. Moxon, and G.A. Gerhardt, "Ceramic-based multisite microelectrodes for electrochemical recordings", *Anal. Chem.,* vol. 72, no. 1, pp. 187-192, 2000.
[http://dx.doi.org/10.1021/ac9907991] [PMID: 10655652]

[13] J.J. Burmeister, F. Pomerleau, M. Palmer, B.K. Day, P. Huettl, and G.A. Gerhardt, "Improved ceramic-based multisite microelectrode for rapid measurements of L-glutamate in the CNS", *J. Neurosci. Methods,* vol. 119, no. 2, pp. 163-171, 2002.
[http://dx.doi.org/10.1016/S0165-0270(02)00172-3] [PMID: 12323420]

[14] J.K. Chapin, and D.J. Woodward, "Distribution of somatic sensory and active-movement neuronal discharge properties in the MI-SI cortical border area in the rat", *Exp. Neurol.,* vol. 91, no. 3, pp. 502-523, 1986.
[http://dx.doi.org/10.1016/0014-4886(86)90048-8] [PMID: 3948958]

[15] J.K. Chapin, and M.A.L. Nicolelis, Brain Control of Sensorimotor Prostheses.*Neural prostheses for restoration of sensory and motor function.,* J.K. Chapin, K.A. Moxon, Eds., CRC Press: Boca Raton, FL, 2001, pp. 235-261.

[16] J.K. Chapin, "Using multi-neuron population recordings for neural prosthetics", *Nat. Neurosci.,* vol. 7, no. 5, pp. 452-455, 2004.
[http://dx.doi.org/10.1038/nn1234] [PMID: 15114357]

[17] J. Chen, and K.D. Wise, "A silicon probe with integrated microheaters for thermal marking and monitoring of neural tissue", *IEEE Trans. Biomed. Eng.,* vol. 44, no. 8, pp. 770-774, 1997.
[http://dx.doi.org/10.1109/10.605437] [PMID: 9254990]

[18] J. Chen, K.D. Wise, J.F. Hetke, and S.C. Bledsoe Jr, "A multichannel neural probe for selective chemical delivery at the cellular level", *IEEE Trans. Biomed. Eng.,* vol. 44, no. 8, pp. 760-769, 1997.
[http://dx.doi.org/10.1109/10.605435] [PMID: 9254989]

[19] K.C. Cheung, "Implantable microscale neural interfaces", *Biomed. Microdevices,* vol. 9, no. 6, pp. 923-938, 2007. epub
[http://dx.doi.org/10.1007/s10544-006-9045-z] [PMID: 17252207]

[20] T.C. Chiganos Jr, W. Jensen, and P.J. Rousche, "Electrophysiological response dynamics during focal cortical infarction", *J. Neural Eng.,* vol. 3, no. 4, pp. L15-L22, 2006.
[http://dx.doi.org/10.1088/1741-2560/3/4/L01] [PMID: 17124326]

[21] J. Csicsvari, D.A. Henze, B. Jamieson, K.D. Harris, A. Sirota, P. Barthó, K.D. Wise, and G. Buzsáki, "Massively parallel recording of unit and local field potentials with silicon-based electrodes", *J. Neurophysiol.,* vol. 90, no. 2, pp. 1314-1323, 2003.
[http://dx.doi.org/10.1152/jn.00116.2003] [PMID: 12904510]

[22] C.C. Della Santina, G.T. Kovacs, and E.R. Lewis, "Multi-unit recording from regenerated bullfrog eighth nerve using implantable silicon-substrate microelectrodes", *J. Neurosci. Methods,* vol. 72, no. 1, pp. 71-86, 1997.
[http://dx.doi.org/10.1016/S0165-0270(96)00159-8] [PMID: 9128171]

[23] K.L. Drake, K.D. Wise, J. Farraye, D.J. Anderson, and S.L. BeMent, "Performance of planar multisite microprobes in recording extracellular single-unit intracortical activity", *IEEE Trans. Biomed. Eng.,* vol. 35, no. 9, pp. 719-732, 1988.
[http://dx.doi.org/10.1109/10.7273] [PMID: 3169824]

[24] G. Ensell, D.J. Banks, P.R. Richards, W. Balachandran, and D.J. Ewins, "Silicon-based microelectrodes for neurophysiology, micromachined from silicon-on-insulator wafers", *Med. Biol. Eng. Comput.,* vol. 38, no. 2, pp. 175-179, 2000.
[http://dx.doi.org/10.1007/BF02344773] [PMID: 10829410]

[25] E.A. Felton, J.A. Wilson, J.C. Williams, and P.C. Garell, "Electrocorticographically controlled brain-computer interfaces using motor and sensory imagery in patients with temporary subdural electrode implants. Report of four cases", *J. Neurosurg.,* vol. 106, no. 3, pp. 495-500, 2007.
[http://dx.doi.org/10.3171/jns.2007.106.3.495] [PMID: 17367076]

[26] C. Fonseca, J.P. Silva Cunha, R.E. Martins, V.M. Ferreira, J.P. Marques de Sá, M.A. Barbosa, and A. Martins da Silva, "A novel dry active electrode for EEG recording", *IEEE Trans. Biomed. Eng.,* vol. 54, no. 1, pp. 162-165, 2007.
[http://dx.doi.org/10.1109/TBME.2006.884649] [PMID: 17260869]

[27] C. Gonzalez, and M. Rodriguez, "A flexible perforated microelectrode array probe for nerve and muscle tissues", *J. Neurosci. Methods,* vol. 72, pp. 189-195, 1997.
[http://dx.doi.org/10.1016/S0165-0270(96)02202-9] [PMID: 9133584]

[28] R.G. Heath, S.M. Peacock Jr, and W. Miller Jr, "Induced paroxysmal electrical activity in man recorded simultaneously through subcortical and scalp electrodes", *Trans. Am. Neurol. Assoc.,* vol. 3, no. 78th Meeting, pp. 247-250, 1953.
[PMID: 13179226]

[29] J.F. Hetke, J.L. Lund, K. Najafi, K.D. Wise, and D.J. Anderson, "Silicon ribbon cables for chronically implantable microelectrode arrays", *IEEE Trans. Biomed. Eng.,* vol. 41, no. 4, pp. 314-321, 1994.
[http://dx.doi.org/10.1109/10.284959] [PMID: 8063297]

[30] L.R. Hochberg, M.D. Serruya, G.M. Friehs, J.A. Mukand, M. Saleh, A.H. Caplan, A. Branner, D. Chen, R.D. Penn, and J.P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia", *Nature,* vol. 442, no. 7099, pp. 164-171, 2006.
[http://dx.doi.org/10.1038/nature04970] [PMID: 16838014]

[31] A.C. Hoogerwerf, and K.D. Wise, "A three-dimensional microelectrode array for chronic neural recording", *IEEE Trans. Biomed. Eng.,* vol. 41, no. 12, pp. 1136-1146, 1994.
[http://dx.doi.org/10.1109/10.335862] [PMID: 7851915]

[32] W. Jensen, K. Yoshida, and U.G. Hofmann, "In-vivo implant mechanics of flexible, silicon-based ACREO microelectrode arrays in rat cerebral cortex", *IEEE Trans. Biomed. Eng.,* vol. 53, no. 5, pp. 934-940, 2006.
[http://dx.doi.org/10.1109/TBME.2006.872824] [PMID: 16686416]

[33] D.R. Kipke, R.J. Vetter, J.C. Williams, and J.F. Hetke, "Silicon-substrate intracortical microelectrode arrays for long-term recording of neuronal spike activity in cerebral cortex", *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 11, no. 2, pp. 151-155, 2003.
[http://dx.doi.org/10.1109/TNSRE.2003.814443] [PMID: 12899260]

[34] E.H. Kossoff, E.K. Ritzl, J.M. Politsky, A.M. Murro, J.R. Smith, R.B. Duckrow, D.D. Spencer, and G.K. Bergey, "Effect of an external responsive neurostimulator on seizures and electrographic discharges during subdural electrode monitoring", *Epilepsia,* vol. 45, no. 12, pp. 1560-1567, 2004.
[http://dx.doi.org/10.1111/j.0013-9580.2004.26104.x] [PMID: 15571514]

[35] G.T. Kovacs, C.W. Storment, and J.M. Rosen, "Regeneration microelectrode array for peripheral nerve recording and stimulation", *IEEE Trans. Biomed. Eng.,* vol. 39, no. 9, pp. 893-902, 1992.
[http://dx.doi.org/10.1109/10.256422] [PMID: 1473818]

[36] G.T. Kovacs, C.W. Storment, M. Halks-Miller, C.R. Belczynski Jr, C.C. Della Santina, E.R. Lewis, and N.I. Maluf, "Silicon-substrate microelectrode arrays for parallel recording of neural activity in peripheral and cranial nerves", *IEEE Trans. Biomed. Eng.,* vol. 41, no. 6, pp. 567-577, 1994.
[http://dx.doi.org/10.1109/10.293244] [PMID: 7927376]

[37] E.C. Leuthardt, G. Schalk, J.R. Wolpaw, J.G. Ojemann, and D.W. Moran, "A brain-computer interface using electrocorticographic signals in humans", *J. Neural Eng.,* vol. 1, no. 2, pp. 63-71, 2004. [http://dx.doi.org/10.1088/1741-2560/1/2/001] [PMID: 15876624]

[38] L. Lin, G. Chen, K. Xie, K.A. Zaia, S. Zhang, and J.Z. Tsien, "Large-scale neural ensemble recording in the brains of freely behaving mice", *J. Neurosci. Methods,* vol. 155, no. 1, pp. 28-38, 2006. [http://dx.doi.org/10.1016/j.jneumeth.2005.12.032] [PMID: 16554093]

[39] N. Ludvig, Drug deliveries into the microenvironment of electrophysiologically monitored neurons in the brain of behaving rats and monkeys. *Neural prostheses for restoration of sensory and motor function.,* J.K. Chapin, K.A. Moxon, Eds., CRC Press: Boca Raton, FL, 2001, pp. 263-283.

[40] T.C. Marzullo, J.R. Dudley, C.R. Miller, L. Trejo, and D.R. Kipke, "Spikes, local field potentials, and electrocorticogram characterization during motor learning in rats for brainmachine interface tasks", *Conf. Proc. IEEE Eng. Med. Biol. Soc.,* vol. 2006, pp. 429-431, 2005. [PMID: 17282206]

[41] K.A. Moxon, S.C. Leiser, G.A. Gerhardt, K.A. Barbee, and J.K. Chapin, "Solid-state electrodes for multichannel multiplexed intracortical neuronal recording", *IEEE Trans. Biomed. Eng.,* vol. 51, p. 647, 2004. [http://dx.doi.org/10.1109/TBME.2003.821037] [PMID: 15072219]

[42] K. Najafi, J. Ji, and K.D. Wise, "Scaling limitations of silicon multichannel recording probes", *IEEE Trans. Biomed. Eng.,* vol. 37, no. 1, pp. 1-11, 1990. [http://dx.doi.org/10.1109/10.43605] [PMID: 2303265]

[43] M.A. Nicolelis, D. Dimitrov, J.M. Carmena, R. Crist, G. Lehew, J.D. Kralik, and S.P. Wise, "Chronic, multisite, multielectrode recordings in macaque monkeys", *Proc. Natl. Acad. Sci. USA,* vol. 100, no. 19, pp. 11041-11046, 2003. [http://dx.doi.org/10.1073/pnas.1934665100] [PMID: 12960378]

[44] C.T. Nordhausen, E.M. Maynard, and R.A. Normann, "Single unit recording capabilities of a 100 microelectrode array", *Brain Res.,* vol. 726, no. 1-2, pp. 129-140, 1996. [http://dx.doi.org/10.1016/0006-8993(96)00321-6] [PMID: 8836553]

[45] J. Olds, W.S. Allan, and E. Briese, "Differentiation of hypothalamic drive and reward centers", *Am. J. Physiol.,* vol. 221, no. 1, pp. 368-375, 1971. [http://dx.doi.org/10.1152/ajplegacy.1971.221.1.368] [PMID: 5555810]

[46] J.W. Pan, J.H. Kim, A. Cohen-Gadol, C. Pan, D.D. Spencer, and H.P. Hetherington, "Regional energetic dysfunction in hippocampal epilepsy", *Acta Neurol. Scand.,* vol. 111, no. 4, pp. 218-224, 2005. [http://dx.doi.org/10.1111/j.1600-0404.2005.00398.x] [PMID: 15740571]

[47] J.J. Pancrazio, P.P. Bey Jr, A. Loloee, S. Manne, H.C. Chao, L.L. Howard, W.M. Gosney, D.A. Borkholder, G.T. Kovacs, P. Manos, D.S. Cuttino, and D.A. Stenger, "Description and demonstration of a CMOS amplifier-based-system with measurement and stimulation capability for bioelectrical signal transduction", *Biosens. Bioelectron.,* vol. 13, no. 9, pp. 971-979, 1998. [http://dx.doi.org/10.1016/S0956-5663(98)00006-2] [PMID: 9839386]

[48] W.R. Patterson, Y.K. Song, C.W. Bull, I. Ozden, A.P. Deangellis, C. Lay, J.L. McKay, A.V. Nurmikko, J.D. Donoghue, and B.W. Connors, "A microelectrode/microelectronic hybrid device for brain implantable neuroprosthesis applications", *IEEE Trans. Biomed. Eng.,* vol. 51, no. 10, pp. 1845-1853, 2004. [http://dx.doi.org/10.1109/TBME.2004.831521] [PMID: 15490832]

[49] R. Rathnasingham, D.R. Kipke, S.C. Bledsoe Jr, and J.D. McLaren, "Characterization of implantable microfabricated fluid delivery devices", *IEEE Trans. Biomed. Eng.,* vol. 51, no. 1, pp. 138-145, 2004. [http://dx.doi.org/10.1109/TBME.2003.820311] [PMID: 14723503]

[50] R.L. Rennaker, A.M. Ruyle, S.E. Street, and A.M. Sloan, "An economical multi-channel cortical

electrode array for extended periods of recording during behavior", *J. Neurosci. Methods,* vol. 142, no. 1, pp. 97-105, 2005.
[http://dx.doi.org/10.1016/j.jneumeth.2004.07.018] [PMID: 15652622]

[51] P.J. Rousche, D.S. Pellinen, D.P. Pivin Jr, J.C. Williams, R.J. Vetter, and D.R. Kipke, "Flexible polyimide-based intracortical electrode arrays with bioactive capability", *IEEE Trans. Biomed. Eng.,* vol. 48, no. 3, pp. 361-371, 2001.
[http://dx.doi.org/10.1109/10.914800] [PMID: 11327505]

[52] G. Ruffini, S. Dunne, E. Farres, P.C.P. Watts, E. Mendoza, S.R.P. Silva, and C. Grau, "ENOBIO: First tests of a dry electrophysiology electrode using carbon nanotubes", *Proceedings of the 28th IEEE EMBS Annual International Conference,* 2006pp. 1826-1829 New York City
[http://dx.doi.org/10.1109/IEMBS.2006.259248]

[53] S. Schmidt, K. Horch, and R. Normann, "Biocompatibility of silicon-based electrode arrays implanted in feline cortical tissue", *J. Biomed. Mater. Res.,* vol. 27, no. 11, pp. 1393-1399, 1993.
[http://dx.doi.org/10.1002/jbm.820271106] [PMID: 8263001]

[54] S.L. Smith, J.W. Judy, and T.S. Otis, "An ultra small array of electrodes for stimulating multiple inputs into a single neuron", *J. Neurosci. Methods,* vol. 133, no. 1-2, pp. 109-114, 2004.
[http://dx.doi.org/10.1016/j.jneumeth.2003.10.001] [PMID: 14757351]

[55] Y.K. Song, W.R. Patterson, C.W. Bull, J. Beals, N. Hwang, A.P. Deangelis, C. Lay, J.L. McKay, A.V. Nurmikko, M.R. Fellows, J.D. Simeral, J.P. Donoghue, and B.W. Connors, "Development of a chipscale integrated microelectrode/microelectronic device for brain implantable neuroengineering applications", *IEEE Trans. Neural Syst. Rehab. Eng,* vol. 13, p. 20, .

[56] K. Sugiyama, W.K. Dong, and E.H. Chudler, "A simplified method for manufacturing glass-insulated metal microelectrodes", *J. Neurosci. Methods,* vol. 53, no. 1, pp. 73-80, 1994.
[http://dx.doi.org/10.1016/0165-0270(94)90146-5] [PMID: 7990516]

[57] H. Takahashi, T. Ejiri, M. Nakao, N. Nakamura, K. Kaga, and T. Hervé, "Microelectrode array on folding polyimide ribbon for epidural mapping of functional evoked potentials", *IEEE Trans. Biomed. Eng.,* vol. 50, no. 4, pp. 510-516, 2003.
[http://dx.doi.org/10.1109/TBME.2003.809483] [PMID: 12723063]

[58] R.J. Vetter, J.C. Williams, J.F. Hetke, E.A. Nunamaker, and D.R. Kipke, "Chronic neural recording using silicon-substrate microelectrode arrays implanted in cerebral cortex", *IEEE Trans. Biomed. Eng.,* vol. 51, no. 6, pp. 896-904, 2004.
[http://dx.doi.org/10.1109/TBME.2004.826680] [PMID: 15188856]

[59] D.J. Warren, E. Fernandez, and R.A. Normann, "High-resolution two-dimensional spatial mapping of cat striate cortex using a 100-microelectrode array", *Neuroscience,* vol. 105, no. 1, pp. 19-31, 2001.
[http://dx.doi.org/10.1016/S0306-4522(01)00174-9] [PMID: 11483297]

[60] K.D. Wise, and J.B. Angell, "A low-capacitance multielectrode probe for use in extracellular neurophysiology", *IEEE Trans. Biomed. Eng.,* vol. 22, no. 3, pp. 212-219, 1975.
[http://dx.doi.org/10.1109/TBME.1975.324562] [PMID: 1116854]

[61] K.D. Wise, J.B. Angell, and A. Starr, "An integrated-circuit approach to extracellular microelectrodes", *IEEE Trans. Biomed. Eng.,* vol. 17, no. 3, pp. 238-247, 1970.
[http://dx.doi.org/10.1109/TBME.1970.4502738] [PMID: 5431636]

[62] J.C. Williams, R.L. Rennaker, and D.R. Kipke, "Long-term neural recording characteristics of wire microelectrode arrays implanted in cerebral cortex", *Brain Res. Brain Res. Protoc.,* vol. 4, no. 3, pp. 303-313, 1999.
[http://dx.doi.org/10.1016/S1385-299X(99)00034-3] [PMID: 10592339]

[63] J.A. Wilson, E.A. Felton, P.C. Garell, G. Schalk, and J.C. Williams, "An ECoG-based brain-computer interface using multimodal control", *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 14, no. 2, pp. 246-250, 2006.
[http://dx.doi.org/10.1109/TNSRE.2006.875570] [PMID: 16792305]

[64] T.H. Yoon, E.J. Hwang, D.Y. Shin, S.I. Park, S.J. Oh, S.C. Jung, H.C. Shin, and S.J. Kim, "A micromachined silicon depth probe for multichannel neural recording", *IEEE Trans. Biomed. Eng.,* vol. 47, no. 8, pp. 1082-1087, 2000.
[http://dx.doi.org/10.1109/10.855936] [PMID: 10943057]

[65] K. Yoshida, W. Jensen, P. Norlin, M. Kindlundh, and U.G. Hofmann, "Hofmann. 2001. Characterization of silicon microelectrodes from the EU VSAMUEL project", *Biomediinische Technik,* 2001.

CHAPTER 3

Potential Use of Tree-based Tools for Chemometric Analysis of Infrared Spectra

Lucas A.C. Minho[1,*], Bárbara E.A. de Magalhães[2] and **Alexandre G.M. de Freitas[3]**

[1] *Departamento de Química, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil*

[2] *Instituto de Química, Universidade Federal da Bahia, Salvador, Brazil*

[3] *Centro de Estudos em Leite, Departamento de Tecnologia Rural e Animal, Universidade Estadual do Sudoeste da Bahia, Itapetinga, Brazil*

Abstract: One of the most elegant and versatile techniques of machine learning is the decision tree. The decision tree is a simple tool to predict and explain the relationship between the object and the target value, recursively partitioning the input space. Tree ensembles such as random forest and gradient boosting trees significantly improve the predictive power of supervised models based on tree weak predictors. In a random forest, the generalized error that is included in the model prediction is dependent on the correlation strength between the trees and the individual predictors' quality. The random selection of features in each node split is at the core of random forest, which makes it as effective as other complex machine learning techniques while having a lower computational cost, which is appealing in the analysis of large data matrices such as those generated by infrared spectroscopy because most analysts do not have computers with high processing capacity for implementing those complex models. Also, techniques based on the decision tree are more robust to noise, which is preferable for the analysis of trace level contaminants. In this chapter, we present the techniques based on decision trees and apply them to solve problems related to classification, regression, and feature selection in spectra obtained experimentally and provided by public repositories. Comparisons of the performance obtained with techniques based on the decision tree in relation to other chemometric tools are also performed.

Keywords: Analytical screening, ATR, CART, Chemometrics, Data mining, Discriminant analysis, Decision trees, Feature selection, FTIR, Gradient boosting machines, Machine learning, Non-parametric models, Non-Linearity, Neural Networks, NIR, Predictive models, Supervised learning, Random forest, Regression, Supervised learning, Validation.

[*] **Corresponding author Lucas A.C. Minho:** Departamento de Química, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil; Tel: +55(31)994055085; E-mail: cavalcantelucas@ufmg.br

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

3.1. INTRODUCTION

The association of spectroscopy and multivariate calibration methods has enabled the analysis of complex spectra of multicomponent systems. Therefore, it has been used in a wide variety of regression methods [1]. The application of new pattern recognition algorithms, such as methods based on decision trees, has grown in recent years due to its advantages and ability to solve complex problems for the purposes of classification and calibration [2]. Decision tree models, capable of modeling linear and non-linear relationships, stand out among the various regression methods for being easy to understand, fast and non-parametric [1].

To improve the predictive performance of the decision tree, a large number of tree-based algorithms have been developed, such as decision forests [3]. The use of decision tree-based methods has become popular in a wide variety of areas [4], for example, food quality and authenticity, metabolomics and ecology [2].

The best-known decision tree method is the Classification and Regression Tree (CART). CART is a binary tree representation capable of describing the relationships between dependent (numerical or categorical) and independent variables with sufficient precision [5]. It is a supervised non-parametric technique used for the purposes of classification and regression [6]. CART is an algorithm used in local linear and non-linear adjustment with categorical (classification) or continuous (regression) variables. This algorithm is used in the random forest (RF) as a tree growth algorithm [1].

The decision forest method best known is the Random Forest (RF) [7]. RF consists of a group of unadjusted decision trees that grow by a bootstrap sampling of training data and random selection of variables [1]. RF has become popular due to the simplicity of training and adjustment parameters and due to the ability to deal with complex non-linear systems [8].

Gradient Boosted Trees (GBT) is another ensemble-based method used to reduce the error of 'weak' predictors as decision trees by repeatedly running and recalibrating the weights [9]. The GBT algorithm can be combined with several machine learning methods promoting an improvement in the accuracy of the prediction results. In addition, the GBT algorithm adds the advantages of having a low degree of overfitting and good generalization performance [10].

In this chapter, we presented some tree-based techniques as CART and random forest (for more details regarding other tools based on decision trees, the study of James *et al*. can be referred to [11]) and applied them to solve problems related to classification, regression and feature selection in FTIR spectra obtained

experimentally and provided by public repositories. The performance of the applied decision tree-based techniques was compared to the performance of other chemometric tools.

3.2. DECISION TREES (DT)

Typically used in classification operations, decision trees are elegant machine learning tools that are based on a decision model that express conditional control instructions. Decision trees generate rectangular subsets, A_j, recursively partitioning the data set, one X attribute at a time, making inferable decisions at each step. All trees start from a root node, from which all other nodes will originate. A node with outgoing edges is called a test or internal node. The rest, which originate from the branches and find their end, are called leaves or decision nodes. In operation, each internal node divides the attribute space into a predetermined number of sub-spaces, according to a discrete function. The number of branches varies according to the complexity of the tree.

Each internal node contains an attribute being tested, and, especially in the classification trees, each leaf represents a different class. Also, it is common for the leaves to have a vector of continuous values that informs the probability that the target value is predicted with certain accuracy [12]. The instances are, therefore, classified from the root to the leaves, as shown in Fig. (**3.1**).

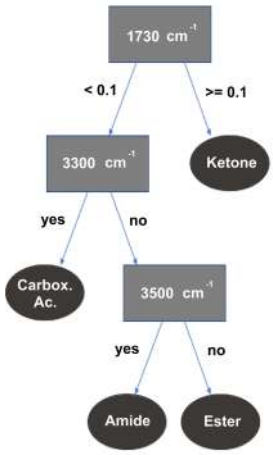

Fig. (3.1). Example of a decision tree using spectral information.

In Fig. (**3.1**), the internal nodes were represented by rectangles and the leaves by ellipses. The first internal node in the series is the root. It is interesting to note that

a rule can be inferred by conjoining the splits to form a logical statement. In the case of Fig. (**3.1**), for example, "If the absorbance obtained at the peak in 1730 cm^{-1} is greater than or equal to 0.1, the analyzed substance is a ketone. Otherwise, if the absorbance at 1730 cm^{-1} is less than 0.1, but there is a peak at 3300 cm^{-1}, the substance is a carboxylic acid or an amide if there is a peak at 3500 cm^{-1}. Otherwise, if the absorbance is less than 0.1 in 1730 cm^{-1} and none of the other peaks is observed, it is an ester. "This property of inferring understandable rules, unlike the black boxes that characterize other non-parametric techniques, makes decision trees very popular with earth science specialists [13-16].

Possibly, one of the first techniques that were based on a conditional control recursive tree algorithm was the Automatic Interaction Detector (AID), which emerged in the 60s and 70s [17]. The well-known classification and regression trees (CART) were introduced in 1984 by Breiman [18], which are characterized by the generation of binary trees; each internal node has only one split, as illustrated in Fig. (**3.1**). The CART is used, in essence, a binomial distribution index, the Gini impurity index (which will be explained in section 3.5.1), as a measure of accuracy in discriminating operations. Alternatively, the trees described by Breiman are able to perform regressions. The regression trees prioritize branches that minimize the quadratic error of continuous value prediction. It is important to mention that, unlike the case of discrete attributes in the classificatory analysis, in the numerical attributes during a regression operation, the trees must be understood as sets of hyperplanes, each orthogonal to each of the axes.

Despite being simple and quite interpretive, decision trees have some disadvantages [12]; (i) due to their greedy characteristic, DTs have great sensitivity to the presence of irrelevant attributes and noise, and so, (ii) DT's "divide and conquer" strategy tends to work very well with sets of small dimensionality and with many relevant attributes, but it loses effectiveness quickly as there is an increase in the complexity and dimensionality of the input space. These disadvantages have been minimized in the techniques that employ an ensemble of trees.

3.3. RANDOM FOREST (RF)

In order to improve the predictive performance of decision trees, instead of working with a single tree, a decision tree ensemble can be used, in which the training of several trees and their predictions are combined. The most popular decision tree ensemble is the random forest [7].

RF is a powerful statistical classifier and a machine learning tool for classification and regression [19 - 21]. The idea was introduced in the mid-1990s,

independently and almost simultaneously, by Yali Amit and Donald Geman (1994) and Tin Kam Ho (1995) [7] and perfected by Leo Breiman [22]. RF consists of a set of unpruned decision trees, created using bootstrap samples of the training data and random feature selection in the induction of the tree [3]. The idea of generating several bootstrap samples and averaging predictors is known as bagging (bootstrap aggregating). Formally speaking, the *M* trees randomized by a random vector Θ, where Θ_1, ..., Θ_M are random attributes in bootstrap samples independent of the original dataset D_n, are combined to generate forest estimates:

$$m_{M.n}(x;\ \Theta_1, \dots, \Theta_M,\ \mathcal{D}_\mathrm{n}) = \frac{1}{M}\sum_{j=1}^{M} m_n(x; \Theta_j,\ \mathcal{D}_\mathrm{n}) \quad \textbf{(3.1)}$$

Also, another great advantage of using this ensemble-based method involves a procedure of producing variable importance [20] (as seen in more detail in section 3.5.1) and in which the prediction is made by gathering the set predictions, considering the majority decision or averaging [3].

The random forest has shown an excellent performance in high-dimensional and ill-posed problems [23], where the number of variables is much greater than the number of observations, being able to deal with highly correlated variables and complex interaction structures [24]. The advantage of RF to deal with multi-dimensional complex data is mainly due to the random distribution of attributes in each tree in the forest, making each element specialize in a section of the input space, and also to the low probability of overfitting since with the increase in trees in the forest, the generalized error converges to a limit value [22].

In comparison with other statistical classifiers, RF presents several advantages; only a few tuning parameters [25], low computational cost [7], nonparametric nature, ability to determine variable importance, high classification accuracy [21], high prediction accuracy [3], capability to model complex interactions among predictor variables, and an algorithm for estimating missing values [19]. A study by Fernández-Delgado *et al*. [26] demonstrated the best performance of RF in relation to other learning methods. In this case, 179 classification algorithms from 17 learning families were compared using 121 data sets, and it was verified that random forest stood out as the best classifier.

Random forests have become popular for their predictive performance [3, 7, 19-22, 25], but RF have the flexibility to perform different types of analysis, involving regression, classification, survival analysis and unsupervised learning [19, 21]. Different versions of random forest algorithms are reported in the literature, varying methodologies for selecting features and trees and selecting the best-split [23].

3.4. EXPERIMENTS

For the discussion of sections 3.5.1.3 and 3.6, we used the spectra provided by Dr. Kalivas [27]. This data set contains 60 different gasoline samples with different compositions and octane numbers (ON). The samples were analyzed by near-infrared spectroscopy (NIR) and measured using diffuse reflectance, between 900 and 1700 nm, with a spacing of 2 nm.

For the discussion of section 3.7, we used the spectra obtained experimentally in a recently published work [28]. The data set consists of 144 samples of pure and tylosin-spiked milk. The samples were analyzed by Attenuated total reflectance-Fourier transform infrared spectroscopy (ATR-FTIR) between 3000 and 1000 cm^{-1} using DairySpec FT equipment (Bentley Instruments, Chaska, USA).

All chemometric analyzes were performed in software R (R Foundation, Austria) version 4.0.0, together with the packages randomForest v. 4.6, ranger v. 0.12.1, rpart v. 3.0.9, pls v. 2.7, neuralnet v. 1.44.2, brglm 0.6.2, gbm v. 2.1.5, through the caret aggregator package (classification and regression training) version 6.0.

3.5. DIMENSIONALITY REDUCTION IN RAW SPECTROSCOPIC SPACE

A typical Fourier transform infrared spectroscopy analysis is usually performed between 4000 and 400 cm^{-1}, with a spectral step of 2 cm^{-1}. A chemometric analysis that considers each wave number obtained in the scan as an independent variable to be analyzed would come across a high-dimensional space containing 1801 potential variables.

Naturally, the number of cases experimentally observed using this instrumental technique is much lesser than the total number of attributes that can be extracted. In such cases, even if wrongly, the parametric machine learning algorithms are preferably used for model fitting. Models based on this type of algorithm summarize data with a set of supervised statistic parameters of fixed size in class estimation. In other words, in the training step, the functional extracts parameters are based on strong statistical assumptions and use some method to estimate the input data. Some examples of parametric machine learning algorithms include linear regression, logistic regression, partial least squares regression (PLS), main components regression (PCR), simple neural networks, among others. These algorithms are also called linear machine learning algorithms [29].

The analysis of data sets obtained by FTIR without any pretreatment by parametric models are frequently affected by the phenomenon described by Bellman [30] and widely reported in the literature [31 - 33], known as “curse of dimensionality”, “Dimensionality problem”, “U-curve problem” or “peaking phenomena” [34], which occurs due to the impossibility of obtaining reliable estimates of an increasing number of parameters for a smaller number of samples. In other words, the increase in attributes that can be extracted generally implies the degradation of the performance of a classifier or regressor if the number of examples for training is very small in relation to the total number of features. What happens is that when the dimension of the input space is high, these models use a good part of their resources to represent irrelevant portions of the search space, hindering the learning process and leading to low accuracy of predictions.

On the other hand, non-parametric classification and regression assume weak assumptions or do not assume any hypothesis about the global distribution of the data. In general, these techniques prioritize the location of regions of low informational density, having the freedom to make estimates and applying to separate hyperplanes between them. Some examples of non-parametric machine learning algorithms include K-nearest neighbor (KNN), support vector machines (SVM), multilayer perceptron (MLP), and decision tree-based algorithms; decision trees like C4.5 and CART and random forest. Such techniques are inherently more robust to the curse of dimensionality [35].

Although minimally affected by the dimensionality problem, decision tree-based algorithms benefit from feature subset selection (FSS) techniques, as they increase the potential to identify relevant features for efficient learning and modeling without losing relevant information, also reducing the resources that are spent on the analysis of irrelevant characteristics and improving the results that are obtained at the voting process. The FSS allows the use of parametric algorithms, which has great popular appeal for simplicity, ease of interpretation and low computational cost, without the risk of loss of efficiency due to the hypervolume of data. In summary, for practical and technical reasons, the selection of variables that leads to the minimum set of attributes without reducing the quality of the information can potentially result in the best possible predictive abilities. This approach is known as the minimal-optimal problem and has been an object of study frequently explored [36 - 38].

3.5.1. Importance Measurements and Feature Ranking with Random Forest

FSSs based on non-parametric algorithms are generally computationally expensive [35]. Fortunately, random forests offer a simpler and more attractive alternative, which is the assessment of the relevance of a predictor based on

importance measures. Breiman [22] introduced variable importance measurements based on some heuristic criteria. In the importance criterion based on Gini impurity, in a classificatory analysis, for each division, the decrease in Gini node impurity is recorded for a given variable X_i belonging to a bootstrapped sample, starting from the following equation:

$$\hat{\Gamma}(t) = \sum_{i=1}^{C} \hat{\phi}_i(t)(1 - \hat{\phi}_i(t)) \quad \textbf{(3.2)}$$

Where C is the total number of classes, and ϕ^ (t) is the probability of randomly selecting a given element of class i in the node. Thus, the sum of all decreases in the Gini impurity in the forest ensemble normalized by the number of trees results in the Gini variable importance measure [39]. Therefore, a division with a great reduction in impurity is considered important, implying that the variables used in this division are also considered important [39, 40].

Another criterion is the measures of importance by permutation. In this case, X_i is characterized as relevant when the lack of this variable implies a negative effect on the predictive capacity, Y^, of the model. Briefly, the link between a variable X_i and the response Y is randomly broken by exchanging the values of all individuals for X_i, and this process is repeated with all subsequent variables $X_1 \ldots X_n$. Also, notably, permutations break the links between X_i and other possible covariates [41]. The difference between OOB prediction errors with and without the permutation process averaged in relation to all trees in the forest, is the importance of the variable X_i (Eq. 3.3).

$$I(X_i) = \mathbb{E}\left[(Y - f(\boldsymbol{X}_{(i)})^2\right] - \mathbb{E}[(Y - f(\boldsymbol{X})^2] \quad \textbf{(3.3)}$$

Then, the raw variable permutation importance measure is applied to each attribute:

$$I(X_j) = \frac{\sum_{t=1}^{ntree} I^t(X_j)}{ntree} \quad \textbf{(3.4)}$$

Where *ntree* is the number of trees in the forest ensemble. Another version of this importance measurement can be obtained by relating each raw importance measure (Eq. 3.4) to its respective standard error. This operation results in the so-called Z score. In short, regardless of which path to follow, the variables can be sorted in order of relevance: greater is the measure of importance by permutation, greater is the impact of this variable on the model's prediction accuracy.

3.5.1.1. Boruta Wrapper Algorithm

In the previous section, we saw that the importance measure of a variable by random forest can be based on the loss of prediction accuracy caused by a random permutation of the attributes, or even classified according to the increase in the decrease in Gini impurity. Despite being very useful in ranking variables, there is no clear statistical criterion for the rejection of low importance variables. In that case, FSSs become more useful.

The feature subset selection techniques are divided into three categories, based on the operating mechanisms: filters, embedded and wrappers. More emphasis will be given to the latter, considering the nature of the object being depicted in this section. It is recommended to read Guyon and Elisseeff [42] for more details about FSSs based on the approach to filters and embedding, as well as other relevant topics that are not of interest in the scope of this chapter.

The wrapper approach was introduced in 1997 by Kohavi and John [43]. The wrappers are black boxes that determine the quality of multiple subsets of characteristics that originate from the input data through the evaluation by an independent learning algorithm after going through many training and testing cycles. At the end of the loop, the best subset of characteristics will be the one with the greatest accuracy of prediction.

The efficiency of a wrapper lies in how the subsets are generated and evaluated. This form depends on the search strategy of the wrapper. The search strategies are commonly classified into optimal strategies, stochastic or sequential selection [42].

In 2010 (with a first prototype presented in 2006 [44]), Kursa and Rudnicki [45], [46] developed a wrapper algorithm called Boruta that is based on random forest, and therefore, uses a stochastic search strategy. Given that Boruta is a deity of Slavic and Russian folklore, the guardian of the forest and its core, the name choice was appropraite Boruta (Leshy) is said to be “changeable as nature” and “varied his appearance as often as he pleased”, “might be tall as a tree or so small that he could slip under a blade of grass” [47]. Boruta's shapeshifter characteristic can be related to the main element of this wrapper, the shadow variables, which are generated and shuffled copies of the elements that are present in each attribute. The Boruta algorithm is illustrated in Fig. (**3.2**).

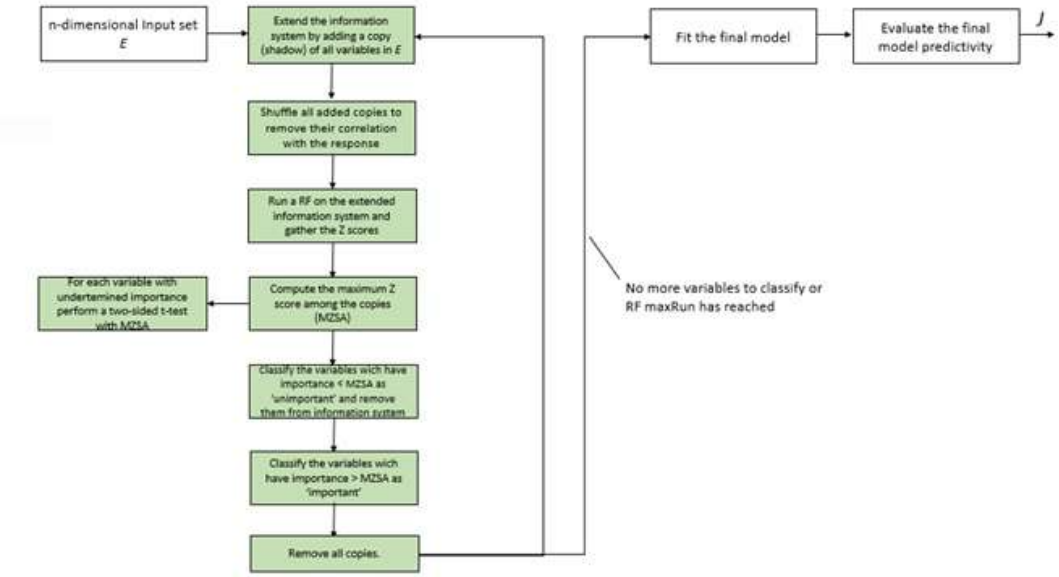

Fig. (3.2). Boruta algorithm. The green rectangles represent the processes initiated by the wrapper and the white rectangles, the normal training procedure.

To deem whether a variable is relevant for predicting responses or whether it can be safely discarded from the input set *E* in order to decrease dimensionality, the developers propose to extend the information system with randomly generated attributes; for each attribute, create a shuffled copy, a 'shadow variable'. Therefore, instead of making the variables compete with each other, as in the measure of importance by permutation, in Boruta, the variables compete with a shadow version of themselves using the Z scores as an importance measurement. The threshold is defined as the highest importance obtained among the shadow variables, MSZA. Then, a two-tailed statistical test of equality is performed between each computed importance of each original variable and the MZSA. As the measures of importance can vary considerably because of the stochasticity of the forests, the algorithm makes successive re-shuffling operations to obtain statistically valid results.

3.5.1.2. Feature Subset Selection with Boruta

Using the data set freely provided by Kalivas [27], Özdemir [48] made a selection of variables using multivariate calibration by genetic algorithms, evaluating the performance of genetic classical least squares (GCLS) and genetic inverse least squares (GILS) methods.

Feature selection based on genetic algorithms (GA) was introduced by Siedlecki and Sklansky [49] and have since been used with great success by several authors [50 - 52]. As GAs and other evolutionary algorithms are focused on the optimization of parameters and have gained a lot of space between the techniques of feature selection, in this section, we make a comparison between the genetic algorithms and the Boruta wrapper, which was introduced in the previous section. For this, we compared the Boruta FSS with a raw permutation importance measurement (BRFR) (Eq. 3.4) and Z scores (BRFZ) in relation to the GAs previously implemented [48] (Table **3.1**).

Table 3.1. Attributes selected by the BRFZ, BRFR, GILS and GCLS algorithms.

	Method			
	BRFZ	BRFR	GILS	GCLS
Selected Wavelenghts (nm)	1072, 1124, 1200, 1202, 1204, 1206, 1208, 1210, 1212, 1214, 1216, 1218, 1220, 1222, 1224, 1588, 1612, 1620, 1630, 1632, 1634, 1636, 1638, 1672	1202, 1204, 1206, 1208, 1210, 1212, 1214, 1216, 1218, 1220, 1638	1002, 1180, 1118, 1148, 1214, 1374, 1434, 1462, 1522, 1582,	1250, 1092, 1556, 1582, 1328, 1336, 1582, 1008, 1146, 1136, 1378, 1106, 1096, 1250, 1092, 1556, 1336, 1582, 1278, 1106, 1086, 1336, 1582

Even though the subsets of attributes obtained with BRFZ (24 selected features) and BRFR (11 selected features) are in full agreement, including a band between 1202 and 1220 nm, with a peak at 1210 nm that corresponds to the second overtone of the CH_2 stretching mode [53, 54] and another band between 1612 and 1672 nm with a peak selected at 1638 nm that corresponds to the first overtone region of the CH stretching vibration [55], the attributes selected with the GILS algorithms (10 selected features) and GCLS (23 selected features) differ considerably from those obtained by RF/Boruta. A probable reason is the consideration of another search region.

In order to evaluate the significance and impact of the attributes selected to the response, CART decision trees with fixed hyperparameters were fitted in a 10-

fold cross-validation regime for each of the subsets obtained in Table **3.1**. Quality parameters of the fitted regression trees are shown in Table **3.2**.

Table 3.2. Quality parameters of the regression trees with the subsets of attributes selected by BRFR, BRFZ, GCLS and GILS.

	Training Set				Test Set			
Method	N	R2	RMSE	MAE	n	R2	RMSE	MAE
BRFR	48 (80%)	0.8111	0.446	0.345	12 (20%)	0.8965	0.277	0.197
BRFZ		0.8078	0.448	0.345		0.8965	0.277	0.197
GILS		0.8163	0.446	0.338		0.8965	0.277	0.197
GCLS		0.4598	0.827	0.667		0.2039	0.836	0.609

The best subset of attributes is the one that lessens the input space dimensionality without, however, losing relevant information that affects the model predictability. In this sense, it is notable that the subsets of attributes extracted from the BRFR and GILS methods are equivalent in efficiency and should be preferably used to solve this problem, with the difference that Boruta is much easier to implement and does not depend on a generating function. The BRFZ method proved to be as effective as the other two, even though it has selected 13 attributes more. This happens because of the greater sensitivity of the measure of importance based on Z score since this index takes into account the small fluctuations in the loss of average accuracy of each tree in the forest [45], therefore, probabilistically, more factors are likely to affect the overall accuracy.

On the other hand, the subset extraction method based on GCLS proved to be inadequate for the selection of attributes obtained by NIR by the exclusion of factors that have relevance, thus resulting in the loss of accuracy of prediction. An explanation for the low efficiency of the GCLS lies in the fact that the author seeks to optimize the elements based on the Lambert-Beer equation, which takes into account ideal measures performed under a linear dynamic range, which apparently is not the case.

3.6. ROBUSTNESS OF TREE-BASED ALGORITHMS TO NOISE

In the previous section, we found that the linear solution based on Lambtert-Beer's law, using a GCLS optimization algorithm, was not sufficiently effective to select relevant variables from the signals obtained experimentally by diffuse reflectance. Similarly, the other variants of the infrared spectroscopy technique will present the same dilemma. This is due to deviations from ideality (non-linearity) in the measurements, for example, the result of intermolecular and intramolecular

interactions (*e.g*., hydrogen bonds) [56, 57], can lead to peak displacement and signal suppression or overestimation, sample heterogeneity, beam scattering due to changes in refractive indices, among others [56]. Regulatory or calibration failures, alignment and optical system problems [58] (which are quite common in ATR-FTIR) can also occur. Those are called matrix effects and are inherent to the complexity of the samples, while the latter are called system effects [56] and are characterized as some of the potential operational errors observed in laboratory practice.

The matrix effect can be minimized by, for example, extracting the substances of interest from the mixture, diluting the samples or doing a cleanup pretreatment, which is known as "sample preparation" techniques, or by refining the spectrum by applying filters to smooth background noise, regularize the baseline, or remove spurious peaks [59]. Often these measures are applied together. Even so, after implementing these basic measures, several machine learning algorithms may not be effective in describing most of the variance observed in the results. Therefore, there is a need for a careful selection of the tool to be applied since the matrix effect and the other undetermined errors cannot be completely mitigated. In this sense, we are interested in an efficient and robust technique for building chemometric models aimed at solving problems related to high dimensional space obtained by infrared spectroscopy.

In this section, we compare the robustness of popular chemometric techniques such as multiple linear regression and partial least squares regression with three tree-based tools, CART, RF and GBT. For that purpose, regression models were fitted from the subset presented in the previous section with the selection of variables made by BRFR. A noise normally distributed around the averages of each attribute, with a variance of 0.1, was randomly introduced into the elements in 5 levels: 1, 5, 10, 35 and 50%, to artificially mimic the matrix effect and the indeterminate errors. The results were computed by the root mean square error metric (RMSE) and illustrated in Fig. (**3.3**).

Without the inclusion of noise, the regression models have very close predictive power. Much of the correlation between the variables and the response is lost by corrupting 10% of the data with noise. PLS and CART were strongly impacted with the included variance, with increases of 68 and 78% in the RMSE, respectively. GBT had an attenuated increase in that interval, while the RF remained insensitive to the increase in noise level. Great lack of fit was registered after the corruption of 50% of the input space for all regression models. The final RMSEs were 1.271, 1.366, 0.845, 0.803 and > 70 for CART, PLS, GBT, RF and multilinear regression with interactions (results not shown) respectively. The results found for multilinear regression in this work were even higher than those

reported by Saseendran *et al*. [60] in another data structure. The authors found an increase of 3.2% in the RMSE for every 10% of noise included for linear and polynomial regression with ridge regularization.

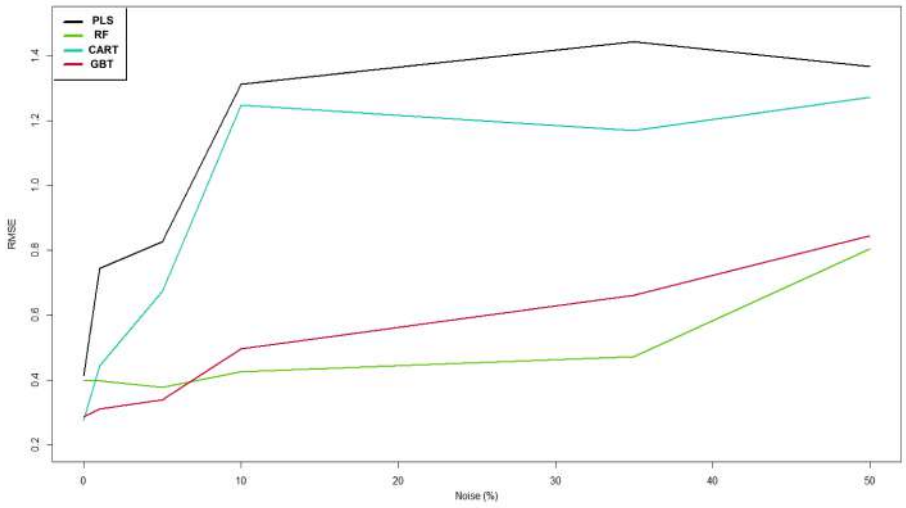

Fig. (3.3). Behavior of different models by including different noise levels.

Tree-based tools that employ some resampling strategy, such as bagging and boosting, and using a large set of weak predictors (trees) in their ensemble proved to be more robust to noise. In particular, the random forest, which when using bagging mechanism to bootstrap subsets, causes at least 1/3 of all attributes to not be used in the formation of trees when using them in the out of bag (OOB) error estimate [45], diluting the number of corrupted attributes between the subsamples and the space used in estimating the error. Also, as the RF uses a limited and small number of attributes in the growth of each tree within the forest, there is a low probability that a tree will grow with all the variables corrupted, and even if it is the case, the vote for these few trees is irrelevant compared to hundreds of other trees. Thus, there is a high probability that most corrupted attributes will be ignored, remaining fitted to the real values even if most of the attributes in total were strongly altered (Fig. **3.4**). In the extreme case (50% of noise), the number of corrupted attributes used in the generation of predictors exceeds the number of out bagging attributes in the ensemble, leading to significant losses in predictability. For example, if the number of random variables used for the growth of a tree (mtry) is 6, it can be said that at least, generically, that 3 of these will be corrupted by noise.

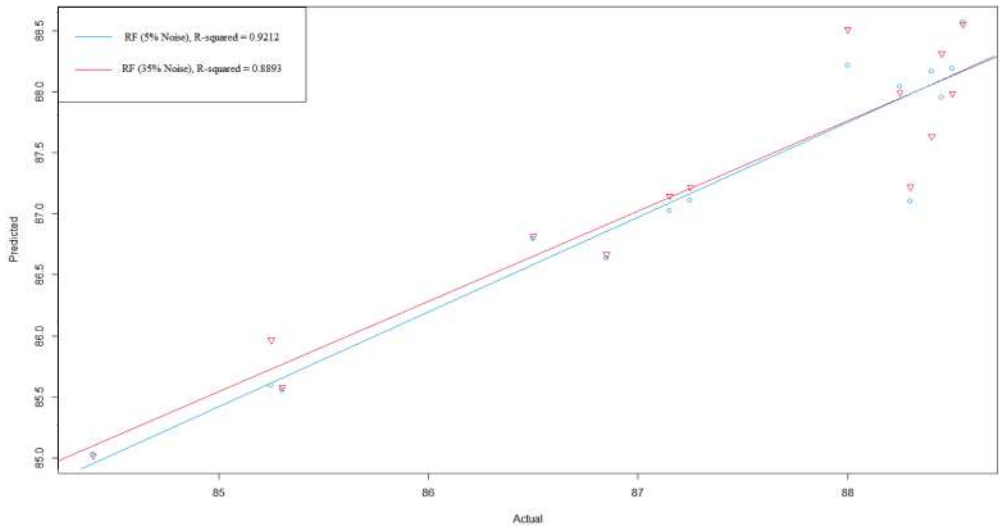

Fig. (3.4). Comparison between the real values and the values predicted by random forest at noise levels of 5 and 35%.

Therefore, RF and GBT are two techniques strongly recommended for the analysis of samples of complex matrices such as biological samples (*e.g.*, blood, urine, saliva, tissues), environmental (*e.g.*, soil, water, mud), food, fuel, polymers, among many others, because they are less sensitive to the sample's spurious nature, with the advantage that RF is a tool that requires less computational cost when compared to tools that employ the boosting mechanism.

3.7. TREE-BASED ALGORITHMS IN DISCRIMINANT ANALYSIS

In several areas of science that deal with the analytical aspect, such as forensics, environmental, pharmaceutical, food, the qualitative or semi-qualitative analysis of a certain contaminant in relation to a limited reference value can be as relevant as the continuous quantification. We recently published an article focusing on the determination of trace levels of tylosin in milk using the Fourier Transform Infrared Spectroscopy (FTIR) technique associated with chemometric techniques [28]. Tylosin is a macrolide antibiotic, a residue from the milk and meat industry, used in the management of cattle for the treatment of bacterial infections such as mastitis and liver abscesses [61].

Before the recent ban [62], the Brazilian Ministry of Agriculture, Livestock and Food Supply (MAPA) stated the maximum residue limit (MRL) of tylosin as 50 $\mu g\ Kg^{-1}$ (50 ppb - parts per billion). It was understood that values above the MRL

could have negative effects on consumer health [63]. On the other hand, values below 50 ppb were considered permissive. It was in our interest to classify the samples as suitable for consumption (tylosin concentrations < 50 ppb) or unsuitable for consumption (tylosin concentrations ≥ 50 ppb). For this, the multilayer perceptron neural network (MLP) technique was used.

The MLP operated in multistart mode, obtaining a better fit with a network with architecture consisting of two hidden layers (6;3), using a logistical activation function. A multiclass factor that corresponds to the origin of the milk was included in the data set due to the effect related to the complexity of the matrix (including interferences such as sugars, proteins, fat, urea and other soluble solids) alongside other 16 wave numbers (cm^{-1}) of interest. The performance in the discriminant analysis of real milk samples by several machine learning algorithms, logistic regression (LogR), partial least squares discriminant analysis (PLS-DA), CART, GBT and RF are compared below with the MLP reference previously developed (Table **3.3**).

Table 3.3. Quality parameters of the discriminatory analysis of a test subset containing 33 milk samples by the logistic regression model (LogR), PLS-DA, CART, GBT, RF and MLP.

Model	True +	False +	True -	False -	Accuracy	CI -	CI +	Kappa	Sensitivity	Specicity	p
LogR	8	7	6	12	0.424	0.255	0.608	-0.13	0.461	0.400	0.989
PLS-DA	17	11	2	3	0.576	0.392	0.745	0.004	0.154	0.850	0.706
CART	20	13	0	0	0.606	0.421	0.771	0	0	1.000	0.575
GBT	18	2	11	2	0.878	0.718	0.966	0.746	0.846	0.900	6.02E-4
RF	18	1	12	2	0.909	0.757	0.981	0.812	0.923	0.900	1.16E-4
MLP [28]	20	0	13	0	1.000	0.894	1.000	1.000	1.000	1.000	6.65E-8

Taking into account the low concentrations of the contaminant, it is expected that the matrix effect will become very prevalent. For the same reasons discussed in the previous section, GBT and RF obtained satisfactory results in predicting the classes compared to the other algorithms, obtaining only two false negatives and a prediction accuracy > 0.85, with a confidence interval, CI, which includes the level of accuracy obtained by MLP. Classification trees, on the other hand, was insensitive to the negative class (C < 50 ppb) and irrelevant results were obtained with the parametric algorithms (LogR and PLS-DA). This is a strong indication of the trend of the non-linear distribution of the response. Only the last three models, including the two techniques based on an ensemble of decision trees with a resampling mechanism, produced statistically significant results ($p < 0.05$).

CONCLUSION

The association of spectroscopy and multivariate calibration methods has enabled the analysis of complex spectra of multicomponent systems, and for that reason, in this chapter, we present some tree-based tools and highlight features of interest compared to other machine learning algorithms. The selection of subsets of attributes by a wrapper algorithm based on random forest, Boruta, proved to be as effective in selecting variables as evolutionary algorithms that are very popular among specialists. In regression problems in the high dimensional space, the ensembles of trees such as random forest and gradient boosting trees proved to be more predictive than the chemometric techniques already established by the literature and, in particular, the random forests proved robust to noise at equal or higher levels to 35%. In the discriminatory analysis of a trace-level contaminant in a complex matrix, RF and GBT proved to be as predictive as multilayer perceptron neural networks, with the advantage that the first two are less computationally costly.

NOTES

The method based on genetic regression (GR) was also presented. However, since the author chose to present a genetic expression instead of the pure attributes, we chose to suppress these findings here.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] J.B. Ghasemi, and H. Tavakoli, "Application of random forest regression to spectral multivariate calibration", *Anal. Methods,* vol. 5, no. 7, pp. 1863-1871, 2013. [http://dx.doi.org/10.1039/c3ay26338j]

[2] A.M. Jiménez-Carvelo, A. González-Casado, M.G. Bagur-González, and L. Cuadros-Rodríguez, "Alternative data mining/machine learning methods for the analytical evaluation of food quality and authenticity - A review", *Food Res. Int.,* vol. 122, pp. 25-39, 2019. [http://dx.doi.org/10.1016/j.foodres.2019.03.063] [PMID: 31229078]

[3] V. Svetnik, A. Liaw, C. Tong, J.C. Culberson, R.P. Sheridan, and B.P. Feuston, "Random forest: a classification and regression tool for compound classification and QSAR modeling", *J. Chem. Inf.*

Comput. Sci., vol. 43, no. 6, pp. 1947-1958, 2003.
[http://dx.doi.org/10.1021/ci034160g] [PMID: 14632445]

[4] F.A. Vega, J.M. Matías, M.L. Andrade, M.J. Reigosa, and E.F. Covelo, "Classification and regression trees (CARTs) for modelling the sorption and retention of heavy metals by soil", *J. Hazard. Mater.,* vol. 167, no. 1-3, pp. 615-624, 2009.
[http://dx.doi.org/10.1016/j.jhazmat.2009.01.016] [PMID: 19200658]

[5] S.M. Tan, J. Jiao, X-L. Zhu, Y-P. Zhou, D-D. Song, H. Gong, and R-Q. Yu, "QSAR studies of a diverse series of antimicrobial agents against Candida albicans by classification and regression trees", *Chemom. Intell. Lab. Syst.,* vol. 103, no. 2, pp. 184-190, 2010.
[http://dx.doi.org/10.1016/j.chemolab.2010.07.005]

[6] D. Custers, T. Cauwenbergh, J.L. Bothy, P. Courselle, J.O. De Beer, S. Apers, and E. Deconinck, "ATR-FTIR spectroscopy and chemometrics: An interesting tool to discriminate and characterize counterfeit medicines", *J. Pharm. Biomed. Anal.,* vol. 112, pp. 181-189, 2015.
[http://dx.doi.org/10.1016/j.jpba.2014.11.007] [PMID: 25476739]

[7] L. Rokach, "Decision forest: Twenty years of research", *Inf. Fusion,* vol. 27, pp. 111-125, 2016.
[http://dx.doi.org/10.1016/j.inffus.2015.06.005]

[8] F.B. de Santana, W. Borges Neto, and R.J. Poppi, "Random forest as one-class classifier and infrared spectroscopy for food adulteration detection", *Food Chem.,* vol. 293, pp. 323-332, 2019.
[http://dx.doi.org/10.1016/j.foodchem.2019.04.073] [PMID: 31151619]

[9] X. Jin, S. Li, W. Zhang, J. Zhu, and J. Sun, "Prediction of soil-available potassium content with visible near-infrared ray spectroscopy of different pretreatment transformations by the boosting algorithms", *Appl. Sci. (Basel),* vol. 10, no. 4, p. 1520, 2020.
[http://dx.doi.org/10.3390/app10041520]

[10] J. Xiu, Z. Xianzhi, L. Shaowen, W. Wencai, and Q. Haijun, "Predicting soil available phosphorus by hyperspectral regression method based on gradient boosting decision tree", *Jiguang Yu Guangdianzixue Jinzhan,* vol. 56, no. 13, p. 131102, 2019.
[http://dx.doi.org/10.3788/LOP56.131102]

[11] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning.* vol. 103. Springer New York: New York, NY, 2013.
[http://dx.doi.org/10.1007/978-1-4614-7138-7]

[12] L. Rokach, and O. Maimon, Decision trees.*Data mining and knowledge discovery handbook.* Springer: Boston, MA, 2005, pp. 165-192.
[http://dx.doi.org/10.1007/0-387-25465-X_9]

[13] R.S. DeFries, M. Hansen, J.R.G. Townshend, and R. Sohlberg, "Global land cover classifications at 8 km spatial resolution: The use of training data derived from Landsat imagery in decision tree classifiers", *Int. J. Remote Sens.,* vol. 19, no. 16, pp. 3141-3168, 1998.
[http://dx.doi.org/10.1080/014311698214235]

[14] K. Zhang, X. Wu, R. Niu, K. Yang, and L. Zhao, "The assessment of landslide susceptibility mapping using random forest and decision tree methods in the Three Gorges Reservoir area, China", *Environ. Earth Sci.,* vol. 76, no. 11, pp. 404-424, 2017.
[http://dx.doi.org/10.1007/s12665-017-6731-5]

[15] E. Pekel, M.C. Akkoyunlu, M.T. Akkoyunlu, and S. Pusat, "Decision tree regression model to predict low-rank coal moisture content during convective drying process", *Int. J. Coal Prep. Util.,* vol. 40, no. 8, pp. 505-512, 2020.
[http://dx.doi.org/10.1080/19392699.2020.1737527]

[16] O.F. Althuwaynee, A. Balogun, and W. Al Madhoun, "Air pollution hazard assessment using decision tree algorithms and bivariate probability cluster polar function: evaluating inter-correlation clusters of PM10 and other air pollutants", *GIsci. Remote Sens.,* vol. 57, no. 2, pp. 207-226, 2020.
[http://dx.doi.org/10.1080/15481603.2020.1712064]

[17] A. Fielding, and C.A. O'Muircheartaigh, "Binary Segmentation in Survey Analysis with Particular Reference to AID", *Statistician,* vol. 26, no. 1, p. 28, 1977.
[http://dx.doi.org/10.2307/2988216]

[18] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, *Classification and Regression Trees.* 1st ed. CRC Press: Boca Raton, 1984.

[19] D.R. Cutler, T.C. Edwards Jr, K.H. Beard, A. Cutler, K.T. Hess, J. Gibson, and J.J. Lawler, "Random forests for classification in ecology", *Ecology,* vol. 88, no. 11, pp. 2783-2792, 2007.
[http://dx.doi.org/10.1890/07-0539.1] [PMID: 18051647]

[20] U. Grömping, "Variable importance assessment in regression: Linear regression versus random forest", *Am. Stat.,* vol. 63, no. 4, pp. 308-319, 2009.
[http://dx.doi.org/10.1198/tast.2009.08199]

[21] V.F. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J.P. Rigol-Sanchez, "An assessment of the effectiveness of a random forest classifier for land-cover classification", *ISPRS J. Photogramm. Remote Sens.,* vol. 67, no. 1, pp. 93-104, 2012.
[http://dx.doi.org/10.1016/j.isprsjprs.2011.11.002]

[22] L. Breiman, "Random forests", *Mach. Learn.,* vol. 45, no. 1, pp. 5-32, 2001.
[http://dx.doi.org/10.1023/A:1010933404324]

[23] Y. Ren, L. Zhang, and P.N. Suganthan, "Ensemble Classification and Regression-Recent Developments, Applications and Future Directions [Review Article]", *IEEE Comput. Intell. Mag.,* vol. 11, no. 1, pp. 41-53, 2016.
[http://dx.doi.org/10.1109/MCI.2015.2471235]

[24] A.L. Boulesteix, S. Janitza, J. Kruppa, and I.R. König, "Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics", *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.,* vol. 2, no. 6, pp. 493-507, 2012.
[http://dx.doi.org/10.1002/widm.1072]

[25] M.W. Ahmad, M. Mourshed, and Y. Rezgui, "Trees vs. Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption", *Energy Build.,* vol. 147, pp. 77-89, 2017.
[http://dx.doi.org/10.1016/j.enbuild.2017.04.038]

[26] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?", *J. Mach. Learn. Res.,,* vol. 15, no. 2014, pp. 3133-3181, 2014.

[27] J.H. Kalivas, "Two data sets of near infrared spectra", *Chemom. Intell. Lab. Syst.,* vol. 37, no. 2, pp. 255-259, 1997.
[http://dx.doi.org/10.1016/S0169-7439(97)00038-5]

[28] A.G. Freitas, B.E. Magalhães, L.A. Minho, D.J. Leão, L.S. Santos, and S. Fernandes, "FTIR spectroscopy with chemometrics for determination of tylosin residues in milk", *J. Sci. Food Agric.,* no. Oct, 2020.
[http://dx.doi.org/10.1002/jsfa.10799] [PMID: 32901945]

[29] K. Kara, D. Alistarh, G. Alonso, O. Mutlu, and C. Zhang, "FPGA-accelerated dense linear machine learning: A precision-convergence trade-off", *Proceedings - IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines, FCCM,* p. 167, 2017.
[http://dx.doi.org/10.1109/FCCM.2017.39]

[30] R.E. Bellman, *Adaptive Control Processes: A Guided Tour.* vol. Vol. 1. 1st ed. Princeton University Press: Princeton, 1961.
[http://dx.doi.org/10.1515/9781400874668]

[31] V. Charles, J. Aparicio, and J. Zhu, "The curse of dimensionality of decision-making units: A simple approach to increase the discriminatory power of data envelopment analysis", *Eur. J. Oper. Res.,* vol.

279, no. 3, pp. 929-940, 2019.
[http://dx.doi.org/10.1016/j.ejor.2019.06.025]

[32] M. Verleysen, and D. François, "The curse of dimensionality in data mining and time series prediction", *Lect. Notes Comput. Sci.,* vol. 3512, pp. 758-770, 2005.
[http://dx.doi.org/10.1007/11494669_93]

[33] L. Lerman, R. Poussier, G. Bontempi, O. Markowitch, and F.X. Standaert, *Template attacks vs. Machine learning revisited (and the curse of dimensionality in side-channel analysis)*, 2015.
[http://dx.doi.org/10.1007/978-3-319-21476-4_2]

[34] M. Ris, J. Barrera, and D.C. Martins Jr, "U-curve: A branch-and-bound optimization algorithm for U-shaped cost functions on Boolean lattices applied to the feature selection problem", *Pattern Recognit.,* vol. 43, no. 3, pp. 557-568, 2010.
[http://dx.doi.org/10.1016/j.patcog.2009.08.018]

[35] H. Hoffmann, S. Schaal, and S. Vijayakumar, "Local dimensionality reduction for non-parametric regression", *Neural Process. Lett.,* vol. 29, no. 2, pp. 109-131, 2009.
[http://dx.doi.org/10.1007/s11063-009-9098-0]

[36] L. Shi, J.A. Westerhuis, J. Rosén, R. Landberg, and C. Brunius, "Variable selection and validation in multivariate modelling", *Bioinformatics,* vol. 35, no. 6, pp. 972-980, 2019.
[http://dx.doi.org/10.1093/bioinformatics/bty710] [PMID: 30165467]

[37] J.M. Peña, and R. Nilsson, "On the complexity of discrete feature selection for optimal classification", *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 32, no. 8, pp. 1517-1522, 2010.
[http://dx.doi.org/10.1109/TPAMI.2010.84] [PMID: 20558881]

[38] M. B. Kursa, and W. R. Rudnicki, "The all relevant feature selection using random forest", 2011.

[39] K.J. Archer, and R.V. Kimes, "Empirical characterization of random forest variable importance measures", *Comput. Stat. Data Anal.,* vol. 52, no. 4, pp. 2249-2260, 2008.
[http://dx.doi.org/10.1016/j.csda.2007.08.015]

[40] S. Nembrini, I.R. König, and M.N. Wright, "The revival of the Gini importance?", *Bioinformatics,* vol. 34, no. 21, pp. 3711-3718, 2018.
[http://dx.doi.org/10.1093/bioinformatics/bty373] [PMID: 29757357]

[41] B. Gregorutti, B. Michel, and P. Saint-Pierre, "Correlation and variable importance in random forests", *Stat. Comput.,* vol. 27, no. 3, pp. 659-678, 2017.
[http://dx.doi.org/10.1007/s11222-016-9646-1]

[42] I. Guyon, and A. Elisseeff, "An Introduction to Variable and Feature Selection", *J. Mach. Learn. Res.,* vol. 3, pp. 1157-1182, 2003.

[43] R. Kohavi, and G.H. John, "Wrappers for feature subset selection", *Artif. Intell.,* vol. 97, no. 1–2, pp. 273-324, 1997.
[http://dx.doi.org/10.1016/S0004-3702(97)00043-X]

[44] W.R. Rudnicki, M. Kierczak, J. Koronacki, and J. Komorowski, "A statistical method for determining importance of variables in an information system", *International Conference on Rough Sets and Current Trends in Computing,* vol. vol. 4259, 2006pp. 557-566
[http://dx.doi.org/10.1007/11908029_58]

[45] M.B. Kursa, A. Jankowski, and W.R. Rudnicki, "Boruta - A system for feature selection", *Fundam. Inform.,* vol. 101, no. 4, pp. 271-285, 2010.
[http://dx.doi.org/10.3233/FI-2010-288]

[46] M.B. Kursa, and W.R. Rudnicki, "Feature selection with the boruta package", *J. Stat. Softw.,* vol. 36, no. 11, pp. 1-13, 2010.
[http://dx.doi.org/10.18637/jss.v036.i11]

[47] J. Sherman, *Storytelling: An Encyclopedia of Mythology and Folklore, 1st ed* vol. 1. , 2008.

[48] D. Özdemir, "Determination of octane number of gasoline using near infrared spectroscopy and genetic multivariate calibration methods", *Petrol. Sci. Technol.,* vol. 23, no. 9–10, pp. 1139-1152, 2005.
[http://dx.doi.org/10.1081/LFT-200035547]

[49] W. Siedlecki, and J. Sklansky, "A note on genetic algorithms for large-scale feature selection", *Pattern Recognit. Lett.,* vol. 10, no. 5, pp. 335-347, 1989.
[http://dx.doi.org/10.1016/0167-8655(89)90037-8]

[50] S. Li, K. Zhang, Q. Chen, S. Wang, and S. Zhang, "Feature Selection for High Dimensional Data Using Weighted K-Nearest Neighbors and Genetic Algorithm", *IEEE Access,* vol. 8, pp. 139512-139528, 2020.
[http://dx.doi.org/10.1109/ACCESS.2020.3012768]

[51] M.L. Raymer, W.F. Punch, E.D. Goodman, L.A. Kuhn, and A.K. Jain, "Dimensionality reduction using genetic algorithms", *IEEE Trans. Evol. Comput.,* vol. 4, no. 2, pp. 164-171, 2000.
[http://dx.doi.org/10.1109/4235.850656]

[52] C.F. Tsai, W. Eberle, and C.Y. Chu, "Genetic algorithms in feature and instance selection", *Knowl. Base. Syst.,* vol. 39, pp. 240-247, 2013.
[http://dx.doi.org/10.1016/j.knosys.2012.11.005]

[53] H.L. Fernandes, I.M. Raimundo Jr, C. Pasquini, and J.J.R. Rohwedder, "Simultaneous determination of methanol and ethanol in gasoline using NIR spectroscopy: effect of gasoline composition", *Talanta,* vol. 75, no. 3, pp. 804-810, 2008.
[http://dx.doi.org/10.1016/j.talanta.2007.12.025] [PMID: 18585150]

[54] M.L.S. Albuquerque, ""Characterization of Buriti (Mauritia flexuosa L.) oil by absorption and emission spectroscopies," *J. Braz. Chem. Soc.*, vol. 16, no. 6 A",

[55] I. Swaid, D. Nickel, and G.M. Schneider, "NIR-spectroscopic investigations on phase behaviour of low-volatile organic substances in supercritical carbon dioxide", *Fluid Phase Equilib.,* vol. 21, no. 1–2, pp. 95-112, 1985.
[http://dx.doi.org/10.1016/0378-3812(85)90062-7]

[56] J.D. Blando, R.J. Porcja, and B.J. Turpin, "Issues in the Quantitation of Functional Groups by FTIR Spectroscopic Analysis of Impactor-Collected Aerosol Samples", *Aerosol Sci. Technol.,* vol. 35, no. 5, pp. 899-908, 2001.
[http://dx.doi.org/10.1080/02786820126852]

[57] M. Dos Santos Silva, "Fourier transform infrared spectroscopy, thermogravimetric analysis, scanning electron microscopy as supporting tools in quality control of antiparasitics", *Quim. Nova,* vol. 41, no. 3, pp. 258-267, 2018.
[http://dx.doi.org/10.21577/0100-4042.20170168]

[58] D.M. Haaland, R.G. Easterling, and D.A. Vopicka, "Multivariate Least-Squares Methods Applied To The Quantitative Spectral Analysis Of Multicomponent Samples", *Appl. Spectrosc.,* vol. 39, no. 1, pp. 73-84, 1985.
[http://dx.doi.org/10.1366/0003702854249376]

[59] W. Zhou, S. Yang, and P.G. Wang, "Matrix effects and application of matrix effect factor", *Bioanalysis,* vol. 9, no. 23, pp. 1839-1844, 2017.
[http://dx.doi.org/10.4155/bio-2017-0214] [PMID: 29171768]

[60] A. T. Saseendran, L. Setia, V. Chhabria, D. Chakraborty, and A. B. Roy, *Impact of Noise in Dataset on Machine Learning Algorithms,* .
[http://dx.doi.org/10.13140/RG.2.2.25669.91369]

[61] T.G. Nagaraja, A.B. Beharka, M.M. Chengappa, L.H. Carroll, A.P. Raun, S.B. Laudert, and J.C. Parrott, "Bacterial flora of liver abscesses in feedlot cattle fed tylosin or no tylosin", *J. Anim. Sci.,* vol. 77, no. 4, pp. 973-978, 1999.

[http://dx.doi.org/10.2527/1999.774973x] [PMID: 10328365]

[62] MAPA, *INSTRUÇÃO NORMATIVA No 1, DE 13 DE JANEIRO DE* , 2020. Brasília, 2020.

[63] L.D.C. Luiz, M.J.V. Bell, R.A. Da Rocha, N.L. Leal, and V.D.C. Dos Anjos, "Detection of Veterinary Antimicrobial Residues in Milk through Near-Infrared Absorption Spectroscopy", *J. Spectrosc.,* vol. 2018, pp. 1-6, 2018.
[http://dx.doi.org/10.1155/2018/5152832]

CHAPTER 4

Applications of Deep Learning in Medical Engineering

Sumit Kumar Jindal[1,*], **Sayak Banerjee**[1], **Ritayan Patra**[1] and **Arin Paul**[1]

[1] *School of Electronics Engineering, Vellore Institute of Technology, Tamil Nadu, India*

Abstract: As a result of considerable breakthroughs in the field of artificial intelligence, deep learning has achieved exceptional success in resolving issues.This work brings forth a historical overview of deep learning and neural networks and further discusses its applications in the domain of medical engineerings - such as detection of brain tumours, sleep apnea, arrhythmia detection, *etc*.

One of the most important and mysterious organs of our body is the brain. Like any other organ, our brain may suffer from various life-threatening diseases like brain tumours which can be malignant or benign. Analysis of the brain MRI images by applying convolution neural networks or artificial neural networks can automate this process by classifying these images into various types of tumours. A faster and more effective method can be provided by this method for detecting the disease at a key stage from where recovery is possible.

Sleep apnea is a sleeping disorder involving irregular breathing. The brain detects a sudden decrease in the level of oxygen and sends a signal to wake the person up while he is sleeping. Cardiac arrhythmia refers to a group of conditions that causes the heart to beat irregularly, too slowly, or too quickly, *e.g*., atrial fibrillation. Deep learning along with bio-medical signal and audio processing techniques on respiratory sound datasets and ECG datasets have huge potential in the detection of these diseases. Deep learning outperforms the existing detection algorithms and a good amount of effort on feature engineering, augmentation techniques, and building effective filters can get a high accuracy result.

Keywords: Artificial intelligence, Artificial neural networks, Atrial fibrillation, Automation, Audio processing, Brain tumours, Bio-medical signal processing, Cardiac arrythmia, Convolution neural networks, Deep learning, ECG, EEG, Feature engineering, Machine learning, MRI imaging, Neural networks, Optimization, Signal processing, Signal analysis, Sleep apnea.

[*] **Corresponding author Sumit Kumar Jindal:** School of Electronics Engineering, Vellore Institute of Technology, Tamil Nadu, India; Tel: +918603559888; E-mail: sumitjindal08@gmail.com

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

4.1. HISTORICAL OVERVIEW OF DEEP LEARNING

4.1.1. Machine Learning

Machine learning can be defined as methods used to make a computer learn and automate various tasks without being explicitly programmed [1]. Tom Mitchell provides a modern definition of machine learning. "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at Tasks T, as measured by P, improves with experience E." These methods or algorithms work on the data, learn important features from that data, and then apply these learned features to make important decisions to new sets of data. For example, machine learning algorithms are used in online music streaming services or video streaming services to make a recommendation to its users based on the user's listening preferences.

In earlier days, when the penetration of internet in the society and the use of technology in everyday life was less, the amount of data generated was less, and the training of machine learning algorithms on such a small amount of data led to problems known as overfitting. Overfitting decreases the accuracy of machine learning algorithms as it performs poorly on test data. So, researchers, data scientists, and machine learning engineers would primarily focus on how to solve this problem of overfitting.

But with time, the size of datasets began to increase. Nowadays, several GBs of data can be found on various topics and the problem of overfitting began to disappear as the dimensionality of the data began to increase. But processing such a huge dataset using machine learning algorithms is not possible as the learning capacity gets saturated, which leads to underfitting. This happens because machine learning algorithms use shallow structures.

The deep neural networks used in deep learning algorithms have large learning capacities as they can handle a large number of parameters. When compared to machine learning techniques, the performance of deep learning owing to overfitting is either the same or worse. So, whenever the dataset is huge, the use of deep learning techniques is most suitable. They increase the learning capacity of the models as they have better optimization techniques and use large computational resources.

4.1.2. Neural Networks

A neural network is a web of neurons [2]. In biology, these are similar to the neurons present in the brains of any living organisms, while in computer science,

these are artificial neurons used to solve complex mathematical, statistical, or artificial intelligence problems. These artificial neurons mimic the working of the human brain. The deep learning algorithms use these deep neural networks, as shown in Fig. (**4.1**), to train the model.

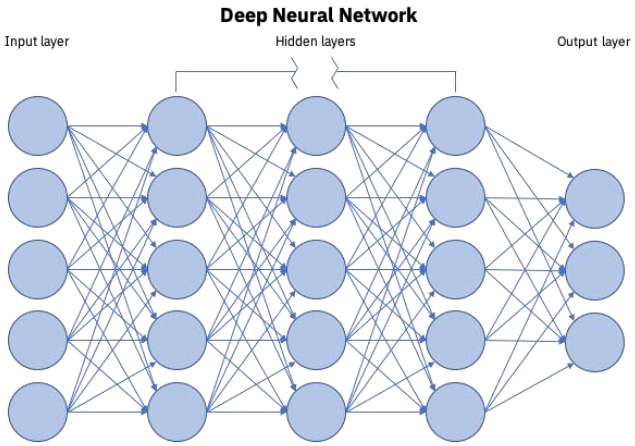

Fig. (4.1). Architecture of a Deep Neural Network.

Neural networks must include numerous layers, not just one or two, to answer difficult real-world artificial intelligence concerns.In Fig. **4.1**, the neurons are arranged in various layers, which are in turn connected to adjacent layers with some numerical weights. In feedforward operations, the neurons in each layer receive some signal and transfer it to the neurons in the adjacent layer after performing some mathematical calculations on the input and passing it through an activation function [3].

In each hidden neuron j, the weighted sum of input neurons is first calculated as in Equation 1a:

$$Net_j = \sum_{i=1}^{d} x_i w_{ji} + w_{j0} \quad \textbf{(1a)}$$

Where Net_j is the total activation value of the neuron, w_{ji} are the weights of connections between the input layer and the hidden layer, w_{j0} are the bias terms associated with each neuron, and x_i is the input value. This value is then passed through a special activation function like $y = g(Net_j)$, which is used to remove the linearity in the data. If the total activation value is greater than the threshold value of the activation function, then the value is transmitted to the next neuron; otherwise, it will not get transferred.

In a multi-layer neural network, the final output looks something like the one shown in Equation 1b –

$$f(x) = g(\sum_{j=1}^{nH} Wkj\ g(\sum_{i=1}^{d} wjig(\dots) + wj0) + wk0) \quad \textbf{(1b)}$$

Another important process that is applied is Backpropagation. It is one of the most important training algorithms in deep learning. It is used to reduce the error rate in the previous epochs. It is done by choosing the proper weight of each neuron in order to increase the accuracy, making the deep learning model more reliable and closer to real-world scenarios. When the feed-forward operation is applied, most of the time, it generates some loss that is fed backwards in the neural network during backpropagation, helping fine tune the weights for the next epoch in order to get a better result (or reduced loss). It uses an optimization function that is different for each deep learning algorithm to determine in which way it will adjust the weights in order to reduce the loss from the present value and produce a better result that is closer to the target value [2, 3].

4.1.3. Deep Learning

Deep learning is a subpart of machine learning. Deep Learning, in simple terms, can be defined as a combination of neural networks and algorithms which consume raw input data from users and performs nonlinear mathematical transformations on the given input data in order to reach the desired output with high accuracy [4].

In 1943, Walter Pitts and Warren McCulloch created a computer model of neural network, which acts like the human brain. Henry J. Kelley developed the continuous Back Propagation Model in 1960. Kunihiko Fukushima developed the first convolutional neural network in 1979. He designed a neural network with multiple pooling and convolutional layers. In the late 20th century, the computational power was not high enough to create complex deep learning neural networks. But in the early 2000s, GPUs were a thousand times more powerful than normal CPUs developed and they acted as fuel in the development of highly efficient neural networks. By 2011, the speed of GPUs was sufficient enough to train whole neural networks together instead of training layer by layer. It became clear that with increased computational speed, the deep learning algorithms have a significant edge over other machine learning algorithms in terms of efficiency and speed.

Deep Learning architectures like Convolutional Neural Network (CNN) or Recurrent Neural Network (RNN) are being applied every day in fields like

speech recognition, image processing, bioinformatics, computer vision, natural language processing, *etc*. For example, Google Assistant or Amazon Alexa works on deep learning algorithms based on natural language processing. Deep Learning works much better in situations where labelled data or prior knowledge required is not available. It is considered to bring about a renaissance in technology [5].

4.2. ACTIVATION FUNCTIONS

Artificial Neural Networks (ANN) consist of neurons in large numbers which make mathematical decisions. A standard ANN comprises an input layer where numeric datapoints are feed into the network, next is the hidden layer where decision making is being done and finally the output layer which fetches us the output. Inside the hidden layer each neuron is being associated with some weights which are multiplied with inputs it receives. The product is finally passed through a gateway known as the Activation function.

Activation Functions are simply mathematical equations that determine the output of a neuron in neural networks. Depending on the nature of the activation function used and the input it receives - the neuron gets activated or not which signifies the importance of that input towards the learning of the model. Generally, the output from the activation function is given in normalized form *i.e*., between [0,1] or [-1,1]. Nowadays, non-linear and back-propagation enabled activation functions are highly used [6].

4.2.1. Binary Activation Function

Binary Activation function is defined by Equation (2a):

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad \textbf{(2a)}$$

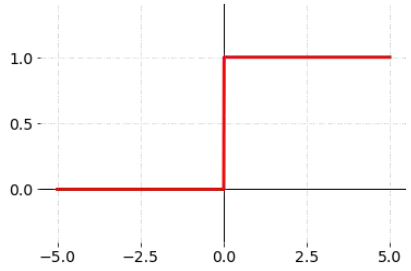

Fig. (4.2). Binary Activation Function.

So, it takes the positive inputs and makes it 1 and all the negative inputs are mapped to 0 as shown in the graph in Fig. (**4.2**) Thus, it is an example of a threshold-based activation function, which is quite helpful in binary classification.

4.2.2. Sigmoid and SoftMax Activation Function

Sigmoid Activation function is defined by Equation (2b) and its derivative given by Equation (2c) respectively:

$$f(x) = \sigma(x) = \frac{1}{1+e^{-x}} \quad \textbf{(2b)}$$

$$f'(x) = f(x)(1 - f(x)) \quad \textbf{(2c)}$$

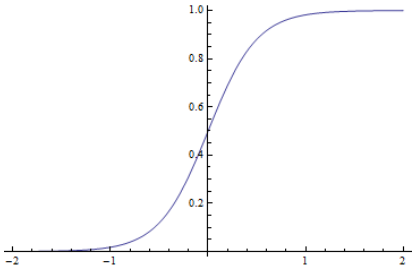

Fig. (4.3). Sigmoid Activation Function.

Sigmoid Activation function is also known as logistic regression function. It is a non-linear activation function whose output ranges from 0 to 1 with the centre at 0.5 as shown in the graph in Fig. (**4.3**). It is highly used in binary classification of mildly scattered inputs. If we notice the graph of sigmoid function, we can see that either end of it, respond less to change in input values thus arising the problem of vanishing gradients [7].

SoftMax Activation function is one of the unique activation functions that is generally used it output layer of Neural network with multi-class classification. It takes a vector of input and generates an output of probability vector where the most probable class is mapped to 1 and others mapped to 0. As we know Sigmoid is used for binary classification whereas SoftMax is used for multi-class classification in logistic regression models. The mathematical expression is defined by Equation (2d).

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}} \quad \textbf{(2d)}$$

4.2.3. TanH Activation Function

TanH Activation - The Hyperbolic Tangent Activation function and its derivative are defined by Equations (2e) and (2f) respectively:

$$f(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad \textbf{(2e)}$$

$$f'(x) = 1 - f(x)^2 \quad \textbf{(2f)}$$

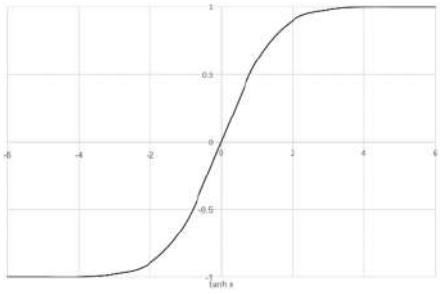

Fig. (4.4). TanH Activation Function.

TanH is a non – linear, zero centred sigmoidal (S – curve) activation function with its range between [-1, 1] as shown in the graph in Fig. (**4.4**). Being zero – centred; it supports backpropagation and provides a better representation to the negative inputs. The gradient in TanH activation function is steeper than that of Sigmoid activation function. It is more extensively used where the inputs are scattered randomly. But it fails to address the vanishing gradient problem which aroused in sigmoid activation function [7].

4.2.4. ReLU and Leaky ReLU Activation Function

ReLU Activation - Rectified Linear Units Activation and its derivative are given by Equations (2g) and (2h) respectively:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad \textbf{(2g)}$$

$$f'(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad \textbf{(2h)}$$

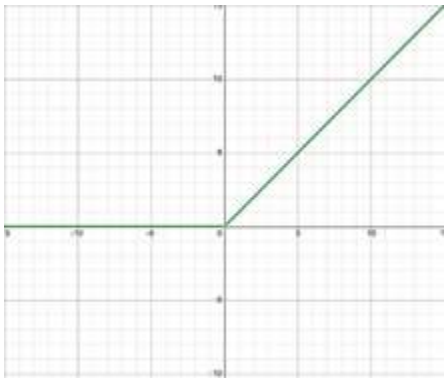

Fig. (4.5). Rectified Linear Unit (ReLU).

ReLU Activation function is non – linear and one of the most popular Activation functions when it comes to deep learning. It is highly preferred over simple logistic regression function due to its advantage of nullifying Vanishing Gradient problem. It provides an output 0 for all negatives input and passes the raw value for positive input as shown in the graph in Fig. (**4.5**). The operation of the ReLU highly implies that of our biological neuron.

But ReLU comes with some drawbacks like the function is not differentiable at zero and another is that of dying ReLU problem which arises due to the presence of negative weights at the time of backpropagation [8]. Leaky ReLU Activation - Leaky Rectified Linear Unit Activation and its derivative are defined by Equations (2i) and (2j) respectively:

$$f(x) = \begin{cases} 0.01x, & x < 0 \\ x, & x \geq 0 \end{cases} \quad \textbf{(2i)}$$

$$f'(x) = \begin{cases} 0.01, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad \textbf{(2j)}$$

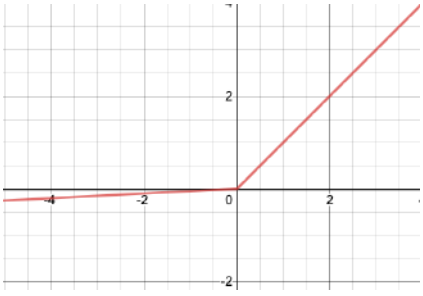

Fig. (4.6). Leaky ReLU.

Leaky ReLU is an improvement of traditional ReLU Activation function. If we compare the graphs, we notice that for input less than 0 the outputs are slightly descending as shown in the graph in Fig. (**4.6**). A slope of 0.01 or any small suitable value is attached to the negative values. This reduces the death of ReLU activated neurons thus attempting to minimize the dying ReLU problem.

4.3. OPTIMIZERS AND LOSS

4.3.1. Optimizers

Suppose we have an Artificial Neural Network comprising an input layer, few hidden layers and an output layer. The output we get from Neural Network consist of loss or errors, which are being reduced using optimizers. Optimizers are algorithms or methods to change the attributes of a Neural Network. It updates weight associated with each node-to-node connection during backpropagation and learning rate dynamically in order to reduce loss and for faster learning.

4.3.1.1. Adagrad

Adagrad stands for Adaptive Gradient Descent. We know that in a traditional gradient descent learning rate (ɳ') is fixed, which poses a threat of overshooting the minima resulting in an increased loss. If we dynamically change the learning rate as the training progress, it gives rise to adaptive gradient descent which happens to be faster and more accurate compared to gradient descent. The formula of Adagrad optimizer is given by Equation (3a) where ɳ' is given by Equation (3b) where át is initial learning rate which is given by Equation (3c) and ϵ is small positive number.

$$W_t = W_{t-1} - \eta'_t \frac{\partial L}{\partial W_{old}} \quad \textbf{(3a)}$$

$$\eta'_t = \frac{\eta}{\sqrt{\alpha_t + \epsilon}} \quad \textbf{(3b)}$$

$$\alpha_t = \sum_{i=1}^{t} \left(\frac{\partial L}{\partial W_t}\right)^2 \quad \textbf{(3c)}$$

4.3.1.2. RMSProp

RMSProp stands for Root Mean Square Propagation, which is quite similar to gradient descent but have momentum with it. It divides the learning rate for a weight by a running average of the magnitudes of recent gradients for that weight.

This restricts its vertical oscillation which facilitate us of choosing a higher learning rate so that it can take longer steps in horizontal direction resulting in faster convergence [9]. It is based upon exponentially weighted average. The formula for weight updation is given by Equation (3d) where Equation (3e) shows the change on S_{dw} and Equation (3f) on η'.

$$W_t = W_{t-1} - \eta'_t \frac{\partial L}{\partial W_{t-1}} \quad \textbf{(3d)}$$

$$S_{dw_t} = \beta S_{dw_{t-1}} + (1-\beta)\left(\frac{\partial L}{\partial W_t}\right) \quad \textbf{(3e)}$$

$$\eta'_t = \frac{\eta}{\sqrt{S_{dw_t} + \epsilon}} \quad \textbf{(3f)}$$

4.3.1.3. Adam Optimizer

Adaptive Moment Estimation popularly known as Adam Optimizer is one the best optimizer present till now. It's a combination of both momentum and RMSprop. Momentum helps in smoothening and RMSprop changes learning rate in an efficient manner [9]. It implements the exponential moving average to scale the learning rate making it efficient and has very little memory requirement. The formula for weight and bias updation formula is given in Equations (3g) and (3h) respectively.

$$W_t = W_{t-1} - \frac{\eta * V_{dw}}{\sqrt{S_{db} + \epsilon}} \quad \textbf{(3g)}$$

$$b_t = b_{t-1} - \frac{\eta * V_{db}}{\sqrt{S_{db} + \epsilon}} \quad \textbf{(3h)}$$

4.3.2. Loss Functions

Neural Network are trained using Stochastic Gradient Descent, which continuously tries to maximize accuracy by updating weights during model training. During the training process we run several epochs on the same data in order to improve accuracy. At the end of each epoch the error is calculated on model prediction on each input provided by using Loss function. It is used to quantify how good or bad the model is performing. Different loss function employs different loss calculation formulas which fits in accordance with the type of learning.

4.3.2.1. Mean Squared Error Loss

Mean Squared Error Loss commonly MSE Loss, is mainly and largely used as a loss function for regression. It is computed as the average squared value of the difference of actual and predicted values. The loss value is always positive, regardless of the sign of actual and predicted values due to squaring of the error. A perfect zero indicates no loss/error *i.e*., a perfect prediction. It is used when we give more importance to larger values compared to smaller ones because it returns large values for outliners. The MSE loss function formula is given by Equation (3i).

$$L(y, \hat{y}) = \frac{1}{N}\sum_{i=0}^{N}(y - \hat{y})^2 \quad \textbf{(3i)}$$

4.3.2.2. Cross – Entropy Loss

Cross – Entropy Loss is also known as Log Loss. It is generally used in logistic regressions or classification problem. Here each probability is being compared with the actual value corresponding to that class and a score is computed that penalizes the probability based on the deviation from the expected value. This deviation is logarithmic, which means it produces a small quantity for smaller differences and larger value for larger difference. It is minimized when smaller value represents better model compared to larger values [10]. A perfect prediction has a cross entropy of zero. Cross Entropy Loss are of two types one being the binary cross entropy and the other one is categorical cross entropy. Binary cross entropy is used when we classify between two classes whereas we use categorical cross entropy for as many numbers of classes used for classification. Therefore, we can say binary cross entropy is a special case of categorical cross entropy, and the equation for loss calculation is same in both cases. The formula for loss calculation for cross entropy is given by Equation (3j).

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i.\log\big(p(y_i)\big) + (1 - y_i).\log(1 - p(y_i)) \quad \textbf{(3j)}$$

4.4. IMAGE RECOGNITION AND CLASSIFICATION

Image Recognition is the process of identification of objects having previously encountered those objects in some events. It is one of the most important tools in today's world. Computational machines with the help of camera and artificial intelligence help us to identify different objects like cars, trees, persons, text, *etc*. from images or videos.

It involves two important steps –

1. Image Detection
2. Image Classification

Image Detection – Also known as Object Detection, is a technique to process the image and identify the objects present inside the image. It does not classify objects but instead it detects different objects that is present in the image.

Image Classification – It is a process of categorizing objects by attaching labels or tags with each object in the image. For Example, if we want to search for car images in the internet, we will instantly get thousands of images of car. The neural network processes the images and detects different objects and attach labels with the images and in this way the search engine provides us with the images of car.

The best way to process image is the use of ***Convolutional Neural Network*** or in short CNN/ConvNet. It is a deep learning algorithm which takes images as input, processes it and apply labelled weights to various objects in the image so that it can distinguish it from the other images.

The architecture of CNN is similar to that of human brain where each neuron responds to a stimulus only from a particular region of the visual field and a group of such fields overlap to form the whole visual area.

The main advantage of using CNN over other simpler Neural Networks is that it can process complex images having a lot of pixel dependencies with high accuracy. It can process temporal and spatial dependencies of the image efficiently by applying suitable filters.

Input will accept the image given into the computer by the user which can be any type of image like RGB, Greyscale, HSV or CMYK, *etc*. The three channels of a RGB image are shown in Fig. (**4.7**).

An input image can have any size from 4x4 to very high dimensions like that of 8K (7680 x 4320) for which the computation complexity is huge and CNN does a pretty good job in processing such huge images without losing its critical features and providing a good accuracy score.

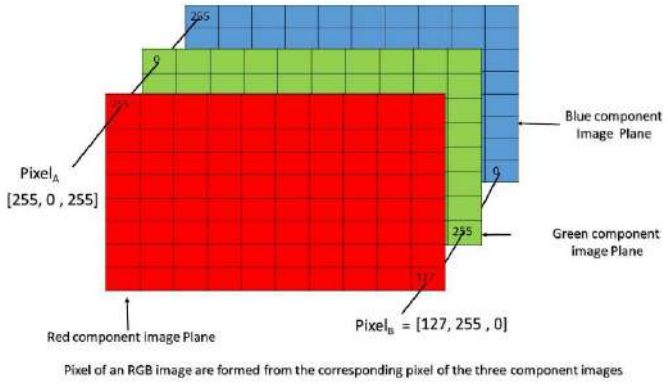

Fig. (4.7). 3D view of a RGB image

4.4.1. Convolution Layer

Convolution Layer is one of the most important layers in the Convolutional Neural Network and it does most of the heavy work. As the name suggest, this layer is responsible for the convolution operations on the raw pixel values of the image. It consists of a kernel or filter which is a matrix whose dimensions are less than that of the image. For Example - $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$ is a filter of size 3 x 3 x 1. The filter is the moved along the matrix in the direction of rows by a stride length which is the amount of length by which the filter will slide across the matrix. If it is 1, then the filter will move by one pixel at a time. When it is 2, we the filter will move by two pixels and so on [11].

An RGB image has 3 channels each having different set of pixel values. In order to apply convolution on this image we need to use filters for each channel and the filters are hovered simultaneously over each channel and finally it is added with the bias together to give the 2D-convoluted matrix of features as in Fig. (**4.8**).

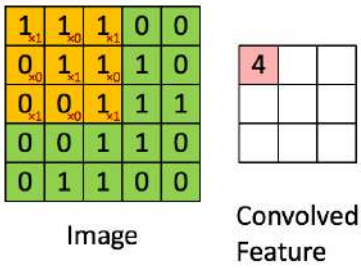

Fig. (4.8). Convolution Operation.

The main motive of this convolution operation is to extract the details like edges, colours, gradient, *etc*. from the input image. With an increased number of convolution operations, we can extract other features from the image and in this way the computer gets an overall understanding of the image [12].

Sometimes we have to add zeros to the boundary of the image which is called zero-padding or simply padding. The size of the padding is a hyperparameter. It is use to control the size of convoluted matrix. There are two types of padding –

- Same Padding – The dimension of the convoluted matrix is same as of that of the original image.
- Valid Padding – The dimension of the convoluted matrix is less than that of the original image.

The Rectified Linear Unit (ReLU) activation function is then applied on the entire convoluted matrix by mapping the negative values to 0 and keeping the positive values.

The main purpose of the of ReLU is to break linearity and to increase the non-linearity in our images as most our images consist of non-linear features like different colours or gradients, *etc*.

4.4.2. Pooling Layer

POOLING layer is responsible for reducing the dimension of the convoluted matrix in order to reduce the computational resources required to process the convoluted matrix. It is also used to extract important and dominant attributes from the image which is unaffected by the position and rotation of the image.

There are generally two kinds of pooling -

- **Max Pooling –** When the filter hovers over a region of the image, it will extract the maximum pixel value from that region of the image as in Fig. (**4.9**).
- **Average Pooling –** When the filter hovers over a region of the image, it finds out the average of all the pixel values present in the region.

Max Pooling is better than Average Pooling as it removes noise or extraneous values from the convoluted matrix apart from reducing spatial dimensions [13].

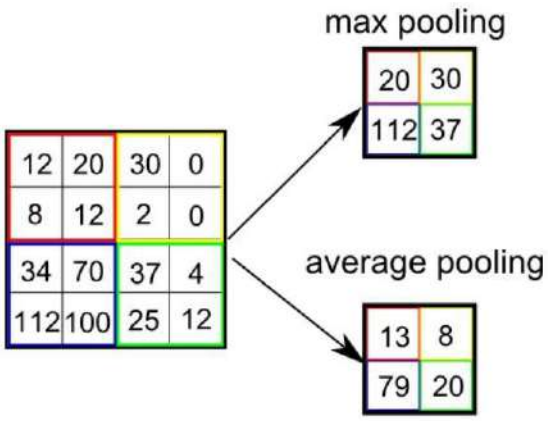

Fig. (4.9). Types of Pooling.

The number of layers of Convolution Layer, RELU and Pooling required depends on how complex the image is. However, a classic CNN model would be similar to the Equation (4a); with n ranging between 2 and 5 depending on the size and complexity of the image.

$$\text{Input} \rightarrow \text{Conv} \rightarrow \text{ReLU} \rightarrow n * (\text{Conv} \rightarrow \text{ReLU} \rightarrow \text{Pool}) \rightarrow \text{Full Connection} \quad \textbf{(4a)}$$

4.4.3. FULL CONNECTION

Fully connected layer is a neural network like in Fig. (**4.10**) where x is the input from the vector formed after flattening the convoluted matrix and y represents different probabilities of what the image might represent [12]. For example, if an image of car is given as input, it might show as 90% car, 7% train, 2% motorcycle, *etc*. The FC layer is one of the simple ways to make a machine learn about high level features of the image during the training of the dataset.

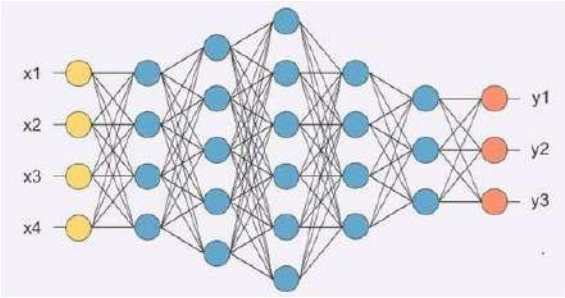

Fig. (4.10). Full Connection.

AlexNet, VGGNet, ResNet, GoogleLeNet are some of the many architectures of CNN developed over the years for the purposes image detection and classification.

4.5. AUDIO SIGNAL PROCESSING

As the title suggests, audio signal processing is the field of study from signal processing which deals with the manipulation of sound signal. Audio Signal Processing is not a new field of study. It has been used earlier in radio, phonography and telephonic communication. Nowadays, deep learning techniques are used for audio signal processing. In audio signal processing, the main focus is always on how to extract important frequency components from the audio sample which can be used to efficiently distinguish the audio signal.

Audio signals are typically in the range of 20 Hz to 20 kHz. These are longitudinal waves created by compression and expansion of air. When the sound enters our ear, it gets transmitted to the cochlea which is filled with thousands of hairs of variable size inside a fluid filled membrane. The short hair responds to higher frequency while the long hair responds to the low frequency of sound. Thus, the ear acts like a natural Fourier transform analyser [14].

The first step in audio processing is sampling which is defined as process of converting continuous signal into discrete values. The amount of information present in the discrete signal is defined by sampling frequency or rate. A higher sampling rate indicates less loss of information but the computational complexity of processing the discrete audio signal increases.

The amplitude of each sample of the discrete signal is also an important feature.

The processing of the entire audio signal at the same time can pose higher computational complexity apart from high time and resource consumption. The best way is to divide the audio signal into small frames of 20 – 40 ms. The power spectral density of each frame is calculated by applying Fourier transform on the autocorrelated audio signal. It shows the distribution of power among various frequency in the frame as shown in Fig. (**4.11**).

One important fact about the human ear is that the sounds with higher frequency is difficult to perceive than the sounds of lower frequency. This why the mel filter banks are used. The mel scale is defined as the scale of pitches judged by listeners to be equal in distance from one another. The reference point is taken such that 1000 mel = 1000 Hz. The mel filter bank acts like a human ear as the mel scale changes rapidly for less frequency than for higher frequency as shown in Fig. (**4.12**).

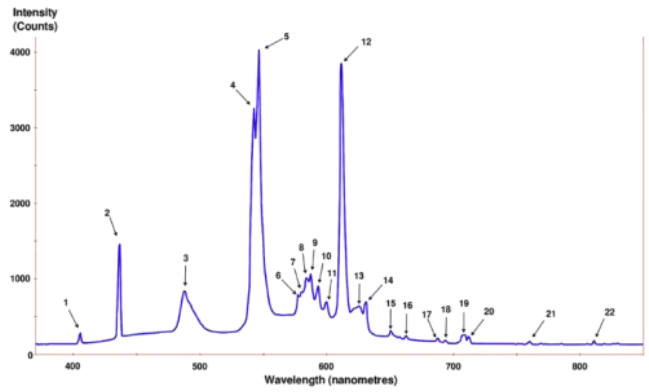

Fig. (4.11). Power Spectral Density.

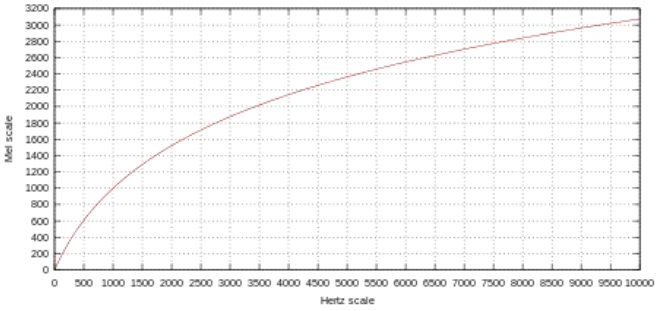

Fig. (4.12). Mel Scale.

The formula used to convert frequency to mel is given by Equation (5a):

$$m = 2595 * log_{10}(1 + \frac{f}{700}) \quad \textbf{(5a)}$$

The sum of energies in each mel filter is taken and passed through discrete cosine transform (DCT). The DCT extracts the most important information and peaks from the audio signal. The peaks represent the gist of the information [15]. This complete process creates a new spectrum known as Mel cepstrum as shown in Fig. (**4.13**).

Typically, the first 13 coefficient obtained from the Mel frequency spectrogram is known as MFCC. It contains the most useful information of the audio signal as it can be feed into machine and deep learning algorithms consisting of recurrent neural networks (RNNs) or other memory networks for audio and signal analysis. It is one of important and state of the art tool to extract information from the audio signal.

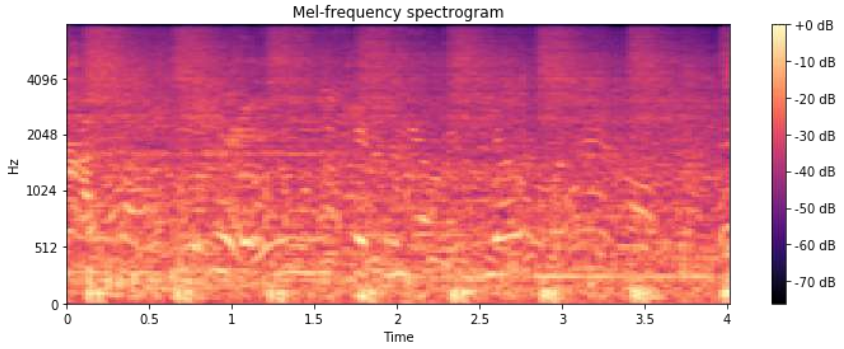

Fig. (4.13). Mel Cepstrum.

4.6. DEEP LEARNING IN DETECTION OF SLEEP APNEA

Sleep Apnea is a sleep disorder where breathing stops abruptly stops for at least ten seconds and starts repeatedly. Loud snoring and feeling of tiredness are two important symptoms of sleep apnea. In the beginning sleep apnea is detected by using full night polysomnography. This is called type 1 sleep study. It consumes a lot of time and requires a lot of space. Moreover, it involves a lot of wirings connected to various parts of the body which is very inconvenient for the patient. In type 4 sleep study which uses acoustic sensors and pulse oximeter makes it more portable and requires a smaller number of signals. In this study a deep convolutional neural network is first trained with training datasets in order to establish the crucial parameters and relationships among them [16]. The designing of a good convolutional neural network involves a good amount of trial and error. The performance of the neural network totally depends on the extracting the essential parameters through feature extraction. It is then used on test data to determine whether a particular patient has sleep apnea or not. The time required in type 4 sleep study during the training and implementation of deep learning model can be high due to complex nature of data and neural network but once the deep model is created, the testing requires little to no time.

4.6.1. System Design

The entire system architecture of the proposed model is shown in Fig. (**4.14**).

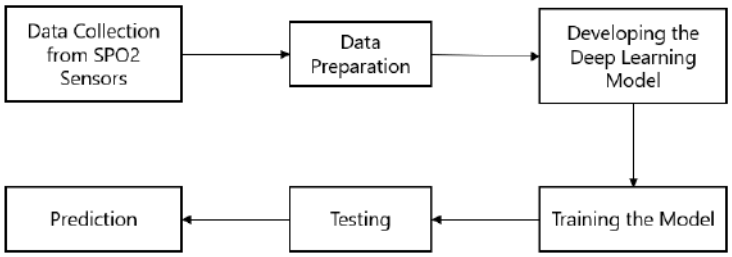

Fig. (4.14). Flowchart of the proposed Model.

The detection of sleep apnea from sleep studies generally consists of three basic steps. The first step is to collect data from the patient using SPO2 sensor to determine oxygen concentration in the blood, thermistor, nasal pressure and thoracic and abdomen movement. So, when apnea occurs, the sensors detect decreased nasal pressure, thoracis and abdomen movement as time-interval T_1 in Fig. (**4.15**) compared to normal breathing levels and the brain will send some signals to muscle to perform breathing activity and this is detected by EEG. During apnea, there is a sudden change in brain activity which is called arousal as the brain shifts from deep sleep to lighter sleep depicted by time-interval T_2 in Fig. (**4.15**). During the arousal event, suddenly the lungs cease to breath and thus the concentration of oxygen in the blood is disrupted causing oxygen desaturation as per the over lapping time-interval T_2 and T_3 in Fig. (**4.15**). which is detected by the SpO2 sensor [16, 17].

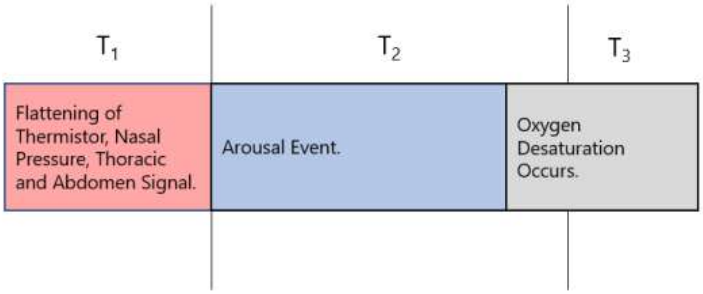

Fig. (4.15). Timeline of events for when Sleep Apnea occurs.

The raw data collected is feed to user defined software which uses deep learning techniques for analysis. This is the second step in system design. It determines whether sleep apnea is occurring or not and if it is occurring what is its duration and various other information.

In the third step, the doctors or health professionals determine the seriousness of this sleeping disorder based on the data produced by the software and accordingly the doctor will prescribe the required medication.

4.6.2. Detection of Apnea or Hypopnea Event

Sleep Apnea and Hypopnea are two different versions of sleep disorder. Apnea is caused due to complete obstruction of air while hypopnea is due to partial obstruction of air. So, in order to detect both of these two disorders, the variation of amplitude and duration of variation is to be calculated from the incoming signal from the sensors. The part of the signal caused due to apnea is flatter than the rest of the signal. The amplitude is smaller than the rest of the sound wave. As shown in Fig. (**4.16**), the ratio of amplitude of attenuated signal with amplitude A_2 at time interval T and the normal signal with amplitude A_1 is calculated. Generally, apnea is detected if the ratio of the normal signal and the attenuated signal at T is twenty percent less than normal and for hypopnea the maximum threshold is less than seventy percent. The time duration T for which this disruption of breathing is detected must be greater than or equal to ten seconds for apnea [18].

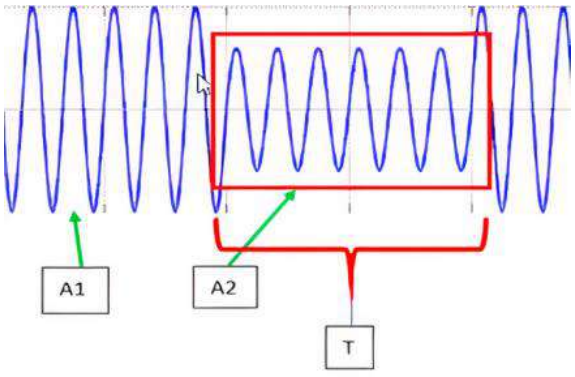

Fig. (4.16). Amplitude Threshold for Apnea Event.

4.6.3. Deep Learning Model

In this work, the authors have proposed a unique deep learning model to classify between healthy patients and the ones suffering from sleep apnea. The deep learning architecture using convolution neural networks is shown in Fig. (**4.17**). The input vector X is assumed to be the sampled SPO2 signal consisting of $\{x_1, x_2, x_3, ..., x_n\}$ samples where n = total number of samples. The layers between the input and the output layer are the hidden layers.

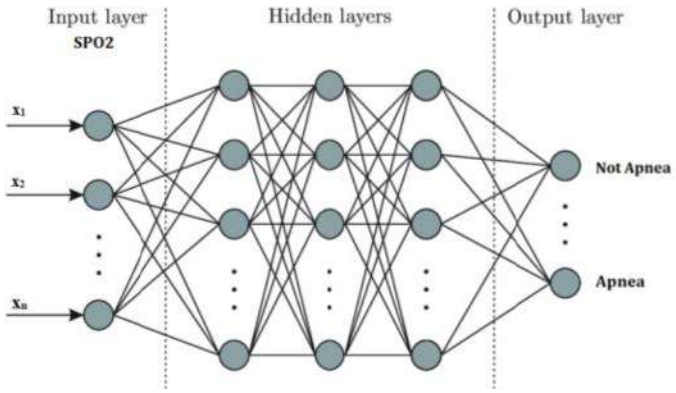

Fig. (4.17). Representation of the DL model.

The proposed deep learning model has ten layers. Three layers of 2-d convolution neural networks (CNNs) have been used in the model. Each 2-d CNN layer is followed by a 2-d MaxPooling Layer with a stride size of 2. The convolution layers are mainly used for the purpose of feature extraction while the max pooling layers down-samples the feature maps effectively removing the redundant data and bringing down the dimensionality. In this work, the authors have made use of the 'glorot uniform' kernel initializer class.

After the deep convolutional neural network is executed – a flatten layer is added to convert the 2-d data into a single 1-d vector for the input to the next layer. A hidden layer with sixty-four activating neurons is added followed by a dropout layer for regularisation. Regularisation reduces the generalisation error within the model but does not alter the training error. This is a binary classification problem as in Fig. (**4.17**). The neurons correspond to the detection of sleep apnea. The output neurons are activated by the sigmoid activation function. 'Adam' optimizer, 'binary crossentropy' loss function and ['accuracy'] metrics have been used to compile the model.

4.6.4. Evaluation of the Model

The proposed deep convolution neural network model was trained and the performance of the model was compared to existing literature. The model was trained using a test to train split ratio of 0.2 and the binary crossentropy loss function. The graphs of both loss and accuracy fluctuated initially for the first twenty epochs. The accuracy score started to stabilize soon and after 40 epochs the accuracy score hovered around 90 percent. So, a conclusion can be drawn from the observation that to achieve the stable condition for training the data 40 to

100 epochs are required. The accuracy score of the model after 100 epochs was 0.9047 or 90.47 percent outperforming the techniques used in other literature. The models built using linear discriminant analysis, SVM and artificial neural networks have accuracy scores of 0.865, 0.9 and 0.901 respectively when trained on the SPO2 signals. So, the proposed model works significantly better using SPO2 signals for sleep apnea classification and can be seen as an alternative to the cumbersome type 1 sleep study using polysomnography for apnea detection in the long run.

4.7. DEEP LEARNING IN CARDIAC ARRHYTHMIA DETECTION

Cardiac Arrhythmia is a condition in which the heart ceases to beat normally *i.e*., either it's in tachycardia or bradycardia resulting in abrupt rhythm. Arrhythmia is mainly caused by Atrial Fibrillation or changes in the heart tissues. Atrial Fibrillation is a medical condition rather than a disease, which causes the atrium in the heart to beat chaotically which eventually results in stroke or heart failure further resulting in permanent paralysis or death. In contrast, Ventricular Tachycardia causes fast, unpredictable pulse originating with anomalous electric signals received in the ventricles. The Electrocardiogram (ECG) is a common test used to diagnose Atrial Fibrillation. The traditional way of doing ECG where the doctor or the cardiologist takes the decision is a tedious and expensive way to monitor. Few conventional methods are already present which uses different classification methods of Machine Learning algorithms. The most commonly used ML algorithms include the likes of SVM, Decision Trees and Random Forests, but those lacks accuracy and precisions. The traditional ways of diagnosis are being replaced by using systems based on Deep Learning architecture. Such systems take continuous input of ECG signal from a health band which eventually notifies about the health condition. Such Deep Learning architecture are made of Convolutional Neural Networks and LSTMs for the extraction essential parameters required to classify between Arrhythmic conditions and normal [19]. Fig. (**4.18**) shows ECG signal for different heart conditions.

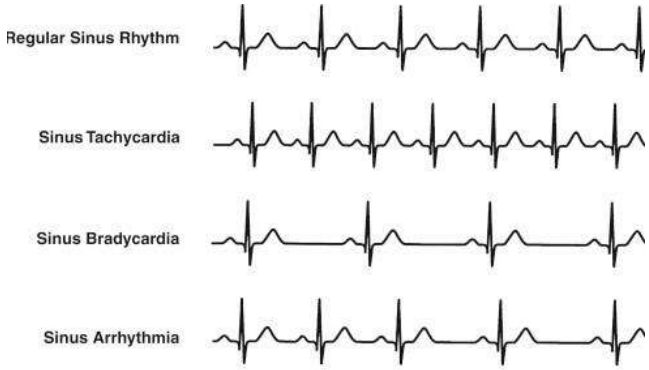

Fig. (4.18). ECG signal for different heart conditions

4.7.1. System Design

The entire system architecture of the proposed model is shown in Fig. (**4.19**).

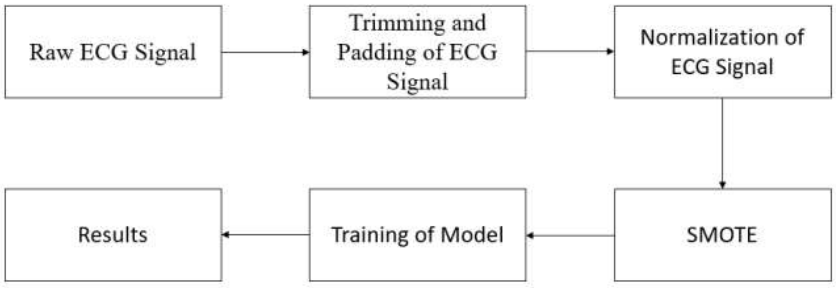

Fig. (4.19). Block Diagram of proposed model.

Deep learning model accuracy is high dependent on the inputs, the user feeds into it for learning. Less samples, noisy signals and inconsistency on distribution of sample in each class may led to wrong interpretation of inputs. Here, as we are dealing with ECG signals, which are continuous in nature are unsuitable as the model urges for discrete normalized numeric values for better learning. The dataset of ECG, which is being used for training are of varied length ranging between 9s to more than 60s and continuous. Therefore, each signal is being sampled at a rate of 300 samples/sec and trimmed till 7500 samples for uniformity. Signals having lesser durations are being right padded with zero. The sampled value of ECG produces a wide range of values, which might mislead the model to learn absolute values of one instance instead of the value for all instances. To prevent such unwanted situation data normalization is being done by

adjusting to a common scale. For better accuracy, the mean value of the data is being subtracted from the original value which is eventually divided by its maximum absolute clue. Thus, the normalized data lies within the scale of [-1, 1]. As mentioned before that the dataset is highly imbalanced, hence to balance the authors have used SMOTE algorithm which over samples the minority class by filling it with artificial data points, restricted till training dataset [20].

Each ECG data consists of important features which are used to derive information about the human heart. Therefore, the processed data are feed in convolutional neural network (CNN) for extraction of essential features that is of high importance for classification [21].

4.7.2. Deep Learning Model

In this work, the authors have proposed a unique deep learning model. The DL model has twelve-layers. Three layers of 1D convolution have been used. These 1D convolutional layers are used for feature extraction from the already pre-processed and sampled ECG signals. For each convolution layer a single 1D MaxPooling layer with a stride size of 2 is used. MaxPooling layers down sample the feature maps from the CNN operation by selecting the maximum value within its stride, thus effectively reducing redundant data in the feature maps and maximizing required information. The CNN layers have kernel sizes of 6, 5, 3 respectively. Dropout layers with values 0.05, 0.10, 0.15 are added for regularization. Regularisation warrants that the model does not overfit training-data.

Next the extracted features are fed into two LSTM layers with 64 output neurons each. LSTM layers vary from traditional/conventional RNN (Recurrent neural network) layers due to its distinctive structure and ability of learning long term dependencies. LSTMs work much better with series prediction data - if there is enough training data which is true for this case. The main advantage of LSTMs over traditional neural network is that LSTM layers have loops in them which acts as memory units. These self-loops allow information to persist. A LSTM layer also has a chain like structure like the RNNs. Instead of having a single NN layer like in RNN they have four NN layers, interacting with the data in a special way. Finally, the output layer has four neurons for the respective output classes. Each neuron is activated by the SoftMax activation function. Each neuron gives a probability value of the signal belonging to the class. The class corresponding to which the classifier gives the maximum probability is taken as the class label for the given signal. ‘Adam’ optimizer, ‘categorical crossentropy’ loss function and [‘accuracy’] metrics have been used to compile the model.

4.7.3. Evaluation of the Model

The neural network model has been implemented using Keras functional API supported by TensorFlow backend. A split ratio of 0.1 is used to divide the training and the test sets *i.e.*, 10% of the dataset is used for testing and prediction while the remaining 90% of the dataset is used for training the model. The model is trained for 75 epochs at the end of which the model gives a training accuracy of 94.95%, a training loss of 0.1215 and validation accuracy of 83.68%, validation loss of 0.5465. The training *vs.* validation accuracy, training *vs.* validation loss in each epoch for the proposed model has been plotted in Fig. (**4.20a**) and Fig. (**4.20b**) for visualization.

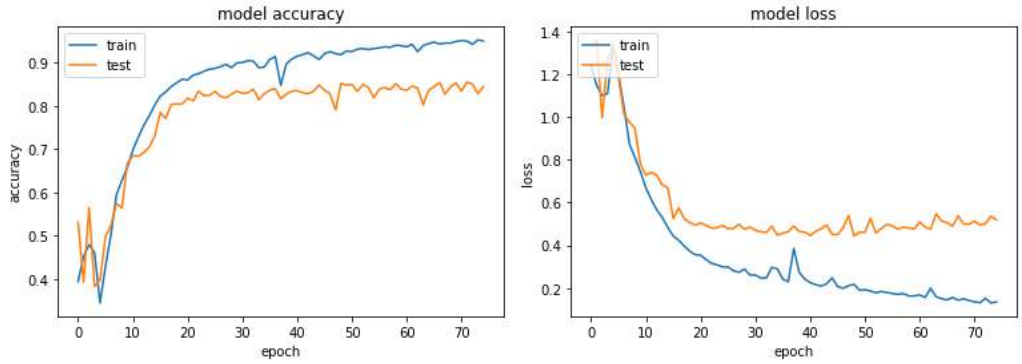

Fig. (4.20). (a) Training *vs.* Validation Accuracy, (b) Training *vs.* Validation Loss.

4.8. DEEP LEARNING IN DETECTION OF BRAIN TUMOURS

Brain is one of the most important organs of our body. Brain cells except nerve cells get replaced when they become old or get damaged. The production of excess of these new cells results in the formation of tumours. Tumours are generally categorized into two groups: benign *i.e.*, non-cancerous and malignant *i.e.*, cancerous.

The detection of brain tumour is hard due to its size, shape and location. Timely detection and treatment of tumour increases the chances of survival of the patient. The detection of tumour is generally done using advanced medical equipment like Magnetic Resonance Imaging (MRI) or Computerized Tomography (CT) scan. MRI imaging is mostly used in the diagnosing brain tumours as it provides better image of brain because of better high contrast image of tissue in human brain as shown in Fig. (**4.21**) [22]. It is the most important method of detecting and diagnosing brain tumour due to higher accuracy.

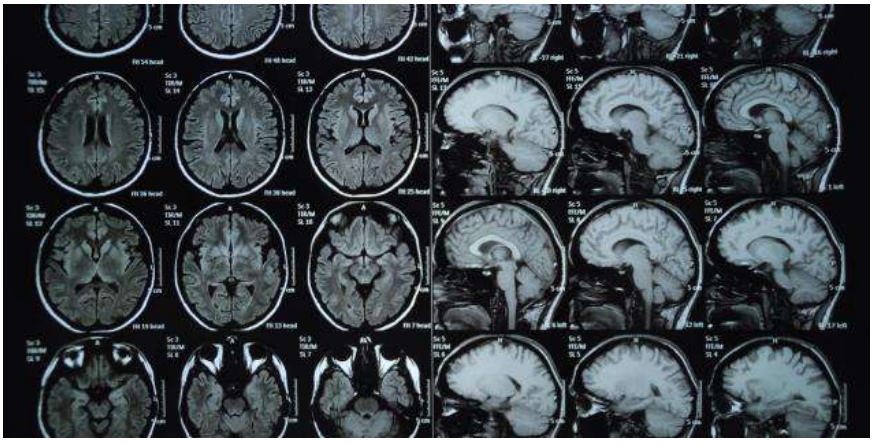

Fig. (4.21). MRI images of the human brain

The detection of brain tumour from MRI images requires years of expertise and hard work. The use of deep learning neural network models is not to replace the doctors and health professionals but to assist them thereby making their work easier and faster. They will be able to administer the required treatment to the patient at the earliest [22, 23].

4.8.1. System Design

The accuracy of the deep learning model totally depends on the architecture of the model and how the model is trained with the datasets. The dataset images obtained from MRI should be of high quality as low-quality images may contain a lot of discrepancies which may lead to wrong interpretation of the image. The images should be pre-processed by removing parts not part of the brain from the image, correcting non-uniform properties, converting coloured image to grey scale for better processing, improving brightness or contrast to provide better clarity, *etc*. Segmentation of the images involve dividing the image into multiple segments for better processing of various parts of the brain and setting up a contour around the region. A large number of images from various patients should be taken in consideration. Different sections of the brain must be properly scanned in those images. Images where the tumour is of benign as well as malignant nature should be included along with the images of normal and healthy brain for proper interpretation by the deep learning model [24].

Each and every image in the dataset consists of important features which are used to derive important information about the brain. In order to obtain a high accuracy, the deep learning model should extract the essential features that is of utmost importance. The selected information about different parts of the brain,

kind of tumour present, *etc*. are then feed into convolutional neural network (CNN) for training [22]. The flowchart of the proposed work is given in Fig. (**4.22**).

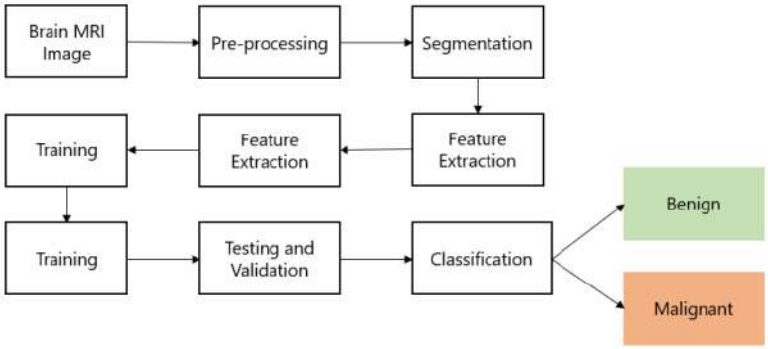

Fig. (4.22). Flowchart of the proposed work.

In order obtain the most out of given dataset clustering algorithm is used to perform feature extraction. In this algorithm, images are divided into specific groups based on similarity of data and are labelled. This algorithm is very adaptable to changes in data and can easily pick out essential features that used to distinguish each group. It is scalable and can easily be interpreted even by an amateur person.

4.8.2. Deep Learning Model

In this work, the authors have proposed a distinctive deep learning model to classify between patients whose brain tumour is benign or malignant. This is typically an image classification problem with two output classes and hence a convolution neural network model has to be built. However, the architecture proposed for this work is quite different from the traditional neural network architecture defined by Equation (4a).

Initially a zero padding with two rows and two columns are added to increase the dimension of the input image. In this work, only one 2d convolution layer is used. Filters of dimension 7*7 are used for the convolution operation with a stride size of 1 and output with 32 kernels. The pixels after the convolution operation are normalized and passed through a ReLU activation function. Two consecutive max pooling layers follow next each with a stride size of 4 units. After the deep convolutional neural network is executed – a flatten layer is added to convert the 2-d data into a single 1-d vector for the input to the dense layer. The neural

network model does not include a hidden layer. As stated earlier this is a binary classification problem. The output layer is activated using the sigmoid activation function. The corresponding output activation signifies whether the brain tumour is of malignant nature or benign. 'Adam' optimizer, 'binary crossentropy' loss function and ['accuracy'] metrics have been used to compile the model.

4.8.3. Training the Model

The neural network model was implemented using Keras functional API supported by TensorFlow backend. A split ratio of 0.2 is used to divide the training and the test sets *i.e.*, 20% of the dataset is used for testing and prediction while the remaining 80% of the dataset is used for training the model. The model is trained for 25 epochs at the end of which the model gives a training accuracy of 97.37%, a training loss of 0.1010 and validation accuracy of 89.03%, validation loss of 0.2915. The training *vs.* validation accuracy, training *vs.* validation loss in each epoch for the proposed model has been plotted in Fig (**4.23a**) and Fig (**4.23b**) for visualization.

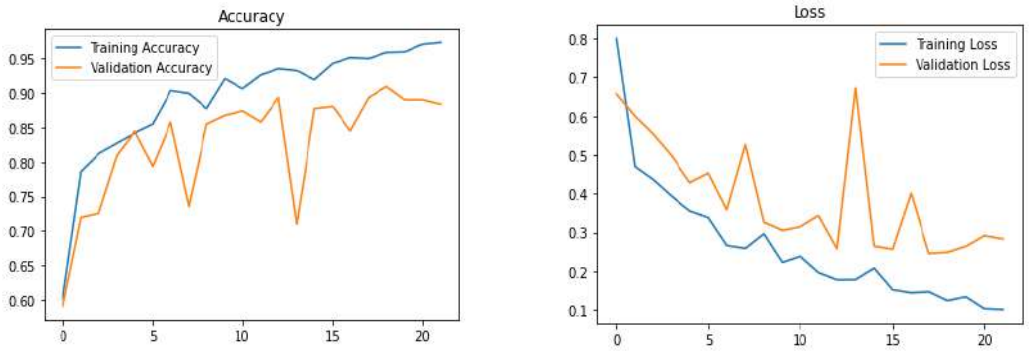

Fig. (4.23). (a) Training *vs* Validation Accuracy; (b) Training *vs.* Validation Loss.

4.8.4. Result

Using 80% of the dataset for training and 20% of the same for testing the model gives a validation/test accuracy of 89.03%. The training accuracy at the end of 25 epochs is nearly 97.37%. The snips of the result along with the actual and predicted class labels are shown in Fig. (**4.24**) and Fig. (**4.25**) respectively with [0] corresponding to the benign class and [1] corresponding to the malignant class.

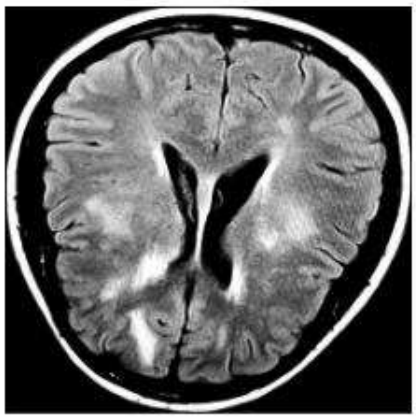

Fig. (4.24). Actual Class - [0] Predicted Class - [0].

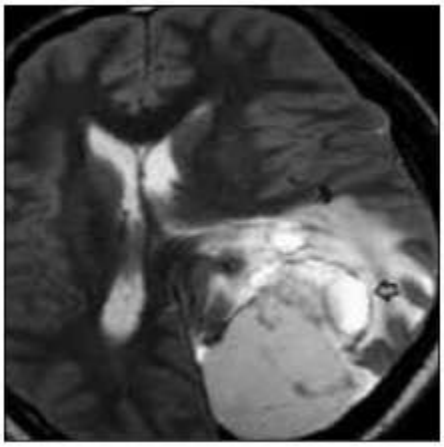

Fig. (4.25). Actual Class - [1] Predicted Class - [1].

DISCUSSION AND FUTURE WORKS

Deep Learning is a state-of-the-art tool in the field of Artificial Intelligence. The use of Convolutional Neural Network (CNN) or Recurrent Neural Network (RNN) as well as other deep learning tools in the field of medical science has profound effect. It assists the health professionals in delivering speedy and effective treatment. The advent of vaccines and medicines in the 20th century has revolutionize the medical field. Similarly, these deep learning techniques along with machine learning and artificial intelligence can be a game changing for the medical science in this 21st century.

The development of new drugs requires a substantial amount of time expanding to more than 5 to 10 years. The drug discovery is primarily targeted towards a

particular group of molecules in the body. But it is very hard to predict how this drug will affect non targeted molecules and whether it will have any kind of adverse side effect. With the current technology it is really hard to predict such result. Deep learning can be used to detect toxic compounds and the side effects of these compounds and thus it can reduce the chances of failure.

Prediction of disease and its prevention is better than finding a cure for the disease. The chances of developing a disease involves an intricate calculation and has various types of outcomes based on the vastness of data available. Deep learning along with big data technologies can be used to predict such disease by processing the data from the electronic heath records and thus can be used to prevent such disease. Epidemic and pandemic situations can be averted by creating similar deep learning simulation.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] Michael G. Pecht, and Myeongsu Kang, "Machine Learning: Fundamentals, in Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things", *IEEE,* pp. 85-109, 2019.
[http://dx.doi.org/10.1002/9781119515326.ch4]

[2] M. James, Derong Liu Keller, and David B. Fogel, "Multilayer Neural Networks and Backpropagation, in Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation", *IEEE,* pp. 35-60, 2016.

[3] M. Madan, *Gupta; Liang Jin; Noriyasu Homma, "Multilayered Feedforward Neural Networks (MFNNs) and Backpropagation Learning Algorithms," in Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory.* IEEE, 2003, pp. 103-170.
[http://dx.doi.org/10.1002/0471427950.ch4]

[4] Xiaogang Wang, "Deep Learning in Object Recognition, Detection, and Segmentation , now, 2016",
[http://dx.doi.org/10.1561/2000000071]

[5] Li Deng, and Dong Yu, Deep Learning: Methods and Applications. 2014.
[http://dx.doi.org/10.1561/2000000039]

[6] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networkS", *International Journal of Engineering Applied Sciences and Technology,* vol. 04, no. 12, pp. 310-316, 2020.
[http://dx.doi.org/10.33564/IJEAST.2020.v04i12.054]

[7] M. Rezaeian Zadeh, S. Amin, D. Khalili, and V.P. Singh, "Daily outflow prediction by multi layer perceptron with logistic sigmoid and tangent sigmoid activation functions", *Water Resour. Manage.,* vol. 24, no. 11, pp. 2673-2688, 2010.
[http://dx.doi.org/10.1007/s11269-009-9573-4]

[8] Schmidt-Hieber, ", Johannes. Nonparametric regression using deep neural networks with ReLU activation function. Ann. Statist. 48 (2020), no. 4, 1875--1897. ",
[http://dx.doi.org/10.1214/19-AOS1875]

[9] F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu, "A Sufficient condition for convergences of adam and RMSProp", *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 11119-11127, 2019.Long Beach, CA, USA.
[http://dx.doi.org/10.1109/CVPR.2019.01138]

[10] L. Li, M. Doroslovački, and M.H. Loew, "Approximating the Gradient of Cross-Entropy Loss Function", *IEEE Access,* vol. 8, pp. 111626-111635, 2020.
[http://dx.doi.org/10.1109/ACCESS.2020.3001531]

[11] C-C. Jay Kuo, "Jay Kuo, Understanding convolutional neural networks with a mathematical model, Journal of Visual Communication and Image Representation, ",

[12] M.D. Zeiler, and R. Fergus, Visualizing and Understanding Convolutional Networks.*Computer Vision – ECCV 2014. ECCV 2014.,* D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars, Eds., vol. 8689. Springer: Cham, 2014.Lecture Notes in Computer Science
[http://dx.doi.org/10.1007/978-3-319-10590-1_53]

[13] D. Scherer, A. Müller, and S. Behnke, Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition.*Artificial Neural Networks – ICANN 2010. ICANN 2010.,* K. Diamantaras, W. Duch, L.S. Iliadis, Eds., vol. 6354. Springer: Berlin, Heidelberg, 2010.Lecture Notes in Computer Science
[http://dx.doi.org/10.1007/978-3-642-15825-4_10]

[14] R.F. Lyon, "Machine hearing: an emerging field", *IEEE Signal Process. Mag.,* vol. 27, no. 5, pp. 131-139, 2010.
[http://dx.doi.org/10.1109/MSP.2010.937498]

[15] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound event detection in multisource environments using source separation", *Workshop on machine listening in Multisource Environments,* 2011

[16] Hnin Chaw, Sinchai Kamolphiwong, and Krongthong Wongsritrang, "Sleep apnea detection using deep learning", *Tehnički glasnik. , .*
[http://dx.doi.org/10.31803/tg-20191104191722]

[17] S.S. Mostafa, F. Mendonŋa, F. Morgado-Dias, and A. Ravelo-García, "SpO2 based sleep apnea detection using deep learning", *2017 IEEE 21st International Conference on Intelligent Engineering Systems (INES), Larnaca,* 2017pp. 000091-000096
[http://dx.doi.org/10.1109/INES.2017.8118534]

[18] D. Dey, S. Chaudhuri, and S. Munshi, "Obstructive sleep apnoea detection using convolutional neural network based deep learning framework", *Biomed. Eng. Lett.,* vol. 8, no. 1, pp. 95-100, 2017.
[http://dx.doi.org/10.1007/s13534-017-0055-y] [PMID: 30603194]

[19] S. Ross-Howe, and H.R. Tizhoosh, "Atrial Fibrillation Detection Using Deep Features and Convolutional Networks", *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI),* 2019pp. 1-4 Chicago, IL, USA
[http://dx.doi.org/10.1109/BHI.2019.8834583]

[20] N.V. Chawla, L.O. Hall, K.W. Bowyer, and W.P. Kegelmeyer, "Smote: Synthetic minority oversampling technique", *J. Artif. Intell. Res.,* vol. 16, pp. 321-357, 2002.
[http://dx.doi.org/10.1613/jair.953]

[21] M. Limam, and F. Precioso, *"Atrial fibrillation detection and ECG classification based on convolutional recurrent neural network," 2017 Computing in Cardiology.* CinC: Rennes, 2017, pp. 1-4.
[http://dx.doi.org/10.22489/CinC.2017.171-325]

[22] S. Sajid, S. Hussain, and A. Sarwar, "Brain Tumor Detection and Segmentation in MR Images Using Deep Learning", *Arab. J. Sci. Eng.,* vol. 44, no. 11, pp. 9249-9261, 2019.
[http://dx.doi.org/10.1007/s13369-019-03967-8]

[23] H. Mohsen, E-S.A. El-Dahshan, E-S.M. El-Horbaty, and A-B.M. Salem, "Classification using deep learning neural networks for brain tumors, Future Computing and Informatics Journal", 3

[24] A. Işın, C. Direkoğlu, and M. Şah, Review of MRI-based Brain Tumor Image Segmentation Using Deep Learning Methods, Procedia Computer Science, Vol. 102, 2016, pp. 317-324, ISSN 1877-0509.

CHAPTER 5

Bankruptcy Prediction Model Using an Enhanced Boosting Classifier based on Sequential Backward Selector Technique

Makram Soui[1,*], Nada Namani Zitouni[2], Salima Smiti[3], Kailash Kumar[1] and **Ahmad Aljabr[1]**

[1] *College of Computing and Informatics, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia*

[2] *University of Gabes, Zrig Eddakhlania, Tunisia*

[3] *University of Manouba, Manouba, Tunisia*

Abstract: Corporate bankruptcy prediction is one of the most crucial issues that impact the economic field, both on the local and global scale. The primary purpose of bankruptcy prediction is to investigate the economic state of any corporation and evaluate its distress level. Several machine learning and deep learning models have been used to predict financial failure. However, there is still no technique that resolves all the problems faced in this field. As such, we propose a machine learning model that constitutes a feature selection phase and a classification phase to predict corporate bankruptcy. This technique combines the sequential backward selector (SBS) with AdaBoost and JRip algorithms. The first phase uses SBS to select the best subset of features for the training. The second phase trains the AdaBoost with the JRip classifier to predict each target class. This model is evaluated using the highly imbalanced Polish bankruptcy dataset. The comparative analysis of our model with other techniques proves the efficiency in predicting corporate bankruptcy with an average of 91% of the AUC metric.

Keywords: Bankruptcy prediction, Boosting technique, Classification, Feature selection, Polish bankruptcy dataset, Python, Rule-based classification, Two-stage method, Weka, Wrapper methods.

5.1. INTRODUCTION

The dilemma of predicting corporate bankruptcy aims to differentiate firms with a likelihood of distress from healthy corporations. It is a phenomenon that may be

[*] **Corresponding author Makram Soui:** College of Computing and Informatics, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia; Tel: +966550310382; E-mail: m.soui@seu.edu.sa

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

caused by unsuitable decision-making. It impacts the economic field on the local and global scale since it renders all institutions involved vulnerable. Thus, emerged the need for tools that allow monetary institutions to predict bankruptcy. This need heightened the interest of researchers and scientists in the field of corporate bankruptcy [1 - 5].

Many studies have proffered several solutions for financial failure prediction. These articles focused on the structure of the data, the features, and the target classes of samples to find a relationship between the characteristics and the target classes to perform the prediction [6]. These existing works are split into statistical methods and artificial intelligence techniques. The statistical models consist of using mathematical and statistical principals to formulate a relationship between variables in the data, such as Multivariate Discriminant Analysis (MDA) [7], Linear Discriminant Analysis (LDA) [8], and Quadratic Discriminant Analysis (QDA) [9]. But these methods fail to obtain valid hypotheses for individualistic features due to their presumptuousness for the multivariate normality, linear separability, and independence of the predictive variables [10]. The artificial intelligence methods aim to build models based on machine learning and deep learning algorithms like Decision Trees [11], Support Vector Machines [12], and Artificial Neural Networks [13]. These algorithms can build cognitive functions/pattern recognition based on high dimensional datasets. Subsequently, they do not make assumptions when dealing with data distribution [14]. Unfortunately, most of these algorithms are black-box models. They fail to accommodate the data imbalance problem, which is a common factor in bankruptcy datasets.

Recently, the use of ensemble methods demonstrated their ability to give relatively precise predictions for bankruptcy, specifically AdaBoost [15], Gradient Boosting [16], and eXtreme Gradient Boosting (XGBoost) [17]. The data imbalance phenomenon is a challenge when dealing with classification since the samples of one class outnumber those of the others, which creates fluctuations in the training sets. Since the ensemble methods use a set of classifiers instead of a single classifier that is capable of dealing with the imbalanced data [18], these methods are sensitive to noisy data and outliers.

In this context, we present a bankruptcy prediction model based on Adaptive Boosting (AdaBoost) algorithm. This algorithm is very apt at dealing with the data imbalance issue. It is also compatible with almost any learning algorithm, which allows for an expanse of experimentation. In our approach, we combined AdaBoost with the JRip algorithm. This novel approach was trained on the extracted data from the Sequential Backward Selection (SBS) technique.

The remainder of this paper follows this structure: Section 2 is a discussion of the related work concerning the bankruptcy prediction problem. Section 3 explains the proposed approach. Section 4 analyses the proposed method and conducts a comparison with other studied algorithms. Finally, section 5 provides a set of perspectives for future work.

5.2. RELATED WORK

Several algorithms have been proposed to predict company bankruptcy. The research splits into statistical and artificial intelligence techniques.

5.2.1. Statistical Techniques

Statistical methods use mathematical and statistical principles to formulate a relationship between variables and their significance in the data. The most popular statistical methods include LDA, MDA, QDA, LRA, and FA.

These techniques were the starter point in predicting corporate bankruptcy. Single Discriminant Analysis [19] and Multivariate Discriminant Analysis [20] were the earliest models used in this field. Although many criticized their inability to adjust to the variables and support their normality [21], statistical methods provided an interesting concept, which motivated researchers to invest more in them. Consequently, this leads to the introduction of the logit and probit model [22]. Still, this technique was also found to be flawed because of the high risk of the predictions falling outside the appropriate range of the classification. The focus then shifted to put more value in the data characteristics. Combining the logit and probit model and factor analysis model procured a solution that gave better results. The solution was eventually considered the most favourable one at the time for its high performance [23].

Statistical techniques are well versed when extracting relationships between variables. However, they are unable to adapt to high dimensional financial datasets. Nowadays, they are deemed ineffective in resolving the issue of corporate bankruptcy prediction.

5.2.2. Artificial Intelligent Techniques

Statistical methods need to work with structured data to predict bankruptcy [10]. Intelligent techniques, however, are successful in solving the problem, thanks to their ability to mould to the data and spontaneously extract knowledge to procure accurate models [24].

There are two types of techniques; machine learning and deep learning.

5.2.2.1. Machine Learning Techniques

Machine learning techniques are capable of finding natural patterns in data. They create assumptions for the training data to develop insight, and help make better decisions and predictions. Several industries and disciplines, such as medical diagnosis, image processing, learning association, and bankruptcy prediction, have used these methods. Machine learning algorithms such as Support Vector Machine (SVM) [25], Decision Tree (DT) [26], and Artificial Neural Network (ANN) [27] have demonstrated prodigious results for bankruptcy prediction.

The Decision Tree algorithm (DT) is well versed in predicting bankruptcy. It is capable of giving coherent classification rules and its prowess when dealing with different data types. The use of this technique to predict bankruptcy, validated on a well-balanced Japanese dataset, outperformed discriminate analysis in terms of accuracy [28]. Additionally, a study conducted on Serbian datasets compared different decision tree models with the discriminant analysis model. The results showed the supremacy of the DT models [29]. However, the use of this method is sometimes frowned upon when addressing multi-class problems because it is inclined to errors. Additionally, DTs tend to overfit when dealing with unbalanced data.

Alternative techniques for bankruptcy prediction are Artificial Neural Networks (ANNs). These methods have become universal given their ability to adapt to the data set and learn from it. They are capable of catching non-linear relationships between variables. Moreover, they do not need prior knowledge of the functional form of the data. The performance of ANN validated on the Belgian dataset achieved performant results when applied with a small number of variables [30]. A recent study examined prominent classification techniques (Decision Tree, Logistic Regression, and ANN). The performance of the classifiers showed that ANN was the best alternative when data availability was high in the dataset [31]. Additionally, a recent study used a neural network with a sigmoidal activation function to predict bankruptcy. The model was compared to a Random Forest model, using the error rate. The results showed that ANN was a more reliable technique for financial failure prediction [13]. However, it is a black-box method, so it is puzzling to explain the outcome compared to other techniques like linear regression. Moreover, while fitting a neural network model, it is best to take extra care of the attributes and data normalization to improve the performance.

Another well-received intelligent technique is SVM. This method's ability to incorporate the structural and empirical threats for minimization principles is what

sets it apart from other methods. SVMs were used as a feature selection technique to identify a set of optimal features to assess the Altman's ratio relevance for bankruptcy prediction nowadays. The results show that the chosen variables significantly improve the accuracy of financial failure prediction. The study shows that they outperformed traditional methods [32]. A recent study introduced a novel technique called Diverse Density-Support Vector Machine (DD-SVM) for bankruptcy prediction. The results inferred that this method outperformed the Logit, MDA and ANN models in the accuracy, error type I and type II metrics. Additionally, this model is presumed to be an adequate warning system for corporate distress [12]. The inhibitions of SVM lay in the fact that first, the kernel function needs to be deftly hand-tuned. Second, much like ANN, SVM has a black-box nature, which is why it is impracticable to obtain a graspable model. Finally, SVM is a computationally expensive algorithm.

The search for a wholesome technique to predict bankruptcy included the use of ensemble methods. Recent research demonstrated that these techniques outperform single classifier methods [33, 34]. A recent study analyzed three AdaBoost models combined with imputation methods for bankruptcy prediction. It was validated on two real-world datasets (USABDS and JPNBDS) with a high ratio of missing data. The results show the robustness of the model for financial distress prediction. The method also outperformed neural networks, logistic regression, and decision tree models in the accuracy metric [15]. eXtreme Gradient Boosting (XGB) is a technique that uses an ensemble of decision trees as a means of learning. The introduction of this method for financial failure prediction demonstrated that boosting is an efficient approach when predicting bankruptcy. The model introduced a new system called synthetic features. It uses econometric measures and arithmetical operations to improve performance. It has been tested on five datasets of Polish companies [35]. Still, just like other techniques, ensemble methods are vulnerable to several factors, such as noise. Thankfully, the introduction of an enhanced boosting process called FS-Boosting based on feature selection helped remediate this vulnerability. The model combined Information Gain (IG) feature selection and the standard boosting procedure to predict bankruptcy [36]. It is validated on two small and balanced real-world bankruptcy datasets.

5.2.2.2. Deep Learning Techniques

Deep learning (DL), otherwise known as deep neural learning, is a sub-component of machine learning. It uses neural networks to learn from unstructured and unlabeled data. In the last decade, the use of deep learning has shown tremendous results for various problems, such as natural language understanding [37],

sentiment analysis [38], language translation [39] and image classification [40]. However, these techniques mostly adapt to analog data, like images, text documents, and audio files. They are not adept at dealing with data in tabular format, which is why their use in predicting bankruptcy is rare. Still, some research does exist. In the case of Deep Belief Network (DBN), Yeh and Wang combined DBN with the Restricted Boltzmann Machine (RBM) to propose a bankruptcy prediction approach verified on the American dataset. This model proved to be very successful. It outperformed support vector machines [41]. In this context, a comparative study of deep belief networks (DBN) feed-forward neural network (FNN) and support vector regression (SVR) used the root mean squared error (RMSE) as a metric. It demonstrated that DBN outperformed the other approaches proving its effectiveness in predicting bankruptcy [42].

In a recent study, Soui, Smiti, Mkaouer, and Ejbali applied Stacked Auto-Encoders with the softmax classifier to perform bankruptcy prediction. This approach utilized a two-layer auto-encoder to ascertain the attributes. Then, it used a softmax classifier layer was to provide the probability of the class labels. Finally, the hidden layers were fine-tuned using back-propagation to heighten the performance of the model. This method proved to be very efficient. It outperformed Random Forest (RF), Support Vector Machine (SVM), and XGBoost (XGB) [43].

Deep learning demonstrates impressive results in financial failure prediction. However, the limitations in DL lie in the fact that first, it is a type of learning that is very abstract, which does not allow direct input to output representation [44]. Second, it is very dependent on relatively balanced data [45]. This criterion is not always possible when dealing with large-scale data, especially in the economic field.

In this work, we conduct a comparative study of a Boosting technique using different single classifiers and feature selection algorithms to determine how they impact corporate bankruptcy prediction.

5.3. BACKGROUND

In this part, we introduce the algorithms used in our work. These techniques are divided into feature selection methods, rule-based classifiers and ensemble learning techniques.

5.3.1. Feature Selection (FS) Algorithms

Feature selection reduces a feature set's dimensionality. It helps determine the variables with the most redundancy, irrelevancy, or noise and eliminates them. It generally helps obtain better data quality as well as speed up and improve an algorithm's performance [46]. There are two types of FS techniques, wrapper methods and filter methods. The feature selection process for wrapper methods requires a predetermined learning algorithm. This algorithm selects the features that ensure the highest performance [47]. The filter methods evaluate features individually and compute some relevant measures for every variable. They use a scoring mechanism to rank the variables to achieve the best classification accuracy with a minimal variety of features [48]. They are usually less accurate than wrapper methods [49].

The computational cost of wrapper methods is high due to the embedding of the classifier in the selection process. However, the ability to choose a learning classifier as a base for selecting the features is very intriguing. In this section, we present the FS techniques used in this study.

5.3.1.1. Sequential Feature Selection (SFS)

The Sequential Feature Selection (SFS) algorithms [20] are wrapper methods. They are specified as the baseline approach for feature selection, as they are widely recognized and broadly utilized in practice. They are search algorithms of greedy nature. They are used to minimize a dimensional feature space to a smaller feature subspace. The automatic selection of a subset of features is a must-have. It has high relevance to the problem of dimensionality minimization. This concept is the motivation behind feature selection algorithms. Based on the classifier performance, they either dispose of or add one variable at a time until the feature subset reaches the desired size.

Sequential Forward Selection or SFS belongs to the Sequential feature selection family. It takes as input a labelled dataset, a learning algorithm, the number of characteristics to be selected, and outputs the generated subset of variables. It initializes with an empty set (“null set”) of a feature subset, the feature set in the dataset, and a criterion function. It starts a loop that appends a random variable to the feature subset. This feature, the classifier, and the feature set then create a hypothesis h(x). It then starts another loop and iterates through the remaining initial variables, chooses one, and updates h(x). If the hypothesis maximizes the criterion function, then the variable is added to the feature subset.

It will continue adding features from the feature set and training them with the determined hypothesis until the feature subset contains the number of desired features specified at the start.

5.3.1.2. Particle Swarm Optimization (PSO)

The PSO method simulates the social behaviours of animals such as birds flocking and fish schooling. Initially, PSO was suggested as an optimization approach for continuous problems. However, the need to optimize many concepts such as feature selection pushed the creators of the PSO algorithm to broaden its usage. To solve the discrete problems, they introduced binary particle swarm optimization (BPSO). The technique encodes the position of every particle via a binary string [50].

The technique works as follows. A swarm refers to a population where viable solutions encode as particles in the search space. This technique starts by randomly initializing a population of particles. Particles roam the search space to find the ultimate solution. The algorithm updates the position of each particle according to its experience and that of its neighbours. The best prior particle position is saved as the personal best (pbest), while the best position acquired so far by the swarm is the global best (gbest). At every iteration, BPSO updates the position and the velocity of every particle andthen finally outputs the best solution [51].

5.3.1.3. Random Subset Feature Selection (RSFS)

Random Subset Feature Selection (RSFS) is used to detect a subset of features, which is more suitable than the original variable set [52]. RSFS evaluates every variable depending on its adequacy in the different feature combinations. The values of the relevance are not based on the preceding options in the selection procedure but are a by-product of several independent tests [53].

In RSFS, the random subset classification is carried out as often as it is crucial to differentiate appropriate features from variables that merely seem beneficial because of the random factors of the process. Consequently, there is no greedy backward elimination needed, and the feature set quality improves progressively as arbitrary aspects in the selection procedure are averaged out [54].

RSFS takes as input an initial set of features (F) alongside a learning algorithm and a criterion function. It initializes the algorithm with a dummy feature set zj, a hypothesis for feature relevance, and a null set. The algorithm starts by selecting a

random subset Si of n features ($|S_i| = n$) from the complete set F by examining a uniform distribution. Then, it performs a classification using the classifier on (S_i) and computes the value of the given criterion function (c_i) to measure the classification performance. It will then proceed to update the relevance of all the used features. With this relevance, the algorithm will check the hypothesis to see if the obtained set of features S □ F completely surpasses the relevance ratings of the dummy features. While the hypothesis is true, the algorithms keep adding the relevant features fj to the set (N). This whole process is replicated until the number of features surpasses the random baseline or if the classifiers run for a specific number of rounds. The final output will present an ensemble feature subsets that exceed the baseline.

5.3.2. Rule-Based Classifiers

A rule-based classification is an impressive tool that is widely used to perform classification. The term rule-based classification refers to any classification algorithm or system that uses IF-THEN rules for prediction. These schemes usually follow three steps: Rule-Induction Algorithm, which extracts appropriate IF-THEN rules from the data, Rule Ranking Measures, which are values used to determine the effectiveness of a condition in acquiring accurate prediction, and Class Prediction Algorithm, which predicts the class of the new example based on the IF-THEN rules that are determined by the rule induction algorithm [55].

In this section, we present the rule-based classifiers that we used in this work.

5.3.2.1. Decision Tree Classifier: CART

The decision tree is a structure constituted of nodes and branches, with sthe last nodes referred to as leaves. A suitable splitting attribute a_i is allotted to each node L_q which is not a leaf. The most critical part of the decision tree algorithm is to assign the element to the considered nodes. An attribute is typically selected based on an impurity measure. The measure is computed for the subset S_q of a training dataset S. This measure determines the split-measure function for each attribute. Depending on the element of choice, each node splits into child nodes. These nodes are linked with their parent nodes by branches. There are two types of decision trees; binary and non-binary [56].

CART is a binary decision tree. The impurity measure of this algorithm is in the form of the Gini index. It starts by splitting a root node L0 into nodes. Each node L_q processes a subset S_q of a training dataset S. In case all elements of S_q are of the same class, the node is identified as a leaf, and there is no split. According to the

Gini index, the algorithm selects the best element to split from the available attributes in the considered nodes. A set of attribute values divide into two sections for each a_i, which in turn splits S_q into two subsets. Among the sections, the one that maximizes the Gini index is the optimal partition. The Gini index is obtained when records are allocated equally among all classes. From this partition, the node L_q that acquires the most valid value for the split-measure function is chosen for the split. The algorithm prolongs the process until there are no more nodes to compute, measure, and split or the algorithm fulfils the stopping criteria of choice.

5.3.2.2. Decision Tree Classifier: J48 (C4.5)

J48, also referred to as C4.5, is a non-binary decision tree. It is an upgraded version of the ID3 algorithm by Ross Quinlan [57]; it establishes the information entropy as a split-measure function. This algorithm follows almost the same steps as CART. However, instead of the Gini index, it uses the ratio of the information entropy and a function called the split information to determine the splits to extract the optimal partitions and nodes.

5.3.2.3. OneR Classifier

OneR is an algorithm implemented in classification. It generates a one-level decision tree. It is a practical and flexible technique capable of handling missing values and numeric attributes [58]. This technique generates a set of classification rules for every element. It computes the error rate of each variable. To do so, it uses the percentage of instances that do not belong to the majority class of the attributes. It subsequently selects the feature with the lowest error rate as its "one rule"; it starts with a selection of elements and produces a set of rules for each one. The algorithm generates these rules as follows: for each attribute, it sums up how often the value of every target class appears and deduces the majority class. Depending on the target class, it generates a set of rules befitting the attribute. It finishes by calculating the error ratio for each set and selects the set of rules with the lowest rate as its primary classification rule set.

5.3.2.4. PART Classifier

PART is a rule learner that uses the separate-and-conquer method. It builds a rule for each instance, then removes said instance and continues creating rules recursively until there are no instances left. It combines the C4.5 and Ripper algorithms and attempts to avoid their respective problem [59]. The algorithm

generates sets of rules named “decision lists”; at each iteration, a pruned DT is built for the current batch of instances. A rule is then created based on the leaf with the broadest coverage. The tree is then discarded. It continues to build decision trees recursively until there are no more instances (this adds flexibility and simplicity to the algorithm). New data is tested with each rule in the list. Then the class of the first matching rule is allotted to the item.

5.3.3. Ensemble Methods

5.3.3.1. Random Forest Classifier

Random Forest (RF) is an ensemble learner. First introduced by Breiman [60], it has several advantages for the remote sensing issues. It is efficient at handling large datasets, capable of handling thousands of input examples without variable deletion, gives estimates of which instances are crucial in the classification [61].

RF is a combination of binary decision tree classifiers, with each classifier contributing to a single vote for the class prediction. This algorithm is often a collection of trees ranging from hundreds to thousands. The tree base learners usually follow a CART methodology but differ from it as it uses a two-stage randomization procedure. This classifier works as follows.

It starts by taking N bootstrap samples from the input dataset and grows a tree for each example. At each node, it randomly selects M variables to be split; then grows the tree so that each final node has enough node size cases. It aggregates the information obtained from the grown trees to determine a hypothesis for the majority vote classification. Then, it finally uses the hypothesis h(x) to compute an out-of-bag (OOB) error rate using the data not present in the bootstrap sample.

5.3.3.2. Boosting Techniques

Boosting is a technique that generally enhances the accuracy of a learning algorithm. It constructs an ensemble of weak learners sequentially. These classifiers are simple models that perform marginally better than random models.

Boosting methods start by distributing the same weight to all instances of the training data then building classifiers iteratively. Within each iteration, a weak classifier computes the weight of the samples according to the accuracy of the classification. The weights of the misclassified examples are updated. The final weights are used to produce a hypothesis, which will be utilized for prediction [62].

There are many widely used and powerful boosting techniques like XGBoost, Gradient Boosting, and LogitBoost. However, as most of these are tree-based classifiers, they require tree-based learners, which donot expand in the case of this study, where some algorithms are not tree-based. In our case, we relied on the most used boosting technique: AdaBoost.

5.4.2.3. Proposed Method

In this section, we present the most impactful algorithms in our study and introduce the proposed approach for corporate bankruptcy prediction.

Fig. (**5.1**) presents the structure that our approach follows. Accordingly, it brings to light three main steps to achieve a corporate bankruptcy prediction model: feature selection phase, classification phase, and testing phase.

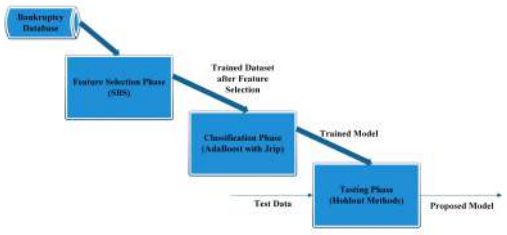

Fig. (5.1). Structure of the Proposed Approach.

5.4.1. Feature Selection Phase

This phase uses the properties of a feature selection algorithm to extract features to be used to achieve the most promising results for the classification. In our method, the best algorithm is Sequential Backward Selection (SBS). It extracted 26 features from the original 64 features from the polish bankruptcy dataset. These features have been deemed the optimal subset for corporate bankruptcy prediction.

Sequential backward selection (SBS) is one of the sequential feature selection algorithms. It differs from SFS primarily in the initialization and selection procedure. Algorithm 1 is the pseudo-code for the SBS algorithm.

As shown in algorithm 1, The SBS algorithm takes as input the complete feature set of the Polish bankruptcy dataset Y, the AdaBoost classifier, the number of features to be selected, and the AUC criterion function. It initializes with a full set X_0 corresponding with Y. It starts the process of eliminating the variables from

X_k. Each feature x that maximizes the criterion function when removed and helps obtain the best classifier performance is removed from X_k. The process repeats until it reaches the specified number of features. The algorithm finally outputs a subset of features containing the eliminated variables.

We remove the obtained feature set from the original dataset and use the remaining variables as input for the classification phase.

Algorithm 1: High-Level Pseudo-code for Sequential Backwards Selection (SBS) Algorithm.

Algorithm: Sequential Backwards Selection (SBS) algorithm
Input: -The set of all features of the Polish bankruptcy dataset Y, -A classifier (AdaBoost), -Number of features to be selected p and criterion function **Output**: A correlated subset of features Xk={x_i l j=1,2,...,k; x_i ϵ Y}, where k=(0,1,2,...,d) **Initialization**: X_o=Y, k=d 1-Step 1 (Exclusion): 1.1-X= arg max$J(x_k-x)$, where x ϵ Xk 1.2-$X_{k-1}=X_k-X$ 1.3-k = k-1 2-Go to Step 1 3-**Termination**: k = p

5.4.2. Classification Phase

The first step of this phase consists of pre-processing the data. It uses the holdout method to split the new Polish bankruptcy dataset, which contains the features extracted in the feature selection phase. The splitting involves 70% for the training samples and 30% for the testing samples.

The classification phase uses the training samples as input for the most effective algorithms to resolve the obstacles of bankruptcy prediction. These algorithms are AdaBoost and JRip. Compared to other algorithms, AdaBoost is less vulnerable to overfitting and quite adaptable in battling imbalanced data. However, it is a black-box model. It is also sensitive to noise and outliers, which is the case for most boosting techniques. With the reduction in feature space, the problem of outliers and noise becomes obsolete. Subsequently, it allows AdaBoost to efficiently manage the issue of data imbalance and boost the learning ability of its base classifier. The base learner used in this study is JRip. It resolves the problem of the black-box nature of the model by providing the rules used in making the prediction.

a. JRip (Ripper) Overview

RIPPER was proposed by Cohen [63]. It is used to conduct a global optimization process on the initial ruleset. The motivation for this algorithm was to place or revise each rule individually to amplify the accuracy of the ruleset [64].

This technique follows a three-step process: Grow, Prune, and Optimize. Algorithm 2 shows the outline of RIPPER.

Algorithm 2: High-level Pseudo-code for JRip algorithm.

Algorithm: JRip algorithm
Input: - A sequence of m instances S = $\{(x_1, y_1)\ldots(x_m, y_m)\}$ where $x_i \epsilon X_k$ with labels $y_i \epsilon Y=\{0,1\}$ from Polish bankruptcy dataset, - F: Number of folds (optional), - O: Number of optimization (optional) - N: Number of minimal weights (optional) - E: Check error rate (default: check) (optional) - P: pruning check (default: prune) (optional) **Output**: final rule set RS **Initialize**: a rule set RS = {} 1- For each class: 2- [Building Stage]: Generate an initial rule set RI 3- Repeat: 4- Grow phase: grow one rule by adding conditions that satisfy the highest information gain: p(log(p/t)-log(P/T)) until the rule is 100% accurate 5- Prune phase: prune each rule according to the pruning metric: p/(p+n) 6- Until error rate >= 0.5 or there are no more positive instances or the description length (DL) of the ruleset is 64 bits greater than the smallest DL 7- [Optimization Stage]: 8 From RI, generate two variants A and B (A from an empty rule and B from the full rule) using the grow phase 8- Prune A and B on a metric: (TP+TN)/(P+N) 9- Compute the original rule according to the description length DL of A and B 10- Select the variant with the minimal DL for RI 11- If there are residual positives: 12- ([Building Stage]) 13- Else: 14- ([Deletion Stage]) 15- [Deletion Stage]: 16- Delete the rules from the ruleset that would increase the DL of the whole ruleset 17- Add resultant ruleset to RS 18- End For

As shown in algorithm 2, JRip is a Java-based implementation of the RIPPER algorithm. This method initializes an empty ruleset RS and starts looping through each class, from the less dominant to the most frequent. The rule generation

process has three stages. The building stage splits into two phases: the grow phase is where every value of each attribute gets computed. The condition that has the highest information gain is selected. One rule is grown by greedily appending conditions until it is 100% accurate. The Prune phase prunes each of the generated rules incrementally, according to a specific pruning metric. These two phases repeat until they meet one of the stopping criteria (the mistake rate is more than 0.5, or no more positive instances occur or the rule set's description length (DL) exceeds the computation's minimum DL. This stage generates an initial rule set RI, which will be utilized in the second stage. The optimization stage generates two variants from randomized data using the grow and prune phases. One variant must be created from an empty rule and the other by greedily appending antecedents to the initial condition. The algorithm selects the variant with the minimal DL as an example in RI. If after all of RI being examined, there are still positive samples, then new rules are grown using the building stage. Otherwise, the ripper moves onto the final stage, which deletes the conditions that could enhance the DL of the complete ruleset. Then, it adds the found ruleset to RS, which will be the final output of the algorithm.

a. AdaBoost Overview

For binary classification, the most commonly used boosting algorithm is AdaBoost. This algorithm was proposed by Freund and Schapire [65]. The following pseudo-code describes the working of this technique.

Algorithm 3: High-level Pseudo-code for AdaBoost Algorithm.

Algorithm: AdaBoost algorithm
Input: -A sequence of m instances S = {(x_1, y_1)...(x_m, y_m)} where $x_i \epsilon X_k$ with labels $y_i \epsilon Y=\{0,1\}$ from Polish bankruptcy dataset, -Weak learner -T (number of iterations) **Output**: The final classifier with hypothesis Hx=sign(∑t=1Tαt htx)**Initialize**: $D_1(i)=1/m$ for all i =1,.....,m 1-For t = 1 to T 2- Call Weak learner using distribution D_t 3- Get Weak classifier and obtain hypothesis ht: X→{-1,+1} 4- Calculate the error rate e_t∑i=1mDti[ht (xi)≠yi] of h_t 5- If $e_t > 0.5$ then T=t-1 and abort loop 6- set αt=12 ln□(1-etet) 7- Update Dt+1(i)=Dt i exp□(-αt yt hti)Zt (Z is a normalization factor)

As shown in algorithm 3, AdaBoost takes as input a training set of m examples from the Polish bankruptcy dataset to perform binary classification. It initializes with a weight function $D_t(x)$, which distributes the same weight to all training samples. In each iteration t, the weak learner is used to create a classifier with a

hypothesis h_t, which computes the error rate e_t of the training samples. The algorithm uses the error rate to adjust the probability distribution $D_t(x)$. This adjustment puts more weight on the misclassified training examples and less weight on the ones that were classified correctly in the previous iteration. This process sequentially constructs several weak classifiers for T iterations. The algorithms finally output a classifier with a hypothesis H, obtained using a weighted vote of the previous individual classifiers. The weight of each of these classifiers is obtained using the accuracy of the distribution D_t that they used.

The following pseudo-code represents a summary of the adapted model for our problem statement.

Algorithm 4 presents the structure of the adapted algorithm for our problem statement, which we referred to as SBSAda-Rip. The algorithm uses 70% of the data and iterates through T rounds. The method initializes with a distribution D_1 and an empty ruleset RS. In each round t, AdaBoost calls upon a JRip classifier using the distribution D_t. The JRip classifier goes through a series of stages. It starts at the building stage. It builds (while satisfying the highest information gain: p(log(p/t)-log(P/T)) and prunes (according to the pruning metric: p/(p+n)) one rule at a time. When it reaches at one of the termination criteria, it creates an initial ruleset RI. The optimization stage uses RI to split the rules and optimize them by re-growing and re-pruning them (using (TP+TN)/(P+N)). The best ones are selected and re-added to RI. The final stage, which is the deletion and selection stage, uses the new RI to produce the final rule set RS. This stage deletes the conditions that hinder the ruleset and adds the resultant ruleset to RS. From the computation performed in the JRip classifier, AdaBoost obtains a hypothesis ht. The error rate of the hypothesis is then calculated, and if it is superior to 0.5, then the distribution D_{t+1} is updated. This whole process is replicated until there are no more misclassified instances in the data. In the end, we obtain the most suitable classifiers with hypotheses H(x) and their rulesets.

5.4.3. Testing Sub-Phase

This phase measures the performance of the trained model with the test data. We obtain the proposed model from the chosen best hypothesis and ruleset, which we consider the optimal option.

5.5. Validation

To evaluate our algorithms' performance for generating prediction rules, this section addresses two primary research questions.

Algorithm 4: High-level Pseudo Code of the enhanced Boosting model.

Algorithm: SBSAda-rip algorithm
Input: -A sequence of m instances S = {(x_1, y_1)...(x_m, y_m)} where $x_i \epsilon X_k$ (X_k the subset obtained from SBS) with labels $y_i \epsilon Y=\{0,1\}$ from Polish bankruptcy dataset -JRip algorithm -T (number of iterations) **Output**: The final classifier with hypothesis H(x) and its final rule set RS **Initialize**: $D_1(i)=1/m$ for all i =1,.....,m, a rule set RS = {} 1-For t = 1 to T 2- Call JRip using distribution D_t: 3- For each class: 4- [Building Stage]: Generate an initial rule set RI 5- Repeat: 6- Grow phase: grow one rule by adding conditions that satisfy the highest information gain: p(log(p/t)-log(P/T)) until the rule is 100% accurate 7- Prune phase: prune each rule according to the pruning metric: p/(p+n) 8- Until error rate >= 0.5 | no more positive instances | description length (DL) of the ruleset is 64 bits greater than the smallest DL 9- [Optimization Stage]: 10- from RI, generate two variants, A and B (A from an empty rule and B from the full rule) using the grow phase 11- prune A and B on metric: (TP+TN)/(P+N) 12- compute the original rule according to the description length DL of A and B 13- select the variant with the minimal DL for RI 14- If there are residual positives: 15- ([Building Stage]) 16- [Deletion Stage]: 17- Delete the rules from the ruleset that would increase the DL of the whole ruleset 18- Add resultant ruleset to RS 19- End For 20- Get JRip classifier and obtain hypothesis h_t 21- Calculate the error rate $e_t \sum i=1mDti[ht(xi) \neq yi]$ of h_t 22- If $e_t > 0.5$ then T=t-1 and abort loop 23- set αt=12 ln□(1-etet) 24- Update Dt+1(i)=Dt i exp□(-αt yt hti)Zt (Z is a normalization factor)

5.5.1. Research Questions

RQ 01: To what extent the proposed approach can discover an accurate classification model from the given dataset compared to other methods?

RQ02: What is the significant comparison between the existing model and the proposed model ?

5.5.2. Description of the Experimental Database

To validate our work, we used the financial information of Polish companies in

the manufacturing department. The data set includes information about bankrupt and non-bankrupt corporations. Several Psolish firms went bankrupt in this sector after 2004 (Fig. **5.2**) [66].

ID Description	ID Description
X1 net profit / total assets	X33 operating expenses / short-term liabilities
X2 total liabilities / total assets	X34 operating expenses / total liabilities
X3 working capital / total assets	X35 profit on sales / total assets
X4 current assets / short-term liabilities	X36 total sales / total assets
X5 [(cash + short-term securities + receivables - short-term liabilities) /(operating expenses - depreciation)] * 365	X37 (current assets - inventories) / long-term liabilities
X6 retained earnings / total assets	X38 constant capital / total assets
X7 EBIT / total assets	X39 profit on sales / sales
X8 book value of equity / total liabilities	X40(current assets - inventory - receivables) / short-term liabilities
X9 sales / total assets	X41 total liabilities / ((profit on operating activities +depreciation) * (12/365))
X10 equity / total assets	X42 profit on operating activities / sales
X11 (gross profit + extraordinary items + financial expenses) / total assets	X43 rotation receivables + inventory turnover in days
X12 gross profit / short-term liabilities	X44 (receivables * 365) / sales
X13 (gross profit + depreciation) / sales	X45 net profit / inventory
X14 (gross profit + interest) / total assets	X46 (current assets - inventory) / short-term liabilities
X15 (total liabilities * 365) /(gross profit + epreciation)	X47 (inventory * 365) / cost of products sold
X16 (gross profit + depreciation) / total liabilities	X48 EBITDA (profit on operating activities - depreciation) / total assets
X17 total assets / total liabilities	X49 EBITDA (profit on operating activities - depreciation) / sales
X18 gross profit / total assets	X50 current assets / total liabilities
X19 gross profit / sales	X51 short-term liabilities / total assets
X20 (inventory * 365) / sales	X52 (short-term liabilities * 365) / cost of products sold)
X21 sales (n) / sales (n-1)	X53 equity / fixed assets
X22 profit on operating activities / total assets	X54 constant capital / fixed assets
X23 net profit/sales	X55 working capital
X24 gross profit (in 3 years) / total assets	X56 (sales - cost of products sold) / sales
X25 (equity - share capital) / total assets	X57 (current assets - inventory - short-term liabilities) /(sales - gross profit - depreciation)
X26 (net profit + depreciation) / total liabilities	X58 total costs /total sales
X27 profit on operating activities /financial expenses	X59 long-term liabilities/equity
X28 working capital / fixed assets	X60 sales / inventory
X29 logarithm of total assets	X61 sales / receivables
X30 (total liabilities - cash) / sales	X62 (short-term liabilities *365) / sales
X31 (gross profit + interest) / sales	X63 sales / short-term liabilities
X32 (current liabilities * 365) /cost of product sold	X64 sales / fixed assets

Fig. (5.2). The Feature Set of the Polish Bankruptcy Dataset.

The economic factors describing the state of nearly 700 companies in financial distress were collected between 2007 and 2013. As for the still-operating companies, the information was obtained between 2000 and 2012. This dataset consists of 64 financial indicators and five data subsets.

Table **1** presents the dataset subsets names, the year of the extraction of financial rates within the forecast period, the bankruptcy status, the number of samples for each class label, and the sum of instances. The attributes considered in the studies are defined in detail in Fig. (**5.2**).

Table 1. Description of the Polish Bankruptcy Dataset for Each Year.

Dataset	Features from	Bankruptcy after	Bankrupt		Not bankrupt		Sum
			N	Rate	N	Rate	
Year 1	1st year	5 years	271	3.86	6,756	96.14	7,027
Year 2	2nd year	4 years	400	3.94	9,773	96.06	10,173
Year 3	3rd year	3 years	495	4.71	10,008	95.29	10,503
Year 4	4th year	2 years	515	5.26	9,277	94.74	9,792
Year 5	5th year	1 year	410	6.94	5,500	93.06	5,910

5.5.3. Evaluation Criteria

As it is critical to validate the results of any learning algorithm, it is imperative to use adequate evaluation measurements. There are several evaluation metrics used to measure the efficiency of a classifier, such as accuracy, precision, sensitivity/recall, specificity. We have chosen to use three evaluation measurements: AUC, G-mean, and F-measure. These metrics can be defined in the confusion matrix as that given in Table **2**:

Table 2. A confusion matrix.

Actual class (%)		
Predicted class	Non-bankrupt (P)	Bankrupt (N)
Non-bankrupt (P)	**TP**	**FP**
Bankrupt (N)	**FN**	**TN**

a) AUC-ROC

The AUC-ROC curve is a measure that uses various threshold settings to determine the performance of classification problems. ROC presents the curve of the probability, and AUC represents the degree of separability. This metric shows a model's ability at discerning classes. As the AUC of a model increases, so does its ability in predicting 0s as 0s and 1s as 1s. By analogy, the higher the AUC, the better the model is in distinguishing between bankrupt and not-bankrupt firms.

$$AUC = \frac{Sensitivity + Specificity}{2}$$

Sensitivity, also called hit rate or recall, measures how capable a classifier is at recognizing positive examples.

$$Sensitivity = \frac{TP}{FN + TP}$$

Specificity measures how capable a classifier is at recognizing negative examples.

$$Specificity = \frac{TN}{FP+TN}$$

G-mean

The Geometric Mean (G-Mean) measures how well a model can balance the classification performance of the majority class (bankrupt) and minority class (not bankrupt). A low G-Mean indicates a poor classification of the positive cases compared to the negative ones. This measure helps in determining if a model is overfitting the negative class and/or under-fitting the positive class.

$$Geometric\ mean\ = \sqrt{Sensitivity\ \times\ Specificity}$$

F1-score

F-measure or F1-score is the harmonic mean of sensitivity and precision. It is used to measure a test's accuracy. The formula for this measure is given by:

$$F1 - score = 2 * \frac{Sensitivity * Precision}{Sensitivity + Precision}$$

Where precision is the ratio of predicted positive examples which are positive.

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

5.6. RESULTS AND DISCUSSION

5.6.1. Parameter Settings

In this section, we present the impactful parameters used for the approach with the performance they accomplished. In it, the following parameters are the ones used in the optimization of the results.

1. SBS: This technique only uses three parameters; the classifier, the number of features to be extracted, and the criteria. In this algorithm, we used the MLExtend library in python. As such, when determining the classifier, we chose to use the default version of AdaBoost, which uses CART as the base learner. For the number of features, we again used the base function of AdaBoost, which allows determining the number of features with the most impact on the classification (26 variables in our case). Finally, for the criteria, we used AUC.
2. AdaBoost.M1: To be able to incorporate several of the algorithms used in the study with AdaBoost, we used the Waikato Environment for Knowledge Analysis (WEKA) wrapper framework in python. This algorithm possesses several classifier-specific parameters. The following are the ones used in the experiments.

Weight Threshold (P): It refers to the percentage of weight mass for the training. Each training sample is accorded weights iteratively. This allows each iteration to focus harder on the previously misclassified portions of the training examples. Its default value is 100, which indicates no weight distribution. We varied this value between 70 and 90 to obtain the best results. Our model exhibited sensitivity to this parameter.

Number of Iterations (I): These are the number of iterations that AdaBoost.M1 calls for the learner, which is JRip in this case. The default value is 10, which we varied between 10 and 80. Our method exhibited sensitivity when tuning this parameter.

Base Classifier (W): It refers to the base classifier that AdaBoostM1 uses as a base learner. The default is the Decision Stump algorithm. In our model, we used JRip.

1. JRip: To use this algorithm, we implemented the WEKA wrapper framework in python. This algorithm possesses several classifier-specific parameters. The following are the ones used in the experiments.

The number of folds (F): It refers to the number of folds used to reduce the error pruning of a set. The default value is 3, and one fold is used as a pruning set. We varied it between 2 and 4. Our model exhibited sensitivity to this parameter.

The number of minimal weights (N): It refers to the minimal weight of an instance within a split. The default value is 2, which we varied between 2 and 4. Our model exhibited some sensitivity to this parameter.

The number of optimization (O): It refers to the number of runs the algorithm needs to perform to optimize the results. The default value is 2, which we varied between 2 and 4. Our model exhibited some sensitivity to this parameter.

Error rate check (E): It refers to whether or not to use 0.5 as a stopping criteria for verifying the error rate. The default value is "to check"; in our experiments, we maintained this value.

Pruning check (P): It refers to whether or not to use pruning in building the rules. The default value is to use pruning. In our experiments, we maintained this value.

In the experimentation process, we selected a set of parameters having the most impact on the training. We used all previously mentioned parameters with varied values to obtain the best results possible for our experiments. Table **3** demonstrates the values of the parameters used and the results obtained for our approach on each dataset.

Table 3. Parameter Settings for the SBSAda-Rip algorithm for Each Dataset.

Dataset	Method	AdaBoost params		JRip params			Results		
		P	I	F	N	O	AUC	G-mean	F
Year1	SBSAda-Rip	80	74	4	2	3	94.5	92.9	97.1
Year2	SBSAda-Rip	90	40	4	2	3	88.3	88.1	96.1
Year3	SBSAda-Rip	90	75	3	2	4	89.4	89.1	95.8
Year4	SBSAda-Rip	90	45	3	3	2	91.2	87.2	96.1
Year5	SBSAda-Rip	90	35	4	3	2	92.8	81.4	94.3

5.6.2. Results for Research Question 1

In this section, we assess the efficiency of our proposed approach with other techniques on the resolution of data imbalance, feature selection, and classification rule extraction.

Table **4** demonstrates the performance in the AUC measure of all the used algorithms in this study. It shows that on account of each dataset, SBSAda-Rip outperforms all other techniques used. The table shows the highest AUC rate of this technique ranging from 88.3% to 94.5%. It is sometimes closely followed or outperformed by AdaBoost with CART when using other feature subsets. These results validate our choice of selected features and the ability of our approach to overcome the data imbalance problem. Thus, we also deduce that the extracted features are the most effective for corporate bankruptcy prediction.

Table 4. Performances of Bankruptcy Prediction Model using AUC Metric.

Feature selection methods	Classifiers	1st year	2nd year	3rd year	4th year	5th year
SBS	**SBSAda-Rip**	**94.5**	**88.3**	**89.4**	**91.2**	**92.8**
	AdaBoost+C4.5	86.3	84.1	82.3	82.6	87.5
	AdaBoost+OneR	82.1	72.5	75.6	82.5	87
	AdaBoost+PART	86.4	78.3	81.2	82.7	86.9
	AdaBoost+CART	91.6	84.3	86.7	90.1	89.7
	AdaBoost+RF	88.3	84.2	70.2	71.2	78.6
SFS	AdaBoost+JRip	91	83.2	86.6	90.5	92.6
	AdaBoost+ C4.5	85.8	78.8	84.6	79.9	87.2
	AdaBoost+OneR	83.7	73.4	76.1	82.3	88.5
	AdaBoost+PART	86.1	75.1	81.1	80.6	89.4
	AdaBoost+CART	92.3	84.4	86.7	88.3	90.8
	AdaBoost+RF	89.5	78.8	70.7	70.2	81.2
PSO	AdaBoost+JRip	90.7	86.9	87.5	90.7	92.5
	AdaBoost+ C4.5	89.5	81.1	78	80.6	85.7
	AdaBoost+OneR	83.1	70	77.5	79.8	87.9
	AdaBoost+PART	83.5	75.6	79.1	81.6	86.8
	AdaBoost+CART	91.5	83.3	86.6	89.3	89.7
	AdaBoost+RF	86.8	84.2	61.4	67	74.2
RSFS	AdaBoost+JRip	90.7	86.6	87.7	91	91
	AdaBoost+ C4.5	89.9	80.1	76.9	81.2	86.9
	AdaBoost+OneR	83.3	72.9	77.9	79.8	87.7
	AdaBoost+PART	83.5	78.5	77.9	82.1	87
	AdaBoost+CART	91.5	82.8	86.3	89.7	71.7
	AdaBoost+RF	86.8	81	61.1	68.1	78.6

Table **5** demonstrates the performance in the geometric mean measure of all the used algorithms in this study. This table shows that there are some instances where some of the other techniques perform better than our proposed approach. However, the average g-mean of SBSAda-Rip, which ranges between 87.2% and 92.9, proves that it is the best compared to the other techniques. It also strengthens the validation of our approach in the resolution of the mentioned obstacles for corporate bankruptcy prediction.

Table 5. Performances of Bankruptcy Prediction Model using G-mean.

Feature selection methods	Classifiers	1st year	2nd year	3rd year	4th year	5th year
SBS	**SBSAda-Rip**	**92.9**	**88.1**	**89.1**	**87.2**	**81.4**
	AdaBoost+ C4.5	78.5	73.6	72.2	63	76.2
	AdaBoost+OneR	77.9	63.9	57.4	64.7	75.4
	AdaBoost+PART	90.2	76.6	83.7	64.1	64.2
	AdaBoost+CART	85	90.6	65.8	57.6	91
	AdaBoost+RF	79.6	75.3	80.2	54.6	78.6
SFS	AdaBoost+JRip	87	78.1	80.3	78.6	81.8
	AdaBoost+ C4.5	78.7	66.2	70.8	49.5	74.1
	AdaBoost+OneR	79.6	47.5	53.5	61.4	73.3
	AdaBoost+PART	94.5	74.7	81.3	66.7	75.1
	AdaBoost+CART	91.5	76.2	67.7	79.5	91
	AdaBoost+RF	83.7	63.9	69.4	41	79.4
PSO	AdaBoost+JRip	89.1	85.5	82	84.6	79.4
	AdaBoost+ C4.5	82.2	85.2	65.2	63.2	71.3
	AdaBoost+OneR	74.2	40.1	63.3	57.5	77.4
	AdaBoost+PART	72.3	73.6	62.1	65.1	78.7
	AdaBoost+CART	86.5	74.3	65.5	77.5	91
	AdaBoost+RF	69.4	61.5	54.8	56	76.6
RSFS	AdaBoost+JRip	92.2	87.3	73.6	76.8	87.4
	AdaBoost+ C4.5	78.4	82	72.4	67	72.4
	AdaBoost+OneR	71.2	41.5	60.1	66	77.6
	AdaBoost+PART	72.3	67	61.3	63.6	76.7
	AdaBoost+CART	86.5	72.5	68.1	79	81.6
	AdaBoost+RF	69.4	55.7	54.3	43.7	72.8

Table **6** demonstrates the performance in the F-measure of all the used algorithms in this study. This table shows that, much like table 5, some instances of other algorithms perform slightly better than our proposed approach. Nevertheless, when compared to the performance of our technique, which ranges between 94.3% and 97.1%, it is evident that on average, it performs better than the others as it maintains stable results on all the datasets. It leads us to validate the performance of SBSAda-Rip for corporate bankruptcy prediction

Table 6. Performances of Bankruptcy Prediction Model using F-measure.

Feature selection methods	Classifiers	1st year	2nd year	3rd year	4th year	5th year
SBS	**SBSAda-Rip**	**97.1**	**96.1**	**95.8**	**96.1**	**94.3**
	AdaBoost+ C4.5	95.6	95.4	94.8	93.8	93
	AdaBoost+OneR	95.8	94.9	94.2	94	93.2
	AdaBoost+PART	96.5	95.8	95.5	93.7	91.3
	AdaBoost+CART	96.2	95.8	94.7	95.3	93.6
	AdaBoost+RF	94.6	95.3	94.9	93.1	92.9
SFS	AdaBoost+JRip	96.8	95.8	96.1	95.6	94.4
	AdaBoost+ C4.5	95.7	95.2	95.1	93.1	92.5
	AdaBoost+OneR	95.5	94.5	94.1	93.9	93
	AdaBoost+PART	96.8	95.4	95.3	94	93
	AdaBoost+CART	96.9	95.3	94.7	95.3	93.6
	AdaBoost+RF	95.7	94.9	94.5	92.9	9.3
PSO	AdaBoost+JRip	96	96.4	95.7	95.2	94.1
	AdaBoost+ C4.5	95.2	95.9	94.6	93.7	92.5
	AdaBoost+OneR	94.9	94.3	94.4	93.5	93.2
	AdaBoost+PART	94.9	95.2	94.7	93.8	92.4
	AdaBoost+CART	96.5	95.3	94.7	95.2	93.6
	AdaBoost+RF	94.6	94.8	94.1	93.1	92.4
RSFS	AdaBoost+JRip	96.8	96.5	95.3	95	94.2
	AdaBoost+ C4.5	95.1	95.4	94.5	93.6	92.4
	AdaBoost+OneR	94.7	94.3	94.4	93.8	93.4
	AdaBoost+PART	94.9	95	94.4	93.7	93.1
	AdaBoost+CART	96.5	95.1	94.7	95.4	94.2
	AdaBoost+RF	94	95.5	94	92.9	91.9

5.6.3. Results for Research Question 2

Numerous studies have been successful in predicting corporate bankruptcy. Thus, we analyze the performance of our approach with that of different techniques conducted on the same Polish datasets. Table **7** presents the results of other studies and our own. This comparison takes into account two factors: the AUC measure and the interpretability of the models. The interpretability examines the ability of the classifier in extracting decision rules, the number of these rules, and the number of the extracted features.

Table 7. Comparison of our proposed model with similar works for Polish Bankruptcy Dataset.

	Methods	AUC	Extracted features	N. rules
Our model	SBSAda-Rip	94.5	26	4
(Zięba *et al*., 2016)	LDA	63.9	-	-
(Q. Zhang, Wang, Lu, Wang, & Ma, 2018)	MLP	54.3	-	-
(Tuong, Son, Vo, Lee, & Baik, 2018)	LR	62.0	-	-
(Uthayakumar, Metawa, Shankar, & Lakshmanaprabu, 2018)	JRip	52.3	-	-
	CJRip	74.5	-	-
	J48	71.7	-	-
	CJ48	65.8	-	-
	AB*	91.6	-	-
	SVM	50.2	-	-
	RF	85.1	-	-
	XGB*	94.5	-	-
	XGBE*	95.3	-	-
	EXGB*	95.9	20	-
	Isolation Forest	93.0	-	-
	One-Class SVM	92.0	-	-
	Multivariate Gaussian	89.0	-	-
	NN	84.0	-	-
	GBDT*	85.0	-	-
	RFCI	86.8	-	-
	FCP	-	-	21

From Table **7**, we can detect that our model surpasses all the ones from T *et al*.'s study [67] and most of J *et al*.'s study [68]. Still, we can detect that it competes with the XGB model in AUC but is outperformed by XGBE and EXGB, making them the better options. This leads us to take into consideration the interoperability measure. In this prospect, we can mention that the number of extracted features of this work is smaller than ours (20 and 26, respectively). However, this work focused more on generating synthetic features and their impact on the prediction rather than extracting classification rules. As such, whether the model is capable of generating optimal decision-rules is still up for debate. This leads us to believe (to our knowledge at least) that in this aspect, our approach is the best choice so far.

CONCLUSION

In the economic field, corporate bankruptcy prediction is considered one of the most crucial and fundamental issues. The research performed on this issue is countless; still, the perfect solution is yet to exist. Each study tends to resolve some of the problems encountered in bankruptcy but fails to solve others. This paper proposes a solution for three primary obstacles for financial failure

prediction; data imbalance, feature space reduction, and rule extraction. To resolve these problems, we used and evaluated several machine learning algorithms. We constructed our proposed model in two phases, feature selection and classification. In the feature selection phase, we used several algorithms to extract an optimal subset of features and identified SBS as the best technique. In the classification phase, we combined the properties of the AdaBoost algorithm of the JRip algorithm to perform the classification. This combination allowed us to resolve the data imbalance issue and obtain comprehensible decision rules for the Polish bankruptcy dataset. The results prove that our model outperforms most of the existing work on the AUC metric. However, when compared to some of the other studies, it falls short in some areas. Still, we find that our approach bested others in the comprehensibility and interpretability of the models. Perhaps, in the future, using a technique similar to the synthetic feature method would optimize our results even more.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] D. Olson, D. Delen, and Y. Meng, "Comparative analysis of data mining methods for bankruptcy prediction", *Decis. Support Syst.,* vol. 52, no. 2, pp. 464-473, 2012.
[http://dx.doi.org/10.1016/j.dss.2011.10.007]

[2] L. Zhou, K. Lai, and J. Yen, "Bankruptcy prediction using SVM models with a new approach to combine features selection and parameter optimisation", *Int. J. Syst. Sci.,* vol. 45, no. 3, pp. 241-253, 2012.
[http://dx.doi.org/10.1080/00207721.2012.720293]

[3] F. Barboza, H. Kimura, and E. Altman, "Machine learning models and bankruptcy prediction", *Expert Syst. Appl.,* vol. 83, pp. 405-417, 2017.
[http://dx.doi.org/10.1016/j.eswa.2017.04.006]

[4] F. Mai, S. Tian, C. Lee, and L. Ma, "Deep learning models for bankruptcy prediction using textual disclosures", *Eur. J. Oper. Res.,* vol. 274, no. 2, pp. 743-758, 2019.
[http://dx.doi.org/10.1016/j.ejor.2018.10.024]

[5] T. Hosaka, "Bankruptcy prediction using imaged financial ratios and convolutional neural networks", *Expert Syst. Appl.,* vol. 117, pp. 287-299, 2019.
[http://dx.doi.org/10.1016/j.eswa.2018.09.039]

[6] A. Chaudhuri, and S. Ghosh, *Bankruptcy prediction through soft computing based deep learning*

technique. Springer, 2017.
[http://dx.doi.org/10.1007/978-981-10-6683-2]

[7] S. Lee, and W. Choi, "A multi-industry bankruptcy prediction model using back-propagation neural network and multivariate discriminant analysis", *Expert Syst. Appl.,* vol. 40, no. 8, pp. 2941-2946, 2013.
[http://dx.doi.org/10.1016/j.eswa.2012.12.009]

[8] J. Wen, "Robust Sparse Linear Discriminant Analysis", *IEEE Trans. Circ. Syst. Video Tech.,* vol. 29, no. 2, pp. 390-403, 2019.
[http://dx.doi.org/10.1109/TCSVT.2018.2799214]

[9] A. Tharwat, "Linear vs. quadratic discriminant analysis classifier: a tutorial", *International Journal of Applied Pattern Recognition,* vol. 3, no. 2, p. 145, 2016.
[http://dx.doi.org/10.1504/IJAPR.2016.079050]

[10] K. Shin, and Y. Lee, "A genetic algorithm application in bankruptcy prediction modeling", *Expert Syst. Appl.,* vol. 23, no. 3, pp. 321-328, 2002.
[http://dx.doi.org/10.1016/S0957-4174(02)00051-9]

[11] N. Ocal, M. Ercan, and E. Kadioglu, "Predicting Financial Failure Using Decision Tree Algorithms: An Empirical Test on the Manufacturing Industry at Borsa Istanbul", *Int. J. Econ. Finance,* vol. 7, no. 7, 2015.
[http://dx.doi.org/10.5539/ijef.v7n7p189]

[12] S. Kim, B. Mun, and S. Bae, "Data depth based support vector machines for predicting corporate bankruptcy", *Appl. Intell.,* vol. 48, no. 3, pp. 791-804, 2017.
[http://dx.doi.org/10.1007/s10489-017-1011-3]

[13] G. Naidu, and K. Govinda, "Bankruptcy prediction using neural networks", *International Conference on Inventive Systems and Control,* vol. vol. 2, 2018pp. 248-251

[14] S. Strohmeier, and F. Piazza, Artificial intelligence techniques in human resource management—a conceptual exploration.*Intelligent techniques in engineering management.* Springer, 2015, pp. 149-172.
[http://dx.doi.org/10.1007/978-3-319-17906-3_7]

[15] L. Zhou, and K. Lai, "AdaBoost Models for Corporate Bankruptcy Prediction with Missing Data", *Comput. Econ.,* vol. 50, no. 1, pp. 69-94, 2016.
[http://dx.doi.org/10.1007/s10614-016-9581-4]

[16] J. Wyrobek, and K. Kluza, "Efficiency of gradient boosting decision trees technique in Polish companies' bankruptcy prediction", *International Conference on Information Systems Architecture and Technology,* Springer, pp. 24-35, 2018.Cham

[17] T. Chen, and C. Guestrin, "Scalable and intelligent learning systemsXgboost: A scalable tree boosting system", *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining,* 2016 pp. 785-794
[http://dx.doi.org/10.1145/2939672.2939785]

[18] P. Chujai, K. Kerdprasop, and K. Kerdprasop, " A Tree-based Ensemble Learning of Minority Cases in the Imbalanced Data". International Information Institute (Tokyo)", *Information,* vol. 18, no. 11, p. 4815, 2015.

[19] M. Pozzoli, and F. Paolone, The Models of Financial Distress. 2017.
[http://dx.doi.org/10.1007/978-3-319-67355-4_3]

[20] C. Peres, and M. Antão, "The use of multivariate discriminant analysis to predict corporate bankruptcy: A review", https://dialnet.unirioja.es/servlet/articulo?codigo=6635579

[21] K. Laurila, Accuracy comparison of accounting-based bankruptcy prediction models of Springate (1978), Ohlson (1980) and Altman (2000) to US manufacturing companies 1990-2018, Master's thesis, Dept. Busi. Acc., Aalto Univ, Finland, 2020.

[22] M. Kovacova, and T. Kliestik, "Logit and Probit application for the prediction of bankruptcy in Slovak companies", *Equilibrium,* vol. 12, no. 4, pp. 775-791, 2017.
[http://dx.doi.org/10.24136/eq.v12i4.40]

[23] V. García, A. Marqués, J. Sánchez, and H. Ochoa-Domínguez, "Dissimilarity-Based Linear Models for Corporate Bankruptcy Prediction", *Comput. Econ.,* vol. 53, no. 3, pp. 1019-1031, 2017.
[http://dx.doi.org/10.1007/s10614-017-9783-4]

[24] Y. Shi, and X. Li, "A bibliometric study on intelligent techniques of bankruptcy prediction for corporate firms", *Heliyon,* vol. 5, no. 12, 2019.e02997
[http://dx.doi.org/10.1016/j.heliyon.2019.e02997] [PMID: 31890956]

[25] B.E. Erdogan, and S.Ö. Akyüz, "Weighted Ensemble Learning by SVM for Longitudinal Data: Turkish Bank Bankruptcy. In Trends and Perspectives in Linear Statistical Inference. Contributions to Statistics. Springer, 2018, Cham",
[http://dx.doi.org/10.1007/978-3-319-73241-1_6]

[26] S. Syed Nor, S. Ismail, and B. Yap, "Personal bankruptcy prediction using decision tree model", *J. Econ. Finance Adm. Sci.,* vol. 24, no. 47, pp. 157-170, 2019.
[http://dx.doi.org/10.1108/JEFAS-08-2018-0076]

[27] T. Slavici, S. Maris, and M. Pirtea, "Usage of artificial neural networks for optimal bankruptcy forecasting. Case study: Eastern European small manufacturing enterprises", *Qual. Quant.,* vol. 50, no. 1, pp. 385-398, 2015.
[http://dx.doi.org/10.1007/s11135-014-0154-0]

[28] S. AOKI and Y. HOSONUMA, *Bankruptcy prediction using decision tree.* Springer: Tokyo, 2004.

[29] A. Gepp, K. Kumar, and S. Bhattacharya, "Business failure prediction using decision trees", *J. Forecast.,* vol. 29, no. 6, pp. 536-555, 2009.
[http://dx.doi.org/10.1002/for.1153]

[30] X. Brédart, "Les symptômes de la faillite: le cas de la Belgique", *Articles,* vol. 90, no. 2, pp. 105-119, 2015.
[http://dx.doi.org/10.7202/1027974ar]

[31] S. Imtiaz, "A Better Comparison Summary of Credit Scoring Classification", *Int. J. Adv. Comput. Sci. Appl.,* vol. 8, no. 7, 2017.
[http://dx.doi.org/10.14569/IJACSA.2017.080701]

[32] U. Dellepiane, M. Di Marcantonio, E. Laghi and S. Renzi, “Bankruptcy Prediction Using Support Vector Machines and Feature Selection During the Recent Financial Crisis., *Int. J. Econ. Finance,* vol. 7, no. 8, 2015.
[http://dx.doi.org/10.5539/ijef.v7n8p182]

[33] Z. Chen, W. Chen, and Y. Shi, "Ensemble learning with label proportions for bankruptcy prediction", *Expert Syst. Appl.,* vol. 146, 2020.113155
[http://dx.doi.org/10.1016/j.eswa.2019.113155]

[34] H. Faris, "Improving financial bankruptcy prediction in a highly imbalanced class distribution using oversampling and ensemble learning: a case from the Spanish market", *Progress in Artificial Intelligence,* vol. 9, no. 1, pp. 31-53, 2019.
[http://dx.doi.org/10.1007/s13748-019-00197-9]

[35] M. Zięba, S. Tomczak, and J. Tomczak, "Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction", *Expert Syst. Appl.,* vol. 58, pp. 93-101, 2016.
[http://dx.doi.org/10.1016/j.eswa.2016.04.001]

[36] G. Wang, J. Ma, and S. Yang, "An improved boosting based on feature selection for corporate bankruptcy prediction", *Expert Syst. Appl.,* vol. 41, no. 5, pp. 2353-2361, 2014.
[http://dx.doi.org/10.1016/j.eswa.2013.09.033]

[37] L. Deng, and Y. Liu, *Deep learning in natural language processing.* Springer, 2018. [http://dx.doi.org/10.1007/978-981-10-5209-5]

[38] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: A survey", *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.,* vol. 8, no. 4, 2018. [http://dx.doi.org/10.1002/widm.1253]

[39] S. Jean, K. Cho, R. Memisevic, and Y Bengio, "On Using Very Large Target Vocabulary for Neural Machine Translation", *arXiv.org,* .https://arxiv.org/abs/1412.2007 [Accessed: 05- Dec- 2020]

[40] R. Ejbali, and M. Zaied, "A dyadic multi-resolution deep convolutional neural wavelet network for image classification", *Multimedia Tools Appl.,* vol. 77, no. 5, pp. 6149-6163, 2017. [http://dx.doi.org/10.1007/s11042-017-4523-2]

[41] S.H. Yeh, C.J. Wang, and M.F. Tsai, "Corporate default prediction via deep learning", *Proceedings of the 34th International Symposium on Forecasting (ISF'14),* vol. Vol. 29, 2014 Rotterdam, The Netherlands

[42] J. Lee, D. Jang, and S. Park, "Deep Learning-Based Corporate Performance Prediction Model Considering Technical Capability", *Sustainability,* vol. 9, no. 6, p. 899, 2017. [http://dx.doi.org/10.3390/su9060899]

[43] M. Soui, S. Smiti, M. Mkaouer, and R. Ejbali, "Bankruptcy Prediction Using Stacked Auto-Encoders", *Appl. Artif. Intell.,* vol. 34, no. 1, pp. 80-100, 2019. [http://dx.doi.org/10.1080/08839514.2019.1691849]

[44] M. Wollowski, *A survey of current practice and teaching of AI,* 2020.https://dl.acm.org/doi/10.5555/3016387.3016491

[45] G. Marcus, "Deep Learning: A Critical Appraisal", *arXiv.org,* .https://arxiv.org/abs/1801.00631 Online

[46] A. Blum, and P. Langley, "Selection of relevant features and examples in machine learning", *Artif. Intell.,* vol. 97, no. 1-2, pp. 245-271, 1997. [http://dx.doi.org/10.1016/S0004-3702(97)00063-5]

[47] S. Hengpraprohm, and P. Chongstitvatana, *Feature Selection by Weighted-SNR for Cancer Microarray Data Classification.* International Journal of Innovative Computing, Information and Control. 2008, 4627-4636

[48] H. Liu, and H. Motoda, *Feature Extraction, Construction and Selection: A Data Mining Perspective.* Springer Science & Business Media, 1998. [http://dx.doi.org/10.1007/978-1-4615-5725-8]

[49] J. Suto, S. Oniga, and P. Pop Sitar, "Comparison of wrapper and filter feature selection algorithms on human activity recognition", *6th International Conference on Computers Communications and Control,* 2016pp. 124-129 [http://dx.doi.org/10.1109/ICCCC.2016.7496749]

[50] A. Whitney, *A Direct Method of Nonparametric Measurement Selection*, 1971. [http://dx.doi.org/10.1109/T-C.1971.223410]

[51] B. Xue, M. Zhang, and W. Browne, "Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms", *Appl. Soft Comput.,* vol. 18, pp. 261-276, 2014. [http://dx.doi.org/10.1016/j.asoc.2013.09.018]

[52] B. Ashok, and P. Aruna, "Comparison of Feature selection methods for diagnosis of cervical cancer using SVM classifier", *Int. Journal of Engineering Research and Applications,* pp. 94-99, 2016.

[53] O. Räsänen, and J. Pohjalainen, *Random subset feature selection in automatic recognition of developmental disorders, affective states, and level of conflict from speech.* Interspeech, 2013, pp. 210-214.

[54] L. Dhyaram, and B. Vishnuvardhan, "RANDOM SUBSET FEATURE SELECTION FOR CLASSIFICATION", *International Journal of Advanced Research in Computer Science,* vol. 9, no. 2, pp. 317-319, 2018.
[http://dx.doi.org/10.26483/ijarcs.v9i2.5496]

[55] M.Z. Asghar, A. Khan, S. Ahmad, M. Qasim, and I.A. Khan, "Lexicon-enhanced sentiment analysis framework using rule-based classification scheme", *PLoS One,* vol. 12, no. 2, 2017.e0171649
[http://dx.doi.org/10.1371/journal.pone.0171649] [PMID: 28231286]

[56] L. Rutkowski, M. Jaworski, L. Pietruczuk, and P. Duda, "The CART decision tree for mining data streams", *Inf. Sci.,* vol. 266, pp. 1-15, 2014.
[http://dx.doi.org/10.1016/j.ins.2013.12.060]

[57] J. Quinlan, *C4.5 - programs for machine learning.* Kaufmann: San Mateo, Calif, 1992.

[58] J. Singh, G. Singh, and R. Singh, "Optimization of sentiment analysis using machine learning classifiers", *Human-centric Computing and Information Sciences,* vol. 7, no. 1, 2017.
[http://dx.doi.org/10.1186/s13673-017-0116-3]

[59] B. Aguinaldo, "21st Century Learning Skills Predictive Model Using PART Algorithm", *Proceedings of the 3rd International Conference on Machine Learning and Soft Computing - ICMLSC 2019,* 2019pp. 134-137
[http://dx.doi.org/10.1145/3310986.3310992]

[60] L. Breiman, "Random Forests", *Mach. Learn.,* vol. 45, no. 1, pp. 5-32, 2001.
[http://dx.doi.org/10.1023/A:1010933404324]

[61] Y. Ao, H. Li, L. Zhu, S. Ali, and Z. Yang, "The linear random forest algorithm and its advantages in machine learning assisted logging regression modeling", *J. Petrol. Sci. Eng.,* vol. 174, pp. 776-789, 2019.
[http://dx.doi.org/10.1016/j.petrol.2018.11.067]

[62] S. Liu, J. Xiao, J. Liu, X. Wang, J. Wu, and J. Zhu, "Visual Diagnosis of Tree Boosting Methods", *IEEE Trans. Vis. Comput. Graph.,* vol. 24, no. 1, pp. 163-173, 2018.
[http://dx.doi.org/10.1109/TVCG.2017.2744378] [PMID: 28866545]

[63] W. Cohen, *Learning to classify English text with ILP methods, 32nd ed. Advances in inductive logic programming*, 1995, pp. 124-143.

[64] E. Frank, and I. Witten, *"Generating accurate rule sets without global optimization", Dept. Comp. science, Univ.* Waikato, 1998.

[65] Y. Freund, and R. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting", *J. Comput. Syst. Sci.,* vol. 55, no. 1, pp. 119-139, 1997.
[http://dx.doi.org/10.1006/jcss.1997.1504]

[66] Q. Zhang, J. Wang, A. Lu, S. Wang, and J. Ma, "An improved SMO algorithm for financial credit risk assessment – Evidence from China's banking", *Neurocomputing,* vol. 272, pp. 314-325, 2018.
[http://dx.doi.org/10.1016/j.neucom.2017.07.002]

[67] T. Le, L. Hoang Son, M. Vo, M. Lee, and S. Baik, "A Cluster-Based Boosting Algorithm for Bankruptcy Prediction in a Highly Imbalanced Dataset", *Symmetry (Basel),* vol. 10, no. 7, p. 250, 2018.
[http://dx.doi.org/10.3390/sym10070250]

[68] J. Uthayakumar, N. Metawa, K. Shankar, and S. Lakshmanaprabu, "Intelligent hybrid model for financial crisis prediction using machine learning techniques", *Inf. Syst. E-Bus. Manag.,* vol. 18, no. 4, pp. 617-645, 2018.
[http://dx.doi.org/10.1007/s10257-018-0388-9]

CHAPTER 6

Detecting Ballot Stuff Collusion Attack in Reputation System for Mobile Agents Security

Priyanka Mishra[1,*]

[1] *Indian Institute of Information Technology Kota, MNIT Campus, Jaipur, India*

Abstract: A Mobile Agent (MA), when dispatched in a decentralized peer-to-peer (P2P) electronic community, is forced to do a transaction with unfamiliar hosts. Such unfamiliar hosts are malicious in nature and can tamper agent's code, state, and data. To solve integrity, confidentiality, availability, and authenticity threats from hosts, this paper proposes a soft security approach. Under this approach, a trust-based reputation model called MRep is proposed. The model considers first-hand information called Direct Reputation (DR) obtained from trust gathered through Source Host (SH). The model assumes SH to be a pre-trusted host that possesses past transaction experience from the destination host. The destination host (DH) is the target host with which the agent wishes to do a transaction in the future. Indirect Reputation (IDR) is obtained from recommenders having a past transactional history with the DH. A collusion attack takes place when these recommenders collaborate to give false recommendations about DH. Ballot Stuff and Bad Mouth collusion occur when recommenders collude to give a positive and negative rating to dishonest and honest DH, respectively. The methodology is based on Similarity Filtering (SF) that uses Euclidean Distance (ED) and single linkage clustering techniques. ED is calculated between consecutive recommender's past recommendation value called 'F-Score' and recommendation value given by SH for DH. Clustering merges recommenders into two clusters. Scatter plots give two clusters. One cluster contains recommenders that gave an exceptionally high or low rating to DH while the other cluster gave a rating close to the rating given by SH. Bernoulli's trial helps to know the effect of collusion on the Final Reputation (FR) of DH when the number of colluders increases and decreases in the system. The reputation errors are calculated and statistically verified using Binomial Probability Distribution. Validation graphs show that when the chance of collusion (p) is less than 0.5, the probability of reputation error p(x) decreases with an increase in the number of colluders(x). When p is equal to 0.5, p(x) first increases and then decreases with an increase in x and when p is greater than 0.5, p(x) increases with an increase in the number of colluders(x). We compare SF with Bayesian Filtering (BF), Outlier Filtering (OF), and No Filtering (NF) when 20%, 40%, 60%, and 80% collusion arises in the system. The proposed SF approach helps filter ballot stuff colluders. MRep gives less error in FR of DH, even when 80% collusion arises in the system.

[*] **Corresponding author Priyanka Mishra:** Indian Institute of Information Technology Kota, MNIT Campus, Jaipur, India; Tel: +919460929966; E-mail: priyanka.cse@iiitkota.ac.in

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

Keywords: Binomial probability distribution, Ballot stuff, Bad mouth, Collusion, Direct reputation, Euclidean Distance, Hosts, Indirect reputation, Mobile agents, P2P, Single-linkage clustering.

6.1. INTRODUCTION

P2P networks consist of some important properties that attracted many researchers to talk about trust and reputation issues. These issues help agents to find the trustworthiness of peers (hosts) before the actual transaction. Since each peer plays the role of both server and client, P2P networks do not require any central control [1]. Peers are also called hosts, where the actual execution of the agent takes place. Reputation-based systems collect facts about the properties of hosts, analyze them, aggregate facts about them, and finally disseminate these aggregated facts to all other hosts in the network [2]. Our earlier paper discussed how these systems could be used in mobile agent technology to secure mobile agents [3]. To ensure security, trust, and reputation-based systems are built where mobile agent finds the trustworthiness of the host's behavior before performing an actual transaction with them. In the earlier paper, the MRep model is proposed to determine the trustworthiness of the host with which mobile agent wants to pursue transaction [4]. Since all hosts are autonomous in nature, they can check each other's trustworthiness using social mechanisms of reputation. A reputation-based system collects facts about the properties of the host. It analyzes and aggregates the facts about it and finally disseminates these aggregated facts to all other hosts in the network [5]. In an e-commerce scenario, an agent is dispatched by the user in the network for selling or buying (transactional purpose) products. For this purpose, it performs possible negotiations with sellers (hosts). Since an agent has no past behavioral information, it carries a lot of suspicion and mistrust on hosts. An entity that trusts are called a trustor and an entity on which the trustor trusts is called the trustee. Our proposed MRep model consists of a trustor (agent) that collects feedback from recommending hosts (recommenders) to derive the reputation of the DH. The model considers recommendations from only those hosts that possess past transaction history with the DH. Recommendation reputation (rating) helps the mobile agent to predict the trustworthiness of the DH. Unfortunately, the recommendations given by recommenders or raters collaborate to give false ratings to make DH's reputation low or high in the market. These recommenders collude to act as if they are one single unit. Such recommenders are called colluders. These colluders keep the information about each host's transaction and recommendation history. Ballot stuff collusion occurs when recommenders collude to give a fake positive recommendation for non-reputable DH while bad-mouth collusion occurs when recommenders give fake negative recommendations for reputable DH [6]. Bad-mouth colluders not only give

negative ratings for the honest (reputable)host but also give fake positive ratings to each other. This results in severe damage to the system. Colluders damage the recommender's recommendation reputation of honest recommending hosts by spreading negative opinions about those hosts with whom they had not even performed transactions. Through such collusion activity, hosts create a conflict in DH transactional behaviors and its recommendation reputation values. Colluders prevent their recommendation reputation by sending honest recommendations for those hosts who agree to collude with them. Section two discusses the proposed reputation model in brief. Section three classifies related works and limitations. Section four explains the proposed similarity-based filtering (SF) methodology to detect colluders. Section five discusses experimental results of SF in ballot stuff and bad-mouth collusion. Section six envisages statistical simulation to show how FR increases and decreases with a probability of reputation error. Section seven shows error values in FR of DH. Section eight highlights comparative analysis and discussions. Section nine concludes the paper with future scopes.

6.2. TRUST BASED REPUTATION SYSTEM

Mobile agents are migrating programs that consist of code, data, and state [7]. In an e-commerce scenario, an agent on behalf of the user is left in the network for selling or buying products. Agent as a buyer carries no idea about host transaction behavior and performs possible negotiations with seller host because sellers are executing host platforms. Agent gets trapped by the execution environment of hosts because it does not carry the host's past behavioral information [8, 9]. An agent has to manage the risk while transacting with the host without its prior experience and true knowledge of the host's reputation. Hence, the need arises to develop strategies to establish trust for ensuring security by assessing the destination host's direct and indirect reputation [4]. The entity that trusts are called the trustor and an entity on which the trustor trusts is called the trustee. The model assumes the mobile agent as trustor and DH as trustee. Reputation systems signify soft security related to trust & reputation of interacting entities while hard security is related to authorizations and certifications of entities [10]. These systems consist of the trustor (agent) that collects feedback from the recommending hosts about the trustee. The feedback score given by recommending hosts is called recommenders. Recommenders give ratings for DH depending on their past transaction and recommendation history. Unfortunately, the feedback score (recommendation)given by recommenders or raters collaborate to give unfair ratings to declare DH's reputation low in the market. These recommenders collude to act as if they are one single unit. Such fake united recommenders are called colluders. Colluders keep information about each host's transaction and recommendation history. The paper aims to identify colluders,

determine the increase and decrease in the probability of reputation error with a varying number of colluders and estimate the error in the final reputation value of DH under ballot stuff and bad mouth attacks. Later, similarity filtering (SF) is compared with other filterings under varying percentages of collusion. MRep calculates FR of DH.MRep calculates the dynamic reputation of DH before pursuing the transaction. DR of DH is obtained from past transaction experience of SH and indirect reputation(IDR) is obtained from recommenders divided into different recommender groups according to their distance from SH. Our earlier papers discuss in detail the methodology to calculateFR via direct and indirect reputation [11]

6.3. RELATED WORKS

Reputation systems that lack mechanisms to provide data authentication and integrity are vulnerable to collusion attacks because we are not able to differentiate between fabricated and legitimate feedback. Several researchers have addressed this problem. Liu *et al* [12] reported that the FR is always misled by inserting unfair ratings in regular ratings [13]. propose a scheme to detect collaborative unfair raters (recommending hosts) based on the similarity in their rating behavior schemes. These schemes identify time intervals in which unfair ratings are highly suspected. Selcuk *et al* [14] suggest keeping records of recommender's trustworthiness which is later updated only in case transactions with such host becomes successful. Rahman and Hailes [15] propose to calculate the semantic distance between the recommender's recommendation and the trustor's (agent's) own perception towards the recommender for the outcome of its transaction. Huynh *et al* [16] suggest calculating credibility ratings of each recommender's recommendation after a transaction is performed between agent and host. These credibility ratings assign weight to recommenders before obtaining future recommendations from them. Dillon *et al* [17] estimated the recommender's reputation strength by finding the difference between its given recommendation and the trustor's evaluation of the transaction with the trustee (host). Dellarocas [18] calculated feedback reputation based on the number of consistent ratings that the trustor gave for the trustee. Allahbakhsh *et al* [19] propose a model that uses a mining technique to detect the collision groups and sub-groups. Ashri *et al* [20] advise incorporating more detailed relationships between hosts and proposed some identification mechanisms for characterization of these relationships. Such relationships change dynamically in e-marketplaces.

6.4. MREP MODEL

A reputation system for securing mobile agents helps the agent to compute the

reputation of a destination host. The destination host (DH) is the host with which the agent wishes to pursue a transaction in the future. In MRep, the agent combines direct trust (DT) obtained from the pre-trusted host (assumed) called source host. The fixed hierarchical network of MRep is shown in Fig. (**6.1**).

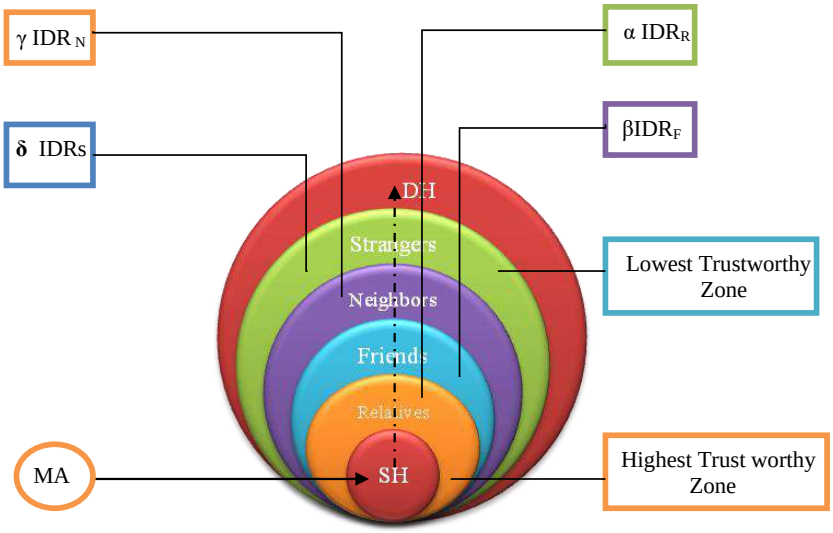

Fig. (6.1). MRep hierarchical view.

The source host(SH) is the host with which the agent had already performed the transaction and declared it pre-trusted. Indirect trust(IT) is obtained from directly or indirectly connected hosts with SH. These hosts are responsible for giving indirect trust (IDT) and are called recommenders. The model divides recommenders into four recommender communities or groups depending on their distance from SH. These four recommender groups are Relatives (R): hosts possessing past transactional history with DH and are at one hop distance from SH. Friends(F): Host possessing past transactional history with DH and is at a two-hop distance from SH. Neighbors(N): hosts possessing past transactional history with DH and is at three-hop distance from SH. Strangers (S): Host possessing past transactional history with DH and is at four or above hop distance from SH. The fixed hierarchical network of MRep is shown in Fig. (**6.1**).

MRep assists MA to do the prior computation of DH reputation. DR score (DRscore) is used to determine the transactional behavior of the host on each date. Transaction Score (Tscore) is the ratio of the number of successful to the total number of transactions given by Equation 6.1. Transaction Evaluation (TE) is the product of T-score on each date with recency in the time given by Equation 6.2. TE is weighted by a time decay function represented as e^{-dt}, where dt is the

difference between the current date and previous date of transaction by DH. Time decay function (e^{-dt}) establishes the principle that states the more recent the transaction higher the value of DRscore. The formula to calculate DRscore is given in equation 6. Agent calculates confidence by estimating divergence (Div) in total TE and DRscore values. Div is given by equation 6.4. Confidence achieved by the agent is given in equation 6.5. Finally, the DR of DH is the product of confidence with the total DRscore value given by equation 6.6.

$$Tscore = s/n \quad \textbf{(6.1)}$$

$$TE = e^{-dt} * Tscore \quad \textbf{(6.2)}$$

$$DRscore = TE/ \sum e^{-dt} \quad \textbf{(6.3)}$$

$$Div = | \sum TE - \sum DRscore | \quad \textbf{(6.4)}$$

$$Conf = k_1 * k_2 * (1-Div) \quad \textbf{(6.5)}$$

$k_1 = 1$ if $(t_c - t_0) * \psi < N$, else $k_1 = N/((t_c - t_0) * \psi)$

$k_2 = 1$ if $(t_c - t_0) * \varphi < S$, else $k_2 = S/((t_c - t_0) * \varphi)$

t_c: current date of transaction with DH

$t_{p:}$ previous dates of transaction with DH.

N: Total number of transaction, S: Total number of successful transaction

t_0: date of first transaction

ψ: Threshold for number of transactions per day

φ: Threshold for successful transactions per day

Thresholds ψ,φ can be set as per the requirements of the user.

$$DR = Conf * \sum DRscore \quad \textbf{(6.6)}$$

Fig. (**6.2**) shows the logical community-level view of MRep where α, β, γ and δ signifies weight to R, F, N and S groups respectively.

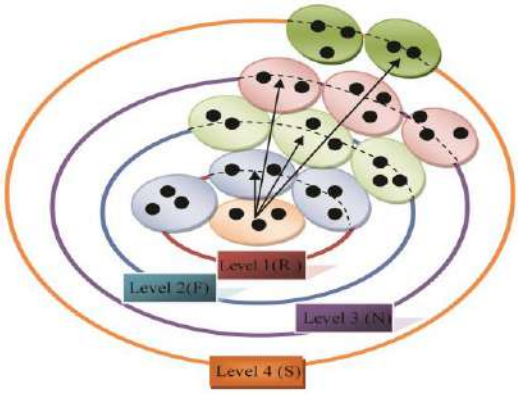

Fig. (6.2). Community level view of MRep.

Indirect Reputation (IDR): Indirect reputation is the product of the normalized recommendation reputation value of recommender with recommendation value for DH. IDR of particular recommender X is given by equation 6.7.

$$IDRx = \sum_{i=1}^{n} \left(\frac{WX_i}{\sum WX} \right) * Rec_{(X_i - DH)} \quad \textbf{(6.7)}$$

WX_i:Weight of recommender 'i' of 'X' recommender group given by its F-Score

Where xε{ R, F, N, S} i = Recommender i of group X

$Rec_{(Xi\text{-}DH)}$: recommendation value given by recommender i of group X.

Total indirect reputation (TIDR) is the weighted sum of reputations obtained from four groups of recommenders who had past transactional history with DH. TIDR is given by equation 6.8.

$$TIDR = \alpha\, IDR_R + \beta\, IDR_F + \gamma\, IDR_N + \delta\, IDR_S \quad \textbf{(6.8)}$$

where $\alpha > \beta > \gamma > \delta$ such that $\alpha+\beta+\gamma+\delta=1$.

α, β, γ,δ: weight parameters to Relative, Friend, Neighbour and Stranger group respectively

α: weight of Relative recommender group

β: weight of Friend recommender group

γ: weight of Neighbor recommender group

δ: weight of Stranger recommender group

The weights α, β, γ and δ are taken such that their sum is unity and α> β > γ > δ*.i.e*. α+β+ γ + δ =1

Mobile agent calculates FR of DH by weighted aggregation of DR and TIDR.A significant factor 'θ' is assumed to assign weight to DR and TIDR. Agent assigns weight 'θ' to DR because it is obtained from pre trusted host *i.e*. SH and (1- θ) to TIDR obtained from indirect reputations given by recommenders belonging to four recommender groups. Value of FR is given by equation 6.9.

$$FR = \theta * DR + (1-\theta)* TIDR \quad \textbf{(6.9)}$$

MA decides the trust level of DH depending on the application. If the value of FR is above or equal to the trust level, MA assigns to show DH's trustworthiness and '0' to show DH's untrustworthiness.

The Presence of a collusion attack in MRep is shown in Fig. (**6.3**). SH sends a request of transaction Req (T) and receives the response of transaction Res (T) from DH. Since SH is an assumed pre-trust host, hence it sends the direct reputation (DR) of DH to the agent. The agent receives indirect reputation (IDR) from recommenders who sent Req (T) to DH and obtained Res(T) from it. Weights to recommendations received from recommender communities, classes, or groups are assigned depending on their distance from SH. Total indirect reputation (TIDR) is obtained by weighted summation of these indirect recommendations. The agent calculates the final reputation (FR) by weighted summation of DR and TIDR as discussed previously. To demonstrate the ballot stuff collusion attack, we assume that the DH has a low reputation and some recommenders of different communities or groups unite to increase the reputation of DH which leads to ballot stuff collusion. These colluders give an exceptionally high rating to increase the reputation of DH. As a result, TIDR increases with the increase in the number of colluders named New TIDR. New_ FR is calculated by the weighted sum of DR and New_TIDR. Ballot stuff collusion is detected when New_FR> FR and bad mouth collusion occurs when New_FR< FR. No collusion is detected when New_FR=FR.

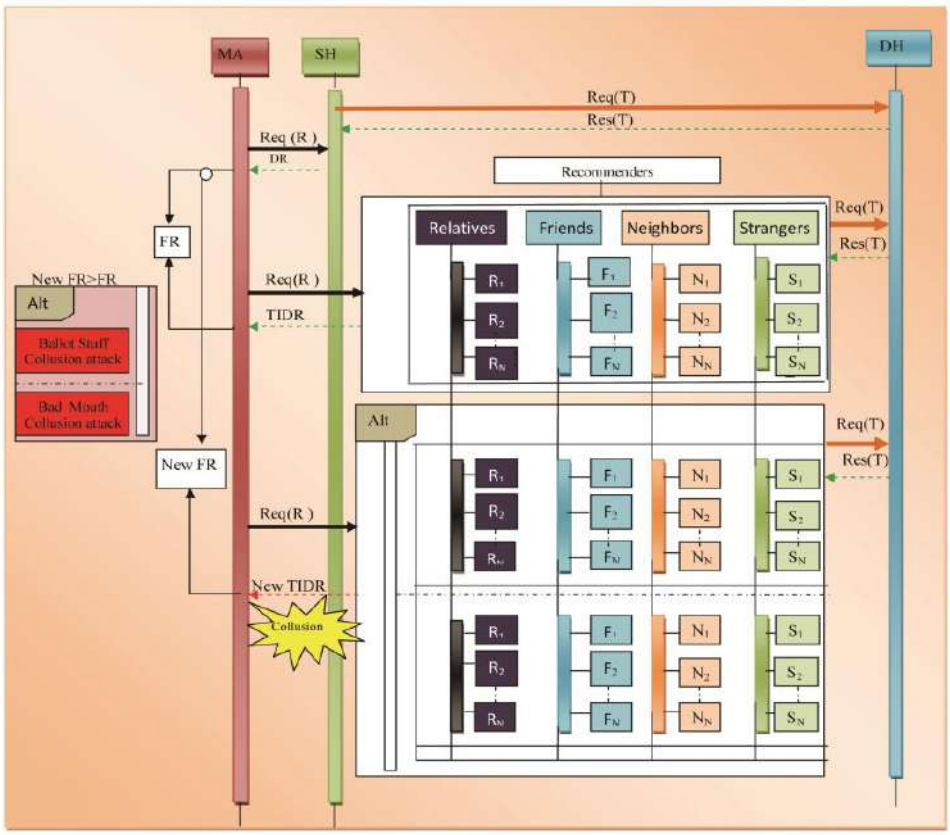

Fig. (6.3). Sequence diagram to show collusion attack in Mrep.

The flowchart of the complete methodology to detect collusion attack is shown in Fig. (**6.4**). After clustering TIDR and New_ FR is calculated. FR is compared with New_FR. Ballot stuff collusion occurs when New_FR> FR and bad mouth collusion occurs when New_FR< FR. Attack not occur when New_FR = FR.

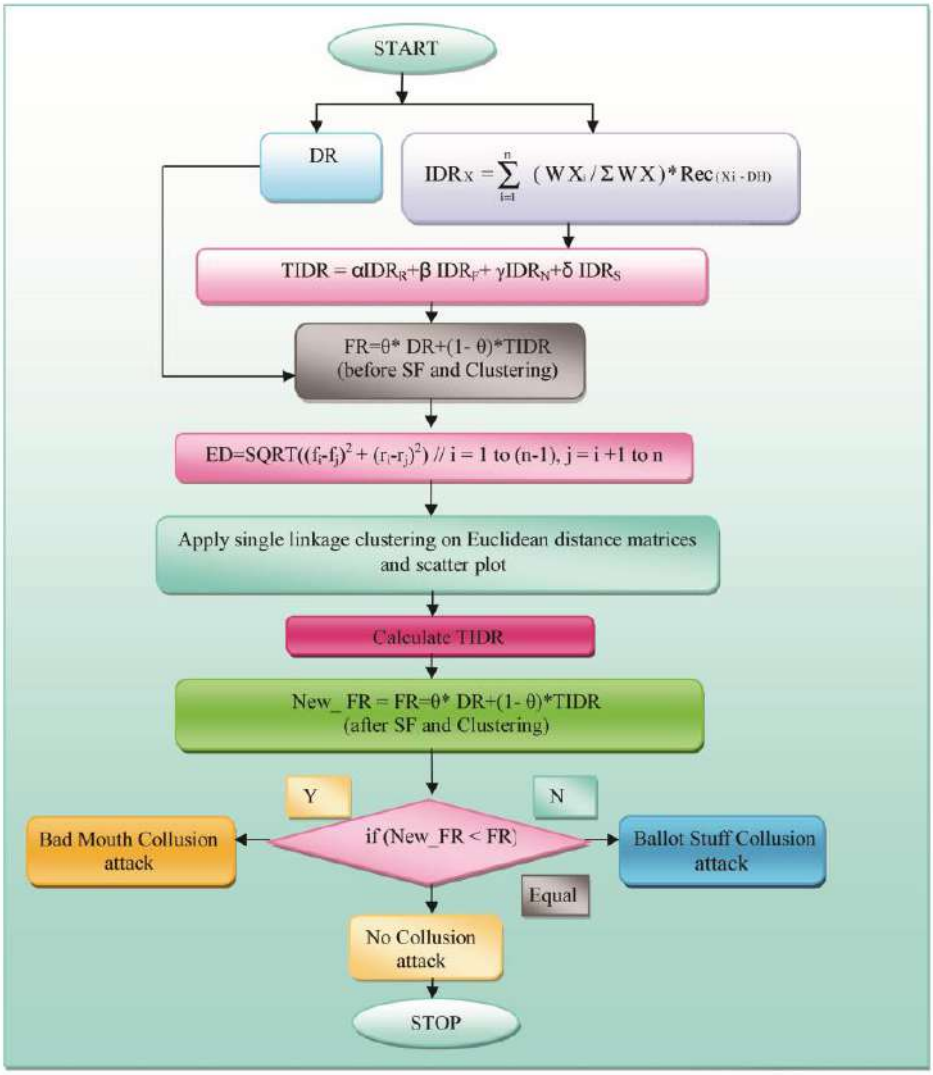

Fig. (6.4). Methodological flowchart for collusion detection.

6.5. DETECTION METHODOLOGY

In this section, we have proposed a new detection methodology and suggested a prevention mechanism for collusion attack. The proposed approach is based on Euclidean distance (ED) based on similarity and clustering. Euclidean distance (or straight-line distance) is the most commonly used type when it comes to analyzing ratio or interval-scaled data.

Methodology to detect colluders uses Euclidean Distance (ED) based similarity and single linkage clustering. ED is calculated between recommender's F-Score and recommendation values denoted by d_{ij} in equation (6.10). F-Score of recommender 'i' and 'j' is given by f_i and f_j respectively. Also, r_i and r_j represent the recommendation of recommender 'i' and 'j' respectively.

$$d_{ij} = \sqrt{(f_i - f_j)^2 + (r_i - r_j)^2}$$

$$d_{11} = \sqrt{(f_1 - f_1)^2 + (r_1 - r_1)^2} \quad \textbf{(6.10)}$$

Here i= 1 to n-1, j=i+1 to n, where n= number of recommenders.

Where, $d_{11} = d_{22} = d_{33}$ -------- $d_{nn} = 0$

$$d_{12} = \sqrt{(f_1 - f_2)^2 + (r_1 - r_2)^2}$$

$$d_{21} = \sqrt{(f_2 - f_1)^2 + (r_2 - r_1)^2}$$

Similarly, we can calculate other Euclidean distances (Table **1**).

Table 6.1. Euclidean Distance Matrix.

	R1	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8**	**R9**	**R10**	**R11**	**Rn**
R1	d_{11}											
R2	d_{21}	d_{22}										
R3	d_{31}	d_{32}	d_{33}									
R4	-	-	-	-								
R5	-	-	-	-	-							
R6	-	-	-	-	-	-						
R7	-	-	-	-	-	-	-					
R8	-	-	-	-	-	-	-	-				
R9	-	-	-	-	-	-	-	-	-			
R10	-	-	-	-	-	-	-	-	-	-		
R11	-	-	-	-	-	-	-	-	-	-	-	
Rn	-	-	-	-	-	-	-	-	-	-	-	d_{nn}

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract.

Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

Bernoulli trails help to find whether colluders will achieve success or failure in colluding the system. The binomial distribution is used to determine the probability of reputation error given by equation 6.11.

$$p(x) = C(n,x)\, p^{x}(1-p)^{n-x} \quad \textbf{(6.11)}$$

In the above equation, n = total number of hosts, x = number of host colludes, p = chance of hosts to collude, p(x)= probability of reputation error due to collusion when 'x' number of hosts out of 'n' colludes. The paper shows three cases when p < 0.5, p = 0.5 and p > 0.5 in Tables **6.2**, **6.3** and **6.4** respectively.

Table 6.2. p(x) Decreases with Inceases in x.

n	x	p	p(x)
12	4	0.1	0.0213081
12	5	0.1	0.0037881
12	6	0.1	0.0004911
12	7	0.1	0.0000468
12	8	0.1	0.0000032
12	9	0.1	0.0000002

Table 6.3. p(x) first Increases then decreases with inceases in x.

n	x	p	p(x)
12	4	0.5	0.12084961
12	5	0.5	0.19335938
12	6	0.5	0.22558594
12	7	0.5	0.19335938
12	8	0.5	0.12084961
12	9	0.5	0.05371094

Table 6.4. p(x) Inceases with Inceases in x.

n	x	p	p(x)
12	4	0.9	0.00000325
12	5	0.9	0.00004680
12	6	0.9	0.00049110

(Table 6.4) cont.....

n	x	p	p(x)
12	7	0.9	0.00378810
12	8	0.9	0.02130810
12	9	0.9	0.08523250

Case 1: p < 0.5

The probability of reputation error p(x) decreases with an increase in the number of colluders. Fig. (**6.5**) shows how MRep exhibits positive skew.

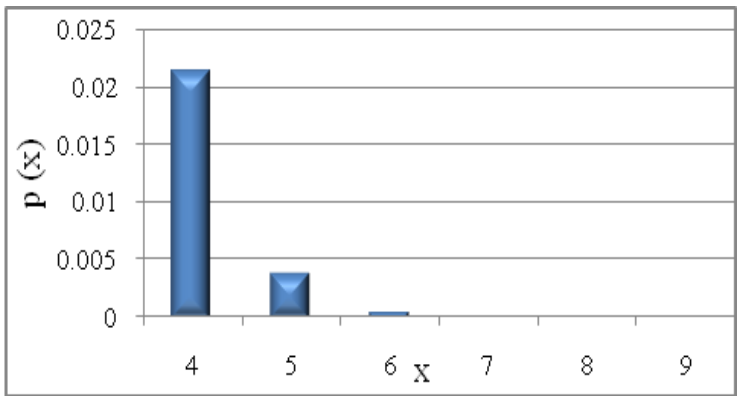

Fig. (6.5). Positive skew in p(x).

Case 2: p = 0.5

As evident in Table **6.3** and Fig. (**6.6**), when p = 0.5, p(x) first increases and then decreases with an increase in the number of colluders. MRep exhibits a symmetric skew.

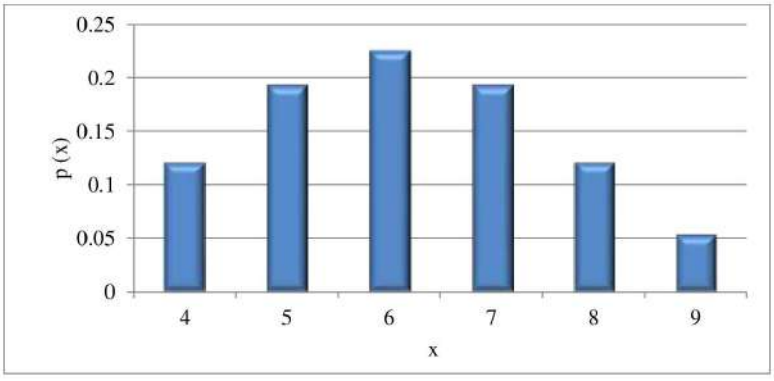

Fig. (6.6). Symmetric curves in p(x).

Case 3: p > 0.5

In Table **6.4** and Fig. (**6.7**), when value of p > 0.5, p (x) increases with an increase in the number of colluders and exhibits a negative skew. FR of DH under ballot stuff collusion (FR_{bs}) is calculated. MRep shows that the value of FR increases with the increase in the number of colluders.

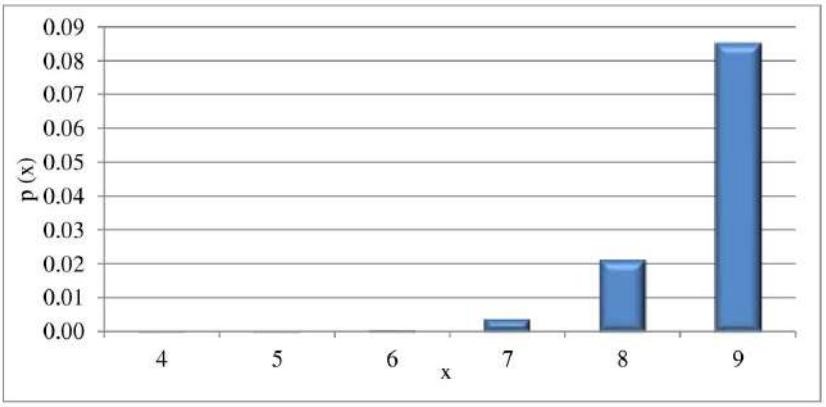

Fig. (6.7). Negative skew in p(x).

The percentage of error in reputation calculated as(E_{bs} = (FR_{bs}-FR) *100 / FR) is shown in Table **6.5**.

Table 6.5. Inceasing Order of Reputation Errors.

n	x	p	p(x)	FR	FR_{bs}	E_{bs}
10	2	0.9	0.00000036	0.35124	0.37282658	6.14582002
10	4	0.9	0.00013778	0.35124	0.38515889	9.65689726
10	6	0.9	0.01116026	0.35124	0.39481406	12.4057791
10	8	0.9	0.19371024	0.35124	0.40543475	15.429549

The percentage of reputation error(E_{bs}) increases with an increase in the number of colluders(x).

Fig. (**6.8**) shows that MRep exihibits negative skew. The results obtained are in synchronization with statistical experiment. Thus, MRep validates proposed methodology and shows error in FR of DH. Collusion attack is detected because value of FR increases with increase in the number of colluders.

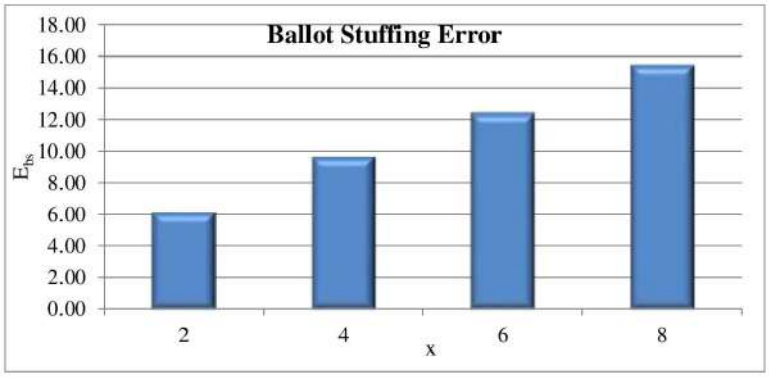

Fig. (6.8). Negative skew in E_{bs}.

6.6. Simulation Results

SF is compared with Outlier Filtering (OF) [21], Bayesian Filtering (BF) [22] and without Filtering (WF) approaches. We took 1024 hosts and the simulation runs until each host performs 20 transactions in average. The length of the simulation is indicated by the average number of transactions. The SH and DH is generated randomly from [1, N]. Probability in reputation error (E_{bs}) is calculated when 20%, 40%, 60% and 80% hosts collude under four filtering. All filtering algorithms give reputation error less than 10% when the percentage of collusion is less than 20%, except WF. When collusion percentage increase to 40% and beyond the reputation error in BF, OF and WF becomes high except SF. Bayesian probability finds the posterior (*i.e*. the updated) reputation score by combining the prior (*i.e*. previous) reputation score with the new rating. With the increase in the number of dishonest recommenders, BF breaks down and detects many dishonest hosts as honest and honest hosts as dishonest. This gives high reputation errors. In OF, the recommendations from dishonest hosts are outliers if the majority of hosts are honest. In this method, E_{bs} is less when honest hosts are more and E_{bs} is high when dishonest hosts are high. SF gives very less reputation error compared to all three methods because SF considers error in reputation when recommendation given by hosts has low F-Score and low recommendation value as shown in Fig. (**6.9**). Under such conditions, SF gives unfair low recommendation value or unfair high recommendation value. So, SF performs well even when the system is dominated by dishonest hosts.

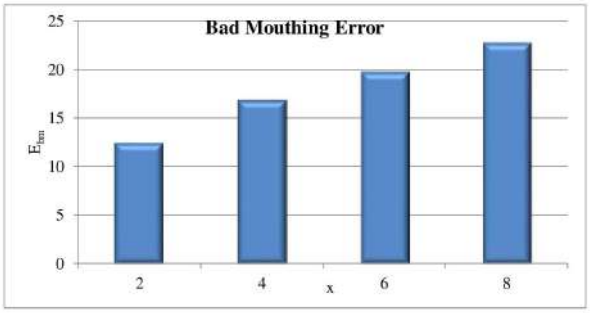

Fig. (6.9). Reputation error due to Ballot stuffing collusion.

CONCLUSION

The Paper discusses the problem of collusion attack. Proposed technique is based on Similarity Filtering (SF) that uses Euclidian Distance (ED) matrices and single linkage clustering. SF shows a great promise to detect collusion attack. Statistical treatment for this problem is given considering the probabilistic collusion of the hosts. Reputation errors are calculated and statistically verified to show how collusion attack results in the increase of reputation errors under ballot stuff collusion attack. SF is compared with BF, OF and WF. SF performs significantly well by giving low reputation errors even when a large number of dishonest hosts rises in the system. WF gives 22.6 reputation errors under 20% collusion. The value of error increases and reaches 90.1 when 80% collusion occurs.BF gives reputation error equal to 7.1 under 20% collusion and reaches 90 under 80% collusion. OF gives reputation error 5.02 and 91.6 when 20% and 80% collusion occurs respectively. SF gives a reputation error of 6.14 under 20% collusion and reaches to only 15.42 under 80% collusion.The above data confirms that SF outperforms well compare to all filtering.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The author declares no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] Chen X, Chen G., and Liu G., Trust Factors in P2P Networks," IEEE Digital Library. 2008.

[http://dx.doi.org/10.1145/544741.544853]

[2] J.M. Pujol, R. Sangüesa, and J. Delgado, "Extracting Reputation in Multi Agent Systems by means of social network typology", *AAMAS,* vol. 02, pp. 467-474, 2002. [http://dx.doi.org/10.1145/544741.544853]

[3] P. Dadhich, K. Dutta, and M.C. Govil, "Reputation–based Trust for mobile agents ", in proceedings of International Conference on Advanced Computing, Networking and Security (ADCONS), LNCS vol no:7135, pp. 379-384, 2011.

[4] Dadhich P., Dutta k., and Govil M. C., "Detection of Slanders through Euclidean Distance Similarity Assessment for Securing E-commerce Agents in P2P Decentralized Electronic Communities", *International Journal of Security and Networks (in Press), Inderscience*.

[5] K. Hoffman, and D. Zage, "A Survey of attacks and defense techniques for reputation systems", *ACM Computer Survey,* vol. 42, no. 1, p. 1:1-1:31, 2009.

[6] Y. Jin, Z. Gu, and Z. Ban, "Restrainning false feedbacks in peer-to-peer reputation systems", *International Conference on semantic computing,* 2007pp. 304-312 [http://dx.doi.org/10.1109/ICSC.2007.16]

[7] M.A. Shibli, R. Masood, Y. Ghazi, and S. Muftic, *MagicNET: mobile agents data protection system,* 2015.http://onlinelibrary.wiley.com/doi/10.1002/ett.v26.5/issuetoc [http://dx.doi.org/10.1002/ett.2742]

[8] Dadhich P., Dutta K, and Govil M. C., "Security Issues in Mobile Agents", *International Journal of Computer Applications (0975 –8887),* vol. 11, no. 4, 2010. [http://dx.doi.org/10.5120/1574-2104]

[9] P. Dadhich, K. Dutta, and M.C. Govil, Security measures to protect Mobile Agents," in proceedings of International Conference on Methods and Models in Science and Technology (ICM2ST-10) , American Institute of Physics , Springer vol no. 1324, pp:298, 2010.

[10] C. Lin, and V. Varadharajan, MobileTrust: a trust enhanced security architecture for mobile agent systems, " in International Journal of Information Security,9(3) pp: 153-178, 2010.

[11] P. Dadhich, K. Dutta, and M.C. Govil, On the approach of combining trust and security for securing mobile agents: Trust enhanced security, " in proceedings of 2nd IEEE International Conference on Computer and Communication Technology (ICCCT), pp: 379-384, 2011.

[12] Y. Liu, Y. Sun, S. Kay, and Q. Yang, "Defending Online Reputation Systems against Collaborative unfair raters through signal modeling and trust", *Proceedings of the 24th ACM Symposium on Applied Computing (SAC'09),* pp. 1308-1315, 2009.

[13] Y. Liu, and Y. Sun, "Security of Online Reputation Systems: Evolution of Attacks and Defenses," Signal Processing Magazine, Special issue on Signal and Information processing for Social Learning and Networking", *IEEE Signal Process. Mag.,* pp. 87-97, 2012.

[14] A.A. Selcuk, E. Uzun, and M.R. Pariente, "A reputation-based trust Management system for P2P networks," in: Proceedings of 4th International Workshop on Global and Peer-to-Peer Computing, Chicago, (GP2PC'04), USA, pp 227-237, 2004.

[15] A. Rahman, and S. Hailes, "Using Recommendations for managing Trust in Distributed System,", *Proceedings of the IEEE Malaysia International Conference on Communications '97, Kualalmpur, (MICC'97), Malaysia,* 1997.

[16] T.D. Huynh, N.R. Jennings, and N. Shadbolt, "On handling inaccurate witness Reports,", *Proceedings of 8th International Workshop on Trust in Agent Societies,* pp. 63-77, 2005.Utrecht, Netherlands.

[17] T.S. Dillon, E. Chang, and F.K. Hussain, "Managing the dynamic nature of trust", *IEEE Journal of Intelligent System,* vol. 19, no. 5, pp. 79-82, 2004.

[18] C. Dellarocas, "Immunizing Online Reputation System against Unfair Ratings and Discriminatory Behavior,", *Proceedings of 2nd ACM Conference on Electronic Commerce (EC'00),* 2000pp. 150-157

New York, USA
[http://dx.doi.org/10.1145/352871.352889]

[19] M. Allahbakhsh, A. Ignjatovic, B. Benatallah, S. Beheshti, E. Bertino, and N. Foo, *Collusion Detection in Online Rating System,,* 2013.http://link.springer.com/book/10.1007/978-3-642-37-01-2http://link.springer.com/bookseries/558https://link.springer.com/chapter/ 10.1007 / [http://dx.doi.org/10.1007/978-3-642-37401-2_21]

[20] R. Ashri, S.D. Ramchurn, J. Sabater, M. Luck, and N.R. Jennings, "Trust evaluation through relationship analysis,", [http://dx.doi.org/10.1145/1082473.1082625]

[21] F. Azzedin, and A. Ridha, "‘ Feedback behavior and its role in trust assessment for peer-to-peer System, ‘", *Telecomm. Syst.,* vol. 44, no. 3–4, pp. 253-266, 2010. [http://dx.doi.org/10.1007/s11235-009-9263-9]

[22] A. Whitby, A. Josang, and J. Indulska, "Filtering out unfair ratings in Bayesian reputation System,", *Proceedings of 7th International Workshop on Trust in Agent Societies Rome,(AAMAS '04), Itlay,* pp. 106-117, 2004.

CHAPTER 7

Crow Search Algorithm: A Systematic Review

Ali Aloss[1], Barnali Sahu[1,*] and **Om Prakash Jena[2]**

[1] *Department of Computer Science and Engineering, Siksha'O'Anusandhan (Deemed to be university), Bhubaneswar, Odisha, India*

[2] *Department of Computer Science, Ravenshaw University, Cattuck, Odisha, India*

Abstract: Cognitive computing and Artificial Intelligence (AI) are Computer Science branches which aim to create machines and ingenious technologies that are capable of working and thinking like humans. Evolutionary computing is a subfield of AI that uses nature-inspired mechanisms (algorithms) and solves problems through processes that mimic the behavior of living organisms. Researchers have focused on several meta-heuristic algorithms, and the Crow Search Algorithm (CSA) is one of the recently developed algorithms dependent on the astute conduct of crows. CSA is a populace-based methodology. It works by storing excess food in hiding places and extracting the food when necessary. This algorithm has been used in different fields such as medical diagnoses, fractional optimization problems, and energy problems. Several modifications have been made to this algorithm, and the current research focuses on a systematic review of the applications of the crow search algorithm in the medical domain and the variants of CSA and its application in different engineering fields.

Keywords: Application of CSA, Crow search algorithm, Evolutionary algorithm, Medical diagnosis, Meta-heuristic algorithm, Variants of CSA.

7.1. INTRODUCTION

One of the most difficult concerns in recent decades has been diagnosing and addressing medical disorders. Cognitive Computing and Artificial intelligence techniques have revolutionized the sphere of diagnosing clinical problems and proposing treatments in addition to their primary ability to analyze over the past years. Algorithms focusing primarily on machine learning are among the first algorithms that were designed and used for analyzing medical data because they comprise many tools that are crucial in this field. One of the fundamental challenges in the health care system is to obtain an efficient utilization of

[*] **Corresponding author Barnali Sahu:** Department of Computer Science and Engineering, Siksha'O'Anusandhan (Deemed to be university), Bhubaneswar, Odisha, India; Tel: +918895278059; E-mail: sahu.barnali08@gmail.com

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

expensive assets while maintaining or providing quality care. To enhance the product-ability in health care system, optimization is applied in each field, starting from activity scheduling to prediction of diseases. In optimization, the objective is to find the best solution among many alternative solutions or the properly adequate solution to a given problem. In daily life, everyone is dealing with optimization problems, *i.e*., finding the shortest route from home to the workplace subject to traffic constraints or organizing our agenda. Most human brains are appropriate in finding solutions to these everyday problems effectively because they are still solvable due to the limited dimension. Some problems, however, appear much larger in scale than computer algorithms are designed to handle. There are no efficient algorithms for these complex problems since the majority of such techniques are typically tailored to the problem at hand, and they seek to take full advantage of the particularities of this problem. Since they are always too greedy, they are usually stuck in a local optimum, and therefore fail to achieve the desired global optimum solution. To obtain the optimal global solution, several metaheuristic algorithms are proposed [1] which were found to be very efficient for solving very complex problems. Heuristics algorithms are designed to solve a particular problem without being able to generalize or refer to other related problems. In contrast, a meta-heuristic method stands for a higher-level heuristic in the context that it guides its design [2]. The meta-heuristic algorithms are broadly classified as single solution-based algorithms and population-based algorithms. In the first case, a random result is produced and enhanced until the optimal solution is found, whereas, in the second category, solutions are generated in a given search space and -try to improve until the optimal solution is achieved. However, the second category can find the global optimum, whereas the first category fails to do so. Therefore, researchers are motivated towards population-based algorithms. Thusly, we may use one of these approaches to develop a particular procedure to gauge an approximate answer for an optimization issue. In the area of global optimization, a large number of Meta-heuristic Algorithms (MA) such as Particle Swarm Optimization [3], Ant Colony Optimization [4], Bat algorithm [5], Artificial Bee Colony Optimization [6], Crow search Algorithm (CSA) [7] had been proposed over the years to solve complex engineering problems in a reasonable amount of time by harmonizing the exploration and exploitation criteria. The classification of meta-heuristic algorithms includes evolutionary-based algorithms, physics-based algorithms, and Swarm Intelligence algorithms. Swarm Intelligence (SI) is a part of the meta-heuristic, which is related to flocks that depend on the interaction between each other by following some basic instructions [8].

One of the meta-heuristic algorithms proposed by Askar Zadeh in 2016 is the Crow search algorithm; it relies upon analyzing the smart behaviors of crows through the potential of crows to conceal their food in protected locations and

chase other crows to obtain their food. It has been applied successfully to different science and engineering fields of optimization [9]. Nonetheless, the current and focused analysis of this algorithm is lacking in the literature. Furthermore, the CSA was no longer compared to new algorithms; since then, many new algorithms have been introduced. Experimenting with CSA on a wider variety of test functions and comparing it to modern and reliable algorithms would, therefore, further expose the use of the algorithm. As a result, the objective of this paper is to first study the principles of CSA in addition to its basic structure and features and secondly,provide a comprehensive and detailed review of the state of the art CSA algorithm. Thirdly, it offers various CSA applications, especially in the medical field. Finally, we conclude the paper by summing up the progress and evaluating future patterns in the study. These work will be of great assistance to researchers in further growth and implementation work in the area.

7.2. CROW SEARCH OPTIMIZATION

7.2.1. Overview of Crow Search Optimization

Optimization has played a very important role in several areas, not limiting to engineering problems. These problems consist of complex objective functions, various decision variables, and a huge set of constraints, which add complexity to an already complicated problem of optimization. Such constraints take the search from a conventional optimization methodology to a modern area of analysis, *i.e.*, swarm intelligence pioneered in the 1980s by Beni and Wang [6]. SI mimics the collective intellect of the living creature of nature [6]. Each new algorithm focuses on two features: first, finding solutions close to the actual optimal solution, thus reducing the gap between them, and second, finding the solution in the least possible time, and thus less search time. Several optimization algorithms have been proposed over the past years, as each has its own advantages and disadvantages. The crow search algorithm is a newly developed algorithm that imitates the social intelligence of crows and their way of collecting food. It is a metaheuristic algorithm inspired by crows' intelligent behaviors. Crows are among the wisest creatures in the world as shown in the mirror test [9]. There are numerous bits of proof indicating crows' astuteness. Crows can recollect faces, trade data with one another, take the nourishment and get it far from others by concealing the assortment of food sources [9]. They are clever criminals as they take additional consideration, for example, changing concealing spots from time to time so they can abstain from being casualties later on. Crows utilize a learned way to deal with concealing their nourishment and taking others' nourishment. In the subsequent sections, the features of CSA, the algorithmic structure of CSA, along with the Pseudocode and flowchart, are prearranged for understanding and implementation of CSA in different fields of optimization.

7.2.2. Features of Crow Search Algorithm

Interestingly, a crow individual, like the other crows of the herd, may tap into other species' food resources. Every crow tries to pelt their surplus food in a hideaway spot and recover the kept food when necessary. Each crow in the flock intends to keep their food in a safe place out of reach of other crows and retrieve that food from that location when needed. Members of the flock chase after each other to discover other crows and determine where to hide their food. In the event that the crow believes that the members of the flock are chasing it to discover its location, it seeks to mislead them by moving to another place. The above are the basic principles of CSA as the crow searches in the search space for the best hideouts that contain the food resource (Maximizing the global optimum). Therefore, the crow's movement within the search space depends on two main features: the first is to locate the hideouts of other members of the herd, and the second is to try to protect the places where their food is stored. In the usual CSA, the crows flock to scatter and look for the perfect hideout spots (global optima) across the decision space.

Below are some features of the Crow search algorithm:

- In CSA, exploration and exploitation are regulated by the parameter of attentiveness probability (AP). This helps to balance exploitation and exploration by adjusting the value of this parameter.
- CSA algorithm is not greedy, which helps in the diversity of solutions established. If the crow creates a new site and this site is not better than the existing site, the crow shifts to the new site.
- Unlike other optimization algorithms, CSA has fewer parameters (FL, AP), which helps in setting parameters during implementation and adjusting them more easily and thus consuming time on the job.
- CSA relies on using the best position to find new, better positions.
- CSA uses a population of researchers to explore the research space and use it to find a good solution within this space.

7.2.3. Algorithm structure of CSO

Let us presume that a d-dimensional environment exists, and this environment has N number of crows, the position of crow i at iteration (iter) is represented by the vector $x^{i,iter} = x_1^{i,iter}, x_2^{i,iter},.....x_d^{i,iter}$. The crows have a good memory which makes them able to remember the hiding place of their food, $m^{i,th}$ represents the position of the hiding place of crow i at iteration (iter), suppose crow j needs to update its

position to a new position and let it be at iteration (iter), and on the same iteration crow i decides to track crow j to discover its hiding place, two cases can occur here, in first case crow j does not notice crow i when crow j follows crow j; as a consequence, crow i will locate the hiding place of crow j, and as per Eq(1), the new location of crow i will be changed as follows:

$$x^{i,iter+1} = x^{i,iter} + ri * fl^{i,iter} * \left(m^{j,iter} - x^{i,iter}\right) \quad \textbf{(1)}$$

ri: is a random number with a uniform distribution between 0 and 1, $fl^{i,iter}$ represents the flight length of crow i at iteration iter.

In the second case, the crow j notices that crow i is pursuing him and will discover his food's hiding place as a result, to guard his hideout from being stolen, crow j deceives crow i by changing its location to another location in the exploration area. Fig. (**1**) depicts the characteristics of the crow flight length(*fl*) on the search process and Fig. (**2**) shows the flow chart of CSA algorithm.

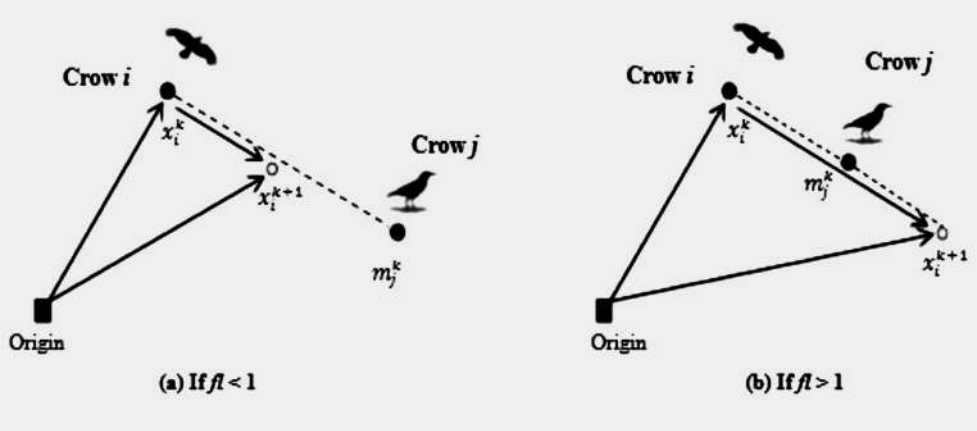

Fig. (1). The crow flight length features on the search process.

States 1 and 2 can be represented as per Eq (2) as follows:

$$x^{i,iter+1} = \begin{cases} x^{i,iter} + ri * fl^{i,iter} * \left(m^{j,iter} - x^{i,iter}\right) rj \geq AP^{j,iter} \\ arandompositionotherwise \end{cases} \quad \textbf{(2)}$$

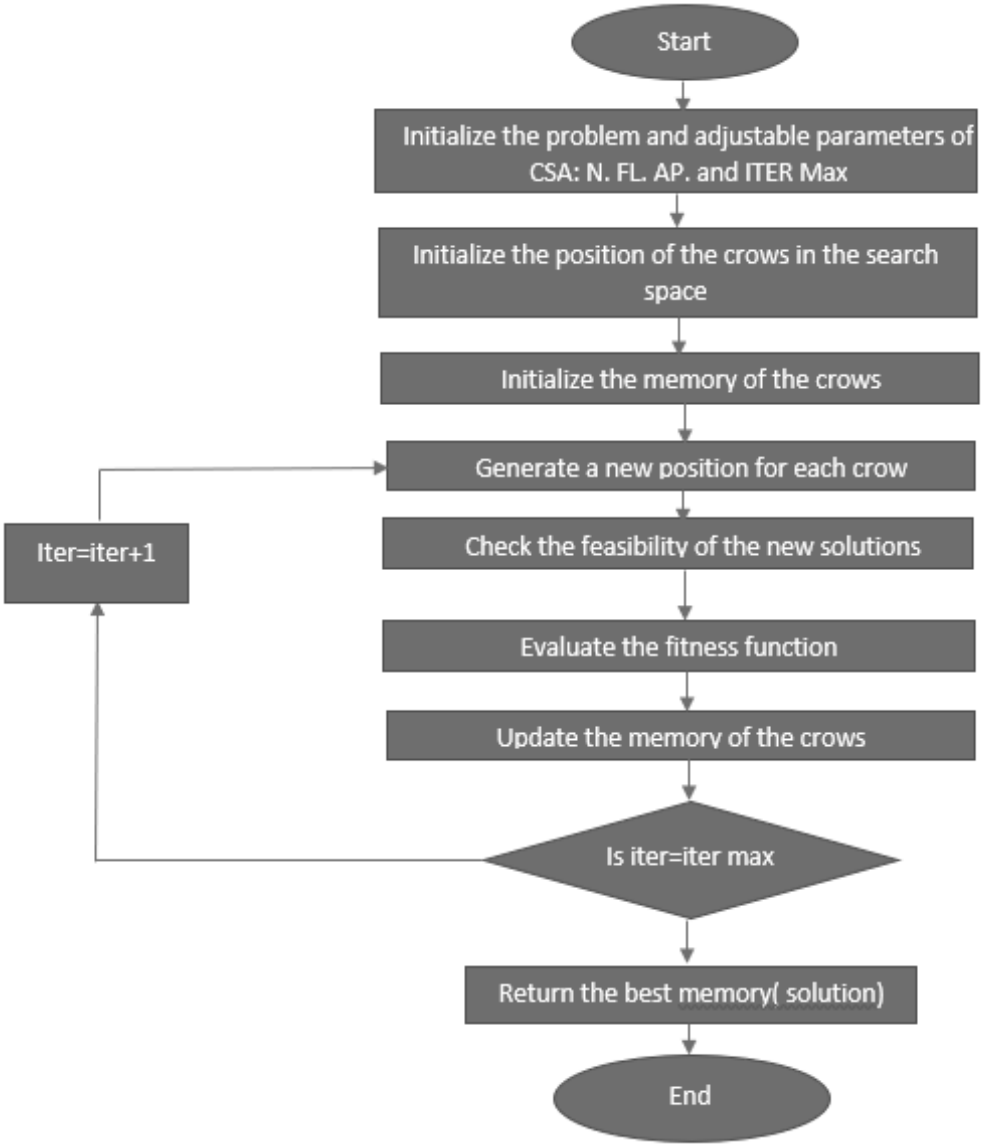

Fig. (2). Flowchart of CSA.

The CSA procedure steps are as follows:

Step 1: Initializing the problem by defining both decision variables and restrictions related to the issue, in addition to specifying the following criteria: number of population (N), Flight length (fl), Maximum number of iterations (iter max), and Awareness probabilities (AP).

Step 2: The position and memory of the crows are randomly configured.

Step 3: Calculate the fitness function for each crow.

Step 4: Generate new solution by Eq (2).

Step 5: Check the feasibility of new solution: By relying on the feasibility or infeasibility of the solution where if the solution is feasible, the crow adjusts its location and the solution is acceptable, otherwise the solution is rejected and the crow remains in its current location.

Step 6: Calculate the fitness function for each crow in a new location.

Step 7: Update memory:

Crows change their memories according to the formula below:

$$m^{i,iter+1} = \begin{cases} x^{i,iter+1} f\left(x^{i,iter+1}\right) is better than f\left(m^{i,iter}\right) \\ m^{i,iter} otherwise \end{cases} \quad \textbf{(3)}$$

Step 8: Check the end criterion by repeating steps from 4-7 until the termination criterion is met.

7.2.4. Pseudocode of CSA

Random initialization of the position of a flock of N crows in the search space

Evaluation of the position of the crows

Initialization of the memory of each crow

While iter < $iter_{max}$

  for i=1: N (all N crows of the flock)

    Random selection of one of the crows to follow (e.g. j)

    Define an awareness probability

While iter $<$ $\text{iter}_{\max}$

for i=1: N (all N crows of the flock)

Random selection of one of the crows to follow (e.g. j)

Define an awareness probability

If $r_j \geq AP^{j,iter}$

$x^{i,iter+1} = x^{i,iter} + ri * fl^{i,iter} * \left(m^{j,iter} - x^{i,iter}\right)$

else

$x^{i,iter+1} =$ a random position of search space

End if

End for

Checking the feasibility of new positions

Evaluation of the new position of the Crow

Updation of the memory of Crows

End while

7.3. CSA STUDIES

We concentrate on documenting advances on CSA in the form of structured publications in this study. They separate developments into three aspects:

a. Changes to CSA include Chaotic CSA, Fuzzy CSA, and other minor modifications.
b. Hybridization of CSA with other metaheuristic methods, including Bat Algorithm (BA), Ant Colony Optimization (ACO), Grey Wolf Optimization (GWP), Genetic Algorithm (GA), Lion Algorithm (LA), Cat Swarm Optimization (CSO).
c. CSA extensions to other fields of optimization including multi-objective, and binary optimization.
d. Other applications

7.3.1. Modifications of CSA

7.3.1.1. Chaotic CSA(CCSA)

CSA has incorporated principles related to chaos theory to boost its efficiency. Gehad [10] introduced chaotic maps with CSA. The proposed algorithm deals

with low dimensional optimization problems to overcome the low convergence rate and avoid being caught in the local optima. Deepak [11] introduced an optimization version of CCSA called OCSA applied it to improve the diagnosis of Parkinson disease. Dunia [3] proposed a novel approach of CCSA deals with high dimensional optimization problems. Khazaei [13] applied CCSA on Islanded Microgrid Operation for finding the optimal solution of this operation and also to provide a fast response. Rizk [14] applied CCSA for solving fractional optimization problems in order to refine the global convergence as well as to adjust the balance between exploitation and exploration.

7.3.1.2. Fuzzy CSA(FCSA)

Fuzzy logic techniques have helped CSA to provide good solutions to complex problems due to their flexibility and permitting modification of rules in addition to using human logic and decision making, which provides the opportunity to model the self-imagination of decision-makers precisely and thus reduce the risk of application of the wrong model and avoid results that do not reproduce the real problem. To make CSA more powerful, fuzzy set theories have been used with it to improvise the predictability, computability, and efficiency of the model. Table **1**, presents the diverse applications of FCSA in different fields.

Table 1. Application of FCSA in different fields.

Paper	Proposed method	Dataset used	Method objective	Findings
Mohamed [15]	CSA-based Neuro-Fuzzy inference framework (ANFIS-CSA)	OHTC data	Used to increase the prediction performance of the oscillatory heat transfer coefficient (OHTC) for a thermoacoustic heat exchanger.	(ANFIS-CSA) was contrasted with, respectively, ANFIS and ANFIS-GA. It ensures high OHTC prediction performance with reasonable accuracy when compared to others
Parvathavarthini [16]	CSA with Fuzzy-C-means clustering	University of Eastern Finland's Benchmark UCI data repository datasets and artificial datasets	the method aims to avoid the delay in convergence rate.	Its results were very efficient, based on multiple parameters such as error rate, objective function value, and cluster validity indices.

(Table 1) cont.....

Paper	Proposed method	Dataset used	Method objective	Findings
Ahmed [17]	Improved fast fuzzy c-means using CSA (FFCM-CSA)	Images from maize field	1.The method is used for crop identification in agriculture. 2. Aims to increase the computational performance of FFCM by using CSA to find clusters centers due to its ability to provide very accurate results in the clustering process, which contributes to avoiding the occurrence of FFCM in the local Optima	The method finds the ideal centers value and avoid falling into the local optima.
Parvathavarthini [18]	Intuitionist fuzzy clustering focused on CSA with neighborhood attraction called(CrSA-IFCM-NA)	Breast cancer data	To detect cancer masses at an early stage through effective clustering with neighborhood attraction.	Its results showed accuracy in detecting cancerous masses at an early stage through its ability to effectively separate cancerous masses from mammography images.
Ahmed [5]	CSA with chaos theory and fuzzy c-means algorithm (CFCSA)	Breast cancer, diabetes, lung cancer, hepatitis, Radiopaedia CT liver, and cardiotocography	Used for feature selection for medical diagnostic problems	It showed good results in the diagnosis of these diseases compared to other algorithms

(Table 1) cont.....

Paper	Proposed method	Dataset used	Method objective	Findings
Xiufang Lin [20]	A modified crow search algorithm with fuzzy logic strategy [MCSA-FLC]	The adjacent full-scale structures connected by MR dampers with the SSI	The approach seeks to define a smart control technique that uses magnetic dampers as communication devices when considering soil-structure interaction..	It has displayed substantial performance superiority over its competitors, *i.e*. passive-off, passive-on, on-off, linear quadratic regulator-clipped voltage rule, and linear-quadratic Gaussian-clipped voltage rule voltage rule regulation.

7.3.1.3. Other updates of CSA

Some researchers make preliminary work to improve CSA's optimization efficiency through other useful strategies. Some of the applications applying the modifications of CSA are listed in Table **2**.

Table 2. Other modifications of CSA.

paper	Proposed method	Method Objectives	Findings
Hossein [21]	Improved Crow Search Algorithm [ICSA]	This method has a new operator parameter added to the Low-Cost Design of Water Distribution Networks, which helped to improve crows' generations to find the global optimum.	To assess its efficiency, CSA and ICSA applied two WDN approaches including (Two-Reservoir and Khorramshahr City). The results of the proposed algorithm showed a better design compared to the original algorithm for the two problems.
Zhaojun [22]	Improved Crow Search Algorithm [ICSA]	This method aims to enhance exploration, exploitation, and convergence speed capabilities	To test the algorithm's efficiency, it was implemented on several standard unconstrained benchmark functions, using different features of each problem.
Farid [23]	Modification of CSA (MCSA)	This method has used for Solving Economic Load Dispatch (ELD) Problem.	Its results, compared to other technologies, showed remarkable superiority in solution quality, durability, and computing time

(Table 2) cont.....

paper	Proposed method	Method Objectives	Findings
Almoataz [24]	Modified Crow Search Algorithm [MCSA]	This technique has been used in a radial distribution network using CSA for the optimum conductor size selection process. The goal of the process of optimal selection of the conductor size is to reduce energy loss in addition to reducing the annual operating cost of the system	The proposed methodology has proven effective in selecting the ideal conductors in both the small and large band compared to other algorithms.
Ali [25]	Improved Crow Search Algorithm [ICSA]	This method has used to classify and process text documents	It helped to save time and effort to find the documents concerned in addition to helping in the process of extracting samples through the ability to identify and select the best samples. The algorithm showed great accuracy in the classification process and also boosted the classification rate by 27% compared to the KNN model.
Primitivo Diaz [26]	Improved Crow Search Algorithm [ICSA]	This method has been used to solve complex energy problems. This algorithm aims to enhance the rate of exploitation and exploration.	To study the effectiveness of the algorithm, it was compared with: DE, ABC, and GSA algorithms where the results proved the high performance of this algorithm through a variety of solutions that offer within the search space in addition to improving union to tough high multi-modal optima.
Marichelvam [27]	Improved Crow Search Algorithm [ICSA]	This method has been used for solving single machine scheduling problems by minimizing the total weighted lateness.	Mean Relative Percentage Deviation (MRPD) and computational time value have been used to assess the efficiency of these algorithms. It proved the efficiency of this proposed algorithm to solve single machine scheduling problems.
Mohamed [28]	A modified Crow Search Algorithm [MCSA]	This method aims to enhance the performance of the active radial distribution networks by determining the location and size of the distributed generators (DGs) that play a major role in reducing energy loss and enhancing the voltage profile.	The results of the proposed method showed the ability to know the optimal location and size of the DG, in addition to reducing real energy loss and enhancing the system's voltage profile. In addition to its superior performance over other algorithms such as PSO, GA, and GA-PSO by providing the minimum value of power loss among all algorithms.

(Table 2) cont.....

paper	Proposed method	Method Objectives	Findings
Hao Wu [29]	Levy Flight Crow Search Algorithm [LFCSA]	A finite element model updating (FEMU) to obtain higher accuracy in the process of analyzing fatigue and vibration of an elastic body. Where the algorithm was applied to update two different states: simple structure (beam) and complex structure (gearbox housing)	The LFCSA algorithm was compared with both CSA and LFPSO as its results showed obvious effects on FEMU as it showed high accuracy in addition to high-convergence speed and global search ability in both simple and complex structures.
Deepak [30]	A modified Crow Search Algorithm [MCSA]	This method aims to reach the optimum solution in the process of extracting the usability features that are most beneficial from the hierarchical model	The results of the algorithm were compared with BBA, CSA, and MWOA where It showed better results in the process of selecting features compared to other algorithms
Ronali [31]	Improved Crow Search Algorithm [ICSA]	The goal of this method is the ability to schedule dynamic tasks in the heterogeneous multiprocessor system which is a difficult problem in a network environment.	Its results showed superiority over many other standard algorithms that were compared with it like Genetic Based Bacteria Foraging (GBF), Bacteria Foraging Optimization (BFO)
Bighnaraj [32]	Enhanced Crow Search Algorithm (ECSA)	This method has used for solving classification problem in data extraction by enhancing the functional neural network linking system using an enhanced CSA to avoid convergence and early stagnation of the system	To assess the performance of the proposed model, the CSA FLA-based FLA trainers were compared with FLANN GA-based trainers and PSO FLANN trainers. The CSA-optimized model showed superiority in the classification speed and accuracy with fewer iterations compared to other models.

7.3.2. Hybridization

CSA has been combined with some conventional and evolutionary optimization algorithms to take advantage of both approaches and counteract each other's weaknesses. Both the traditional and evolutionary optimization algorithms have their own strength and weakness. Traditional algorithms generally rely on finding a random solution that is chosen randomly, and then it tries to find the optimal solution repeatedly as it depends on two basic parameters, which are the search direction and the length of the step [33]. Traditional optimization tools are classified into two groups, namely direct search, and gradient-based methods, as these tools suffer from a lack of guarantee of a globally optimal solution due to the dependence of the final solution on the random initial solution in addition to

its ability to solve one type of problem and its lack of multiple-use improvement methods [33]. On the other hand, non-traditional algorithms have succeeded in solving the complex optimization problems that conventional tools have failed to solve [33]. Where these tools depend on the modeling and simulation of natural processes, such as artificial biological and physical processes, in order to develop optimization tools to solve complex optimization problems [33]. There is a huge literature on non-traditional optimization tools like Genetic Algorithm (GA) [34], Ant Colony Optimization (ACO) [12], Tabu Search (TS) [35], Swarm Intelligence (SI) [8], Particle Swarm Optimization (PSO) [7]. The type of CSA which combines the goodness of both traditional and nontraditional(evolutionary) optimization algorithm is called hybridized CSA. In the present section (Table **3**) a review of the application of hybridized CSA on different engineering fields is given.

Table 3. Application of hybridized CSA in different engineering fields.

Paper no.	hybridized with Traditional Optimization Algorithm	Hybridized with Evolutionary Algorithm	Name of Hybridized Algorithm	Method Objectives	Findings
Sankalp [36]		√	A hybrid GWO with CSA (GWOCSA)	To solve the optimization of functions and feature selection problems	This algorithm helps to explore the Global Optima. Where its results showed superiority in avoiding the local Optima in addition to the speed of high convergence.
Dhanya [37]		√	Hybrid Crow Search and Ant Colony Optimization Algorithm (ACO-CSA)	To address capacitated routing problem for vehicles (CVRP)	The algorithm was evaluated on instances of Augerat CVRP and also showed better results with better computational time

(Table 3) cont.....

Paper no.	hybridized with Traditional Optimization Algorithm	Hybridized with Evolutionary Algorithm	Name of Hybridized Algorithm	Method Objectives	Findings
Pamir [38]		√	A hybrid bat and crow search algorithm (BCSA)	To manage household electrical energy in the smart grid	Critical Peak Pricing System (CPP) was used to calculate the amount of electricity consumed. The results indicated a decrease in the load rate to fewer periods decreasing in the total cost of electricity payments.
Soukaina [39]		√	A Crow Search and Genetic Algorithm (CSGA)	For solving 2-dimensional bin packing problem (2D-BPP)	The efficiency of the algorithm is compared with both standard GA and BPSO swarm optimization algorithm where it showed very promising results
Mahesh [40]		√	Hybridization of Dolphin echolocation and crow search optimization Algorithm (DECSA)	To solve the energy-aware cluster routing in wireless sensor networks (WSN)	The results of the simulation process demonstrated the effectiveness of the proposed method as it provided a better lifespan for the residual energy network.
Kaladhar [41]		√	Crow search mating-based lion algorithm (CSM-LA)	To solve the optimal position and configuration in distribution networks of centralized power quality conditioner (UPQC).	The hybrid algorithm uses a multi-objective, nonlinear method that decreases the cost of energy and UPQC as well as Voltage Stability Index (VSI) costs.

(Table 3) cont.....

Paper no.	hybridized with Traditional Optimization Algorithm	Hybridized with Evolutionary Algorithm	Name of Hybridized Algorithm	Method Objectives	Findings
Raj Kumar [42]		√	Hybrid Cat Swarm and Crow Search Algorithm (CSO-CSA)	To solve the combined economic emission dispatch (CEED) model in the smart grid.	The algorithm is simulated and it is compared to algorithms like PSO and GWO where the results of the analysis are superior to the algorithm mainly in CEED cost model
Seyed [43]	√		Sine Cosine Crow Search Algorithm (SCCSA)	To solve the global optimization problems	The outcomes verified the capacity of the algorithm to find a competitive solution in most standard functions in addition to their convergence to the optimal solution without falling into the local Optima
Abul Ela [44]	√		A hybrid crow search algorithm based on rough searching scheme (RCSA)	To solve engineering optimization problems	Its results showed its ability to solve complex engineering problems with high efficiency and effectiveness
KO-WEI [45]	√		A hybrid Crow Search Algorithm (HCSA)	To minimize the makespan of permutation flow shop scheduling problems (PFSPs)	The findings demonstrate that the HCCA algorithm is performing much better than the other algorithms.

(Table 3) cont.....

Paper no.	hybridized with Traditional Optimization Algorithm	Hybridized with Evolutionary Algorithm	Name of Hybridized Algorithm	Method Objectives	Findings
Mohcin [47]	√		A hybrid algorithm-based on CSA and Local Search method called (CSA-P2M □Fit)	This method has used for solving the DNA fragment assembly problem.	The proposed algorithm outperforms other algorithms in solving the problem of DNA fragment assembly by finding new solutions to the permutations problem.
R.S. Chithra [49]	√		Fractional crow search-based support vector neural network (FC-SVNN)	The objective of this method is to classify Tuberculosis (TB) patients automatically in terms of severity to reduce mortality based on arithmetic speed.	In addition to providing better performance in terms of accuracy rate, false-positive rate(FPR), and true positive rate (TPR). the findings showed that the classifier had a high accuracy.
Ze-Xue Wu [46]		√	A hybrid algorithm based on Whole optimization algorithm (WOA) and Crow search algorithm (CSA) called (HWCA)	For solving data clustering	To assess its performance, it was compared with both CSA and WOA on familiar UCI benchmarks, where its results have advanced accurateness compared to the listed algorithms.

(Table 3) cont.....

Paper no.	hybridized with Traditional Optimization Algorithm	Hybridized with Evolutionary Algorithm	Name of Hybridized Algorithm	Method Objectives	Findings
Nitin [48]		√	A hybrid algorithm-based on CSA and dragonfly algorithm (DA) called (D-Crow)	The goal of this method is to create a new virtual machine migration (VMM) strategy to achieve load balance in the cloud model	The proposed algorithm was compared with current techniques, with simulation results showing better values for the proposed model by achieving minimum values for load 7.371%, energy consumption 10.0368%, and migration cost 11.0639%
Ishtiaq [50]		√	CSA with harmony search algorithm (CSA-HAS)	The hybrid algorithm aims to improve the efficiency of the home energy management system by reducing energy consumption, the cost of electricity, and the peak to average ratio (PAR)	The findings showed there would always be a trade-off between the cost of electricity and the waiting time.
NimmolP.John [51]		√	hybrid firefly-crow optimization algorithm (HFCOA)	The goal of this hybridization is to find an effective virtual machine consolidation based on multi-objectives.	The proposed hybrid algorithm was contrasted with the FA and CSA, where the simulation results showed a significant performance superiority to the proposed algorithm compared to others.

(Table 3) cont.....

Paper no.	hybridized with Traditional Optimization Algorithm	Hybridized with Evolutionary Algorithm	Name of Hybridized Algorithm	Method Objectives	Findings
Senthil Kumar [52]		√	hybrid Firefly Crow Search Algorithm) FF-CSA)	This hybrid algorithm aimed to solve the problem of cloud computing scheduling tasks by reducing the makespan and increasing the cloud system's throughput	The results of the proposed algorithm showed higher performance compared to both FA and CSA
Ashok George [53]		√	Crow search based Lion search algorithm (C-LION)	The proposed approach aims to create a new technique to protect privacy in cloud systems during the data transfer process using both the Dyadic product and the proposed optimization algorithm.	The C-LION achieved total utility of 0.909 for the breast cancer dataset with privacy of 0.864.

7.3.3. Multi-objective and Binary Optimization.

To deal with multi-objective optimization problems, both Multi-objective CSA (MOCSA) and Binary CSA (BCSA) are suggested. Where Pareto Optimization is one of the multi-objective decision-making areas that are concerned with solving the problems of mathematical optimization with multiple objective functions to improve it simultaneously. It has been used in areas that need to make optimal decisions in the event of a comparison between two or more conflicting goals, the most important of which are engineering, economics and logistics. On the other hand, In both computer vision and machine learning, binary optimization captures a number of applications. where decision variables can only take +1 or -1 values. In this section, Table **4** discusses the applications of MOCSA and BSA in different engineering fields.

Table 4. Applications of BCSA and MOCSA in different fields

Paper	Binary CSA	Multi-objective CSA	Data set used	Objective of the method	Findings
Hadi [54]		√	13 unconstrained multi-objective tests from the CEC 2009 Special Session and Competition	For multi-objective optimization problems	This algorithm relies on improving the effectiveness of the search space by defining the fitness function by using a specific set of weight vectors designed in a way that equally achieves the Pareto front, Apart from using the Min-Max approach.
Zahra [55]		√	Data was collected from the electricity distribution company of Kerman	Designing a hybrid energy system for photovoltaic-diesel generators in the presence of an operating reserve	To evaluate the proposed model, it was evaluated with MOPSO. Looking at the results, the proposed algorithm showed a cost-effective and reliable system for generating electrical energy through the hybridization of VP and DG systems in addition to providing a well-distributed Pareto interface due to its promising results more than MOPSO.
Rodrigo [56]	√		Six benchmark data sets: Breast, Ionosphere, Spect, Stat log, Wine, Zoo.	To boost the efficiency of the CSA classic algorithm in the feature selection process to decrease the dimensions of data sets due to their ease of implementation, convergence speed, and high efficiency	The proposed algorithm showed 100% accuracy for choosing one set of data in addition to obtaining a computational cost relatively close to BPSO
Yassine [58]	√		DIMACS benchmark instances	To solve the graph coloring problem so that fast convergence is avoided towards the local Optima in addition to the diversity of performance evaluation solutions	DIMACS benchmark instances were used to assess the algorithm's efficiency and Compare it to five other algorithms, including MCOACOL,NPGA,HPGAs, MACOL, and BEECOL, and showed a high-performance accuracy.

(Table 4) cont.....

Paper	Binary CSA	Multi-objective CSA	Data set used	Objective of the method	Findings
Jacob John [59]		√	NA	The proposed algorithm aims to overcome the problem of data transmission between nodes using routing protocols in wireless sensor networks by selecting the optimal cluster head for the energy-aware model using MOTCO	An architectural WSN was used with 50, 100 nodes to evaluate the results, which showed that despite the loss of energy between the nodes during the increase in the number of rounds, MOTOC outperformed other algorithms compared to the amount of energy remaining in the nodes.

7.3.4. Other applications: To date, CSA has been used in many other applications in varied academic and industrial fields. Table 5 shows the other applications of CSA.

Table 5. Other Applications of CSA

paper	Proposed method	Method Objectives	Findings
Nur Arif [60]	CSA	For solving travelling salesman problem	The findings have shown an important and viable solution despite the algorithm itself which has not been optimized to reach its full potential yet.
Nurhadi [61]	CSA	The goal of the algorithm application is to develop a metaphoric model to improve the utility of a small commercial aircraft that has an ATR-72 propeller motor.	To test the efficiency of the algorithm, it was contrasted with other conventional algorithms such as PSO, PSO-GRASP, and its results showed superiority over PS, while it provided worse performance compared to PSO-GRASP.
Dina A. Zak [62]	CSA	The proposed method is intended to increase the efficiency of the inverter-based generation system by using the CSA PI parameter controller..	The simulation outcomes of the CSA algorithm were related with the GA algorithm where the CSA showed superiority at a time Implementation and the number of iterations in addition to the higher-level energy cycles.

(Table 5) cont.....

paper	Proposed method	Method Objectives	Findings
Dong Liu [63]	CSA-ELM	This model was used to assess the quality of groundwater with multiple parameters by relying on finding a good method for evaluation and finding good parameters that suit the evaluation process using an improved model for the ELM as well as adjusting the income weights and thresholds of neurons with hidden layers in the ELM using CSA	To evaluate the assumed model, it was tested based on a set of training and test samples. The results showed an accurate assessment of the quality of groundwater at a high level.
K. R. Prasanna [64]	CSA	This method was used to find a suitable virtual machine for the task in addition to reducing the makespan in task scheduling in the cloud computing	Experimental algorithm tests were performed and compared with ACO and MIN-MIN. CSA showed superiority in reducing the values of makespan compared to other algorithms.
K. M. Dhanya [65]	CSA	This method was used for solving capacitated vehicle routing problem (CVRP)	To assess its performance, analytical design was used ANOVA using different parameter settings where the evaluation results showed the large role of AP on CSA performance in the case of a large CVRP instance.
Asma Meddeb [66]	CSA	The goal of this research is to solve the problem of optimal dispatch of reactive power (ORPD) using CSA to achieve the least power loss when meeting a number of nonlinear limitations.	The following test systems were used IEEE 14-bus, IEEE 30-bus, and Tunisia's large-scale 86-bus system to assess the performance of the algorithm, as CSA received results of dominance and statistically relevant results of ORPD problems (for IEEE-14 $p < 0.0006$, IEEE-30 bus < 0.006, Tunisian 86 bus $p < 0.0000001$)

7.4. APPLICATION OF CSA IN MEDICAL DOMAIN

The CSA algorithm has been applied in different engineering fields. The algorithm is also being applied in the field of medicine and being appreciated by the researchers for its performance. In the present section the application of variations of CSA in the medical domain is presented. Table **6**, presents a systematic survey of the variations of CSA in medical domain. The variations are categorized as modifications of CSA, Hybridization of CSA and Extensions of CSA. The section gives a brief information about the method used and the objective of the method in the research paper and the findings of the paper.

Table 6. Application of variations of CSA in different medical field

Modifications of CSA	Hybridization of CSA	Extension of CSA	Medical Data	Objective of the Method	Findings
CCSA [67]			Sectional images of lungs	This method has been used for early detection of lung cancer based on features classification where CCSA has used to select the features from the extracted features.	The results of the applied algorithm were measured based on several metrics like: Specificity, Sensitivity, Positive and Negative Predictive Values, Where the results showed accuracy of up to 90
CSA-IFCM-NA [18]			Mammographic Image Analysis Society (mini-MIAS) database	Diagnosing breast cancer by relying on mammography. Where this technique has used to identify the most important areas, by extracting features from the previously processed images.	The results of the algorithm guarantee the ability to separate masses by mammographic images effectively and accurately, by distinguishing cancerous masses from the normal tissue regions.
MCET-CSA [68]			Brain-web Database	This method has been used for magnetic resonance brain image segmentation to evaluate the optimal entropy thresholding values which consider as an important method for image segmentation.	It showed good outcomes in terms of superiority and reliability in generating fractionated magnetic resonance imaging by dividing brain images into different depths.

(Table 6) cont.....

Modifications of CSA	Hybridization of CSA	Extension of CSA	Medical Data	Objective of the Method	Findings
OCSA [11]			20 UCI Machine Learning Repository Benchmark Datasets	This algorithm used to diagnose Parkinson's disease at an early stage because of the feature extraction usability and prediction, where it depends on picking the most optimal subset of features from the dataset	The results have shown that OCSA offers more accuracy and reliability in the feature extraction process by creating an ideal subset of the features.
FCSA [69]			CT abdomen datasets	This method has used for the image segmentation using Fuzzy-C means	Satisfactory results were shown in the segmentation of abdomen CT images related with ABC, Firefly, SA algorithms
	CFCSA [19]		Cardiotocography, IL PD, liver diseases, hepatitis, arrhythmias, dibeties, pulse, radiopaedia, CT liver, breast cancer, lung cancer	1.CFCSA is applied for feature selection in medical diagnosis 2. To prevent the sensitivity of local optimization, CFCSA adapts the global optimization technique. The fuzzy c-means (FCM) goal function is used as a cost method for the chaotic crow search optimization algorithm.	The binary crow search algorithm (BCSA), chaotic ant lion optimization algorithm (CALO), binary ant lion optimization algorithm (BALO), and bat algorithm are benchmarked against CFCSA and proved better in comparison to the above-mentioned algorithm in feature selection

(Table 6) cont.....

Modifications of CSA	Hybridization of CSA	Extension of CSA	Medical Data	Objective of the Method	Findings
		V-shaped BCSA [57]	Breast cancer data	The method has used for feature selection	The method demonstrates good results with few numbers of selected features in terms of classification precision and computational cost.
	HWCA [46]		Breast cancer data	The method has used for data clustering	Its results have a higher accuracy rate compared to the listed algorithms compared in the literature.
	FC-SVNN [49]		ZNSM-iDB database	The proposed method aims to classify Tuberculosis (TB) patients automatically in terms of severity to reduce mortality based on arithmetic speed and reducing the time spent analyzing the test samples.	The experimental results demonstrated a high accuracy of the proposed classifier in addition to providing enhanced accuracy in terms of accuracy rate, false-positive rate (FPR) and true positive rate(TPR).
		Cross-Entropy Based CSA [70]	Database of IXI, MIRIAD, and ADNI 2	This method depends on the different brain tissue in MR images to detect dementia by using the Crow Search Algorithm and Structure Tensor Features Focused on Minimal Cross-Entropy.	Its result demonstrated that CSA achieved a better segmentation of brain tissue compared to BFO focused on measurements of similarity and time of computation.

7.5. DISCUSSION AND CRITICAL ANALYSIS

The number of CSA-related publications shown in Fig. (**4**) during the year. We clearly notice that the number of applications has increased significantly from 2016 to 2020 and therefore we notice increased interest in the use of CSA in various areas in addition to the expectations of obtaining more satisfactory results over the next few years. One of the problems that CSA sometimes suffers from is the lack of proper handling between exploitation and exploration, which leads to a rapid convergence towards the local Optima. Therefore, to solve this problem, the researchers suggested a lot of the solutions that we mentioned in Section (3) including: major modifications to the algorithm like Fuzzy-CSA, Chaotic-CSA, In addition to hybridization of CSA with other algorithms such as ACO, GWP, CSE, etc. After that, CSA was expanded to include other areas of improvement, such as multi-objective optimization and binary optimization. With the widespread use of CSA over the past two years, there are dozens of CSA variants in addition to the multitude of test functions at the present time, which makes it difficult to choose the best modification that has been used with CSA due to the good results presented in different areas of its use. Hence, it is better to create a platform where researchers search and suggest, then choose the best CSA variant based on a fair and comprehensive comparison. Fig. (**3**) depicts the publication number of CSA in different category.

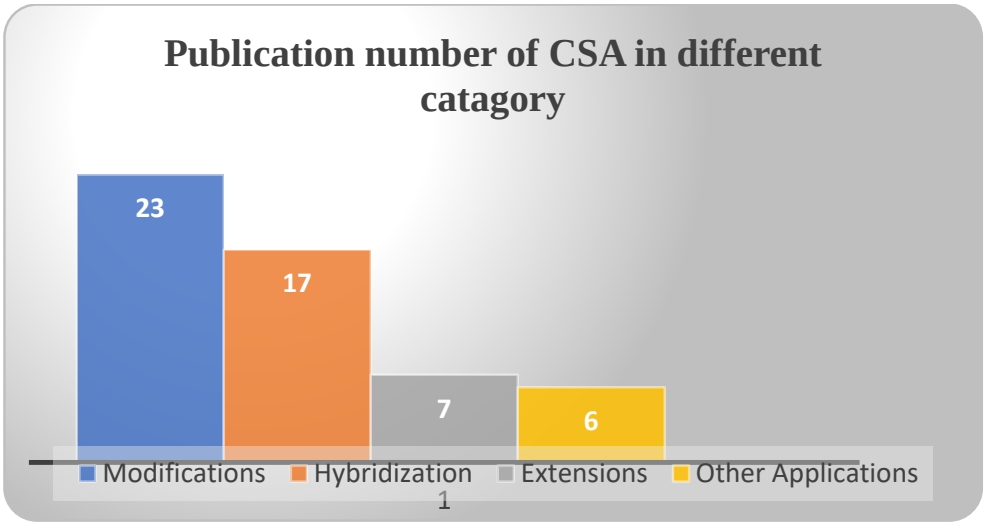

Fig. (3). Publication Number of CSA.

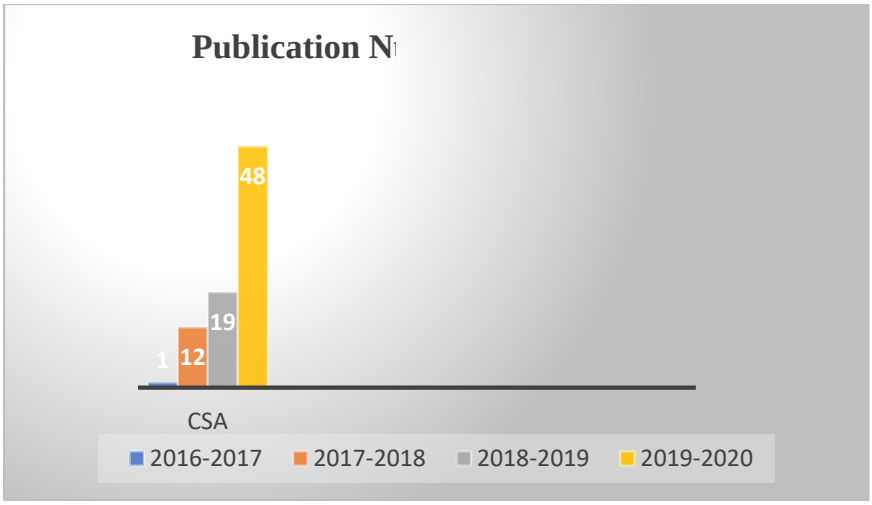

Fig. (4). Publication Number against year.

CONCLUSION

Evolutionary algorithms form a subset of evolutionary computation which is a subfield of Artificial Intelligence, in that they typically require only techniques that enforce biological evolution-inspired processes such as replication, mutation, recombination, natural selection and survival of the fittest. Crow Search Algorithm showed through the amendments and applications made by the researchers a great diversity in the existing solutions, unlike other optimization algorithms that we compared with them. It relied on using the best position to find new functions, in addition to containing less parameters (FL, AP) and thus saving time during the process of tuning and setting parameters. CSA performed well in the different fields we studied, particularly in the classification and selection of features, as its findings showed great efficacy in the medical field, in particular its contribution, in addition to many other diagnoses, to the diagnosis of breast cancer. We find from the survey that CSA has been applied in different fields of engineering, but not so much in the medical field. Its satisfactory results in this field make CSA one of the most important optimization algorithms that can be relied upon and used in future research related to the medical field. We will focus our next research on making a comprehensive comparison of CSA with the various other algorithms used to solve problems related to the medical field.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] I. Stojanović, I. Brajević, P.S. Stanimirović, L.A. Kazakovtsev, and Z. Zdravev, "Application of Heuristic and Metaheuristic Algorithms in Solving Constrained Weber Problem with Feasible Region Bounded by Arcs", In: *Math. Probl. Eng.*, 2017.
[http://dx.doi.org/10.1155/2017/8306732]

[2] I. Fister, X.S. Yang, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization.", In: *Elektroteh. Vestnik/Electrotechnical Rev* vol. 80. , 2013, no. 3, pp. 116-122.

[3] D.S. Tahir, and R.S. Ali, "A Chaotic Crow Search Algorithm for High-Dimensional Optimization Problems", *Basrah J. Eng. Sci.,* vol. 17, no. 1, pp. 16-25, 2017.
[http://dx.doi.org/10.33971/bjes.17.1.3]

[4] D. Karaboga, and B. Basturk, *Artificial Bee Colony (ABC) optimization algorithm for solving constrained optimization problems*, 2007.
[http://dx.doi.org/10.1007/978-3-540-72950-1_77]

[5] A.M. Anter, and M. Ali, "Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems", *Soft Comput.,* vol. 24, no. 3, pp. 1565-1584, 2020.
[http://dx.doi.org/10.1007/s00500-019-03988-3]

[6] H.R. Ahmed, and J.I. Glasgow, "Swarm Intelligence: Concepts, Models and Applications", *Queen's Univ. Sch. Comput. Tech. Reports,* vol. 585, pp. 1-50, 2012.
[http://dx.doi.org/10.13140/2.1.1320.2568]

[7] X.L. Xu, X. Cheng, Z.C. Yang, X.H. Yang, and W.L. Wang, "Improved particle swarm optimization for Traveling Salesman Problem", *27th Eur. Conf. Model. Simulation, ECMS 2013,* 2013pp. 857-862
[http://dx.doi.org/10.7148/2013-0857]

[8] D. Karaboga, and B. Akay, "A survey: Algorithms simulating bee swarm intelligence", *Artif. Intell. Rev.,* vol. 31, no. 1–4, pp. 61-85, 2009.
[http://dx.doi.org/10.1007/s10462-009-9127-4]

[9] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm", *Comput. Struc.,* vol. 169, pp. 1-12, 2016.
[http://dx.doi.org/10.1016/j.compstruc.2016.03.001]

[10] G.I. Sayed, A.E. Hassanien, and A.T. Azar, "Feature selection via a novel chaotic crow search algorithm", *Neural Comput. Appl.,* vol. 31, no. 1, pp. 171-188, 2019.
[http://dx.doi.org/10.1007/s00521-017-2988-6]

[11] D. Gupta, S. Sundaram, A. Khanna, A. Ella Hassanien, and V.H.C. de Albuquerque, "Improved diagnosis of Parkinson's disease using optimized crow search algorithm", *Comput. Electr. Eng.,* vol. 68, pp. 412-424, 2018.
[http://dx.doi.org/10.1016/j.compeleceng.2018.04.014]

[12] M. Dorigo, and C. Blum, "Ant colony optimization theory: A survey", *Theor. Comput. Sci.,* vol. 344, no. 2–3, pp. 243-278, 2005.
[http://dx.doi.org/10.1016/j.tcs.2005.05.020]

[13] R.K. Khadanga, S. Padhy, S. Panda, and A. Kumar, "Design and analysis of tilt integral derivative controller for frequency control in an islanded microgrid: a novel hybrid dragonfly and pattern search algorithm approach", *Arab. J. Sci. Eng.,* vol. 43, no. 6, pp. 3103-3114, 2018.
[http://dx.doi.org/10.1007/s13369-018-3151-0]

[14] R.M. Rizk-Allah, A.E. Hassanien, and S. Bhattacharyya, "Chaotic crow search algorithm for fractional optimization problems", *Appl. Soft Comput.,* vol. 71, pp. 1161-1175, 2018.
[http://dx.doi.org/10.1016/j.asoc.2018.03.019]

[15] M.A. Elaziz, A.H. Elsheikh, and S.W. Sharshir, "Improved prediction of oscillatory heat transfer coefficient for a thermoacoustic heat exchanger using modified adaptive neuro-fuzzy inference system", *Int. J. Refrig.,* vol. 102, pp. 47-54, 2019.
[http://dx.doi.org/10.1016/j.ijrefrig.2019.03.009]

[16] S. Parvathavarthini, N.K. Visalakshi, S. Shanthi, and J.M. Mohan, "Crow search optimization based fuzzy C-means clustering for optimal centroid initialization", *Taga J Graphic Technol,* vol. 14, pp. 3034-3035, 2018.

[17] A.M. Anter, A.E. Hassenian, and D. Oliva, "An improved fast fuzzy c-means using crow search optimization algorithm for crop identification in agricultural", *Expert Syst. Appl.,* vol. 118, pp. 340-354, 2019.
[http://dx.doi.org/10.1016/j.eswa.2018.10.009]

[18] P. S, K.V. N, and S. S, "Breast cancer detection using crow search optimization based intuitionistic fuzzy clustering with neighborhood attraction", *Asian Pac. J. Cancer Prev.,* vol. 20, no. 1, pp. 157-165, 2019.
[http://dx.doi.org/10.31557/APJCP.2019.20.1.157] [PMID: 30678427]

[19] S. Srivastava, and S.K. Sahana, "Application of Bat Algorithm for Transport Network Design Problem", *Appl. Comput. Intell. Soft Comput.,* vol. 2019, pp. 1-10, 2019.
[http://dx.doi.org/10.1155/2019/9864090]

[20] X. Lin, S. Chen, and W. Lin, "Modified crow search algorithm–based fuzzy control of adjacent buildings connected by magnetorheological dampers considering soil–structure interaction. Journal of Vibration and Control",
[http://dx.doi.org/10.1177/1077546320923438]

[21] H. Fallah, O. Kisi, S. Kim, and M. Rezaie-Balf, "A New Optimization Approach for the Least-Cost Design of Water Distribution Networks: Improved Crow Search Algorithm", *Water Resour. Manage.,* vol. 33, no. 10, pp. 3595-3613, 2019.
[http://dx.doi.org/10.1007/s11269-019-02322-8]

[22] Z. Shi, Q. Li, S. Zhang, and X. Huang, "Improved Crow Search Algorithm with Inertia Weight Factor and Roulette Wheel Selection Scheme", *Proc. - 2017 10th Int. Symp. Comput. Intell. Des. Isc.,* vol. 1, 2017pp. 205-209
[http://dx.doi.org/10.1109/ISCID.2017.140]

[23] F. Mohammadi, and H. Abdi, "A modified crow search algorithm (MCSA) for solving economic load dispatch problem", *Appl. Soft Comput.,* vol. 71, pp. 51-65, 2018.
[http://dx.doi.org/10.1016/j.asoc.2018.06.040]

[24] A.Y. Abdelaziz, and A. Fathy, *A novel approach based on crow search algorithm for optimal selection of conductor size in radial distribution networks*, 2017.
[http://dx.doi.org/10.1016/j.jestch.2017.02.004]

[25] A. Allahverdipour, and F. Soleimanian Gharehchopogh, "An improved k-nearest neighbor with crow search algorithm for feature selection in text documents classification", *Journal of Advances in Computer Research,* vol. 9, no. 2, pp. 37-48, 2018.

[26] P. Díaz, M. Pérez-Cisneros, E. Cuevas, O. Avalos, J. Gálvez, S. Hinojosa, and D. Zaldivar, "An improved crow search algorithm applied to energy problems", *Energies,* vol. 11, no. 3, p. 571, 2018.

[http://dx.doi.org/10.3390/en11030571]

[27] M.K. Marichelvam, K. Manivannan, and M. Geetha, *Solving Single Machine Scheduling Problems using an Improved Crow Search Algorithm*, 2016.

[28] M. Abdelbadea, T.A. Boghdady, and D.K. Ibrahim, "Enhancing active radial distribution networks by optimal sizing and placement of DGs using modified crow search algorithm", *Indones. J. Electr. Eng. Comput. Sci.,* vol. 16, no. 3, pp. 1179-1188, 2019.
[http://dx.doi.org/10.11591/ijeecs.v16.i3.pp1179-1188]

[29] H. Wu, P. Wu, K. Xu, and F. Li, "Finite element model updating using crow search algorithm with Levy flight", *Int. J. Numer. Methods Eng.,* vol. 121, no. 13, pp. 2916-2928, 2020.
[http://dx.doi.org/10.1002/nme.6338]

[30] D. Gupta, J.J.P.C. Rodrigues, S. Sundaram, A. Khanna, V. Korotaev, and V.H.C. de Albuquerque, "Usability feature extraction using modified crow search algorithm: a novel approach", *Neural Comput. Appl.,* vol. 6, 2018.
[http://dx.doi.org/10.1007/s00521-018-3688-6]

[31] R.M. Sahoo, and S.K. Padhy, Improved Crow Search Optimization for Multiprocessor Task Scheduling: A Novel Approach
[http://dx.doi.org/10.1007/978-3-030-30271-9_1]

[32] B. Naik, D. Mishra, J. Nayak, D. Pelusi, and A. Abraham, *Perturbation based efficient crow search optimized FLANN for system identification: A novel approach.* vol. Vol. 734. Springer International Publishing, 2018.
[http://dx.doi.org/10.1007/978-3-319-76351-4_21]

[33] X-S. Yang, and S. Koziel, *Computational optimization and applications in engineering and industry.* vol. Vol. 359. Springer Science & Business Media, 2011.
[http://dx.doi.org/10.1007/978-3-642-20986-4]

[34] J. McCall, "Genetic algorithms for modelling and optimisation", *J. Comput. Appl. Math.,* vol. 184, no. 1, pp. 205-222, 2005.
[http://dx.doi.org/10.1016/j.cam.2004.07.034]

[35] R.A. Gallego, R. Romero, and A.J. Monticelli, "Tabu search algorithm for network synthesis", *IEEE Trans. Power Syst.,* vol. 15, no. 2, pp. 490-495, 2000.
[http://dx.doi.org/10.1109/59.867130]

[36] S. Arora, H. Singh, M. Sharma, S. Sharma, and P. Anand, "A New Hybrid Algorithm Based on Grey Wolf Optimization and Crow Search Algorithm for Unconstrained Function Optimization and Feature Selection", *IEEE Access,* vol. 7, no. c, pp. 26343-26361, 2019.
[http://dx.doi.org/10.1109/ACCESS.2019.2897325]

[37] K.D. Shinde, K.A. Kumar, D.S. Rashmi, R.S. Rukhsar, H.R. Shilpa, and C.R. Vidyashree, *Hybrid Crow Search-Ant Colony Optimization Algorithm for Capacitated Vehicle Routing Problem K.* vol. Vol. 837. Springer Singapore, 2018.

[38] N. Javaid, S.M. Mohsin, A. Iqbal, A. Yasmeen, and I. Ali, "A Hybrid Bat-Crow Search Algorithm Based Home Energy Management in Smart Grid", *Conference on Complex, Intelligent, and Software Intensive Systems,* 2018pp. 75-88

[39] S. Laabadi, M. Naimi, H. El Amri, and B. Achchab, "A Crow Search-Based Genetic Algorithm for Solving Two-Dimensional Bin Packing Problem", *In Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz) ,* 2019pp. 203-215 Springer, Cham
[http://dx.doi.org/10.1007/978-3-030-30179-8_17]

[40] N. Mahesh, and S. Vijayachitra, "DECSA: hybrid dolphin echolocation and crow search optimization for cluster-based energy-aware routing in WSN", *Neural Comput. Appl.,* vol. 31, no. S1, pp. 47-62, 2019.
[http://dx.doi.org/10.1007/s00521-018-3637-4]

[41] K. Gaddala, and P.S. Raju, "Merging Lion with Crow Search Algorithm for Optimal Location and Sizing of UPQC in Distribution Network", *J. Control. Autom. Electr. Syst.,* vol. 31, no. 2, pp. 377-392, 2020.
[http://dx.doi.org/10.1007/s40313-020-00564-1]

[42] R. Kumar, "Journal of Computational Mechanics", *Power System and Control Hybrid Cat Swarm and Crow Search Algorithm to Solve the Combined Economic Emission Dispatch Model for Smart Grid,* vol. 2, no. 3, pp. 10-18, 2019.

[43] Pasandideh , Seyed Hamid Reza , and Soheyl Khalilpourazari , "Sine cosine crow search algorithm: a powerful hybrid meta heuristic for global optimization." arXiv preprint arXiv",

[44] A.E. Hassanien, R.M. Rizk-Allah, and M. Elhoseny, "A hybrid crow search algorithm based on rough searching scheme for solving engineering optimization problems", *J. Ambient Intell. Humaniz. Comput.,* vol. 0, no. 0, pp. 1-25, 2018.
[http://dx.doi.org/10.1007/s12652-018-0924-y]

[45] K.W. Huang, A.S. Girsang, Z.X. Wu, and Y.W. Chuang, "A hybrid crow search algorithm for solving permutation flow shop scheduling problems", *Appl. Sci. (Basel),* vol. 9, no. 7, p. 1353, 2019.
[http://dx.doi.org/10.3390/app9071353]

[46] Z.X. Wu, K.W. Huang, and A.S. Girsang, "A Whole Crow Search Algorithm for Solving Data Clustering", *Proc. - 2018 Conf. Technol. Appl. Artif. Intell. TAAI,* 2018pp. 152-155
[http://dx.doi.org/10.1109/TAAI.2018.00040]

[47] M. Allaoui, B. Ahiod, and M. El Yafrani, "A hybrid crow search algorithm for solving the DNA fragment assembly problem", *Expert Syst. Appl.,* vol. 102, pp. 44-56, 2018.
[http://dx.doi.org/10.1016/j.eswa.2018.02.018]

[48] N.S. More, and R.B. Ingle, ""“Energy-Aware vm migration using dragonfly-crow optimization and support vector regression model in cloud,” Int. J. Model. Simulation", *Sci. Comput.,* vol. 9, no. 6, pp. 1-24, 2018.
[http://dx.doi.org/10.1142/S1793962318500502]

[49] R.S. Chithra, and P. Jagatheeswari, "Fractional crow search-based support vector neural network for patient classification and severity analysis of tuberculosis", *IET Image Process.,* vol. 13, no. 1, pp. 108-117, 2019.
[http://dx.doi.org/10.1049/iet-ipr.2018.5825]

[50] I. Ali, M. S. Khan, and H. A. Sadiq, *Home Energy Management Based on Harmony Search Algorithm and Crow Search Algorithm,* .
[http://dx.doi.org/10.1007/978-3-319-65521-5_19]

[51] John , Nimmol P. , and V. R. Bindu , "Energy-Efficient Hybrid Firefly–Crow Optimization Algorithm for VM Consolidation", *International Conference on Intelligent Computing and Communication,* 2019pp. 413-427

[52] S. Kumar, A. Malleswaran, and B. Kasireddi, "An Efficient Task Scheduling Method In A Cloud Computing Environment Using Firefly Crow Search Algorithm", *International Journal of Scientific & Technology Research,* vol. 8, no. 12, 2019.

[53] A. George, and A. Sumathi, "Dyadic product and crow lion algorithm based coefficient generation for privacy protection on cloud", *Cluster Comput.,* vol. 22, no. 1, pp. 1277-1288, 2019.
[http://dx.doi.org/10.1007/s10586-017-1589-6]

[54] M. Optimization, H. Nobahari, and A. Bighashdel, "MOCSA : A Multi-Objective Crow Search Algorithm", *2017 2nd Conf. Swarm Intell. Evol. Comput.,* 2017pp. 60-65
[http://dx.doi.org/10.1109/CSIEC.2017.7940171]

[55] Z. Movahediyan, and A. Askarzadeh, "Multi-objective optimization framework of a photovoltaic-diesel generator hybrid energy system considering operating reserve", *Sustain Cities Soc.,* vol. 41, pp. 1-12, 2018.

[http://dx.doi.org/10.1016/j.scs.2018.05.002]

[56] Rodrigo Thom De Souza, Leandro Coelho, Camila Macedo, and Juliano Pierezan, A v-shaped binary crow search algorithm for feature selection. pp. 1-8, 07 2018.

[57] R.C.T. De Souza, L.D.S. Coelho, C.A. De MacEdo, and J. Pierezan, "A V-Shaped Binary Crow Search Algorithm for Feature Selection", *2018 IEEE Congr. Evol. Comput. CEC 2018 - Proc.,* pp. 1-8, 2018.
[http://dx.doi.org/10.1109/CEC.2018.8477975]

[58] Y. Meraihi, M. Mahseur, and D. Acheli, "A Modified Binary Crow Search Algorithm for Solving the Graph Coloring Problem", *Int. J. Appl. Evol. Comput.,* vol. 11, no. 2, pp. 28-46, 2020.
[http://dx.doi.org/10.4018/IJAEC.2020040103]

[59] J. John, and P. Rodrigues, "MOTCO: Multi-objective Taylor Crow optimization algorithm for cluster head selection in energy aware wireless sensor network", *Mob. Netw. Appl.,* vol. 24, no. 5, pp. 1509-1525, 2019.
[http://dx.doi.org/10.1007/s11036-019-01271-1]

[60] N.A. Azezan, H. Masran, and M.F. Ramli, "Preliminary design of crow search metaheuristics algorithm for travelling salesman problem", *AIP Conf. Proc.,* vol. 2138, 2019.040004
[http://dx.doi.org/10.1063/1.5121083]

[61] N. Siswanto, A. N. Adianto, H. A. Prawira, and A. Rusdiansyah, *A crow search algorithm for aircraft maintenance check problem and continuous airworthiness maintenance program, .*
[http://dx.doi.org/10.30656/jsmi.v3i2.1794]

[62] D.A. Zaki, H.M. Hasanien, N.H. El-Amary, and A.Y. Abdelaziz, "Crow search algorithm for improving the performance of an inverter-based distributed generation system",
[http://dx.doi.org/10.1109/MEPCON.2017.8301251]

[63] D. Liu, C. Liu, Q. Fu, T. Li, K.M. Imran, S. Cui, and F.M. Abrar, "ELM evaluation model of regional groundwater quality based on the crow search algorithm", *Ecol. Indic.,* vol. 81, pp. 302-314, 2017.
[http://dx.doi.org/10.1016/j.ecolind.2017.06.009]

[64] K.R. Prasanna Kumar, and K. Kousalya, "Amelioration of task scheduling in cloud computing using crow search algorithm", *Neural Comput. Appl.,* vol. 32, no. 10, pp. 5901-5907, 2020.
[http://dx.doi.org/10.1007/s00521-019-04067-2]

[65] K.D. Shinde, K.A. Kumar, D.S. Rashmi, R.S. Rukhsar, H.R. Shilpa, and C.R. Vidyashree, *Soft Computing Systems.* vol. Vol. 837. Springer Singapore, 2018.

[66] A. Meddeb, N. Amor, M. Abbes, and S. Chebbi, "A novel approach based on crow search algorithm for solving reactive power dispatch problem", *Energies,* vol. 11, no. 12, p. 3321, 2018.
[http://dx.doi.org/10.3390/en11123321]

[67] S.C. S R, and H. Rajaguru, "Lung cancer detection using probabilistic neural network with modified crow-search algorithm", *Asian Pac. J. Cancer Prev.,* vol. 20, no. 7, pp. 2159-2166, 2019.
[http://dx.doi.org/10.31557/APJCP.2019.20.7.2159] [PMID: 31350980]

[68] D. Oliva, *PT US CR, 2017*.
[http://dx.doi.org/10.1016/j.eswa.2017.02.042]

[69] A. Lenin Fred, S.N. Kumar, P. Padmanaban, B. Gulyas, and H. Ajay Kumar, "Fuzzy-Crow Search Optimization for Medical Image Segmentation", *Studies in Computational Intelligence,* vol. 890, pp. 413-439, 2020.
[http://dx.doi.org/10.1007/978-3-030-40977-7_18]

[70] N.A. Priyanka, and G. Kavitha, Detection of Dementia from Brain Tissues Variation in MR Images Using Minimum Cross-Entropy Based Crow Search Algorithm and Structure Tensor Features.*Soft Computing for Problem Solving.* Springer: Singapore, 2019, pp. 377-390.

CHAPTER 8

The Quantitative and Qualitative Assessment of Re-Search Conducted Using Computational Intelligence for the Diagnosis or Treatment of COVID-19

Mallikarjun Kappi[1,*], Madhu S.[2], Balabhim Sankrappa Biradar[3] and **B.U. Kannappanavar[4]**

[1] Department of Library and Information Science, Kuvempu University, Shankaraghatta, Karnataka, India

[2] Department of Studies in Library and Information Science, Kuvempu University, Shankarghatta, Shimoga Dist, Karnataka 577451, India

[3] Department of Library and Information Science, Kuvempu University, Shankaraghatta, Karnataka, India

[4] Sahayadri College, Shivamogga, Karnataka, India

Abstract: The effect of the COVID-19 pandemic has prompted a large number of studies targeted at understanding, monitoring, and containing the disease. However, it is still unclear whether the studies performed so far have filled existing knowledge gaps. We used computational intelligence (CI)/Machine Learning (ML) technologies and alliance areas to analyse this massive amount of information at scale. This chapter assesses the scholarly progress and prominent research domains in the use of CI/ML technologies in COVID-19 research, focusing on the specific literature on computational intelligence and related fields that have been employed for "diagnosis and treatment" of COVID-19 patients.The "Web of Science" database was used to retrieve all existing and highly cited papers published up to November 2020. Based on bibliometric indicators, a search query ("Computational Intelligence or Neural Networks or Fuzzy Systems or Evolutionary Computation & Diagnosis or Treatment & Coronavirus or Corona Virus or COVID-19") was used to retrieve the data sets. The growth of research publications, elements of research activities, publication patterns, and research focus tendencies were computed using 'Biblioshiny' software and data visualization software 'VOS viewer.' Further, bibliometric/scientometrics techniques were incorporated to know the most productive countries, most preferred sources & their impact, three-field plot, and the most cited papers. This analysis provides a comprehensive overview of the "COVID-19" and CI-related research, helping researchers, policymakers, and practitioners better understand COVID-19 related CI

[*] **Corresponding author Mallikarjun Kappi:** Department of Library and Information Science, Kuvempu University, Shankaraghatta, Karnataka, India; Tel: +917829283101; E-mail: mkmallikarjun@gmail.com

Om Praksh Jena, Alok Ranjan Tripathy, Brojo Kishore Mishra and Ahmed A. Elngar (Eds.)

research and its possible practical impact. Future CI / ML Studies should be committed to filling the gap between CI / ML research.

Keywords: Computational intelligence, Bibliometric study, China, Computational modelling, Corona virus, Coronavirus, COVID-19, Diagnosis, Diagnosis tools, Evolutionary computation, Fuzzy sets, Fuzzy systems, India, Machine learning, Neural networks, Pandemic, Scientometrics, Treatment, Visualization, Web of science.

8.1. INTRODUCTION, BACKGROUND, AND OVERVIEW

Because this infectious disorder (virus) used to be first recognized in the year 2019 December, it has emerged as a universal epidemic and takes induced infections in lots of people. As of July 2020, the 'coronavirus' dying toll exceeded 6,87,000 global, and the variety of infections and deaths continues to increase. The weekly decline trend of Europe and Southeast Asia has not been interrupted iduring the COVID-19 pandemic and new cases and deaths in the region, the largest contribution of the Americas region continue to be the same. However, when the number of cases is relatively low, new cases (15%) and deaths (15%) in Africa this month represent the highest prosperity. Eastern Mediterranean and Western Pacific also noted that new cases and deaths increase this month. Such a serious condition is directed to increased risk in global health care systems and immense harm in the world economic system [1]. To fight 'COVID-19', various nations remain working to nourish modern; and positive apparatuses to overaw this disaster. Managers, leaders of the enterprise, and scholars alike are dedicating ample sources and effort to reducing the consequences of this disease. A few months ago, several kinds of research and solutions for the fight towards 'COVID-19' were developed and utilized. For instance, speedy screening strategies, the use of various kinds of clinical information, along with 'X-rays,' 'Computed Tomography' (CT) scans, and necessary signs, have permitted a suitable diagnosis and virus observing. Using social media data, computer structures are being designed for risk profiling, affected person investigation, contact tracing, or propagation modelling.

The present research proposes that Computational Intelligence processes can also gain a comprehensive overview of human specialists in certain medical image diagnosis tasks, consisting of lung diseases [2 - 4]. Compared to other lung diseases, such as lung nodule detection [5, 6], tuberculosis diagnosis, and lung cancer screening [7] (Ardila *et al.*, 2019), isolating COVID-19 from special pneumonia has unique difficulty, *i.e.*, the high similarity of pneumonia of various kinds and massive variants in specific phases of the identical type. Therefore, an

emerging CI diagnosis algorithm particular to 'COVID-19' is necessary. The CI diagnosis algorithm also has the benefits of high efficiency, high repeatability, and effortless large-scale deployment.

For now, the combinations of computational intelligence mechanisms with numerous strategies and systems under numerous application circumstances may require exclusive types of computational intelligence techniques, consisting of 'records analytics ',computational modeling', high-speed computing, 'artificial intelligence'; and specifically it's the subfield of 'machine learning', several scientists have dedicated their efforts to growing structures of 'Computational Intelligence', especially for the fight towards 'COVID-19'. By mid-November 2020, over 118,883 scholarly articles have been published about COVID-19, SARS-CoV-2, nCOV-19, and other associated coronaviruses [8]. But these scholarly articles didn't study the significant issues in applying computational intelligence in detail to wrestle the 'COVID-19' pandemic. Hence, it would be crushing to refine and review research related to "Computational Intelligence" from such a massive number of articles. To consider the above perceptions, now is the time to scientifically categorize and analyse the present development of bibliometric study on CI.

8.2. RELATED STUDIES

Bibliometric analysis was performed by the faculty of different countries from Taiwan, South Africa, Nigeria, USA, and Malaysia [9] from a few disciplines to know approaches applied by machine learning in controlling coronavirus. A good volume of literature was referred to see the background work against the Coronavirus and machine learning applications. The different academic database was used for data collection (DBLP, ACM Digital Library, IEEExplore, Science Direct, Springer Link, PubMed, Scopus, and Web of Science) which stores prominent and peer-reviewed journal citations, abstracts, and publications. The keywords included for the study were *"deep learning and COVID-19", "convolutional neural networks and COVID-19", "artificial neural networks and COVID-19", "machine learning and COVID-19", "decision tree and COVID-19", "COVID-19 and diagnosis tool", "COVID-19 and decision support system"*. Each topic briefly discussed techniques and mechanized by summarizing the algorithm, performances, contribution, and benefit for controlling COVID-19. For analysis, 30 documents were selected for the study purely on machine learning applications; the top publication was performed on the implementation of CT Scans. Most of these algorithms were developed using convolutional neural networks on COVID-19 in majority of the publications published in 2020. Collaborative works dominated in machine learning, and the top productive country was China followed by the USA; the top coupling department was

Guangzhou Centre for Disease Control in China and lastly, the top citation came from Journal of Microbes, and the top cited reference came from Wuhan University and the Department of Microbiology of the University of Hong Kong.

Based on emerging technologies on computational intelligence, a survey conducted different techniques for combatting COVID-19 [10]. The paper discussed various categories of computational intelligence and related studies which applied for combatting the COVID-19 virus by using *"Neural Networks "Fuzzy Logic," "Evolutionary Computation," "Computational Learning Theory," "Probabilistic Methods"* and further a complete observation was made to know various issues which can be traced on combatting the COVID-19 through applications of the computational intelligence techniques by categorizing five major issues: *"Tracking and Predicting Virus Propagation (TPVP)," "Characterization of Symptoms of Virus Infection(CSVI)," "Treatment Design (TrD)," and "Precaution Development (PD)."*

Important research conducted on "COVID-19" or "Novel coronavirus disease 19" pandemic through analysis of the global literature on COVID-19 aspects [11] referred all major citation & abstracting database Scopus, Pubmed & Web of Science (WoS) and search terms were used according to WHO naming process virus, disease, and causes. The major investigation used only the top 10 documents that were the most cited papers according to the citation count, which was reflected in the citation database. Studying the co-occurrence of keywords and most used frequent terms in respective papers and presented in VOS viewer format, further employed *"Latent Dirichlet allocation (LDA)"* to find the latest topics from the title and its abstract of top documents.

A systematic review was conducted by [12] to know the 'Artificial Intelligence' techniques for finding and classifying "COVID-19" clinical imageries. A comprehensive study performed and selected the benchmarking trends by analysing the retrieved literature using a reliable database, namely IEEExplore; Web of Science (WoS); PubMed; Science-Direct and Scopus on "*Artificial Intelligence*" and "*COVID-19*". After the screening of the overall document, 36 related studies were obtained and short-listed and only 11 articles met the standard criteria for the study. Topics works were published in the Science Direct database and the highest number of publication contributions came from Turkey. Taxonomy analysis was performed on selected articles and categorized by two types (Review and Research studies) on medical images using Artificial Intelligence on 'COVID-19'. Each article conducted deep analysis & critically reviewed and focused on the obstacle and research slots in selected literature on the respective subject. The literature results in the appropriate bench-making techniques evaluated based on classification tasks (binary; multi-class; multi-

labeled; and hierarchical classifications) on 'COVID-19 medical images. Based on the classification of AI techniques on *"COVID-19 medical images,"* the multi-complex attribute problem is more complex, so adopting the *"multi-criteria decision analysis (MCDA)"* suit more has a crucial and productive advanced technique to tackle the complex problem.

A theoretical overview of the Internet of things (IoT) by [13] discussed the research from physical science and engineering contributions of eminent researchers who took up such challenges to study problems and build new theories to explain the user-friendly solutions to the modern problems edify civilian and themselves. The study aims to provide consciousness about new emerging technology and evident implementation for the "*COVID-19*" pandemic. Presently IoT is used in employing the service for the health sector "Internet of Healthcare things (IoHT)" or "Internet of medical things (IoHT)." IoT in the health sector uses a proper monitoring system for patients with high risks of COVID-19; it makes it easy to track and monitor using an internet-based networking system. It's used primarily in bio-metric measurement by analysing the blood pressure, heartbeat, and glucose level. The chief merit of using IoT in the health sector is that it improves the efficiency of medical staff by reducing the workload and capture the real-time data for health professional with the COVID-19 infected patient by superior treatment, enhanced diagnosis, effective control, lesser expenses, and reduced chances of mistakes through the remote location by the virtual management system. Some of the challenges and issues of IoT applications on COVID-19 are Safety, Security, and Privacy.

A bibliometric analysis was conducted by [14] to know the research trends that emerged using machine learning for COVID-19. The data was retrieved using the Scopus database which resulted in 1883 selected citations for the study. For statistical analysis biblioshiny software was used and VOS viewer software to visualize the data. The analysed data resulted in China having the highest productivity with par rest of the world. Due to the pandemic, collaboration works were highly dominated, and the journal 'The Lancet' got the most citations to count.

To find out the explosive growth of the big data and its research trends [15] bibliometrics approach by applying different methodologies, first in-depth reviewed the literature related to big data, further interviewed with subject experts, prepared online questionnaires, and surveyed which resulted in knowing the core areas of big data and framed proper key terms and searched in the web of science (WoS) database for retrieval of citation data. The searched query resulted in 6572 citations on big data and related terms from late 1980-2015. ANOVA and T-test involved proving the hypothesis statistically and relational among the

variables. Research conducted to know publications outcome on the artificial intelligence of when confined to India research outcome, using scientometrics assessment between 2007-16 by [16] total publication accounted 9730 from Scopus database, registered yearly growth average of 24.45% and 2.76% citation per paper averaged. Anna University, Chennai contributed the highest of 294 publications, and S Das topped the highest publication of 36 publications from Jadavpur University. The top communication channel came from the applied soft computing journal of 84 manuscripts.

Big Data is solely the enormous large volume of data which is very multiplex to analyse and a tough task to manage documentation or preserve with existing conventional data processing applications or tools. The idea of 'Five V' generally explains the concept of big data, *i.e*., "Volume, Variety, Velocity, Value, and Variability." Big data intervention in the agricultural field includes "meteorological data, survey data, financial data, soil, water, geospatial based data, external market data (price and sales data), open government data, and social media-based data." Aside from this technical importance, big data can be used for precise and timely crop yield forecasting, reducing the farmer's risk in production, food safety measures, and agriculture equipment management. The IoT concept intercession in agriculture was defined as "Internet of Agriculture Things (IoAgT)." The majorly big data concept implemented from 2013 onwards as the research progressed in agriculture using IoT with a combination of Big data. The present paper focused on the scientometrics and visualization study of the "global Agriculture Big Data (ABD)" investigation performed by [17] however, as per the evaluation is concerned, limited research has been conducted in the field of ABD. The study aims to cover the present status of 'Agriculture Big Data' research through network analysis and visualization of the ABD research publications. A total of 379 publication data were recorded from the Clarivate Analytics 'Web of Science' database, including research performed from "all years." VOSviewer, MS-Excel, and R statistical software are handed-down for data analysis. Due to the various document types (Article, Review, Proceedings, Editorial Materials, and Book Chapters). The study focused on only research articles of 330 (87.7), which is highest when par with other document types. The United States took the top-quality work with a total of 1132 citations from 76 highest papers with (14.90) of average article citation till 2019, followed by China with 530 citations using 49 papers with an average of (10.82). Colorado state university, situated in the USA, has topped 13 papers. Pate R; Klise G and Wu B. entitled *"Resource Demand Implications for us Algae Biofuels Production Scale-Up"* communicated in 'applied energy' journal in 2011 with the citation count of 168 times. One of the best procedures for achieving artificial intelligence is using neural network algorithms. Deep learning is one of the specialized forms of machine learning research. Generally, it is based on how the human brain operates, extracts

information, and learns. In deep learning, neural networks use pattern recognition and classify tasks based on texts, images, and audio or sound. The present study focused on the deep learning concept to know the quantity and quality of deep learning research conducted using the scientometrics assessment using Scopus index documents with a coverage period of 14 years from 2004-2017 by [18] to know the global outcome of 10027 documents. The largest share came to contribution to China with 2633 (29.25%) followed by the USA with 2653(26.46%). Compared to the overall subject contribution, computer science stream topped the highest of 7707 (76.86%), a core domain learning subject. (Tsinghua University, China) contributed the highest number of research towards deep learning 197 publications, but when it comes to top productive author Prof. Mark Nathan Billinghurst from (University of Canterbury, New Zealand) topped the highest publication of 204, and the top profile author was Y. Bengio from (University of Montreal, Canada) with publication 30 top-cited by 10026 publications. Nero computing was the most used communication channel for publication research work with 91 publications.

"Artificial Intelligence" is a branch of modern computer science, where AI performs on the basic human thought process, function, and characteristics. An elaborative study was conducted [19] to investigate the "Artificial Intelligence AI" from 1968 to 2014 using Scopus has listed citation only core collection of Indian publication output using Scientometrics studies to measuring the scientific activates to know the growth and nature of literature related AI. The study used Normative Scientometrics to explore standards, instructions, and heuristics to understand the areas that have covered the broad areas of research and its core principles in the respective field or subject [20] with descriptive studies concrete only on individual works. Based on the download citations of 6,529 papers from the Scopus database when confined to India. The majority of publications emerged in the conference proceeding of 3,846 (58.9%), which indicates that AI is an emerging specialized field for Indian researchers. This highlights that researchers like more conferences rather than submitting manuscripts for journal publication, where a conference allows cross-discussion with a fellow professional with the presentation, which helps in the refinement of newer ideas in exchange. Keyword analysis was made to understandthe important topics and related aspects of any particular subject. Based on keyword analysis highest of 5,806 papers recorded has "Artificial Intelligence" leading "Algorithms with 1,233 papers, followed by "Neural Networks" with 522 papers and "Optimization" with 494 papers during the prescribed years. Anna University Chennai topped the list in the paper contribution in India with a total of 239 papers; Swagtan Das made the highest contribution of 37 papers with the total publication of 135 from ISI Kolkata; top collaboration came between India and the USA with 286 papers.

8.3. OVERVIEW OF COMPUTATIONAL INTELLIGENCE

Computational Intelligence (CI) practice has been effectively combined into multiple systems to deal with the basic challenges of mass epidemic viruses. Before introducing the use of computational intelligence to solve the unique problems of COVID-19 operations, we need to understand the history and different categories of this method (Fig. **1**). Based on the idea of calculating intelligence, we can clarify which issues can be handled when using COVID-19 using CI.

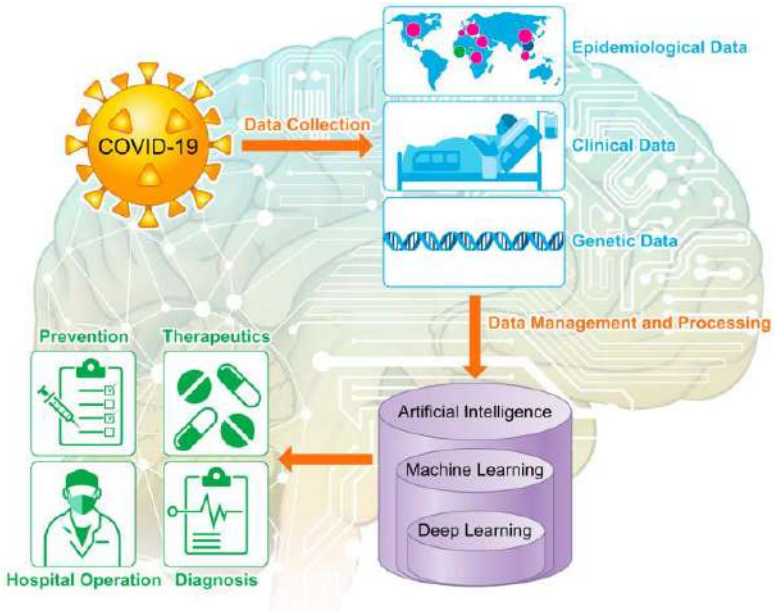

Fig. (1). Implementation of Computational intelligence (CI) in the fight against COVID-19.

8.3.1. What is Computational Intelligence (CI)

Computational Intelligence (CI) is the concept, design, practicality, and improvement of computational paradigms influenced by biology and language. Traditionally, the three pillars of CI are neural networks, fuzzy systems, and evolutionary calculations. However, over time, many natural factors have stimulated the development of calculation standards. Therefore, CI is a developing area and added value; three main elements include computational models such as environmental intelligence, artificial life, cultural learning, artificial endocrine network, social reasoning, and artificial hormone network. CI plays a major role in creating a successful intellectual system, game, and cognitive development system. In the past few years, research on deep learning has been surprising in a particular depth convolutional neural network. Today,

deep learning has become the core technology of artificial intelligence. Some of the most successful AI systems are CI.

8.3.2. Types of Computational Intelligence

Computational intelligence is a branch of "artificial intelligence" defined by Bezdek [21]. The fields of artificial intelligence and computational intelligence remain the same; this is to understand general intelligence. Marks [22] defined the difference between artificial intelligence and computational intelligence by claiming that the former is made from tough computing technologies, the latter is made of soft computing techniques. Hence, we can anticipate that two types of machine intelligence are: a) artificial intelligence, which is developed by the idea of hard computing; b) computational intelligence, which is developed with the aid of softcomputing. Similar to hard-computing-based artificial intelligence, computational intelligence can adapt to many different conditions through the advantages of soft computing. Hard-computing strategies are designed with a Boolean logic established simply on true or false tenets that information engineering relies upon. One serious issue in Boolean logic is that Boolean values are unable to take natural language, certainly. Though, based on fuzzy logic, soft computing strategies can deal with tentative cases. This kind of logic is one exclusive feature of computational intelligence, and through combining facts into confined facts, it estimates how the human mind acts (Fig. **2**).

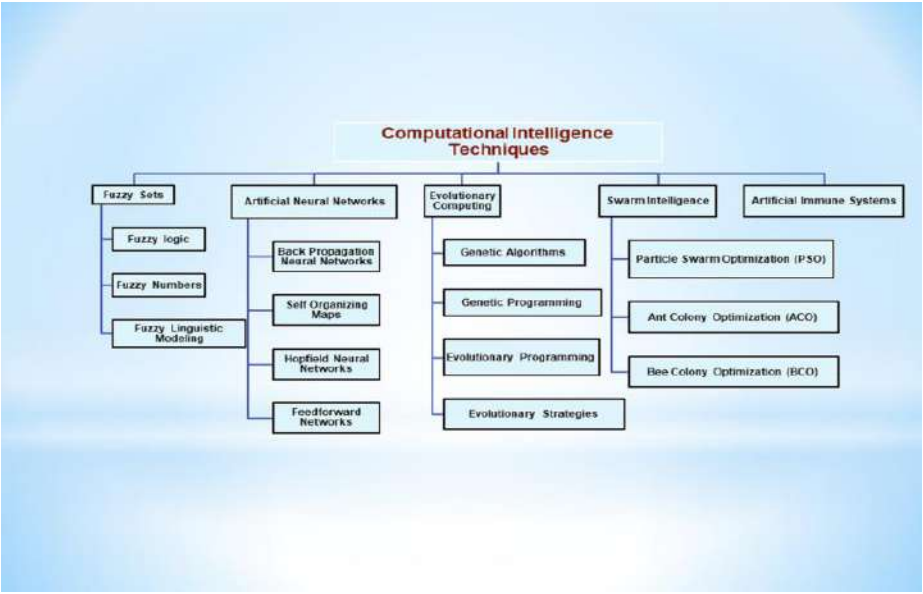

Fig. (2). Computational intelligence techniques.

8.3.2.1. Fuzzy (Logic) Sets

L. A. Zadeh has proposed the concept of fuzzy sets [23]. Since this seminal work, many researchers have significantly contributed to the growth of fuzzy sets theory and its applications, resulting in great success from the theoretical and scientific points of view. Fuzzy sets theory offers to deal with unclear limitations, representing vague thoughts and working with linguistic variables. Fuzzy logic is the selected soft computing method for executing the proposed 'COVID-19' diagnosis system based on the following reasons;

- fuzzy algorithms are frequently strong, the reasoning method is often simple, so computing power is saved,
- Fuzzy systems generally have a quicker growth time than conventional methods. This is a very exciting feature, exclusively in actual time systems such as online diagnose applications,
- Fuzzy logic is flexible and easy to gadget machine learning techniques,
- It is a very convenient technique for uncertain or approximate reasoning.

However, fuzzy logic suffers from difficulty finding suitable membership values for fuzzy systems. It also suffers from difficulty storing the rule-base that might require a substantial amount of memory. Moreover, fuzzy logic must be made with the full supervision of experts [24, 25].

Quick and precise detection of COVID-19 is increasingly vital to prevent the sources of infection as well as helping patients to prevent disease progression. Soft Computing (SC) techniques, such as; fuzzy logic, neural networks, and genetics have proven as potential tools in disease detection [26, 27]. They can support decision-making only for immediate isolation and suitable patient treatment [28]. Several techniques have been proposed for detecting COVID-19 infections. However, hopefully, detection accuracy has not been reached yet. Fuzzy Logic (FL) describes systems in terms of a mix of numeric and symbolic [24, 25]. This has advantages over pure mathematical (numerical) approaches or pure symbolic approaches because system knowledge is often available in such a combination.

8.3.2.2. Artificial Neural Network

An artificial neural network (ANN) is the piece of a computing tool designed to simulate the way the human brain analyses and processes data. It is the substance of artificial intelligence (AI) and solves problems that would prove impossible or tough by human or statistical standards. ANN's have self-learning abilities that allow them to produce higher results as greater records become available.

Artificial neural networks can be used for regression or classification modelling for prediction and automatic control. A large number of simulation data uses limited data sets. This structure is the foundation of deep learning, which is good at representation learning. Accordingly, artificial neural networks process and learn information from data via distributed information processing systems. One of the crucial properties of artificial neural networks is fault tolerance, which is approximately modelled on how the human brain operates. Neural networks have been widely applied to data analytics, clustering, classification, and automatic control engineering based on these characteristics. In real-world applications, such methods aim to analyse and classify medical data, recognize human faces, detect computer fraud, and deal with the nonlinearity of a system for better process control.

8.3.2.3. Evolutionary Computing (EC)

Evolutionary computation (EC) is a global optimization technique inspired by biological evolution and the subfield of artificial intelligence and soft computing studying these algorithms [29]. EC systems solve problems by inhabitants, error and success, meta-heuristics, or stochastic optimization. An initial set of candidate solutions is generated and updated iteratively, such as removing less-desired solutions and including the noise. A population of keys is subject to natural selection or artificial selection and alteration and hence progresses and adapts—*i.e.*, raises fitness. EC is popular in computational intelligence as its outcomes in near-optimal solutions in a comprehensive range of frameworks [30] where there are numerous options and additions for specific data structures and problems.

8.3.2.4. Swarm Intelligence (SI)

COVID-19 pandemic is a dynamical complex system with swarm intelligence, and swarm intelligence (SI) is a part of the computational intelligence. It is collective behaviour exhibited by entities, mainly animals. Swarm intelligence is the collective behaviour of decentralized, self-organized systems. The idea is employed in work on artificial intelligence.

8.3.2.5. Artificial Immune Systems (AIS)

Artificial Immune Systems (AIS) are a class of computationally intelligent, rule-based machine learning systems inspired by the principles and processes of the vertebrate immune system. The artificial immune systems (AIS) area has grown dramatically in the last 10 years. AIS algorithms are computer programs modeled on different aspects of the human immune system and used to tackle various problems, from computer virus detection to data mining to robot control.

8.4. COVID-19 DIAGNOSIS TOOLS

Presently, the responsiveness for 'reverse transcription-polymerase chain reaction (RT–PCR) based epidemiologic nucleic acid tests are used as the reference normal method to check COVID-19 infection [31]. Still, that research laboratory test is time-intensive, and the supply of test kits may be restricted access for a rapidly increasing suspicious population even for many developed countries such as the US, UK, USSR, Germany, *etc*... Most significantly, early false-negative or weakly positive RT–PCR test results were found in several later-confirmed cases, whereas extremely suspicious 'Computed Tomography' (CT) imaging features were extant [32, 33]. The treatment and screening of COVID-19 can be more effective when the deep learning method, CT features, and real-time RT–PCR results are integrated [34]. AI and deep learning can help in develop diagnostic tools and decide on treatment [35, 36]. Consequently, various diagnostic tools were developed based on the machine learning algorithm to fight COVID-19. For example, Apostolopoulos and Mpesiana (2020) applied transmission learning with CNN to identify COVID-19 from X-ray images covering collective bacterial pneumonia and normal incidents, and well-known COVID-19 infection. Transfer learning CNN was used to diagnose COVID-19 cases from X-ray datasets. The results indicated that VGG19 diagnosed COVID-19 confirmed cases with better accuracy on two- and three-classification problems than MobileNet v2, Inception, Xception, and Inception ResNet v2. The proposed approach can help develop a cost-effective, fast, and automatic COVID-19 diagnostic tool and reduce the exposure of medical workers to COVID-19. Similarly [37], developed an automated 'Computer-Aided Diagnosis' (CAD) system for the detection of COVID-19 samples from healthy cases and cases with pneumonia via chest X-ray (CXR) images. Their study demonstrated the effectiveness of applying deep transfer learning techniques to identify COVID-19 cases with CXR images.

8.5. DATA SOURCE AND METHODOLOGY

According to the sources of data, the status of the Computational Intelligence for Diagnosis or Treatment of COVID-19 domain can be described, including the growth of publications, document types of publications, preferred journals, Most prolific countries and institutions, highly prolific authors, and their affiliations, co-occurrence network of categories and most preferred keywords in this section.

Web of Science (WOS) is the source database for our bibliometric analysis. This database provides access to over 75 million scientific works of literature from more than 5000 publishers around the world. It also provides a profile of more than 16 million authors and 70,000 institutions [38]. We retrieved the publications from the Web of Science database using a set of keywords related to

Computational intelligence, Diagnosis, and COVID-19 (Table **1**). A total of 61 papers were retrieved to meet our study. The database search happened on 17th November 2020. Later extracting the available literature, we studied the titles and abstracts of those publications to evaluate their eligibility for bibliometric analysis. Publications were excluded if they did not refer to any CI-related keywords or did not focus on COVID-19. Due to a limited number of publications, articles irrespective of their study design were included in this study with no language restrictions. The study period was restricted to 2016 to 2020, whereas the COVID-19 epidemic started in December 2019; hence all the articles related to our search query were published in 2020.

Table 1. Sample Data Description.

Description	Results
Documents (NP)	61
Preferred Sources (Journals, Books, *etc*)	40
Keywords Plus (ID)	117
Author's Keywords (DE)	241
Timespan	2020:2020
Average citations per documents	1.852
Authors	371
Author Appearances	384
Authors of single-authored documents	1
Authors of multi-authored documents	370
Single authored documents	1
Documents per Author	0.164
Authors per Document	6.08
Co-Authors per Documents	6.3
Collaboration Index	6.17

Search Query.

Sl No	Search Query	Results
# 1	TS=(Computational Intelligence or Neural Networks or Fuzzy Systems or Evolutionary Computation) Indexes=SCI-EXPANDED, SSCI, A&HCI Timespan=2016-2020	137,313
# 2	TS=(Diagnosis or Treatment) Indexes=SCI-EXPANDED, SSCI, A&HCI Timespan=2016-2020	1,637,172
# 3	TS=(Coronavirus or Corona Virus or COVID-19) Indexes=SCI-EXPANDED, SSCI, A&HCI Timespan=2016-2020	49,835

Sl No	Search Query	Results
# 4	#3 AND #2 AND #1 Indexes=SCI-EXPANDED, SSCI, A&HCI Timespan=2016-2020	61

In this study, we used the 'Bibliometrix' package [39] (http://www.bibliometrix.org) developed by [40] Aria & Cuccurullo (2017) in R (an open-source statistical application) to perform the analysis. Bibliometrix package is well-known for its broad features and is used in a growing number of publications [41, 42].

Visualization software called VOSviewer was used to current a bibliometric study on Computational intelligence, Diagnosis, and COVID-19. VOSviewer software is a tool for creating and visualizing bibliometric maps, such as journals, research, or individual publications. These networks can be created based on citations, co-citation, bibliographic coupling, or co-authorship relations [43]. This VOSviewer software also offers text mining functionality that can be used to construct and visualize co-occurrence networks of important terms extracted from a body of scientific literature (www.vosviewer.com). We only used 49,835 publications with the keyword "Coronavirus or Corona Virus or COVID-19" (# 3), 1,637,172 publications with the keyword "Diagnosis or Treatment" (# 2) and 137, 313 publications with the keyword "Computational Intelligence or Neural Networks or Fuzzy Systems or Evolutionary Computation" (# 1) that were retrieved from Web of Science database for the bibliometric analysis presented in this study. Only 61 document results were extracted using the keyword set # 4 (#3 AND #2 AND #1) from the web of science database. We were not considered papers from other sources since most of these publications have not been peer-reviewed and are not available online in the form of preprint publications. Excel 2019 was used to preprocess the data of the clustering table, to draw the geographic density distribution maps.

8.6. RESULTS ANALYSIS AND DISCUSSION

As we are involved in the chronological nature of research and publications, we have considered the key phrases like 'Computational Intelligence' 'Neural Networks' 'Fuzzy Systems' 'Evolutionary Computation' 'Diagnosis' 'Treatment' 'Coronavirus' 'Corona Virus' and 'COVID-19' to get a more extensive coverage and to find out potential areas for future research on COVID-19. Besides, we have included the keywords like 'SARS,' 'MERS,' 'severe acute respiratory syndrome,' and 'Middle East Respiratory Syndrome' to represent the historical alliance with COVID-19. Moreover, we refine our query results with the 'SCI-EXPANDED, SSCI, A&HCI' category of WoS, which allows us to preserve the focus to the publications which consider different aspects. Table **1** reviews the search results:

8.6.1. Most Productive Countries

This section examines the territorial distribution of research publications considering countries. Table **2** displays the most Productive Countries. Based on the search query reflected on the web of science database, both India and China contributed 11 (18.033%) papers, each with the collaborative effort to combat the "COVID-19" virus by conducting research using different computational intelligence and related techniques followed by Canada, Egypt and USA contributed 7 (11.475%) each. One of the major impacts of India compared to the rest of the world's computational intelligence and alliance areas is the emerging field. Further, when it compares to average citation per article (ACPA), Greece citation impacted highest of (16.50%) with 2 papers, followed by Iran ACPA (16%) with 4 papers, Malaysia ACPA (16%) with 3 papers Singapore and Taiwan ACPA (14%) with one paper. Fig. (**3**) shows the countries' collaborations publications network.

Table 2. Most Productive Countries.

Countries	Records	% of 61	Citations	ACPA
India	11	18.033	14	1.27
Peoples R China	11	18.033	14	1.27
Canada	7	11.475	5	0.71
Egypt	7	11.475	0	0.00
USA	7	11.475	6	0.86
South Korea	6	9.836	9	1.50
Australia	5	8.197	3	0.60
England	5	8.197	4	0.80
Saudi Arabia	5	8.197	1	0.20
Turkey	5	8.197	5	1.00
Iran	4	6.557	16	4.00
Pakistan	4	6.557	3	0.75
Malaysia	3	4.918	16	5.33
Mexico	3	4.918	7	2.33
Spain	3	4.918	3	1.00
Bangladesh	2	3.279	1	0.50
Greece	2	3.279	33	16.50
Morocco	2	3.279	0	0.00
Qatar	2	3.279	1	0.50

(Table 2) cont.....

Countries	Records	% of 61	Citations	ACPA
Scotland	2	3.279	3	1.50
Czech Republic	1	1.639	2	2.00
Denmark	1	1.639	0	0.00
France	1	1.639	0	0.00
Iraq	1	1.639	1	1.00
Japan	1	1.639	0	0.00
Jordan	1	1.639	2	2.00
Netherlands	1	1.639	0	0.00
Norway	1	1.639	3	3.00
Singapore	1	1.639	14	14.00
Sudan	1	1.639	11	11.00
Switzerland	1	1.639	0	0.00
Taiwan	1	1.639	14	14.00
Tunisia	1	1.639	0	0.00
Uganda	1	1.639	0	0.00
Vietnam	1	1.639	0	0.00

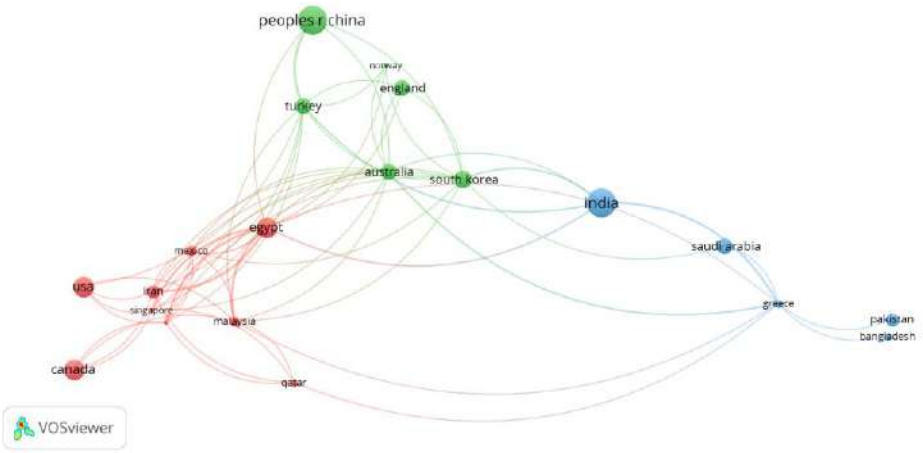

Fig. (3). Countries collaborations Publications Network.

8.6.2. Most Preferred Sources

Computational intelligence and alliance areas are major contributors in medical fields in the using (X-ray & CT-Scan) with the help of new datasets, different varieties of studies have been conducted during pre-covid period and post-covid

period, so to explore research outcome author prefer reputed Journal publication that has good impact factor and related to subject to choose for authors to submit their manuscripts, so the selection made for the preferred communication channel covered for the study results are IEEE Access published 9 scientific publications related to computational intelligence and allied subjects followed by Applied Intelligence 5 scientific publications, IEEE Transactions on Medical Imaging 4 scientific publications. However, the ranking modifies if we consider the TC and h-index. Though the *IEEE Transactions on Medical Imaging* tops the list having a total citation of 17 and an h-index of 3, the journal *Computers in Biology and Medicine* having (TC=14 and h-index 1) and *Biomed Research International* (TC=11 and h-index 1) total citations and h-index respectively with significantly fewer publications (NP=4 & 1) compared to the top journal in Table **3** and Fig. **4**. We observe an increasing publication trend for all the top journals in this area of research in the current years. Prominently, the growing trend has experienced a sharp rise in 2020. The trend shows an increased interest by the journals to publish relevant research articles, and authors can consider the chance to publish their modern scientific research outputs, especially on the current coronavirus outbreak-associated topics.

Table 3. Most Preferred Sources.

Source	NP	h-index	g-index	m-index	TC
IEEE Access	9	2	2	2	8
Applied Intelligence	5	0	0	0	0
IEEE Transactions on Medical Imaging	4	3	4	3	17
Applied Sciences-Basel	2	1	1	1	2
Computational and Mathematical Methods in Medicine	2	0	0	0	0
Computers in Biology and Medicine	2	1	2	1	14
PEERJ	2	0	0	0	0
Romanian Journal of Information Science and Technology	2	0	0	0	0
Soft Computing	2	1	1		1
Biomed Research International	1	1	1	1	11
Biomedical Engineering Online	1	0	0	0	0
Biomedical Signal Processing and Control	1	1	1	1	2
Chaos Solitons & Fractals	1	1	1	1	5
Childs Nervous System	1	0	0	0	0
Complex & Intelligent Systems	1	0	0	0	0
Computational Mechanics	1	1	1	1	1
eLife	1	1	1	1	2

(Table 3) cont.....

Source	NP	h-index	g-index	m-index	TC
Environmental Management	1	0	0	0	0
European Radiology	1	0	0	0	0
European Review for Medical and Pharmacological Sciences	1	1	1	1	5

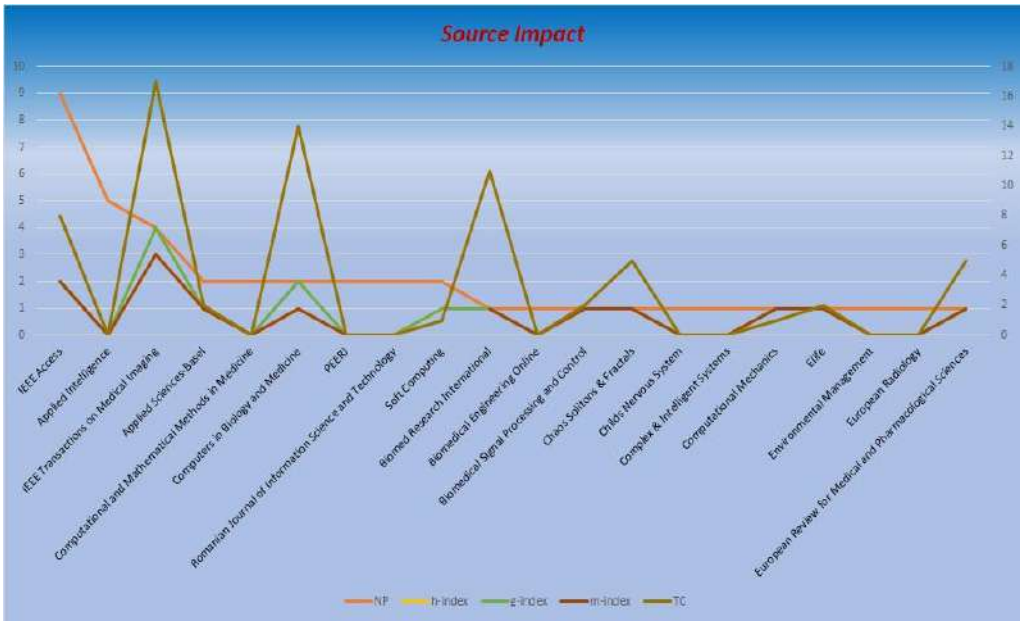

Fig. (4). Most Preferred Sources and Impact.

8.6.3. Highly Prolific Institutions

Table **4** shows the most productive institutions in terms of total research publications during our study period. All over the world, different universities and research institutions are trying their level best to find a solution against the covid virus for its treatment and vaccine with an enormous of funding has provided by institutions. At the institutional level, we notice a significant dominance of China institutions. Based on research productivity on computational intelligence and allied areas, the top publications came from the *Huazhong University of Science Technology* with 3 publications with 8 times cited, followed by the *University of Toronto* 3 publications with 5 times cited and the rest of the institutions published 2 papers each. The *University of Patras* placed top in the list with 33 citations. As shown in Fig. (**5**), Peoples, Republic of China dominates in a collaborative effort with (India, Morocco, Mexico, England, Egypt, and Japan) and the second-highest followed by India with collaboration with (China, Mexico, Morocco, Norway, England, and Japan).

Table 4. Highly prolific institutions.

Institutions	Records	% of 61	Citations
Huazhong University of Science Technology	3	4.918	8
University of Toronto	3	4.918	5
Arab Academy for Science Technology Maritime Transport	2	3.279	0
Hubei Prov Key Lab Mol Imaging	2	3.279	4
Iran University of Medical Sciences	2	3.279	0
King Abdulaziz University	2	3.279	0
Korea Advanced Institute of Science Technology KAIST	2	3.279	7
Menofia University	2	3.279	0
National University of Sciences Technology Pakistan	2	3.279	0
Near East University	2	3.279	0
Qatar University	2	3.279	1
Sunnybrook Health Science Center	2	3.279	2
Sunnybrook Research Institute	2	3.279	1
Tehran University of Medical Sciences	2	3.279	2
Torrens University Australia	2	3.279	0
Tsinghua University	2	3.279	0
University of Oxford	2	3.279	4
University of Patras	2	3.279	33
University of Sevilla	2	3.279	2
Yonsei University	2	3.279	0

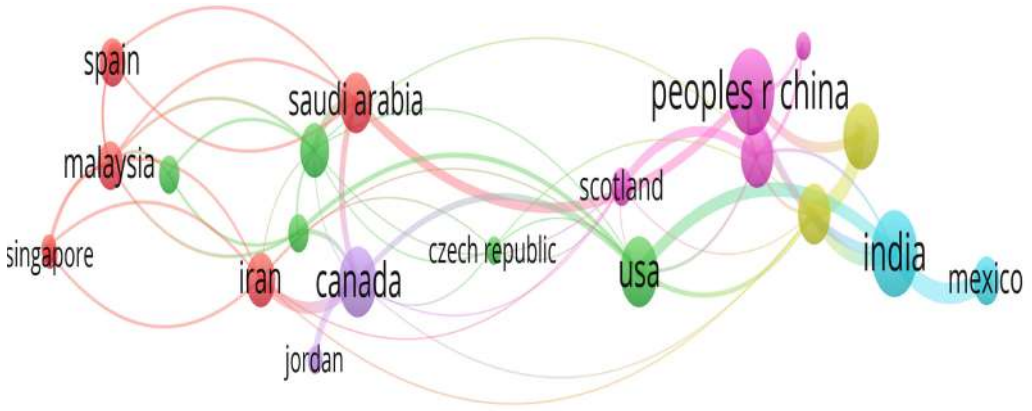

Fig. (5). Co-authorship countries.

8.6.4. Highly Prolific Authors

In this section, we emphasized the Prolific Authors' knowledge in the fields of coronavirus and related research. Table **5** shows the 20 most Prolific authors and affiliations with their total research publications (NP), Total Citations (TC), and corresponding h-index, g-index & m-index values to know the productivity as well as the impact of their publications.

Table 5. Highly Prolific Authors.

Author	Affiliation	NP	TC	h-index	g-index	m-index
Apostolopoulos I D	University of Patras	2	33	2	2	2
Mohammadi A	Urmia University Med Sci	2	15	1	2	1
Feng J	Tsinghua University	2	4	1	2	1
Zheng C	Huazhong University Sci and Technology	2	4	1	2	1
Zhu H	Southern Med University	2	3	1	1	1
Kim Y	Cheongju University	2	1	1	1	1
Zhang X	Huazhong University Sci and Technology	2	1	1	1	1
Attallah O	Arab ACAD Sci Technology and Maritime Transport	2	0	0	0	0
Mirjalili S	Torrens University Australia	2	0	0	0	0
Ozsahin I	Near East University	2	0	0	0	0
Ragab D A	Arab ACAD Sci Technology and Maritime Transport	2	0	0	0	0
Abdelmageed M I	University of Khartoum	1	11	1	1	1
Roshani S	Islamic Azad University	1	2	1	1	1
Abbasi A	Masha Hoshmand-Kochi	1	1	1	1	1
Abbas A	Huazhong University Sci and Technology	1	0	0	0	0
Abd El-Latif A A	Menoufia University	1	0	0	0	0
Abd El-Rahiem B	Menoufia University	1	0	0	0	0
Abd El-Samie F E	Menoufia University	1	0	0	0	0
Abdel	Menoufia University	1	0	0	0	0
Abdel-Raheem A	Alneelain University	1	0	0	0	0

Considering the number of research papers first 11 authors published 2 papers each and 9 authors published 1 each. But, if we consider the impact of the scientific output of the authors, *Apostolopoulos I D* from the Institute of *University of Patras (Greece)* placed top and received the most citations (TC=33, 2 h-index, g-index, and m-index), followed by *Mohammadi A* from the institute of

Urmia University of Medical Sciences (Iran) received citations (TC=15, 1 h-index, 2 g-index, and 1 m-index) and *Abdelmageed M I* from *University Khartoum* received citations (TC=11, 1 h-index, g-index, and m-index) respectively. Surprisingly, we note that Attallah O, Mirjalili S, Ozsahin I, and Ragab D A contributed 2 papers each. However, they failed to receive citations so far.

Three-field plot (Fig. **6**) displays the topmost productive institutions, authors, and countries with the collaborative nature by performing research under serious crisis in covid times which reflects the major impact the contribution to the knowledge domain for the welfare or improving the condition which is major indeed during the covid crisis and its immersive pleasure of the coverage. (Fig. **7**) shows that the majority of authors collaborated and produced the research output.

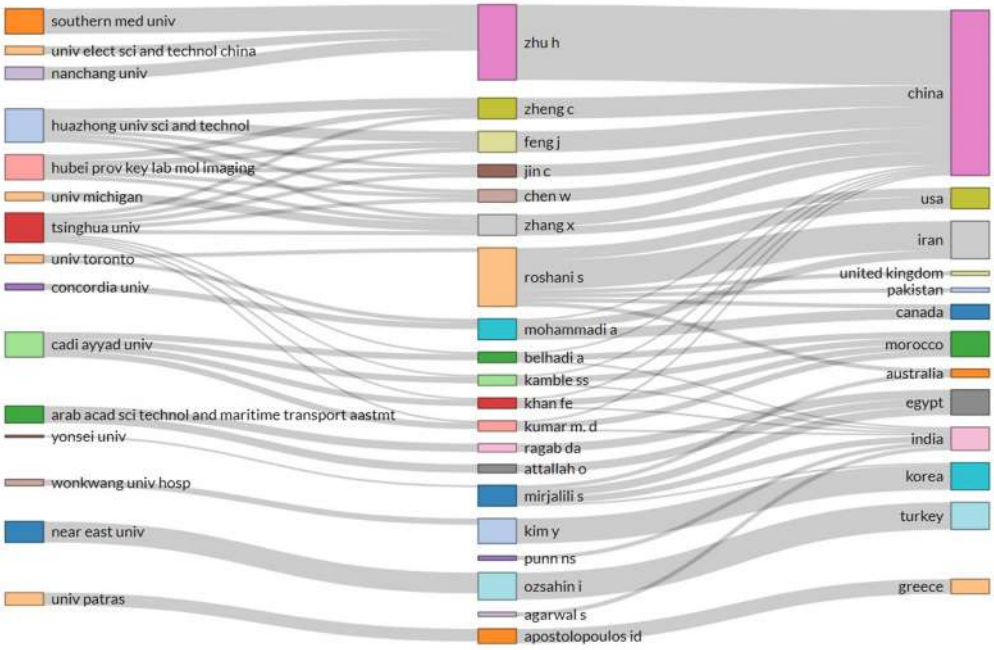

Fig. (6). Three-Fields Plot Most Productive Authors – Affiliations – Countries.

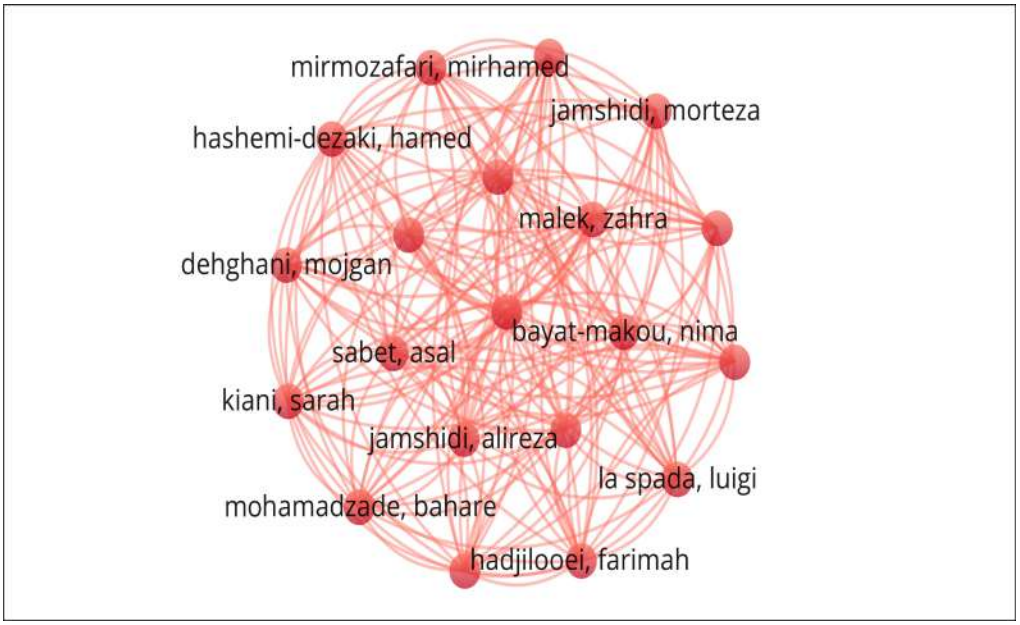

Fig. (7). Co-authorship-Authors Network.

8.6.5. Most Global Cited Papers

The most cited papers are essential to the reputation of a university [44]. The top 20 papers relating to coronavirus, computational intelligence, and related research were ranked according to citation count (Table **6**). The median (IQR) number of citations was 3.75 (5.75–2). *"COVID-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks"* published in *Physical and Engineering Sciences in Medicine* in 2020 was top in the list with total citations of 28, followed by *"Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks"* published in *Computers in Biology and Medicine* in 2020 with citations of 14 and "*Design of a Multiepitope-Based Peptide Vaccine against the E Protein of Human COVID-19: An Immunoinformatics Approach"* published in *BioMed Research International* in 2020 with citations of 11. 20 Most Global Cited papers are published in 16 journals, with IEEE access published in 2 and IEEE Transactions on Medical Imaging in 4 papers. Fig. (**8**) presents the factorial map of the most cited papers in the coronavirus, computational intelligence, and related research field and consists of two clusters. Cluster # 1 represented 19 papers, and Cluster # 2 represented 1 paper.

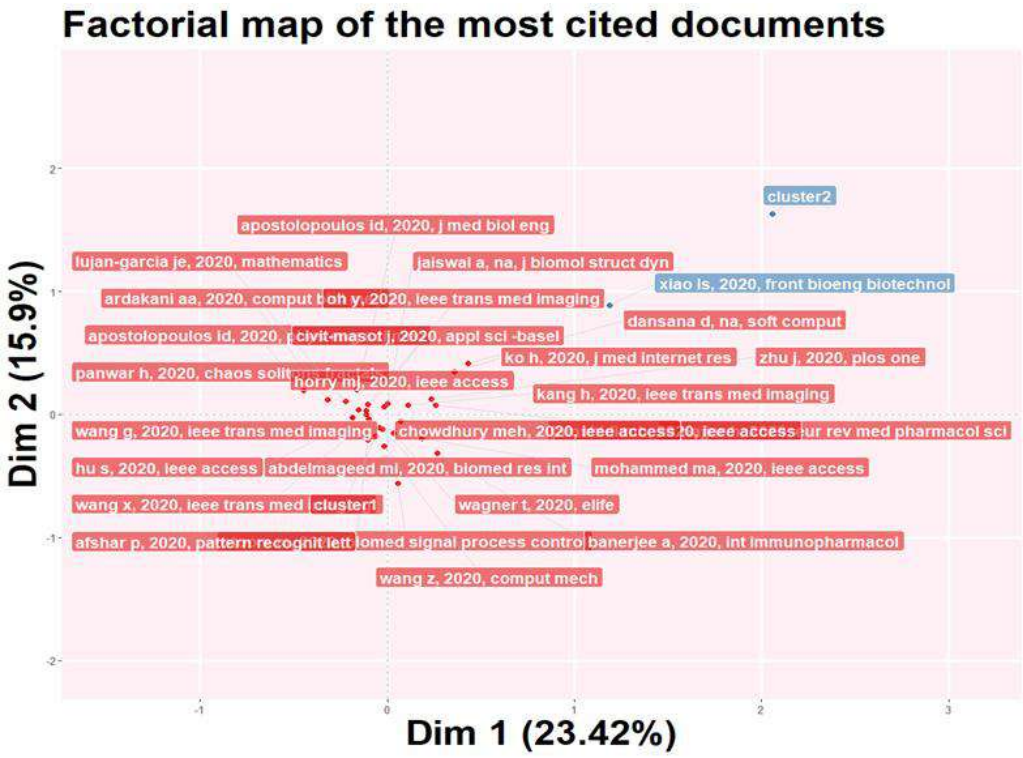

Fig. (8). Factorial map of the most cited papers.

Table 6. Most cited papers.

Paper	DOI	TC
APOSTOLOPOULOS ID, 2020, PHYS ENG SCI MED	10.1007/s13246-020-00865-4	28
ARDAKANI AA, 2020, COMPUT BIOL MED	10.1016/j.compbiomed.2020.103795	14
ABDELMAGEED MI, 2020, BIOMED RES INT	10.1155/2020/2683286	11
OH Y, 2020, IEEE TRANS MED IMAGING	10.1109/TMI.2020.2993291	7
JAISWAL A, NA, J BIOMOL STRUCT DYN	10.1080/07391102.2020.1788642	6
PANWAR H, 2020, CHAOS SOLITONS FRACTALS	10.1016/j.chaos.2020.109944	5
APOSTOLOPOULOS ID, 2020, J MED BIOL ENG	10.1007/s40846-020-00529-4	5
AL-NAJJAR H, 2020, EUR REV MED PHARMACOL SCI	NA	5
KANG H, 2020, IEEE TRANS MED IMAGING	10.1109/TMI.2020.2992546	4
WANG X, 2020, IEEE TRANS MED IMAGING	10.1109/TMI.2020.2995965	4
HU S, 2020, IEEE ACCESS	10.1109/ACCESS.2020.3005510	3
OTOOM M, 2020, BIOMED SIGNAL PROCESS CONTROL	10.1016/j.bspc.2020.102149	2

(Table 6) cont.....

Paper	DOI	TC
WANG G, 2020, IEEE TRANS MED IMAGING	10.1109/TMI.2020.3000314	2
WAGNER T, 2020, ELIFE	10.7554/eLife.58227	2
CIVIT-MASOT J, 2020, APPL SCI -BASEL	10.3390/app10134640	2
JAMSHIDI MB, 2020, IEEE ACCESS	10.1109/ACCESS.2020.3001973	2
AFSHAR P, 2020, PATTERN RECOGNIT LETT	10.1016/j.patrec.2020.09.010	1
LUJAN-GARCIA JE, 2020, MATHEMATICS	10.3390/math8091423	1
BANERJEE A, 2020, INT IMMUNOPHARMACOL	10.1016/j.intimp.2020.106705	1
DANSANA D, NA, SOFT COMPUT	10.1007/s00500-020-05275-y	1

8.6.6. Most Frequent Author Keywords

In this section, we studied the research keywords used by the authors in Coronavirus, computational intelligence, and related research over time. Bibliometric analysis of author keywords can offer directions in research, which may be a useful way to research the growth of scientific outputs [45]. Table **7** also discusses different research clusters wherein the studies are focused frequently through the co-occurrence of keywords and research dynamics. Figs. (**9** and **10**) shows the co-occurrence network of keywords included in the retrieved documents. The node size represents the occurrence of keyword occurrence in proportion, and the thickness of a line represents the occurrence of the two keywords co-occurrence in the same documents. The top 10 keywords with the highest frequency are *COVID-19* (46), *Deep Learning* (23), *Machine Learning* (11), *Computed Tomography* (10), and *Lung* (10).

Table 7. Most Frequent author Keywords.

Words	Occurrences
COVID-19	46
Deep Learning	23
Machine Learning	11
Computed Tomography	10
Lung	10
Pneumonia	9
Artificial Intelligence	6
Classification	6
Convolutional Neural Network	6
Diseases	6

(Table 7) cont.....

Words	Occurrences
Learning	6
Transfer Learning	6
CNN	5
Coronavirus	5
Diagnosis	5
X-Ray	5
Neural Network	4
Sars-Cov-2	4
Training	4
Deep	3

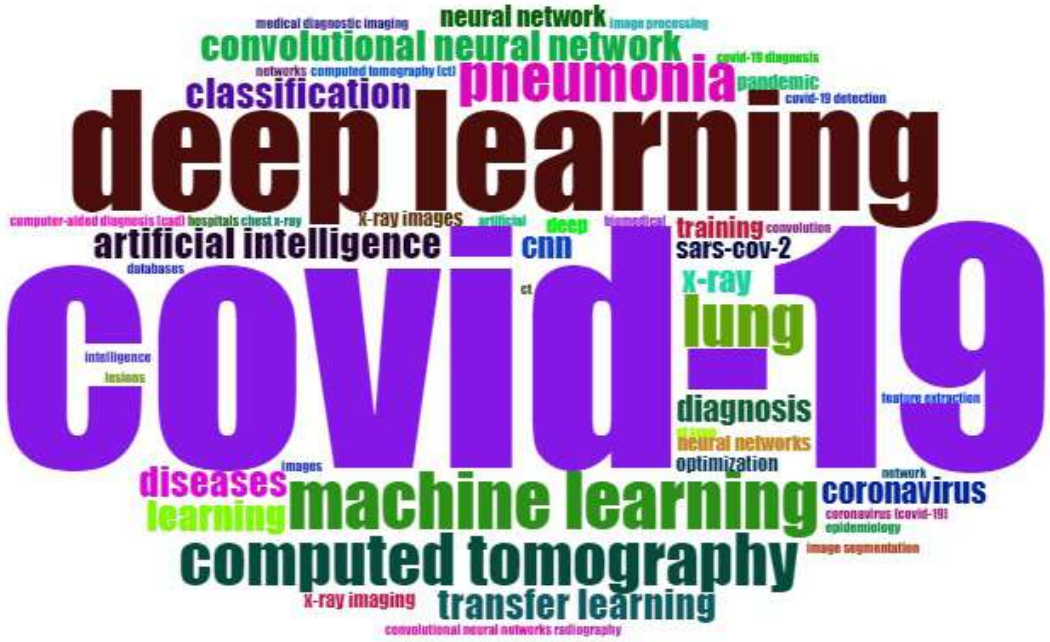

Fig. (9). Most frequent author keywords.

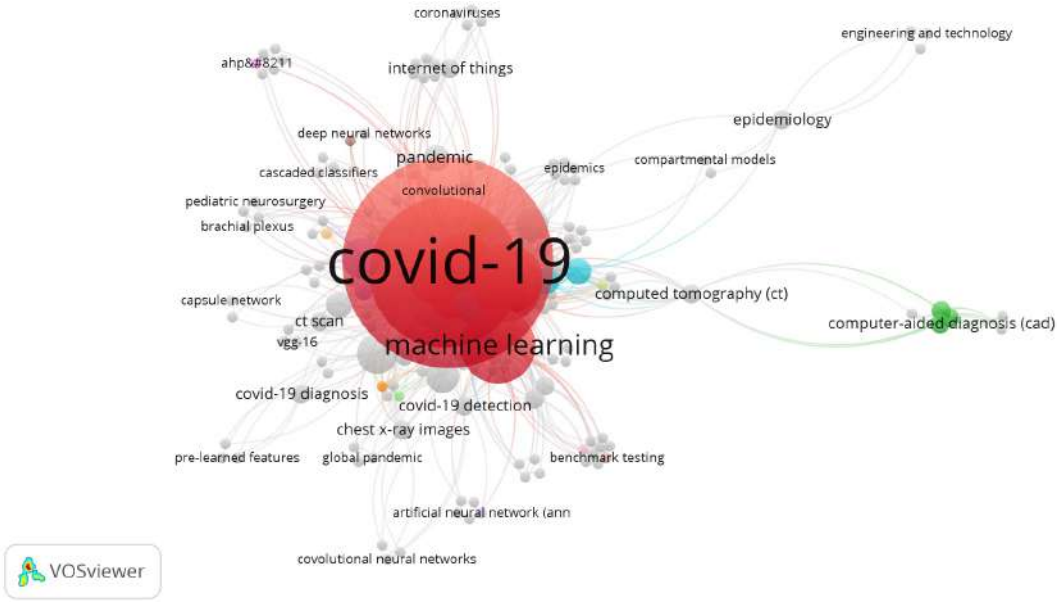

Fig. (10). Co-occurrence network of author keywords.

8.7. CONCLUSION AND FORTHCOMING OUTLOOK

Computer intelligence (CI) is a properly described and challenging area of research that is speedily growing all through this COVID-19 disease to handle the race in vaccine designing. Now need a vaccine, and the scientific group is at its excessive time looking for the vaccine for this virus. Several vaccine categories are under experimental, and researchersattempt to investigate vaccine techniques for treating this lethal virus. It was also known that the typical vaccine consisting of the whole disease might also not work accurately in growing an effective vaccine [46]. Therefore, Developing technologies furnish many possibilities for enhancing the precision of infectious disease prediction [47]. Figs. (**9**) shows that science and technology-other topics are the top three subject categories. Recently, some research has mentioned the emerging technologies to predict infectious diseases, consisting of remote sensing technology [48], artificial intelligence [49], big data analysis [50, 51], social media [52]. The emergence of new technologies is essential for researchers and scholars to enhance the prediction precision of infectious diseases. As a result, how to find more effective and reliable new techniques for speedy responses to the demands of the prediction can be explored in the future.

Many factors were measured in the prediction of infectious diseases. Based on Table **7**, we can find that COVID-19 and Deep Learning are high-frequency keyword co-occurrence by authors, and these may be pathogenic factors. Some

factors often discussed in the prediction models of infectious diseases include human behaviors, temperature variation, population mobility, the relationship between humans and wildlife, *etc* [52 - 55].

We intended to provide an overview of the entirety of computational intelligence/ machine learning and related specific areas that have been used for the diagnosis and treatment of "COVID-19" patients' research. This study provides a comprehensive overview of the computational intelligence and related specific areas of research conducted which have been used for diagnosis and treatment of "COVID-19" patients. The detection of the virus was the most important and challenging task, but modern-day technology detection, diagnosis and treatment made it possible to cover the possible aspects of detectingthe virus. CI/ML and alliance research provide various aspects of different strategies to diagnose and treat the needed "COVID-19" patients by CT Scan and X-ray by their respective datasets. Hence, there are 11 (18.033%) research papers (using different computational intelligence and related techniques) each from India and China that contributed to the collaborative effort to combat against the "Covid- 19". IEEE Access published 9 scientific publications related to computational intelligence and allied subjects. However, the IEEE Transactions on Medical Imaging tops the list with total citations of (TC=17 and h-index 3). Highly prolific institute 'Huazhong University of Science Technology' contributed 3 publications with 8 citation count. Apostolopoulos I D from 'University of Patras (Greece)' placed top author and received the most citations (TC=33, 2 h-index, 2 g-index, and 2 m-index). The median (IQR) number of citations was 3.75 (5.75–2). "COVID-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks," published in Physical and Engineering Sciences in Medicine in 2020 was top in the list with total citations of 28. The most frequently occurred top keywords are: COVID-19 (46), Deep Learning (23), Machine Learning (11), Computed Tomography (10). The Computational intelligence and related specific areas of research conducted which have been used for diagnosis and treatment of "COVID-19" are accelerating quickly, with potential applications being verified across different areas of medicine. However, there are presently rare examples of such techniques effectively deployed into clinical practice. Future computational intelligence research should be dedicated to filling the gap between computational intelligence and related specific areas of research for diagnosis and treatment of "COVID-19".

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

The authors wish to thank Dr. Chaman Sab M, Dr. Vitthal T. Bagalkoti for their work in extracting data from the web of science and suggestions on preparing the manuscript.

REFERENCES

[1] W. H. O. (OMS), *COVID-19 Weekly Epidemiological Update,* 2020. https://www.who.int/docs/default-source/coronaviruse/situation-reports/20201012-weekly-epi-update-9.pdf

[2] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare", *Nat. Med.,* vol. 25, no. 1, pp. 24-29, 2019.
[http://dx.doi.org/10.1038/s41591-018-0316-z] [PMID: 30617335]

[3] E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence", *Nature Medicine,* vol. 25, no. 1, 2019. Nature Publishing Group, pp. 44–56.
[http://dx.doi.org/10.1038/s41591-018-0300-7]

[4] D. Ardila, A.P. Kiraly, S. Bharadwaj, B. Choi, J.J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado, D.P. Naidich, and S. Shetty, "End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography", *Nat. Med.,* vol. 25, no. 6, pp. 954-961, 2019.
[http://dx.doi.org/10.1038/s41591-019-0447-x] [PMID: 31110349]

[5] F. Liao, M. Liang, Z. Li, X. Hu, and S. Song, "Evaluate the Malignancy of Pulmonary Nodules Using the 3-D Deep Leaky Noisy-OR Network", *IEEE Trans. Neural Netw. Learn. Syst.,* vol. 30, no. 11, pp. 3484-3495, 2019.
[http://dx.doi.org/10.1109/TNNLS.2019.2892409] [PMID: 30794190]

[6] L. Gong, S. Jiang, Z. Yang, G. Zhang, and L. Wang, "Automated pulmonary nodule detection in CT images using 3D deep squeeze-and-excitation networks", *Int. J. CARS,* vol. 14, no. 11, pp. 1969-1979, 2019.
[http://dx.doi.org/10.1007/s11548-019-01979-1] [PMID: 31028657]

[7] B. Mallikarjun, and M. Kappi, *Visualisation of Asthma Research Output in India during 2010-2019. Libr. Philos. Pract.,* 2020.https://digitalcommons.unl.edu/libphilprac/4253

[8] "COVID-19 Open Research Datase", https://search.bvsalud.org/global-literature-on-novel-coronavirus-2019-ncov/

[9] H. Chiroma, A.E. Ezugwu, F. Jauro, M.A. Al-Garadi, I.N. Abdullahi, and L Shuib, "Early survey with bibliometric analysis on machine learning approaches in controlling coronavirus", *medRxiv,* p. 2020.11.04.20225698.
[http://dx.doi.org/10.1101/2020.11.04.20225698]

[10] V.S. Tseng, J. Jia-Ching Ying, S.T.C. Wong, D.J. Cook, and J. Liu, "Computational Intelligence Techniques for Combating COVID-19: A Survey", *IEEE Comput. Intell. Mag.,* vol. 15, no. 4, pp. 10-22, 2020.
[http://dx.doi.org/10.1109/MCI.2020.3019873]

[11] B.X. Tran, G.H. Ha, L.H. Nguyen, G.T. Vu, M.T. Hoang, H.T. Le, C.A. Latkin, C.S.H. Ho, and R.C.M. Ho, "Studies of Novel Coronavirus Disease 19 (COVID-19) Pandemic: A Global Analysis of Literature", *Int. J. Environ. Res. Public Health,* vol. 17, no. 11, pp. 1-16, 2020.
[http://dx.doi.org/10.3390/ijerph17114095] [PMID: 32521776]

[12] O.S. Albahri, A.A. Zaidan, A.S. Albahri, B.B. Zaidan, K.H. Abdulkareem, Z.T. Al-Qaysi, A.H. Alamoodi, A.M. Aleesa, M.A. Chyad, R.M. Alesa, L.C. Kem, M.M. Lakulu, A.B. Ibrahim, and N.A. Rashid, "Systematic review of artificial intelligence techniques in the detection and classification of COVID-19 medical images in terms of evaluation and benchmarking: Taxonomy analysis, challenges, future solutions and methodological aspects", *J. Infect. Public Health,* vol. 13, no. 10, pp. 1381-1396, 2020.
[http://dx.doi.org/10.1016/j.jiph.2020.06.028] [PMID: 32646771]

[13] R.P. Singh, M. Javaid, A. Haleem, and R. Suman, "Internet of things (IoT) applications to fight against COVID-19 pandemic", *Diabetes Metab. Syndr.,* vol. 14, no. 4, pp. 521-524, 2020.
[http://dx.doi.org/10.1016/j.dsx.2020.04.041] [PMID: 32388333]

[14] F. De Felice, and A. Polimeni, *Coronavirus disease (COVID-19): A machine learning bibliometric analysis, In Vivo (Brooklyn).* vol. 34. , 2020, pp. 1613-1617.
[http://dx.doi.org/10.21873/invivo.11951]

[15] A. Kalantari, "A bibliometric approach to tracking big data research trends", *J. Big Data,* vol. 4, no. 1, pp. 1-18, 2017.
[http://dx.doi.org/10.1186/s40537-017-0088-1]

[16] B.M. Gupta, and S.M. Dhawan, "Artificial intelligence research in India: A scientometric assessment of publications output during 2007-16", *DESIDOC J. Libr. Inf. Technol.,* vol. 38, no. 6, pp. 415-422, 2018.
[http://dx.doi.org/10.14429/djlit.38.6.12309]

[17] G. Trivedi, *Visualization and Scientometric Mapping of Global Agriculture Big Data Research. Libr. Philos. Pract.,* 2019.https://digitalcommons.unl.edu/libphilprac/2478%0A

[18] B.M. Gupta, and S.M. Dhawan, "Deep Learning Research: Scientometric Assessment of Global Publications Output during 2004 -17", *Emerg. Sci. J.,* vol. 3, no. 1, p. 23, 2019.
[http://dx.doi.org/10.28991/esj-2019-01165]

[19] R. Shrivastava, and P. Mahajan, *Artificial Intelligence Research in India : A Scientometric Analysis Artificial Intelligence Research in India : A Scientometric, .*
[http://dx.doi.org/10.1080/0194262X.2016.1181023]

[20] A. Sidorova, N. Evangelopoulos, J.S. Valacich, and T. Ramakrishnan, *Uncovering the intellectual core of the information systems discipline,* 2008.https://arizona.pure.elsevier.com/en/publications/uncovering-the-intellectual-core-of-the-information-systems-disci
[http://dx.doi.org/10.2307/25148852]

[21] J. Bezdek, *What is computational intelligence?,* 1994.https://www.osti.gov/biblio/81587

[22] R.J. Marks, "Intelligence : computational versus artificial", *IEEE Trans. Neural Networks,* vol. 4, 1993no. 5, pp. 737-739.https://ci.nii.ac.jp/naid/10010976199

[23] L.A. Zadeh, FUZZY SETS.*Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems.,* G.J. Klir, B. Yuan, Eds., , 1996, pp. 394-432.
[http://dx.doi.org/10.1142/9789814261302_0021]

[24] A.H. Rabie, S.H. Ali, H.A. Ali, and A.I. Saleh, "A fog based load forecasting strategy for smart grids using big electrical data", *Cluster Comput.,* vol. 22, no. 1, pp. 241-270, 2019.
[http://dx.doi.org/10.1007/s10586-018-2848-x]

[25] O. Užga-Rebrovs, and G. Kuļešova, "Comparative Analysis of Fuzzy Set Defuzzification Methods in the Context of Ecological Risk Assessment", *Inf. Technol. Manag. Sci.,* vol. 20, no. 1, pp. 25-29, 2018.
[http://dx.doi.org/10.1515/itms-2017-0004]

[26] N. Aydin, and G. Yurdakul, *Assessing countries ' performances against COVID-19 via WSIDEA and machine learning algorithms,* 2020.http://www.sciencedirect.com/science/article/pii/S1568494620307304
[http://dx.doi.org/10.1016/j.asoc.2020.106792]

[27] G.V. Gayathri, and S.C. Satapathy, "A Survey on Techniques for Prediction of Asthma", *Smart Innov. Syst. Technol.,* vol. 159, pp. 751-758, 2020.
[http://dx.doi.org/10.1007/978-981-13-9282-5_72]

[28] F. Pegoraro, E.A. Portela Santos, E. de Freitas Rocha Loures, and F.W. Laus, "A hybrid model to support decision making in emergency department management", *Knowl. Base. Syst.,* vol. 203, p. 106148, 2020.
[http://dx.doi.org/10.1016/j.knosys.2020.106148]

[29] N. Siddique, and H. Adeli, *Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing.* John Wiley & Sons: United Kingdom, 2013.
[http://dx.doi.org/10.1002/9781118534823]

[30] K. De Jong, "Evolutionary computation: A unified approach",
[http://dx.doi.org/10.1145/3377929.3389871]

[31] V.M. Corman, O. Landt, M. Kaiser, R. Molenkamp, A. Meijer, D.K. Chu, T. Bleicker, S. Brünink, J. Schneider, M.L. Schmidt, D.G. Mulders, B.L. Haagmans, B. van der Veer, S. van den Brink, L. Wijsman, G. Goderski, J.L. Romette, J. Ellis, M. Zambon, M. Peiris, H. Goossens, C. Reusken, M.P. Koopmans, and C. Drosten, "Detection of 2019 novel coronavirus (2019-nCoV) by real-time RT-PCR", *Euro Surveill.,* vol. 25, no. 3, pp. 1-8, 2020.
[http://dx.doi.org/10.2807/1560-7917.ES.2020.25.3.2000045] [PMID: 31992387]

[32] X. Xu, C. Yu, J. Qu, L. Zhang, S. Jiang, D. Huang, B. Chen, Z. Zhang, W. Guan, Z. Ling, R. Jiang, T. Hu, Y. Ding, L. Lin, Q. Gan, L. Luo, X. Tang, and J. Liu, "Imaging and clinical features of patients with 2019 novel coronavirus SARS-CoV-2", *Eur. J. Nucl. Med. Mol. Imaging,* vol. 47, no. 5, pp. 1275-1280, 2020.
[http://dx.doi.org/10.1007/s00259-020-04735-9] [PMID: 32107577]

[33] X. Xie, Z. Zhong, W. Zhao, C. Zheng, F. Wang, and J. Liu, "Chest CT for Typical Coronavirus Disease 2019 (COVID-19) Pneumonia: Relationship to Negative RT-PCR Testing", *Radiology,* vol. 296, no. 2, pp. E41-E45, 2020.
[http://dx.doi.org/10.1148/radiol.2020200343] [PMID: 32049601]

[34] L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang, B. Kong, J. Bai, Y. Lu, Z. Fang, Q. Song, K. Cao, D. Liu, G. Wang, Q. Xu, X. Fang, S. Zhang, J. Xia, and J. Xia, "Using Artificial Intelligence to Detect COVID-19 and Community-acquired Pneumonia Based on Pulmonary CT: Evaluation of the Diagnostic Accuracy", *Radiology,* vol. 296, no. 2, pp. E65-E71, 2020.
[http://dx.doi.org/10.1148/radiol.2020200905] [PMID: 32191588]

[35] A.S.R. Srinivasa Rao, and J.A. Vazquez, "Identification of COVID-19 can be quicker through artificial intelligence framework using a mobile phone-based survey when cities and towns are under quarantine", *Infect. Control Hosp. Epidemiol.,* vol. 41, no. 7, pp. 826-830, 2020.
[http://dx.doi.org/10.1017/ice.2020.61] [PMID: 32122430]

[36] F. Shi, "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19", *IEEE Rev. Biomed. Eng.,* vol. •••, pp. 1-11, 2020.
[PMID: 32305937]

[37] M.M. Rahaman, C. Li, Y. Yao, F. Kulwa, M.A. Rahman, Q. Wang, S. Qi, F. Kong, X. Zhu, and X. Zhao, "Identification of COVID-19 samples from chest X-Ray images using deep learning: A comparison of transfer learning approaches", *J. XRay Sci. Technol.,* vol. 28, no. 5, pp. 821-839, 2020.
[http://dx.doi.org/10.3233/XST-200715] [PMID: 32773400]

[38] M.i.e. Fernández, P.L. Barbosa, and A.P. Guerrero, *Web of science vs. scopus: a quantitative study in chemical engineering | Information Science Journal.* vol. Vol. 13. Inf. Sci. J., 2010, pp. 159-175.https://revistas.um.es/analesdoc/article/view/107121 [Online]

[39] https://www.bibliometrix.org/

[40] M. Aria, and C. Cuccurullo, "bibliometrix: An R-tool for comprehensive science mapping analysis", *J.*

Informetrics, vol. 11, no. 4, pp. 959-975, 2017.
[http://dx.doi.org/10.1016/j.joi.2017.08.007]

[41] J.M. Alonso, C. Castiello, and C. Mencar, "A bibliometric analysis of the explainable artificial intelligence research field", *Commun. Comput. Inf. Sci.,* vol. 853, pp. 3-15, 2018.
[http://dx.doi.org/10.1007/978-3-319-91473-2_1]

[42] M.K. Linnenluecke, M. Marrone, and A.K. Singh, "Conducting systematic literature reviews and bibliometric analyses", *Aust. J. Manag.,* vol. 45, no. 2, pp. 175-194, 2020.
[http://dx.doi.org/10.1177/0312896219877678]

[43] A.E. Ezugwu, A.K. Shukla, M.B. Agbaje, O.N. Oyelade, A. José-García, and J.O. Agushaka, *Automatic clustering algorithms: a systematic review and bibliometric analysis of relevant literature.* vol. Vol. 4. , 2020.

[44] X. Zhu, Q. Wu, Y. Zheng, and X. Ma, "Highly cited research papers and the evaluation of a research university: A case study: Peking University 1974-2003", *Scientometrics,* vol. 60, no. 2, pp. 237-347, 2004.
[http://dx.doi.org/10.1023/B:SCIE.0000027795.69665.09]

[45] H. Du, L. Wei, M.A. Brown, Y. Wang, and Z. Shi, "A bibliometric analysis of recent energy efficiency literatures: An expanding and shifting focus", *Energy Effic.,* vol. 6, no. 1, pp. 177-190, 2013.
[http://dx.doi.org/10.1007/s12053-012-9171-9]

[46] K.K. Bharadwaj, *Computational Intelligence in Vaccine Design Against COVID-19.* Springer Singapore, 2021.
[http://dx.doi.org/10.1007/978-981-15-8534-0_16]

[47] W. Yang, J. Zhang, and R. Ma, "The prediction of infectious diseases: A bibliometric analysis", *Int. J. Environ. Res. Public Health,* vol. 17, no. 17, pp. 1-19, 2020.
[http://dx.doi.org/10.3390/ijerph17176218] [PMID: 32867133]

[48] T.E. Ford, R.R. Colwell, J.B. Rose, S.S. Morse, D.J. Rogers, and T.L. Yates, "Using satellite images of environmental changes to predict infectious disease outbreaks", *Emerg. Infect. Dis.,* vol. 15, no. 9, pp. 1341-1346, 2009.
[http://dx.doi.org/10.3201/eid/1509.081334] [PMID: 19788799]

[49] Z. Yang, Z. Zeng, K. Wang, S.S. Wong, W. Liang, M. Zanin, P. Liu, X. Cao, Z. Gao, Z. Mai, J. Liang, X. Liu, S. Li, Y. Li, F. Ye, W. Guan, Y. Yang, F. Li, S. Luo, Y. Xie, B. Liu, Z. Wang, S. Zhang, Y. Wang, N. Zhong, and J. He, "Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions", *J. Thorac. Dis.,* vol. 12, no. 3, pp. 165-174, 2020.
[http://dx.doi.org/10.21037/jtd.2020.02.64] [PMID: 32274081]

[50] X. Sun, F. Ren, and J. Ye, "Trends detection of flu based on ensemble models with emotional factors from social networks", *IEEJ Trans. Electr. Electron. Eng.,* vol. 12, no. 3, pp. 388-396, 2017.
[http://dx.doi.org/10.1002/tee.22389]

[51] J. Salerno, B.M. Knoppers, L.M. Lee, W.M. Hlaing, and K.W. Goodman, "Ethics, big data and computing in epidemiology and public health", *Ann. Epidemiol.,* vol. 27, no. 5, pp. 297-301, 2017.
[http://dx.doi.org/10.1016/j.annepidem.2017.05.002] [PMID: 28595734]

[52] R.A. Weiss, and A.J. McMichael, "Social and environmental risk factors in the emergence of infectious diseases", *Nat. Med.,* vol. 10, no. 12, suppl. Suppl., pp. S70-S76, 2004.
[http://dx.doi.org/10.1038/nm1150] [PMID: 15577934]

[53] K.E. Jones, N.G. Patel, M.A. Levy, A. Storeygard, D. Balk, J.L. Gittleman, and P. Daszak, "Global trends in emerging infectious diseases", *Nature,* vol. 451, no. 7181, pp. 990-993, 2008.
[http://dx.doi.org/10.1038/nature06536] [PMID: 18288193]

[54] S. S. Morse, *Prediction and prevention of the next pandemic zoonosis,* .
[http://dx.doi.org/10.1016/S0140-6736(12)61684-5]

[55] W.I. Lipkin, "The changing face of pathogen discovery and surveillance", *Nat. Rev. Microbiol.,* vol. 11, no. 2, pp. 133-141, 2013.
[http://dx.doi.org/10.1038/nrmicro2949] [PMID: 23268232]

SUBJECT INDEX

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Om Prakash Jena

Dr. Om Prakash Jena: Om Prakash Jena is currently working as an assistant professor at the Department of Computer Science, Ravenshaw University, Cuttack, and Odisha. He has 10 years of teaching and research experience at undergraduate and postgraduate level. He has contributed several technical papers in international journals/conferences/edited book chapters with reputable publishers. He is a member of IEEE, IETA, IAAC, IRED, IAENG and WACAMLDS. He has guided many projects and a thesis at the UG and PG levels. He has many edited books, published by Wiley, CRC press, and Bentham Science Publishers to his credit and is also the author of two textbooks under Kalyani Publisher.

Alok Ranjan Tripathy

Dr. Alok Ranjan Tripathy is currently working as an assistant professor at the Department of Computer Science, Ravenshaw University, Cuttack, and Odisha. He has more than 15 years of teaching experience at undergraduate and postgraduate levels. He obtained his M.Tech and Ph.D. degrees in Computer Science from the Utkal University. He has contributed several technical papers in international journals/conferences/ edited book chapters with reputable publishers. He is a life member of ISTE, CSI and Odisha Information Technology Society (OITS).

Brojo Kishore Mishra

Dr. Brojo Kishore Mishra is currently working as associate dean (International Affairs) and professor at the Department of Computer Science & Engineering, School of Engineering & Technology, GIET University, Gunupur, India. He received his Ph.D. degree in the field of Computer Science from Berhampur University, India in 2012. His main research areas include Data Mining, Machine Learning and Security.

Ahmed A. Elngar

Dr. Ahmed A. Elngar is an assistant professor of Computer Science, Chair of Scientific Innovation Research Group (SIRG) Director of Technological and Informatics Studies Center Managing Editor of Journal of Cyber Security and Information Management (JCIM), Beni-Suef University, Faculty of Computers & Artificial Intelligence, Egypt. Dr. Elngar is the Founder and Head of Scientific Innovation Research Group (SIRG).