

# Malware Analysis Using Artificial Intelligence and Deep Learning

Mark Stamp · Mamoun Alazab ·  
Andrii Shalaginov  
Editors

# Malware Analysis Using Artificial Intelligence and Deep Learning

 Springer

*Editors*

Mark Stamp  
Department of Computer Science  
San Jose State University  
San Jose, CA, USA

Mamoun Alazab   
College of Engineering, IT & Environment  
Charles Darwin University  
Darwin, NT, Australia

Andrii Shalaginov  
Faculty of Information Technology  
and Electrical Engineering  
Norwegian University of Science  
and Technology  
Gjøvik, Norway

ISBN 978-3-030-62581-8      ISBN 978-3-030-62582-5 (eBook)  
<https://doi.org/10.1007/978-3-030-62582-5>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG  
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

# Preface

Artificial intelligence (AI) is changing the world as we know it. From its humble beginnings in the late 1940s as little more than an academic curiosity, AI has gone through multiple boom and bust cycles. With recent advances in machine learning (ML) and deep learning (DL), AI has finally taken root as a fundamental transformative technology. The changes wrought by AI already affect virtually every aspect of daily life, yet we are clearly only in the early stages of an AI-based revolution.

In the field of information security, there is no topic that is more significant than malware. The sheer volume of malware and the cost of dealing with its consequences are truly staggering. It is therefore timely to consider ML, DL, and AI in the context of malware analysis.

The chapters in this book apply numerous cutting-edge AI techniques to a wide variety of challenging problems in the malware domain. The book includes no less than 8 survey articles, which can serve to bring a reader quickly up to speed with the current state of the art. The heart of the book consists of 11 chapters that are tightly focused on AI-based techniques for malware analysis. We have also included 6 chapters where AI is applied to information security topics that are not strictly malware, but are closely related.

We are confident that this book will prove equally valuable to practitioners working in the trenches and to researchers at all levels. New and novel techniques as well as clever applications abound, yet we have strived to make the material accessible to the widest possible audience. It is our fervent hope—and firm belief—that the tools and techniques presented in the chapters of this book will play a major role in taming the malware threat.

San Jose, USA  
Darwin, Australia  
Gjøvik, Norway  
December 2020

Mark Stamp  
Mamoun Alazab  
Andrii Shalaginov

# Contents

## Surveys

<b>A Selective Survey of Deep Learning Techniques and Their Application to Malware Analysis</b> . . . . .	3
Mark Stamp	
1 Introduction . . . . .	3
2 A Brief History of ANNs . . . . .	4
3 Why Deep Learning? . . . . .	8
4 Decisions, Decisions . . . . .	9
5 Multilayer Perceptrons . . . . .	10
5.1 Overview of MLPs . . . . .	10
5.2 MLPs in Malware Analysis . . . . .	11
6 Convolutional Neural Networks . . . . .	12
6.1 Overview of CNNs . . . . .	12
6.2 Convolutions and CNNs . . . . .	12
6.3 Example CNN . . . . .	14
6.4 CNNs in Malware Analysis . . . . .	18
7 Recurrent Neural Networks . . . . .	19
7.1 Backpropagation Through Time . . . . .	22
7.2 Long Short-Term Memory . . . . .	24
7.3 Gated Recurrent Units . . . . .	27
7.4 Recursive Neural Network . . . . .	29
7.5 Last Word on RNNs . . . . .	29
7.6 RNNs in Malware Analysis . . . . .	30
8 Residual Networks . . . . .	30
8.1 ResNet in Malware Analysis . . . . .	32
9 Generative Adversarial Network . . . . .	32
9.1 GANs in Malware Analysis . . . . .	36

- 10 Extreme Learning Machines . . . . . 36
  - 10.1 ELMs in Malware Analysis . . . . . 39
- 11 Word Embedding Techniques . . . . . 39
  - 11.1 HMM2Vec . . . . . 40
  - 11.2 PCA2Vec . . . . . 42
  - 11.3 Word2Vec . . . . . 43
  - 11.4 Word Embeddings in Malware Analysis . . . . . 46
- 12 Conclusion . . . . . 46
- References . . . . . 46

**Malware Detection with Sequence-Based Machine Learning and Deep Learning . . . . . 53**

William B. Andreopoulos

- 1 Introduction . . . . . 53
- 2 Data Extraction . . . . . 54
  - 2.1 Static Data . . . . . 54
  - 2.2 Dynamic Data . . . . . 55
  - 2.3 Hybrid Analysis . . . . . 56
  - 2.4 Alternative Approaches That Use Raw Data . . . . . 56
  - 2.5 Evaluation of Malware Detection Accuracy . . . . . 56
- 3 Recent Research Examples . . . . . 57
  - 3.1 Hybrid Analysis . . . . . 58
- 4 HMM Architecture . . . . . 60
  - 4.1 Training for Malware Detection . . . . . 61
  - 4.2 Metamorphic Malware Detection . . . . . 61
- 5 LSTM Architecture . . . . . 61
  - 5.1 LSTM Training . . . . . 62
- 6 Tools . . . . . 65
  - 6.1 IDA Pro . . . . . 65
  - 6.2 OllyDbg . . . . . 65
  - 6.3 Ether . . . . . 65
  - 6.4 API Logger . . . . . 66
  - 6.5 WinAPIOverride . . . . . 67
  - 6.6 API Monitor . . . . . 67
  - 6.7 BSA . . . . . 67
- 7 Conclusion . . . . . 67
- References . . . . . 68

**Review of the Malware Categorization in the Era of Changing Cybethreats Landscape: Common Approaches, Challenges and Future Needs . . . . . 71**

Andrii Shalaginov, Geir Olav Dyrkolbotn, and Mamoun Alazab

- 1 Introduction . . . . . 72
- 2 Background: From Malware Developers to Malware Analysts . . . . . 73

- 2.1 Severity of Malware Infection and Modus Operandi . . . . . 74
- 2.2 Detection and Approach Strategy . . . . . 75
- 2.3 Preliminary Analysis and Dissection . . . . . 76
- 2.4 Malware Categorization and Cybersecurity Awareness . . . . . 76
- 3 Malware Classification: State of the Art . . . . . 77
  - 3.1 Characteristics-Based Detection for Multinomial Classification . . . . . 78
  - 3.2 Commonly Used Malware Naming . . . . . 79
  - 3.3 Auxiliary Software Tools and Research Datasets . . . . . 79
- 4 Analysis of Community—and Commercially—Accepted Malware Taxonomies . . . . . 80
  - 4.1 Overall Software Category . . . . . 80
  - 4.2 Risk Level/Threat Level . . . . . 81
  - 4.3 Malware Targets/Platforms/Operating Systems . . . . . 81
  - 4.4 Malware Type/General Categories . . . . . 83
  - 4.5 Malware Family/Functionality-Specific Categories . . . . . 84
  - 4.6 System and Digital Forensic-Related Artefacts . . . . . 84
  - 4.7 Malware Variants . . . . . 85
  - 4.8 Malware Name Suffix . . . . . 85
  - 4.9 Binary Compilation Timestamps/Timeline . . . . . 85
  - 4.10 Country/Adversarial Groups Origins . . . . . 86
- 5 Review of the Existing Anti-virus Naming Schemes . . . . . 86
  - 5.1 Computer Antivirus Research Organization (CARO) . . . . . 86
  - 5.2 Common Malware Enumeration (CME) . . . . . 87
  - 5.3 Malware Attribute Enumeration and Characterization (MAEC) . . . . . 88
  - 5.4 Malware Information Sharing Platform (MISP) . . . . . 88
- 6 Analysis of Existing Approaches to Malware Categorization . . . . . 88
- 7 Practical Implications of Malware Naming in the Light of Big Data . . . . . 90
- 8 Discussions and Conclusions . . . . . 92
- References . . . . . 93

**Addressing Malware Attacks on Connected and Autonomous Vehicles: Recent Techniques and Challenges . . . . . 97**

Aiman Al-Sabaawi, Khamael Al-Dulaimi, Ernest Foo, and Mamoun Alazab

- 1 Introduction . . . . . 98
  - 1.1 Important Technologies in Intelligent Transportation System in Smart Cities . . . . . 99
  - 1.2 Benefits of Intelligent Transportation System . . . . . 99
  - 1.3 Challenges of Intelligent Transportation System . . . . . 100

2	Literature Review . . . . .	101
2.1	Attack . . . . .	105
2.2	Defence . . . . .	112
2.3	Detect . . . . .	114
3	Recent Techniques and Challenges . . . . .	115
4	Conclusion . . . . .	116
	References . . . . .	116
	<b>A Survey of Intelligent Techniques for Android Malware Detection . . .</b>	<b>121</b>
	Rajesh Kumars, Mamoun Alazab, and WenYong Wang	
1	Introduction . . . . .	121
2	Static, Dynamic, and Hybrid Analysis of Android Malware	
	Background . . . . .	123
2.1	Static Analysis . . . . .	123
2.2	Dynamic Analysis . . . . .	129
2.3	Hybrid Analysis . . . . .	134
2.4	A Comparison of Static, Dynamic, and Hybrid Analysis . . . . .	139
3	Android Malware Detection Approaches . . . . .	140
3.1	Basic Proposed Framework to Detect Android Malware . . . . .	140
3.2	Basic Proposed Algorithms for Android Malware Features . . . . .	143
3.3	Feature Selection-Based Algorithms . . . . .	147
3.4	Association Rule-Based Algorithms . . . . .	147
3.5	Model Evaluation Measures . . . . .	148
4	Experimental Analysis and Dataset Discussion . . . . .	149
4.1	Publicly Available Most Popular Dataset . . . . .	149
4.2	Dataset Other Research Work . . . . .	149
5	Experimental Analysis . . . . .	151
5.1	Permission-Based Experimental Analysis . . . . .	151
5.2	Clustering-Based Experimental Analysis . . . . .	154
5.3	Classification Experimental Analysis . . . . .	154
6	Additional Challenges of Android Malware Detection . . . . .	155
7	Conclusion . . . . .	157
	References . . . . .	157
	<b>Deep Learning in Malware Identification and Classification . . . . .</b>	<b>163</b>
	Balram Yadav and Sanjiv Tokekar	
1	Malware and Malware Analysis . . . . .	164
1.1	Malware . . . . .	164
1.2	Malware Analysis . . . . .	168
1.3	Malware Classification . . . . .	171
2	Deep Learning . . . . .	175
2.1	What is Deep Learning? . . . . .	175
2.2	Deep Learning Algorithms . . . . .	177
2.3	Steps for Building a Deep Learning Model . . . . .	181



- 3 Malware Classification Based on Malware Visualization and Deep Learning . . . . . 183
  - 3.1 Related Work: Recent Innovations in Malware Classification Using Deep Learning and Visualization . . . . . 184
  - 3.2 Performance Metrics: To Measure the Performance of the Deep Learning Model . . . . . 191
  - 3.3 A Practical Implementation of Malware Classification Using CNN and Malware Image Visualization . . . . . 193
- 4 Challenges and Open Issues . . . . . 202
- 5 Conclusion . . . . . 203
- References . . . . . 204

**Review of Artificial Intelligence Cyber Threat Assessment Techniques for Increased System Survivability . . . . . 207**

Nikolaos Doukas, Peter Stavroulakis, and Nikolaos Bardis

- 1 Introduction . . . . . 208
- 2 AI Support to Survivability . . . . . 209
  - 2.1 Security Threat Detection for Preventive Survivability . . . . . 210
  - 2.2 Email Message Filtering by Linear Classifiers . . . . . 211
  - 2.3 Malware Detection . . . . . 212
  - 2.4 Collusion Attacks . . . . . 213
  - 2.5 Anomaly Detection . . . . . 214
  - 2.6 Dynamic Analysis of Malware . . . . . 216
- 3 Cooperative Infrastructure Defense . . . . . 217
- 4 Post Attack Survivability . . . . . 219
- 5 Conclusions . . . . . 221
- References . . . . . 221

**On Ensemble Learning . . . . . 223**

Mark Stamp, Aniket Chandak, Gavin Wong, and Allen Ye

- 1 Introduction . . . . . 223
- 2 Ensemble Classifiers . . . . . 224
  - 2.1 Examples of Related Work . . . . . 224
  - 2.2 A Framework for Ensemble Classifiers . . . . . 226
  - 2.3 Classifying Ensemble Classifiers . . . . . 227
  - 2.4 Ensemble Classifier Examples . . . . . 229
- 3 Experiments and Results . . . . . 232
  - 3.1 Dataset and Features . . . . . 233
  - 3.2 Metrics . . . . . 235
  - 3.3 Software . . . . . 235
  - 3.4 Overview of Experiments . . . . . 236
  - 3.5 Standard Techniques . . . . . 236
  - 3.6 Bagging Experiments . . . . . 236
  - 3.7 Boosting Experiments . . . . . 237

- 3.8 Voting Experiments . . . . . 238
- 3.9 Discussion . . . . . 238
- 4 Conclusion and Future Work . . . . . 240
- Appendix: Confusion Matrices . . . . . 241
- References . . . . . 244

**Malware Analysis**

**Optimizing Multi-class Classification of Binaries Based on Static Features . . . . . 249**

Lasse Øverlier

- 1 Introduction . . . . . 249
- 2 Related Work and Background . . . . . 250
- 3 Methodology . . . . . 252
  - 3.1 Selecting the Dictionary . . . . . 252
- 4 Experiments . . . . . 255
  - 4.1 Microsoft Malware Classification Challenge . . . . . 256
  - 4.2 Google Code Jam (GCJ) Data . . . . . 257
- 5 Results . . . . . 258
  - 5.1 MMCC Malware . . . . . 259
  - 5.2 Google Code Jam (GCJ) Results . . . . . 262
- 6 Discussion . . . . . 265
  - 6.1 Length of N-Gram . . . . . 265
  - 6.2 Simplification of Code . . . . . 265
  - 6.3 Size of Training Data . . . . . 266
  - 6.4 64-Bit Optimized Binaries Versus 32-Bit Non-optimized Binaries . . . . . 266
  - 6.5 Classification Algorithm . . . . . 266
- 7 Conclusion . . . . . 267
- References . . . . . 267

**Deep Learning Techniques for Behavioral Malware Analysis in Cloud IaaS . . . . . 269**

Andrew McDole, Maanak Gupta, Mahmoud Abdelsalam, Sudip Mittal, and Mamoun Alazab

- 1 Introduction and Motivation . . . . . 269
  - 1.1 Relevance in Cloud IaaS . . . . . 270
- 2 Machine Learning-Based Malware Detection . . . . . 271
  - 2.1 File Classification . . . . . 271
  - 2.2 Online Malware Detection . . . . . 272
- 3 Literature Review . . . . . 273
- 4 Cloud Security Monitoring Overview . . . . . 273
- 5 Behavioral Features and Characteristics . . . . . 278

- 6 Experimental Setup and Methodology . . . . . 279
- 7 Deep Learning Techniques . . . . . 281
  - 7.1 Comparative Analysis . . . . . 282
- 8 Conclusion . . . . . 283
- References . . . . . 284

**A Comparison of Word2Vec, HMM2Vec, and PCA2Vec for Malware Classification . . . . . 287**

Aniket Chandak, Wendy Lee, and Mark Stamp

- 1 Introduction . . . . . 287
- 2 Related Work . . . . . 288
- 3 Background . . . . . 289
  - 3.1 Neural Networks . . . . . 290
  - 3.2 Hidden Markov Models . . . . . 292
  - 3.3 Principal Component Analysis . . . . . 297
  - 3.4 Classifiers . . . . . 298
- 4 Word Embedding Techniques . . . . . 301
  - 4.1 HMM2Vec . . . . . 302
  - 4.2 PCA2Vec . . . . . 303
  - 4.3 Word2Vec . . . . . 304
- 5 Experiments and Results . . . . . 307
  - 5.1 Dataset . . . . . 307
  - 5.2 Classifier Parameters . . . . . 309
  - 5.3 Baseline Results . . . . . 310
  - 5.4 HMM2Vec Results . . . . . 310
  - 5.5 PCA2Vec Results . . . . . 312
  - 5.6 Word2Vec Results . . . . . 313
  - 5.7 Overfitting . . . . . 315
  - 5.8 Discussion . . . . . 317
- 6 Conclusion and Future Work . . . . . 318
- References . . . . . 318

**Word Embedding Techniques for Malware Evolution Detection . . . . . 321**

Sunhera Paul and Mark Stamp

- 1 Introduction . . . . . 321
- 2 Related Work . . . . . 322
- 3 Implementation . . . . . 324
  - 3.1 Dataset . . . . . 324
  - 3.2 Feature Extraction . . . . . 326
  - 3.3 Classification Techniques . . . . . 326
- 4 Experiments and Results . . . . . 331
  - 4.1 HMM-Based Secondary Test . . . . . 331
  - 4.2 Opcode-SVM Results . . . . . 332

- 4.3 Opcode  $n$ -Gram-SVM Results . . . . . 333
- 4.4 Opcode-Word2Vec-SVM Results . . . . . 334
- 4.5 Opcode-HMM2Vec-SVM Results . . . . . 336
- 5 Conclusion and Future Work . . . . . 339
- References . . . . . 342
- Reanimating Historic Malware Samples . . . . . 345**
- Paul Black, Iqbal Gondal, Peter Vamplew, and Arun Lakhotia
- 1 Introduction . . . . . 346
  - 1.1 Motivation . . . . . 347
  - 1.2 Emulator Architecture . . . . . 347
- 2 Manual Construction . . . . . 348
- 3 Zeus C2 Server Emulator . . . . . 349
- 4 Ransomware C2 Server Emulators . . . . . 352
  - 4.1 CryptoLocker C2 Server Emulator . . . . . 352
  - 4.2 CryptoWall C2 Server Emulator . . . . . 353
- 5 Semi-automated Generation of C2 Server Emulators . . . . . 356
- 6 Limitations . . . . . 357
- 7 Conclusion . . . . . 358
- References . . . . . 359
- Cluster Analysis of Malware Family Relationships . . . . . 361**
- Samanvitha Basole and Mark Stamp
- 1 Introduction . . . . . 361
- 2 Background . . . . . 362
  - 2.1 Related Work . . . . . 362
  - 2.2 Dataset . . . . . 362
  - 2.3 Metrics . . . . . 364
  - 2.4  $K$ -Means . . . . . 366
  - 2.5 Elbow Plots . . . . . 369
- 3 Experiments and Results . . . . . 370
  - 3.1 Clustering by Family . . . . . 370
  - 3.2 Clustering Families of Different Type . . . . . 374
  - 3.3 Clustering Families of the Same Type . . . . . 376
- 4 Conclusion and Future Work . . . . . 377
- References . . . . . 378
- Beyond Labeling: Using Clustering to Build Network Behavioral Profiles of Malware Families . . . . . 381**
- Azqa Nadeem, Christian Hammerschmidt, Carlos H. Gañán, and Sicco Verwer
- 1 Introduction . . . . . 382
- 2 The Problem with AV Labels . . . . . 384
- 3 Related Work . . . . . 385

- 3.1 Challenges in Malware Labeling . . . . . 385
- 3.2 Research Objectives: Detection Versus Analysis . . . . . 386
- 3.3 Challenges in Malware Behavior Modeling . . . . . 387
- 4 MalPaCA: Malware Packet Sequence Clustering and Analysis . . . . . 389
  - 4.1 Connection Generation (P1) . . . . . 390
  - 4.2 Feature-Set Extraction (P2) . . . . . 391
  - 4.3 Distance Measure (P3) . . . . . 392
  - 4.4 HDBScan Clustering (P4) . . . . . 393
  - 4.5 Cluster Visualization (P5) . . . . . 394
- 5 Experimental Setup . . . . . 395
  - 5.1 Experimental Dataset . . . . . 395
  - 5.2 MalPaCA Parameters . . . . . 395
- 6 Malware Capability Assessment . . . . . 397
  - 6.1 Cluster Characterization . . . . . 399
  - 6.2 Constructing Behavioral Profiles . . . . . 400
  - 6.3 Showing Relationships Using DAG . . . . . 401
- 7 Comparative Analysis . . . . . 402
  - 7.1 Comparison with Traditional Family Labels . . . . . 402
  - 7.2 Comparison with Statistical Features . . . . . 404
- 8 Limitations and Future Work . . . . . 405
- 9 Conclusions . . . . . 406
- References . . . . . 406

**An Empirical Analysis of Image-Based Learning Techniques for Malware Classification . . . . . 411**

Pratikkumar Prajapati and Mark Stamp

- 1 Introduction . . . . . 411
- 2 Background . . . . . 412
  - 2.1 Related Work . . . . . 412
  - 2.2 Learning Techniques . . . . . 413
  - 2.3 Dataset . . . . . 416
  - 2.4 Hardware . . . . . 417
  - 2.5 Software . . . . . 417
- 3 Deep Learning Experiments and Results . . . . . 418
  - 3.1 Multilayer Perceptron Experiments . . . . . 419
  - 3.2 Convolutional Neural Network Experiments . . . . . 420
  - 3.3 Recurrent Neural Networks . . . . . 422
  - 3.4 Transfer Learning . . . . . 424
  - 3.5 Discussion . . . . . 425
- 4 Conclusions and Future Work . . . . . 426
- Appendix: Confusion Matrices . . . . . 427
- References . . . . . 434

**A Novel Study on Multinomial Classification of x86/x64 Linux ELF Malware Types and Families Through Deep Neural Networks . . . . . 437**  
Andrii Shalaginov and Lasse Øverlier

1 Introduction . . . . . 437

2 State of the Art: Machine Learning for Linux Malware Detection . . . . . 439

3 Linux Malware: Automated Features Extraction and Classification . . . . . 440

4 Methodology: Malware Analysis and Detection . . . . . 442

5 Experimental Design . . . . . 443

    5.1 Dataset . . . . . 443

    5.2 Experimental Setup . . . . . 446

6 Results and Analysis . . . . . 446

    6.1 Feature Selection . . . . . 447

    6.2 Classification Accuracy: State-of-the-Art Methods . . . . . 447

    6.3 Deep Learning . . . . . 448

7 Discussions and Conclusions . . . . . 450

References . . . . . 451

**Fast and Straightforward Feature Selection Method . . . . . 455**  
Sergii Banin

1 Introduction . . . . . 456

2 Background . . . . . 458

    2.1 Problem Description . . . . . 458

    2.2 Literature Overview . . . . . 461

3 Intersection Subtraction Selection Method . . . . . 463

    3.1 The Context . . . . . 463

    3.2 Feature Selection Algorithm . . . . . 463

    3.3 Computational Complexity . . . . . 464

    3.4 Theoretical Assessment . . . . . 465

4 Experimental Evaluation . . . . . 466

    4.1 Dataset . . . . . 467

    4.2 Experimental Environment . . . . . 468

    4.3 Memory Access Operations . . . . . 468

    4.4 Data Collection . . . . . 468

    4.5 Feature Selection and Machine Learning Algorithms . . . . . 469

    4.6 Time Complexity . . . . . 470

    4.7 Analysis of Selected Feature Sets . . . . . 470

    4.8 Classification Performance . . . . . 471

5 Discussion and Future Work . . . . . 473

6 Conclusions . . . . . 474

References . . . . . 474

**A Comparative Study of Adversarial Attacks to Malware Detectors Based on Deep Learning** . . . . . 477

Corrado Aaron Visaggio, Fiammetta Marulli, Sonia Laudanna, Benedetta La Zazzera, and Antonio Pirozzi

1 Introduction . . . . . 477

2 The Deep Learning Models Adopted in Malware Detection . . . . . 479

    2.1 The Deep Learning Models in a Nutshell . . . . . 479

3 Adversarial Attacks Against Deep Learning-Based Malware Detection System . . . . . 482

4 Generative Adversarial Attacks Against Malware Detection Systems . . . . . 484

5 Case Study . . . . . 485

    5.1 Case Study Design . . . . . 486

    5.2 General Architecture . . . . . 488

    5.3 Adversary Logic . . . . . 492

    5.4 Dataset . . . . . 493

    5.5 Performance Metrics . . . . . 495

    5.6 Case Study Treatments . . . . . 496

    5.7 Case Study Results and Performance Evaluation . . . . . 499

6 Conclusions . . . . . 507

References . . . . . 508

**Related Topics**

**Detecting Abusive Comments Using Ensemble Deep Learning Algorithms** . . . . . 515

Ravinder Ahuja, Alisha Banga, and S C Sharma

1 Introduction . . . . . 515

2 Literature Survey . . . . . 517

3 Materials and Methods . . . . . 519

    3.1 Dataset . . . . . 519

    3.2 Data Pre-processing . . . . . 519

    3.3 Text Representation Techniques . . . . . 520

    3.4 Traditional Machine Learning Methods . . . . . 523

    3.5 Deep Learning Methods . . . . . 525

4 Methodology Used . . . . . 529

5 Experimental Results and Analysis . . . . . 530

6 Conclusion and Future Work . . . . . 531

References . . . . . 533

**DURLD: Malicious URL Detection Using Deep Learning-Based Character Level Representations** . . . . . 535

Sriram Srinivasan, R. Vinayakumar, Ajay Arunachalam, Mamoun Alazab, and KP Soman

1 Introduction . . . . . 536

2 Related Works . . . . . 537

3 An Overview of Uniform Resource Locator (URL) . . . . . 538

4 Background Details of Deep Learning Models . . . . . 539

4.1 Hybrid Architecture—Convolutional Neural Network and Long Short-Term Memory (CNN-LSTM) with Character Level Keras Embedding . . . . . 539

4.2 Character-Based Models . . . . . 540

4.3 Problem Formulation . . . . . 542

5 Shortcomings in Malicious URL Detection . . . . . 543

6 Description of Data Set . . . . . 543

7 Model Configuration of Malicious URL Detection Engine . . . . . 543

8 Proposed Architecture—DeepURLDetect (DURLD) . . . . . 547

9 Performance Measures . . . . . 550

10 Evaluation Results and Observations . . . . . 551

11 Conclusion . . . . . 552

References . . . . . 552

**Sentiment Analysis for Troll Detection on Weibo** . . . . . 555

Zidong Jiang, Fabio Di Troia, and Mark Stamp

1 Introduction . . . . . 555

2 Background . . . . . 557

2.1 Trolls . . . . . 557

2.2 Machine Learning Techniques . . . . . 558

2.3 Evaluation Metric . . . . . 561

3 Related Work . . . . . 562

4 Datasets . . . . . 563

4.1 Chinese Segmentation Dataset . . . . . 563

4.2 Sentiment Analysis Dataset . . . . . 564

4.3 Troll Detection Dataset . . . . . 566

5 Implementation and Results . . . . . 569

5.1 Weibo Crawler . . . . . 569

5.2 HMM for Chinese Segmentation . . . . . 570

5.3 HMM for Emotion Classification . . . . . 570

5.4 Sentiment Score Calculation . . . . . 572

5.5 Troll Detection with XGBoost and SVM . . . . . 572

5.6 Chrome Extension for Troll Detection . . . . . 575

6 Conclusion and Future Work . . . . . 577

References . . . . . 577



<b>Log-Based Malicious Activity Detection Using Machine and Deep Learning</b> . . . . .	581
Katarzyna A. Tarnowska and Araav Patel	
1 Introduction . . . . .	581
2 Related Work . . . . .	584
3 Methods . . . . .	587
3.1 Solution Design . . . . .	587
3.2 User Behavior Modeling . . . . .	587
3.3 Anomaly Detection . . . . .	588
3.4 Scenario . . . . .	589
4 Experiments . . . . .	590
4.1 Distance-Based Outlier Detection . . . . .	591
4.2 Machine and Deep Learning for Anomaly Detection . . . . .	594
5 Conclusions . . . . .	600
5.1 Future Work . . . . .	601
References . . . . .	602
<b>Image Spam Classification with Deep Neural Networks</b> . . . . .	605
Ajay Pal Singh and Katerina Potika	
1 Introduction . . . . .	605
2 Problem Statement and Motivation . . . . .	606
3 Background . . . . .	607
3.1 Spam Categories . . . . .	607
3.2 Classification Techniques . . . . .	608
3.3 Quality Metrics . . . . .	611
4 Related Work . . . . .	612
5 Framework . . . . .	613
5.1 Datasets . . . . .	613
5.2 Data Pre-processing . . . . .	615
5.3 Image Features . . . . .	615
5.4 Techniques Used . . . . .	618
5.5 Deep Neural Networks . . . . .	618
5.6 Transfer Learning . . . . .	622
6 Experimental Results . . . . .	622
6.1 Neural Network Results . . . . .	622
6.2 Deep Neural Network Results . . . . .	625
6.3 Image Spam Hunter . . . . .	626
6.4 Dredze Dataset . . . . .	626
6.5 Convolution Neural Networks and Transfer Learning Results . . . . .	627
7 Conclusion and Future Work . . . . .	629
References . . . . .	630

- Universal Adversarial Perturbations and Image Spam Classifiers . . . . . 633**
- Andy Phung and Mark Stamp
- 1 Introduction . . . . . 633
- 2 Background . . . . . 634
  - 2.1 Image Spam Filtering . . . . . 634
  - 2.2 Adversarial Learning . . . . . 635
- 3 Evaluating Adversarial Attacks . . . . . 637
  - 3.1 Experimental Design . . . . . 637
  - 3.2 Analysis . . . . . 639
- 4 Inceptionism-Augmented Universal Perturbations . . . . . 643
  - 4.1 Procedure . . . . . 643
  - 4.2 Implementation . . . . . 644
  - 4.3 Performance Evaluation . . . . . 645
  - 4.4 Proposed Dataset Analysis . . . . . 646
- 5 Conclusion and Future Work . . . . . 649
- References . . . . . 650