Criterion :3 Research, Innovations and Extension

3.4 Research Publications and awards

3.4.1 The Institution ensures implementation of its stated Code of Ethics for research through the following:

1. **Inclusion of research ethics in the research methodology course work**
2. **Presence of Ethic committee**
3. **Plagiarism check through software**
4. **Research Advisory Committee**

S. No.	Description	Page Number
1.	Inclusion of research ethics in the research methodology course work	2
2.	Presence of Ethic committee	21
3.	Plagiarism check through software	24
4.	Research Advisory Committee	90

The Institution ensures implementation of its stated Code of Ethics for research through the following:

1. **Inclusion of research ethics in the research methodology course work**
2. **Presence of Ethic committee**
3. **Plagiarism check through software**
4. **Research Advisory Committee**

INCLUSION OF RESEARCH ETHICS IN THE RESEARCH METHODOLOGY COURSE WORK

E.G.S Pillay Engineering College pays more attention towards research and development activities, which motivates all research scholars, faculties and students to carry out the research. EGSPEC has become hub for real aspirants of research and faculty member. EGSPEC has clearly defined code of ethics and guidelines to check malpractices and plagiarism in research. Implementation of certain code of ethics is obligatory to bring out quality research, which in turn provides productive outcome. In this context, EGSPEC has included the Research Methodology as one of course in PG curriculum. Researchers Researchers, while doing their course work have to opt this paper one among the other subjects. In Master of Business Administration programme, we have Business Research Methods to make our MBA students aware of research design and measurement, data collection, data preparation and analysis, report design, writing and ethics in business research. We include Operations Researchcourse for UG students to make them aware of simulation methodologies.

The R&D cell of EGSPEC has framed a research ethics assessment pattern for periodical monitoring of the problem statement, tools/data used for the research project, work plan objectives and methodology to be followed at the time of research. The researcher has to adhere to the guidelines of Anna University, where a Doctoral Committee has to be formed and the committee will guide the scholars in the progress of their research. He should present his/her progress of research in the review meeting once in six months and submit the Half Yearly Progress Report in the prescribed format to the department in which they have registered. The schedule for the review meeting would be prepared and announced by the concerned department. Under the esteemed guidance of Head of Department and Research supervisor candidate's progress will be checked and monitored periodically. Along with the above said information a separate research advisory committee for monitoring and coordinating the research activities in the institution and separate Project evaluation committee to evaluate the progress and outcomes of the project like publications, patent, copy right, feasibility of the product to society.

ATTESTED

Dr. S. RAMABALAN, M.E., Ph.D.,
PRINCIPAL
E.G.S. Pillay Engineering College,
Thethi, Nagore - 611 002.
Nagapattinam (Dt) Tamil Nadu.

- Currently, institution has 35 research supervisors recognized by Anna University. Everyone have a separate login ID for online access to Urkund Plagiarism check software by Centre for Research, Anna University, Chennai.
- Research Scholar should submit the copy of Urkund plagiarism check report to the concern supervisor for a paper before submitting to the journal/conference. Apart from the recommended software Plagiarism on thesis, research paper, and assignments can be checked online at free of cost. The important top 10 free Plagiarism Checkers are DupliChecker.com, Plagiarismsoftware.net, Plagiarism CheckerX, PlagScan.com Plagramme.com, Smallseotools.com, Plagiarisma.net, Quetext.com, Easybib.com and Noplag.com.
- The research scholar Synopsis/ Thesis should accompany Urkund plagiarism report while submitting same to Center for Research, Anna University.
- All Post Graduate projects, dissertation, thesis should include the copy of Plagiarism check report.

ATTESTED

Dr. S. RAMABALAN, M.E., Ph.D.,
PRINCIPAL
E.G.S. Pillay Engineering College,
Thethi. Nagore - 611 002.
Nagapattinam (Dt) Tamil Nadu.

PRINCIPAL

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)

Approved by AICTE, New Delhi | Affiliated to Anna University, Chennai
Accredited by NAAC with 'A' Grade | Accredited by NBA (CSE, EEE, MECH)
NAGAPATTINAM – 611 002

M.E. POWER ELECTRONICS AND DRIVES

Full Time Curriculum and Syllabus

First Year – Second Semester

Course Code	Course Name	L	T	P	C	Maximum Marks		
						CA	ES	Total
Theory Course								
1701PE201	Research Methodology	3	0	0	3	40	60	100
1702PE202	Solid State DC Drives	3	0	0	3	40	60	100
1702PE203	Solid State AC Drives	3	0	0	3	40	60	100
1702PE204	Power Quality Issues and Solutions	3	0	0	3	40	60	100
1702PE205	Modelling and Design of SMPS	3	0	0	3	40	60	100
	Elective-II	3	0	0	3	40	60	100
Laboratory Course								
1704PE206	Electrical Drives Laboratory	0	0	2	1	50	50	100
1704PE207	Technical Seminar	0	0	2	1	100	0	100
1704PE208	Communication Skills Lab II	0	0	2	1	100	0	100

L – Lecture | T – Tutorial | P – Practical | C – Credit | CA – Continuous Assessment | ES – End Semester

1701PE201	RESEARCH METHODOLOGY	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES:

1. To understand the fundamentals of Research Methodology.
2. To analyze the various sampling methods.
3. To perform different test in research methodology.

UNIT I INTRODUCTION 10 Hours

Research methodology – definition, mathematical tools for analysis, Types of research, exploratory research, conclusive research, modeling research, algorithmic research, Research process- steps. Data collection methods- Primary data – observation method, personal interview, telephonic interview, mail survey, questionnaire design. Secondary data- internal sources of data, external sources of data.

UNIT II SCALES AND SAMPLING 11 Hours

Scales – measurement, Types of scale – Thurstone's Case V scale model, Osgood's Semantic Differential scale, Likert scale, Q- sort scale. Sampling methods- Probability sampling methods – simple random sampling with replacement, simple random sampling without replacement, stratified sampling, cluster sampling. Non-probability sampling method – convenience sampling, judgment sampling, quota sampling.

UNIT III HYPOTHESIS TESTING 7 Hours

Hypothesis testing – Testing of hypotheses concerning means (one mean and difference between two means - one tailed and two tailed tests), concerning variance – one tailed Chi-square test.

UNIT IV MULTIVARIATE STATISTICAL TECHNIQUES 8 Hours

Data Analysis – Factor Analysis – Cluster Analysis – Discriminant Analysis – Multiple Regression and correlation – Canonical Correlation – Application of statistical (SPSS) Software Package in Research.

UNIT V RESEARCH REPORT 9 Hours

Purpose of the written report - Concept of Audience – Basics of written reports. Integral Parts of Report – Title of a Report, Table of Contents, Abstract, Synopsis, Introduction, Body of a Report – Experimental, Results and Discussion – Recommendations and Implementation Section – Conclusions and Scope for future work.

TOTAL: 45 HOURS

FURTHER READING:

Report writing for Assignments – A Case Study

COURSE OUTCOMES:

On the successful completion of the course, students will be able to

CO1: Explain the fundamentals of research methodology.
CO2: Elucidate the classification of scales and sampling methods.
CO3: Apply the hypothesis testing in research methodology.
CO4: Explain the methods of Data Analysis in research.
CO5: Discuss about report writing.

REFERENCES:

1. Panneerselvam, R., Research Methodology, Prentice-Hall of India, New Delhi, 2004.
2. Kothari, C.R., Research Methodology –Methods and Techniques, New Age International.

CENTRE FOR RESEARCH

ANNA UNIVERSITY
CHENNAI - 600 025

Dr. R. JAYAVEL
DIRECTOR

Telephone +91-44-2235 7186/2235 [illegible]
Fax +91-44-2220 1213
Email dircresearch@annauniv.edu
dircresearchau@gmail.com

Procs. No.17144597127/Ph.D./AR7 **Date : 27.06.2017**

Sub : Ph.D. Programme – Mr./Ms. Raju K, Research Scholar – Constitution of Doctoral Committee – Orders – Issued

Ref : This office Lr.No CR / Ph.D. / Admn / JUL / 2017 dated: 27.06.2017.

****

Mr./Ms. **Raju K** has been granted provisional registration for Ph.D. Programme under the guidance of Dr. M Chinnadurai vide reference cited. Accordingly a Doctoral Committee has been constituted comprising of the following members as required in the Clause 12.1 of Ph.D. Regulations.

1. **Dr.M Chinnadurai,Professor** — Supervisor/ Convener
   Department of Computer Science and Engineering
   E.G.S.Pillay Engineering College
   Nagappattinam - 611002 .
2. **Dr. K.Velmurugan,Professor** — Member
   Department of Computer Science and Engineering
   Anjalai Ammal Mahalingam Engineering College Kovilvenni-614403
3. **Dr. J.I.Sheeba,Professor** — Member
   Department of Computer Science and Engineering
   Pondicherry Engineering College Pondicherry-605014

The Doctoral Committee shall meet within 3 weeks from the date of receipt of this communication to prescribe course works to the scholar as per clause 13.1 of Ph.D. Regulation. **The Meetings of the Doctoral Committee shall be informed to the Director (Research) and Head of the Department/Director of the Centre well in advance. The minutes of the meetings / any other communications should be forwarded by the respective Head of the Department/Director of the Centre.**

The research scholar shall have a CGPA of 6.50 in the course works in order to become eligible for comprehensive examination. A pass in the comprehensive examination is required for the confirmation of Ph.D. Registration.

The Committee shall function as per rules of the Ph.D. regulations. Sitting Fee, TA / Conveyance shall be paid as per the norms. The Supervisor/ Convener shall convene the meeting of the Doctoral Committee and send the progress report periodically. As per Clause 6.4 **"The Scholar, Supervisor, Doctoral Committee Members and Examiners shall not be relatives to each others"**.

DIRECTOR

27/6/17

27/6

To
The Supervisor
Copy to : All Members

CENTRE FOR RESEARCH
ANNA UNIVERSITY, CHENNAI - 600 025

MINUTES OF THE FIRST DOCTORAL COMMITTEE MEETING

The Doctoral Committee Meeting of the Ph.D. Scholar, **Mr. RAJU K** (Reg.No. **17144597127**) was held on /07/2017 at 10.15 A.M. in the Department of COMPUTER SCIENCE AND ENGINEERING. The following embers were present

1. **Dr. M. CHINNADURAI** (Supervisor & Convener)
   Professor & HoD /CSE
   E.G.S Pillay Engineering College
   Nagapattinam – 611 002

2. **Dr. K. VELMURUGAN** (Member)
   Professor & Principal, Department of CSE
   Anjalai Ammal Mahalingam Engineering College
   Thiruvarur – 614403

3. **Dr. J.I. SHEEBA** (Member)
   Professor, Department of CSE
   Pondicherry Engineering College,
   Pondicherry - 605014

Mr. RAJU K has presented the overview of the proposed research work. The Doctoral Committee has pproved the research topic as "**IDENTITY BASED AUTHENTICATION FOR ENSURING DATA ECURITY IN CLOUD COMPUTING**". The Committee has recommended the scholar to undertake the llowing course works

Course Code	Course Title	Credits	Core Course / Elective / Special Elective
17CP105	DESIGN AND MANAGEMENT OF COMPUTER NETWORKS	4	Core Course
17CP202	SECURITY IN COMPUTING	3	Core Course
17CP004	CLOUD COMPUTING	3	Elective
17MF023	RESEARCH METHODOLOGY	3	Elective

Number of course works as applicable to the scholars

Member
(Signature with Name)
Dr. K. VELMURUGAN

J. I. Sheeba
Member
(Signature with Name)
Dr. J.I. SHEEBA.

Supervisor
Signature with Name and seal
Dr. M.CHINNADURAI, M.E. Ph D
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611002.

Joint Supervisor
(Signature with Name and seal)
(if applicable)

Forwarded

Signature of the HOD/Director of the Centre/Supervisor
(And Seal)
Dr. M.CHINNADURAI, M.E. Ph.D
Professor & Head
Department of CSE

E.G.S. PILLAY ENGINEERING COLLEGE
(Autonomous)

NAGAPATTINAM - 611 002.
(An ISO 9001:2015 Certified Institution)
(Approved by AICTE, New Delhi, Approved by Govt.of Tamil Nadu)
(Affiliated to Anna University Chennai)
(Accredited by NAAC with 'A' Grade)
(NBA Accredited Programmes B.E - CSE,EEE & Mechanical))

TÜV NORD
ISO 9001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

23/05/2018

SEMINAR NOTIFICATION

As per Anna University, Chennai Ph.D Regulations – 2017, there will be a seminar presentation by **Mr. K. RAJU (Register Number: 17144597127)** Ph.D Scholar of Department of Computer Science & Engineering, **E.G.S. Pillay Engineering College, Nagapattinam** and the details are as follows:

Title of the Seminar	: **DATA SECURITY IN CLOUD COMPUTING USING IDCRYPT**
Date & Session	: **25/05/2018 & AN**
Venue	: **SEMINAR HALL, DEPT. of CSE, E.G.S Pillay Engg College, Nagapattinam**

-: All are cordially Invited: -

Supervisor
Dr. M. CHINNADURAI
Professor
Department of CSE,
E. G. S. Pillay Engineering college,
Nagapattinam – 611002.

Professor & Head
Dr. M. CHINNADURAI
Professor
Department of CSE,
E. G. S. Pillay Engineering college,
Nagapattinam – 611002.

Copy to:

1. Department File / Notice Board /Staff and Student Circulation.

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)

NAGAPATTINAM - 611 002.
(An ISO 9001:2015 Certified Institution)
(Approved by AICTE, New Delhi, Approved by Govt.of Tamil Nadu)
(Affiliated to Anna University Chennai)
(Accredited by NAAC with 'A' Grade)
(NBA Accredited Programmes B.E - CSE,EEE & Mechanical))

TÜV NORD
TÜV India Private Ltd.
ISO 9001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ATTENDANCE PARTICULAR OF SEMINAR PRESENTATION

Name of the Scholar	: K. RAJU
Register Number	: 17144597127
Topic of the Seminar	: DATA SECURITY IN CLOUD COMPUTING USING IDCRYPT
Venue	: Seminar Hall, Dept. of CSE, EGS Pillay Engg College
Name of the Supervisor	: Dr. M. CHINNADURAI
Date & Session	: 25/05/2018 & AN

Sl. No.	Name of the Participant	Designation & Department	Name of the College	Signature
1	Dr. M. Chinnadurai	Prof.	EGSPEC	
2	S. MANIKANDAN	Asst.Prof/Sr	EGSPEC	
3	S. Kalidass	AP/CSE	EGSPEC	
4	S. Dhanalakshmi	AP/ECE	EGSPEC	
5	K. Krishnaram	AP/EEE	EGSPEC	
6	M. Rajakumaran	AP/CSE	EGSPEC	
7	R. G. Gokila	AP/IT	EGSPEC	
8	Dr. V. SIVARAMAN	Proy/Mech.	E.G.S.P.E.C.	
9	R. Lavanya	AP/IT	E.G.S.PEC	

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)
NAGAPATTINAM - 611 002.
(An ISO 9001:2015 Certified Institution)
(Approved by AICTE, New Delhi, Approved by Govt.of Tamil Nadu)
(Affiliated to Anna University Chennai)
(Accredited by NAAC with 'A' Grade)
(NBA Accredited Programmes B.E - CSE,EEE & Mechanical))

TUV NORD
TUV India Private Ltd.
ISO 9001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

10	G. Harinarayanan	AP/MECH	EGSPEC	30/8
11	A. BASKAR	ASST PROF	EGSPEC	A. Bh
12	S. Palani Murugan	ASST. PROF	EGSPEC	
13	Dr. N. Murali	Asso. Prof	EGSPEC	Murali
14	R ANAND BABU.	AP/IT	EGSPEC	
15	V H. Suresh	AP/IT	Egspec	

Signature of the Supervisor
Dr. M. CHINNADURAI

Dr. M.CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

ANNA UNIVERSITY, CHENNAI - 600 025

CENTRE FOR RESEARCH

1 7 1 4 4 5 9 7 1 2 7

MINUTES OF THE DOCTORAL COMMITTEE MEETING FOR CONFIRMATION OF PROVISIONAL REGISTRATION

The Doctoral Committee Meeting of the Ph.D. Scholar, Mr./Ms. **Raju K** (Reg.No. **17144597127/Part-Time**) was held on 28-05-18 at 11.00 A.M./~~P.M.~~* in the Department of Computer Science & Engineering the following members were present

Dr. Chinnadurai M (Supervisor & Convener)
Dr. K.Velmurugan (Member)
Dr. J.I.Sheeba (Member)

Mr./Ms. **Raju K** has successfully completed the following course works recommended by the Doctoral Committee. He/She has obtained the following grades in the course works.

Sl.No	Course Code	Course title	Credits	Category	Grade / Marks
1	17CP004	CLOUD COMPUTING	3	Elective	9
2	17CP105	DESIGN AND MANAGEMENT OF COMPUTER NETWORKS	4	Core	9
3	17MF023	RESEARCH METHODOLOGY	3	Elective	10
4	17CP202	SECURITY IN COMPUTING	3	Core	9
5					
6					
7					
8					
				CGPA	~~8.6~~ 9.23
		Comprehensive Examination :		✓ Pass / ~~Fail~~*	

COE signed result sheet of the course works should be duly attested by the Supervisor with seal

Dr. M. CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

The scholar had completed the first seminar presentation **DATA SECURITY IN CLOUD COMPUTING USING IDCRYPT** on **25.05.2018** to the faculty members and research scholars. The attendees list is enclosed herewith. The committee also evaluated the research work carried out by the scholar and satisfied / ~~not satisfied~~ with the performance of the scholar. Hence the Committee recommends / ~~not recommends~~ the confirmation of Provisional registration of the scholar in the Faculty of **Information and Communication Engineering** , and permits / ~~not permits~~ the scholar to proceed with his/~~her~~ research work.

Dom Irugan
Member
(Signature with Name)
Dr K. VELMURUGAN

Member
(Signature with Name)
Dr. J. I. SHEEBA

Supervisor
(Signature with Name and Seal)
Dr. M. CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

Joint Supervisor
(Signature with Name and Seal)
(if applicable)

* Strike off whichever is not applicable

CENTRE FOR RESEARCH

ANNA UNIVERSITY, CHENNAI-600 025

MINUTES OF THE DOCTORAL COMMITTEE MEETING FOR SUBMISSION OF SYNOPSIS

The Doctoral Committee Meeting of the Ph.D. Scholar, **Mr. Raju K (Reg.No.:17144597127)** was held on 19.08.2021 at 02:00-pm in the Seminar Hall, Department of CSE, E.G.S. Pillay Engineering College, Nagapattinam

The following members were present

1. Dr. Chinnadurai.M,Professor, Department of Computer Science and Engineering, E.G.S.Pillay Engineering College,Nagappattinam — Supervisor

2. **Dr. K.Velmurugan,Professor** — Member
   Department of Computer Science and Engineering
   Anjalai Ammal Mahalingam Engineering College Kovilvenni-614403

3. **Dr. J.I.Sheeba,Professor** — Member
   Department of Computer Science and Engineering
   Pondicherry Engineering College Pondicherry-605014

The Doctoral Committee critically reviewed the research work title **"IDENTITY BASED AUTHENTICATION IN CLOUD ENVIRONMENT USING DEEP LEARNING AND CRYPTOGRAPHIC TECHNIQUES"** carried out by Mr. Raju K and the contents of the draft Synopsis. The scholar had completed the second seminar presentation on 16.08.2021 to the faculty members and research scholars. The attendees list is enclosed herewith. The scholar has 01 publications from his research work in the journals listed in Annexure.

The Committee is satisfied with the research performance of the scholar and approves the Synopsis submission. The committee also recommends the panel of Indian and Foreign Examiners for the evaluation of the Thesis.

Member 19/08/21
(Signature with Name and date)
Dr. K. Velmurugan

J.I. Sheeba
Member 19/8/21
(Signature with Name and date)
(Dr. J.I. Sheeba)

19/08/2021
Supervisor
(Signature with Name date and seal)

Dr. M.CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

Forwarded

19/08/2021
Signature of the HOD/Director of the Centre of the Supervisor*
(Name and Seal)

Dr. M.CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

*Note :
If the supervisor is from non-recognized centre, HoD/Director of the centre of the joint supervisor should forward the minutes of the meeting.

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)
NAGAPATTINAM - 611 002.
(An ISO 9001:2015 Certified Institution)
(Approved by AICTE, New Delhi, Approved by Govt. of Tamil Nadu)
(Affiliated to Anna University Chennai)
(Accredited by NAAC with 'A' Grade)
(NBA Accredited Programmes B.E - CSE, EEE, Mech, IT, ECE, Civil))

TUV NORD
TUV India Private Ltd.
ISO 9001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

13/08/2021

SEMINAR NOTIFICATION

As per Anna University, Chennai Ph.D Regulations – 2017, there will be a seminar presentation by **Mr. K. RAJU (Register Number: 17144597127)** Ph.D Scholar of Department of Computer Science & Engineering, **E.G.S. Pillay Engineering College, Nagapattinam** and the details are as follows:

Title of the Seminar	: **DESIGN AND ANALYSIS OF DEEP LEARNING BASED IRIS RECOGNITION TECHNOLOGIES FOR CLOUD SECURITY**
Date & Session	: **16/08/2021 & FN**
Venue	: **SEMINAR HALL, DEPT. of CSE, E.G.S. Pillay Engg College, Nagapattinam**

-: All are cordially Invited: -

13/08/2021
Supervisor
Dr. M. CHINNADURAI
Professor
Department of CSE,
E. G. S. Pillay Engineering college,
Nagapattinam – 611002.

13/08/2021
Professor & Head
Dr. M. CHINNADURAI
Professor
Department of CSE,
E. G. S. Pillay Engineering college,
Nagapattinam – 611002.

Copy to:

1. Department File / Notice Board /Staff and Student Circulation.

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)

NAGAPATTINAM - 611 002.
(An ISO 9001:2015 Certified Institution)
(Approved by AICTE, New Delhi, Approved by Govt. of Tamil Nadu)
(Affiliated to Anna University Chennai)
(Accredited by NAAC with 'A' Grade)
(NBA Accredited Programmes B.E - CSE, EEE, Mech, IT, ECE, Civil))

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ATTENDANCE PARTICULAR OF SEMINAR PRESENTATION

Name of the Scholar	: K. RAJU
Register Number	: 17144597127
Topic of the Seminar	: DESIGN AND ANALYSIS OF DEEP LEARNING BASED IRIS RECOGNITION TECHNOLOGIES FOR CLOUD SECURITY
Venue	: Seminar Hall, Dept. of CSE, E.G.S. Pillay Engineering College
Name of the Supervisor	: Dr. M. CHINNADURAI
Date & Session	: 16/08/2021 & FN

Sl. No.	Name of the Participant	Designation & Department	Name of the College	Signature
1	Dr.M.Chinnadurai	Prof/CSE	EGSPEC	
2	Dr.S.MANIKANDAN	ASSOC.PROF HOD/IT	EGSPEC	
3	Dr.S.KANNAN	Professor	EGSPEC	Skann
4	Dr. K. Balasubramanian	ASP/CSE	EGSPEC	
5	Y.M.SURESH	AP/IT	EGSPEC	Y. M. Sur
6	Dr.M.Irshad Ahamed	ASP/ECE	EGSPEC	
7	K Nagalakshmi	ASP/IT	EGSPEC	
8	R. Anand Babu	Asst.Prof	EGSPEC	

E.G.S. PILLAY ENGINEERING COLLEGE
(Autonomous)
NAGAPATTINAM - 611 002.
(An ISO 9001:2015 Certified Institution)
(Approved by AICTE, New Delhi, Approved by Govt. of Tamil Nadu)
(Affiliated to Anna University Chennai)
(Accredited by NAAC with 'A' Grade)
(NBA Accredited Programmes B.E - CSE, EEE, Mech, IT, ECE, Civil))

TUV NORD
TUV India Private Ltd.
ISO 9001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

9	Dr. T. GANESAN	P/CSE	EGSPEC	
10	R. LAVANYA	AP/ IT	EGSPEC	
11	A. BASKAR	AP/CSE	EGSPEC	A. Bskr
12	M. MARKCO	AP/CSE	EGSPEC	
13	S. PRAVEEN KUMAR	AP/CSE	EGSPEC	
14	Dr. R. Manivannan	ASP/CSE	EGSPEC	
15	R. ANANDARAJ	ASP/EEE	EGSPEC	R. A
16	P.J. SURESH BABU	AP/EEE	EGSPEC	
17	S. JimHAWKINSON	AP/ECE	EGSPEC	
18	Dr. N. MURALI	ASP/CSE	EGSPEC	
19	S. Palani Murugan	AP/CSE	EGSPEC	
20	Dr. A. Sundar Raj	ASP/ECE	EGSPEC	

16/08/2024
Signature of the Supervisor
Dr. M. CHINNADURAI

Dr. M. CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

ANNA UNIVERSITY::CHENNAI - 25

Department of Computer Science and Engineering
E.G.S.Pillay Engineering College
Nagappattinam - 611 002

DH : AR6

Printed Date : 24-02-2022 10:17:37-am

Notification for Ph.D. Public Viva-Voce Examination

Name of the Scholar	:	Raju K
Registration Number	:	17144597127
Degree / Category	:	Ph.D. / Part-Time
Faculty	:	Information and Communication Engineering
Title of the Thesis	:	IDENTITY BASED AUTHENTICATION IN CLOUD ENVIRONMENT USING DEEP LEARNING AND CRYPTOGRAPHIC TECHNIQUES
Date and Time of Viva-Voce Examination	:	04.03.2022 & 12:00-pm
Venue	:	Seminar Hall A/C Department of CSE E.G.S. Pillay Engineering College, Nagapattinam Tamilnadu, 611002
Name and Address of the Supervisor	:	Dr.M.Chinnadurai Professor Department of Computer Science and Engineering E.G.S.Pillay Engineering College Nagappattinam - 611 002
Name and Address of the Joint Supervisor/Research Co-ordinator/Supervisor In-charge	:	Not Applicable
Online Meeting Details	:	Ph.D. Public Viva - K. Raju Friday, March 4 · 12:00 – 2:00pm Google Meet joining info Video call link: https://meet.google.com/fjk-ggmx-sxf Or dial: ?(US) +1 260-245-3674? PIN: ?337 663 157?#

All are cordially invited to attend the Ph.D. Public Viva-Voce Examination

Date : 24-02-2022
Place : E.G.S. Pillay Engineering College, Nagapattinam, Tamilnadu, 611002

24/02/2022
Signature of the Supervisor
(Name,date and seal)

Dr.M.CHINNADURAI, M.E.,Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

Signature of the Joint Supervisor / Research Co-ordinator / Supervisor In-charge
(Name,date and seal)
(If applicable)

24/02/2022
Signature of the HoD/Director of the Centre of the Supervisor
(Name,date and seal)

Dr.M.CHINNADURAI, M.E.,Ph.D.
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

Copy to:

Personal Secretary to Vice Chancellor, Anna University, Chennai - 25.
Personal Assistant to Registrar, Anna University, Chennai - 25.
The Controller of Examinations, Anna university, Chennai - 25.
The Additional Controller of Examinations, Anna university, Chennai - 25.
The Director, Centre for Research, Anna University, Chennai - 25.
The Director, Ramanujan Computing Centre, Anna University, Chennai - 25.
Dean / Director/ Principal of Government, Government Aided and Affiliated Engineering Colleges.
The Principal, E.G.S. Pillay Engineering College, Nagapattinam, Tamilnadu - 611 002
All Faculty Members, E.G.S. Pillay Engineering College, Nagapattinam - 611002.

CENTRE FOR RESEARCH

ANNA UNIVERSITY, CHENNAI-600 025

Proceedings of the Ph.D. Viva-Voce Examination of Mr.Raju.K held at 03:00 PM on 04.03.2022 in Seminar Hall A/C, Department of CSE, EGS Pillay Engineering College, Nagapattinam - 611002

The Ph.D. Viva-Voce Examination of Mr.Raju.K (Reg. No. 17144597127) on his/~~her~~ Ph.D. Thesis Entitled " IDENTITY BASED AUTHENTICATION IN CLOUD ENVIRONMENT USING DEEP LEARNING AND CRYPTOGRAPHIC TECHNIQUES " was conducted on 04.03.2022 at 03:00 PM in the Seminar Hall A/C, Department of CSE, EGS Pillay Engineering College, Nagapattinam - 611002.

The following Members of the Oral Examination Board were present:

1. Dr. K Vasudevan,Professor Emeritus,Department of Electronics Engineering,Cochin University of Science and Technology,Kalamassery Cochin Kerala - 682 022	Indian Examiner
2. Dr. R Manikandan,Associate Professor,Department of Computer Science and Engineering,Government College of Engineering Thanjavur,Thanjavur - 613 402	Subject Expert
3 Dr. Chinnadurai.M,Professor, Department of Computer Science and Engineering, E.G.S.Pillay Engineering College,Nagapattinam	Supervisor

The research scholar, Mr. Raju.K presented the salient features of his/~~her~~ Ph.D. work. This was followed by questions from the board members. The questions raised by the Foreign and Indian Examiners were also put to the scholar. The scholar answered the questions to the full satisfaction of the board members.

The corrections suggested by the Indian/Foreign examiner have been carried out and incorporated in the Thesis before the Oral examination.

Based on the scholars research work, his/~~her~~ presentation and also the clarifications and answers by the scholar to the questions, the board recommends that Mr.Raju.K be awarded Ph.D. degree in the Faculty of Information and Communication Engineering.

Indian Examiner

Subject Expert

04/03/2022

Supervisor

Dr. M.CHINNADURAI, M.E., Ph.D.,
Professor & Head
Department of CSE
E.G.S. Pillay Engineering College,
Nagapattinam - 611 002.

E.G.S. PILLAY ENGINEERING COLLEGE

(Autonomous)

Approved by AICTE, New Delhi | Affiliated to Anna University, Chennai
Accredited by NAAC with „A‟ Grade | Accredited by NBA (CSE, EEE, MECH, CIVIL, ECE, IT)
NAGAPATTINAM – 611 002

MASTER OF BUSINESS ADMINISTRATION

Full Time Curriculum and Syllabus

First Year – Second Semester

Course Code	Course Name	L	T	P	C	Maximum Marks		
						CA	ES	Total
Theory Course								
2002BA201	Operations Management	3	0	0	3	40	60	100
2002BA202	Financial Management	4	0	0	4	40	60	100
2002BA203	Marketing Management	3	0	0	3	40	60	100
2002BA204	Human Resource Management	3	0	0	3	40	60	100
2002BA205	Applied Operations Research	3	2	0	4	40	60	100
2002BA206	Business Research Methods	3	0	0	3	40	60	100
Laboratory Course								
2002BA207	Data Analysis and Business Modelling	0	0	4	2	50	50	100
2004BA208	Indian Ethos and Business Ethics	0	0	4	2	100	-	100
2004BA209	Life Skills II	0	0	2	1	100	-	100

L – Lecture | T – Tutorial | P – Practical | CA – Continuous Assessment | ES – End Semester

2002BA206	BUSINESS RESEARCH METHODS	L	T	P	C
		3	0	0	3

UNIT I INTRODUCTION 9 Hours

Business Research – Definition and Significance – the research process – Types of Research – Exploratory and causal Research – Theoretical and empirical Research – Cross –Sectional and time – series Research – Research questions / Problems – Research objectives – Research hypotheses – characteristics – Research in an evolutionary perspective – the role of theory in research.

UNIT II RESEARCH DESIGN AND MEASUREMENT 9 Hours

Research design – Definition – types of research design – exploratory and causal research design– Descriptive and experimental design – different types of experimental design – Validity of findings – internal and external validity – Variables in Research – Measurement and scaling – Different scales – Construction of instrument – Validity and Reliability of instrument.

UNIT III DATA COLLECTION 9 Hours

Types of data – Primary Vs Secondary data – Methods of primary data collection – Survey Vs Observation – Experiments – Construction of questionnaire and instrument – Validation of questionnaire – Sampling plan – Sample size – determinants optimal sample size – sampling techniques – Probability Vs Non–probability sampling methods.

UNIT IV DATA PREPARATION AND ANALYSIS 9 Hours

Data Preparation – editing – Coding –Data entry – Validity of data – Qualitative Vs Quantitative data analyses – Bivariate and Multivariate statistical techniques – Factor analysis – Discriminant analysis – cluster analysis – multiple regression and correlation – multidimensional scaling – Application of Statistical software for data analysis.

UNIT V REPORT DESIGN, WRITING &ETHICS IN BUSINESS RESEARCH 9 Hours

Research report – Different types – Contents of report – need of executive summary – chapterization – contents of chapter – report writing – report format – title of the report – ethics in research.

Total: 45 Hours

COURSE OUTCOMES:

After completion of the course, Student will be able to

1.Apply the concepts, types of research and problems while conducting research.
2.Use research on a scientific basis and select appropriate research design.
3.Make use of the various data collection methods and sampling techniques.
4.Manipulate the collected data using appropriate statistical tools for interpretation ofthe data
5.Produce the research report adopting the right tools for enhancing the quality of presentation.

REFERENCES:

1. Adrian Thornhill, Philip Lewis, Mark N. K. Saunders, Research Methods For Business Students, PEARSON, 2019.
2. Uma Sekaran and Roger Bougie, Research methods for Business, 7th Edition, Wiley India, New Delhi, 2016.
3. HK Dangi, Shruti Dewen, Business Research Methods, Cengage Learning, 2016
4. Mark N.K. Saunders, Philip Lewis, Adrian Thornhill, Research Methods for Business Students, Pearson; 7 edition, 2015.
5. C.R.Kothari, Research Methodology, New age International Publisher Ltd., 2014
6. Donald R. Cooper, Pamela S. Schindler and J K Sharma, Business Research methods, 11th Edition, Tata Mc Graw Hill, New Delhi, 2012.
7. William G Zikmund, Barry J Babin, Jon C.Carr, Atanu Adhikari, Mitch Griffin, Business Research methods, A South Asian Perspective, 8th Edition, Cengage Learning, New Delhi, 2012.
8. Zikmund, Babin, Carr, Adhikari, Griffin, Business Research Methods – A South Asian Perspective, Cengage Learning, 2012.
9. Alan Bryman and Emma Bell, Business Research methods, 3rd Edition, Oxford University Press, New Delhi, 2011.
10. Naval Bajpai, Business Research Methods, Pearson, 2011.

1901MA404	OPERATIONS RESEARCH	L	T	P	C
		3	0	0	3

MODULE I LINEAR PROGRAMMING 9 Hours

Linear programming – Examples from industrial cases, formulation & definitions, Matrix form. Basic concepts, Special cases – infeasibility, unboundedness, redundancy and degeneracy, Sensitivity analysis. Simplex Algorithm–slack, surplus & artificial variables, computational details, big-M method, identification and resolution of special cases through simplex iterations. Duality – formulation, results, fundamental theorem of duality, dual-simplex and primal-dual algorithms.

MODULE II TRANSPORTATION AND ASSIGNMENT PROBLEMS 9 Hours

TP - Examples, Definitions – decision variables, supply & demand constraints, formulation, Balanced &unbalanced situations, Solution methods – NWCR, minimum cost and VAM, test for optimality (MODI method), degeneracy and its resolution. AP-Examples, Definitions–decision variables, constraints formulation, Balanced &unbalanced situations, Solution method–Hungarian, test for optimality (MODI method), degeneracy &its resolution.

MODULE III PERT – CPM AND INVENTORY CONTROL 9 Hours

Project definition, Project scheduling techniques – Gantt chart, PERT & CPM, Determination of critical paths, Estimation of Project time and its variance in PERT using statistical principles, Concept of project crashing/time-cost trade-off Inventory Control: Functions of inventory and its disadvantages, ABC analysis, Concept of inventory costs, Basics of inventory policy (order, lead time, types), Fixed order-quantity models–EOQ, POQ & Quantity discount models. EOQ models for discrete MODULEs, sensitivity analysis and Robustness.

MODULE IV QUEUING THEORY 9 Hours

Definitions– queue (waiting line), waiting costs, characteristics (arrival, queue, service discipline) of queuing system, queue types (channel vs. phase). Kendall‟s notation, Little‟s law, steady state behavior, Poisson‟s Process & queue, Models with examples - M/M/1 and its performance measures; M/M/m and its performance measures.

MODULE V SIMULATION METHODOLOGY 9 Hours

Definition and steps of simulation, random number, random number generator, Discrete Event System Simulation–clock, event list, Application in Scheduling, Queuing systems and Inventory systems.

TOTAL: 45 HOURS

REFERENCES:

1. Operations Research: An Introduction. H.A. Taha.
2. Linear Programming. K. G. Murthy.
3. Linear Programming. G. Hadley.
4. Principles of OR with Application to Managerial Decisions. H.M. Wagner.
5. Introduction to Operations Research. F.S. Hiller and G.J. Lieberman.
6. Elements of Queuing Theory. Thomas L. Saaty.
7. Operations Research and Management Science, Hand Book: Edited By A. Ravi Ravindran.
8. Management Guide to PERT/CPM Wiest & Levy.

E.G.S. PILLAY ENGINEERING COLLEGE (AUTONOMOUS)

NAGAPATTINAM – 611 002. TAMILNADU, INDIA
Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai
(Accredited by NAAC with 'A' Grade and NBA)
Email: principal@egspec.org website: www.egspec.org Ph: 04365-251112

PRESENCE OF ETHIC COMMITTEE

The Research and Development Cell of EGSPEC envisages integrated growth of rural scholars and communities with advanced and socially contextualized multi-disciplinary research projects and programmes. EGSPEC also brings up theoretical research by the faculty and students in their respective disciplines. The institution undertakes to comply with the provisions formulated by Anna University, Chennai for the promotion of Academic Integrity and Prevention of Plagiarism. The Research Coordination Committee of the college promotes intellectual honesty and integrity among researchers and encourage ideal research habits. The practice of code of ethics in research is promoted among the faculty to deter them from research and publication misconducts. The RCC will examine research publications for any violation of code of conduct of the college for suitable action. Accordingly, the college will take all incidents of research misconduct seriously, and will ensure that the procedure for the inquiry, investigation and adjudication of any misconduct are just for all parties involved.

ATTESTED

Dr. S. RAMABALAN, M.E., Ph.D.,
PRINCIPAL
E.G.S. Pillay Engineering College,
Thethi, Nagore - 611 002.
Nagapattinam (Dt) Tamil Nadu.

PRINCIPAL

E.G.S. PILLAY ENGINEERING COLLEGE (AUTONOMOUS), NAGAPATTINAM

RESEARCH AND DEVELOPMENT CELL

MEMBERS OF THE R&D CELL AND RCC WITH RESPONSIBILITIES

Sl.No.	Name of the Faculty	Designation / Roll in R&D	Responsibilities
1.	Dr.S.Ramabalan	Principal Professor-Mechanical Engineering	Monitoring and mentoring the overall activities of R & D Cell
2.	Dr.V.Mohan	Vice Principal / Director-Academics Professor - Dept. of EEE	Advisor to R & D Cell
3.	Dr.EdwardAnand.E	Coordinator –R&D Cell Professor-Science & Humanities	Liaising with departments in informing the R&D activities through Department Coordinators. Reviewing project proposals to be sent to funding agencies
4.	Dr.M.K.Mishra	Professor-S&H	IPR and Patenting
5.	Dr.V.Sivaraman	Director-Industry Institute Partnership Cell Professor-Mechanical Engineering	Liaising with Industry in acquiring Projects and Training for students
6.	Dr.Ganesan@Subramanian	Convener-R&D Cell	Liaising with departments in preparation of project proposals Maintaining R&D related documents

ATTESTED

Dr. S. RAMABALAN, M.E., Ph.D.,
PRINCIPAL
E.G.S. Pillay Engineering College,
Thethi, Nagore - 611 002.
Nagapattinam (Dt) Tamil Nadu.

7.	Dr.IrshadAhamed	Department Coordinator-ECE	Liaising with R&D cell in informing the R&D activities to the Department. Maintaining the R&D related files in the department
8.	Dr.Vijayakumar.M	Department Coordinator-EEE	
9.	Dr.Chockalingam.S	Department Coordinator-Mech	
10.	Mrs. K.Nagalakshmi	Department Coordinator-IT	
11.	Dr.Ganesan.T	Department Coordinator-CSE	
12.	Mr.ShyamSundar	Department Coordinator-Civil	
12.	Dr. Arunkumar.P	Department Coordinator-MCA	
13.	Mr.Sathish	Department Coordinator-MBA	
14.	Dr.Charles.S	Department Coordinator-S&H	

Director/R&D

Dr. EDWARD ANAND.E, M.Tech., Ph.D.,
Director-Research & Development,
E.G.S.Pillay Group of Institutions,
Nagapattinam - 611 002

ATTESTED

Dr. S. RAMABALAN, M.E., Ph.D.,
PRINCIPAL
E.G.S. Pillay Engineering College,
Thethi. Nagore - 611 002.
Nagapattinam (Dt) Tamil Nadu.

E.G.S. PILLAY ENGINEERING COLLEGE (AUTONOMOUS)

NAGAPATTINAM – 611 002. TAMILNADU, INDIA
Approved by AICTE, New Delhi, Affiliated to Anna University, Chennai
(Accredited by NAAC with 'A' Grade and NBA)
Email: principal@egspec.org website: www.egspec.org Ph: 04365-251112

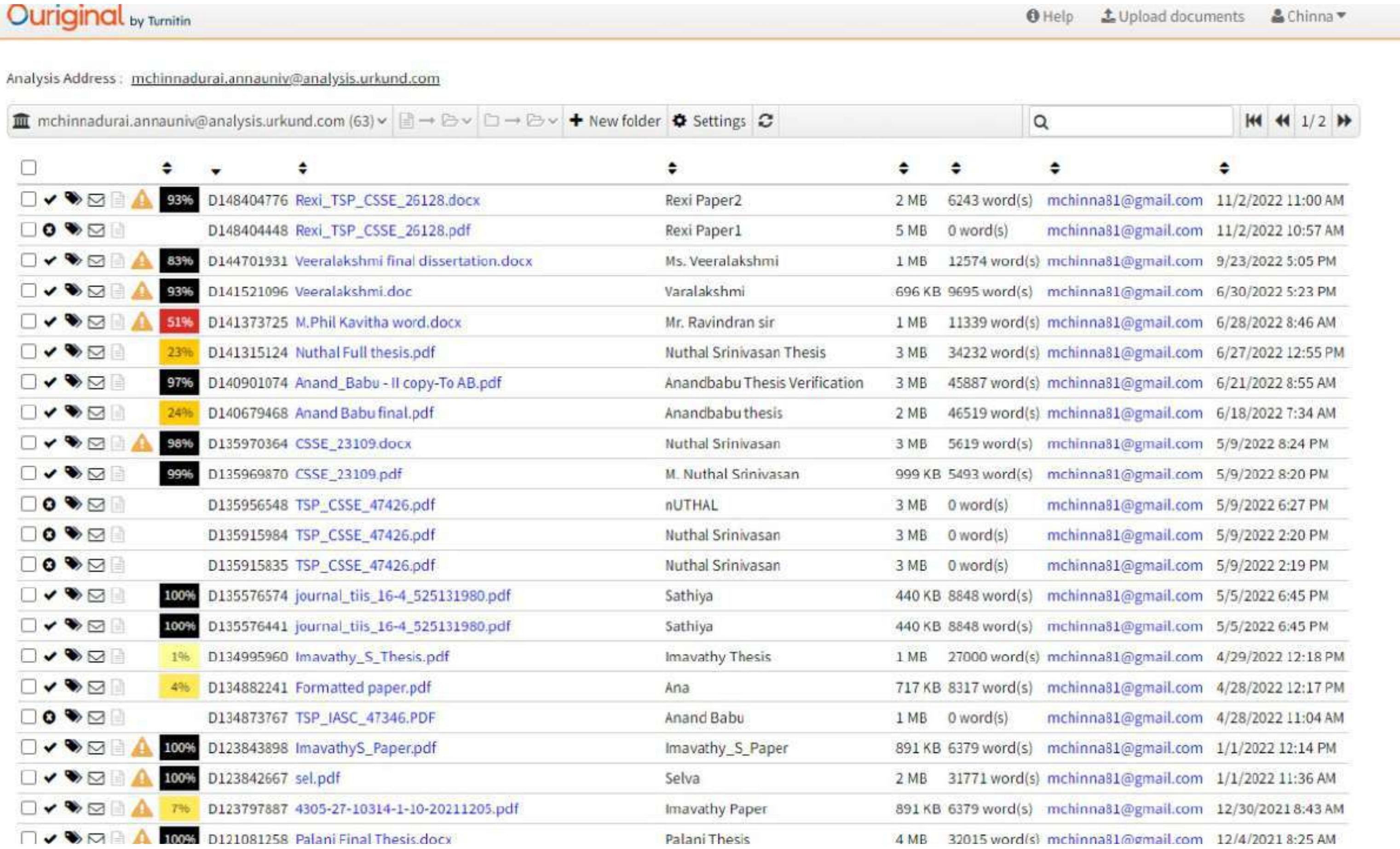
Plagiarism check through software

E.G.S. Pillay Engineering College has clearly defined code of ethics and guidelines to check malpractices and plagiarism in research. Currently, institution has 35 research supervisors recognized by Anna University. Everyone have a separate login ID for online access to Urkund Plagiarism check software by Centre for Research, Anna University, Chennai. Research Scholar should submit the copy of Urkund plagiarism check report to the concern supervisor for a paper before submitting to the journal/conference. Apart from the recommended software Plagiarism on thesis, research paper, and assignments can be checked online at free of cost. The important top 10 free Plagiarism Checkers are DupliChecker.com, Plagiarismsoftware.net, Plagiarism CheckerX, PlagScan.com Plagramme.com, Smallseotools.com, Plagiarisma.net, Quetext.com, Easybib.com and Noplag.com. The research scholar Synopsis/ Thesis should accompany Urkund plagiarism report while submitting same to Center for Research, Anna University. All Post Graduate projects, dissertation, thesis should include the copy of Plagiarism check report.

ATTESTED

Dr. S. RAMABALAN, M.E., Ph.D.,
PRINCIPAL
E.G.S. Pillay Engineering College,
Thethi, Nagore - 611 002.
Nagapattinam (Dt) Tamil Nadu.

PRINCIPAL



Analysis Address : mchinnadurai.annauniv@analysis.urkund.com

mchinnadurai.annauniv@analysis.urkund.com (63)

□			↕	▾	↕	↕	↕	↕	↕	↕
□	✓	⚠	93%	D148404776	Rexi_TSP_CSSE_26128.docx	Rexi Paper2	2 MB	6243 word(s)	mchinna81@gmail.com	11/2/2022 11:00 AM
□	✖			D148404448	Rexi_TSP_CSSE_26128.pdf	Rexi Paper1	5 MB	0 word(s)	mchinna81@gmail.com	11/2/2022 10:57 AM
□	✓	⚠	83%	D144701931	Veeralakshmi final dissertation.docx	Ms. Veeralakshmi	1 MB	12574 word(s)	mchinna81@gmail.com	9/23/2022 5:05 PM
□	✓	⚠	93%	D141521096	Veeralakshmi.doc	Varalakshmi	696 KB	9695 word(s)	mchinna81@gmail.com	6/30/2022 5:23 PM
□	✓	⚠	51%	D141373725	M.Phil Kavitha word.docx	Mr. Ravindran sir	1 MB	11339 word(s)	mchinna81@gmail.com	6/28/2022 8:46 AM
□	✓		23%	D141315124	Nuthal Full thesis.pdf	Nuthal Srinivasan Thesis	3 MB	34232 word(s)	mchinna81@gmail.com	6/27/2022 12:55 PM
□	✓		97%	D140901074	Anand_Babu - II copy-To AB.pdf	Anandbabu Thesis Verification	3 MB	45887 word(s)	mchinna81@gmail.com	6/21/2022 8:55 AM
□	✓		24%	D140679468	Anand Babu final.pdf	Anandbabu thesis	2 MB	46519 word(s)	mchinna81@gmail.com	6/18/2022 7:34 AM
□	✓	⚠	98%	D135970364	CSSE_23109.docx	Nuthal Srinivasan	3 MB	5619 word(s)	mchinna81@gmail.com	5/9/2022 8:24 PM
□	✓		99%	D135969870	CSSE_23109.pdf	M. Nuthal Srinivasan	999 KB	5493 word(s)	mchinna81@gmail.com	5/9/2022 8:20 PM
□	✖			D135956548	TSP_CSSE_47426.pdf	nUTHAL	3 MB	0 word(s)	mchinna81@gmail.com	5/9/2022 6:27 PM
□	✖			D135915984	TSP_CSSE_47426.pdf	Nuthal Srinivasan	3 MB	0 word(s)	mchinna81@gmail.com	5/9/2022 2:20 PM
□	✖			D135915835	TSP_CSSE_47426.pdf	Nuthal Srinivasan	3 MB	0 word(s)	mchinna81@gmail.com	5/9/2022 2:19 PM
□	✓		100%	D135576574	journal_tiis_16-4_525131980.pdf	Sathiya	440 KB	8848 word(s)	mchinna81@gmail.com	5/5/2022 6:45 PM
□	✓		100%	D135576441	journal_tiis_16-4_525131980.pdf	Sathiya	440 KB	8848 word(s)	mchinna81@gmail.com	5/5/2022 6:45 PM
□	✓		1%	D134995960	Imavathy_S_Thesis.pdf	Imavathy Thesis	1 MB	27000 word(s)	mchinna81@gmail.com	4/29/2022 12:18 PM
□	✓		4%	D134882241	Formatted paper.pdf	Ana	717 KB	8317 word(s)	mchinna81@gmail.com	4/28/2022 12:17 PM
□	✖			D134873767	TSP_IASC_47346.PDF	Anand Babu	1 MB	0 word(s)	mchinna81@gmail.com	4/28/2022 11:04 AM
□	✓	⚠	100%	D123843898	ImavathyS_Paper.pdf	Imavathy_S_Paper	891 KB	6379 word(s)	mchinna81@gmail.com	1/1/2022 12:14 PM
□	✓	⚠	100%	D123842667	sel.pdf	Selva	2 MB	31771 word(s)	mchinna81@gmail.com	1/1/2022 11:36 AM
□	✓	⚠	7%	D123797887	4305-27-10314-1-10-20211205.pdf	Imavathy Paper	891 KB	6379 word(s)	mchinna81@gmail.com	12/30/2021 8:43 AM
□	✓	⚠	100%	D121081258	Palani Final Thesis.docx	Palani Thesis	4 MB	32015 word(s)	mchinna81@gmail.com	12/4/2021 8:25 AM

Ouriginal

Document Information

Analyzed document	Anand_Babu - II copy-To AB.pdf (D140901074)
Submitted	6/21/2022 8:55:00 AM
Submitted by	
Submitter email	mchinna81@gmail.com
Similarity	2%
Analysis address	mchinnadurai.annauniv@analysis.urkund.com

Sources included in the report

SA	**Anna University, Chennai / Formatted paper.pdf** Document Formatted paper.pdf (D134882241) Submitted by: mchinna81@gmail.com Receiver: mchinnadurai.annauniv@analysis.urkund.com	⊞	3
SA	**Anna University, Chennai / Anand Babu R-1-18194591230.docx** Document Anand Babu R-1-18194591230.docx (D135638144) Submitted by: cfrsynopsis@gmail.com Receiver: cfrsynopsis.annauniv@analysis.urkund.com	⊞	3
W	URL: https://libgen.ggfwzs.net/book/74240522/e250fe Fetched: 6/21/2022 8:56:00 AM	⊞	1
W	URL: https://downloads.hindawi.com/journals/scn/2021/9947059.pdf Fetched: 1/4/2022 3:19:56 PM	⊞	2
W	URL: https://content.iospress.com/articles/informatica/infor457 Fetched: 5/25/2022 1:52:20 PM	⊞	1
SA	**College of Engineering, Pune / Paper Published Update 3-converted.pdf** Document Paper Published Update 3-converted.pdf (D53209344) Submitted by: aparna.comp@coep.ac.in Receiver: aparna.comp.coep@analysis.urkund.com	⊞	1

Entire Document

TOWARDS EFFECTIVE ENSEMBLE CLASSIFICATION
FOR ANOMALY-BASED
INTRUSION DETECTION
A THESIS Submitted by ANAND BABU R
in partial fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING ANNA UNIVERSITY CHENNAI 600 025
JUNE 2022
ii ANNA UNIVERSITY CHENNAI 600 025 BONAFIDE CERTIFICATE The research work embodied in the present Thesis entitled ‘
TOWARDS EFFECTIVE ENSEMBLE CLASSIFICATION FOR ANOMALY-BASED INTRUSION DETECTION’ has been carried out in the

Department of Computer Science and Engineering, E.G.S. Pillay Engineering College, Nagapattinam.
The work reported herein is original and does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion or to any other scholar. I understand the University's policy on plagiarism and declare that the thesis and publications are my own work, except where specifically acknowledged and has not been copied from other sources or been previously submitted for award or assessment.
ANAND BABU R Dr. S. KANNAN RESEARCH SCHOLAR SUPERVISOR Professor
Department of
Computer Science and Engineering
E.G.S. Pillay Engineering College Nagapattinam – 611 002.
iii

ACKNOWLEDGEMENT I
wish to record my deep sense of gratitude and profound thanks to
College Chairman Smt.S.Jothimani, Secretary Shri.S.Senthil Kumar, Joint Secretary Shri.S.Sankar Ganesh, Advisor Chev.Dr.S.Paramesvaran, E.G.S. Pillay Group of Institutions, Nagapattinam, Principal Dr.S.Ramabalan, E.G.S. Pillay Engineering College Nagapattinam and
my research supervisor Dr.S.Kannan, Professor,
Department of Computer Science and Engineering,
E.G.S. Pillay Engineering College, Nagapattinam
for his keen interest, inspiring guidance, and
constant encouragement with my work during all stages, to bring this thesis to fruition. I am extremely indebted to my research Doctoral Committee members Dr.N.Nandhagopal, Associate Professor, Department of Electronics and Communication Engineering, Excel Engineering College, Namakkal, and Dr.E.Sivasankar, Assistant Professor, Department of Computer Science and Engineering,
National Institute of Technology, Tiruchirappalli
for their valuable suggestions and support during the course of my research.
I would like to
express my sincere thanks to Dr.M.Chinnadurai, Professor and Head, Department of Computer Science and Engineering, and Dr.S.Manikandan, Associate Professor and Head,
Department of Information Technology, E.G.S. Pillay Engineering College, Nagapattinam.
I
thank the faculty and non-teaching staff members of the Department of Information Technology for their valuable support throughout the course of my research work.
I also thank almighty, parents, family members, and friends. ANAND BABU R

iv ABSTRACT

As information technology rolls out, the applications of the Internet continue to impact our daily routines including communication, e- commerce, entertainment, e-learning, etc. The advent of
computing and communicating devices as well as the infiltration of intrusive actions and hacking tools into the networks make data communication increasingly vulnerable. Generally, an intrusion would cause a loss of confidentiality, integrity, and availability (CIA triad) of information. Also, it enables unauthorized exploitation of network
resources or renunciation of network services. Therefore,
the inevitability of network security has drawn substantial interest from industries and academia around the world. Nevertheless, the utilization of different security applications such as firewalls, data encryption, user authentication, and malware protection methods, numerous intrusion detection frameworks have been developed at many organizations, and some of them have also been applied on an investigational basis in some active applications scenarios to
sense and classify intrusive activities.
An Intrusion Detection System (IDS) is widely employed to detect cyberattacks preferably in real-time and to protect the valuable information of the users. Albeit, numerous Machine Learning (ML) algorithms
have been proposed to improve the performance of IDS, it is a challenge to process massive unrelated and redundant information in current big data
environments. This research proposes an Intelligent Classifier using Ensemble Technique (ICET) to increase the accuracy, attack detection rate, and reduce the false alarm rate in classifying the intrusive activities significantly. The proposed ICET includes two elements: (i) feature selection module; and (ii) ensemble classifier.
To cope with high dimensional feature-rich inbound traffic in large networks,
the feature selection module exploits a Correlation- based Feature Selection (CFS) algorithm to select the appropriate features. It
v calculates the correlation between the

features and selects the optimal subset for training and testing
phases.

Besides, it exploits the optimized RelieF algorithm to calculate the quality of attributes. The attributes with a low- quality index are eliminated to reduce the dimensionality of the feature space. The performance of the proposed feature selection approach is further enhanced by integrating CFS with Bat-inspired Optimization (BIO) algorithm. This integration (hereafter called BIOCFS) is embedded in an ensemble classifier to increase the performance of the IDS. This study proposes an ensemble classifier that includes three different classifiers including Balanced Forest (BF), Random Forest (RF), and C4.5 decision tree. The BF exploits the Forest by Penalizing Attributes (FPA) algorithm to construct a set of highly balanced and accurate decision trees. The RF classifier integrates the concept of bagging and random subspace algorithms which selects features arbitrarily at the node level. C4.5
is one of the conventional classifiers established to generate a decision tree from a dataset using the Iterative Dichotomiser 3 (ID3) algorithm. This algorithm determines the optimal split to increase the gain ratio by visiting each node in the decision tree. The
proposed ensemble classifier exploits a Voting Mechanism (VM) using an Average of Probabilities (AoP) rule to combine the results from base classifiers effectively. The established ICET supports handling unbalanced and multi- class datasets with higher accuracy. This model is trained and assessed using two real-world datasets such as
Network Security Laboratory-Knowledge Discovery and Data Mining (NSL-KDD)
and Canadian Institute for Cybersecurity–IDS 2017 (CIC-IDS 2017) using Weka 3.8.3 workbench. The empirical analysis demonstrates that the proposed ICET model outperforms other related approaches in terms of appropriate evaluation metrics such as classification accuracy, precision, recall, F-measure, the false alarm rate, and
vi the attack detection rate. More decisively, the ICET model along with BIOCFS takes less time for model building and testing related to other individual and ensemble classifiers. The integration of BIO and CFS with the proposed ensemble classifier reduces the time taken for the training and testing process from 139.43s and 12.25s to 19s and 12s when applied to NSL- KDD, correspondingly. In the case of the CIC-IDS 2017 dataset, it decreases the time consumption of the training and testing process from 143s and 14s to 47s and 18s, respectively. More precisely, the ICET is a very viable IDS model for identifying and classifying intrusion in networks.
vii

xviii LIST OF SYMBOLS AND ABBREVIATIONS AIDS - Anomaly-based Intrusion Detection system ANN - Artificial Neural Network AFS - Assessment Function ADR - Attack Detection Rate - Attribute level AoP - Average of Probabilities BF - Balanced Forest BIOCFS - Bio-Inspired Optimizer with Correlation-based CIC-IDS - Canadian Institute for Cybersecurity–IDS CART - Classification and Regression Trees CIA - Confidentiality, Integrity, and Availability CFS - Correlation-based Feature Selection CSEW - Crime Survey for England and Wales DT - Decision Tree DoS - Denial-of-Service DBSCAN - Density-Based Spatial Clustering of Applications with Noise EC - Ensemble Classifier ENML - Ensemble ML Classifiers FAR - False Alarm Rate Feature Selection FSP - Feature selection process FPA - Forest by Penalizing Attributes - Gain ratio HIDS - Host-based Intrusion detection system IG Information Gain

xix ICET - Intelligent Classifier using Ensemble Technique IDS - Intrusion Detection System ID3 - Iterative Dichotomiser 3 KNN - K-Nearest Neighbor ML - Machine Learning - Maximum fitness value - Maximum values of feature - Minimum values of feature MOGA - Multi-Objective Genetic Algorithm MLP - Multiple Layer Perceptron NB - Naïve Bayes NIST - National Institute of Standards and Technology NSL- KDD - Network Security Laboratory-Knowledge Discovery and Data Mining NIDS - Network-based intrusion detection¯ - Normalized outcome ONS - Office for National Statistics ORelieF - Optimized RelieF PART - Partial Decision List PSO - Particle Swarm Optimization PCA - Principal Component Analysis RF - Random Forest SIDS - Signatures-based Intrusion Detection System SVM - Support Vector Machine VM - Voting Mechanism

1 CHAPTER 1 INTRODUCTION 1.1 OVERVIEW Finance, energy, medical, manufacturing, telecom, and media are just some of the industries that have been revolutionized by the proliferation of digital computing and communication technologies. With this accelerating digitizing trend in the world economy, threats and cyberattacks have also become a ubiquitous, all-pervasive phenomenon. Spurred by the mushrooming of smart maneuvers, the swift growth of intrusive incidents and hacking tools makes communication networks increasingly vulnerable (Jing et al. 2022; Sarker et al.2020; Meryem & Ouahidi 2020). Due to the rapid development and implementation of the fast-evolving Internet-of-Things (IoT), assailants have great motivation to execute well-orchestrated cyberattacks. Moreover,

the increased connectivity of smart portable devices provides greater access to unknown users and makes it easier for penetrators to circumvent their credentials.

To sneak into the network, intruders might intentionally use the

undiscovered vulnerabilities

of the computer network and create various threats, which results in revealing sensitive data, alteration of data, or the maximum complete data loss in enterprises and organizations (

Rizwan et al. 2022). Generally, an intrusion would cause a loss of the CIA triad, unauthorized exploitation of resources, and renunciation of network facilities. Currently, highly-proficient intruders can exploit the susceptibilities of

2 various networked applications. In the meantime, the jeopardy of security breaches escalates vividly as an application or software susceptibility always remains without a patch. By taking advantage of such undiscovered susceptibility (known as a zero-day attack), attackers gain access to a target system and can monitor or steal valuable data. It is often very difficult for a network engineer and network administrator to identify zero-day attacks with conventional protective tools since the signature pattern of these attacks is anonymous (Singh et al. 2019). A single illicit infiltration may generate a cascading failure in network systems (Wang et al. 2019). Users have much to lose their sensitive information, privacy, control of their resources, and perhaps theft of their identities. As such, the deliberate or unintentional disclosures of sensitive information can lead to disastrous impacts which makes it critical to develop dependable security procedures. Conversely, this could cause undesirable network complexities, specifically with respect to communication delay. As the era of the Internet has radically revolutionized numerous domains, the prevailing network security mechanisms are frail and ineffective (Pekaric et al. 2021; Jing et al. 2022). Most of the existing IDSs can only recognize the acquainted cyberattacks. The primary focus of a network security system is the balanced protection of the confidentiality, integrity, and availability of information (Rizwan et al. 2022). Also, it ensures improved credibility of the network service against cyberattacks. Communication

and data integrity services are related to the authenticity, accuracy, non-corruptibility, and reliability of transactions between

devices. This service must certify the correctness of the system hardware and firmware, and it should be secure against an illegal alteration of information as well as tags. Data confidentiality

is about defending data against illegitimate exposure.

The disclosure of sensitive information to unapproved users is a compromise that this service should secure.

Availability is defending the performance of support systems and

3 certifying data is copiously accessible at the point in time when it is requested by its customers. Traditional network security approaches including user authentication, data encryption, firewalls, malware prevention techniques, or access control mechanisms are no longer alone able to detect sophisticated cyberattacks against networks (Li & Liu 2021). Thus, the need to sense new cyber threats and anomalies is inexorable. IDS

is a retrofit approach for creating a protective shield against unauthorized assaults in current network traffic while enabling them to execute in their "open" mode so as to stimulate user productivity. The key goal
of IDS is to identify unauthorized use, abuse, and misuse of network resources in real-time of both
inside attackers (i.e., ratified users who target to
abuse their rights) and outside invaders. Solving the issues in IDS is a challenging task owing to the rapid development of assorted network elements, the overheads related to the fast-paced malicious activities, and the difficulty of extracting anomalous patterns from big data contaminated with catastrophic cyber threats (Mahfouz et al. 2020).

The key idea of IDSs is based on the assumption that the activities of an attacker will deviate remarkably from that of a genuine user; consequently, several illegal behaviors are noticeable. To
date, machine learning algorithms such as linear discriminant analysis, support vector machine (SVM), decision trees (DT), classification and regression trees (CART), random forest (RF), principle component analysis (PCA), artificial neural network (ANN), etc., are extensively used in intrusion detection applications (Saranya et al. 2020).

The wide variety of attacks and attributes of network traffic imposes considerable challenges for ML-based IDSs as they increase the problem space and consume more time and computational overhead.

The class imbalance of a high-dimensional dataset is a significant issue in ML-based IDS in which one of the classes has 4 more instances than other classes (Desuky & Hussain 2021). This problem may mislead the classification algorithm and be biased toward the majority class (Song et al. 2019; Wang & Yao 2012). Current datasets contain numerous unrelated and redundant features (Danasingh et al. 2020). These inherent characteristics of datasets may reduce the effectiveness of the IDS in generating credible solutions. Therefore, it is required to reduce the dimension of the datasets by finding suitable attribute subsets and solving the class imbalance problem. If the intruders become more sophisticated, unknown and obfuscated attacks are developed easily and the threat of
critical systems being compromised extensively upsurges quickly. In order to identify and manage new intrusive actions, we need efficient IDSs with good performance.

To make matters worse, intruders can use open-source invasive tools or exploits the dark web. It does not come as a surprise that malware and attacks become more and more smart, stealthy, and strong against conventional security systems. Recent insidious attacks such as Mirai, Wannacry, Stuxnet, etc., and their consequences reveal the perseverance for developing smart intrusion detection systems and tools to detect newfound attacks (Yaacoub et al. 2020). Of late, several intrusion detection tactics have been developed in the literature against cyberattacks and an intelligent IDS has become an indispensable part of the security architecture in almost every organization. This research proposes an intelligent classifier using ensemble learning to increase the accuracy, attack detection rate (ADR), and reduce the false alarm rate (FAR) in classifying the intrusive activities significantly. The proposed ICET includes two elements including a feature selection module and an ensemble classifier. To cope with high dimensional data traffic, the feature selection module exploits a CFS algorithm to select the appropriate features. The proposed feature selection module calculates the correlation between
5 attributes and selects the optimal subset for training and testing
processes. Besides, it exploits the optimized RelieF algorithm to calculate the quality of attributes. The attributes with a low-quality index are eliminated to reduce the dimensionality of the feature space. The performance of the proposed feature selection approach is further enhanced by integrating CFS with BIO. This integration (i.e., BIOCFS) is embedded in the proposed ensemble classifier to increase the performance of the IDS. The proposed ensemble classifier includes three different classifiers including BF, RF, and C4.5 based on the voting mechanism through the rule of AoP. The BF classifier is a decision forest that uses a decision forest algorithm called FPA. This classifier aims to construct a set of highly balanced and precise decision trees by exploiting the potential of all non-class features extant in a dataset. RF classifier integrates the concept of random subspace and bagging algorithms which selects features arbitrarily at the node level. C4.5
is one of the conventional classifiers developed to construct a decision tree from a dataset by applying the ID3 algorithm. This algorithm determines the optimal division to increase the gain ratio by considering each node in the tree. The proposed classifier exploits a VM using the AoP rule to solve the multi-label problem in the classification task. The established ICET model aids to handle unbalanced and multi-class datasets with higher accuracy. The classifier is trained and assessed by applying two real-world datasets such as NSL-KDD and CIC-IDS 2017 using Weka 3.8.3 workbench. 1.1.1 Cyberattacks against network infrastructure Security flaws or vulnerabilities in computer systems and communication networks are ubiquitous. Hence, these systems are inherently prone to security threats. Even the resilient deterrent controls may be overwhelmed by the extraordinary speed, scale, intensity, and preeminence of threats enabled by global interconnectivity and swift technological

6 transformation. Any unapproved cybercrime intended for violating the security policy of a network and leading to harm, interruption of the services or access to the data of the specific resource is known as a cyberattack (Motsch et al. 2020). The siloed computing systems such as information technology, mobile computing, healthcare, production, water, gas, transportation, etc. are currently assimilated to construct a system-of-systems, possibly divulging a terrific attack surface. Due to the proliferation of IoT and the impending global rollout of 5G cellular networks, this umbrella of tightly coupled systems is getting convoluted and prodigious. Hence, cybersecurity is a continued and progressive warfare on a multilayered, multidimensional, and asymmetric battlespace. Denial-of-Service (DoS), Worm, Virus, Trojan horse, Sniffer, logical bomb, and Botnet are the most significant cyberattacks in the network scenarios as illustrated in Figure 1.1 (Li & Liu 2021). Due to the DoS attacks the ratified users' access to the network is lost. Indeed, the adversary from one point starts immersing the target systems in innumerable packets and hindering the authorized data flow. This averts any system from availing of the Internet service or collaborating with other devices. In another technique, known as pervasive DoS, rather than instigating a cyberattack from a single source, they target a large number of decentralized systems concurrently (Mahjabin et al. 2017). This is often achieved through transmissible worms and reproducing them on numerous systems to attack the target. A virus contaminates digital files, which are usually operable databases, by injecting a copy of it into those files. By inserting infected files into storage devices, these variants operate and enable the virus to affect other files. The Trojan horse holds a hazardous program and normally resembles a useful code that the user is eager to execute (Mat et al. 2021). Sniffer is also a malicious code that snoops on transmitted data and seeks
7 particular data like passwords by analyzing every message in the data stream (Zhao et al. 2016). A logic bomb is also a malicious threat that is activated when a logical constraint is satisfied (e.g., after a certain number of communications have been carried out, or on an exact date (i.e., time bomb) (Li & Liu 2021). In such a situation, the program inevitably implements some detrimental activities. A botnet is a collection of distantly controlled infested systems, which is employed to disseminate malware, synchronize cyberattacks, spam, and snip information (Soe et al. 2020). Botnets are executed confidentially on the target system, permitting the illegal user to distantly control the target system to realize their malevolent objectives. They are also called electronic soldiers. Figure 1.1 Various attacks on information system
8 1.1.2 Cost and Consequences of Cyberattacks To date, cybercriminal activity is one of the leading problems facing humanity in this digital era. Cyber threats are the recklessly increasing crime on a global scale, and they are growing in dimension, complexity, and cost. A cyber threat could undermine the thrift of a city, state, or whole country. Furthermore, trade and commercial transactions are more dependent on technology now than ever before and making them a primary target for cyber threats. Threats can have a distressing effect on trades, costing them millions of dollars in damages. Regardless of the diligent efforts of security professionals through protective systems, attackers have always established ways to take off targeted resources from sensitive and most reliable sources globally through sophisticated, versatile, and automatic threats. Accordingly, this leads to a remarkable disaster for businesses, organizations, governments, and even people (Sarker et al. 2020). According to an annual report released by Cybersecurity Ventures in 2019, it is predicted that cybercrime will cost the world over $6 trillion annually by 2021, up from $3 trillion in 2015. It is also expected that global cybercrime costs to increase by 15% annually over the next three years, reaching $10.5 trillion by 2025 (Cybersecurity Ventures, 2019). This indicates the maximum transfer of financial capital in history, jeopardizes the incentives for improvement and investment, is exponentially larger than the impairment imposed from natural calamities in a year, and will be more lucrative than the international trade of all the most important prohibited drugs combined. Cybercrime costs comprise damage and devastation of data, theft of money, stolen intellectual property, stolen individual and economic statistics, lost productivity in industries, embezzlement, fraud, post-attack distraction to the normal course of business, forensic study, restoration and removal of hacked systems and data, and reputational harm.
9 A salient data breach (the second largest ever) hampered by Marriott and divulged near the end of 2018, is projected to have disclosed 500 million customer details. The Yahoo hack (the biggest ever) was estimated to have exposed 3 billion customer details, and the Equifax breach in 2017 (with 145.5 million users underwent) surpassed the biggest openly exposed attacks ever informed up until that time (Trautman & Ormerod 2017). These main attacks in conjunction with the NotPetya and WannaCry threats, which happened in 2017, are not only bigger scale and more sophisticated than earlier hacks, but also they are a mark of the times. 1.1.3 Synopsis of the Intrusion Detection System Owing to the mammoth growth of network events involving valuable data increases several organizations struggle with various threats and intrusive activities. Hence, the question of how to defend the network system from cyberattacks, distraction, and other anomalous actions from hacker become vital and urgent to address. Furthermore, the existing traditional tools including firewalls, intrusion detection and prevention models, secure network protocol, access control list, and encryption techniques cannot continuously safeguard the system effectively (Shah et al. 2017). In the wake of growing difficulties in network security, more modern and potential IDS is mandatory to secure computer network systems. The National Institute of Standards and Technology (NIST) defines cyberattack identification as "
the process of observing the events befalling in a computer system or network
and analyzing them for patterns of

attacks, defined as activities to compromise the CIA or to evade the security models of an information system or network" (Bace & Mell 2001). In general, an IDS is characterized as a network security system that continually inspects the inbound traffic and classifies them into genuine or suspicious activities to recognize illicit efforts to the network resources
10 effectively (Jazi et al. 2017). It includes monitoring and analysis of the activities of the customer as well as the system, examining susceptibilities and different system structures, assessing the integrity of data logs and critical systems, statistical study of user behaviors, the study of anomalous events, and audit of the operating system. Generally, IDS creates a log file about the identified threat for imminent analysis or to integrate these records with other data to make decisions and policies. Most of the IDSs sense malevolent acts and then sends an appropriate notification to the system administrator. Initially, IDS was developed as a single autonomous system to identify and classify cyberattacks by processing audit logs. In the present day, IDS is developed as a decentralized system that comprises multiple integrated components (Kumar et al. 2022). Even though these models are very different in the techniques they used to collect and study data, most of them rely on a comparatively common architecture. Figure 1.2 illustrates the generalized architecture of IDS widely used in practice, which includes three modules such as data gathering system, analyzer, and monitor.
Figure 1.2 General architecture of IDS (Kumar et al. 2022)
11 The data gathering system in the IDS model consists of an event generator or sensor module to collect data packets for analysis. It accepts data packets from the large external network and provides them to the analyzer in the form of activities. Indeed, the activity generators are basic filters that carry out audit trails, monitor a network, and generate germane actions for the packets in the network, or application program in a dataset which creates activities describing database transactions (Kumar et al. 2022). The raw data gathered by the sensing elements are given to the analyzers to categorize either as a threat or normal activity. The analyzer comprises two elements including a knowledge base and a detection engine. A detection engine is used to filter data and drops unrelated data observed by sensors. The knowledge base contains the information including profiles of normal activity, threat patterns, and significant features (i.e., thresholds) used for attack detection and classification. The initial study finds out the criticality of the attack and further analysis defines the scope, purpose, or occurrence of the cyberattack. A considerable volume of previous data logs may necessitate precise calculation. A good analyzer can relate two threats or regulate the inapt correlation between attacks (Kumar et al. 2022). The monitor is an interface to direct an alert to the network administrator about the attack identified by IDS. Ahmad et al. (2022) identified three desirable properties of IDSs including classification performance, time performance, and fault tolerance. In order to evaluate the effectiveness of the IDS, simple performance metrics like classification accuracy are not enough to assess the system. For instance, the network intrusions usually signify a very small ratio (e.g. 1%) of the whole traffic, and a trivial IDS that tags all the packets as genuine can realize 99% accuracy. To achieve decent classification performance, an IDS requires
12 to meet two conditions (i) it must be capable of properly detecting intrusions; and (ii) it must not classify genuine acts in a system environment as an abnormal activity. Typical estimates for assessing the classification performance of IDSs include detection rate, FAR, precision, recall, and F1- measure. These measures are discussed in Chapter 6.5 elaborately. The time performance is defined as the total time that the IDS requires to identify an intrusion. This time comprises the computational time and the transmission time. The computational time depends upon the processing speed of the IDS, which is the speed of the IDS to analyze network activities. If this speed is not high enough, then the real-time processing of the security system may be impossible. The transmission time is the time required for processed data to transmit to the system administrator. Both times need to be as low as possible to enable the administrator to respond to a threat in time before much impairment has occurred and to thwart a hacker from changing audit information or hacking the IDS itself. An IDS must be robust, reliable, and resilient to cyberattacks, and should be able to recover rapidly from effective threats and continue providing a secure service. 1.1.4 Challenges in Existing Intrusion Detection Systems With the increasing rate of cyber threats, now there is a substantial demand for intelligent IDS to sense both well-known and anonymous threats effectively. The proliferation of malware (malicious software) imposes a critical challenge on the development of IDS models (Khraisat et al. 2019). Malevolent threats have become more complex and the primary difficulty is to detect an anonymous and complicated threat, as the malware creators exploit diverse circumventing methods for hiding information to avert recognition by an IDS. Besides, there is an escalation in cyberattacks like zero-day attacks intended to target internet users (Singh et al. 2019). Failure

13 to detect the intrusions could reduce the reliability of security services like loss of CIA of sensitive user information and services. The concept drift (Iwashita 2019), higher dimensional dataset (Ghaddar & Naoum-Sawaya 2018), computational overhead (Al-yaseen et al. 2017), and network data imbalance (Priya & Uthra 2021) are some of the vital issues that need to be addressed in the domain of intrusion detection. Drift is triggered by the unremitting appraisal of the numerical features of the data engendered from network traffic while executing an attack, which is controlled by appraising the classification model. The higher dimensional datasets, in which the number of features is higher than the number of instances, are pervasive and are analyzed frequently by big data scientists both in industry and in academia (Jia et al. 2022). This issue becomes more perplexing when the data are streamed owing to the lack of a storage system for keeping the data to conduct future analysis. The processing overhead imposed by the large computations for achieving learning updates of the classification algorithm is an important issue in the development of the IDS model. The imbalanced distribution of datasets can make the classifier tend to misclassify labels with fewer instances into labels with more instances (Zhai et al. 2014). Most ML algorithms are sensitive to imbalanced data, which can cause lower ADR of rare threats. Currently, IDSs experience some more challenges in increasing the accuracy of minority class recognition, decreasing FAR, and identifying anonymous threats. IDS enables a salubrious setting for businesses and detects malicious incidents. Contrasting firewalls, which are employed at the boundary of the network and act as the gatekeeper by observing received data packets and deciding whether they can be permitted into the network or endpoint at all, IDSs observe inbound data packets and label mistrustful and
14 malevolent events. Therefore, an IDS can categorize not only threats that pass the firewall but also threats that instigate from within the network. Albeit, IDSs are considered a vital element of network security systems, they have some drawbacks that should be considered before installing them for security applications (Balaganesh et al. 2018). Indeed, IDSs using ML algorithms are widely employed to sense the malevolent attacks of a network or a node in various decentralized systems and deliver swift detection methods to stop further infections and spread. Although ML-based IDSs work better in recognizing new attacks, they are often affected by a higher FAR (Papamartzivanos et al. 2019). Some of these restrictions include (i) most IDSs provide a higher FAR, which wastes the time of security professionals and in some cases leads to a destructive automatic alert system; (ii) although most IDSs are advertised as real-time systems, it may consume significant time to automatically notify a threat; (iii) IDSs' automatic alert system is occasionally incompetent against cutting-edge threats; (iv) in several IDSs there are no manageable interfaces that enable customers to run them; (v) to exploit the maximum reimbursements from the established IDS, an expert administrator needs to monitor IDS functionalities and retort as needed; and (vi) several IDSs are not dependable, as they may not be well protected from threats or devastation. 1.1.5 Developing IDS using Ensemble Learning Machine learning is a flourishing artificial intelligence method that can create
an
automatic learning model to learn salient information from massive datasets and help mitigate unfortunate cyberattacks. Ensemble learners exploit a group of ML-based classifiers to achieve greater classification accuracy than could be acquired from an individual classifier. The key concept of Ensemble Classifier (EC) is to integrate many ML algorithms to take advantage of every deployed base classifier for developing
15 a more robust classification model (Zhong et al. 2020). ECs are mostly useful if the problem can be divided into sub-tasks so that every sub-task can be allocated to one base learner. Based on the configuration of the EC, every unit contains one or more ML algorithms. In the field of cyberattacks, as the signs of various threats are relatively dissimilar from one another, it is common to have diverse attributes and various ML algorithms to classify various kinds of threats. Hence, it is apparent that an individual IDS cannot handle all kinds of input data or detect various cyberattacks (Abdullah et al. 2018). Several research works have proved that classification problems can be resolved with higher performance while implementing ECs rather than individual classifiers (Sakhnini et al. 2021). Mostly, ensemble learners are employed for decreasing the FAR and enhancing accuracy related to the enactment of an individual classifier. The key challenge in using ECs is how to select the optimal set of classifiers to build the ensemble method and which function is used to integrate the decisions of those classifiers. 1.1.6 Feature Selection in IDS Feature (attribute) selection is a fundamental ML concept that exploits particular selection metrics to automatically or manually select more relevant subsets among the huge redundant and unrelated attributes of an originally identified database. It must exhibit less processing complexity and high training accuracy without adversely impacting the prediction performance (Cai et al. 2018).
The key objective of this phase is to minimize the number of selected attributes into an abridged set and to select the most significant features indispensable to create an alarm once an intrusion is identified. Besides, it also involves removing redundant as well as unrelated features that do not act any significant role in detecting intrusion; therefore, the total temporal overhead of the algorithm is reduced with significant performance improvement in terms of prediction accuracy and generalization.
16 The generalization is an important property of classification as it supports the algorithm to evade overfitting on a particular database. Furthermore, it also enables
a

well-balanced and low-dimensional input dataset, thus significantly reducing the overfitting problem, decreasing the time of model building, increasing the training accuracy, and improving the direct implications of the learning outcomes (Song et al. 2019). The feature selection algorithms can be pigeonholed into supervised, semi-supervised, and unsupervised attribute selection methods. The widely employed attribute selection approaches are wrapper, filter, and embedded. The key idea of supervised attribute selection is exploiting the relationship and
significance between attributes and labels to choose significant
attribute subsections. Whereas, unsupervised attribute selection
is enabled by the important notion of selecting an important attribute subsection without a target variable but instead exploits assessment measures and clustering to increase the accuracy (Cai et al. 2018). Furthermore, the four rudimentary phases of a
normal attribute selection technique are subsection creation, subsection assessment,
terminating condition, and authentication technique. Initially, an identified search strategy is used to choose an appropriate attribute subset, evaluated based on particular performance metrics. Then, according to the termination criteria, the attribute subset that is completely related to the selected assessment benchmark dataset and is designated as the germane attributes, which can be recognized by validation
data points
or a domain expert (Dy et al. 2004). 1.2 RESEARCH MOTIVATION Cybercrime has become the largest menace to all types of businesses and charities. According to the report released by the UK government in 2021, 39% of industries and 26% of charities had cyberattacks or security breaches in the last 12 months. Similar to earlier years, this is
17 higher among high-income charities (51%), large-scale industries (64%), and medium-scale industries (65%) (Nguyen et al. 2022). Likewise, based on a report released by Crime Survey for England and Wales (CSEW), there was a drop in identified deceit events from 3.6 million in 2020 to 3.2 million in 2021. The Office for National Statistics (ONS) survey also exposed that irrespective of the overall decline in deceit activities in 2021, attacks and security breaches against businesses and charities increased up to 64% in 2021. The unknown attacks are expected to intensify with the new trend of computing paradigm which involves cloud technology, big data analytics, and IoT networks. The amplified level of cyberattacks and other network security breaches as well as the financial risks related to such attacks point out the dire need for an efficient method for sensing network intrusions. Identifying invasions in communication and computing systems is one of the urgent demands owing to their higher level of exploitation and dissemination of a huge volume of valuable data over networks. In spite of enormous efforts by the network security experts and engineers, IDS still experiences challenges in increasing classification accuracy while decreasing FAR and in identifying novel attacks due to concept drift, higher dimensional dataset, computational overhead, and network data imbalance (Iwashita 2019; Priya & Uthra 2021; Ghaddar & Naoum-Sawaya 2018; Al-yaseen et al. 2017). The motivation of this study is that most of the current IDSs do not tackle the above problems in their working procedure. Therefore, there is an urgent demand to develop an effective IDS with techniques to manage concept drift and data imbalance while reaching superior accuracy and real- time recognition of cyberattacks. Motivated by this, this work attempts to develop a new ensemble classifier that is more accurate, reliable, and robust than others existing IDS models.
18 1.3 PROBLEM STATEMENT Of late, numerous intrusion detection approaches have been developed in various fields, but conventional methods cannot be directly used to certify network security owing to their growing intricacy and more byzantine threats. These approaches can be confronted by the increasing amount of data and require domain-specific knowledge, which needs advanced solutions assimilating cutting-edge ML algorithms along with numerous data sources such as IoT sensors, topology, and network information. Nowadays high-dimensional big data transfers in/out of business organizations in huge volume and very high speed. Examining all the inbound data packets has become a very expensive and challenging problem regarding predictive accuracy, prediction time, and system requirements. Innumerable efforts have been devoted to increasing the classification performance of IDSs by training the models on a subset of attributes from real-world datasets. But, most of the approaches have been trained using the well-known KDDCup 99 benchmark database which is nearly 22 years old and cannot mimic modern network traffic perfectly (Kulariya et al. 2016; Jianjian et al. 2018). This has made most of these approaches superseded for using real-time applications. At the same time, studies using newer databases have not balanced prediction accuracy and recognition time. The review of the state-of-the-art in IDSs (discussed in Chapter 2 of this dissertation) has posed the following problems to be considered for further research. Problem 1: There is an urgent need for efficient IDS, which can process large real-time datasets and can identify cyberattacks with higher accuracy, and lower FAR with lesser overhead.

19 Problem 2: The infrequent classes are not correctly categorized due to the considerable imbalance among the class labels in the multi-class database. Hence, there is a demand for effective IDS, which can further able to detect and classify the types of intrusion with higher classification accuracy and lower FAR even in the case of the class imbalanced dataset. Problem 3: Though numerous security professionals have developed various IDS models they gave less emphasis on optimal feature selection methods for large datasets to achieve better classification performance. Hence, there is an urgent need for designing an efficient IDS to handle large datasets. 1.4 SCOPE AND OBJECTIVE OF THE RESEARCH The Internet has become one of the generally utilized resources. The massive proliferation of internet usage helps people perform digitized transactions. The high adoption rate of ubiquitous computing and internet technologies enable colossal volumes of data to be engendered online (Kumar et al. 2019). Nowadays, these technologies are integral elements of human life, such that it is difficult to catch an individual without an online presence. The growth of online presence has also caused the exchange or upkeep the valuable private data online. Though the internet and related technologies fetch lots of convenience regarding access, they are also susceptible to cyberattacks or threats. Such penetration might cause enormous losses with respect to the money and exposure of personal data to unauthorized users. At present, the utilization of mobile devices enables more customers entering into the digital world. The explosion of the number of web pages that are developed for mobile-based expedients is also amplified. The massive utilization of portable computing maneuvers has resulted in many customers selecting mobile-based e-commerce applications. The high

20 adoption levels of mobile devices and the Internet and the upsurge in e- commerce dealings happening via networks have resulted in the growth of cyber threats. This reveals the growing need for an intelligent IDS to deliver enhanced security services to Internet users. It also mandates the utilization of IDSs with superior predictive performance and shows the lack of efficacy in the prevailing IDS models used in practices. The goal of this study is "to develop an effective, accurate, and smart ensemble classifier with appropriate optimal feature selection algorithm for IDS to ensure network security in real-time". To realize this goal, the following specific objectives have been set: 1. To create an intelligent ensemble classification approach that effectively handles the intrinsic data imbalance problem in the IDS model to classify the network traffic into normal and anomalous accurately. 2. To incorporate an optimal feature selection algorithm with the classifier to increase the classification accuracy, reduce FAR, and reduce the time consumption of IDS for training and testing the classifier. 3. To demonstrate the effectiveness of the proposed methods by comparing their performance with other state-of-the-art

approaches with respect to performance metrics. 1.5 RESEARCH METHODOLOGY To identify different cyber threats in the network, this work develops an intelligent classifier using the ensemble technique to increase the accuracy, and ADR and reduce the FAR in classifying the intrusive activities significantly. The proposed ICET includes two elements as a feature selection module and an ensemble classifier. To cope with high-dimensional datasets,

21 the proposed feature selection module exploits a CFS algorithm to select the appropriate features. The performance of the proposed feature selection approach is further enhanced by integrating CFS with BIO. This integration is embedded in the proposed ensemble classifier to increase the performance of the IDS. The proposed ensemble classifier module includes three different classifiers including BF, RF, and C4.5 decision tree based on a voting mechanism using the AoP rule. The established ICET aids to tackle imbalanced and multi-class datasets with higher accuracy. In this study, the ICET model is trained and assessed using two real-world datasets such as NSL-KDD and CIC-IDS 2017 using Weka 3.8.3 workbench. Extensive empirical results prove that the proposed

intrusion detection model significantly outperforms other state-of-the-art methods.

Figure 1.3 illustrates how this study was intended and distributed over the 4 phases. Literature survey: A critical review of previous germane works was performed to realize all the requirements for developing an effective IDS. Considering these demands is imperative to define the goals and the research problem. Through an inclusive review, the advantages and disadvantages of each existing IDSs are assessed to sense problems related to earlier works proposed by several researchers. System design: All the indispensable requirements to establish the ICET model are collected including the requirements for feature selection, hyperparameter optimization, and ensemble classifiers. The feature selection is performed to reduce the dimension of the feature space by selecting an optimal subset of features. The performance of the feature selection process is improved by applying an appropriate optimization algorithm. To classify the inbound data traffic, this study implements an intelligent ensemble classification approach using suitable base learners. In this phase, the

22 proposed BIOCFS and EC algorithms are integrated to find out the malicious network activities. Figure 1.3 The phases of the research development process Implementation: To assess the effectiveness of the intended BIOCFS algorithm and ICET approach, the experimental analyses are

carried out on a 3.6 GHz Intel Core i7-4790 CPU with 16GB RAM and

Windows 10 operating system using Weka 3.8.3 testbed.

23 Evaluation: The performance of the proposed intrusion detection model is assessed by relating the research findings with other prevailing IDSs from the literature in terms of performance metrics. 1.6 RESEARCH CONTRIBUTION The major contributions of this work are to develop an effective IDS model for detecting and classifying cyber attacks in network traffic accurately. Several transformative new technologies including preprocessing, feature selection, ensemble learning, and classification methods that are more appropriate for this domain are identified through an extensive literature survey. The classification accuracy is further increased by developing and implementing a novel bio-inspired optimization algorithm with a correlation- based feature selection process. The performance of our proposed methods is meticulously assessed on the real-time datasets. The efficacy of the proposed methods is verified by relating its performance with other state-of-the-art methods regarding evaluation metrics. The contributions of this article can be summarized as follows: 1. To understand cyberattacks against networks, this study explores various cyberattacks and prevailing ML-based IDS models which enable us to train an efficient and accurate detection model. 2. This work introduces an ICET model that takes advantage of CFS, and BIO algorithms to reduce the size of the feature space. This model evaluates the correlation between pairs of attributes and selects more significant attributes. Then, the selected subset that includes a reduced dimension is applied for classification.

24 3. An ensemble classifier is developed

by coalescing decisions from different classifiers including BF, RF, and C4.5 into one to increase the classification performance. Furthermore, this study uses a voting mechanism using the AoP rule to handle the multi-class problem. 4. The proposed ICET model is implemented and the results are compared with other

relevant individuals as well as ensemble classifiers approaches on a workbench containing datasets, viz., NSL-KDD and CIC-IDS 2017. 1.7 THESIS ORGANIZATION The thesis is organized into 7 chapters

with

various phases of the research development process. These are summarized as follows: Chapter 1: This chapter provides a brief description of the background to classifying cyber threats in the communication network. It defines the research motivation, problem statements, scope, and goal of the study. The research methodology, contribution of the work, and a brief description of the remaining chapters of the thesis are also given in this chapter. Chapter 2: This chapter gives a comprehensive analysis of previous works related to the feature selection process and ensemble classifiers used in IDS models. Also, this chapter provides some background

information and investigations into current IDS approaches to identify different cyber threats.

Chapter 3: This chapter provides the overall architecture of the proposed ICET framework. The details of all the elements of the proposed structural design such as the data preprocessing techniques, attribute selection module, ensemble classifier, and attack detection unit are discussed.

25 Chapter 4: This chapter provides the detailed design of the proposed BIOCFS attribute selection algorithm in order to increase the classification performance of the IDS model. Chapter 5: This chapter discusses the proposed ensemble classifier to improve detection performance by solving the problem related to data imbalance. The selection of base classifiers used in this ensemble approach is also elaborated in this chapter. Chapter 6: This chapter presents the implementation details of the intended ICET approach. The performance evaluation of the proposed feature selection and intrusion

recognition algorithms is performed using the Weka simulator. Chapter 7: This chapter reviews the results obtained from the proposed intrusion detection model.

It concludes the research by discussing the advantages, disputes, and limitations related to the proposed work followed by recommendations for future directions.

26 CHAPTER 2 BACKGROUND AND LITERATURE SURVEY This chapter reviews

the

approaches used in conventional intrusion detection models. The key problem in the domain of IDS research is correct methods and algorithms should be used to identify malevolent activities

as well as attackers. Hence, this study considers many cyberattack detection

methods to present an inclusive summary and the theoretical background of earlier studies.

Due to

the proliferation of effective networking environments and technologies, detection of malevolent threats becomes a smart and automated technology to realize cooperative data procurement, distributed data handling, and computation. The state-of-the-art in IDS architecture to explain their solicitation in this study and their key restrictions are delineated in this chapter. Besides, it explores the nature, scale, and consequence of cyber threats and other security breaches on network infrastructure over the past year. This chapter presents some of the latest contributions in the field of cyber threat detection and classification. 2.1 CYBERATTACKS ON NETWORK INFRASTRUCTURE A cyber threat is defined as any incident with the capability to attack a blow to tasks, missions, images, national cyber assets, or personnel, through illegal access, devastation, modification of data, and/or hindrance of (disruptive) service delivery. It is more pervasive as attackers exploit system weaknesses for stealing sensitive information, gaining monetary benefits gain, or even devastating the whole network architecture. Security experts use

27 diverse detection and prevention approaches to decrease the risk of cyberattacks. Recently, the Federal Bureau of Investigation made a high- impact cybersecurity alert about the growing number of cyber threats on public organizations. Government representatives have advised major cities that such attacks are an alarming trend that is likely to remain. The window for sensing some cyberattacks can be calculated in days, as cybercriminals are aware of prevailing security systems and are constantly increasing their intrusive activities. In many cases, a security breach is expected, which help network administrators develop the best plan for timely identification and mitigation of cyberattack. This proliferation of the internet and related technologies increases the intricacy of information systems regarding understanding the information assurance principle (i.e., CIA) globally. Since hackers are exploiting the advanced tools to create a large amount of well-refined cyberattacks on the various network elements they are challenging to identify (Janarthanan & Zargari 2017). Additionally, some of the attackers are very skilled professionals with extremely powerful tools and cutting-edge technologies that can easily infiltrate any information system or network if not secured sufficiently. This makes various information systems and networks susceptible to cyber threats (Wang et al. 2018). To alleviate the security breaches of network systems, an IDS is frequently installed at an Internet gateway to protect the network infrastructures. It constantly observes all data packets and identifies possible signatures of malevolent behaviors. On the other hand, the IDS analyzes only incoming and outgoing data flow, but not inside a data flow. To handle this issue, a distributed implementation approach, where the IDSs are assimilated into gateways and routers within the networks, is required (Gajewski et al. 2019).

28 Motivated by the success of ML algorithms in numerous domains (e.g., automation, robotics, computer vision, etc.), several commercial and scientific societies focus on utilizing machine learning algorithms to increase the performance of IDS (Liu & Lang 2019). The past decade has perceived the traditional IDS approaches improved with several powerful machine learning security models. Besides, with suitable learning data samples, ML- based IDSs can obtain attractive detection outcomes and good generalization (Liu & Lang 2019). Regardless of the reasonable enactment of the ML-based IDSs, achieving effectiveness and trustworthiness is a key issue owing to high-dimensional, redundant, unrelated attributes; incompetent recognition of all kinds of new threats by a single

machine learning-based ML classifier; expensive and incorrectly labeled learning data samples

with considerable FAR and more training and testing time. 2.2 NEED FOR MACHINE LEARNING ALGORITHMS AGAINST CYBERATTACKS Machine learning is the domain of information systems that enhance through learning (training) and experience, allowing a computer to make correct predictions when fed data without being explicitly programmed. These techniques exploit a subset of a huge database, called learning data or sample data, to formulate a mathematical model to make decisions or predictions based on a given problem. ML algorithms are used in various fields such as computer vision, search engines, spam filtering, voice recognition, recommendation systems, etc. Likewise, in the domain of cybersecurity, different ML algorithms are employed to observe and examine network traffic to identify different anomalies. Generally, these algorithms are trained on a set of normal network traffic traces that are gathered over a long period. Most of these algorithms detect abnormalities by measuring deviations from a regular traffic model.

29 There are several motives for investigators to apply various ML algorithms to identify and classify threats in networks. One such reason is the capability of machine learning to determine the correlation among a dataset with a huge volume of samples. The main assumption of ML-based intrusion detection is that an intrusive activity makes discernible signatures within the network traffic and these signatures can be effectively sensed through this algorithm (Fadlullah et al. 2017). These algorithms provide automatic recognition that deduces statistics about the malevolent threat from the huge volume of existing traffic traces. Moreover,

an

ML algorithm can be employed to recognize intrusion in traffic without demanding any prior knowledge about those data. By integrating the features of ML algorithms and powerful processing units, the researchers can develop a more influential tool for retorting against cyberattacks. Furthermore, currently, we have a great deal of data, but human proficiency to analyze those data is limited and expensive. Hence, by applying machine learning

algorithms, we can automatically detect patterns that individuals may not be able to identify due to the huge volume of heterogeneous datasets. Without applying ML algorithms, network administrators would need to form rules manually, which would not be scalable. 2.3 TAXONOMY OF INTRUSION DETECTION SYSTEM IDSs can be pigeonholed using five parameters including intruder type, detection methodology, deployment method, detection behavior, and processing methods of gathered data. Figure 2.1 shows this classification in detail. Based on intruder type the IDSs are divided into two types including internal intrusion and external intrusion. Internal invaders are customers with authorized access or privileges to a system with either an account on a server or physical access to the network (Li & Liu 2021). External invaders are

30 individuals who do not belong to the network domain. All attackers, whether internal or external, can be organized in different ways and contain individual hackers to spy agencies working for a government. The effect of an intrusion hinges on the targets to be accomplished. An individual hacker could have small objectives while spy agencies could have larger motives (Li & Liu 2021). Figure 2.1 Classification of IDS models Based on the methodology used for detection, these models are classified into two types such as signature (misuse-based) IDSs and anomaly- based IDSs (Otoum & Nayak 2021; Bhati et al. 2020). In

the

misuse detection approach, threat detection is based on well-defined signatures or traffic patterns. The Signature-based IDS (SIDS) can capture signatures of known attacks. For every threat, its signature is to be generated. The patterns define

a

suspicious, crew of sequences of actions that can be feasibly detrimental and stored in a database. If it is matched with the signature stored, then an alert will be created. The procedure for SIDS is illustrated in Figure 2.2.

31 Based on the techniques used for threat identification, SIDSs are divided into simple rule-based, string matching-based, state modeling-based, and expert system-based IDS (Kreibich & Crowcroft 2004). However, this type of intrusion detection cannot recognize anonymous (never-seen-before) attacks (Kaur & Singh 2019). For example, polymorphic variants of the malware and the increasing amount of new threats can further undermine the suitability of this conventional model (Khraisat & Alazab 2021). A potential solution to this issue would be to use anomaly-based IDSs, which operate by profiling what is a normal activity instead of

an abnormal one. Figure 2.2 Intrusion detection using the

SIDS approach Anomaly-based Intrusion Detection System (AIDS) has gained more attention from a lot of researchers due to its ability to mitigate the restriction of SIDS. In anomaly-based detection, the standard behavior of a legitimate user is modelled using knowledge-based, statistical-based, or machine learning, approaches. Any considerable difference between the perceived activity and the model is viewed as a threat, which can be interpreted as an invasion (Otoum & Nayak 2021). This type of model operates on the fact that malevolent behavior is diverse from the behavior of the legitimate user. There are two stages in the development of AIDS: the training process and the testing process. In the training process, the regular traffic profile is employed to train the model of usual activities. In the testing

32 process, a new sample is employed to develop the system's capacity to generalize to hitherto unobserved attacks (Bhati et al. 2020). In this approach, normal system behavior is modelled using knowledge-based, statistical-based, or ML-based methods (Otoum & Nayak 2021). The major benefit of AIDS is the capacity to detect zero-day attacks since identifying the anomalous user behavior does not depend on a signature database (Aljawarneh et al. 2018). AIDS generates an alert when the observed activity diverges from the regular activity. For example, a profile for a system may designate that 15% of average network bandwidth is utilized by the internet border for the Web activity during a normal day. The AIDS then matches the features of the incident to predefined values in the profile. If Web activity utilizes bandwidth higher than 15%, then AIDS treats it as an attack and creates an alarm. Profiles can be created for several actions, for instance, the number of emails sent by a user, the number of failed login attempts for a node, and processor utilization for a host in a specified period. Profiles are created by observing the system and network for a certain period (usually days, occasionally weeks). AIDS can create either a dynamic or static profile. A dynamic profile updates frequently when new incidents occur whereas a static profile cannot be altered. The process of the anomaly-based approach is shown in Figure 2.3. The anomaly-based approaches have several advantages. It can identify internal attacks. If an attacker starts making transactions in a stolen account that are different from normal behavior, it creates an alert. It is very hard for an attacker to recognize what is a standard user activity without generating an alarm as the system is developed from tailored profiles.

33 Figure 2.3 Intrusion detection using the AIDS approach Based on the type of deployment, IDS models can be divided into two types: host-based IDS (HIDS), network-based IDS (NIDS), and hybrid IDS (Kumar et al. 2021). The host-based approaches monitor the behavior of a specific host and the incidents occurring on that node for malevolent activities (Martins et al. 2022). It is usually installed in the critical nodes to provide security by observing the system objects and their features. HIDS observes processes, system logs, file access, network traffic, etc. It reports threats by sending e-mails or writing logs. It utilizes a database to hoard features and entities. Host-Based IDS can be categorized into 4 types including kernel-based IDSs to detect kernel-level suspicious activity, log file analyzers to search patterns representing malevolent actions, connection analyzers to check connection attempts from and to a node, and file system monitors to verify the reliability of files and directories (Martins et al. 2022). Generally, NIDS comprises a network device with a Network Interface Card (NIC) that functions in promiscuous mode. The IDS is employed on the border or along a part of a network to observe most of the traffic on that network part. NIDS can be inactively used, without any variations to systems or networks. It is very efficient for observing both inbound and outbound traffic (Kumar et al. 2021). A single NIDS can defend the whole network and switching off a target system will not distress the
34 NIDS. However, it cannot manage the network if the system bandwidth is encumbered. Hybrid IDS enhances
the
host-based detection system by making it capable to observe the network traffic (flowing either into or out of the specific host). Hybrid managers integrate the operations of different host- based detection systems with network-based sensor technology that is just to examine the data packets flowing through the specific node where the hybrid agent is deployed. The processor utilization in a hybrid agent is higher than that of a HIDS agent. Based on the method of processing gathered data, IDS are classified into three types such as centralized, distributed, and hierarchical (Soliman et al. 2012). In centralized IDS, the data analysis is carried out in a fixed number of locations that do not hinge on the number of nodes being observed. The distributed IDS analyzes the data at several hosts (locations) that are being supervised. Hierarchical approaches are developed for multi- layer (clustering) network architectures. Cluster heads are liable for supervising their associate hosts, and partaking in the global intrusion detection verdicts. Based on detection behavior, IDS can be categorized as attempted break-ins (sensed by general activity profiles), Masquerade threats (sensed by security constraint defilements), penetration (sensed by looking for specific behavioral patterns), leakage (sensed by consumption of system resources), malicious use (sensed by activity profiles, application of distinct rights, by security condition defilements) (Khraisat et al. 2019). 2.4 CURRENT RESEARCH STATUS OF IDS The proliferation of Internet connections and penetration of a wide range of smart computing and communicating devices continues to grow extremely and imposes new security and privacy challenges to the network. Several cyberattacks have developed vividly with
the instigation of the internet applications and the proliferation of radical information
35 technologies. In spite of the assiduous attempts of network professionals regarding security measures, attackers always attempt to take off resources from valuable and most trusted sources globally through adaptable, complicated, and automatic threats. This leads to remarkable devastation in productions, trades, governments,
and even individuals (Sarker et al. 2020). For instance, Damaševiˇcius et al. (2021), interestingly provide a summary of different threats and their significance. Firstly, the authors discuss the prediction of $6 trillion of cyberattacks by 2021 and the different worldwide leading-edge attacks that could cause the loss of $1 billion.
As well, the total cost for cyber threats committed globally will reach $1.5 trillion by compromising 2
to 5 million computing devices every day. Hence, Organizations are increasing their investment in research to increase the enactment of the IDS model in terms of attack detection of these threats. Consequently, the last few years have perceived the growing popularity of IDSs due to their intrinsic capability to sense an attack in real-time (Meryem & Ouahid 2020). 2.4.1 Feature Selection-based IDS Models The aggressive growth
of the advanced Internet technologies and related applications has led to the influx of high dimensional and unstable big
data at a high speed, significantly challenging
conventional ML algorithms.
However,
the feature selection process (FSP) over the last decade has been theoretically and practically proved to be more efficient in handling
huge datasets in
different research fields, specifically in intrusion detection (Zhou et al. 2020; Mahfouz et al. 2020; Aburomman & Reaz 2017;), thus leading to boosted enactment of classifiers in IDSs (Zhou 2009). FSP is a fundamental ML notion that utilizes explicit selection criteria to automatically or manually select optimal subsections among the huge redundant and unrelated attributes of an initial database with less processing complexity and high training
36 performance without adversely impacting the prediction accuracy (Nguyen et al. 2022). FSP acts a substantial role in handling preprocessing challenges

of huge datasets, model building, learning, assessments, and interpretation. For instance, decreasing the system intricacy leads to minimized processing complexity. Similarly, reducing the adverse effect of the curse of dimensionality increases the performance of the classifier. In addition, it also provides a well-balanced and low-dimensional dataset input, thereby considerably solving overfitting problems, decreasing the time of model building, increasing the learning accuracy, and improving the direct implications of the training outcomes. The FSP algorithms can be pigeonholed into unsupervised, semi- supervised, and supervised methods (Song et al. 2007). The key idea of supervised FSP is exploiting the relationship and significance among attributes and class labels to choose a significant subsection of attributes. Conversely, unsupervised FSP is enabled by the key concept of selecting a substantial attribute subsection without a target variable but instead exploits assessment function and clustering to increase the classification accuracy (Cai et al. 2018). Furthermore, the four rudimentary phases of a standard FSP are subsection creation, subsection assessment, ending condition, and endorsement process. Initially, a particular search approach will be used to choose a potential subset, assessed using a specific performance metric. Then, according to the stopping criteria, the attribute subset that is completely related to the assessment standard is considered the germane attribute, which can be authorized by a domain expert or validation data. As mentioned earlier, the widely employed FSPs are wrapper, filter, and embedded. The filter approach is an effective, fast, and scalable approach for differentiating the learning bias from the attribute selection 37 process, facilitating the dependable selection of significant attributes based on the vital properties of a given dataset using a measured score. The measured score aids to select attributes with a high score and removes those with a low score (Ambusaidi et al. 2016). The conventional filter algorithms exploit a specific criterion to rank attributes and train the classifier with the highest- ranking attribute. The information gain-based methods, RelieF of algorithms (Robnik & Konenko 2003), and Fisher score (Peng et al. 2005) are leading performance criteria widely used in practice. The wrapper approaches exploit the classification accuracy, which is primarily trained on a training dataset to choose the optimum attribute subsets by assessing the ideal enactment of different possible attribute subsections to eliminate the significant shortcomings of the filter approach, such as ignoring the impacts of the selected attribute subsection on the performance of a specific classifier (Khammassi & Krichen 2017). However, the choice of an ideal attribute subset based on predictive accuracy comes with some advantages and shortcomings. For example, the guarantee that designated attributes from the wrapper methods is dependable with better predictive performance. Nonetheless, low classification accuracy cannot be assured the same. Besides, the correct classifications of wrapper approaches come with the demand for more processing power. Also, a typical wrapper approach achieves the following simple steps: (i) select the subset of attributes randomly; (ii) evaluate the designated subsets of attributes using the enactment of the classifier; and (iii) repeat the above two phases until a selected optimum attribute subsection is gained. Genetic algorithm (GA) and recursive attribute removal are some of the renowned wrapper methods (Bai et al. 2014). The embedded approach is a precise and effective method developed to alleviate the difficulties of the wrapper and filter approaches. It 38 integrates the potentials of both methods stated earlier, including picking an optimum subsection with the maximum predictive performance and reducing the processing overhead. Conversely, it demands a parameter that describes the cut-off value for the calculated scores of the attributes. Moreover, this method selects optimum attributes and creates model fitness simultaneously. The attributes with the high predictive score are employed, and those with a low score are eliminated similar to the wrapper approach (Li & Chen 2020). But, it has a lower processing overhead since the model fitting is performed at once to calculate these values rather than the iterative nature of the wrapper model (Li & Chen 2020). The embedded methods are employed with combined attribute selection including ID3, C4.5, and objective function- based regularization approaches that imposed the factor to be a small score or closely zero while reducing the fitness error. To realize a more dependable and effective classification, Hota & Shrivas (2014) proposed an attribute selection approach to remove the irrelevant attributes from the dataset. The author demonstrated that the intended approach using C4.5 with a score known as information gain (

IG) realizes superior accuracy with 17 selected attributes from NSL-KDD datasets. They employed selected attributes to construct several DT
models, out of which C4.5 integrated with symmetrical uncertainty and IG, registers the optimum accuracy of 99.68% and 99.64% with only 17 and 11
designated attributes, correspondingly. Malik et al. (2015) proposed an attribute selection method using an RF classifier with
Particle Swarm Optimization (PSO). In this
integrated approach,
more suitable attributes for each class are designated to achieve a low
FAR with greater predictive accuracy
as compared with
other methods. Dhanabal & Shantharajah (2017) studied the performance of a correlation- based feature selection algorithm on the NSL-KDD dataset using ML
39 classifiers including J48, SVM, and Naïve Bayes (NB) classifiers in terms of evaluation measures. The authors demonstrated that J48 outperformed SVM and NB in terms of accuracy. Akashdeep et al. (2018) proposed an attribute ranking method using IG and correlation. In this model, dimensionality reduction of the attribute is achieved by integrating ranks gained from both IG and correlation using an approach to classify useless and useful attributes. These selected attributes are then given to a feed-forward neural network for training and testing on the KDD99 dataset. Bolón-Canedo et al. (2018) proposed correlation and symmetrical uncertainty to decrease attributes. The authors achieve 80% of feature reduction. Le et al. (2019) proposed an attribute selection method using the DT model with a sequence forward selection algorithm to select the optimum attribute subset. Farahani (2020) proposed a new cross-correlation-based attribute selection technique for the ideal feature subset. The effectiveness of the FSP is demonstrated using four classifiers such as SVM, NB, DT, and K- nearest neighbor (KNN). The empirical analysis
of
the various datasets proves the superiority of this method in terms of accuracy, precision, recall, and F1- measure related to the other two approaches in different classifiers. Pandey (2019) developed a new attribute selection method to achieve high accuracy and low FAR. Firstly, data are filtered by the VM; hence, the IG will get associated with a base learner to select the required attributes. Then, various classification algorithms are employed. Ren et al. (2020) developed an efficient IDS using hybrid data optimizer which contains data splitting and an attribute selection algorithm. In data splitting, the
proposed algorithm
is employed to remove outliers, GA to fine-tune the splitting percentage, and the RF classification algorithm to achieve the optimum learning process. In attribute selection, RF and GA are employed to get the optimum attribute subsection. Then, an IDS
model using
40 the RF algorithm
is constructed using the optimum learning dataset achieved by data splitting, and the attributes are designated by attribute selection. Jaw & Wang (2021) proposed an efficient hybrid attribute selection that effectively picks appropriate attributes and delivers reliable threat identification. The hybrid attribute selection assimilates the potentials of correlation-based attribute selection,
genetic exploration, and a rule-based engine to efficiently choose subsections of attributes with high correlation. Also, it significantly decreases the intricacy of the system and improves the generalization of training methods. Reviewing various approaches in the literature, as mentioned, can conclude that several approaches have been proposed to decrease the number of attributes to sense the threats rapidly with the lowest error in the IDS. In this work, a CFS algorithm is proposed to select the appropriate features. It calculates the correlation between the
features and selects the optimal subset for training and testing
phases. Besides, it exploits the optimized RelieF algorithm to calculate the quality of attributes. The attributes with a low-quality index are eliminated to reduce the dimensionality of the feature space. The performance of the proposed feature selection approach is further enhanced by integrating CFS with
the
BIO algorithm. This integration BIOCFS is embedded in an ensemble classifier to increase the performance of the IDS.
2.4.2 Ensemble Classifier in Attack Detection Ensemble learning exploits a crew of ML algorithms to provide higher classification performance than could be gained from an individual classifier (Ryu et al. 2010). The key concept of EC is to combine multiple machine learning algorithms to utilize the merits of each deployed classifier to realize a more robust and reliable classifier. EC is predominantly useful if the problem can be divided into sub-problems so that each one can be

41 allocated to one unit of the EC. Based on the configuration of the EC, each unit can contain one or more classifiers. In the world of cyberattacks, since the patterns of diverse threats are quite distinctive, it is obvious to have various sets of attributes and various ML classifiers to identify various threats. It is thus apparent that a single IDS cannot handle all types of input data or classify various threats (Bala Ganesh et al. 2018). Numerous researchers have demonstrated that classification problems can be resolved with better accuracy when applying ECs rather than single classifiers (Jaw & Wang 2021; Mahfouz et al. 2020; Abdullah et al. 2018). The results of multiple ML classification algorithms in an intrusion detection problem can be pooled to improve the performance of the IDS. The main difficulty in implementing ECs is how to select the optimal set of classifiers to establish the EC and which decision function to use to combine the results of those algorithms (Jaw & Wang 2021). Paulauskas & Auskalnis (2017) developed an EC to combine four different classifiers such as J48, C5.0, NB, and PART. This method is developed to combine many weaker classifiers to build a powerful learner. The empirical analysis proved that the EC realizes higher predictive accuracy. Additionally, few researchers employed different techniques to reduce the dimension of the databases. Khammassi & Krichen (2017) proposed a GA- based wrapper approach with a logistic regression-based learning method for IDS to choose the significant attributes. The experimental results prove that this method attains higher ADR with a subset of only18 attributes in the KDD99 database and 20 attributes in the UNSW-NB15 dataset. Belouch et al. (2017) proposed a two-phase classifier using the RepTree algorithm to identify cyber threats effectively. In the first stage, this model splits the inbound data packets into three kinds of protocols TCP, UDP, or other, then categorizes them into normal or cyberattack. In the second
42 phase, a multi-label algorithm categorizes the anomaly sensed in the first stage to detect the threat label for UNSW-NB15 and NSL-KDD datasets. The number of attributes is reduced from 40 to less than 20 attributes, based on the protocol, using attribute selection methods. Abdullah et al. (2018) proposed an enhanced IDS (EIDS) that divides the input dataset into various subsets based on each threat. Then this approach performed an FSP using an IG filter for each subset. Then the optimum attribute set is created by merging the list of attribute sets that are gained for each threat. The empirical fallouts that are carried out on the NSL- KDD dataset reveal that the established approach with a smaller number of attributes enhances the predictive accuracy while reducing the complexity. Furthermore, a comparative study is carried out to assess the effectiveness of the attribute selection algorithm using different classifiers. To increase the overall performance, another stage is executed using RF and Partial Decision List (PART) on VM. The results show that superior accuracy is realized when applying the product probability rule. Mahfouz et al. (2020) proposed an Ensemble ML Classifiers (ENML) with different learning models to address the issue of the accuracy and FAR in the intrusion detection process. This approach contains three ML algorithms from different classifier families. This model exploits C4.5, KNN, and Multiple Layer Perceptron (MLP) classifiers for attack detection. These classifiers work simultaneously, and each one forms a different model of the data. The outputs of the three classifiers are integrated through the majority voting method to acquire the final output of the EC. Umar et al. (2021) proposed a machine learning-based IDS (ML-IDS) approach with an attribute selection algorithm and intrusion detection method. The attribute selection algorithm is wrapper-based with a decision tree as the attribute evaluator. The projected attribute selection approach is employed in combination with some
43 selected ML algorithms to build IDS models using the UNSW-NB15 dataset. Some IDS models are created as a baseline in a single modeling approach using the complete attributes of the dataset. Jaw & Wang (2021) developed a promising hybrid feature selection with an EC, which effectively chooses significant attributes and offers reliable attack detection. The hybrid feature selection efficiently selects subsections of attributes with higher correlation, which significantly decreased the system intricacy and improved the generalization of training algorithms, both of which are symmetry learning features. Furthermore, using VM and AoP, this work exploits KNN, OneClass SVM, Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and an expectation-maximization (KODE) as an improved classification model that reliably categorizes the unequal probability distributions between normal and malevolent samples. HFS-KODE realizes notable outcomes through UNSW-NB15, NSL-KDD, and CIC-IDS2017 databases

in terms of performance measures. 2.5 RESEARCH GAPS Despite there being numerous topical research works to sense network intrusion, still some challenges (e.g., reduced ADR, deprived accuracy, high FAR, etc.) due to huge data streams that are not explored adequately. The main difficulties in developing IDS are the deficiency of rich data to train models owing to the complex nature of the security domain, the requirement for an extravagant feature engineering phase directed by human domain experts, and the incompetence of prevailing approaches to form reasonable models. From the literature, most of the IDS models hampered by dataset-enabled difficulties are the primary cause why the application of optimization algorithms seems impractical. Prevailing methods are appropriate for the DoS type of threats. These methods are not proficiently identifying other types of threats including probing, remote to the user, and
44 user to root. Most of the existing models only consider small datasets without applying any method to manage large datasets and speed up the data analytics process (e.g., learning time) to satisfy the significant requirements of volatilized and distributed network scenarios. Likewise, applying ML algorithms on huge databases includes noise components that reduce the effectiveness of the detection process. Up till now, the traditional ML algorithms employed in IDSs have potential issues in scaling up to meet the security requirements of the networks. To avoid overfitting problems, small datasets need approaches that have low complexity or high bias. Moreover, most of the studies considered only accuracy and precision as performance metrics to analyze the system. However, recall, F1-measure, and false alarm rate are effective metrics mostly in cases where classes are not evenly distributed. Therefore, it is vital to understand the context before choosing evaluation metrics because each model attempts to resolve an issue with a different objective function using different databases. In reality, making predictions on imminent data is highly anticipated. Therefore, it is an important issue that the IDS models need to handle. Furthermore, threats are transitory and the procured traffic traces have to be treated quickly in an extremely dynamic network configuration.
In other words,
the reliability of data needs to be exposed correctly and quickly. From the state-of-the-art review, the characteristics of a perfect IDS are identified as follows: 1. The application of IDS should not add any extra vulnerability
to the existing system. Besides, a perfect IDS requires exploiting the least
network resources and should not decrease the network performance by increasing system complexities.
45 2. It should deliver protection against threats without any human intervention. In unison, it should operate always evidently without others knowing about its functionality. 3. An IDS should be able to respond to multiple threats with equal capability.
At the same time,
it should have the capability of self-defense and monitor itself to decide whether it has been compromised by the attacker or not. 4. An ideal IDS should be simple and lightweight. There should not be any trade-off between ADR and detection overhead. It should work equally for both sparse and dense topologies without decreasing the ADR and network performance. 5. It should be interoperable with other attack detection models. IDS should not only sense the threat but also need to be able to recognize the origin of the threat. 6. It should deliver maximum ADR and minimum FAR with the least training and testing time. By bearing these requirements in mind and bridging the research gaps with respect to classification accuracy, ADR, FAR, and
computational cost,
and also overcoming the limitations of prevailing IDS in literature, this work introduces an accurate and efficient IDS by assimilating the concepts of ensemble classification with attribute selection. 2.6 SUMMARY The state-of-the-art literature has revealed that investigators have achieved implausible development in increasing the classification accuracy of IDSs with reduced FAR through applications of feature selection and ensemble learning. Nonetheless, there are no research works that deliver all- inclusive experimentations on the abilities of an ML-based ensemble IDSs
46 that contains balanced forest, random forest, and C4.5 decision tree with comprehensive comparative analysis using real-world databases including NSL-KDD and CIC-IDS 2017. Moreover, this model offers an efficient and reliable optimized attribute extraction method, which effectively chooses significant attributes among the myriad and imbalanced data samples of the several standard databases to facilitate a real-time, efficient, and credible classification of cyber threats. It is worth mentioning that the objective of applying these two methods simultaneously is to target symmetry (i.e., both methods are critically important and equally
contribute to resolving
the issues of cyberattack identification).

47 CHAPTER 3 DESIGN AND ARCHITECTURE OF THE PROPOSED INTELLIGENT CLASSIFIER USING ENSEMBLE TECHNIQUE Of late, a sizeable number of global corporate sectors and government organizations are undergoing the problem of security breaches or attacks against network infrastructures. With the ever-increasing scale and landscape of cyber threats, the necessity of a more robust network security infrastructure is being realized almost in every sphere of our connected life. The financial gains entice cyber criminals to unremittingly attack technology service providers, financial institutions, and the related network infrastructure. There are other motivational aspects too. Furthermore, breaches or threats and related protective tools are constantly evolving and the participants (i.e., both defenders and invaders) are continuously working and refining their tactics and countermeasures to stay ahead (Khader et al. 2021). Hitherto, numerous architectures have been developed by several research communities all over the world to thwart cyber threats or reduce the impairment caused by them. This work develops a novel ensemble-based intelligent IDS framework to sense and identify the inbound network traffic into normal activity or the malicious attacks with improved accuracy. This chapter describes the overall structural design of the proposed ICET framework. The proposed framework integrates the feature selection process, optimization

48 algorithm, and ensemble classification to develop highly-efficient IDS for detecting and classifying intrusive activities in a network system. This intrusion detection framework implements an ensemble classification approach to identify cyber threats. This model also proposes CFS and BIO algorithms to increase the performance of threat detection and classification. Furthermore, this chapter provides a brief description of the datasets used in building and testing the proposed IDS model. 3.1 INTRODUCTION Cybercriminals continue to wreak havoc across the globe. They are motivated to steal sensitive data, gain illegitimate returns, and find out new targets. Intrusion detection is a process of sensing and investigating data packets and retorts immediately when a cyberattack happens with the signs of intrusion. From the extensive literature survey, it is found that a single learner may not be powerful enough to create a good anomaly-based detection model due to high-dimensional and imbalanced datasets. The restrictions of the usage of a single anomaly-based classifier lead to the idea of building a more complex, but less accurate and lower FAR model. Several studies demonstrate that the performance of the ensemble approach is higher as compared to the performance of individual classifiers. Furthermore, some studies have established that the implementation of the ensemble model can provide a versatile architecture and certainly increase the classification accuracy and attack recognition rate (Gao et al. 2019; Sharma et al. 2019). With an appropriate feature selection process and optimization algorithm, this model seems to increase the prediction performance (Zainal et al. 2009). By applying ensemble learning along with an appropriate voting mechanism, this research can decrease the indecision in the generalization performance of using an individual base learner (Rajagopal et al. 2020).

49 These are the reasons to select an ensemble model for improving the performance of individual classifiers in an anomaly-based detection model. 3.2 ARCHITECTURE OF PROPOSED ICET MODEL This research introduces an intelligent ensemble classification approach composed of three individual basic classifiers with different learning models to handle the problem of the accuracy and FAR in existing IDS models. The ICET model exploits an intelligent ensemble classifier to increase the accuracy as well as ADR, and reduce the FAR in classifying the intrusive activities significantly. The proposed ICET includes two major elements including a feature selection module and an ensemble classifier. To handle high-dimensional inbound traffic in large networks, the feature selection module exploits a CFS algorithm to select the appropriate features. To select the best classifier set from a pool of classifiers (this work considers around 30 classifiers in the pool), this study considers classifier diversity as the most vital characteristic. For selecting the best subset of a diverse collection of base-classifiers, this study uses multi-objective optimization via a GA rather than depend on heuristics or insubstantial predefined user parameters (which is explained in Section 5.2.1). The proposed feature selection module calculates the correlation of the observed attributes and selects the optimal subset for training and testing processes. Besides, it exploits the optimized RelieF algorithm to calculate the quality of attributes. The attributes with a low-quality index are eliminated to reduce the dimensionality of the feature space. The performance of the proposed feature selection approach is further enhanced by integrating CFS with the BIO algorithm for hyperparameter optimization. This integration (BIOCFS) is embedded in the proposed ensemble classifier to increase the performance of the IDS.

50 This ensemble learner exploits a correlation-based feature selection algorithm with a bat-inspired optimizer to select the best feature subsets to detect different attacks. It integrates three basic classifiers including RF, BF, and C4.5 using the rule of AoP. Indeed, the base classifier RF is one of the widely used tree-based ensemble classification algorithms. It integrates the concept of bagging and random subspace algorithms which selects features arbitrarily at the node level. BF classifier is a decision forest that uses a decision forest algorithm called Forest by Penalizing Attributes (Forest PA). This classifier aims to construct a set of extremely balanced and precise decision trees by taking advantage of all non-class features available in a dataset. C4.5

is one of the efficient algorithms intended to build a decision tree by applying the ID3 algorithm. It determines the optimum split to increase the gain ratio by considering every node in the tree. The

proposed classifier exploits a VM along with the

AoP rule to tackle the multi-label data imbalance issue in the classification process. The key goal of the proposed ICET framework is to make the system effective and improve its performance with respect to higher accuracy

and

decreased false alarm rate. Figure 3.1 depicts the structural design of the proposed ICET model which includes three modules namely (i) data preprocessing; (ii)

feature selection, (ii) training the ensemble classifier, and (iii) attack detection and classification. For solving

class imbalance problems, BIOCFS is used to select the ideal subset from the dataset comprising a very large number of attributes and eliminate unrelated attributes from the dataset.

The base classifiers of

the intended EC are selected after extensive analysis of assembling various base learners such as BF, RF, and C4.5 to identify normal as well as known and anonymous intrusive activities. The cooperation of BF, RF, and C4.5 classifiers provides better classification performance. Moreover, a

voting mechanism is employed to assimilate the distribution of the

51 probability of the simple learning algorithms

to increase the classification accuracy.

Figure 3.1 also depicts the critical steps involved in the integration of the BIOCFS algorithm and EC approach. Firstly, the dataset selected for implementation is given to the preprocessing phase to transform unpolished data into useful information for analysis and understanding. The next phase involves the FSP, where this model estimate and selects attribute subsets with higher correlation using a correlation-based feature selection method with a bat-inspired optimization algorithm. The learning process exploits these attributes to construct an effective and dependable EC containing

a

random forest, balanced forest, and C4.5 decision tree. Furthermore, we use the final trained model (i.e., detection engine) for classification with the AoP rule and voting mechanism to categorize the test dataset as normal or different cyberattacks within the test dataset according to the selected features during the feature selection process. Figure 3.1 The architecture of the proposed ensemble classification model

52 3.3 DATA PREPROCESSING PHASE Data preprocessing is a vital phase before any ML algorithm can be implemented since the algorithms learn from the dataset and the learning results profoundly hinges on the appropriate input data (i.e., feature) required to solve a specific problem. Data preprocessing is a method of transforming this raw (unpolished) data into a required form so that useful information can be extracted from the dataset, which is given into the learning model for making effective decisions. Preprocessing is the most convoluted but essential phase in

an information system since it can reduce the size of data by encoding mechanisms

and increase the efficiency of data mining processes. Besides, datasets are frequently gathered from heterogeneous sources and can be redundant, noisy, partial, and incompatible. Raw data before data preprocessing is generally not appropriate for getting correct implications. This section describes an overall outline for data cleaning and covers various data preprocessing steps involved in this study. 3.3.1 Data Cleaning and Removal of White Spaces The objective of data cleaning is to confirm that the dataset employed in the analytic process offers an exact depiction of the datasets that are being studied. Datasets frequently comprise many abnormalities including codes or outliers to specify different forms of missing data (i.e, truly missing, denials, skip patterns, etc.), which can considerably affect the outcomes of the analytic process if the abnormalities are accidentally incorporated. Conversely, databases can be prepared to generate superior results using data management and processing methods that make improved analytic decisions and results owing to a more illustrative and precise database. This proposed model begins with an indispensable process of eliminating improper, redundant, or partial samples and determining missing values within the specified databases, generally called data cleaning or scrubbing, which

53 guarantees effective, accurate, and dependable classification. The redundant attributes of NSL-KDD and CIC-IDS 2017 datasets are removed using the in- built functions available in the simulator. Similarly, this model removes

the blank or white spaces of the multi-class labels to circumvent confusing the

systems in the

learning process. 3.3.2 Label Encoding Some attributes of the databases are categorical values, which are not

a suitable input for most ML techniques like

a

random forest. Hence, it is indispensable to encode these attributes into numerical values before using them in this model. One-hot and ordinal encoding techniques are most popular in the domain of data mining for converting categorical values into numerical ones. By applying these methods, this work calculates integer values for each categorical value (Zhou et al. 2020). For example, the values of 0, 1, and 2 are allocated to the protocols ICMP, TCP, and UDP, respectively for using the NSL-KDD dataset. A similar method is used for residual categorical values to convert into numerical values. Hence,

binary class labels of all the samples are already in 1's and 0's. 3.3.3 Data Normalization

As stated earlier the imbalanced measures of attributes degrade the effectiveness of the prediction process. Hence, it is essential to normalize these differences

to

a satisfactory level.

For example, the large values of "Duration", "Dst Bytes", and "Src Bytes", of the CIC-IDS2017 database can dominate the trivial values of "Num Failed Logins".

Accordingly, this work analytically selects the min-max method (Kotsiantis et al. 2020) to make the attributes of the database within the standardized level

of [0, 1], facilitating an effective understanding of the

information. Equation (3.1) shows the formula of the min-max technique.

54 ̄= – – (3.1)

where ̄is the normalized outcome; and are the minimum (0) and maximum (1) values of feature ; and the residual values would be within these ranges, which makes the attributes have the base point and same range. Therefore, this method disables the concerns of bias, greatly decreases the time for learning and validation, and facilitates a rapid convergence rate, thus increasing the dependability and

performance of the classification process (Jaw & Wang 2021). 3.4 FEATURE SELECTION In today's digital world, ML has become a substantial slice of our lives. When used in real-life applications, ML experiences the difficulty of high-dimensional datasets. Redundant and unrelated attributes can be observed in the datasets. The enactment of classifiers used in prediction is suffered from these redundant attributes. The main phase in emerging any decision support system is to select significant attributes. In the same way, the immense unrelated and superflux attributes of real-life applications have significantly challenged the efficient and dependable identification of cyberattacks by IDSs. Also, IDSs constructed on these challenges frequently need more time for learning and testing processes with high demand on computing resources and, remarkably, very hard to understand. Therefore, the FSP is a crucial phase in this study. FSP aims to decrease the number of selected attributes or variables involved in the classification process. It selects more representative attributes indispensable to create an alarm when an intrusion is identified. Moreover, it also involves removing features that do not act any significant role in the attack identification process. Besides, the redundant and irrelevant attributes

55 are eliminated; therefore, the total temporal overhead of the algorithm is reduced with substantial performance improvement in terms of prediction

accuracy and generalization. The generalization is a substantial property of classification as it supports the algorithm to evade overfitting on a

particular database. This work offered an optimized attribute selection process, which includes the CFS algorithm, BIO algorithm, and rule-based engine. The proposed feature selection algorithm computes the relationship between all features and classes. The feature-class correlation with a resilient relationship is more likely to be selected, called feature assessment. BIO algorithm measures the significance of each attribute according to this feature assessment. If two subsets have identical assessment values, the rule engine selects the attribute subset with the minimum frequency count. Finally, the designated attributes are given as input to the proposed ensemble classifier for identifying attacks and building the model. 3.4.1 Correlation-based Feature Selection CFS is one of the traditional filtering techniques that select a

subset based on relationships among pairs of attributes using an assessment function (AF) (Singh & Singh 2018). The purpose of is to select subsets whose attributes are extremely correlated with the class but unrelated to each other. The attributes which show a

weak correlation with the class label have to be eliminated

and recurrent attributes are selected due to a robust correlation with at least one of the rest. The choice of an attribute will depend on the number of predicted classes in sample space not up till now expected by

different attributes. Decisively, CFS is one of the multivariate attribute selection approaches that exploit a heuristic search to analyze the ideal attributes to be

56 employed in the dataset. The best attributes are selected based on the level and the substantial correlation value between the attribute and its class. Consequently, this aptitude makes CFS one of the most extensively employed methods applied for attribute selection, especially in the case of large datasets. However, CFS provides more significant solutions that support decision- makers in improving the performance of the decision-making system. However, CFS is more laborious than other feature selection methods (Mohamad et al. 2021). Several researchers attempt to improvise and enhance the performance of the CFS algorithm by assimilating it with other optimization algorithms (Chormunge & Jena 2018; Mursalin et al. 2017). For example, Tiwari & Singh (2010) proposed an integrated feature selection approach to combine original CFS with GA to select the optimal subset from the available features. Mohamad et al. (2021) integrate the best first search with CFS for the feature analysis process. This work attempts to combine a bat-inspired optimization algorithm with the CFS process to select the optimal subset to increase the performance of the selection process. At present, several metaheuristic optimization algorithms including GA, particle swarm optimization, artificial bee colony, simulated annealing, firefly algorithm, cuckoo search, and bat-inspired optimization algorithm are used to improve the performance of the attribute selection process (Sayes et al. 2007). Some researchers proved that the BIO has become more robust than GA, particle swarm optimization, artificial bee colony, and other algorithms (Alia & Taweel 2021). Therefore, BIO is observed as one of the potential solutions for the problem in data mining and management processes like feature selection and detection of attacks in IDS models. 3.4.2 Bat-inspired Optimization
57 The bat-inspired optimization algorithm is a comparatively new population-based metaheuristic optimization algorithm that mimics the echolocation characteristics of bats. Bats are interesting animals and they are the only mammals with wings that have greater ability of echolocation to sense prey, circumvent obstacles and find their resting crevices in the dark. They produce a very loud sound pulse and listen for the echo that reflects from the neighboring objects. Their pulses differ in properties and can be related to their stalking approaches, based on the species. Most bats generate short, frequency-modulated signals to sweep through about an octave, whereas others more often generate constant-frequency signals for echolocation. Their signal bandwidth differs according to the species and is frequently augmented through harmonics. The performance of the CFS algorithm is improved by integrating it with BIO for arbitrarily selecting the optimal attributes. BIO is suggested for the selection of optimal attributes from the dataset. Likewise, the indispensable attributes are designated from the dataset according to the types of features. The unrelated attributes are removed by considering them as obstacles. The optimization of feature selection is discussed in Section 4.3. 3.4.3 Rule-based Decision Engine A rule-based engine is an algorithm or heuristic that picks one among many selections based on the correlation between the data and the rule. Moreover, it matches the input values to the listed rules and competently selects the optimal value for implementation is known as conflict resolution (Bridges et al. 2019). The rule base engine returns a subset of the attribute ( ) with minimum attributes ( ) in case there are several subsections for attributes (&lt; 1) with related fitness measures; else selecting attributes subsections with maximum fitness value ( ) to the base learner as
given in Equation (3.2).
58 = { , ∈ &lt;∩ , ∩ ∅ (3.2) 3.5
ENSEMBLE CLASSIFICATION In ensemble classification
approaches, several different, unbalanced, and good classifiers are combined in a particular way (
Feng et al. 2018). The EC
approaches are prevailing to solve the classification problem and cooperatively achieve results with higher accuracy and dependability by employing and assimilating several individual classifiers (Li & Sun 2013). The conventional fields for using EC methods are computational reason, statistical reason, and representative issue. For instance, in some cases, there is a problem when the classification is a computationally too exhaustive and laborious process for a single classifier to define an appropriate hypothesis. In some cases, an individual
classifier may cause a feeble result if the input dataset is not enough to train the learning
process. In some other cases, a single classifier is not sufficient to represent the research space.
Boosting and bagging are the two most well-known approaches in collaborative learning, usually generating better classification solutions and being extensively selected to construct several ensemble frameworks (
Freund & Schapire 1996).
Besides, the other recognized collaborative learning approaches such as Stacking (
Hung & Chen 2009), Bayesian parameter averaging (
Friston et al. 2010),
and voting (Hu 2018) are used for increasing the efficiency of the classification process. Similarly, ensemble approaches have been used to increase classification accuracy in several applications, including
the
identification of intrusive activity. Furthermore, ensemble classifiers deliver tools to investigate the similarity between malicious and genuine samples.

This work focuses on an EC model that combines three different classifiers, namely C4.5, RF, and BF to increase the predictability of
59 IDS. These classifiers are employed to implement a voting mechanism using the AoP rule. 3.5.1 C4.5 Decision Tree
A classification tree or decision tree is an intuitive model which maps the observations about a parameter to decisions about its target value. It targets to categorize the input data streams into the equivalent class labels based on their attributes. The key concept behind the construction of DTs is known as the Iterative Dichotomizer 3 (ID3) algorithm introduced by Quinlan. ID3 algorithm forms a DT by applying a top-down methodology where a training dataset is used to test the features through greedy searching. It computes the information gain and entropy to select a particular feature to test at every node in the DT. In this tree configuration, leaf nodes denote labels (or classes), non-leaf nodes represent the possible value of attributes, and branches denote correlation among attributes that bring about the decisions. C4.5 is a powerful and reliable DT algorithm. It provides results with better classification accuracy irrespective of the volume of the data to be mined. The algorithm can process an incomplete training dataset, and prune the resultant DT to optimally select the decision path as well as decrease its dimension. It also has the potential to handle numerical features and continuous data streams using the process of binarization. The continuous features are substituted by the discrete ones by applying predefined values which divide the dataset into two parts. Even for C4.5 and other approaches that can directly handle continuous data streams, learning is often less competent and ineffectual. C4.5
algorithms
were
developed to create a decision tree from a dataset using the ID3 algorithm (Hssina et al. 2014). This algorithm finds the
60 ideal split to maximize the gain ratio (GR) by visiting each node in the decision tree.
For classification, an attribute with the maximum GR is selected as a dividing attribute for the node. Infogain denotes how much indecision in the dataset is decreased after it is divided based on
the selected attribute.
C4.5 algorithm can represent or categorize both continuous and discrete attributes and can neglect missing information. Some research works revealed that the C4.5 classifier shows some vulnerability in applications with continuous data streams. However, it deals with numerical features using a new binarization approach, or by adopting some efficient methods in the C4.5 discretization technique. In order to achieve binarization, first, the continuous features are sorted using the quick sort method with a complexity of (( )). Hence, the time complexity of this process is very high and it is not useful for huge databases. Besides, numerous authors proved that the performance of the training process depends on the sorting technique used for continuous features. The generalization limit of this feature space is the more vital issue in implementing the C4.5 classifier. The designated predefined value cannot determine and reveal the potential of generalization of the numerical feature. Hence, this classifier may exploit features with a poor generalization enactment to split data. Deciding based on those features will increase the size of DT and consequently reduce the classification accuracy. 3.5.2
Random Forest Classifier It is also a decision tree-based classification approach developed by Breiman (2001). It works by creating several decision trees.
It accepts a large number of input parameters without variable exclusion and categorizes them according to their reputation.
More specifically, RF employs a group of classification trees. Each tree in the forest provides a vote for the most recurrent class in input records.
RF classifier considers only fewer parameters
61 as compared with the other machine learning methods (e.g.,
ANN, SVM, etc.).
The key to the success of this classifier is the formation of the forest.
In order to train each tree in the forest, the RF classifier built a bootstrapped subset of the training process. Hence, each tree utilizes approximately 2/3 of the training dataset. The idle
instances are known as the out-of-bag instances which employ in
the
inner cross-validation process to calculate the classification accuracy. Furthermore, RF has the minimum computational overhead, and it is oblivious to the outliers and parameters. Also, the over-fitting issue is less related to single decision tree-based approaches and it is not required to prune the tree which is a difficult and time-consuming process. 3.5.3
Balanced Forest
Classifier The balanced tree uses the concept of FPA to classify the inbound traffic into normal or malicious.
Contrasting some classic classification approaches found in the literature, BF
exploits a subset of the non-class features. This approach forms a group of decision trees with higher accuracy based on the strength of all non-class features existing in a dataset. Subsequently, weight allocation and weight augmentation policies are considered to preserve accuracy and strong diversity. BF will arbitrarily calculate the weights for features that

are present in the most recent tree. 3.6
VOTING MECHANISM The estimation of each partaking classifier in the ensemble approach may be treated as a vote for a specific class, i.e., genuine or malicious. It takes advantage of many single classifier approaches and exploits a combination rule for making decisions. For example, maximum probability, minimum probability,
an average of probabilities, a
product of
62 probabilities, and majority voting are used as combination rules. In this work, we apply an average of probabilities method to make a decision.
In this approach,
the class label is designated according to the maximum value of AoP. Let denotes the number of classifiers = { 1 , 2 ,.... }
with c classes Ω = {Ω 1 , Ω 2 ,....Ω }. In our experiment, we choose
the values = 15 and = 3. A classifier : → [0,1]
gets an input instance ∈ and gives the output as a vector (1 |), (2 |),.... (|),
where (|)
represents the probability allocated by that input instance fits into class . Let is the AoP allocated by the classifiers for each class. It can be estimated using Equation (3.3). = 1 ∑ (|) =1 (3.3)
Consider = [1 , 2 ,...]
is the set of AoPs for c classes and is allocated to the weight if
has a
higher value in . 3.7
TRAINING AND TESTING PHASE To realize a more precise evaluation of a model's enactment, the - fold cross-validation (CV) is used (Vaiyapuri & Binbusayyis 2020). The proposed ensemble learner assigns equals 10. In a 10-fold CV, the whole database is divided into 10 parts. For each folding, one part of the database is employed for testing and the other sections are used for training. Then, we compute the average value of results across all ten autonomous trials. The advantage of this method is that all testing samples are self-regulating and the reliability of the results could be increased. It is worth mentioning that only one repetition of the 10-fold CV will not generate acceptable outcomes for assessment due to the uncertainty in data fragmentation. Hence, all the outputs are specified on an average of 10 runs to obtain accurate results. Indeed, a separate set of data samples for testing and validation are not presented in the database. Therefore, we have deliberately split the existing
63 dataset into 70% for learning, 20% for testing, and 10% for validation. Furthermore, since we used 10-fold CV, the total data samples are split into 10 parts (each of 10%). Now, one fold (10%) is applied for testing, while the residual samples (90%) are split for validation and training. The utilization of 10-fold CV guarantees that each slide in the dataset gets to be in a trial only once. 3.8 SUMMARY
The intrusion detection system
is extensively used to identify attacks and to achieve integrity, confidentiality, and availability of sensitive information.
Though several unsupervised and supervised machine learning methods have been employed to improve the efficiency of the
intrusion detection process,
it is still a challenge to handle several redundant and unrelated data in big data scenarios. In this work, we develop an IDS using machine learning techniques to improve the performance of attack detection. In order to cope with high dimensional feature-rich traffic in large networks,
this work develops a novel ensemble-based intelligent IDS framework to sense and identify the inbound network traffic into normal activity or the malicious attacks with improved accuracy. This chapter describes the overall structural design of the proposed ICET framework. The proposed framework integrates the feature selection process, optimization algorithm, and ensemble classification to develop highly-efficient IDS for detecting and classifying intrusive activities in a network system. This intrusion detection framework implements an ensemble classification approach to identify cyber threats. This model also proposes CFS and BIO algorithms to increase the performance of threat detection and classification. Furthermore, this chapter provides a brief description of the datasets used in building and testing the proposed IDS model.

64 CHAPTER 4 CORRELATION-BASED FEATURE SELECTION WITH BAT-INSPIRED OPTIMIZER FOR DEVELOPING AN EFFECTIVE IDS MODEL Attribute (or feature) selection is the process of choosing relevant features from the database that are more pertinent to the classification problem (Karimi et al. 2013). The definition of relevance varies from approach to approach. According to its indulgence of importance, an attribute selection method statistically articulates a criterion to assess a set of attributes engendered by a method that searches over the feature space. Kohavi & John (1997) categorized attributes into three distinct types, viz., weakly relevant, strongly relevant, and irrelevant attributes. The strong relevance of an attribute specifies that the attribute is always essential for an optimum subset; it cannot be eliminated without distressing the initial class dispersal. Weak relevance advocates that the attribute is not always essential but may become indispensable for an optimum subsection under certain conditions. Irrelevance designates that the attribute is not essential at all. Attribute selection methods help in building an accurate predictive model by selecting attributes that will provide better performance and less complexity while demanding fewer data. It removes irrelevant or redundant attributes from the initial dataset to reduce the dimensionality of the feature space in the dataset, reduce computational and storage complexity, and make it easier to interpret and analyze data. This chapter proposes a correlation-

65 based feature selection with a bat-inspired optimizer to increase the effectiveness of the IDS model in terms of improved accuracy and reduced FAR. 4.1 INTRODUCTION The huge redundant and irrelevant attributes of real-time applications have significantly challenged the efficient and dependable identification of cyberattacks. Also, IDSs fabricated on these issues generally consume more

time for learning and testing the models

with high demand for computational facilities and, remarkably, very hard to understand (Zhou et al. 2020.). Therefore, attribute selection is an important pre-processing phase in any ML classifier which targets to remove massive unrelated and redundant attributes while maintaining or even improving the performance of the IDS. The key goal of this attribute selection

process is to minimize the number of selected variables or attributes into a reduced set and to select the most significant attributes indispensable to create a notification when an intrusion is identified. Also, it involves

removing redundant as well as unrelated features and information that does not act any significant role in IDS; therefore, the total time, space, and computational overhead of the detection process is reduced with substantial performance augmentation in terms of

classification accuracy and generalization. The generalization is an important property of classifiers as it helps the algorithm evade overfitting problems on a particular dataset

This study proposes an optimized feature selection method, which includes a correlation-based feature selection process and a bat-inspired optimization algorithm. A subset evaluator calculates the relationship among all features and classes (Ran et al. 2019). The feature-class correlation with a sturdier relationship (feature assessment) is more likely to be selected. The bat-inspired algorithm evaluates the significance of each property according

66 to this assessment. If two-element subsets have identical values, the rule engine selects the attribute subset with the lowest frequency count. Finally, the designated attributes are given input to the proposed ensemble classifier for threat classification and produce the model. 4.2 CORRELATION-BASED FEATURE SELECTION Attribute selection is the process of recognizing and eliminating as much unrelated and unnecessary information as possible. This decreases the data dimensionality and may enable learning algorithms to run faster and more efficiently. In some scenarios, the accuracy of the classification system can be enhanced; in others, the output is more compact and thus the depiction of the target concept is easily inferred. This work claims that attribute selection for classification tasks in ML algorithm can be achieved by applying the concept of correlation (i.e., a degree of dependence or predictability of one variable with another) among attributes and that such an attribute selection process can be useful to common ML algorithms. This section illustrates the concepts of a CFS based on this claim; the following sections study the performance of CFS in attack classification under different conditions and prove that CFS can extract useful and most relevant (i.e., significant) attributes for ensemble classifiers developed in this study. 4.2.1 Defining Relevance In this section, the definition of relevance that has been suggested in the literature is presented. The input to any ML algorithm is a set of learning samples. Every sample is a component of the set 1 × 2 × … where is the domain of the i th attribute. Usually, learning samples are tuples ⟨ , ⟩ where is the class label (output). Given a sample, the value of attribute is denoted by . The role of the induction algorithm is to make a structure (e.g., decision tree) such that, given a new sample, it is feasible to

67 correctly calculate the label . This work assumes a probability measure on the space 1 × 2 ×... × . This study does not define any hypotheses on the attributes or the class; they can be structured, linear, continuous, or discrete and the output may be multi-valued or a single-valued vector of random size. According to Gennari et al. (1989), an attribute is significant if its values differ systematically with class membership. More precisely, an attribute is beneficial if it is related to or predictive of the class; or else it is irrelevant. Kohavi & John (1997) formalize the following definitions to explain the characteristics of the features. Definition 1: An attribute is said to be relevant to a class label if appears in every Boolean formula that denotes and unrelated otherwise. Definition 2: An attribute is relevant iff there is some and for which (=) &lt; 0 such that (= | =) ≠ (=) (4.1) The attribute in Definition 2 is significant if knowing its value can alter the assessments for , more precisely, if is reliant on . It is worth mentioning that this description fails to represent the significance of attributes in the notion of parity, and may be reformed as follows: Let be the collection of all features except , i.e., = { 1 ,.., −1 , −2 ..... }. The term denotes a value allocated to each attribute in . Definition 3: An attribute is significant iff there exists some , , and for which (=) &lt; 0 such that this definition, is significant if the

68 probability of the label (given all attributes) can vary when the information about the value of is removed. (= | = | =) ≠ (= , =) (4.3)

Definition 4: An attribute is significant iff there exists some , , and for which (= , =) &lt; 0 such that (= | = , =) ≠ (= , =) (4.4)

The following example illustrates all the above mentioned definitions. Example 1: Let attributes 1, 2 ,.... 5 be Boolean. The feature space is such that 2 and 3 are adversely related with 4 and 5 , correspondingly, i.e., 4 = $2^{\equiv}$ and 5 = $3^{\equiv}$. There are only 8 potential combinations, and they are assumed to be equiprobable. The (deterministic) target concept is = 1 ⊕ 2 (4.5) where ⊕ denotes XOR operation. It is worth mentioning that the target concept has an equivalent Boolean expression, viz., = 1 ⊕ $4^{\equiv}$. The attributes 3 and 5 are unrelated in the robust possible sense. 1 is vital, and one of 2 , 4 can be removed, but we must have one of them. Table 4.1 shows the relevant and irrelevant features for each definition.

Table 4.1 Attribute relevance for the correlated XOR problem Definition Relevant Irrelevant Definition 1 1 2 , 3, 4 , 5 Definition 2 - 1, 2 , 3, 4 , 5 Definition 3 1, 2 , 3, 4 , 5 - Definition 4 1 2 , 3, 4 , 5

69 1. Based on Definition 1, the attributes 3 and 5 are obviously unrelated; both attributes 2 and 4 are unrelated because each can be substituted by the negation of the other. 2. Based on Definition 2, all the attributes are unrelated since for any output value and attribute value , two samples agree with the values. 3. Based on Definition 3, each attribute is significant, as calculating its value alters the probability of 4 of the 8 possible combinations from 1 8 to zero. 4. Based on Definition 4, 3 and 5 are clearly unrelated, and both 2 and X4 are unrelated, since they do not include any information to 4 and 2 , correspondingly. Even though such meek adverse relationships are improbable to happen, domain limitations provide an analogous outcome. When an insignificant feature such as color is coded as input to a classifier, it is normal to utilize a local coding, where each value is denoted by a pointer variable. For instance, the local coding of a 4-valued number {a, b, c, d} would be {0001, 0010, 0100, 1000}. In this encoding scheme, any sole pointer variable is superfluous and can be calculated by the rest. Therefore, most descriptions of relevancy will state all pointer variables to be unrelated. Hence, two degrees of relevance are essential. Definition 4 describes robust relevance. Sturdy relevance advocates that the attribute is essential since it cannot be eliminated without loss of classification accuracy. Definition 5: (Weak relevance): An attribute is weakly relevant iff it is not strongly relevant, and there exists a subsection of attributes ′ of for which there exists some , ,

and ′ with (= , ′ = 1) &lt; 0 such that

70 (= | = , ′ = 1) ≠ (= | ′ = 1) (4.6)

Weak relevance indicates that the attribute can occasionally contribute to classification accuracy. Attributes are relevant if they are either sturdily or weakly relevant, and are unrelated otherwise. Unrelated attributes can never contribute to classification accuracy, by definition. In Example 1, attribute 1 is sturdily relevant; attributes 2 and 4 are weakly relevant; and 3 and 5 are unrelated. Experimental results from the attributes selection literature show that in consort with unrelated attributes, unnecessary attributes should be removed as well (Pirgazi et al. 2019). An attribute is said to be unnecessary if one or more of the other attributes are strongly related to it. The above definitions for redundancy and relevance create the following hypothesis, on which the attribute selection process proposed in this work is based: “An optimal attribute subset covers attributes extremely related with the class, yet unrelated with each other.” If the relationship between each of the attributes in a test and the outside variable ( ) is calculated, and the inter-relationship between each pair of attributes is given, then the relationship between composite tests containing the summed attributes and can be predicted from Equation (4.7). = $^{\equiv}$√ + (−1)$^{\equiv}$(4.7) where , and are the component, outside variable and composite variable, respectively. Then denotes the relationship between the summed attributes and is the number of attributes, is mean relationships between the attributes and the outside variable, and is the mean inter-relationship between attributes. Equation 4.6 is, in fact, Pearson’s relationship factor, 71 where all variables have been standardized. It reveals that the relationship between an outside variable and a composite is a function of the number of attribute variables in the composite and the value of the inter-relationships among them, together with the value of the relationships between the attributes and . By applying two descriptive values for in Equation 4.6, and allowing the values of and to change, the formula is resolved for . From this equation the following conclusions can be drawn: • The higher relationships between the attributes and , the higher the relationship between composite and . • The lower the inter-relationship among the attributes, the higher the correlation between the composite and . • As the number of attributes in the composite increases (assuming the added attributes are the same as the original attributes regarding their average inter-relationship with the other attributes and with), the relationship between the composite and increases. In this work, Equation 4.6 is employed as a heuristic measure of the "merit" of attribute subsets in the classification process. In this case, becomes C (the class); the problem remaining is to find out appropriate methods of calculating the attribute class relationship and attribute-attribute inter-relationship. Supervised learning processes contain various data attributes, any of which may be binary, nominal, ordinal, and continuous. In order to have a common base for estimating the relationship, it is required to have a general method of handling various types of attributes. Discretization is employed as a preprocessing phase to transform continuous attributes into insignificant ones. For classification it is clear that unnecessary features should be removed —if the predictive ability of the

72 given attribute is covered by another then it can securely be removed. Some ML algorithms need this to increase classification accuracy. Conversely, for data mining applications where entire outputs are of utmost significance, it is not always obvious that unnecessary attributes should be removed. For instance, a rule may give more "sense" to a user if a feature is substituted by one highly related to it. The proposed CFS algorithm adopts this condition by providing a report generation facility. For any given feature in the final subset, CFS can list its close alternates, either regarding the total merit of the final subset of the feature in question was to be substituted by one of the alternates, or simply relationship with the feature in question. 4.2.2 Correlating Nominal Features When all attributes and the class are handled in the same way, the attribute-class relationship and attributes-attribute inter-relationship in Equation 4.6 may be computed. Study on DT induction has provided several techniques for calculating the quality of a feature —that is, how predictive one feature is of another. Metrics of feature quality describe the inconsistency that exists in the group of samples related to the values of a specific feature. Hence, they are occasionally called impurity functions (Disha & Waheed 2022). A crew of samples is considered pure if each sample is the same in terms of the value of a second feature; the crew of samples is impure (to some degree) if samples vary with the value of the second feature. DT induction usually only comprises determining how predictive features are of the class. This is related to the attribute-class relationship in Equation 4.6. To measure the merit of an attribute, the attribute-attribute relationship must be calculated. Since DT classifiers execute a greedy simple- to-complex hill-climbing search, their overall inductive bias is to favor reduced trees over larger ones. One aspect that can influence both the size of the tree and how well it generalizes to new samples is the bias inherent in the

73 feature quality index employed to choose among features to test at the nodes of the tree. Some quality indexes are identified to illegally favor features with more values over those with fewer values (Kononenko 1995). This can lead to the building of bigger trees that may overfit the learning database and generalize poorly. Likewise, if such indexes are employed as the correlations in Equation 4.6, an attribute subsection comprising attributes with more values may be selected—a situation that could lead to lower enactment by a DT classifier if it is controlled using such a subsection. Kononenko (1995) studies the biases of 11 metrics for calculating the quality of features (Kononenko 1995). For the inter-relationship between two attributes, a measure is required that describes the analytical capability of one attribute for another and vice versa. This study implements an algorithm called Optimized RelieF (ORelieF) to estimate the quality of features. ORelieF is a variant of the original Relief algorithm. 4.2.3 Measuring Quality of Attributes As an individual assessment filtering attribute selection approach, Relief estimates a proxy statistic for each attribute that can be employed to measure attribute 'quality' or 'significance' to the target concept (i.e. calculating endpoint value) (Urbanowicz et al. 2018). The original Relief calculates the quality of features using a KNN that finds neighbors (samples) based on the feature vector. The quality index for each attribute is calculated based on whether the nearest neighbor (nearest hit, H) of an arbitrarily designated sample from the same class and the nearest from the other class (nearest miss, M) have similar or different values. This process of regulating weights is repeated for samples (Moore & White 2007). But, the classic Relief algorithm is restricted to binary classification problems and had no tool

74 to manage missing data. Methods to extend Relief to multi-label or continuous endpoint problems are required to solve this issue. This work proposes an optimized RelieF (ORelieF) algorithm to measure the quality of attributes in the proposed CFS process. The proposed ORelieF exploits a much meeker iterative method that can easily be wrapped around any other core relief-based algorithms. ORelieF is a recursive attribute removal method. In each iteration, the minimum scoring attribute is removed from further calculation with respect to both distance calculations and attribute weight updates. On the other hand, choosing the number of iterations (n) is not trivial. Algorithm 4.1: ORelieF algorithm f←number of features Variable: number of iterations for i=1 to n do run ReliefF and calculate attribute quality index sort attributes by quality index eliminate n/f of remaining attributes with the lowest quality index end for return last ReliefF quality index for remaining attributes 4.3 BAT-INSPIRED OPTIMIZATION ALGORITHM To enhance the performance of the proposed feature selection algorithm this work implements a bat-inspired optimization algorithm that finds out the best combination of features. BIO can cope with the problem of the high-dimensional dataset and selecting the most informative attributes in a search space. This optimizer can inevitably zoom into an area where optimal solutions have been found which is accompanied by the automatic switch

75 from explorative moves to local rigorous exploitation. Accordingly, it has a quick convergence rate, at least at the early stages of the iterations compared with other algorithms. A bat-inspired meta-heuristic

optimization algorithm was developed by Yang (2010). This algorithm is motivated by the echolocation nature of bats and has been used in many engineering applications, especially in the domain of optimization

problems (Yang 2014).

Since this algorithm exploits frequency tuning, it is, indeed, the first algorithm based on the idea of computational intelligence and optimization.

Bat Algorithm has employed some rules: 1. All bats exploit echolocation to measure distance, and they also 'recognize' the dissimilarity between food/prey and background obstacles in some mystic way; 2. Bats fly arbitrarily with velocity at position with a fixed frequency , and loudness to look for prey/food. They can inevitably regulate the frequency (or wavelength) of their released pulses and regulate the pulse emission rate ∈ [0, 1], based on the proximity of their target; 3. Though the loudness can differ in many ways, we assume that the loudness fluctuates from a large (positive) to a minimum constant value .

Let each bat arbitrarily flies with a velocity at position and frequency in t time step in a d–dimensional space. The problem solution is symbolized by the bat position that can be defined by a vector. Amongst the bats in the population, for each iteration, the best solution estimated hitherto can be stored. For every iteration, the value of

76 and are updated using the method followed by Yang (2013) as given

in

below Equations (4.7) – (4.9). = + (–) (4.8) = –1 + (–1 –) (4.9) = –1 + (4.10)

The term ∈ [0,1] denotes

the random vector obtained from a uniform distribution which serves as a parameter for frequency calculation. For each bat, a new solution is selected from the current best results by implementing a random walk approach

as given in

Equation (4.10). = + (4.11)

The term ∈ [–1,1] denotes

a random vector derived from a Gaussian distribution or a uniform distribution and is the mean loudness of all the bats at this iteration. For every iteration, along with the value of the emission rate of pulses also updated as given

in Equations (4.11) and (4.12). –1 = (4.12) +1 = 0 (1 – –) (4.13)

The parameters ∈ [0,1] and &lt; 0 are constants.

The application of BIO is more byzantine than many other meta- heuristic algorithms since each agent (bat) is allocated a set of cooperating parameters including velocity, position, loudness, pulse rate, and frequencies. The algorithm includes the following modules: initialization of parameters,

77 generation of new solutions, local search, generation of a new solution by flying arbitrarily, find the current optimal solution.

The pseudocode for the proposed BIO approach is given in Algorithm 4.2.

Algorithm 4.2: BIO algorithm Input: Datasets for training and testing phases Output: Identified feature subset, 1: Initialize the number of bats n in the population, =(1, 2,... ,) , = 1,2,3...

and 2: Initialize , and 3: Initialize ( ) and 4: Initialize ( ) and ( )

to store the result 5: while 1 ≤ ≤ Max number of iterations do 6: for 1 ≤ ≤ do 7: Calculate new 8:

Update and 9: if &gt; rand (0,1) then 10: Select a from 11: Calculate 12: end if 13: Calculate ( ) 14: if((&gt;( )) and N(0,1)&gt; then 15: ( ) ← ( ) 16: ( ) ← 17: Increase

and decrease 18: end if 19: if ( ) ≥ Max (( )) then

78 20: ← 21:

end if 22: end for 23: t = t + 1 24: end while 4.4

INTEGRATION OF BIO AND CFS ALGORITHMS
In this work, we developed the BIOCFS algorithm to assess the significance and the relationship between the identified feature subsets. To create the fitness functions and to estimate
the
data integrity of the selected subset BIOCFS algorithm exploits correlation-based feature selection. For a given subset with features, = { 1 , 2 , 3 .... }, CFS approach evaluates inter-correlation among features and the average correlation between class labels and features using.
The feature selection process finds the subset of representative features essential to generate an alert when an intrusion is suspected. This process exploits the correlation-based evaluation function to calculate the degree of correlation between features in the traffic flow.
Conversely, this feature subset may not be the optimal solution due to uncorrelated features. Bat algorithm is used to eliminate the uncorrelated features and decrease the size of the datasets. In this algorithm, the position of the bat is considered the solution to a problem of interest. Bats fly to find the best solution in the search space. During its movement, every bat finds and stores the best solution. 4.5
SUMMARY Feature selection is the process of choosing relevant features from the dataset that are more pertinent to the classification problem. There are three distinct types such as weakly relevant, strongly relevant, and irrelevant
79 attributes. The strong relevance of an attribute specifies that the attribute is always essential for an optimum subsection; it cannot be eliminated without affecting the original conditional class distribution. Weak relevance advocates that the attribute is not always essential but may become indispensable for an optimum subsection under certain conditions. Irrelevance designates that the attribute is not essential at all. Feature selection methods help in building an accurate predictive model by selecting attributes that will provide better performance and less complexity while demanding fewer data. It removes irrelevant or redundant attributes from the initial dataset to reduce the dimensionality of the feature space in the dataset, reduce computational and storage complexity, and make it easier to interpret and analyze data. This study proposes a CFS with BIO to increase the effectiveness of the IDS model in terms of improved accuracy and reduced FAR.
80 CHAPTER 5 ENSEMBLE CLASSIFICATION FOR DEVELOPMENT OF
AN
EFFECTIVE IDS MODEL Several IDSs exploit individual classifiers for categorizing network traffic as normal or anomalous. Owing to the huge volume of big data, these individual classification algorithms fail to yield higher predictive accuracy with lower FAR. Recently, several ensemble methods are found in the literature to increase the performance of IDSs, constructing an ensemble leaner that can be widely used for any type of network traffic is still a challenging endeavor. As mentioned earlier, ensemble learners exploit a group of classifiers to achieve greater classification accuracy than could be acquired from an individual classifier. The key concept of EC is to integrate many ML algorithms to take advantage of every deployed base classifier for developing a more robust classification model. For selecting
the best subset of a diverse collection of individual classifiers, this study uses
a
multi-objective genetic algorithm (MOGA) rather than depend on heuristics or fragile user-defined constraints. This study proposes an ensemble classifier for integrating base learners such as BF, RF, and C4.5. The objective of this model is to build base learners for each type of
cyber attack and combine the scores of all the base classifiers
to predict the label. The final decision is achieved using
the
AoP voting mechanism. This chapter presents the details of constructing
an
ensemble leaner used in this work.
81 5.1 INTRODUCTION Ensemble learning is a swiftly increasing advanced ML technique that exploits more than one weak classifier to provide a greater classification performance as compared with single classification models for a given problem. As mentioned earlier, the key goal of
the
ensemble classifier in ICET is to increase the predictive accuracy and decrease the FAR of learners by integrating the potentials and competencies of different weak classifiers. Furthermore, ensemble approaches attempt to form a set of hypotheses or learners and integrate them to realize a reliable and efficient classification framework, endeavoring to train a single hypothesis from the learning instances (Zhou 2009). The critical potential of
the
ensemble classifier in the ICET model is dividing a problem into sub-task that could be allocated to the different units of the EC with various
algorithms and

sets of attributes to identify various cyberattacks. There may be different ML approaches in each base classifier with
an
appropriate fusion method. It does not need exceptional performance by
the base classifier but is slightly improved than haphazard prediction (Li & Chen 2020). This method has been recognized to be more efficient in scheming and implementing detection models. An individual
classifier cannot accept all kinds of input data or identify all kinds of cyber threats (Jaw & Wang 2021). Hence, various weak classifiers are used to detect cyberattacks. Conversely, the important problem with ensemble methods is selecting the apt base learners and the fusion technique to integrate the outcomes of the selected base learners (Mahfouz et al. 2020). This study selects BF, RF, and C4.5 as base classifiers and voting mechanism with
the
AoP rule to integrate the results from base learners.
82 5.2 ENSEMBLE OF CLASSIFIERS An EC includes various weak classification algorithms whose distinct results are merged in some ways to generate an ultimate decision. The combined results of ensemble learners generally provide greater enactment than the individual learner. The key objective of using
an
ensemble is to achieve greater classification accuracy by exploiting the advantages of an ensemble of base learners than any individual classifier. It decreases the probability of misclassification made by an individual classifier and also overcomes the deficiency of
the
individual classifiers. Based on the structure used for classification, the ECs are classified into two types including parallel and sequential as shown in Figures 5.1 and 5.2, correspondingly. Figure 5.1 Parallel structure of EC In parallel ensemble classifiers, the fused result is achieved via some sort of combination function in a parallel manner. Every result in the base learner is produced independently. In a sequential EC, a set of classifiers are trained sequentially. In this, when the primary classifier is unable to categorize a given input pattern then
the
secondary classifier is trained, and so on. This work selects parallel architecture to develop
an ensemble classifier in the
ICET model. The construction of ICET consists of two steps: (1) selecting the base classifiers, and (2) combining the results of base classifiers.
83 Figure 5.2 Sequential structure of EC 5.2.1 Selecting base Classifiers using Multi-objective GA Classifiers selection for an ensemble learner became a critical problem for developing effective IDS. To choose the optimum classifier set from a pool of classifiers, the classifier diversity is the most imperative characteristic to be taken into account. In this work, GA is used to select appropriate classifiers to gain good classification enactment. This approach considers accuracy and generalization as key factors to select the base classifiers. For selecting
the
best subset of a diverse collection of base classifiers, this study uses MOGA rather than depend on heuristics or fragile predetermined values set by the user. Only two user-defined factors are involved, with both being observed to have large windows of values that generate statistically indistinguishable results, demonstrating the low level of proficiency required from the user to realize better outcomes. Moreover, when given a huge original set of trained base classifiers, this study proves that a MOGA attempting to enhance classification performance and generalization will prefer specific types of classifiers over others. The total number of selected learners is also remarkably small – only 2.86 classifiers on average, out of an initial pool of 30 classifiers. This befalls without any obvious preference for small ensembles of learners. Even
with
this
small number of learners, considerably lower observed predictive error is realized related to the present state-of-the- art.
84 Instead of relying on heuristics or user-defined parameters, the ensemble exploits MOGA to enable the model to select its optimum result. This approach considers a dataset with dimension of × as input, with each of the rows ∈ demonstrating an autonomously and identically dispersed instance from some universe . Every is built with of values distinctively labeling the features of based on a set of attributes, where each attribute is a set of either ordered (numerical) or unordered (categorical) values. Every has a label that denotes the prediction label that belongs to some output space . The suggested ensemble learner ( ) is trained and tested using to predict the label of hitherto undetected (i.e. test) instances ∈ ~ . The proposed classifier ( ) is a pool of many base learners ( ), whose results for the labels of ∈ are fused (also called "combined") together through
a
voting mechanism as defined in Equation (5.1).˜ = ( ) = (( ); ∀ (5.1) The performance of ( ) can then be calculated as the subset of dataset ∈ whose labels are properly classified as given in Equation (5.2). = 1 | | ∑ (˜ ≠) (5.2)
where (·)

is the Boolean function, providing 1 if the statement̃ ≠ is true, and 0 otherwise. The operations MOGA is given b as follows: 1. Separate into one subsection comprising of the samples for learning, and another subsection comprising the other (1 −) samples for classifier selection, 0 &gt; &gt; 1. 2. Generate bootstrap bags of dimension | | by sampling with replacement.

85 3. Train classifiers for each bag . 4. Select the ideal subsection of the classifiers that identifies the samples in optimally. The proposed MOGA attempts to reduce prediction error and the relationship among the inaccurate votes made by the designated classifiers. If the Pareto front encompasses many data points, choose the point with the least prediction error, breaking draws based on which one has the least number of designated classifiers. 5. The selected subsection of classifiers is the final ensemble. The key notion of the suggested procedure is that a heterogeneous ensemble has more advantages than an identical ensemble. Diversity is added

to

the process in three ways as discussed below. Data diversity: Bagging offers a simple technique for adjusting the learning samples entered into each base-learner. By entering samples that are tested with replacement into each base-learner, the core results and ultimate decisions of the individual learners are varied. Classifier diversity: Through training a various range of classifier types, the MOGA is provided with more chances to determine new correlations and patterns in the records. Classifier "types" denotes diverse structural designs of learners including DT (Saranya et al. 2020), SVM (Li et al. 2012), NB (Farahani 2020), Discriminant Analysis classifiers (DA) (Mika et al. 1999), KNN (Farahani 2020), BF (Akintola et al. 2022), kNN (Freund & Schapire 1995), RF (Farnaaz, & Jabbar 2016) and ANN (Saranya et al. 2020), C4.5(Wathiq et al. 2015), etc. While using many classification algorithms brings new openings for novel detections, it does not assure it, nor is there any assurance that some

86 of the learners will not be identical (i.e. have low diversity). This menace is avoided by the MOGA, wherein one of the two purposes is diversity; if two learners have low diversity, the GA will target to eliminate one of them. Stimulatingly, the MOGA selects some kinds of learners over others, based on the dataset being learned from. Output diversity: the third place that diversity is incorporated into the MOGA is in combining the output of the individual classifiers. After the classifiers have been trained on the bags, the classifiers are pruned down to a reduced set. This reduced set of classifiers is adapted to not only have high accuracy, but also high diversity. This is accomplished by

a

MOGA, where: 1. One objective is to select the set of learners that has the minimum classification errors when using AoP voting to classify the labels of selection data; and 2. The other objective is to select the set of learners where each pair of learners both make classification error as minimum as possible. Formally, the first objective function (,) targets minimize as given in Equation (5.3). (,) = ∑ ((( ) ; ∀ ∈) ≠ (5.3)

where is the set of trained learners, is the selected data samples, and ( ) is the predicted label of learner for data sample . The second objective (,) targets to minimize the double-fault estimate as given in Equation (5.4). (,) = | | × | |(| |−1) 2 × ∑ ∑ ∑ (| | ≠ | | ( ) ≠ ∧ ( ) ≠ (5.4)

87

which at its core is a summation of the number of pairs of the classifier that both compute improper labels, for sample ∈ . Reducing the double-fault amount aligns well with the objective of the diversity of having uncorrelated errors. Now, MOGA will consider converging on a Pareto front of results to fine-tune both objective functions effectively.

The

Pareto front contains all non-dominated results; that is, all results where it is not possible to reduce one objective function without increasing the other. This denotes that excluding getting stuck in local minima (the GA is intended to circumvent this problem using techniques like random mutations), the result with the minimum validation error is assured to be incorporated on the Pareto front, and if several results with equally low validation error were learned, the solutions with the best diversity were selected. The Wilcoxon rank-sum test is performed to determine whether the proposed ensemble approach provides a significant improvement as compared with other approaches or not. The test is carried out using the effects of the proposed ensemble approach and related to each of the other approaches at a level of 5% statistical significance. The

ρ-

values&gt; 0.05 specify that the null hypothesis is precluded, i.e., there is a significant variance at a level of 5% significance. In contrast, the

ρ-

values&lt;0.05 indicate that there is no noteworthy variance between the related values. From the results, it is found that most of the p-values achieved by the proposed ensemble approach are less than 0.05 which proves that the enhancement realized by the proposed ensemble approach is statistically significant. To select appropriate base learners for ensemble,

this work set = 9, with the classifiers including C4.5, ANN, SVM, kNN, DA, BF, DT, NB, and RF. The proposed MOGA is implemented using

a Weka testbed with the
88 selected
base learners. Table 5.1 shows the average number of times each of the 9 types of base classifier is designated by the MOGA, across the 10 trails. The last column presents the total number of base learners designated, and the final row presents the average number of times each learner type was designated across all databases. The first observation is that both the total number of designated base learners and the specific types of designated classifiers depend profoundly on the database. For some databases, like Vehicle, two base learners are required to make a highly precise classification. In the meantime, the Page blocks database needs 4
learners
to realize the same high enactment. The types of base learners for each database vary significantly as well. BF classifiers are the most favored learner on average. In contrast, DA and SVM only appear in small numbers, but they still give a non-zero number of ensembles, representing that even rarely happening classifier types are worth including in the algorithm. Table 5.1 The number of each type of classifier selected by the MOGA Dataset C4.5 ANN SVM kNN DA BF DT NB RF Total Sonar 0.290 0.652 0.592 0.261 0.172 0.732 0.322 0.210 0.682 3.913 Ionosphere 0.552 0.485 0.425 0.365 0.011 0.675 0.355 0.472 0.211 3.551 Balance 0.353 0.152 0.092 0.032 0.028 0.351 0.522 0.273 0.482 2.285 WBC 0.540 0.374 0.314 0.254 0.194 0.566 0.144 0.460 0.504 3.350 Pima Indian 0.410 0.151 0.091 0.031 0.171 0.641 0.421 0.430 0.581 2.927 Vehicle 0.174 0.114 0.054 0.012 0.234 0.614 0.184 0.094 0.564 2.044 Waveform 0.080 0.341 0.281 0.221 0.021 0.521 0.411 0.110 0.221 2.097 Page Blocks 0.403 0.444 0.384 0.324 0.264 0.424 0.414 0.323 0.674 3.654 Average 0.350 0.339 0.279 0.188 0.137 0.566 0.347 0.283 0.490 2.978
89 While some types of classifiers (e.g., BF, RF, and C4.5) are more frequently preferred than others (e.g., DA and SVM), it finally depends on the features of the dataset. Even rarely-preferred learners are still sometimes preferred, and the MOGA can prune away all replicas of a classifier type if they are redundant. Remarkably, only a very small number of base learners are preferred by the optimization algorithm for the final ensemble; out of an initial pool of 30 classifiers, an average of only 2.978 are required to make an EC outperform several state-of-the-art techniques. 5.2.2 Combining Classifiers The second phase exploits the method to fuse the results of the base learners. The base-learner choice and combiners are two important techniques to fuse the results of individual learners (Kuncheva, 2004). Different fusion techniques have been found in the literature as bagging, boosting, stacking, majority voting, weighted majority voting, NB combination, behavior knowledge space method, probabilistic approximation, decision templates, singular value decomposition, Dempster–Shafer combination, and elementary combiners (Kuncheva, 2004). Elementary combiner: Among
the aforementioned fusion techniques, the
most widely employed technique based on algebraic fusion rules is elementary combiners. This technique combines the results of learners that can be presented as posteriori probability. The major advantage of using this technique is its simplicity and it does not need any training. It includes several procedures such as sum, minimum, maximum, product, average, and majority voting rules. These techniques fuse the results of the learners on the measurement level. Let { 1 , 2 ,... } be the set of base learners and { 1 , 2 ,... } be the set of class labels. The fusion technique combines the results of all 90 to yield the final class label for the input →→→ . The results of all can be considered as a posteriori likelihoods. Let input →→→ is finally allotted to label , where is one of the possible labels. The sum, minimum, maximum, product, average, and majority voting techniques
that
can be used to predict are defined as follows: Sum rule: This rule adds the results of all the individual learners for each label and then allocates the label to specified input data with the highest score as shown in Equation (5.5). = =1,2,... ∑ (=1 | →→→) (5.5) Minimum rule: This rule selects the minimum of the scores of individual learners for each label, and then allocates the label to specified input with the maximum score as given in Equation (5.6). = =1,2,... =1,2... (| →→→) (5.6) Maximum rule: This rule selects the maximum of the scores of individual learners for each label, and then allocates the class label to specified input with the maximum score as given in Equation (5.7). . = =1,2,... =1,2... (| →→→) (5.7) Product rule: This rule multiplies the scores of individual learners for each label, and then allocates the label to specified input with the maximum score as given in Equation (5.8). = =1,2,... ∏ (=1 | →→→) (5.8) Majority voting rule: This rule is derived from the sum rule as given in Equation (5.9).
91 = =1,2,... ∑ ∆ =1 (5.9) Average rule: This rule selects the average of the scores of base learners for each label and then allocates the label to particular data input with the maximum score as given in Equation (5.10). = =1,2,... 1 ∑ (=1 | →→→) (5.10)
This study applies the above-mentioned techniques to
integrate the results from base classifiers (discussed in Section 6.6.7). Among these techniques,
a voting mechanism with the AoP rule outperforms all other methods. Hence, a
voting mechanism using AoP is implemented in ICET to increase the effectiveness of the intrusion detection process. 5.3 PROPOSED

ENSEMBLE CLASSIFIER
In EC approaches, several different, unbalanced, and good classifiers are integrated in a specific way. Ensemble classification approaches are prevailing to handle the classification problem
and cooperatively realize results with higher accuracy and reliability by implementing and integrating many autonomous classifiers. The traditional domains for using EC approaches are computational reason, statistical reason, and representational issue. For example, in some cases, there is a problem when the classification is a computationally too intensive and time-consuming process for a single classifier to define an appropriate hypothesis. In some cases, an individual
classifier may cause a feeble result if the input dataset is not enough to train the learning
process. In some other cases, a single classifier is not sufficient to represent the research space.
Boosting and Bagging are the two most well-known approaches in collaborative learning, usually generating better classification solutions and
92 being extensively selected to construct several ensemble frameworks. Besides, the other recognized collaborative learning approaches such as Stacking, Bayesian parameter averaging, and voting
mechanism is
used
for increasing the efficiency of the classification process.
Similarly, ensemble approaches have been used to increase classification accuracy in several applications, including the
identification of intrusive activity. Furthermore, ensemble classifiers deliver tools to investigate the similarity between malicious and genuine samples.
This work focuses on an EC model that combines three different
classifiers,
namely C4.5, RF, and BF to increase the predictability of IDS. These classifiers are employed to implement
a voting mechanism using AoP. 5.3.1 C4.5 Decision Tree
Classifier A classification tree or decision tree is an intuitive model which maps the observations about a parameter to decisions about its target value. It targets to categorize the input data streams into the equivalent class labels based on their attributes. The key concept behind the construction of DTs is known as the ID3 algorithm introduced by Quinlan. ID3 algorithm forms a DT by applying a top-down methodology where a training dataset is used to test the features through a greedy approach. It computes the information gain and entropy to select a particular feature to test at every node in the DT. In this tree configuration, leaf nodes denote labels (or classes), non-leaf nodes represent the possible value of attributes, and branches denote correlation among attributes that bring about the decisions. C4.5 constructs DTs from a learning dataset and develops contextual decision rules to categorize the data streams. It determines the optimal split of the dataset in such a way that to maximize the gain ratio (GR) by searching all the nodes in the DT. The gain ratio is defined as given in Equation (5.11).
93 ( ) = ( ) ( ) (5.11) where ( ) is information gain and ( ) is splitting criteria. In data stream classifiers, a feature set ( ) with the maximum is designated as a splitting feature for the node. Information gain signifies the amount of hesitancy in the dataset is reduced after it is segmented with respect to the designated feature set.
The hesitancy in is estimated by entropy (( )) using Equation (5.12). ( ) = −∑ ( ) 2 ∈ ( ) (5.12)
where denotes the class label in and ( ) is the fraction of the number of instances in label to the number of cases in .
Likewise, ( ) describes how the data are consistently segmented by the feature as defined
in
Equation (5.13). ( ) = −∑ | | | | =1 2 (| | | |) (5.13)
where | | | | represents the score of the h split in . Additionally, the C4.5 can classify both numerical and nominal features and can ignore missing data. C4.5 exploits a divide and conquer technique to form a DT using a given training dataset ( ). This training dataset comprises a set of samples according to the dimension of the database employed for learning. Each data sample contains appropriate features with the class label. It estimates the class frequency for samples in . If all input samples are of the same class, then the node is labeled with that class. Nonetheless, if holds samples of more than one class, the optimal feature is selected by ( ) for dataset segmentation. The training set is divided into k different subsets as { 1 , 2 ,..... } based on designated features. The algorithm is executed
94 recursively for every non-empty splitting. Algorithm 1 gives the pseudocode for the basic C4.5 classifier for constructing a DT. Algorithm 1: C4.5 classifier Input: Training dataset ( ) and attributes Output: A set of DTs 1: if (== ∅) then 2: Display error message 3: end if 4: Select a decision node 5: If all instances from are of the same label then 6: Make as a leaf node with label 7: end if 8: If == ∅ then 9: Make as a leaf node with the general label through the voting process 10: end if 11: Select a feature from with maximum ( ) 12: Label with feature 13: For each of 14: Create a branch from with constraint = 15: Assume is the subset in
with = . 16: If == ∅ 17: Give

label to the leaf node with the general class in 18: else join the node to the DT 19: end if 20: end for

95 5.3.2 Random Forest Classifier

RF classifier has comparatively high prediction accuracy related to other conventional classifiers, and it is more tolerant of noisy data points, which has led to several theoretic and empirical studies focusing on the utilization of these classifiers.

For a database with the size of , an arbitrary parameter = { 1 , 2 ,... } is defined as a set of predictor parameters and is a response

96 parameter, where RF realizes a prediction function ( ) for properly classifying . In the classification procedure, the minimum error denoted by a loss function (, ( ) defines the relevance between ( ) and . This classifier also has the facility to penalize ( ) for resilient deviation from . RF has been intended to handle both regression and classification problems. But, in line with the goal of this work the study has been confined to the classification only. From the prediction viewpoint, (, ( ) is denoted by Equation (5.15). (, ( ) = { 0, = ( ) 1, ℎ (5.15) More precisely, the prediction function ( ) for each can be denoted as the Bayes rule which is given in Equation (5.16). ( ) = ∈ (= | =) (5.16) Equation (5.16) regulates the classification process for an individual DT. Conversely, for number of DTs, the prediction function of RF is given in Equation (5.17). ( ) = ∈ ∑ (= ℎ =1 ( )) (5.17) where is the function of the indicator. It is worth mentioning that the RF depends on the concept of recursive binary splitting. The predictor space employed to be divided on the discrete variable of . Regarding the prediction, the partitioning condition for each non-leaf node is estimated using the Gini index as defined in Equation (5.18). = ∑ˆˆ' = ' (5.18)

97 where, represnts the number of class lables,ˆrepresnts the contributed ratio of class in the nodeˆ. The nodeˆ' can be measured as given in Equation (5.19).ˆ' = 1 ∑ (= =1) (5.19) In the data partitioning process, two child nodes are formed on the right and left sides which are further split like the parent node. The procedure is repeated until the end nodes are achieved (on terminating condition). Every end-node offers an anticipated result. The expected output at the end nodes for each data point is the highest iterating class for that data point. This concept is appropriate to continuous attributes. On the other hand, for categorical features, the solution is measured using voting of the classification result of DTs. 5.3.3 Balanced Forest Classifier A new balanced decision forest known as decision forest by penalizing attributes has been introduced by Adnan & Islam (2017) with a facility for penalizing features during the DT construction procedure. Contrasting some classic classification approaches found in the literature, BF

exploits a subset of the non-class features. This approach forms a group of decision trees with higher accuracy based on the strength of all non-class features existing in a dataset.

The

BF exploits the CART as an efficient decision-maker. Similar to RF algorithm, the BF produces a bootstrap instance from the initial learning sample T. The DTs are constructed upon these instances, where the merit score ( ) of the features determine the optimal split point as defined in Equation (5.20). = × (5.20) here denotes the prediction capability and signifies weights of features. The initial tree is created with the default weight 1. The weight of the features

98 is progressively amplified during the DT construction process. The height of the DT determines the ultimate weight of the features. Hence,

dynamic weight assignment enables the DTs to classify the unlabeled data samples. The classification decisions of the DTs are integrated and the final result about the class label of the data sample is calculated using a voting mechanism. 5.4 SUMMARY The immense technical innovations emerged in the recent years have great influence on all aspects of life. However, this leads to illegal mining of sensitive information. Hence, efficient and dependable detection models are inevitable. At the same time, prevailing research works

have

99 proved that the curse of high dimensional dataset from the unstable network traffic, low accuracy, high FAR, low detection rates, and the problem to achieve adequate pigeonholed observations remain a challenge. Hence, this work developed an all-inclusive and effective feature selection algorithm (i.e., BIOCFS) and an ensemble classifier (i.e., ICET) that proficiently selects few significant attributes and provides a precise and reliable detection

of the mainstream threats within the databases.

100 CHAPTER 6 EXPERIMENTAL DATA AND EVALUATION This chapter provides a summary of comprehensive analysis and systematic performance evaluation of the proposed system in classifying the inbound network traffic into normal or malicious activity. For example, assessing how well it can effectively choose the most significant attributes among thousands of data samples and exploit these few designated attributes to categorize data packets into either normal or malicious accurately. Also, it presents an in-depth performance appraisal of the base classifiers and ICET on each of the two standard datasets (i.e., NSL-KDD and CIC-IDS 2017), the designated attributes, different combination techniques, and state-of-the-art classifiers in terms of performance measures such as classification accuracy, precision, F-measure, the FAR, and the ADR. The time consumption for model building and testing is also considered in this study to assess the real-time performance of the proposed model. Furthermore,

to alleviate the difficulties in data splitting, the k-fold CV portions are fixed to explicit bounds of learning and testing ratios that are not deceptive in the learning phase to validate the performance, generalizability, and
credibility of the proposed IDS. To realize a more precise evaluation of the model's enactment, the 10-fold CV is used. The experiments are conducted on 3.6 GHz with 16GB RAM, Intel Core i7-4790 processor
with Windows 10 operating system. The proposed ICET model is realized and the results are compared with other IDS models using Weka 3.8.3 workbench containing real-world datasets.
101 6.1 INTRODUCTION In this work, a new IDS is developed to identify various cyberattacks with higher accuracy and efficiency. To reduce the dimensionality of the feature space by selecting an appropriate subset, this work adopts an optimized feature selection, called the BIOCFS algorithm.
Contrasting detection or
classification
processes applied in other fields, the disparity between normal and abnormal packet flow has an adversarial effect on classification performance. To address this class imbalance problem,
this study employs an effective ensemble
classification
model with three different classifiers such as balanced forest, random forest, and C4.5 decision tree to increase the rate of attack recognition and reduce the bias and difference between different learning datasets. In this proposed ICET model, outcomes from different
classifiers
are combined using a voting mechanism. The performance of optimized feature selection with an ensemble classification approach is evaluated in the Weka workbench. 6.2 SIMULATION ENVIRONMENT The Weka workbench is publically available software providing a crew of several MLTs. This tool is introduced by the University of Waikato and is disseminated under the conditions of the GNU General Public License for Windows, Linux, and Macintosh. It is accessible online at the website of http://www.cs.waikato.ac.nz/ml/weka/. It is developed for rapidly trying out prevailing data mining approaches on any database. Weka has become one of the most extensively employed tools and contains methods for solving data mining problems including data preprocessing, association rule mining, feature selection, regression, clustering, classification, and visualization. The algorithms can be implemented directly to a dataset or using Java code. This tool is also well appropriate for developing new algorithms (Bouckaert et al.
102 2013). The algorithms accept their input in the form of a single relational table in the attribute-relational file format. The Weka testbed helps the user create, execute, alter, and evaluate algorithms more conveniently. For instance, the user can carry out a trial that implements some methods against a series of datasets and then examines the results to decide if one of the methods is (statistically) better than the other methods. Weka can be employed to implement an approach to a dataset directed for learning more about the data or to implement various training schemes to relate their enactment to select one for implementation. 6.3 BENCHMARK
DATASET TO MODEL TRAFFIC FLOW The choice of datasets for assessing the established IDS models
is becoming a perplexing endeavor. Acquiring a dataset that imitates the real- time networking activities without any sort of alteration or anonymization is
a vital issue that has been continuously explored by numerous investigators (Aldwairi et al. 2018.).
In some applications where the data is allowed to be exchanged or disseminated for open access; it will be rigorously untraceable or severely modified. This will
make many critical data to be lost or no longer
credible. Moreover, the prodigious effort of the data security professionals in defining security measures for real-time communication scenarios and networking systems including data anonymization, encryption, and some data privacy policies has imposed an implausible challenge in constructing datasets for evaluating the proposed IDS models. Thereby leads to substantial challenges in procuring real-time network traffic and databases. However, the past decades have witnessed several fabricated databases to handle these issues. Nearly all of these databases reflect the vital attributes of real-time network traffics (Zhou et al. 2020). According to this current trend and the methodical study of
103 consistency and reliability, this study selects NSL-KDD (Tavallaee et al. 2009) and CIC-IDS2017 (Sharafaldin et al. 2018) datasets to assess the proposed IDS model. The subsequent sections concisely describe the selected databases for the assessment of the proposed system. 6.3.1
NSL-KDD Dataset The traffic traces of NSL-KDD were generated as a refined form of the original dataset
known as KDDCup'99 (Tavallaee et al. 2009). The KDDCup'99 dataset is constructed and maintained by MIT Lincoln Labs by collecting 4 GB of compressed raw (binary) tcpdump data for 7 weeks to model a typical U.S. Air Force LAN with genuine and malevolent user activities. Though
this dataset
is slightly older and there have been few studies pointing out its
flaws (

i.e., the insufficient reflection of existing low signature threat situations) (Tavallaee et al. 2009), it is still considered the most selected
benchmark dataset and utilized by topical research in different field of engineering domains (
Khammassi & Krichen 2017; Luo et al. 2018). Through conducting statistical analysis on this data set Tavallaee et al. (2009) observed two vital issues which extremely distress the enactment of assessed systems, and lead to a very deprived assessment of IDS models. To handle these problems, Tavallaee et al. (2009) offered a new dataset, NSL- KDD, which contains designated data samples of the complete KDD dataset and does not suffer from any of the stated limitations. The new version of the KDD dataset, NSL-KDD is an open-access dataset for scholars on the website
http://nsl.cs.unb.ca/NSL-KDD. The attack instances in the NSL-KDD dataset are pigeonholed into 4 groups according to their probability distributions as follows. • DoS: It is a threat in which the attacker makes computing or storage resources extremely occupied or very busy to receive
104 genuine requests, or rejects
ratified entry of the
users to a system. • Probing: It is an attempt to collect statistics about nodes in a system to change the state of network security. • Remote to Local (R2L) – In this attack,
the vulnerability of a system permits an attacker to gain access locally to
an approved user account without having their account. • User to Root (U2R) - It is an attack where the attacker accesses the network as a legitimate user through an authorized user account (possibly achieved through a dictionary attack, social engineering, or sniffing passwords). The intruder can exploit some weaknesses to realize root access to the system.

The following Table 6.1 gives some examples of the attacks found in the NSL-KDD dataset. Table 6.1 Example of attacks in NSL-KDD Class Example attacks
DoS neptune, smurf, back, teardrop, pod, land Probing ipsweep, satan, portsweep, nmap, R2L warezclient, guess_passwd, warezmaster, multihop, spy, phf, ftp_write, imap U2R buffer_overflow, rootkit, loadmodule, perl,
The attributes in this dataset are categorized into three types (Tavallaee et al. 2009):
105 • Fundamental attributes - It captures all the features that can be retrieved from a TCP/IP connection and most of these attributes cause an inherent latency in the detection process. • Content attributes - Unlike most of the Probing and DoS attacks, the R2L and U2R cyberattacks
do not have any intrusion with regular signatures. This is due to the Probing and DoS threats containing several service links to some node(s) for a moment; but, the U2R and
the R2L cyberattacks are implanted in the payload of the packets, and generally contain only one link. To identify these types of cyberattacks, content attributes are required to seek mistrustful activities in the payload (e.g., number of unsuccessful login attempts). • Traffic features - this group contains attributes that are calculated in terms of a window interval and are classified into two subtypes (Lee et al. 1999): (i) same node attributes which study only the service links in the last 2s that have the same endpoint as the present link and analyze information about
service, the performance of the protocol, and so on; and (ii) same service attributes which study the service links
in the last 2s that have the similar service as the present link. These attributes are known as time-based
attributes.

Conversely, many slow probing attacks scan the nodes (or ports) using a much larger time interval than 2s, for instance, one in every minute. Thus, these cyberattacks do not generate intrusion patterns with a time window of 2s. To address this issue, the same node and same service attributes are re-computed but according to the connection window of 100 connections instead of a time window of 2s. These attributes are known as connection-oriented traffic attributes.
106 The NSL-KDD dataset contains 41 features for differentiating malevolent or normal network traffic as shown in Table 6.2 (Vaiyapuri & Binbusayyis 2020). Table 6.2 Features of
NSL-
KDD dataset S. No Feature Name S. No Feature Name
S. No Feature
Name 1
duration 2
protocol_type 3 service 4 flag 5
logged_in 13
is_host_login 22 is_guest_login 23
r ate Although the NSL-KDD dataset
preserved the valuable and significant features of its original database, it addressed some shortcomings inherited from the KDDCup'99
database such as

preservation of the diversity of selected data points, accumulation of a more sensible volume of records, and the elimination of redundant samples. Predominantly, the key physiognomy of
the
NSL-KDD dataset is that it was collected to upturn the
107 intricacy level of classification. Numerous standard classification models are employed to evaluate the original database. Each data sample is marked with the number of its successful classifications and intricacy levels of classifications (Bala & Nagpal 2019). There are 5 different types of intricacy levels. The number of
selected data points is inversely proportional to
the ratio of samples extant in the database.
Table 6.3 lists the statistics of the threat and normal data samples of the NSL-KDD database. Table 6.3 Statistics of the NSL-KDD dataset Class Training Set Testing Set Normal 67,343 45,000 Attack 80,046 2600 Total Number of Records 125,973 47,600 6.3.2
CIC-IDS 2017 Dataset The CIC-IDS 2017 dataset is designed for IDSs by the Canadian Institute for Cybersecurity to construct a dependable and open-access database on a realistic network for evaluating IDS models with different modern threat scenarios (Panigrahi & Borah 2018). Contrasting the NSL- KDD dataset, this dataset encompasses original samples. CIC-IDS 2017 dataset is intended to tackle the issue of the lack of an up-to-date and reliable dataset for evaluating IDS models. CIC-IDS 2017 dataset provides 5 days of traffic. The first-day traffic encompasses only genuine data, while the 4 subsequent days contain genuine data and 14 types of cyberattacks as given in Table 6.4. The dataset encompasses 3119345 records and 84 attributes comprising 15 class labels (1 normal + 14 attack labels).
108 Table 6.5 lists the wide-ranging feature set of the CIC-IDS 2017 dataset. Table 6.4 Example of attacks in CIC-IDS 2017 dataset Class Example attacks Training Set Testing Set Normal - 536937 453877 DDos DDoS 34880 25597 DoS Heartbleed, DoS slowloris, DoS Slowhttptest, DoS
Hulk, DoS GoldenEye 60765 50317 Web attacks Brute Force, SQL Injection, XSS 4320 423 Brute-force SSH-Patator and FTP- Patator 3538 1177 Botnet ARES Bot 1180 324 Probe Infiltration 36176 31753 Total 682559 565,053
109 Table 6.5 Features of the
CIC-IDS 2017
dataset S. No. Feature Name S. No Feature Name S. No Feature Name S. No Feature Name 1
Flow ID 2
Source
IP 3 Source Port 4 Destination IP 5 Destination Port 6 Protocol 7 Timestamp 8
Bwd PSH Flags 39 Fwd URG Flags 40 Bwd URG Flags 41 Fwd Header Length 42 Bwd Header Length 43
Fwd Packets/s 44 Bwd Packets/s 45
Fwd Header Length 63
Subflow Fwd Packets 70 Subflow Fwd Bytes 71 Subflow Bwd
Packets 72 Subflow Bwd
Max 84
Idle Min
110 6.4
DATA PREPROCESSING It is a critical and time-consuming phase in the data mining process. Real-world data
is
often retrieved from different applications and can be
incomplete, redundant, inconsistent, noisy, and/or missing certain trends or behaviors (
Li et al. 2018).
Therefore, it is essential to convert raw data into an appropriate format suitable for knowledge discovery and evaluation. In this work, we employ preprocessing for eliminating redundant features and outliers (filtration), transformation, and data normalization. 6.4.1 Data Filtration The real-life raw data from
assorted sources contain redundant and abnormal attributes which may have a negative influence on the classifier accuracy.
To handle this issue, redundant records must
be removed from the dataset at the beginning of the experiments. For example, to apply the proposed model in CIC-IDS 2017 dataset, 8
files are concatenated into one same table and all the records that have the attribute asFlow Packets/as equal to 'NaN' or 'Infinity' are removed. Thus, 8 attributes which have the same value for all records, such as Bwd
Avg Bulk Rate, Bwd Avg Packets/Bulk, Bwd Avg Bytes/Bulk, Fwd Avg Bulk Rate,
Fwd Avg Packets/Bulk, Fwd Avg Bytes/Bulk, Bwd
URG Flags, and Bwd PSH Flags are removed. 6.4.2 Data
Normalization NSL-KDD dataset encompasses discrete, continuous, and symbolic values.

For example, the attribute 'protocol type' contains symbolic values like 'icmp'. 'udp', and 'tcp'.
Since most of the classifiers recognize only numerical values, the transformation process is essential and this process has a considerable effect on the performance of intrusion detection. In our work, we assign numerical values for every single symbolic feature. Furthermore,
111 different scales among features can reduce the performance of the classifier, for instance, features with large numeric values. Therefore, normalization is considered a 'scaling down' transformation process. It relates every feature to a standardized range. In this work, we use a simple and fast technique known as a min-max
technique (
Kotsiantis et al. 2006) for normalization. 6.4.3
Creating a Balanced Dataset It can be easily found that the number of attack records is quite low compared to normal records. This makes sense since attacks
do not usually occur as frequently as normal traffic. However, the ratio of anomalous to normal records is a major issue that can significantly affect the experiment training and learning process. To create a balanced dataset
of normal/anomalous records from the
dataset, we need to include an equal number of normal and anomalous records
in both training and testing subsets. At the same time, we make sure that the same record cannot appear in both subsets, which guarantees proper training and leads to better accuracy during
the testing phase. 6.4.4 K-fold Cross-validation
In order to guarantee
reliable and effective solutions, all the fallouts obtained in this study are based on the mean outcomes of all recurrences of
the
k-fold CV method. The proposed classification model assigns =10. In a 10-fold CV, the whole database is divided into 10 parts. For each folding, one part of the database is employed for testing and the other sections are used for training. Then, we compute the average value of results across all ten autonomous trials. The advantage of this method is that all testing samples are self-regulating and the reliability of the results could be increased. It is worth mentioning that only one repetition of the 10-fold CV will not generate acceptable outcomes for assessment due to the uncertainty in data 112 fragmentation. Hence, all the outputs are specified on an average of 10 runs to obtain accurate results. Indeed, a separate set of data samples for testing and validation are not presented in the database. 6.5 PERFORMANCE MEASURES FOR EVALUATION To evaluate the performance of the ICET using a suitable CFS algorithm, this study considers some significant performance metrics including detection
accuracy, precision, recall, F-measure, false alarm rate, and attack detection rate. The methods of estimation of performance measures are derived from (
Elhag et al. 2019). The classification accuracy of the model is estimated as given in Equation (6.1). Accuracy (ACC) = TN+TP TN+TP+FN+FP (6.1)
where TP denotes the number of true positives which signify an instance that is properly categorized as a threat instance; FN defines the number of false negatives that signify an attack instance that is inaccurately pigeonholed as a normal instance. TN denotes the number of true negatives that signify an instance is properly predicted as a normal instance and FP signifies the number of false positives that represents a normal instance is incorrectly categorized as a cyberattack. Precision is the proportion of the number of true positive (TP) instances predicted to be a particular class to the total number of positive instances (TP+FP) predicted as the related class as shown in Equation (6.2). ( ) = + (6.2)
F-measure is a measure of the accuracy of the detection system on a dataset. It is used to evaluate binary classification systems to reflect the
113 compromise between recall and precision. The formula for the standard F- measure is the harmonic mean of the precision and recall. A perfect model has an F-score of 1. F1 – Measure (1) = 2 × + (6.3) The recall is the ratio of the number of properly categorized abnormal packets (TP) to the total number of packets (TP+FN) as shown in Equation (6.4). Recall (REC) = + (6.4) The false alarm rate is also a significant metric to evaluate the performance of the intended approach. It is defined as the ratio of false- positive and the total samples in the dataset as given in Equation (6.5). False alarm rate (FAR) = + (6.5) Attack detection rate represents the ratio of true positive and the total samples identified by the ICET model, where TP and FN are the scores of true positive and false negative as given in Equation (6.6). Attack detection rate (ADR) = + (6.6) Model training and testing time are the time consumption to build and test the proposed models to detect cyber threats on a new dataset, which reduces the consumption of the feature selection process. This work considers these timing overheads to assess the model.
114 6.6 PERFORMANCE EVALUATION OF PROPOSED ICET MODEL
The efficiency of the proposed ICET model is assessed according to its competence in categorizing input samples into an appropriate class. The proposed IDS has been appraised by training and testing subsets of the NSL- KDD
and CIC-IDS 2017 datasets. More precisely, for a

given dataset, we derive the confusion matrix during the testing phase of
the ICET approach, and relate the efficiency of this
approach without implementing any feature selection process and also some related feature selection approaches with respect to performance measures such as
ACC, PRE, REC, F1M, FAR, and ADR.
First, the important features are identified using our BIOCFS algorithm. Then, the representative features are retrieved from the intact dataset for the subsequent phases.
The classifiers used for comparison are given in Table 13. The efficacy of the proposed ICET model is assessed by relating the numerical results with that of 9 similar approaches, including conventional SVM (Li et al. 2012), C.4.5 (Wathiq et al. 2015), RF (Farnaaz, N & Jabbar 2016), BF (Akintola et al. 2022), EIDS (Abdullah et al. 2018), ML-IDS (Umar et al. 2021), ENML (Mahfouz et al. 2020), REPTree (Belouch et al. 2017), KODE (Jaw & Wang 2021). Table 6.6 Classifiers selected for comparison IDS Models Dataset Attribute Selection Base learners FAR (%) ACC (%) DR (%) SVM NSL-KDD Wrapper SVM NA 86.11 87.42 C.4.5 KDD Cup 1999 Hybrid C4.5 NA 86.35 90.24 RF NSL-KDD IG-Filters RF NA 83.62 86.34 BF NSL-KDD BF EIDS NSL-KDD IG-Filters Voting (RF, and PART) 0.01 86.697 NA REPTree NSL-KDD IG-Filters Bagging (REPTree) NA 89.85 NA ML-IDS UNSW-NB15 Wrapper-based
Voting (ANN, 0.28 86.41 97.95
115
with a decision tree SVM, KNN, RF, NB) ENML Game Theory and Cyber Security CICFlowMeter Voting (C4.5, KNN, MLP) 0.04 88.67 NA KODE NSL-KDD and CIC-IDS2017 Hybrid feature selection Voting (K-means, SVM, DBSCAN, EM) 0.09 96.73 96.64 As discussed earlier, this study aims
to present an efficient IDS with maximum classification accuracy and minimum FAR. With this aim,
this work integrates BIOCFS an ensemble classifier to find an optimal subset. The integrated model, ICET is trained and tested on NSL-KDD and CIC-IDS 2017 traffic traces. Experimental results illustrate that the ICET outdoes every individual and ensemble classifiers by realizing higher classification efficiency. The subsequent sections highlight the performance in detail. 6.6.1 Performance of ICET without using BIOCFS on the NSL-KDD Dataset This section rigorously assesses the enactment of the intended classification algorithm using the entire attributes of the two databases and the designated attributes obtained from the feature selection algorithm (BIOCFS). Firstly, the proposed EC approach using BF, RF, and C4.5 classifiers is applied to two datasets NSL-KDD and CIC-IDS 2017 without using any feature selection methods. Hence, all the features of the datasets (i.e 41 features in the NSL-KDD dataset and 84 features in the CIC-IDS 2017 dataset) are considered to detect and classify the inbound data samples. The confusion matrices presented in Figures 6.1 (a) and (b) show the effectiveness of the intended attribute selection and ensemble approach on a 10-fold CV. The confusion matrices display the fallouts gained from the ICET model. It is observed that the ICET approach provides adequate performance. Nonetheless, some cyberattacks cannot be detected successfully, for instance, the attacks 'U2R' in NSL-KDD and HTTPDoS in CIC-IDS 2017. Besides, the ICET model does
not consider
the samples of a specific label; it is
116 anticipated for classifying attributes from the complete database, which could not guarantee the performance of each class of
attack detection. On the other hand, since the fallouts gained for normal samples are very good; this model can be used to identify and classify cyber threats. (
a) (b) Figure 6.1 Confusion matrix (a) for NSL-KDD (b) for CIC IDS-2017 Table 6.7 present the results obtained by the individual classifiers (i.e., SVM, C.4.5, RF, and BF), some recent state-of-the-art ensemble classifiers (i.e., EIDS, REPTree, ML-IDS, ENML, and KODE), and the proposed EC classifier on the test dataset NSL-KDD. The results are based on the mean and standard deviation (SD) values of the 10-fold CV
approach on
the NSL-KDD database. The test results reveal that the proposed EC cannot achieve as anticipated results on the original database attributes. It provides only nominal performance with a classification accuracy of 93.50%, precision of 90.28%, recall of 94.87%, F1-measure of 98.15%, and detection rate of 92.65%. Moreover, the proposed ensemble classifier consumes the highest training time of 439.43s with a massive FAR of 0.016%, while the RF takes only 10.65s
for learning
but a colossal 562.15s of testing time. The reason behind the poor performance of the proposed EC
117 classifier is the higher dimensional feature space in the dataset since the classifier exploits the entire feature space.

118 Table 6.7 Results obtained by ICET without using BIOCFS on the NSL-KDD dataset Algorithm ACC (%) PRE (%) REC (%) F1M (%) FAR (%) ADR (%) ρ- value Training (s) Testing (s) SVM 83.29 77.42 82.25 98.10 0.143 80.04 0.050 128.4 1656.6 C.4.5 86.83 78.98 86.97 95.02 0.120 82.85 0.075 11.56 4.19 RF 85.87 80.78 89.99 95.95 0.078 78.21 0.052 10.65 62.15 BF 86.98 81.36 91.42 96.17 0.072 81.37 0.032 35.57 0.23 EIDS 87.33 82.48 90.99 96.35 0.018 83.08 0.033 235.2 7.8 ML-IDS 91.79 83.95 92.15 97.29 0.030 90.15 0.031 12.4 531.6 ENML 92.56 84.98 94.74 97.57 0.055 90.85 0.043 65.4 671.4 REPTree 89.06 85.80 92.11 94.18 0.029 90.79 0.044 127.8 208.8 KODE 92.96 85.59 93.38 98.41 0.014 92.24 0.039 441.6 2140.2 ICET 93.50 90.28 94.87 99.80 0.016 93.65 0.020 139.43 12.25 Interestingly, SVM records the nominal performance regarding classification accuracy (83.29%), precision (77.42%), recall (82.25%), F1- measure (98.10%), and attack detection rate (80.04%). It consumes only 128.4s as model building time. On the other hand, it consumes a colossal 1656.6s for testing and a massive FAR of 0.143 among the individual classifiers. Therefore, SVM-based classifiers are not appropriate for large datasets. Also, it does not execute very well when the dataset has more noise i.e. target classes are overlapping. In cases where the number of attributes for each data point surpasses the number of learning instances, the SVM will underperform. The decision tree-based classifier C4.5 outperforms SVM in terms of performance metrics with 86.83% accuracy, 78.98% of precision, 86.97% of recall, 95.02% of F1-measure, and 82.85% of

detection rate. It will take

119 only 11.56s for model building and 4.19s for testing with a false alarm rate of 0.120. Hence, C4.5 shows the best results as compared to other algorithms in terms of timing complexity of training and testing processes by permitting numerical features, allowing missing values imputation, and performing tree pruning. Figure 6.2 Performance of ICET without using BIOCFS in terms of ACC, PRE, REC, FIM, and ADR The other two decision forest-based classifiers RF and BF yield similar results as C4.5. The random forest records the decent performance

in terms of classification accuracy (85.87%), precision (80.78%), recall (89.99%), F1-measure (95.95%), and attack detection rate (78.21%).

Also, it shows better enactment in terms of other measures, such as an extremely reduced training and testing time of 10.65s and 62.15s, correspondingly. However, it provides a vast FAR of 0.078 since RF is still lacking in false positive rate control and it constructs several trees to integrate their results.

120 Figure 6.3 Performance of ICET without using BIOCFS in terms of FAR and

ρ-

value The classifier using balanced forest achieves 86.98%

accuracy, 81.36% of precision, 91.42% of recall, 96.17% of F1-measure, and 81.37% of

detection rate. It will take only 35.57s for model building and 0.23s for testing with a nominal false alarm rate of 0.072. Since BF exploits randomly assigned weights, often a good feature does not acquire a low weight that its merit value will be lesser than that of a feature with relatively inferior classification ability. This study also attempts to compare the proposed classifier with other ensemble classifiers to analyze the effectiveness of the optimized CFS algorithm on the NSL-KDD dataset with original features. The EIDS classification model provides 87.33% of classification performance, 82.48% of precision, 90.99% of recall, 96.35% of F1-measure, and 83.08% of attack detection rate. It records 235.2s model building time and 7.8s testing time with a reduced false alarm rate of 0.018.

121 Figure 6.4 Performance of ICET without using BIOCFS in terms of training and testing time Another ensemble classifier ENML using C4.5, KNN, and MLP as the base learners achieve higher performance as compared to the EIDS classification model with classification accuracy (92.56%), precision (84.98%), recall (94.74%), F1-measure (97.57%), and attack detection rate (90.85%). This classifier takes 65.4s and 671.4s for model building and testing, correspondingly with the false alarm rate of 0.055. The ensemble classifier REPTree achieves a lower FAR (0.029) with 89.06% of classification accuracy, 85.80% of precision, 92.11% of recall, 94.18% of F1- measure, and 90.79% of

attack detection rate. The time taken by the classifier for training and testing the model is 127.8s and 208.8s, correspondingly. KODE provides better results for the original attributes of the dataset with respect to performance metrics, including predictive accuracy (92.56%), precision (85.59%), recall (93.38%), F1-measure (99.41%), attack detection rate (93.24%) and the lowermost FAR of only 0.014. However, it exhibits poor enactment in terms of other measures, like an increased model

122 training time (441.6s) and (2140.2s). The proposed ensemble classifier outperforms all other individual and ensemble learners in terms of 93.50% of classification

accuracy, 90.28% of precision, 94.87% of recall, 98.15% of F1- measure, and 92.65% of

attack detection rate. More importantly, it provides only a 0.016 false alarm rate. However, it takes colossal time for training and testing such as 139.43s and 12.25s, respectively. The Wilcoxon rank-sum test is performed to determine whether the proposed ensemble approach provides a significant improvement as compared with other approaches or not. The test is carried out using the effects of the proposed ensemble approach and related to each of the other approaches at a level of 5% statistical significance. Tables 6.7 describes the

ρ-values obtained by the rank-sum test, where the ρ-values&gt; 0.05 specify that the null hypothesis is precluded, i.e., there is a significant variance at a level of 5% significance. In contrast, the

ρ-

values&lt;0.05 indicate that there is no noteworthy variance between the related values. From the results, it is found that most of the p-values achieved by the proposed ensemble approach are less than 0.05 which proves that the enhancement realized by the proposed ensemble approach is statistically significant. Table 6.8 lists the obtained results of the proposed ensemble classifier for NSL-KDD in terms of the standard deviation of performance measures without using any feature selection algorithms. It is stimulating to observe that the SD gained by the proposed EC is lesser than that of all most all other classifiers which prove that the proposed EC can provide more robust and dependable results. 123 Table 6.8 Performance of ICET without using BIOCFS in terms of SD values Algorithm ACC PRE REC F1M FAR ADR ρ-value SVM 0.049 0.011 0.041 0.011 0.041 0.021 0.015 C.4.5 0.028 0.013 0.046 0.013 0.046 0.026 0.031 RF 0.014 0.008 0.033 0.008 0.033 0.041 0.015 BF 0.048 0.010 0.019 0.010 0.019 0.033 0.021 EIDS 0.026 0.011 0.018 0.011 0.018 0.031 0.014 ML-IDS 0.018 0.006 0.029 0.006 0.029 0.029 0.017 ENML 0.018 0.004 0.015 0.004 0.015 0.026 0.016 REPTree 0.019 0.021 0.015 0.021 0.015 0.023 0.018 KODE 0.019 0.005 0.019 0.005 0.019 0.021 0.020 ICET 0.011 0.008 0.009 0.008 0.009 0.017 0.008 Figure 6.5 Performance of ICET without using BIOCFS in terms of SD values

124 Tables 6.9 to Table 6.15 display the experimental outcomes results of all the methods including ICET for different folding on the NSL-KDD dataset. The mean and SD values obtained by each classifier are quantified in each table and the optimal numerical values are highlighted in bold. It is observed that the enactment metrics gained by the proposed ICET approach are superior to all other methods. It is worth noting that the proposed ICET exhibits better results as related to other individual and ensemble classifiers in the majority of the scenarios. This demonstrates that the integration of the proposed ICET has considerably increased the performance of the classification model. Table 6.9 Accuracy of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.787 0.877 0.852 0.801 0.870 0.898 0.894 0.893 0.878 0.918 #2 0.738 0.872 0.861 0.798 0.877 0.891 0.937 0.894 0.939 0.919 #3 0.779 0.849 0.865 0.906 0.857 0.927 0.928 0.891 0.942 0.925 #4 0.845 0.823 0.842 0.906 0.880 0.895 0.941 0.863 0.939 0.948 #5 0.857 0.838 0.863 0.872 0.899 0.944 0.935 0.871 0.922 0.938 #6 0.853 0.857 0.866 0.931 0.915 0.935 0.890 0.868 0.938 0.939 #7 0.885 0.866 0.852 0.929 0.896 0.920 0.942 0.919 0.933 0.945 #8 0.839 0.903 0.869 0.843 0.842 0.922 0.927 0.913 0.938 0.939 #9 0.854 0.890 0.834 0.862 0.834 0.919 0.928 0.903 0.925 0.937 #10 0.892 0.908 0.883 0.850 0.863 0.928 0.934 0.891 0.942 0.942 Mean 0.833 0.868 0.859 0.870 0.873 0.918 0.926 0.891 0.930 0.935 S.D 0.049 0.028 0.014 0.048 0.026 0.018 0.018 0.019 0.019 0.011

125 Table 6.10 Precision of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.672 0.737 0.781 0.829 0.843 0.837 0.839 0.838 0.818 0.897 #2 0.737 0.758 0.806 0.742 0.837 0.760 0.858 0.792 0.832 0.878 #3 0.792 0.770 0.809 0.838 0.805 0.878 0.822 0.847 0.837 0.839 #4 0.794 0.772 0.726 0.856 0.826 0.875 0.819 0.858 0.828 0.924 #5 0.767 0.760 0.812 0.726 0.859 0.814 0.827 0.877 0.857 0.920 #6 0.788 0.783 0.820 0.814 0.860 0.807 0.904 0.887 0.877 0.921 #7 0.794 0.790 0.810 0.806 0.849 0.835 0.835 0.876 0.876 0.931 #8 0.782 0.859 0.822 0.847 0.792 0.871 0.844 0.881 0.881 0.907 #9 0.814 0.871 0.812 0.844 0.784 0.851 0.880 0.868 0.878 0.891 #10 0.802 0.798 0.880 0.834 0.793 0.867 0.870 0.857 0.876 0.922 Mean 0.774 0.790 0.808 0.814 0.825 0.840 0.850 0.858 0.856 0.903 S.D 0.042 0.043 0.038 0.045 0.029 0.038 0.028 0.028 0.025 0.028 Table 6.11: Recall of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.840 0.898 0.855 0.870 0.895 0.903 0.933 0.923 0.830 0.940 #2 0.833 0.847 0.892 0.902 0.920 0.895 0.946 0.905 0.971 0.955 #3 0.818 0.815 0.878 0.938 0.906 0.893 0.957 0.936 0.953 0.974 #4 0.863 0.934 0.890 0.900 0.944 0.954 0.950 0.940 0.954 0.959 #5 0.850 0.916 0.899 0.909 0.933 0.970 0.954 0.924 0.957 0.951 #6 0.810 0.882 0.888 0.903 0.910 0.893 0.919 0.931 0.912 0.964 #7 0.867 0.872 0.980 0.980 0.895 0.919 0.959 0.920 0.927 0.910 #8 0.837 0.906 0.909 0.921 0.903 0.925 0.956 0.931 0.953 0.959 #9 0.762 0.798 0.919 0.915 0.896 0.904 0.967 0.908 0.947 0.964 #10 0.745 0.829 0.889 0.904 0.897 0.959 0.933 0.893 0.934 0.911 Mean 23 0.870 0.900 0.914 0.910 0.922 0.947 0.921 0.934 0.949 S.D 0.041 0.046 0.033 0.019 0.018 0.029 0.015 0.015 0.019 0.009

126 Table 6.12 F1-measure of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.980 0.946 0.964 0.970 0.976 0.967 0.968 0.957 0.996 0.983 #2 0.998 0.933 0.951 0.953 0.970 0.979 0.975 0.949 0.998 0.970 #3 0.991 0.972 0.950 0.973 0.971 0.973 0.979 0.950 0.997 0.969 #4 0.994 0.931 0.949 0.941 0.946 0.970 0.980 0.950 0.997 0.988 #5 0.981 0.962 0.953 0.963 0.958 0.980 0.977 0.956 0.995 0.972 #6 0.978 0.956 0.974 0.962 0.951 0.978 0.977 0.952 0.989 0.993 #7 0.980 0.952 0.960 0.969 0.966 0.978 0.975 0.952 0.984 0.989 #8 0.973 0.959 0.967 0.955 0.961 0.977 0.975 0.952 0.981 0.986 #9 0.971 0.945 0.963 0.963 0.959 0.967 0.981 0.903 0.986 0.992 #10 0.964 0.946 0.964 0.968 0.980 0.961 0.974 0.901 0.998 0.993 Mean 0.981 0.950 0.960 0.962 0.963 0.973 0.976 0.942 0.984 0.992 S.D 0.011 0.013 0.008 0.010 0.011 0.006 0.004 0.021 0.005 0.008 Table 6.13 Attack detection rate of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.797 0.851 0.703 0.838 0.810 0.938 0.873 0.923 0.929 0.913 #2 0.785 0.838 0.792 0.819 0.870 0.927 0.891 0.913 0.917 0.891 #3 0.774 0.837 0.847 0.789 0.827 0.904 0.919 0.922 0.906 0.920 #4 0.772 0.838 0.792 0.836 0.816 0.928 0.947 0.920 0.904 0.948 #5 0.826 0.870 0.779 0.813 0.868 0.871 0.922 0.904 0.958 0.942 #6 0.787 0.801 0.783 0.788 0.801 0.910 0.943 0.915 0.919 0.943 #7 0.827 0.807 0.833 0.833 0.811 0.849 0.893 0.875 0.959 0.923 #8 0.814 0.782 0.743 0.761 0.806 0.887 0.920 0.872 0.945 0.946 #9 0.802 0.820 0.762 0.789 0.842 0.883 0.898 0.890 0.934 0.928 #10 0.819 0.843 0.788 0.873 0.859 0.920 0.875 0.947 0.951 0.935 Mean 0.800 0.829 0.782 0.814 0.831 0.902 0.909 0.908 0.932 0.937 S.D 0.021 0.026 0.041 0.033 0.031 0.029 0.026 0.023 0.021 0.017

127 Table 6.14 FAR of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.160 0.148 0.033 0.028 0.003 0.011 0.041 0.031 0.038 0.018 #2 0.153 0.097 0.070 0.060 0.028 0.003 0.054 0.013 0.179 0.013 #3 0.138 0.065 0.056 0.096 0.014 0.001 0.065 0.044 0.131 0.012 #4 0.183 0.184 0.068 0.058 0.052 0.062 0.058 0.048 0.062 0.017 #5 0.170 0.166 0.077 0.067 0.041 0.078 0.062 0.032 0.195 0.019 #6 0.130 0.132 0.066 0.061 0.018 0.001 0.027 0.039 0.120 0.012 #7 0.187 0.122 0.158 0.138 0.003 0.027 0.067 0.028 0.095 0.018 #8 0.157 0.156 0.087 0.079 0.011 0.033 0.064 0.039 0.091 0.017 #9 0.082 0.048 0.097 0.073 0.004 0.012 0.075 0.016 0.065 0.012 #10 0.065 0.079 0.067 0.062 0.005 0.067 0.041 0.001 0.062 0.019 Mean 0.143 0.120 0.078 0.072 0.018 0.030 0.055 0.029 0.104 0.016 S.D 0.041 0.046 0.033 0.019 0.018 0.029 0.015 0.015 0.019 0.009 Table 6.15

ρ-

value

of the proposed model Vs other approaches Fold SVM C.4.5 RF BF EIDS ML- IDS ENML REP Tree KODE ICET #1 0.035 0.071 0.055 0.011 0.012 0.045 0.025 0.049 0.032 0.026 #2 0.054 0.048 0.052 0.033 0.020 0.051 0.035 0.064 0.051 0.019 #3 0.046 0.043 0.047 0.017 0.024 0.015 0.045 0.033 0.031 0.028 #4 0.056 0.068 0.047 0.025 0.017 0.010 0.056 0.034 0.042 0.029 #5 0.047 0.071 0.037 0.050 0.039 0.028 0.071 0.069 0.057 0.024 #6 0.044 0.133 0.057 0.023 0.051 0.028 0.039 0.057 0.065 0.022 #7 0.051 0.114 0.048 0.074 0.045 0.001 0.018 0.012 0.014 0.020 #8 0.080 0.092 0.091 0.022 0.025 0.048 0.058 0.048 0.064 0.003 #9 0.063 0.079 0.053 0.010 0.044 0.044 0.033 0.047 0.019 0.019 #10 0.025 0.035 0.036 0.052 0.048 0.035 0.049 0.025 0.015 0.010 Mean 0.050 0.075 0.052 0.032 0.033 0.031 0.043 0.044 0.039 0.020 S.D 0.015 0.031 0.015 0.021 0.014 0.017 0.016 0.018 0.020 0.008

128 6.6.2

Performance of ICET Model without using BIOCFS on CIC- IDC 2017 Dataset Similar results can be obtained when the proposed EC approach is applied

to

CIC-IDS 2017 dataset. The experimental results obtained from CIC-IDS 2017 dataset using various individual and ensemble classifiers are reported in Table 6.16. The mean value of performance measures and SD gained from this dataset by each approach is illustrated in Figures 6.5 and Figure 6.6. From this table we can perceive that the conventional SVM approach has achieved nominal classification performance with 82.4% accuracy, 78.37% precision, 81.40% recall, 96.15% F1-measure, and 79.12% attack detection rate. Also, it takes 232s for model building and a vast 4880s for testing due to the very high dimension of the dataset. Besides, the false alarm rate related to this model is also high (0.138). By eliminating the overfitting problem, the C4.5 classifier achieves better results related to the SVM classifier regarding classification performance

in terms of accuracy (85.58%), precision (79.93%), recall (86.12%), F1-measure (93.07%), false alarm rate (0.115), and attack detection rate (81.93%).

The time consumption for building and testing the model is also very low as 17s and 5s, respectively. Table 6.16 Results obtained by ICET without using BIOCFS on the CIC-IDS dataset Algorithm ACC PRE REC F1M FAR ADR ρ- value Training time (s) Testing time (s) SVM 82.04 78.37 81.40 96.15 0.138 79.12 0.025 232 4880 C.4.5 85.58 79.93 86.12 93.07 0.115 81.93 0.050 17 5 RF 84.62 81.73 89.14 94.00 0.073 77.28 0.027 14 64 BF 85.73 82.31 90.57 94.22 0.067 80.45 0.007 42 4 EIDS 86.08 83.43 90.14 94.40 0.013 82.16 0.008 276 11 ML-IDS 90.54 84.90 91.30 95.34 0.025 89.23 0.006 16 612 ENML 91.31 85.93 93.89 95.62 0.050 89.93 0.018 71 692 REPTree 87.81 86.75 91.26 92.23 0.024 89.87 0.019 128 214 KODE 91.71 86.54 92.53 97.46 0.009 92.32 0.014 321 4958 ICET 92.25 91.23 94.02 96.20 0.011 91.73 0.006 143 14

129 The RF classification model provides better performance

in terms of accuracy, precision, recall, F1-measure, false alarm rate, and attack detection rate

with 84.62%, 81.73%, 89.14%, 94%, 0.073%, and 77.28%. This model requires more time to test the model (64s) as compared to training the model (14s) since it combines numerous decision trees to determine the appropriate class. By systematically imposing weights, the BF approach achieves higher performance with 85.73% accuracy, 82.31% precision, 90.57% recall, 94.22% F1-measure, 0.067% false alarm rate, 80.45% attack detection rate. However, it provides an increased false alarm rate (0.067). This model requires less time to model (42s) and test the model (4s). As compared with the individual classifiers, EIDS achieves increased performance measures as 86.08%, 83.43%, 90.14%, 94.40%, 0.032, 82.16% with respect to
accuracy, precision, recall, F1-measure, false alarm rate, and attack detection rate,
correspondingly. Figure 6.6 Performance of ICET without using BIOCFS in terms of ACC, PRE, REC, FIM, and ADR on the CIC-IDS 2017 dataset
130 Another ensemble classifier ENML using C4.5, KNN, and MLP as the base learners achieve higher performance as compared to the EIDS classification model with
classification
accuracy (91.31%), precision (85.93%), recall (93.89%), F1-measure (95.62%), false alarm rate (0.050), and attack detection rate (90.85%).
This classifier takes 71s and 692s for building the model and testing, correspondingly. The ensemble classifier REPTree achieves a lower false alarm rate (0.024) with 87.81% of classification
accuracy, 86.75% of precision, 91.26% of recall, 92.23% of F1- measure, and 89.87% of
attack detection rate. The time taken by the classifier for training and testing the model are 128s and 214s, correspondingly. Figure 6.7 Performance of ICET without using BIOCFS in terms of FAR and
ρ-
value on the CIC-IDS 2017 dataset KODE provides better results for the original features in terms of evaluation measures, such as accuracy (91.71%), precision (86.54%), recall (92.53%), F1-measure (97.46%), attack detection rate (92.32%), and the lowermost FAR of 0.009. However, it exhibits poor enactment for the other evaluation measures, such as increased training and testing time of 321s and 4958s, respectively. The proposed ensemble classifier outperforms all other
131 individual and ensemble learners in terms of 92.25% of classification
accuracy, 91.23% of precision, 94.02% of recall, 96.20% of F1-measure, and 91.73% of
attack detection rate. More importantly, it provides only a 0.011 false alarm rate. However, it takes colossal time for training and testing such as 143s and 14s, respectively. Figure 6.8 Performance of ICET without using BIOCFS in terms of training and testing time on the CIC-IDS 2017 dataset Table 6.17 Results obtained by ICET without using BIOCFS on the CIC-IDS dataset in terms of SD values Algorithm ACC PRE REC F1M FAR ADR ρ-value SVM 0.026 0.016 0.026 0.047 0.081 0.021 0.020 C.4.5 0.025 0.013 0.031 0.048 0.064 0.026 0.036 RF 0.040 0.015 0.046 0.043 0.068 0.041 0.020 BF 0.053 0.015 0.038 0.050 0.024 0.033 0.026 EIDS 0.029 0.012 0.036 0.034 0.023 0.031 0.019 ML-IDS 0.034 0.017 0.041 0.039 0.028 0.029 0.024 ENML 0.023 0.009 0.039 0.033 0.029 0.026 0.021 REPTree 0.028 0.014 0.044 0.038 0.034 0.023 0.026 KODE 0.033 0.019 0.049 0.043 0.039 0.021 0.031 ICET 0.016 0.008 0.031 0.033 0.014 0.017 0.013
132 Table 6.17 lists the obtained results of the proposed ensemble classifier for NSL-KDD in terms of the standard deviation of performance measures without using any feature selection algorithms. Figure 6.9 Performance of ICET without using BIOCFS in terms of SD values on the CIC-IDS 2017 dataset It is stimulating to observe that the SD gained by the proposed EC is lesser than that of all most all other classifiers which prove that the proposed EC can provide more robust and dependable results. From these results, it is concluded that the proposed EC achieves better results when applied toNSL-KDD as compared to CIC-IDS 2017 due to the complexity of the original features of the dataset. Since the CIC-IDS 2017 dataset contains high dimensional data samples (3119345 samples) as compared to NSL-KDD (125973 samples). Hence an appropriate feature selection algorithm is required to increase the classification performance with its objective of severely curtailing the FAR, learning time, and testing time, thus outdoing the individual classification algorithms as anticipated.
133 6.6.3 Performance of IDS with BIOCFS Algorithm on NSL-KDD Dataset This work implements BIOCFS with the proposed ensemble approach to demonstrate the ability to efficiently select significant attributes for classifying the network traffic into genuine or malicious ones. The proposed BIOCFS selects 7 significant features from the NSL KDD dataset and 12 features from the CIC-IDS 2017 dataset for improving the performance of the classification model. The selected features in both datasets are given in Table 6.18.
It is easily observed that the size of the dataset is decreased considerably by applying the proposed BIOCFS approach. Finally, in order to classify the
network traffic, the ICET model is used with a voting mechanism. Table 6.18 Selected attributes of the NSL-KDD and CIC-IDS 2017 databases NSL-KDD CIC-IDS 2017 Attribute number Attribute Name Attribute number Attribute Name 4 flag 12 Total Length of Bwd Packets 5 src_bytes 14 Fwd Packet Length Min 6 dst_bytes 18 Bwd Packet Length Min 15 su_atempted 20 Bwd Packet Length Std 17 num_file_creations 22 Flow Packets/s 26 srv_serror_rate 25 Flow IAT Max 30 diff_srv_rate 39 dest_host_srv_serror_

rate 44

Bwd Packets/s 45 Min Packet Length 58 Down/Up Ratio 70 Subflow Fwd Bytes 83 Idle Max

134 The proposed classifier is implemented with the BIOCFS algorithm and the comprehensive outputs obtained by this model regarding performance measures are listed in Table 6.19. From this table, we can observe that the SVM classifier has achieved nominal classification enactment with 88.65% ACC, 84.78% PRE, 87.61% REC, 99.46% F1M, and 86.63% ADR with 0.13% FAR and 0.031

ρ-

value. It consumes 32s for model building and a massive 1657s for testing since its time convergence characteristic is poor. The RF classification model is generated reasonable outputs related to the SVM classifier. The performance of RF is better than SVM in terms of ACC (91.23%), PRE (88.14%), REC (95.35%), F1M (97.31%), and ADR (84.80%) with FAR (0.07) and

ρ-

value (0.033). It consumes only 3s for model building and 62s for testing due to the capacity of RF in constructing a number of powerful decision trees quickly. Table 6.19 Results obtained by ICET using BIOCFS on NSL-KDD dataset Algorithm ACC PRE REC F1M FAR ADR

ρ-

value Training time (s) Testing time (s) SVM 88.65 84.78 87.61 99.46 0.13 86.63 0.031 32 1657 C.4.5 92.19 86.34 92.33 96.38 0.11 89.44 0.056 2 4 RF 91.23 88.14 95.35 97.31 0.07 84.80 0.033 3 62 BF 92.34 88.72 96.78 97.52 0.06 87.96 0.013 27 2 EIDS 92.69 89.84 96.35 97.70 0.01 89.67 0.014 152 8 ML-IDS 97.15 91.31 97.51 98.64 0.02 96.74 0.012 114 532 ENML 97.92 93.34 98.10 98.93 0.05 97.44 0.024 73 671 REPTree 94.42 95.16 97.47 95.53 0.02 97.38 0.025 88 209 KODE 98.32 92.95 98.74 97.06 0.00 99.83 0.020 1212 2150 ICET 99.85 98.64 98.23 99.51 0.01 99.24 0.001 19 12

135 Our empirical results demonstrate that the C4.5 classifier realized substantial classification performance related to the other classifiers like SVM and RF in classifying normal and threat activities. The C4.5 classifier provides better classification enactment with 92.19% ACC, 86.34% PRE, 92.33% REC, 96.38% F1M, and 89.44% ADR with 0.11% FAR and 0.056

ρ-

value without increasing the processing overhead. It consumes only 2s for model building and a massive 4s for testing since its time convergence characteristic is better than SVM and RF. By applying the concept of FPA, the balanced forest achieves 92.34% ACC, 88.72% PRE, 96.78% REC, 97.52% F1M, and 87.96% ADR with 0.06% FAR and 0.013

ρ-

value. The time required for training and testing is 27s and 2s, respectively. The FPA used in the BF can improve the prediction stability of the classification process. The ensemble classifiers EIDS and REPTree provide almost similar results. But, the time complexity of REPTree is very high due to its complex nature. Figure 6.10 Performance of ICET with BIOCFS in terms of ACC, PRE, REC, F1M, and ADR on the NSL-KDD dataset

136 The ML-IDS model exploits the different efficient MLTs with majority voting as a combiner. This model yields improved performance with

accuracy, precision, recall, F1-measure, false alarm rate, and attack detection rate

with 97.15%, 91.31%, 97.51%, 98.64%, 0.073%, and 96.74%. This model also achieves 0.02% FAR and 0.012

ρ-

value. This model requires more time to test the model (532s) as compared to training the model (114s) since it combines numerous decision trees to determine the appropriate class. The ENML model achieves improved solutions regarding ACC (97.92%), PRE (93.34%), REC (98.10%), F1M (98.93%), and ADR (97.44%) with FAR (0.05) and

ρ-

value (0.024). It consumes 73s for model building and 671s for testing due to the capacity of constructing a number of powerful decision trees quickly. Figure 6.11 Performance of ICET with BIOCFS in terms of FAR and

ρ-

value on the NSL-KDD dataset The ensemble classifier, KODE exploits a hybrid feature selection algorithm to assimilate lower-level attributes and efficiently combine them with multi-level attributes through a biased loss function. Regarding ACC, PRE, REC, F1M, ADR, FAR, and

ρ-

value, KODE provides 98.32%, 92.95%,

137 98.74%, 97.06%, 99.83%, 0.01%, and 0.02. But the time complexity related to this model is very high as compared to all other techniques discussed in this study. Figure 6.12 Performance of ICET with BIOCFS in terms of training and testing time on the NSL-KDD dataset The enactment of the intended ensemble classifier, ICET is greater than the classification performance of each base learner used in ensemble systems such as SVM, C4.5, RF, and BF. Also, the ICET can overcome the problems of base learners and ultimately realize greater accuracy than individual classifiers. In a multi-label classification of intrusion, the ICET model remarkably outperforms other classifiers in terms of all the performance measures. The testing accuracy of ensemble learners is generally greater than the testing accuracy of each base learner. As an average of 10- fold trials, the ICET classifier achieved better results related to other ensemble classifiers. The proposed ICET with BIOCFS classifier reveals a profound enhancement in the classification of intrusion into corresponding classes on the NSL-KDD dataset. It achieves much higher classification performance with an ACC of 99.85%, PRE of 98.64%, REC of 98.23%, F1M of 99.51% and ADR of 99.24%, FAR of 0.01%, and ρ-value 0.001. It takes
138 73s for training and 671s for testing the data samples. Figures 6.10 – 6.12 illustrates the performance of ICET with BIOCFS on the NSL-KDD dataset. This empirical analysis demonstrates the capability of the proposed ICET with the BIOCFS classifier to reliably categorize network traffic properly. Table 6.20 Results obtained by ICET using BIOCFS on the NSL-KDD dataset in terms of SD value Algorithm ACC PRE REC F1M FAR ADR ρ-value SVM 0.024 0.014 0.024 0.045 0.080 0.019 0.019 C.4.5 0.023 0.012 0.030 0.047 0.063 0.025 0.035 RF 0.038 0.013 0.045 0.042 0.067 0.040 0.019 BF 0.052 0.013 0.036 0.048 0.023 0.031 0.025 EIDS 0.027 0.011 0.035 0.033 0.022 0.030 0.018 ML-IDS 0.032 0.016 0.040 0.038 0.027 0.027 0.023 ENML 0.022 0.008 0.037 0.031 0.028 0.025 0.020 REPTree 0.027 0.013 0.042 0.036 0.033 0.022 0.025 KODE 0.032 0.018 0.047 0.041 0.038 0.019 0.030 ICET 0.014 0.006 0.030 0.032 0.013 0.016 0.012 The mean value of evaluation metrics in terms of SD value gained from the NSL-KDD dataset by each classifier is displayed in Figure 6.20. It can be observed that the ICET surpasses all other classification models in terms of SD values of evaluation measures. The main cause of the greater enactment of ICET is that the intelligent BIOCFS and ensemble classifier can increase the effectiveness of the system. From Figure 6.20, it can be witnessed that the SD of the ICET was less than all other classifiers with respect to the evaluation metrics. Hence, the ICET classification model
139 delivers much more reliable outcomes for identifying intrusion than the others. More precisely, ICET is a very viable method for detecting intrusion. Figure 6.13 Performance of ICET with BIOCFS in terms of SD values on the NSL-KDD dataset 6.6.4 Performance of IDS with BIOCFS on CIC-IDC 2017 dataset The proposed BIOCFS selects 12 features from CIC-IDS 2017 dataset for improving the performance of the classification model. The selected features in both datasets are given in Table 6.18. To reduce the dimension of feature space, BIOCFS is applied to the dataset and the selected features are used to classify the network traffics. The proposed classifier is implemented with the BIOCFS algorithm and the comprehensive outputs obtained by this model regarding performance measures are listed in Table 6.21. From this table, we can observe that the SVM classifier has achieved decent classification enactment with 89.17% ACC, 85.50% PRE, 86.57% REC, 98.32% F1M, and 81.29% ADR with 0.13% FAR and 0.055
ρ-
value. It consumes 266s for model building and a massive 5204s for testing since its time convergence characteristic is poor. The RF classification model is
140 generated reasonable outputs related to the SVM classifier. The performance of RF is better than SVM in terms of ACC (91.75%), PRE (88.86%), REC (94.31%), F1M (96.17%), and ADR (79.45%) with FAR (0.068) and
ρ-
value (0.057). It consumes only 52s for model building and 75s for testing due to the capacity of RF in constructing several powerful decision trees quickly. Table 6.21 Results obtained by ICET using BIOCFS on the CIC-IDS 2017dataset in terms of the mean value Algorithm ACC PRE REC F1M FAR ADR ρ-value Training time (s) Testing time (s) SVM 89.17 85.50 86.57 98.32 0.133 81.29 0.055 266 5204 C.4.5 92.71 87.06 90.22 95.24 0.110 84.10 0.080 43 8 RF 91.75 88.86 94.31 96.17 0.068 79.45 0.057 52 75 BF 92.86 89.44 95.74 96.39 0.062 82.62 0.037 76 38 EIDS 93.21 90.56 95.31 96.57 0.008 84.33 0.038 279 15 ML-IDS 97.67 92.03 96.47 97.51 0.020 91.40 0.036 50 646 ENML 98.44 93.06 99.06 97.79 0.045 92.10 0.048 105 726 REPTree 94.94 93.88 96.43 94.40 0.019 92.04 0.049 162 248 KODE 98.84 93.67 97.70 99.63 0.004 94.49 0.044 355 4992 ICET 99.37 98.36 99.19 99.87 0.003 99.90 0.007 147 18 By eliminating the overfitting problem, the C4.5 classifier achieves better results related to the SVM classifier regarding classification performance
in terms of accuracy (92.71%), precision (87.06%), recall (90.22%), F1-measure (95.24%), false alarm rate (0.110), and attack detection rate (84.10%).
The time consumption for building and testing the model is also very low as 43s and 8s, respectively. By applying the concept of FPA, the balanced forest achieves 92.86% ACC, 89.44% PRE, 95.74% REC, 96.39% F1M, and 82.62% ADR with 0.062% FAR and 0.037 ρ-value. The time required for training and testing is 27s and 2s, respectively. The FPA
141 used in the BF can improve the prediction stability of the classification process. Figure 6.14 Performance of ICET with BIOCFS in terms of ACC, PRE, REC, F1M, and ADR on the CIC-IDS dataset Figure 6.15 Performance of ICET with BIOCFS in terms of FAR and ρ-value on the CIC-IDS 2017 dataset

142 The EIDS classification model provides 93.21% of classification performance, 90.56% of precision, 95.31% of recall, 96.57% of F1-measure, and 84.33% of attack detection rate. Also, it provides 0.008 of FAR and 0.038 of ρ-value. It records 279s model building time and 15s testing time with a reduced false alarm rate of 0.018. The ML-IDS model exploits the different efficient MLTs with majority voting as a combiner. This model yields improved performance with accuracy, precision, recall, F1-measure, false alarm rate, and attack detection rate
with 97.67%, 92.03%, 96.47%, 97.51%, 0.020%, and 91.40%. This model also achieves 0.036 ρ-value and requires more time to test the model (50s) related to training the model (646s) since it combines numerous decision trees to determine the appropriate class. Figure 6.16 Performance of ICET with BIOCFS in terms of training and testing time on the CIC-IDS 2017 dataset The ENML model achieves improved solutions regarding ACC (98.44%), PRE (93.06%), REC (99.06%), F1M (97.79%), and ADR (92.10%) with FAR (0.05) and
ρ-
value (0.048). It consumes 105s for model building and 726s for testing due to the capacity of constructing many powerful decision trees quickly. The ensemble classifier REPTree achieves a lower FAR (0.019) with 94.94% of classification
accuracy, 93.88% of precision,
143 96.43% of recall, 94.40% of F1-measure, and 92.04% of
attack detection rate. The time taken by the classifier for training and testing the model are 162s and 248s, correspondingly. KODE provides better solutions for the original attributes of the dataset regarding performance metrics, including accuracy (98.84%), precision (93.67%), recall (97.70%), F1-measure (99.63%), attack detection rate (94.49%), the minimum false alarm rate of only 0.004, and ρ-
values of 0.044. However, it exhibits poor enactment in terms of other measures, like an increased model training time (355s) and (4992s). The proposed ensemble classifier outperforms all other individual and ensemble learners in terms of 99.37% of classification
accuracy, 98.36% of precision, 99.19% of recall, 99.87% of F1-measure, and 99.90% of
attack detection rate. More importantly, it provides only
a 0.003 of false alarm rate and ρ-
values of 0.007. However, it takes colossal time for training and testing such as 147s and 18s, respectively. Table 6.22 Results obtained by ICET using BIOCFS on the CIC-IDS 2017dataset in terms of SD value Algorithm ACC PRE REC F1M FAR ADR ρ-value SVM 0.018 0.012 0.018 0.039 0.053 0.013 0.012 C.4.5 0.017 0.009 0.023 0.040 0.036 0.018 0.028 RF 0.032 0.011 0.038 0.035 0.040 0.033 0.012 BF 0.045 0.011 0.030 0.042 0.016 0.025 0.018 EIDS 0.021 0.008 0.028 0.026 0.015 0.023 0.011 ML-IDS 0.026 0.013 0.033 0.031 0.000 0.021 0.016 ENML 0.015 0.005 0.031 0.025 0.021 0.018 0.013 REPTree 0.020 0.010 0.036 0.030 0.026 0.015 0.018 KODE 0.025 0.015 0.041 0.035 0.031 0.013 0.023
144 ICET 0.008 0.004 0.023 0.025 0.006 0.009 0.005 From Figure 6.17, it can be witnessed that the SD of the ICET was less than all other classifiers with respect to the evaluation metrics. Hence, the ICET classification model delivers much more reliable outcomes for identifying intrusion than the others. More precisely, ICET is a very viable method for detecting and classifying intrusion. Figure 6.17 Performance of ICET with BIOCFS in terms of SD values on the CIC-IDS 2017 dataset 6.6.5
Assessment of the Proposed BIOCFS with other Feature Selection Methods To further evaluate the reliability and effectiveness of the intended optimized attribute selection approach BIOCFS, this study carries out copious experiments, which relate the enactment of the proposed model with other renowned attribute selection approaches. Some of the efficient preferred approaches for this comparison are
Information Gain (IG) (Abdullah et al. 2018), Passive-Aggressive (PA) (Wang et al. 2021),
Bat Algorithm (BA) (Yang & He 2018), CFS (Singh & Singh 2018), and HFS (Jaw & Wang
145 2021).
Tables 6.23 and 6.24 compare the performance improvement of the proposed BIOCFS attribute selection algorithm with other feature selection algorithms used in some popular IDS models. Table 6.23 Evaluation of the proposed BIOCFS with other methods in terms of ACC, FAR, and ADR Attack type Accuracy False alarm rate Attack detection rate NSL- KDD CIC-IDS 2017 NSL- KDD CIC-IDS 2017 NSL- KDD CIC-IDS 2017 CFS 95.41 95.62 0.063 0.071 88.16 97.42 BA 98.88 98.90 0.054 0.063 92.29 97.41 HFS (KODE) 99.08 99.15 0.014 0.040 95.59 98.60 IG (EIDS) 95.17 96.70 0.051 0.067 86.91 90.85 PA 95.69 95.98 0.036 0.054 86.62 96.49 BIOCFS 99.75 99.87 0.011 0.003 99.24 99.90 Table 6.24 Evaluation of the proposed BIOCFS with other methods in terms of PRE, REC, and F1M Attack type Precision Recall F1-measure NSL- KDD CIC-IDS 2017 NSL- KDD CIC-IDS 2017 NSL- KDD CIC-IDS 2017 CFS 95.61 94.83 94.52 93.26 95.3 95.1 BA 96.22 95.68 96.53 94.27 98.8 98.5 HFS (KODE) 97.34 96.52 94.29 91.73 99.7 99.8 IG (EIDS) 89.31 87.56 88.56 86.41 95.1 96.6 PA 92.37 91.45 90.16 90.12 95.4 95.9 BIOCFS 98.64 98.36 98.23 99.19 99.506 99.87
146 It is observed from these tables, BIOCFS outperforms other cutting-edge feature selection methods in terms of performance measures. Figures 6.18 (a) – (f) summarize the result of the assessment of BIOCFS with

different pioneering attribute selection approaches using two standard databases regarding many measures. Figure 6.18 (a) demonstrates that the recommended technique (BIOCFS) realizes the greatest classification accuracy on both databases related to other state-of-the-art approaches. The BIOCFS realizes a predictive
accuracy of 99.75% and 99.87%, on the NSL- KDD, and CIC-IDS 2017 databases, correspondingly. Conversely, the HFS scheme provides better accuracy related to the other implemented methods, like 99.08% and 99.15% on the two databases, it is still below the enactment of BIOCFS.
Likewise,
BIOCFS achieved the greater ADR for both the databases, with a maximum of 99.90% on the CIC-IDS 2017 dataset and 99.24% ADR on NSL-KDD as shown in Figure 6.18 (b).
Similarly, the HFS technique employed in the KODE approach provides enhanced ADR such as 95.59%
and 98.60% on NSL-KDD and CIC-IDS 2017 databases, respectively. The feature selection method using BA also provides decent ADR such as 92.29% and 97.41% on NSL-KDD and CICIDS-2017 datasets, respectively. The IG filter-based feature selection achieves 90.85% ADR on the CICIDS-2017 dataset; however, it only achieves 86.91% on
the
NSL- KDD dataset. The CFS and PA-based methods achieve their best results such as 97.42% ADR on CIC-IDS 2017 and 96.49% ADR on
the
NSL-KDD dataset. Regarding false alarm rate, BIOCFS provides its best result (0.011 on the NSL-KDD dataset and 0.003 FAR on the CICIDS-2017 dataset) as given in Figure 6.18 (c). It outperforms other feature selection methods in terms of precision (98.64% on the NSL-KDD and 98.36% on the CIC-IDS 2017), recall (98.23% on the NSL-KDD and 99.19% on the CIC-IDS 2017),
147
and F1-measure (99.50% on the
NSL-KDD and 99.87% on the CIC-IDS 2017). Bearing the complete enactment of the BIOCFS method in mind, it is worth mentioning that it has considerably outdone the other traditional feature selection methods, thus making it greater among the designated methods in many recent studies regarding
effectiveness, dependability, and enactment. Among all the implemented approaches, BIOCFS achieves similar precision for the NSL-KDD dataset (98.64%) and the CIC-IDS 2017 (98.36%) as compared to CFS, BA, HFS, IG, and PA. Figure 6.18(d) shows the performance of all the implemented methods in terms of precision. The proposed BIOCFS outperforms all other methods in terms of recall as given in Figure 6.18(e). Similarly, the BIOCFS outperforms all other methods in terms of F1-measure as illustrated by Figure 6.18(f). (
a)
148 (b) Figure 6.18 (Continued) (c)
149 (d) Figure 6.18 (Continued) (e)
150 (f) Figure 6.18 Evaluation of BIOCFS in terms of performance measure 6.6.6 Effect of Number of Selected Features on Performance of Classifier This study attempts to reduce the number of selected features in order to reduce the processing and timing overhead of the IDS model. Table 6.25 illustrates the impact of a number of features on the performance measure related to the proposed IDS model.
Figure 6.19 displays
the assessment results of the number of designated attributes with their corresponding time for BIOFCS and other approaches. For instance, Figure 6.19 demonstrated that the BIOFCS technique achieves a noteworthy enactment regarding the selected features, such as selecting the minimum number of attributes (7 attributes for the NSL-KDD and 12
features for the CIC-IDS 2017 datasets), correspondingly.
151
Table 6.25 Impact of number of features of performance measure Attack type Number of features Time for selection NSL-KDD CIC-IDS 2017 NSL- KDD CIC-IDS 2017 CFS 18 21 4.95 20.13 BA 9 11 10.11 43.34 HFS (KODE) 8 13 5.12 14.36 IG (EIDS) 13 17 2.56 11.12 PA 5 13 10.08 22.45 BIOCFS 7 12 4.18 10.26
However, PA records the lowest number of selected features (5 features) as compared to other methods HFS (8 features), BA (9 features), IG (13 features), and CFS (18 features) for the NSL-KDD dataset. In the case of the CIC-IDS 2017 dataset, the proposed BA achieves a minimum number of features (11 features) as compared to other methods BIOCFS (12 features), HFS (13 features), PA (13 features), IG (17 features), CFS (21 features) for the NSL-KDD dataset.
152 Figure 6.19 Number of selected features Finally,
in spite of the substantial reduction in the training and testing time of the classifiers on the designated attributes from both databases, BIOCFS has disappointed in terms of attribute selection time (4.18s for the NSL-KDD dataset and 10.26s for the
CIC-IDS 2017 dataset) related to other approaches, as given in Figure 6.20, particularly for the IG method (2.56s for

the NSL-KDD dataset and 11.12s for the
CIC-IDS 2017 dataset), CFS method (4.95s for NSL-KDD dataset and 20.13s for CIC-IDS 2017 dataset), HFS method (5.12s for
the NSL-KDD dataset and 14.36s for the
CIC-IDS 2017 dataset). Thus, IG, HFS, and CFS methods outperformed this proposed BIOCFS feature selection method in terms of the time taken to select features. But, this is not very imperative when bearing the complete enactment of BIOCFS
in mind.
153
Figure 6.20 Time consumption for feature selection 6.6.7
Comparison of Various Adopted Combinations This section elaborates on the various experimental results obtained using selected combination rules with the ICET-based ensemble method. The widely used combination rules for constructing ECs in prevailing studies are the majority voting, product of probabilities, minimum probability, maximum probability,
and
AoP (Catal & Nangir 2017). Besides, since the numbers of classes are more significant than the base classifiers and the decent performance of the AoP over majority voting.
Hence, it is concluded that the adoption of the AoP combination rule is well-suited for ensemble models to integrate the results from individual classifiers. Table 6.26 displays the results gained by the proposed classifier in terms of accuracy for various combination rules on the NSL-KDD datasets. From the results, it is observed that all the combination rules realized good performances on the normal records. The minimum probability records the nominal performance in terms of accuracy of 98.61% for normal data samples and 98.36% for DoS attacks, 97.18% for probing attacks,
and 91.56%
for
154 R2L. However, it fails to provide the same striking performance for the other attacks, particularly U2R. It achieves only 53.73% of accuracy for U2R attacks. The other two combination rules maximum probability and product of probability provide similar results. The average of probabilities provides the maximum accuracy of 99.96% for normal samples and 99.63% for probe, and 99.62% for R2L attacks. However, it fails to show the same extraordinary performance for some other attacks, particularly U2R. It achieves only 71.95% of accuracy for U2R attacks. But this is optimal performance as compared to other combination rules. Table 6.26 Impact of various combiners on the accuracy of the classifier on the NSL-KDD dataset Attack type Minimum Probability Maximum Probability Product of Probability Majority Voting Average probabilities Benign 98.61 99.51 98.24 99.94 99.96 DoS 98.36 99.06 99.06 99.83 99.14 Probing 97.18 98.37 97.5 98.62 99.63 R2L 91.56 91.61 96.65 97.72 97.62 U2R 53.73 51.97 55.15 71.39 71.95 Finally, Table 6.27 displays the results obtained by ICET with various
combination rules on the CIC-IDS 2017 database. These results proved that the AoP combiner is the best performing rule for most of the threats in the database. For example, DDoS, Web Attack, and normal samples gained the maximum accuracy of 99.98%, 99.96%, and 99.95%, correspondingly. However, the classification accuracy in SSH_Patator and probing are 81.85% and 89.34%,
correspondingly.
Table 6.27 Accuracy comparison of various combiners on the CIC- IDS2017 test dataset
155 Attack type Minimum Probability Maximum Probability Product of Probability Majority Voting Average probabilities Normal 96.91 95.68 98.1 99.67 99.95 Botnet 95.92 93.41 96.98 98.34 99.15 DoS 94.53 92.9 95.32 96.89 98.89 DDoS 96.07 95.33 97.94 99.04 99.98 FTP_Patator 96.42 94.92 97.54 97.89 99.09 Probing 85.55 83.32 86.78 87.85 89.34 SSH_Patator 76.16 75.56 78.69 79.86 81.85 Web attack 92.56 90.9 94.89 97.73 99.96
Albeit SSH_Patator contains more samples than Botnet threat, it registers the minimum
predictive performance for all the rules used
for integrating the results, the reason which will be examined in impending research. The Probing shows the second-lowest enactment with a mean accuracy of 85.54% for all the combiners, only performing better than SSH_Patator amongst all the threats in the
database. From these findings, it is concluded that
regardless of the outstanding enactment of the other combiners, it is undeniable that the AoP realized the greatest enactment for all types of cyberattacks on the different databases. Hence, this present study implements the AoP as the combiner in the developed ensemble classifier. 6.7 CONCLUSION This chapter presents a summary of the comprehensive empirical analysis carried out in this research. The empirical analysis has been performed on two real-world datasets using the proposed ICET-based intrusion detection models with appropriate feature selection algorithms. Also, it presents an in-depth performance appraisal of the base classifiers and

156 ICET on each of the two standard datasets (i.e., NSL-KDD and CIC-IDS 2017), the designated attributes, different combination techniques, and state- of-the-art classifiers in terms of performance measures such as classification accuracy, precision, F-measure, the false alarm rate, and the attack detection rate. The time consumption for model building and testing is also considered in this study to assess the real-time performance of the proposed model. Furthermore, to alleviate data splitting problems, the k-fold CV portions are fixed to explicit bounds of learning and testing proportions that are not deceptive in the learning phase to validate the performance, credibility, and generalizability of the proposed IDS. To realize a more precise evaluation of the model's enactment, the 10-fold CV is used. The experimental results reveal that the proposed ICET model

outperforms other state-of-the-art approaches in terms of

performance measures.

157 CHAPTER 7 CONCLUSION AND FUTURE DIRECTION The pervasiveness of internet technology and its tremendous communication speed has led to the development of several networks by numerous industries across the vertical market. At present, a colossal number of global organizations carry out business dealings over the internet. This has intensified the volume of network traffic flowing in and out of business organizations making real-time analysis a challenging endeavor for security engineers and network managers. Therefore, the increased number of business dealings has allured an outrageous number of hackers to the business information systems. The attackers exploit unconventional methods and cutting-edge technologies to launch new and well-refined threats every day. To enable the identification of new and anonymous threats, innumerable research efforts have focused on increasing and optimizing the performance of IDS by selecting only germane attributes for training the IDS. Indeed, the modern network traffic involved a large number of features many of which are unrelated for data classification as either genuine or cyberattack. Selecting significant features can greatly decrease the intricacy of the IDS model, make it more understandable, increase classification accuracy, and evade overfitting problems. This research proposes an Intelligent Classifier using Ensemble Technique (ICET) to increase the accuracy as well as ADR and reduce the FAR in classifying the intrusive activities significantly. The proposed ICET includes a feature selection module and an ensemble classifier.

To cope with high dimensional feature-

158 rich inbound traffic in large networks,

the feature selection module exploits the CFS algorithm to select the appropriate features. The proposed feature selection module estimates

the correlation of the identified features and chooses the ideal subset for the training and testing phases.

Besides, it exploits the optimized RelieF algorithm to calculate the quality of attributes. The attributes with a low-quality index are eliminated to reduce the dimensionality of the feature space. The performance of the proposed feature selection approach is further enhanced by integrating CFS with BIO. The integration

of BIO and CFS algorithms

is embedded in the proposed ensemble classifier to increase the performance of the IDS. The proposed ensemble classifier module includes three different classifiers including BF, RF, and C4.5 decision tree based on VM using the rule of AoP. The established ICET aids to handle unbalanced and multi-class datasets with higher accuracy. In this study, the ICET model is trained and assessed using real-world datasets such as NSL-KDD and CIC-IDS 2017 using Weka 3.8.3 workbench. In this chapter, we discuss the results of the proposed IDs approach. For each point, this chapter describes the research findings and interprets them accordingly. It concludes the research by discussing the future directions. 7.1 INTRODUCTION The findings in this thesis can contribute to a deeper understanding of the practical challenges of implementing IDS, which can result in a further step toward more secure network services. Security is one critical aspect of this network that needs unremitting attention to thwart malevolent and evil- intentioned intruders from disturbing the normal behavior of the network and to avoid the devastating consequences from it. Hence, there is a dire need of exploring the different cyberattacks and various available security solutions in the literature. IDSs are already proven to provide overall security in various other networks so it is a good step to explore the outcomes of deploying IDS

159 in the network. With this motivation, this work attempts to develop an intelligent IDS model to identify potential cyberattacks effectively. Firstly, this work proposes an optimized correlation-based feature selection algorithm to reduce the dimension of the feature space by selecting the optimal feature subset. Then, an ensemble classifier is used to classify the network traffic based on the selected feature. For evaluating the performance of the proposed optimized IDS model, a comprehensive set of empirical analyses is carried out and the results are analyzed with respect to several performance metrics. In this work, the Weka software tool is used for more accurate evaluation and validation. Extensive simulation results prove that the proposed approach considerably outdoes other existing approaches. Thus, the proposed approach ensures the normal functioning of the system, identifies the incidence of security attacks and increases the availability of the services, equipment, and system. 7.2 CONCLUSIONS ON PERFORMANCE EVALUATION OF BIOCFS ALGORITHM The effectiveness of the feature selection process is evaluated by assessing the enactment of the ICET model with and without using BIOCFS in classifying network traffics into normal or attack. From the experimental results, the following conclusions have been drawn: 1. The test results reveal that the proposed ICET did not perform well on the

entire set of attributes of the database

without using the feature selection algorithm. 2. When applying on NSL-KDD, it provides only nominal performance with a classification accuracy of 93.50%, precision of 90.28%, recall of 94.87%, F1-measure of 98.15%, the detection rate of 92.65%. Moreover, the ICET records the

160 highest model training time of 439.43s with a massive FAR of 0.016%. 3. When applying to the CIC-IDS 2017 dataset, the proposed ensemble classifier outperforms all other individual and ensemble learners in terms of 92.25% of classification

accuracy, 91.23% of precision, 94.02% of recall, 96.20% of F1-measure, and 91.73% of

attack detection rate. It provides only 0.011 of FAR. However, it takes colossal time for training and testing such as 143s and 14s, respectively. 4. It is found that the performance measures obtained by the ICET model without BIOCFS are superior to all other approaches such as SVM, C.4.5, RF, BF, EIDS, ML-IDS, ENML, REPTree, and KODE algorithms. 5. However, without applying BIOCFS feature selection, ICET performs poorly. The maximum performance of this model is 94.8% of accuracy, 93.1% of precision, 97.4% of recall, 99% of F1-measure, 94.8% of attack detection rate, 0.12% FAR, and 0.003

ρ-

value. 6. From the results, it is found that most of the p-values achieved by the proposed ensemble approach from the Wilcoxon rank- sum test are less than 0.05 which proves that the enhancement realized by the proposed ensemble approach is statistically significant. 7. The key reason behind the lower performance of an ICET classifier is the higher dimensional feature space in the dataset since the classifier exploits the entire feature space (41 features in NSL-KDD and 84 features in CIC-IDS 2017 dataset)

161 7.3 CONCLUSIONS ON PERFORMANCE EVALUATION OF ICET MODEL In the second phase of the main experimentation, we compare the proposed approach ICET using BIOCFS to other approaches in terms of classification accuracy, precision, F1-measure, FAR, and ADR. Even though the proposed ICET classifier provides better results as compared to some individual classifiers, its performance is not superior with respect to some measures (e.g., FAR, time for testing, and training process) without applying the feature selection process. Hence, the proposed ICET model exploits BIOCFS to achieve performance enhancement. It is observed that the ICET using the BIOCFS algorithm outperforms other methods in terms of performance measures. From the experimental results, the following conclusions have been drawn: 1. Through conducting extensive experiments, this study finds out an optimal number of features in each dataset. It selects 7 features from NSL-KDD and 12 features from the CIC-IDS 2017 dataset. 2. Through empirical analysis this study find out the efficient combination rule for integrating results from individual classifiers in ensemble learners. The experimental results prove that AoP is the more effective rule as compared to other rules used in various ensemble classifiers. 3. The proposed ICET with BIOCFS classifier reveals a profound enhancement in the classification of intrusion into corresponding classes on the NSL-KDD dataset. It achieves much higher classification performance with an ACC of 99.85%, PRE of 98.64%, REC of 98.23%, F1M of 99.51%,

162 ADR of 99.24%, FAR of 0.01%, and ρ-value 0.001. It takes 73s for training and 671s for testing the data samples. 4. The proposed ensemble classifier outperforms all other individual and ensemble learners in terms of 99.37% of classification accuracy, 98.36% of precision, 99.19% of recall, 99.87% of F1-measure, and 99.90% of

attack detection rate. It provides only 0.003 FAR and 0.007 ρ-values. However, it takes colossal time for training and testing such as 147s and 18s, respectively. 5. It can be observed that the standard deviation of the performance measure achieved by the ICET is lower than all other classifiers in terms of the evaluation metrics. 6. Hence, the ICET classification model delivers much more reliable outcomes for identifying intrusion than the others. More precisely, ICET is a very viable method for detecting and classifying intrusion. By this research, it is concluded that the proposed ICET with BIOCFS can provide better solutions in a network. 7.4 LIMITATION Although the proposed model is effective in detecting intrusions on large-scale data with imbalance, these models could not process data with borderline and noisy samples. When combined with data imbalance, noise, and borderline samples further complicate the process of classification of large-scale data. Borderline data refers to those samples on the borderline between the two classes and that makes it very difficult for the classifier to demarcate samples that are located safely among the instances of a different class. them. Besides, the proposed models are not effective in sensing zero- day variances, as they need enough training data to learn the patterns. The

163 absence of signatures for those new attacks makes it impossible for a supervised model to detect their presence. Also, attack patterns that exhibit trends, seasonal patterns, and drifts are not handled by the current model. 7.5 FUTURE DIRECTION Intrusion detection systems are extremely helpful tools that aid security administrators in the ever-evolving task of securing the network. The art of managing intrusion detection systems is not simple and needs continuous effort and attention. Regardless of the remarkable enactment of ICET using BIOCFS, it has some obvious downsides that still need enhancement. For example, the training and testing time of the proposed model on NSL-KDD are 147s and 18s, respectively.

In the future, we intend to: • Build a correlation module to reduce the
training and testing drastically. • Investigate why HFS-based KODE has
related enactments with the proposed model on the original and designated attributes of both databases. •
Evaluate the ICET with other existing datasets. • Develop more precise base classifiers particularly, for the detection of minority attacks such as U2R and R2L. • Develop algorithms to combine kernel methods with other classification methods for pattern analysis and optimization techniques for individual classifiers in ensemble learners.

164 REFERENCE 1.

Adnan, N & Islam,
Z 2017, '
pp. 389–403. 2.
Abdullah,
M, Alshannaq, A, Balamash, A & Almabdy, S 2018, 'Enhanced Intrusion Detection System using Feature Selection Method and Ensemble Learning Algorithms',
International Journal of Computer Science and Information Security,
vol. 16,
pp. 23-34. 3.
Aburomman, A & Reaz, MBI 2017, 'A survey of intrusion detection systems based on ensemble and hybrid classifiers',
Computer Security,
vol. 65, pp. 135–152. 4. Ahmad, I, Ul Haq, QE, Imran, M, Alassafi, MO & AlGhamdi, RA 2022, 'An Efficient Network Intrusion Detection and Classification System', Mathematics, vol. 10, no. 530, pp. 1-15. 5. Akashdeep,
Manzoor, I & Kumar, N 2017, 'A feature reduced intrusion detection system using ANN classifier,' Expert Systems with Applications,
vol. 88, pp. 249-257. 6. Akintola, A, Balogun, A & Capretz, LF 2022, 'Empirical Analysis of Forest Penalizing Attribute and Its Enhanced Variations for Android Malware Detection', Applied Sciences,
vol. 12, no. 9, no. 4664. 7. Aldwairi, T, Perera, D & Novotny, MA 2018, '
An evaluation of the performance of restricted boltzmann machines as a model for anomaly network intrusion detection', Computer Networks,
vol. 144, pp. 111- 119. 8.
Alia, A, & Taweel, A 2021, 'Enhanced Binary Cuckoo Search With Frequent Values and Rough Set Theory for Feature Selection,' IEEE Access,
vol. 9, pp. 119430-119453. 9.
Aljawarneh, S, Aldwairi, M & Yassein, MB 2018, 'Anomaly-based intrusion
detection system through feature selection analysis and building hybrid efficient model,'
Journal of Computational Science,
vol. 25, pp. 152-160.
165 10.
Al-yaseen, WL, Ali, Z, Zakree, M & Nazri, A 2017, '
Real-time multi- agent system for an adaptive intrusion detection system', Pattern Recognition Letters,

vol. 85, pp. 56-64. 11.
Ambusaidi, MA,
He, X, Nanda, P & Tan, Z 2016, '
Building an Intrusion Detection System Using a Filter-Based Feature Selection Algorithm', IEEE
Transaction on Computers, vol. 65, pp. 2986-2998. 12.
Bace, R & Mell, P, 2001, 'Intrusion Detection Systems,' Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD. 13.
Bai, L, Wang, Z, Shao, YH & Deng, NY 2014, 'A novel feature selection method for twin support vector machine', Knowledge-based
System vol. 59, pp. 1-8. 14. Bala Ganesh, D, Chakrabarti, A & Midhunchakkaravarthy, D 2018, 'Smart Devices Threats, Vulnerabilities and Malware Detection Approaches: A Survey', European Journal of Engineering Research and Science, vol. 3,
pp. 7-12. 15.
Bala, R & Nagpal, R 2019, 'A review on KDD cup99 and NSL NSL- KDD dataset', International Journal of Advanced Research in Computer Science,
vol. 10, no. 2,
pp. 64-67. 16.
Belouch, M, Idhammad, M, El, S 2017, 'A Two-Stage Classifier Approach using RepTree Algorithm for Network Intrusion Detection',
International Journal Advanced Computing, vol. 8, pp. 389-394. 17. Bhati, BS, Rai, CS, Balamurugan, B & Al-Turjman, F 2020, 'An intrusion detection scheme based on the ensemble of discriminant classifiers,' Computers & Electrical Engineering,
vol. 86, no. 106742. 18. Bolón-Canedo, V, Sánchez-Maroño, N & Alonso-Betanzos, A 2018, 'Feature selection
and classification in multiple class datasets: An application to KDD Cup 99 dataset,' Expert Systems with Applications, vol. 38, no. 5, pp. 5947-5957. 19. Bouckaert, RR, Frank, E, Hall M, Kirkby R & Reutemann, P 2013, 'WEKA Manual for Version 3-7-8', Hamilton, New Zealand. 20. Breiman, L, 2001, 'Random forests,' Machine learning, vol. 45, no. 1, pp. 5-32.
166 21. Bridges, RA, Glass-Vanderlan, TR, Iannacone & Vincent, MS 2019, 'A Survey of Intrusion Detection Systems Leveraging Host Data', ACM Computing Survey, vol. 52, no. 6, no. 128. 22.
Cai, J, Luo, J, Wang, S & Yang, S 2018, 'Feature selection in machine learning: A new perspective', Neurocomputing, vol. 300, pp. 70-79. 23.
Chormunge, S & Sudarson Jena, S 2018, 'Correlation based feature selection with clustering for high dimensional data,' Journal of Electrical Systems and Information Technology, vol. 5, no. 3, pp. 542-549. 24. Cybersecurity Ventures, 2019, Official Annual Cybercrime Report, Herjavec Group, https://www.herjavecgroup.com/wp-content/uploads/ 2018/12/CV-HG-2019-Official-Annual-Cybercrime-Report.pdf 25. Damaševiˇcius, R, Venˇckauskas,
A, Toldinas, J & Grigaliunas, Š 'Ensemble-Based Classification Using Neural Networks and Machine Learning Models for Windows PE Malware Detection', Electronics,
vol. 10, no. 485. 26. Danasingh, AAGS, Subramanian, A & Epiphany, J 2020, 'Identifying redundant features using unsupervised learning for high-dimensional data,' SN Applied Sciences, vol. 2, no. 1367. 27. Desuky, AS & Hussain, S 2021, 'An Improved Hybrid Approach for Handling Class Imbalance Problem', Arabian Journal for Science and Engineering, vol. 46,
pp. 3853-3864. 28.
Dhanabal, L &
Shantharajah, SP 2017, 'A study on NSL-KDD dataset for intrusion detection system based on classification algorithms', International Journal of Advanced Research in Computing and Communication Engineering,
vol. 4,
pp. 446-452. 29.
Disha, RA & Waheed S 2022, 'Performance analysis of machine learning models for intrusion detection system using Gini Impurity- based Weighted Random Forest (GIWRF) feature selection technique', Cybersecurity, vol. 5. 30.
Elhag, S, Fern´andez,
A, Altalhi, A, Alshomrani, S & Herrera, F 2019, 'A multi-objective evolutionary fuzzy system to obtain a broad and accurate set of solutions in intrusion detection systems,' Soft Computing,
vol. 23, no. 4, pp. 1321-1336. 31.
Fadlullah, ZM, Tang, F, Mao, B, Kato, N, Akashi, O, Inoue, T & Mizutani, K 2017, 'State-of-the-Art Deep Learning: Evolving Machine
167 Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems', IEEE Communication Survey Tutor, vol. 19,
pp. 2432-2455. 32.

Farahani, G 2020, ‘Feature Selection Based on Cross-Correlation for the Intrusion Detection System’, Security and Communication Networks, vol. 2020. 33.
Farnaaz, N &
Jabbar, MA 2016, 'Random Forest Modeling for Network Intrusion Detection System,' Procedia Computer Science, vol. 89, pp. 213-21. 34.
Feng,
X, Xiao, Z, Zhong, B, Qiu, J & Dong, Y 2018, ‘Dynamic ensemble classification for credit scoring using soft probability,’ Applied Soft Computing, vol. 65, pp. 139-151. 35.
Freund, Y & Schapire, R 1995, ‘A desicion-theoretic generalization of on-line learning and an application to boosting,’ Computer Learning theory, vol. 55, pp. 119-139. 36. Freund, Y & Schapire, RE 1996, ‘
Experiments with a new boosting algorithm,’ Machine learning,
vol. 96, pp. 148-156. 37. Friston, K, Stephan, K, Li, B & Daunizeau, J 2010, ‘Generalised filtering,’ Mathematical Problems in Engineering,
vol. 2010,
pp. 78-86. 38. Gajewski, M, Batalla, JM, Mastorakis, G & Mavromoustakis, CX 2019, ‘A distributed IDS architecture model for Smart Home systems’, Cluster Computing, vol. 22, pp. 1739-1749. 39.
Gao, X, Shan, C, Hu, C, Niu, Z & Liu, Z 2019, ‘An Adaptive Ensemble Machine Learning Model for Intrusion Detection,’ IEEE Access,
vol. 7, pp. 82512-82521. 40. Gennari, JH, Langley, P & Fisher, D 1989, ‘Models of incremental concept formation’, Artificial Intelligence’, vol. 40, pp. 11-61. 41. Ghaddar, B & Naoum-Sawaya, J 2018, ‘High dimensional data classification and feature selection using support vector machines’, European Journal of Operational Research,
vol. 265, pp. 993-1004. 42. Hota, HS &
Shrivas, AK 2014, ‘
Decision Tree Techniques Applied on NSL-KDD data and its Comparison with Various Feature Selection Techniques’, In Advanced Computing, Networking and Informatics; Springer:
Berlin/Heidelberg, Germany,
vol. 1, pp. 1-24.
168 43. Hssina,
B, Merbouha, A, Ezzikouri, H & Erritali, M 2014, ‘A comparative study of decision tree ID3 and C4. 5,’
International Journal of Advanced Computer Science and Applications, vol. 4, no. 2,
pp. 10-20. 44.
Hu,
J 2018, ‘
An approach to EEG-based gender recognition using entropy measurement methods,’ Knowledge-Based Systems,
vol. 140,
pp. 134-141. 45.
Hung, C & Chen, JH 2009, ‘
A selective ensemble based on expected probabilities for bankruptcy prediction,’ Expert systems with applications,
vol. 36, no. 3, pp. 5297-5303. 46.
Iwashita, AS 2019, ‘An Overview on Concept Drift Learning’, IEEE Access, vol. 7, pp. 1532-1547. 47.
Janarthanan, T & Zargari, S 2017, ‘Feature selection in UNSW-NB15 and KDDCUP'99 datasets,’ IEEE 26th international symposium on industrial electronics (ISIE),
pp. 1881-1886. 48.
Jaw, E & Wang, X 2021, ‘Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach’, Symmetry,
vol. 13, no. 10. 49. Jazi, HH, Gonzalez, H, Stakhanova, N & Ghorbani, AA 2017, ‘Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling’, Computer Networks, vol. 121, no. 2017, pp. 25-36. 50. Jia, W, Sun, M, Lian, J & Hou, S 2022, ‘Feature dimensionality reduction: a review’, Complex & Intelligent Systems, vol. 8, pp. 2663-2693. 51. Jianjian, D, Yang, T & Feiyue, Y 2018, ‘Novel Intrusion Detection System based on IABRBFSVM for Wireless Sensor Networks,’ Procedia Computer Science, vol. 131, pp. 1113-1121. 52. Jing Yu, J, Ye, X & Li, H 2022, ‘A high precision intrusion detection system for network security communication based on multi-scale convolutional neural network,’ Future Generation Computer Systems, vol. 129, pp. 399-406. 53. Karimi, Z, Kashani, MMR & Harounabadi, A 2013, ‘Feature Ranking in Intrusion Detection Dataset using Combination of Filtering
169 Methods’, International Journal of Computer Applications, vol. 78, pp. 21-27. 54. Kaur, S & Singh, MJ 2019, ‘Hybrid intrusion detection and signature generation using deep recurrent neural networks’, Neural Computing Applications, vol. 32, pp. 7859-7877. 55. Khader, M, Karam, M & Fares, H 2021, ‘Cybersecurity awareness framework for academia’, Information,

vol. 12, no.10, no. 417. 56.
Khammassi,
C & Krichen,
S 2017, ‘
A GA-LR wrapper approach for feature selection in network intrusion detection,’
Computers
and Security, vol. 70, pp. 255-277. 57. Khraisat, A & Alazab, A 2021, ‘A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public
datasets and challenges’, Cybersecurity, vol. 4. 58.
Khraisat,
A, Gondal, I, Vamplew, P & Kamruzzaman, J 2019, ‘
Survey of intrusion detection systems: techniques, datasets and challenges’, Cybersecurity,
vol. 2, no. 20. 59.
Kohavi, R & John, GH 1997, ‘Wrappers for feature subset selection’, Artificial Intelligence, vol. 97, pp. 273-324. 60.
Kononenko, I 1995, ‘On biases in estimating multi-valued attributes’, in Proceedings of the 14th international joint conference on Artificial intelligence,
vol. 2, pp. 1034-1040. 61. Kotsiantis,
S, Kanellopoulos, D & Pintelas, P 2006, ‘Data preprocessing for supervised leaning,’ International Journal of Computer Science,
vol. 1, no. 2, pp. 111-117. 62.
Kreibich, C & Crowcroft, J 2004, ‘Honeycomb: creating intrusion detection signatures using honeypots. SIGCOMM Computer Communication Review, vol. 34, no.1, pp.51-56. 63. Kulariya, M, Saraf, P, Ranjan, R & Gupta, GP 2016, ‘Performance analysis of network intrusion detection schemes using Apache Spark," 2016 International Conference on Communication and Signal Processing (ICCSP), pp. 1973-1977. 64. Kumar, R, Kumar, P, Tripathi, R, Gupta, GP, Sahil Garg, S & Hassan, MM 2022, ‘A distributed intrusion detection system to detect DDoS attacks
170 in blockchain-enabled IoT network,’ Journal of Parallel and Distributed Computing, vol. 164, pp. 55-68. 65. Kumar, S, Gupta, S & Arora, S 2021, ‘Research Trends in Network- Based Intrusion Detection Systems: A Review,’ IEEE Access, vol. 9, pp. 157761-157779. 66. Kumar, S, Tiwari, P & Zymbler, M 2019, ‘Internet of Things is a revolutionary approach for future technology enhancement: a review’, Journal of Big Data, vol. 6. 67.
Le, TTH, Kim, Y & Kim, H 2019, ‘Network intrusion detection based on novel feature selection model and various recurrent neural networks,’ Applied Science, vol. 9, no. 1392,
pp. 1-29. 68.
Lee, W, Stolfo, SJ &
Mok, KW 1999, ‘
A data mining framework for building intrusion detection models,’ Proceedings of the 1999 IEEE Symposium on Security and Privacy’,
pp.120-132. 69. Li,
H & Sun, J 2013, ‘Predicting business failure using an RSF-based CASE-based reasoning ensemble forecasting method,’ Journal of Forecasting, vol. 32, no. 2, pp. 180-192. 70.
Li, J, Cheng, K, Wang, S, Morstatter, F, Trevino, RP, Tang, J & Liu, H 2018, ‘Feature selection: A data perspective,’ ACM Computing Surveys (CSUR),
vol. 50, no. 6,
pp. 94-102. 71.
Li, Y & Chen, W 2020, ‘A Comparative Performance Assessment of Ensemble Learning for Credit Scoring,’ Mathematic, vol. 8, pp. 1756-1765. 72. Li, Y & Liu, Q 2021, ‘A comprehensive review study of cyber-attacks and cyber security; Emerging trends and recent developments,’ Energy Reports, vol. 7, pp. 8176-8186. 73. Li, Y, Xia, J, Zhang, S, Yan, J, Ai, X & Dai, K 2012, ‘An efficient intrusion detection system based on support vector machines and gradually feature removal method,’ Expert Systems with Applications,
vol. 39, no. 1, pp. 424-430. 74.
Liu, H & Lang, B 2019, ‘Machine
learning and deep learning methods for intrusion detection systems: A survey’,
Applied Science, vol. 9. 75.
Luo,
C, Wang, L & Lu, H 2018, ‘Analysis of LSTM-RNN based on attack type ofkdd-99dataset’, Proceeding of International Conference on Cloud Computing and Security, Springer, pp. 326-333.
171 76.

Maddikunta, PKR, Parimala, M, Koppu, S, Gadekallu, TR, Chowdhary, CL & Alazab, M 2020, 'An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture', Computer Communication, vol. 160, pp. 139-149. 77.
Mahfouz, A, Abuhussein, A, Venugopal, D & Shiva, S 2020, 'Ensemble Classifiers for Network Intrusion Detection Using a Novel Network Attack Dataset',
Future Internet, vol. 12, no. 11. 78. Mahjabin, T, Xiao, Y, Sun, G & Jiang, W 2017, 'A survey of distributed denial-of-service attack, prevention, and mitigation techniques', International Journal of Distributed Sensor Networks, vol. 13, no. 12, pp. 1- 27. 79. Malik, AJ, Shahzad, W & Khan, FA 2015, '
Network intrusion detection using hybrid binary PSO and random forests algorithm,' Security and Communication Networks,
vol. 8,
no. 16, pp. 2646-2660. 80.
Martins, I, Resende, JS, Sousa, PR, Silva, S, Antunes, S & Gama, J 2022, 'Host-based IDS: A review and open issues of an anomaly detection system in IoT,' Future Generation Computer Systems, vol. 133, pp. 95-113. 81. Mat, SRT, Ab Razak, MF, Kahar, MNM, Arif, JM, Mohamad, S & Firdaus, A, 2021, 'Towards a systematic description of the field using bibliometric analysis: malware evolution,' Scientometrics, vol.126, pp. 2013-2055. 82.
Meryem, A & EL Ouahidi, B 2020, 'Hybrid intrusion detection system using machine learning',
Network Security, vol. 4, no. 2020, pp. 8-19. 83. Mika, S, Ratsch, G, Weston, J, Schölkopf, B & Muller, KR 1999, 'Fisher discriminant analysis with kernels,' IEEE Signal Processing Society Workshop, pp. 41-48. 84. Mohamad, M, Selamat, A, Krejcar, O, Crespo, RG, Herrera-Viedma, E & Fujita, H 2021, 'Enhancing Big Data Feature Selection Using a Hybrid Correlation-Based Feature Selection. Electronics, vol. 10. 85. Moore, JH & White, BC 2007, 'Tuning Relief for genome-wide genetic analysis', European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics, Springer, pp. 166-175.
172 86. Motsch, W, David, A, Sivalingam, K, Wagner, A & Ruskowski, M 2020, 'Approach for Dynamic Price-Based Demand Side Management in Cyber-Physical Production Systems,' Procedia Manufacturing, vol. 51, pp. 1748-1754. 87. Mursalin, M, Zhang, Y, Chen,Y & Chawla, NV 2017, 'Automated epileptic seizure detection using improved correlation-based feature selection with random forest classifier,' Neuro computing, vol. 241, pp. 204-214. 88. Nguyen, XH, Nguyen, XD, Huynh, HH & Le, KH 2022, 'Realguard: A Lightweight Network Intrusion Detection System for IoT Gateways', Sensors, vol. 22. 89. Otoum, Y & Nayak, A 2021, 'AS-IDS: Anomaly and Signature Based IDS for the Internet of Things',
Journal of Network and Systems Management, vol. 29. 90. Pandey, SK 2019, 'Design and performance analysis of various feature selection methods for anomaly-based techniques in intrusion detection system,' Security and Privacy, vol. 2, no. e56, pp. 1–14. 91.
Panigrahi,
R & Borah, S 2018, '
A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems',
in
Proceeding of 2018International Journal of Engineering and Technology, vol. 7, no. 3, pp. 479-482. 92. Papamartzivanos, D, Mármol, FG & Kambourakis, G 2019, 'Introducing Deep Learning Self-Adaptive Misuse
Network Intrusion Detection Systems,' IEEE Access,
vol. 7, pp. 13546-13560. 93.
Paulauskas, N & Auskalnis,
J 2017, 'Analysis of data pre-processing influence on intrusion detection using nsl-kdd dataset,' in 2017
Open Conference of Electrical, Electronic and Information Sciences (eStream),
IEEE,
pp. 1-5. 94.
Pekaric, I, Sauerwein, C, Haselwanter, S & Felderer, M 2021, 'A taxonomy of attack mechanisms in the automotive domain,' Computer Standards & Interfaces, vol. 78. 95.
Peng, H, Long, F, Ding, C, 2005, 'Feature selection based on mutual information criteria of max-dependency, max-relevance, and min- redundancy', IEEE
Transaction on Pattern Analysis and Machine Intelligence, 2005, vol. 27, pp. 226-1238.
173 96. Pirgazi, J, Alimoradi, M, Abharian, T & Olyaee, MH, 2019, 'An Efficient hybrid filter-wrapper metaheuristic-based gene selection method for high dimensional datasets', Scientific Reports, vol. 9. 97. Priya, S & Uthra, RA, 2021, 'Comprehensive analysis for class imbalance data with concept drift using ensemble based classification', Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 4943–4956. 98.
Rajagopal, S, Kundapur, PP & Hareesha, KS 2020, 'A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets', Security and Communication Networks, vol. 2020. 99.

Ran, J, Ji, Y & Tang, B, 2019, 'A Semi-Supervised Learning Approach to IEEE 802.11 Network Anomaly Detection', In Proceedings of the 2019 IEEE 89th Vehicular Technology Conference (VTC2019- Spring), Kuala Lumpur, Malaysia, pp. 1-5. 100. Ren, J, Guo, J, Qian, W, Yuan, H, Hao, X & Jingjing, H 2020, 'Building an effective intrusion detection system by using hybrid data optimization based on machine learning algorithms,' Security and Communication Networks, vol. 2020. 101.
Rizwan, M, Shabbir, A, Javed, AR, Srivastava, G, Gadekallu, TR, Shabir, M & Hassan, MA 2022, 'Risk monitoring strategy for confidentiality of healthcare information,' Computers and Electrical Engineering, vol. 100. 102.
Robnik, M & Konenko, I 2003, 'Theoretical and empirical analysis of ReliefF and RReliefF',
Machine Learning, vol. 53, pp. 23-69. 103. Ryu, J, Kantardzic, M & Walgampaya, C 2010, 'Ensemble Classifier based on Misclassified Streaming Data', In Proceedings of the 10th IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria. 104. Saeys, Y, Inza, I & Larrañaga, P 2007, 'A review of feature selection techniques in bioinformatics,' Bioinformatics, vol. 23, no. 19, pp. 2507-2517. 105. Saranya, T, Sridevi, S, Deisy, C, Chung, TD & Khan, MKA 2020, 'Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review,' Procedia Computer Science, vol. 171, pp. 1251-1260.
174 106. Sarker, IH, Kayes, ASM,
Badsha, S & Alqahtani, H 2020, 'Watters, P.; Ng, A. Cybersecurity data science: An overview from machine learning perspective',
Journal of Big Data, vol. 7, pp. 1-29. 107. Shah, RA, Qian, Y, Kumar, D, Ali, M & Alvi, MB 2017, 'Network intrusion detection through discriminative feature selection by using sparse logistic regression', Future Internet,
vol. 9,
pp. 81-92. 108.
Sharafaldin, I, Lashkari, AH &
Ghorbani, AA 2018, '
Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization',
In
Proceedings of the
ICISSP 2018, Madeira, Portugal, pp. 108–116. 109.
Sharma, J, Giri, C & Granmo, OC 2019, 'Multi-layer intrusion detection system with Extra Trees feature selection, extreme learning machine ensemble, and softmax aggregation,' EURASIP Journal on Information Security, vol. 15 pp. 22-38. 110.
Singh, S &
Singh, AK 2018, 'Detection of spam using particle swarm optimisation in feature selection', Pertanika Journal of Science & Technology,
vol. 26, no. 3, pp. 150-168. 111.
Singh,
UK, Joshi, C & Kanellopoulos, D 2019, 'A framework for zero- day vulnerabilities detection and prioritization,' Journal of Information Security and Applications, vol. 46, pp. 164-172. 112. Soe, YN, Feng, Y, Santosa, PI, Hartanto, R & Sakurai, K 2020, 'Machine Learning-Based IoT-Botnet Attack Detection with Sequential Architecture, Sensors, vol. 20, no. 16. 113.
Soliman, HH, Hikal, NA, Sakr, NA 2012, 'A comparative performance evaluation of intrusion detection techniques for hierarchical wireless sensor networks,' Egyptian Informatics Journal, vol. 13, no. 3, pp. 225-238. 114.
Song, L, Smola, A, Gretton, A, Borgwardt, KM & Bedo, J 2007, 'Supervised feature selection via dependence estimation', In Proceedings of the 24th International Conference on Machine learning,
New York, NY, USA, pp. 823-830. 115.
Song,
Q, Guo, Y & Shepperd, M 2019, 'A Comprehensive Investigation of the Role of Imbalanced Learning for Software Defect Prediction,' IEEE Transactions on Software
Engineering,
vol. 45, no. 12,
pp. 1253-1269.
175 116.
Tavallaee, M, Bagheri, E, Lu, W & Ghorbani, AA 2009, 'A detailed analysis of the KDD cup 99 data set,'
IEEE
Symposium on Computational Intelligence for Security and Defense Applications,
pp. 1-6. 117.
Tiwari, R, Pratap, M & Singh, K 2010, 'Correlation-based Attribute Selection
using Genetic Algorithm', International Journal of Computer Applications, vol. 4, no. 8, pp. 45-58. 118.
Trautman, LJ & Ormerod, PC, 2017, 'Corporate Dir ate Directors' and Officers' Cybersecurity Standar ors' and Officers' Cybersecurity Standard of Car d of Care: The Yahoo Data Breach', American University Law Review, vol. 66, no. 5. 119.

Umar, MA,
Zhanfang, C & Liu, Y 2021, 'Network Intrusion Detection Using Wrapper-based Decision Tree for Feature Selection', Future Internet, vol. 13, no. 11, pp. 5-13. 120. Urbanowicz, RJ, Meeker, M, Cava, WL, Olson, RS & Moore, JH 2018, 'Relief-based feature selection: Introduction and review', Journal of Biomedical Informatics, vol. 85, pp. 189-203. 121. Vaiyapuri, T & Binbusayyis, A, 2020, 'Application of deep autoencoder as a one-class classifier for unsupervised network intrusion detection: A comparative evaluation', PeerJ
Computer. Science, vol. 6. 122. Wang, CR, Xu,RF, Lee, SJ & Lee, CH 2018, '
Network intrusion detection using equality constrained-optimization-based extreme learning machines, Knowledge-Based
Systems, vol. 147, pp. 68-80. 123. Wang, S &
Yao,
X 2012, 'Multiclass Imbalance Problems: Analysis and Potential Solutions,' IEEE Transactions on Systems, Man, and Cybernetics,
vol. 42, no. 4, pp. 1119-1130. 124.
Wang, Y, Liu, Y & Li, J 2019, 'Deducing cascading failures caused by cyberattacks based on attack gains and cost principle in cyber-physical power systems', Journal of Modern Power Systems and Clean Energy, vol. 7, pp. 1450-1460. 125. Wathiq, L, Othman, ZA, Nazri, MZA 2015, 'Hybrid Modified -Means with C4.5 for Intrusion Detection Systems in Multiagent Systems', The Scientific World Journal, vol. 2015.
176 126. Yaacoub, JA, Salman, O, Noura, HN, Kaaniche, N, Chehab, A & Malli M 2020, 'Cyber-physical systems security: Limitations, issues and future trends', Microprocessors Microsystem, vol. 77. 127. Yang, XS & He,
X 2013, 'Bat algorithm: literature review and applications,' International Journal of Bio-Inspired Computation, vol. 5, no. 3, pp. 141-149 128.
Yang, XS 2010, 'A new metaheuristic bat-inspired algorithm,' in Nature inspired cooperative strategies for optimization (NICSO 2010). Springer, pp. 65-74. 129. Yang, XS 2014, 'Nature-inspired optimization algorithms',
Elsevier,
vol. 4,
pp. 52-68 130. Zainal, A, Maarof, AM, Shamsuddin, SM, 'Ensemble classifiers for network intrusion detection system', Journal of Information Assurance and Security, vol. 4, pp. 217-225. 131. Zhai, Y,
Wang, SP, Ma, N, Yang, BR & Zhang DZ 2014, 'A data mining method for imbalanced datasets based on one-sided link and distribution density of instances', Acta Electonica Sinica, vol. 42, no. 7. 132. Zhao, Z, Gong, D, Lu, B, Liu, F & Zhang, C, 2016, 'SDN-Based Double Hopping Communication against Sniffer Attack', Mathematical Problems in Engineering, vol. 2016. 133.
Zhong, Y, Chen, W, Wang, Z, Chen, Y, Wang, K, Li, Y, Yin, X, Shi, X & Yang, J 2020, 'HELAD: A novel network anomaly detection model based on heterogeneous ensemble learning',
Computer Networks, vol. 169. 134. Zhou, J, Ma, C, Long, D, Xu, G, Ding, N, Zhang, H, Xie, P & Liu, G 2020, 'Hierarchy-Aware Global Model for Hierarchical Text Classification', In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, vol. 3,
pp. 1106-1117. 135.
Zhou, Y, Cheng, G, Jiang, S & Dai, M, 2020, '
Building
an efficient intrusion detection system based on feature selection and ensemble classifier',
Computer Networks,
vol. 174. 136.
Zhou, Y,
Mazzuchi, TA & Sarkani, S 2020, 'M-AdaBoost-A based ensemble system for network intrusion detection,' Expert Systems with Applications, vol. 162.
177 137. Zhou, ZH 2009, '
Ensemble Learning', In Encyclopedia of Biometrics; Li, S.Z., Jain, A., Eds.; Springer: Boston, MA, USA, pp. 270-273.
LIST OF PUBLICATIONS International Journal 1. Anand Babu, R & Kannan, S, 2022, '
Bat-Inspired Optimization for Intrusion Detection Using an Ensemble Forecasting Method,'
Intelligent Automation & Soft Computing, vol. 34, no. 1, pp. 307-323, Impact Factor : 1.647.

94% **MATCHING BLOCK 1/11** SA Formatted paper.pdf (D134882241)

It works by creating several DTs. It accepts a large number of input parameters without variable exclusion and categorizes them based on their reputation. More specifically, RF employs a group of classification trees. Each tree in the forest provides with a vote for the most recurrent class in input records. RF classifier considers only fewer parameters as compared with the other machine learning methods such as ANN, SVM, etc.. In this classifier, a group of single tree classifiers can be denoted by the following Equation (5.14). { (,) = 1,2,3.... ...} (5.14) We refer to as a function of RF classification. In the above equation, { } represents random vectors and each tree has a prediction (vote) for the most popular class at input variable . The characteristic and size of depend on its utilization. The key to the success of this classifier is the formation of the forest. In order to train each tree in the forest, the RF classifier built a bootstrapped subset of the training process. Hence, each tree utilizes approximately 2/3 of the training dataset. The idle instances are known as the out-of-bag instances which employ in inner cross-validation process to calculate the classification accuracy. Furthermore, RF has the minimum computational overhead, and it is oblivious to the outliers and parameters. Also, the over-fitting issue is less as related to single decision tree-based approaches and it is not required to prune the tree which is a difficult and time-consuming process.

95% **MATCHING BLOCK 2/11** SA Formatted paper.pdf (D134882241)

weight allocation and weight augmentation policies are considered to preserve accuracy and strong diversity. BF will arbitrarily calculate the weights for features that present in the most recent tree. The weight range (W) can be calculated as = { [0.00, − 1], = 1 [− 1 −1 + , − 1], &lt; 1 } (5.21) where denotes the attribute level and the parameter is employed to guarantee the weight range for various levels be non-overlying. When the feature presents in the root node, we select the value for equal to 1. When the feature appears at a child, we select = 2. Likewise, to describe the adverse impact of holding weights that does not exist in the most recent tree, BF has a technique to progressively improve the attribute weights. Consider an attribute is verified at level of the −1 th tree with height h and its weight is . The increment of weight is estimated as = 1− (h+1)− (5.22) The procedure of

100% **MATCHING BLOCK 3/11** SA Formatted paper.pdf (D134882241)

num_compromised 14 root_shell 15 su_atempted 16 num_root 17 num_file_creations 18 num_shells 19 num_access_files 20 num_outbound_cmds 21

87% **MATCHING BLOCK 4/11** SA Anand Babu R-1-18194591230.docx (D135638144)

src_bytes 6 dst_bytes 7 land 8 wrong_fragment 9 urgent 10 hot 11 num_failed_login 12

82% **MATCHING BLOCK 5/11** SA Anand Babu R-1-18194591230.docx (D135638144)

count 24 srv_count 25 serror_rate 26 srv_serror_rate 27 rerror_rate 28 srv_rerror_rate 29 same_srv_rate 30 diff_srv_rate 31 srv_diff_host_rate 32 dst_host_count 33 dst_host_same_count 34 dst_host_diff_srv_rate 35 dst_host_diff_srv_rat e 36 dst_host_same_src_port_ rate 37 dst_host_srv_diff_host_ rate 38 dst_host_serror_rate 39 dst_host_srv_serror_rate 40 dst_host_rerror_rate 41 dst_host_srv_rerror_

75% **MATCHING BLOCK 6/11** W https://libgen.ggfwzs.net/book/74240522/e250fe

Flow Duration 9 Total Fwd Packets 10 Total Backward Packets 11 Total Length of Fwd Packets 12 Total Length of Bwd Packets 13

60%	MATCHING BLOCK 7/11	W	https://downloads.hindawi.com/journals/scn/202 ...

Fwd Packet Length Max 14 Fwd Packet Length Min 15 Fwd Packet Length Mean 16 Fwd Packet Length Std 17 Bwd Packet Length Max 18 Bwd Packet Length Min 19 Bwd Packet Length Mean 20 Bwd Packet Length Std 21 Flow Bytes/s 22 Flow Packets/s 23 Flow IAT Mean 24 Flow IAT Std 25 Flow IAT Max 26 Flow IAT Min 27 Fwd IAT Total 28 Fwd IAT Mean 29 Fwd IAT Std 30 Fwd IAT Max 31 Fwd IAT Min 32 Bwd IAT Total 33 Bwd IAT Mean 34 Bwd IAT Std 35 Bwd IAT Max 36 Bwd IAT Min 37 Fwd PSH Flags 38

74%	MATCHING BLOCK 8/11	W	https://downloads.hindawi.com/journals/scn/202 ...

Min Packet Length 46 Max Packet Length 47 Packet Length Mean 48 Packet Length Std 49 Packet Length Variance 50 FIN Flag Count 51 SYN Flag Count 52 RST Flag Count 53 PSH Flag Count 54 ACK Flag Count 55 URG Flag Count 56 CWE Flag Count 57 ECE Flag Count 58 Down/Up Ratio 59 Average Packet Size 60 Avg Fwd Segment Size 61 Avg Bwd Segment Size 62

100%	MATCHING BLOCK 9/11	W	https://content.iospress.com/articles/informat ...

Fwd Avg Bytes/Bulk 64 Fwd Avg Packets / Bulk 65 Fwd Avg Bulk Rate 66 Bwd Avg Bytes/ Bulk 67 Bwd Avg Packets /Bulk 68 Bwd Avg Bulk Rate 69

88%	MATCHING BLOCK 10/11	SA	Anand Babu R-1-18194591230.docx (D135638144)

Forest PA: Constructing a decision forest by penalizing attributes used in previous trees', Expert System Application, vol. 89,

80%	MATCHING BLOCK 11/11	SA	Paper Published Update 3-converted.pdf (D53209344)

Bytes 73 Init Win bytes forward 74 Init Win bytes backward 75 act data pkt fwd 76 min seg size forward 77 Active Mean 78 Active Std 79 Active Max 80 Active Min 81 Idle Mean 82 Idle Std 83 Idle

Hit and source - focused comparison, Side by Side

Submitted text	As student entered the text in the submitted document.
Matching text	As the text appears in the source.

1/11 **SUBMITTED TEXT** 297 WORDS **94%** **MATCHING TEXT** 297 WORDS

It works by creating several DTs. It accepts a large number of input parameters without variable exclusion and categorizes them based on their reputation. More specifically, RF employs a group of classification trees. Each tree in the forest provides with a vote for the most recurrent class in input records. RF classifier considers only fewer parameters as compared with the other machine learning methods such as ANN, SVM, etc.. In this classifier, a group of single tree classifiers can be denoted by the following Equation (5.14). { (,) = 1,2,3.... ...} (5.14) We refer to as a function of RF classification. In the above equation, { } represents random vectors and each tree has a prediction (vote) for the most popular class at input variable . The characteristic and size of depend on its utilization. The key to the success of this classifier is the formation of the forest. In order to train each tree in the forest, the RF classifier built a bootstrapped subset of the training process. Hence, each tree utilizes approximately 2/3 of the training dataset. The idle instances are known as the out-of-bag instances which employ in inner cross-validation process to calculate the classification accuracy. Furthermore, RF has the minimum computational overhead, and it is oblivious to the outliers and parameters. Also, the over-fitting issue is less as related to single decision tree-based approaches and it is not required to prune the tree which is a difficult and time-consuming process.

It works by creating several decision It accepts a large number of input parameters without variable exclusion and categorizes them according their reputation. More specifically, RF employs a group of classification trees. Each tree in the forest provides with a vote for the most recurrent class in input records. RF classifier considers only fewer parameters as compared with the other machine learning methods (e.g., artificial neural network, support vector etc.). In this classifier, a group of single tree classifiers can be denoted by the following equation. { (,) = 1,2,3.... ...} (11) We refer to as a function of RF classification. In the above equation, { } represents random vectors and each tree has a prediction (vote) for the most popular class at input variable . The characteristic and size of depend on its utilization. The key to the success of this classifier is the formation of the forest. In order to train each tree in the forest, the RF classifier built a bootstrapped subset of the training process. Hence, each tree utilizes approximately 2/3 of the training dataset. The idle instances are known as the out-of-bag instances which employ in inner cross-validation process to calculate the classification accuracy. Furthermore, RF has the minimum computational overhead, and it is oblivious to the outliers and parameters. Also, the over-fitting issue is less as related to single decision tree-based approaches and it is not required to prune the tree which is a difficult and time-consuming process [33].

SA Formatted paper.pdf (D134882241)

2/11	SUBMITTED TEXT	296 WORDS	95% MATCHING TEXT	296 WORDS
	weight allocation and weight augmentation policies are considered to preserve accuracy and strong diversity. BF will arbitrarily calculate the weights for features that present in the most recent tree. The weight range (W) can be calculated as = { [0.00, – 1], = 1 [– 1 –1 + , – 1], &lt; 1 } (5.21) where denotes the attribute level and the parameter is employed to guarantee the weight range for various levels be non-overlying. When the feature presents in the root node, we select the value for equal to 1. When the feature appears at a child, we select = 2. Likewise, to describe the adverse impact of holding weights that does not exist in the most recent tree, BF has a technique to progressively improve the attribute weights. Consider an attribute is verified at level of the −1 th tree with height h and its weight is . The increment of weight is estimated as = 1– (h+1)– (5.22) The procedure of		weight allocation and weight augmentation policies are considered to preserve the accuracy and strong diversity. FPA will arbitrarily calculate the weights for features that present in the most recent tree. The weight range (W) can be calculated as = { [0.00, – 1], = 1 [– 1 –1 + , – 1], &lt; 1 } (12) where denotes the attribute level and the parameter is employed to guarantee the weight range for various levels be non-overlying. When the feature presents in the root node, we select the value for equal to 1. When the feature appears at a child, we select = 2. Likewise, to describe the adverse impact of holding weights that does not exist in the most recent tree, FPA has a technique to progressively improve the attribute weights. Consider an attribute is verified at level of the −1 th tree with height h and its weight is . The increment of weight is estimated h + 1) – (13) Mechanism: The estimation of	

SA Formatted paper.pdf (D134882241)

3/11	SUBMITTED TEXT	17 WORDS	100% MATCHING TEXT	17 WORDS
	num_compromised 14 root_shell 15 su_atempted 16 num_root 17 num_file_creations 18 num_shells 19 num_access_files 20 num_outbound_cmds 21		num_compromised, root_shell, su_atempted, num_root, num_file_creations, num_shells, num_access_files, num_outbound_cmds,	

SA Formatted paper.pdf (D134882241)

4/11	SUBMITTED TEXT	15 WORDS	87% MATCHING TEXT	15 WORDS
	src_bytes 6 dst_bytes 7 land 8 wrong_fragment 9 urgent 10 hot 11 num_failed_login 12		src_bytes, dst_bytes, wrong_fragment, urgent, hot, num_failed_login,	

SA Anand Babu R-1-18194591230.docx (D135638144)

5/11 SUBMITTED TEXT 37 WORDS	82% MATCHING TEXT 37 WORDS
count 24 srv_count 25 serror_rate 26 srv_serror_rate 27 rerror_rate 28 srv_rerror_rate 29 same_srv_rate 30 diff_srv_rate 31 srv_diff_host_rate 32 dst_host_count 33 dst_host_same_count 34 dst_host_diff_srv_rate 35 dst_host_diff_srv_rat e 36 dst_host_same_src_port_ rate 37 dst_host_srv_diff_host_ rate 38 dst_host_serror_rate 39 dst_host_srv_serror_rate 40 dst_host_rerror_rate 41 dst_host_srv_rerror_	count, srv_count, serror_rate, srv_serror_rate, rerror_rate, srv_rerror_rate, same_srv_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_count, dst_host_diff_src_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_srv_serror_rate, dst_host_rerror_rate, dst_host_srv_rerror_

SA Anand Babu R-1-18194591230.docx (D135638144)

6/11 SUBMITTED TEXT 24 WORDS	75% MATCHING TEXT 24 WORDS
Flow Duration 9 Total Fwd Packets 10 Total Backward Packets 11 Total Length of Fwd Packets 12 Total Length of Bwd Packets 13	flow duration, total forward packets, total backward packets, total length of forward packets, total length of backward packets,

W https://libgen.ggfwzs.net/book/74240522/e250fe

7/11 SUBMITTED TEXT 107 WORDS	60% MATCHING TEXT 107 WORDS
Fwd Packet Length Max 14 Fwd Packet Length Min 15 Fwd Packet Length Mean 16 Fwd Packet Length Std 17 Bwd Packet Length Max 18 Bwd Packet Length Min 19 Bwd Packet Length Mean 20 Bwd Packet Length Std 21 Flow Bytes/s 22 Flow Packets/s 23 Flow IAT Mean 24 Flow IAT Std 25 Flow IAT Max 26 Flow IAT Min 27 Fwd IAT Total 28 Fwd IAT Mean 29 Fwd IAT Std 30 Fwd IAT Max 31 Fwd IAT Min 32 Bwd IAT Total 33 Bwd IAT Mean 34 Bwd IAT Std 35 Bwd IAT Max 36 Bwd IAT Min 37 Fwd PSH Flags 38	Fwd Packet Length Max, Fwd Packet Length Min, Fwd Packet Length Mean, Bwd Packet Length Max, Bwd Packet Length Min, Bwd Packet Length Mean, Bwd Packet Length Std, Flow Packets/s, Flow IAT Mean, Flow IAT Std, Flow IAT Max, Fwd IAT Total, Fwd IAT Mean, Fwd IAT Std, Fwd IAT Max, Bwd IAT Std, Bwd IAT Max, Fwd PSH Flags,

W https://downloads.hindawi.com/journals/scn/2021/9947059.pdf

8/11 SUBMITTED TEXT 71 WORDS	74% MATCHING TEXT 71 WORDS
Min Packet Length 46 Max Packet Length 47 Packet Length Mean 48 Packet Length Std 49 Packet Length Variance 50 FIN Flag Count 51 SYN Flag Count 52 RST Flag Count 53 PSH Flag Count 54 ACK Flag Count 55 URG Flag Count 56 CWE Flag Count 57 ECE Flag Count 58 Down/Up Ratio 59 Average Packet Size 60 Avg Fwd Segment Size 61 Avg Bwd Segment Size 62	Min Packet Length, Max Packet Length, Packet Length Mean, Packet Length Std, Packet Length Variance, FIN Flag Count, SYN Flag Count, PSH Flag Count, ACK Flag Count, URG Flag Count, Down/Up Ratio, Average Packet Size, Avg Fwd Segment Size, Avg Bwd Segment Size,

W https://downloads.hindawi.com/journals/scn/2021/9947059.pdf

9/11 SUBMITTED TEXT 31 WORDS	100% MATCHING TEXT 31 WORDS
Fwd Avg Bytes/Bulk 64 Fwd Avg Packets / Bulk 65 Fwd Avg Bulk Rate 66 Bwd Avg Bytes/ Bulk 67 Bwd Avg Packets /Bulk 68 Bwd Avg Bulk Rate 69	Fwd Avg Bytes/Bulk', 'Fwd Avg Packets/Bulk', 'Fwd Avg Bulk Rate', 'Bwd Avg Bytes/Bulk', 'Bwd Avg Packets/Bulk', 'Bwd Avg Bulk Rate',
W https://content.iospress.com/articles/informatica/infor457	

10/11 SUBMITTED TEXT 19 WORDS	88% MATCHING TEXT 19 WORDS
Forest PA: Constructing a decision forest by penalizing attributes used in previous trees', Expert System Application, vol. 89,	Forest pa: Constructing a decision forest by penalizing attributes used in previous trees," Expert Systems with vol. 89,
SA Anand Babu R-1-18194591230.docx (D135638144)	

11/11 SUBMITTED TEXT 42 WORDS	80% MATCHING TEXT 42 WORDS
Bytes 73 Init Win bytes forward 74 Init Win bytes backward 75 act data pkt fwd 76 min seg size forward 77 Active Mean 78 Active Std 79 Active Max 80 Active Min 81 Idle Mean 82 Idle Std 83 Idle	
SA Paper Published Update 3-converted.pdf (D53209344)	