Contents

List of Figures **xv**

Foreword xxiii

Preface xxvii

Acknowledgments xxxi

Acronyms xxxv

1 Introduction **1**

1.1 Why we wrote this book and who it is for 2

1.2 Do you need deep learning? 4

1.3 Book structure 8

1.4 Some relevant regulatory frameworks 10

1.4.1 The European Union 11

1.4.2 The United States 14

1.4.3 The United Kingdom 16

1.4.4 Canada 17

1.4.5 Singapore 18

1.5 Technical background 20

1.5.1 Early days: the Perceptron 21

1.5.2 The Multilayer Perceptron 25

1.5.3 Training MLP: Backpropagation 29

1.5.4 From Multilayer to Deep Learning 34

2 Image Processing and Convolutional Neural Networks 39

2.1 Image Data in Banking 41

2.2 Regulatory Constraints 43

2.3 Deep Learning for Image Processing: Convolutional Neural Networks 44

2.3.1 Introduction to CNNs 45

2.3.2 Basic Structure 48

2.3.3 Notable models 58

2.4 Training CNNs 64

2.4.1 Transfer Learning of Image Models 66

2.5 CNN applications in banking and finance 67

2.5.1 Image Classification 67

2.5.2 Content-Based Image Retrieval 69

2.5.3 Object Localization and Detection 69

2.5.4 Video Analysis 70

2.5.5 Generative Convolutional Networks 72

2.6 Case Study: LiDAR-Based Mortgage Default Rate Prediction . 74

2.6.1 Data loaders and preprocessing 75

2.6.2 A simple CNN 78

2.6.3 Explaining the predictions: Grad-CAM 85

2.6.4 Fine-tuning a ResNet-50 model 87

3 Time series and Recurrent Models **93**

3.1 Time Series and Panel Data in Banking 94

3.2 Regulatory Aspects of Time-Series and Tabular Data 96

3.3 Deep Learning Time-Series Models 100

3.3.1 Recurrent Neural Networks 100

3.3.2 Long Short-Term Memory 103

3.3.3 Gated Recurrent Unit 108

3.3.4 Echo State Networks 109

3.3.5 Attention 110

3.3.6 Temporal Convolutional Networks 112

3.4 Time series applications in banking and finance 115

3.5 Case study: Behavioral Scoring Using Deep Learning 118

3.5.1 Data processing 119

3.5.2 Single channel models 123

3.5.3 A first bimodal model 125

3.5.4 Training and testing 126

4 Text Data and Transformers **135**

4.1 Text Data in Banking 136

4.2 Regulatory and Ethical Concerns 138

4.3 Recurrency and Attention for Text Analytics 142

4.4 Multi-head Attention 146

4.5 Transformers 148

4.6 Transformer-based architectures for text: from BERT and beyond 153

4.6.1 BERT: Bidirectional Encoder Representation from Transformers . . . 154
4.6.2 RoBERTa: a robustly optimized BERT . . . 155
4.6.3 ALBERT: a Lite BERT . . . 156
4.6.4 DistilBERT: a distilled BERT . . . 158
4.6.5 ModernBERT . . . 159
4.7 Fine-tuning of pre-trained models . . . 160
4.8 Training a text model from scratch . . . 162
4.9 Text-based applications in banking and finance . . . 165
4.10 Case study: Recession chances arising from Fed speeches using BERT . . . 168
4.10.1 Data Loading and Preprocessing . . . 169
4.10.2 Model fine tuning . . . 172
4.10.3 Inference over new speeches . . . 176
5 Financial Contagion and Network Models 179
5.1 Network data in banking . . . 182
5.1.1 Network fundamentals . . . 182
5.1.2 Explicit Networks . . . 188
5.1.3 Implicit networks . . . 189
5.2 Regulatory concerns of using network data . . . 196
5.3 Network features: centrality and neighborhood . . . 198
5.3.1 Centrality measures . . . 198
5.3.2 Link-based measures . . . 203

5.4 Graph Neural Networks 206

5.4.1 Graph Neural Networks Origins 207

5.4.2 GraphSAGE 211

5.4.3 Graph Convolutional Networks 212

5.4.4 Graph Attention Networks 213

5.4.5 Graph Isomorphism Networks 215

5.4.6 Transductive and inductive learning 216

5.4.7 Temporal Graph Neural Networks 219

5.5 Network applications in banking and finance 221

5.6 Case study: Default correlation in mortgage lending 224

5.6.1 Building the networks 225

5.6.2 Defining GNN models 229

5.6.3 Training and testing 233

5.6.4 Model performance 235

6 Generative AI and Large Language Models 241

6.1 Large Language Models in Banking 242

6.2 Regulatory focus on Large Language Models 245

6.3 LLM Theory 248

6.4 Alignment and Reinforcement Learning from Human Feedback (RLHF) 252

6.5 LLMs as Agents, Not Just Predictors 254

6.5.1 Mixture of Experts (MoE) 255

6.5.2 Retrieval-Augmented Generation (RAG) 256

6.6 Agentic AI 259

6.7 Licensing issues and data use 263

6.8 LLM applications in banking and finance 268

6.9 Case-study: Comparison of prompt engineering and fine-tuning for a credit scoring explainer 269

6.9.1 Importing the models and calculating the scoring system 271

6.9.2 Zero-Shot Prompting and Prompt Engineering . . . 273

6.9.3 Fine-tuning and Quantizing an LLM 279

6.9.4 Testing on new data 282

7 Multimodal Data and Information Fusion 285

7.1 AI Multimodal Models 287

7.1.1 Data fusion: When to fuse the data 288

7.1.2 Data fusion: How to combine multiple data sources 290

7.2 Creating a Multimodal Model 294

7.2.1 Training the Model 295

7.2.2 Deploying Multimodal Models 296

7.3 Some Specific Challenges in the Financial Sector 298

7.4 Operational Implications 300

7.5 Case study: A multimodal mortgage model 301

7.5.1 Data preprocessing for multimodal deep learning . . 302

7.5.2 Building the data loaders 305

7.5.3 Defining and training the models 306

7.5.4 Evaluation 310

8 Fairness, Accountability, Explainability and Causality 313

8.1 Fairness 316

8.1.1 The challenge of fairness 317

8.1.2 The different approaches to fairness 319

8.1.3 Mitigating unfairness 325

8.2 Accountability 327

8.3 Explainability 332

8.3.1 SHAP 334

8.3.2 Explainable AI in Operational Research 339

8.4 Explainability vs Causality 341

8.5 Causal Deep Learning 348

8.6 Case study: Fairness and Explainability 353

8.6.1 Fairness 353

8.6.2 Explainability 360

9 Perspectives on the future of AI in banking 365

9.1 ROI analysis and project prioritization 366

9.2 Deploying deep learning models 368

9.2.1 Cloud or on-premise? 375

9.3 Future global perspectives 376

9.4 Closing Words 378

Bibliography 381

List of Figures

1.1 Example of a tabular dataset for supervised learning inspired by the credit scoring task. 22

1.2 The architecture of the Perceptron model. 23

1.3 Graphical representation of various activation functions. . . 25

1.4 The architecture of the Multilayer Perceptron (MLP) model. 27

1.5 Illustration of different data representations: scalar (0D), vector (1D), matrix (2D), and a three-dimensional tensor. While tensors can have any number of dimensions, the 0D, 1D, and 2D cases are so commonly used that they are given special names. 34

2.1 Free LiDAR data coverage in the US. Map services and data available from U.S. Geological Survey, National Geospatial Program. 43

2.2 Samples from the MNIST handwritten digits dataset. 46

2.3 Example of a standard CNN. 50

2.4 Functioning of the convolution and max-pooling layers. . . . 56

2.5 Example of a LiDAR image for MSA 1018. Source: USGS data under open license. 78

2.6 Loss plots for different learning rates. 82

2.7 SmoothGrad-CAM++ for the simple CNN model. 86

3.1 Representation of the RNN model (with backward connections and unrolled). 102

3.2 Temporal Convolutional Network (TCN). Each hidden layer "dilates" the input point by d steps, which can increase as more layers are stacked. This is memory-efficient and easier to parallelize over long sequences. 114

3.3 Bimodal recurrent network with static features. 125

3.4 ROC curves for all models. 132

4.1 RNN architecture adapted to NLP. 144

4.2 Overview of the Transformer architecture, structured around an encoder-decoder framework. Key stages include input embedding, encoding of the input sequence, decoding with feedback from the partially generated output, and iterative generation of the final output. 150

4.3 Structure of a encoder block with multi-head self-attention and feed-forward sublayers, repeated $N = 6$ times. 151

4.4 Structure of a decoder block with masked self-attention, encoder-decoder attention, and feed-forward sublayers, repeated $N = 6$ times. 152

4.5 Transformer Architecture. Adapted from Vaswani et al. (2017). 153

4.6 ROC curve for the Fed speech economic forecasting model. 176

5.1 A simple network 183

5.2 Examples of networks that are either weighted or unweighted and directed or undirected 185

5.3 Adjacency matrices of the networks in figure 5.2 186

5.4 An example of a bipartite network. It has two types of nodes, dark and light, with no edges between dark nodes and no edges between light nodes. The edges only connect a dark node to a light node. The two adjacency matrices show different ways to represent this network. 187

5.5 Example of a spatial network. Node are connected based on their physical or geographic distance with edge weight indicating the distance. 190

5.6 A multilayer bipartite network 191

5.7 A multilayer unipartite network 192

5.8 A flattened –or edge colored– network 192

5.9 Graph showing the connections between the companies in a return volatility series graph filtered by TMFG. 195

5.10 A network of people. How does the position of the people a_1,a_2,b_1 and b_2 differ in terms of having influence or spreading risk? 199

5.11 An external event that propagates through a network from a source node. 203

5.12 Example for computing link-based features 204

5.13 The inner workings of a GNN layer 209

5.14 Architecture for a temporal graph neural network 220

5.15 A subset of the network. Two nodes are connected if they share an area or a provider. The nodes are colored by the target variable, where light gray nodes a nodes are non-delinquent (label 0) and dark gray nodes are delinquent (label 1). 228

5.16 Loss curves for all models. 236

5.17 ROC curves for all models. 237

5.18 Confusion matrices for all models. 238

6.1 Difference between Encoder-Only and Decoder-Only Transformer Architectures 249

6.2 Architecture of a Retrieval-Augmented Generation System . 258

6.3 SHAP values of a case. A detailed explanation of the method can be found in Section 8.3.1. 274

6.4 Weights & Biases Report 282

7.1 Different information fusion strategies. In this diagram, there are three modalities (structured data, text and images) and they are either processed by a function to be combined, as in the early fusion example, or they are processed by a model that outputs an embedding, a numerical representation of the data. A final AI model then transforms these modalities into a desired final output, such as a prediction or a regression value. Adapted from Tavakoli et al. (2025). 289

7.2 Cross attention vs Self-Attention. Adapted from Tavakoli et al. (2025). 293

7.3 Concatenation Multimodal Model 307

7.4 Cross-Attention Multimodal Model 307

7.5 ROC curves for the two models. 311

7.6 Confusion Matrices for the two models. 311

8.1 A waterfall SHAP plot for a single observation. 335

8.2 A summary SHAP plot for all variables 336

8.3 A beeswarm SHAP plot for all variables and observations. . 337

8.4 A dependence SHAP plot showing the interaction between two variables. 338

8.5 A causal graph: Directed Acyclic Graph (DAG) with X as a confounder affecting both the bid B and the target Y. . . . 342

8.6 Distribution of defaulters(1) and non-defaulters(0) among the four groups of the sensitive attribute. 354

8.7 Barplots showing (from left to right, top to bottom) the count, accuracy, selection rate, precision, true positive rate, true negative rate, false positive rate and false negative rate of the credit scoring model for the four groups. 357

8.8 ROC curves per group and the feasible area 359

8.9 SHAP text explanations 364

8.10 SHAP bar explanations 364

9.1 A simple project prioritization matrix. Quick wins are models that should take priority. Opportunities are difficult models that can create “moats”. Back burners are projects that can be developed if there is spare capacity. No-Go are projects with low ROI that are also very risky to develop. 367

9.2 Some ROI factors. Be mindful of identifying quantitative and qualitative ones! 369

9.3 General inference server architecture. 371

List of Algorithms

2.1 Image Data: Data loading and preprocessing 76

2.2 Image Data: Create data loaders. 79

2.3 Image Data: Define simple CNN regressor. 80

2.4 Image Data: Train simple model. 83

2.5 Image Data: Evaluate simple model. 84

2.6 Image Data: Grad-CAM for model explainability. 85

2.7 Image Data: ResNet-50 model 89

2.8 Image Data: Train ResNet-50. 90

2.9 Image Data: Evaluate ResNet-50 91

3.1 Time-Series: Data Loading. 120

3.2 Time-Series: Data preprocessing 122

3.3 Time-Series: Create data loaders. 123

3.4 Time-Series: Create Recurrent Model. 124

3.5 Time-Series: Load static data. 126

3.6 Time-Series: Define hybrid model. 127

3.7 Time-Series: Training loop. 129

3.8 Time-Series: Validation loop. 130

3.9 Time-Series: Running the models - Hybrid GRU model . . . 131

4.1 Text data: Data loading 170

4.2 Text data: Preprocessing 171

4.3 Text data: Fine tuning a BERT model 174

4.4 Text data: Inference on new data 177

5.1 Network data: Data loading and network creation 226

5.2 Network data: Define GNN models 230

5.3 Network data: Training and validation loop 234

5.4 Network data: Model Evaluation 235

6.1 LLM: Initial Setup 272

6.2 LLM: Zero-shot prompting 276

6.3 Detailed LLM Prompt Engineering and Fine-Tuning for Credit Score Explainability 278

6.4 LLM Fine-tuning 281

6.5 Testing on new cases 282

7.1 Multimodal Learning: Load and process the data sources . . 303

7.2 Multimodal Learning: Preprocess all different channels . . . 304

7.3 Multimodal Learning: Integrate Features and Create Datasets 306

7.4 Multimodal Learning: Define the models. 308

7.5 Multimodal Learning: Model training loop. 309

7.6 Multimodal Learning: Model evaluation. 310

8.1 Model Explainability: Pre-processing, SHAP value calculation and visualization 362