

1. Record Nr.	UNINA9910829843503321
Autore	Abdel-Basset Mohamed <1985->
Titolo	Deep learning approaches for security threats in IoT environments / / Mohamed Abdel-Basset, Zagazig University, Egypt, Nour Moustafa, UNSW Canberra at the Australian Defence Force Academy, Australia, Hossam Hawash, Zagazig University, Egypt
Pubbl/distr/stampa	Hoboken, New Jersey : , : John Wiley & Sons, Inc., , [2023] ©2023
ISBN	1-119-88417-9 1-119-88415-2
Descrizione fisica	1 online resource (387 pages)
Disciplina	004.678
Soggetti	Internet of things - Security measures - Data processing Deep learning (Machine learning)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Cover -- Title Page -- Copyright Page -- Contents -- About the Authors -- Chapter 1 Introducing Deep Learning for IoT Security -- 1.1 Introduction -- 1.2 Internet of Things (IoT) Architecture -- 1.2.1 Physical Layer -- 1.2.2 Network Layer -- 1.2.3 Application Layer -- 1.3 Internet of Things' Vulnerabilities and Attacks -- 1.3.1 Passive Attacks -- 1.3.2 Active Attacks -- 1.4 Artificial Intelligence -- 1.5 Deep Learning -- 1.6 Taxonomy of Deep Learning Models -- 1.6.1 Supervision Criterion -- 1.6.1.1 Supervised Deep Learning -- 1.6.1.2 Unsupervised Deep Learning -- 1.6.1.3 Semi-Supervised Deep Learning -- 1.6.1.4 Deep Reinforcement Learning -- 1.6.2 Incrementality Criterion -- 1.6.2.1 Batch Learning -- 1.6.2.2 Online Learning -- 1.6.3 Generalization Criterion -- 1.6.3.1 Model-Based Learning -- 1.6.3.2 Instance-Based Learning -- 1.6.4 Centralization Criterion -- 1.7 Supplementary Materials -- References -- Chapter 2 Deep Neural Networks -- 2.1 Introduction -- 2.2 From Biological Neurons to Artificial Neurons -- 2.2.1 Biological Neurons -- 2.2.2 Artificial Neurons -- 2.3 Artificial Neural Network -- 2.3.1 Input Layer -- 2.3.2 Hidden Layer -- 2.3.3 Output Layer -- 2.4 Activation Functions --

2.4.1 Types of Activation -- 2.4.1.1 Binary Step Function -- 2.4.1.2 Linear Activation Function -- 2.4.1.3 Nonlinear Activation Functions -- 2.5 The Learning Process of ANN -- 2.5.1 Forward Propagation -- 2.5.2 Backpropagation (Gradient Descent) -- 2.6 Loss Functions -- 2.6.1 Regression Loss Functions -- 2.6.1.1 Mean Absolute Error (MAE) Loss -- 2.6.1.2 Mean Squared Error (MSE) Loss -- 2.6.1.3 Huber Loss -- 2.6.1.4 Mean Bias Error (MBE) Loss -- 2.6.1.5 Mean Squared Logarithmic Error (MSLE) -- 2.6.2 Classification Loss Functions -- 2.6.2.1 Binary Cross Entropy (BCE) Loss -- 2.6.2.2 Categorical Cross Entropy (CCE) Loss -- 2.6.2.3 Hinge Loss. 2.6.2.4 Kullback-Leibler Divergence (KL) Loss -- 2.7 Supplementary Materials -- References -- Chapter 3 Training Deep Neural Networks -- 3.1 Introduction -- 3.2 Gradient Descent Revisited -- 3.2.1 Gradient Descent -- 3.2.2 Stochastic Gradient Descent -- 3.2.3 Mini-batch Gradient Descent -- 3.3 Gradient Vanishing and Explosion -- 3.4 Gradient Clipping -- 3.5 Parameter Initialization -- 3.5.1 Zero Initialization -- 3.5.2 Random Initialization -- 3.5.3 Lecun Initialization -- 3.5.4 Xavier Initialization -- 3.5.5 Kaiming (He) Initialization -- 3.6 Faster Optimizers -- 3.6.1 Momentum Optimization -- 3.6.2 Nesterov Accelerated Gradient -- 3.6.3 AdaGrad -- 3.6.4 RMSProp -- 3.6.5 Adam Optimizer -- 3.7 Model Training Issues -- 3.7.1 Bias -- 3.7.2 Variance -- 3.7.3 Overfitting Issues -- 3.7.4 Underfitting Issues -- 3.7.5 Model Capacity -- 3.8 Supplementary Materials -- References -- Chapter 4 Evaluating Deep Neural Networks -- 4.1 Introduction -- 4.2 Validation Dataset -- 4.3 Regularization Methods -- 4.3.1 Early Stopping -- 4.3.2 L1 and L2 Regularization -- 4.3.3 Dropout -- 4.3.4 Max-Norm Regularization -- 4.3.5 Data Augmentation -- 4.4 Cross-Validation -- 4.4.1 Hold-Out Cross-Validation -- 4.4.2 k-Folds Cross-Validation -- 4.4.3 Stratified k-Folds' Cross-Validation -- 4.4.4 Repeated k-Folds' Cross-Validation -- 4.4.5 Leave-One-Out Cross-Validation -- 4.4.6 Leave-p-Out Cross-Validation -- 4.4.7 Time Series Cross-Validation -- 4.4.8 Rolling Cross-Validation -- 4.4.9 Block Cross-Validation -- 4.5 Performance Metrics -- 4.5.1 Regression Metrics -- 4.5.1.1 Mean Absolute Error (MAE) -- 4.5.1.2 Root Mean Squared Error (RMSE) -- 4.5.1.3 Coefficient of Determination (R2) -- 4.5.1.4 Adjusted R2 -- 4.5.2 Classification Metrics -- 4.5.2.1 Confusion Matrix -- 4.5.2.2 Accuracy -- 4.5.2.3 Precision -- 4.5.2.4 Recall -- 4.5.2.5 Precision-Recall Curve -- 4.5.2.6 F1-Score. 4.5.2.7 Beta F1 Score -- 4.5.2.8 False Positive Rate (FPR) -- 4.5.2.9 Specificity -- 4.5.2.10 Receiving Operating Characteristics (ROC) Curve -- 4.6 Supplementary Materials -- References -- Chapter 5 Convolutional Neural Networks -- 5.1 Introduction -- 5.2 Shift from Full Connected to Convolutional -- 5.3 Basic Architecture -- 5.3.1 The Cross-Correlation Operation -- 5.3.2 Convolution Operation -- 5.3.3 Receptive Field -- 5.3.4 Padding and Stride -- 5.3.4.1 Padding -- 5.3.4.2 Stride -- 5.4 Multiple Channels -- 5.4.1 Multi-Channel Inputs -- 5.4.2 Multi-Channel Output -- 5.4.3 Convolutional Kernel 1 x 1 -- 5.5 Pooling Layers -- 5.5.1 Max Pooling -- 5.5.2 Average Pooling -- 5.6 Normalization Layers -- 5.6.1 Batch Normalization -- 5.6.2 Layer Normalization -- 5.6.3 Instance Normalization -- 5.6.4 Group Normalization -- 5.6.5 Weight Normalization -- 5.7 Convolutional Neural Networks (LeNet) -- 5.8 Case Studies -- 5.8.1 Handwritten Digit Classification (One Channel Input) -- 5.8.2 Dog vs. Cat Image Classification (Multi-Channel Input) -- 5.9 Supplementary Materials -- References -- Chapter 6 Dive Into Convolutional Neural Networks -- 6.1 Introduction -- 6.2 One-Dimensional Convolutional Network -- 6.2.1 One-Dimensional Convolution -- 6.2.2 One-Dimensional Pooling -- 6.3 Three-Dimensional Convolutional Network -- 6.3.1 Three-

Dimensional Convolution -- 6.3.2 Three-Dimensional Pooling -- 6.4 Transposed Convolution Layer -- 6.5 Atrous/Dilated Convolution -- 6.6 Separable Convolutions -- 6.6.1 Spatially Separable Convolutions -- 6.6.2 Depth-wise Separable (DS) Convolutions -- 6.7 Grouped Convolution -- 6.8 Shuffled Grouped Convolution -- 6.9 Supplementary Materials -- References -- Chapter 7 Advanced Convolutional Neural Network -- 7.1 Introduction -- 7.2 AlexNet -- 7.3 Block-wise Convolutional Network (VGG) -- 7.4 Network in Network -- 7.5 Inception Networks. 7.5.1 GoogLeNet -- 7.5.2 Inception Network v2 (Inception v2) -- 7.5.3 Inception Network v3 (Inception v3) -- 7.6 Residual Convolutional Networks -- 7.7 Dense Convolutional Networks -- 7.8 Temporal Convolutional Network -- 7.8.1 One-Dimensional Convolutional Network -- 7.8.2 Causal and Dilated Convolution -- 7.8.3 Residual Blocks -- 7.9 Supplementary Materials -- References -- Chapter 8 Introducing Recurrent Neural Networks -- 8.1 Introduction -- 8.2 Recurrent Neural Networks -- 8.2.1 Recurrent Neurons -- 8.2.2 Memory Cell -- 8.2.3 Recurrent Neural Network -- 8.3 Different Categories of RNNs -- 8.3.1 One-to-One RNN -- 8.3.2 One-to-Many RNN -- 8.3.3 Many-to-One RNN -- 8.3.4 Many-to-Many RNN -- 8.4 Backpropagation Through Time -- 8.5 Challenges Facing Simple RNNs -- 8.5.1 Vanishing Gradient -- 8.5.2 Exploding Gradient -- 8.5.2.1 Truncated Backpropagation Through Time (TBPTT) -- 8.5.2.2 Penalty on the Recurrent Weights W_{hh} -- 8.5.2.3 Clipping Gradients -- 8.6 Case Study: Malware Detection -- 8.7 Supplementary Materials -- References -- Chapter 9 Dive Into Recurrent Neural Networks -- 9.1 Introduction -- 9.2 Long Short-Term Memory (LSTM) -- 9.2.1 LSTM Gates -- 9.2.2 Candidate Memory Cells -- 9.2.3 Memory Cell -- 9.2.4 Hidden State -- 9.3 LSTM with Peephole Connections -- 9.4 Gated Recurrent Units (GRU) -- 9.4.1 CRU Cell Gates -- 9.4.2 Candidate State -- 9.4.3 Hidden State -- 9.5 ConvLSTM -- 9.6 Unidirectional vs. Bidirectional Recurrent Network -- 9.7 Deep Recurrent Network -- 9.8 Insights -- 9.9 Case Study of Malware Detection -- 9.10 Supplementary Materials -- References -- Chapter 10 Attention Neural Networks -- 10.1 Introduction -- 10.2 From Biological to Computerized Attention -- 10.2.1 Biological Attention -- 10.2.2 Queries, Keys, and Values -- 10.3 Attention Pooling: Nadaraya-Watson Kernel Regression -- 10.4 Attention-Scoring Functions. 10.4.1 Masked Softmax Operation -- 10.4.2 Additive Attention (AA) -- 10.4.3 Scaled Dot-Product Attention -- 10.5 Multi-Head Attention (MHA) -- 10.6 Self-Attention Mechanism -- 10.6.1 Self-Attention (SA) Mechanism -- 10.6.2 Positional Encoding -- 10.7 Transformer Network -- 10.8 Supplementary Materials -- References -- Chapter 11 Autoencoder Networks -- 11.1 Introduction -- 11.2 Introducing Autoencoders -- 11.2.1 Definition of Autoencoder -- 11.2.2 Structural Design -- 11.3 Convolutional Autoencoder -- 11.4 Denoising Autoencoder -- 11.5 Sparse Autoencoders -- 11.6 Contractive Autoencoders -- 11.7 Variational Autoencoders -- 11.8 Case Study -- 11.9 Supplementary Materials -- References -- Chapter 12 Generative Adversarial Networks (GANs) -- 12.1 Introduction -- 12.2 Foundation of Generative Adversarial Network -- 12.3 Deep Convolutional GAN -- 12.4 Conditional GAN -- 12.5 Supplementary Materials -- References -- Chapter 13 Dive Into Generative Adversarial Networks -- 13.1 Introduction -- 13.2 Wasserstein GAN -- 13.2.1 Distance Functions -- 13.2.2 Distance Function in GANs -- 13.2.3 Wasserstein Loss -- 13.3 Least-Squares GAN (LSGAN) -- 13.4 Auxiliary Classifier GAN (ACGAN) -- 13.5 Supplementary Materials -- References -- Chapter 14 Disentangled Representation GANs -- 14.1 Introduction -- 14.2

Disentangled Representations -- 14.3 InfoGAN -- 14.4 StackedGAN --
14.5 Supplementary Materials -- References -- Chapter 15 Introducing
Federated Learning for Internet of Things (IoT) -- 15.1 Introduction --
15.2 Federated Learning in the Internet of Things -- 15.3 Taxonomic
View of Federated Learning -- 15.3.1 Network Structure -- 15.3.1.1
Centralized Federated Learning -- 15.3.1.2 Decentralized Federated
Learning -- 15.3.1.3 Hierarchical Federated Learning -- 15.3.2 Data
Partition -- 15.3.3 Horizontal Federated Learning -- 15.3.4 Vertical
Federated Learning.
15.3.5 Federated Transfer Learning.
