Machine Learning for Text

Charu C. Aggarwal

Machine Learning for Text

Springer

Charu C. Aggarwal
IBM T. J. Watson Research Center
Yorktown Heights, NY, USA

ISBN 978-3-319-73530-6 ISBN 978-3-319-73531-3 (eBook)
https://doi.org/10.1007/978-3-319-73531-3

Library of Congress Control Number: 2018932755

Printed on acid-free paper

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

To my wife Lata, my daughter Sayani,
and my late parents Dr. Prem Sarup and Mrs. Pushplata Aggarwal.

Preface

> “ If it is true that there is always more than one way of construing a text, it is not true that all interpretations are equal.” – Paul Ricoeur

The rich area of text analytics draws ideas from information retrieval, machine learning, and natural language processing. Each of these areas is an active and vibrant field in its own right, and numerous books have been written in each of these different areas. As a result, many of these books have covered some aspects of text analytics, but they have not covered all the areas that a book on learning from text is expected to cover.

At this point, a need exists for a focussed book on machine learning from text. This book is a first attempt to integrate all the complexities in the areas of machine learning, information retrieval, and natural language processing in a holistic way, in order to create a coherent and integrated book in the area. Therefore, the chapters are divided into three categories:

1. *Fundamental algorithms and models:* Many fundamental applications in text analytics, such as matrix factorization, clustering, and classification, have uses in domains beyond text. Nevertheless, these methods need to be tailored to the specialized characteristics of text. Chapters 1 through 8 will discuss core analytical methods in the context of machine learning from text.

2. *Information retrieval and ranking:* Many aspects of information retrieval and ranking are closely related to text analytics. For example, ranking SVMs and link-based ranking are often used for learning from text. Chapter 9 will provide an overview of information retrieval methods from the point of view of text mining.

3. *Sequence- and natural language-centric text mining:* Although multidimensional representations can be used for basic applications in text analytics, the true richness of the text representation can be leveraged by treating text as sequences. Chapters 10 through 14 will discuss these advanced topics like sequence embedding, deep learning, information extraction, summarization, opinion mining, text segmentation, and event extraction.

Because of the diversity of topics covered in this book, some careful decisions have been made on the scope of coverage. A complicating factor is that many machine learning techniques depend on the use of basic natural language processing and information retrieval methodologies. This is particularly true of the sequence-centric approaches discussed in Chaps. 10 through 14 that are more closely related to natural language processing. Examples of analytical methods that rely on natural language processing include information extraction, event extraction, opinion mining, and text summarization, which frequently leverage basic natural language processing tools like linguistic parsing or part-of-speech tagging. Needless to say, natural language processing is a full fledged field in its own right (with excellent books dedicated to it). Therefore, a question arises on how much discussion should be provided on techniques that lie on the interface of natural language processing and text mining without deviating from the primary scope of this book. Our general principle in making these choices has been to focus on *mining* and *machine learning* aspects. If a specific natural language or information retrieval method (e.g., part-of-speech tagging) is not *directly* about text analytics, we have illustrated how to *use* such techniques (as black-boxes) rather than discussing the internal algorithmic details of these methods. Basic techniques like part-of-speech tagging have matured in algorithmic development, and have been commoditized to the extent that many open-source tools are available with little difference in relative performance. Therefore, we only provide working definitions of such concepts in the book, and the primary focus will be on their utility as off-the-shelf tools in mining-centric settings. The book provides pointers to the relevant books and open-source software in each chapter in order to enable additional help to the student and practitioner.

The book is written for graduate students, researchers, and practitioners. The exposition has been simplified to a large extent, so that a graduate student with a reasonable understanding of linear algebra and probability theory can understand the book easily. Numerous exercises are available along with a solution manual to aid in classroom teaching.

Throughout this book, a vector or a multidimensional data point is annotated with a bar, such as $\overline{X}$ or $\overline{y}$. A vector or multidimensional point may be denoted by either small letters or capital letters, as long as it has a bar. Vector dot products are denoted by centered dots, such as $\overline{X} \cdot \overline{Y}$. A matrix is denoted in capital letters without a bar, such as R. Throughout the book, the $n \times d$ document-term matrix is denoted by D, with n documents and d dimensions. The individual documents in D are therefore represented as d-dimensional row vectors, which are the bag-of-words representations. On the other hand, vectors with one component for each data point are usually n-dimensional column vectors. An example is the n-dimensional column vector $\overline{y}$ of class variables of n data points.

Yorktown Heights, NY, USA Charu C. Aggarwal

Acknowledgments

I would like to thank my family including my wife, daughter, and my parents for their love and support. I would also like to thank my manager Nagui Halim for his support during the writing of this book.

This book has benefitted from significant feedback and several collaborations that i have had with numerous colleagues over the years. I would like to thank Quoc Le, Chih-Jen Lin, Chandan Reddy, Saket Sathe, Shai Shalev-Shwartz, Jiliang Tang, Suhang Wang, and ChengXiang Zhai for their feedback on various portions of this book and for answering specific queries on technical matters. I would particularly like to thank Saket Sathe for commenting on several portions, and also for providing some sample output from a neural network to use in the book. For their collaborations, I would like to thank Tarek F. Abdelzaher, Jing Gao, Quanquan Gu, Manish Gupta, Jiawei Han, Alexander Hinneburg, Thomas Huang, Nan Li, Huan Liu, Ruoming Jin, Daniel Keim, Arijit Khan, Latifur Khan, Mohammad M. Masud, Jian Pei, Magda Procopiuc, Guojun Qi, Chandan Reddy, Saket Sathe, Jaideep Srivastava, Karthik Subbian, Yizhou Sun, Jiliang Tang, Min-Hsuan Tsai, Haixun Wang, Jianyong Wang, Min Wang, Suhang Wang, Joel Wolf, Xifeng Yan, Mohammed Zaki, ChengXiang Zhai, and Peixiang Zhao. I would particularly like to thank Professor ChengXiang Zhai for my earlier collaborations with him in text mining. I would also like to thank my advisor James B. Orlin for his guidance during my early years as a researcher.

Finally, I would like to thank Lata Aggarwal for helping me with some of the figures created using PowerPoint graphics in this book.

Contents

1 Machine Learning for Text: An Introduction 1
1.1 Introduction 1
1.1.1 Chapter Organization 3
1.2 What Is Special About Learning from Text? 3
1.3 Analytical Models for Text 4
1.3.1 Text Preprocessing and Similarity Computation 5
1.3.2 Dimensionality Reduction and Matrix Factorization 7
1.3.3 Text Clustering 8
1.3.3.1 Deterministic and Probabilistic Matrix Factorization Methods 8
1.3.3.2 Probabilistic Mixture Models of Documents 8
1.3.3.3 Similarity-Based Algorithms 9
1.3.3.4 Advanced Methods 9
1.3.4 Text Classification and Regression Modeling 10
1.3.4.1 Decision Trees 11
1.3.4.2 Rule-Based Classifiers 11
1.3.4.3 Naïve Bayes Classifier 11
1.3.4.4 Nearest Neighbor Classifiers 12
1.3.4.5 Linear Classifiers 12
1.3.4.6 Broader Topics in Classification 13
1.3.5 Joint Analysis of Text with Heterogeneous Data 13
1.3.6 Information Retrieval and Web Search 13
1.3.7 Sequential Language Modeling and Embeddings 13
1.3.8 Text Summarization 14
1.3.9 Information Extraction 14
1.3.10 Opinion Mining and Sentiment Analysis 14
1.3.11 Text Segmentation and Event Detection 15
1.4 Summary 15
1.5 Bibliographic Notes 15
1.5.1 Software Resources 16
1.6 Exercises 16

2 Text Preparation and Similarity Computation 17
2.1 Introduction . . . 17
2.1.1 Chapter Organization . . . 18
2.2 Raw Text Extraction and Tokenization . . . 18
2.2.1 Web-Specific Issues in Text Extraction . . . 21
2.3 Extracting Terms from Tokens . . . 21
2.3.1 Stop-Word Removal . . . 22
2.3.2 Hyphens . . . 22
2.3.3 Case Folding . . . 23
2.3.4 Usage-Based Consolidation . . . 23
2.3.5 Stemming . . . 23
2.4 Vector Space Representation and Normalization . . . 24
2.5 Similarity Computation in Text . . . 26
2.5.1 Is idf Normalization and Stemming Always Useful? . . . 28
2.6 Summary . . . 29
2.7 Bibliographic Notes . . . 29
2.7.1 Software Resources . . . 30
2.8 Exercises . . . 30

3 Matrix Factorization and Topic Modeling 31
3.1 Introduction . . . 31
3.1.1 Chapter Organization . . . 33
3.1.2 Normalizing a Two-Way Factorization into a Standardized Three-Way Factorization . . . 34
3.2 Singular Value Decomposition . . . 35
3.2.1 Example of SVD . . . 37
3.2.2 The Power Method of Implementing SVD . . . 39
3.2.3 Applications of SVD/LSA . . . 39
3.2.4 Advantages and Disadvantages of SVD/LSA . . . 40
3.3 Nonnegative Matrix Factorization . . . 41
3.3.1 Interpretability of Nonnegative Matrix Factorization . . . 43
3.3.2 Example of Nonnegative Matrix Factorization . . . 43
3.3.3 Folding in New Documents . . . 45
3.3.4 Advantages and Disadvantages of Nonnegative Matrix Factorization . . . 46
3.4 Probabilistic Latent Semantic Analysis . . . 46
3.4.1 Connections with Nonnegative Matrix Factorization . . . 50
3.4.2 Comparison with SVD . . . 50
3.4.3 Example of PLSA . . . 51
3.4.4 Advantages and Disadvantages of PLSA . . . 51
3.5 A Bird's Eye View of Latent Dirichlet Allocation . . . 52
3.5.1 Simplified LDA Model . . . 52
3.5.2 Smoothed LDA Model . . . 55
3.6 Nonlinear Transformations and Feature Engineering . . . 56
3.6.1 Choosing a Similarity Function . . . 59
3.6.1.1 Traditional Kernel Similarity Functions . . . 59
3.6.1.2 Generalizing Bag-of-Words to N-Grams . . . 62
3.6.1.3 String Subsequence Kernels . . . 62

3.6.1.4 Speeding Up the Recursion . . . 65
3.6.1.5 Language-Dependent Kernels . . . 65
3.6.2 Nyström Approximation . . . 66
3.6.3 Partial Availability of the Similarity Matrix . . . 67
3.7 Summary . . . 69
3.8 Bibliographic Notes . . . 70
3.8.1 Software Resources . . . 70
3.9 Exercises . . . 71

4 Text Clustering **73**
4.1 Introduction . . . 73
4.1.1 Chapter Organization . . . 74
4.2 Feature Selection and Engineering . . . 75
4.2.1 Feature Selection . . . 75
4.2.1.1 Term Strength . . . 75
4.2.1.2 Supervised Modeling for Unsupervised Feature Selection . . . 76
4.2.1.3 Unsupervised Wrappers with Supervised Feature Selection . . . 76
4.2.2 Feature Engineering . . . 77
4.2.2.1 Matrix Factorization Methods . . . 77
4.2.2.2 Nonlinear Dimensionality Reduction . . . 78
4.2.2.3 Word Embeddings . . . 78
4.3 Topic Modeling and Matrix Factorization . . . 79
4.3.1 Mixed Membership Models and Overlapping Clusters . . . 79
4.3.2 Non-overlapping Clusters and Co-clustering: A Matrix Factorization View . . . 79
4.3.2.1 Co-clustering by Bipartite Graph Partitioning . . . 82
4.4 Generative Mixture Models for Clustering . . . 83
4.4.1 The Bernoulli Model . . . 84
4.4.2 The Multinomial Model . . . 86
4.4.3 Comparison with Mixed Membership Topic Models . . . 87
4.4.4 Connections with Naïve Bayes Model for Classification . . . 88
4.5 The k-Means Algorithm . . . 88
4.5.1 Convergence and Initialization . . . 91
4.5.2 Computational Complexity . . . 91
4.5.3 Connection with Probabilistic Models . . . 91
4.6 Hierarchical Clustering Algorithms . . . 92
4.6.1 Efficient Implementation and Computational Complexity . . . 94
4.6.2 The Natural Marriage with k-Means . . . 96
4.7 Clustering Ensembles . . . 97
4.7.1 Choosing the Ensemble Component . . . 97
4.7.2 Combining the Results from Different Components . . . 98
4.8 Clustering Text as Sequences . . . 98
4.8.1 Kernel Methods for Clustering . . . 99
4.8.1.1 Kernel k-Means . . . 99
4.8.1.2 Explicit Feature Engineering . . . 100
4.8.1.3 Kernel Trick or Explicit Feature Engineering? . . . 101
4.8.2 Data-Dependent Kernels: Spectral Clustering . . . 102

4.9 Transforming Clustering into Supervised Learning . . . 104
4.9.1 Practical Issues . . . 105
4.10 Clustering Evaluation . . . 105
4.10.1 The Pitfalls of Internal Validity Measures . . . 105
4.10.2 External Validity Measures . . . 105
4.10.2.1 Relationship of Clustering Evaluation to Supervised Learning . . . 109
4.10.2.2 Common Mistakes in Evaluation . . . 109
4.11 Summary . . . 110
4.12 Bibliographic Notes . . . 110
4.12.1 Software Resources . . . 111
4.13 Exercises . . . 111

5 Text Classification: Basic Models 113
5.1 Introduction . . . 113
5.1.1 Types of Labels and Regression Modeling . . . 114
5.1.2 Training and Testing . . . 115
5.1.3 Inductive, Transductive, and Deductive Learners . . . 116
5.1.4 The Basic Models . . . 117
5.1.5 Text-Specific Challenges in Classifiers . . . 117
5.1.5.1 Chapter Organization . . . 117
5.2 Feature Selection and Engineering . . . 117
5.2.1 Gini Index . . . 118
5.2.2 Conditional Entropy . . . 119
5.2.3 Pointwise Mutual Information . . . 119
5.2.4 Closely Related Measures . . . 119
5.2.5 The χ^2-Statistic . . . 120
5.2.6 Embedded Feature Selection Models . . . 122
5.2.7 Feature Engineering Tricks . . . 122
5.3 The Naïve Bayes Model . . . 123
5.3.1 The Bernoulli Model . . . 123
5.3.1.1 Prediction Phase . . . 124
5.3.1.2 Training Phase . . . 125
5.3.2 Multinomial Model . . . 126
5.3.3 Practical Observations . . . 127
5.3.4 Ranking Outputs with Naïve Bayes . . . 127
5.3.5 Example of Naïve Bayes . . . 128
5.3.5.1 Bernoulli Model . . . 128
5.3.5.2 Multinomial Model . . . 130
5.3.6 Semi-Supervised Naïve Bayes . . . 131
5.4 Nearest Neighbor Classifier . . . 133
5.4.1 Properties of 1-Nearest Neighbor Classifiers . . . 134
5.4.2 Rocchio and Nearest Centroid Classification . . . 136
5.4.3 Weighted Nearest Neighbors . . . 137
5.4.3.1 Bagged and Subsampled 1-Nearest Neighbors as Weighted Nearest Neighbor Classifiers . . . 138
5.4.4 Adaptive Nearest Neighbors: A Powerful Family . . . 140
5.5 Decision Trees and Random Forests . . . 142
5.5.1 Basic Procedure for Decision Tree Construction . . . 142

5.5.2 Splitting a Node . . . 143
5.5.2.1 Prediction . . . 144
5.5.3 Multivariate Splits . . . 144
5.5.4 Problematic Issues with Decision Trees in Text Classification . . . 145
5.5.5 Random Forests . . . 146
5.5.6 Random Forests as Adaptive Nearest Neighbor Methods . . . 147
5.6 Rule-Based Classifiers . . . 147
5.6.1 Sequential Covering Algorithms . . . 148
5.6.1.1 Learn-One-Rule . . . 149
5.6.1.2 Rule Pruning . . . 150
5.6.2 Generating Rules from Decision Trees . . . 150
5.6.3 Associative Classifiers . . . 151
5.6.4 Prediction . . . 152
5.7 Summary . . . 152
5.8 Bibliographic Notes . . . 153
5.8.1 Software Resources . . . 154
5.9 Exercises . . . 154

6 Linear Classification and Regression for Text 159
6.1 Introduction . . . 159
6.1.1 Geometric Interpretation of Linear Models . . . 160
6.1.2 Do We Need the Bias Variable? . . . 161
6.1.3 A General Definition of Linear Models with Regularization . . . 162
6.1.4 Generalizing Binary Predictions to Multiple Classes . . . 163
6.1.5 Characteristics of Linear Models for Text . . . 164
6.1.5.1 Chapter Notations . . . 165
6.1.5.2 Chapter Organization . . . 165
6.2 Least-Squares Regression and Classification . . . 165
6.2.1 Least-Squares Regression with L_2-Regularization . . . 165
6.2.1.1 Efficient Implementation . . . 166
6.2.1.2 Approximate Estimation with Singular Value Decomposition . . . 167
6.2.1.3 Relationship with Principal Components Regression . . . 167
6.2.1.4 The Path to Kernel Regression . . . 168
6.2.2 LASSO: Least-Squares Regression with L_1-Regularization . . . 169
6.2.2.1 Interpreting LASSO as a Feature Selector . . . 170
6.2.3 Fisher's Linear Discriminant and Least-Squares Classification . . . 170
6.2.3.1 Linear Discriminant with Multiple Classes . . . 173
6.2.3.2 Equivalence of Fisher Discriminant and Least-Squares Regression . . . 173
6.2.3.3 Regularized Least-Squares Classification and LLSF . . . 175
6.2.3.4 The Achilles Heel of Least-Squares Classification . . . 176
6.3 Support Vector Machines . . . 177
6.3.1 The Regularized Optimization Interpretation . . . 178
6.3.2 The Maximum Margin Interpretation . . . 179
6.3.3 Pegasos: Solving SVMs in the Primal . . . 180
6.3.3.1 Sparsity-Friendly Updates . . . 181
6.3.4 Dual SVM Formulation . . . 182

6.3.5 Learning Algorithms for Dual SVMs 184
6.3.6 Adaptive Nearest Neighbor Interpretation of Dual SVMs 185
6.4 Logistic Regression 187
6.4.1 The Regularized Optimization Interpretation 187
6.4.2 Training Algorithms for Logistic Regression 189
6.4.3 Probabilistic Interpretation of Logistic Regression 189
6.4.3.1 Probabilistic Interpretation of Stochastic Gradient Descent Steps 190
6.4.3.2 Relationships Among Primal Updates of Linear Models . 191
6.4.4 Multinomial Logistic Regression and Other Generalizations 191
6.4.5 Comments on the Performance of Logistic Regression 192
6.5 Nonlinear Generalizations of Linear Models 193
6.5.1 Kernel SVMs with Explicit Transformation 195
6.5.2 Why Do Conventional Kernels Promote Linear Separability? 196
6.5.3 Strengths and Weaknesses of Different Kernels 197
6.5.3.1 Capturing Linguistic Knowledge with Kernels 198
6.5.4 The Kernel Trick 198
6.5.5 Systematic Application of the Kernel Trick 199
6.6 Summary 203
6.7 Bibliographic Notes 203
6.7.1 Software Resources 204
6.8 Exercises 205

7 Classifier Performance and Evaluation 209
7.1 Introduction 209
7.1.1 Chapter Organization 210
7.2 The Bias-Variance Trade-Off 210
7.2.1 A Formal View 211
7.2.2 Telltale Signs of Bias and Variance 214
7.3 Implications of Bias-Variance Trade-Off on Performance 215
7.3.1 Impact of Training Data Size 215
7.3.2 Impact of Data Dimensionality 217
7.3.3 Implications for Model Choice in Text 217
7.4 Systematic Performance Enhancement with Ensembles 218
7.4.1 Bagging and Subsampling 218
7.4.2 Boosting 220
7.5 Classifier Evaluation 221
7.5.1 Segmenting into Training and Testing Portions 222
7.5.1.1 Hold-Out 223
7.5.1.2 Cross-Validation 224
7.5.2 Absolute Accuracy Measures 224
7.5.2.1 Accuracy of Classification 224
7.5.2.2 Accuracy of Regression 225
7.5.3 Ranking Measures for Classification and Information Retrieval . . . 226
7.5.3.1 Receiver Operating Characteristic 227
7.5.3.2 Top-Heavy Measures for Ranked Lists 231
7.6 Summary 232
7.7 Bibliographic Notes 232
7.7.1 Connection of Boosting to Logistic Regression 232

7.7.2 Classifier Evaluation 233
7.7.3 Software Resources 233
7.7.4 Data Sets for Evaluation 233
7.8 Exercises 234

8 Joint Text Mining with Heterogeneous Data 235
8.1 Introduction 235
8.1.1 Chapter Organization 237
8.2 The Shared Matrix Factorization Trick 237
8.2.1 The Factorization Graph 237
8.2.2 Application: Shared Factorization with Text and Web Links 238
8.2.2.1 Solving the Optimization Problem 240
8.2.2.2 Supervised Embeddings 241
8.2.3 Application: Text with Undirected Social Networks 242
8.2.3.1 Application to Link Prediction with Text Content 243
8.2.4 Application: Transfer Learning in Images with Text 243
8.2.4.1 Transfer Learning with Unlabeled Text 244
8.2.4.2 Transfer Learning with Labeled Text 245
8.2.5 Application: Recommender Systems with Ratings and Text 246
8.2.6 Application: Cross-Lingual Text Mining 248
8.3 Factorization Machines 249
8.4 Joint Probabilistic Modeling Techniques 252
8.4.1 Joint Probabilistic Models for Clustering 253
8.4.2 Naïve Bayes Classifier 254
8.5 Transformation to Graph Mining Techniques 254
8.6 Summary 257
8.7 Bibliographic Notes 257
8.7.1 Software Resources 258
8.8 Exercises 258

9 Information Retrieval and Search Engines 259
9.1 Introduction 259
9.1.1 Chapter Organization 260
9.2 Indexing and Query Processing 260
9.2.1 Dictionary Data Structures 261
9.2.2 Inverted Index 263
9.2.3 Linear Time Index Construction 264
9.2.4 Query Processing 266
9.2.4.1 Boolean Retrieval 266
9.2.4.2 Ranked Retrieval 267
9.2.4.3 Term-at-a-Time Query Processing with Accumulators . . 268
9.2.4.4 Document-at-a-Time Query Processing with Accumulators 270
9.2.4.5 Term-at-a-Time or Document-at-a-Time? 270
9.2.4.6 What Types of Scores Are Common? 271
9.2.4.7 Positional Queries 271
9.2.4.8 Zoned Scoring 272
9.2.4.9 Machine Learning in Information Retrieval 273
9.2.4.10 Ranking Support Vector Machines 274

9.2.5 Efficiency Optimizations . . . 276
9.2.5.1 Skip Pointers . . . 276
9.2.5.2 Champion Lists and Tiered Indexes . . . 277
9.2.5.3 Caching Tricks . . . 277
9.2.5.4 Compression Tricks . . . 278
9.3 Scoring with Information Retrieval Models . . . 280
9.3.1 Vector Space Models with tf-idf . . . 280
9.3.2 The Binary Independence Model . . . 281
9.3.3 The BM25 Model with Term Frequencies . . . 283
9.3.4 Statistical Language Models in Information Retrieval . . . 285
9.3.4.1 Query Likelihood Models . . . 285
9.4 Web Crawling and Resource Discovery . . . 287
9.4.1 A Basic Crawler Algorithm . . . 287
9.4.2 Preferential Crawlers . . . 289
9.4.3 Multiple Threads . . . 290
9.4.4 Combatting Spider Traps . . . 290
9.4.5 Shingling for Near Duplicate Detection . . . 291
9.5 Query Processing in Search Engines . . . 291
9.5.1 Distributed Index Construction . . . 292
9.5.2 Dynamic Index Updates . . . 293
9.5.3 Query Processing . . . 293
9.5.4 The Importance of Reputation . . . 294
9.6 Link-Based Ranking Algorithms . . . 295
9.6.1 PageRank . . . 295
9.6.1.1 Topic-Sensitive PageRank . . . 298
9.6.1.2 SimRank . . . 299
9.6.2 HITS . . . 300
9.7 Summary . . . 302
9.8 Bibliographic Notes . . . 302
9.8.1 Software Resources . . . 303
9.9 Exercises . . . 304

10 Text Sequence Modeling and Deep Learning 305
10.1 Introduction . . . 305
10.1.1 Chapter Organization . . . 308
10.2 Statistical Language Models . . . 308
10.2.1 Skip-Gram Models . . . 310
10.2.2 Relationship with Embeddings . . . 312
10.3 Kernel Methods . . . 313
10.4 Word-Context Matrix Factorization Models . . . 314
10.4.1 Matrix Factorization with Counts . . . 314
10.4.1.1 Postprocessing Issues . . . 316
10.4.2 The GloVe Embedding . . . 316
10.4.3 PPMI Matrix Factorization . . . 317
10.4.4 Shifted PPMI Matrix Factorization . . . 318
10.4.5 Incorporating Syntactic and Other Features . . . 318
10.5 Graphical Representations of Word Distances . . . 318

10.6 Neural Language Models 320
10.6.1 Neural Networks: A Gentle Introduction 320
10.6.1.1 Single Computational Layer: The Perceptron 321
10.6.1.2 Relationship to Support Vector Machines 323
10.6.1.3 Choice of Activation Function 324
10.6.1.4 Choice of Output Nodes 325
10.6.1.5 Choice of Loss Function 325
10.6.1.6 Multilayer Neural Networks 326
10.6.2 Neural Embedding with Word2vec 331
10.6.2.1 Neural Embedding with Continuous Bag of Words 331
10.6.2.2 Neural Embedding with Skip-Gram Model 334
10.6.2.3 Practical Issues 336
10.6.2.4 Skip-Gram with Negative Sampling 337
10.6.2.5 What Is the Actual Neural Architecture of SGNS? 338
10.6.3 Word2vec (SGNS) Is Logistic Matrix Factorization 338
10.6.3.1 Gradient Descent 340
10.6.4 Beyond Words: Embedding Paragraphs with Doc2vec 341
10.7 Recurrent Neural Networks 342
10.7.1 Practical Issues 345
10.7.2 Language Modeling Example of RNN 345
10.7.2.1 Generating a Language Sample 345
10.7.3 Application to Automatic Image Captioning 347
10.7.4 Sequence-to-Sequence Learning and Machine Translation 348
10.7.4.1 Question-Answering Systems 350
10.7.5 Application to Sentence-Level Classification 352
10.7.6 Token-Level Classification with Linguistic Features 353
10.7.7 Multilayer Recurrent Networks 354
10.7.7.1 Long Short-Term Memory (LSTM) 355
10.8 Summary 357
10.9 Bibliographic Notes 357
10.9.1 Software Resources 358
10.10 Exercises 359

11 Text Summarization 361
11.1 Introduction 361
11.1.1 Extractive and Abstractive Summarization 362
11.1.2 Key Steps in Extractive Summarization 363
11.1.3 The Segmentation Phase in Extractive Summarization 363
11.1.4 Chapter Organization 363
11.2 Topic Word Methods for Extractive Summarization 364
11.2.1 Word Probabilities 364
11.2.2 Normalized Frequency Weights 365
11.2.3 Topic Signatures 366
11.2.4 Sentence Selection Methods 368
11.3 Latent Methods for Extractive Summarization 369
11.3.1 Latent Semantic Analysis 369
11.3.2 Lexical Chains 370
11.3.2.1 Short Description of WordNet 370
11.3.2.2 Leveraging WordNet for Lexical Chains 371

11.3.3 Graph-Based Methods . . . 372
11.3.4 Centroid Summarization . . . 373
11.4 Machine Learning for Extractive Summarization . . . 374
11.4.1 Feature Extraction . . . 374
11.4.2 Which Classifiers to Use? . . . 375
11.5 Multi-Document Summarization . . . 375
11.5.1 Centroid-Based Summarization . . . 375
11.5.2 Graph-Based Methods . . . 376
11.6 Abstractive Summarization . . . 377
11.6.1 Sentence Compression . . . 378
11.6.2 Information Fusion . . . 378
11.6.3 Information Ordering . . . 379
11.7 Summary . . . 379
11.8 Bibliographic Notes . . . 379
11.8.1 Software Resources . . . 380
11.9 Exercises . . . 380

12 Information Extraction 381
12.1 Introduction . . . 381
12.1.1 Historical Evolution . . . 383
12.1.2 The Role of Natural Language Processing . . . 384
12.1.3 Chapter Organization . . . 385
12.2 Named Entity Recognition . . . 386
12.2.1 Rule-Based Methods . . . 387
12.2.1.1 Training Algorithms for Rule-Based Systems . . . 388
12.2.1.2 Top-Down Rule Generation . . . 389
12.2.1.3 Bottom-Up Rule Generation . . . 390
12.2.2 Transformation to Token-Level Classification . . . 391
12.2.3 Hidden Markov Models . . . 391
12.2.3.1 Visible Versus Hidden Markov Models . . . 392
12.2.3.2 The Nymble System . . . 392
12.2.3.3 Training . . . 394
12.2.3.4 Prediction for Test Segment . . . 394
12.2.3.5 Incorporating Extracted Features . . . 395
12.2.3.6 Variations and Enhancements . . . 395
12.2.4 Maximum Entropy Markov Models . . . 396
12.2.5 Conditional Random Fields . . . 397
12.3 Relationship Extraction . . . 399
12.3.1 Transformation to Classification . . . 400
12.3.2 Relationship Prediction with Explicit Feature Engineering . . . 401
12.3.2.1 Feature Extraction from Sentence Sequences . . . 402
12.3.2.2 Simplifying Parse Trees with Dependency Graphs . . . 403
12.3.3 Relationship Prediction with Implicit Feature Engineering: Kernel Methods . . . 404
12.3.3.1 Kernels from Dependency Graphs . . . 405
12.3.3.2 Subsequence-Based Kernels . . . 405
12.3.3.3 Convolution Tree-Based Kernels . . . 406
12.4 Summary . . . 408

12.5 Bibliographic Notes . . . 409
12.5.1 Weakly Supervised Learning Methods . . . 410
12.5.2 Unsupervised and Open Information Extraction . . . 410
12.5.3 Software Resources . . . 410
12.6 Exercises . . . 411

13 Opinion Mining and Sentiment Analysis 413
13.1 Introduction . . . 413
13.1.1 The Opinion Lexicon . . . 415
13.1.1.1 Dictionary-Based Approaches . . . 416
13.1.1.2 Corpus-Based Approaches . . . 416
13.1.2 Opinion Mining as a Slot Filling and Information Extraction Task . 417
13.1.3 Chapter Organization . . . 418
13.2 Document-Level Sentiment Classification . . . 418
13.2.1 Unsupervised Approaches to Classification . . . 420
13.3 Phrase- and Sentence-Level Sentiment Classification . . . 421
13.3.1 Applications of Sentence- and Phrase-Level Analysis . . . 422
13.3.2 Reduction of Subjectivity Classification to Minimum Cut Problem 423
13.3.3 Context in Sentence- and Phrase-Level Polarity Analysis . . . 423
13.4 Aspect-Based Opinion Mining as Information Extraction . . . 424
13.4.1 Hu and Liu's Unsupervised Approach . . . 424
13.4.2 OPINE: An Unsupervised Approach . . . 426
13.4.3 Supervised Opinion Extraction as Token-Level Classification . . . 427
13.5 Opinion Spam . . . 428
13.5.1 Supervised Methods for Spam Detection . . . 428
13.5.1.1 Labeling Deceptive Spam . . . 429
13.5.1.2 Feature Extraction . . . 430
13.5.2 Unsupervised Methods for Spammer Detection . . . 431
13.6 Opinion Summarization . . . 431
13.6.1 Rating Summary . . . 432
13.6.2 Sentiment Summary . . . 432
13.6.3 Sentiment Summary with Phrases and Sentences . . . 432
13.6.4 Extractive and Abstractive Summaries . . . 432
13.7 Summary . . . 433
13.8 Bibliographic Notes . . . 433
13.8.1 Software Resources . . . 434
13.9 Exercises . . . 434

14 Text Segmentation and Event Detection 435
14.1 Introduction . . . 435
14.1.1 Relationship with Topic Detection and Tracking . . . 436
14.1.2 Chapter Organization . . . 436
14.2 Text Segmentation . . . 436
14.2.1 TextTiling . . . 437
14.2.2 The C99 Approach . . . 438
14.2.3 Supervised Segmentation with Off-the-Shelf Classifiers . . . 439
14.2.4 Supervised Segmentation with Markovian Models . . . 441

14.3 Mining Text Streams 443
14.3.1 Streaming Text Clustering 443
14.3.2 Application to First Story Detection 444
14.4 Event Detection 445
14.4.1 Unsupervised Event Detection 445
14.4.1.1 Window-Based Nearest-Neighbor Method 445
14.4.1.2 Leveraging Generative Models 446
14.4.1.3 Event Detection in Social Streams 447
14.4.2 Supervised Event Detection as Supervised Segmentation 447
14.4.3 Event Detection as an Information Extraction Problem 448
14.4.3.1 Transformation to Token-Level Classification 448
14.4.3.2 Open Domain Event Extraction 449
14.5 Summary 451
14.6 Bibliographic Notes 451
14.6.1 Software Resources 451
14.7 Exercises 452

Bibliography **453**

Index **489**

Author Biography

Charu C. Aggarwal is a Distinguished Research Staff Member (DRSM) at the IBM T. J. Watson Research Center in Yorktown Heights, New York. He completed his undergraduate degree in Computer Science from the Indian Institute of Technology at Kanpur in 1993 and his Ph.D. from the Massachusetts Institute of Technology in 1996. He has worked extensively in the field of data mining. He has published more than 350 papers in refereed conferences and journals and authored over 80 patents. He is the author or editor of 17 books, including textbooks on data mining, recommender systems, and outlier analysis. Because of the commercial value of his patents, he has thrice been designated a Master Inventor at IBM. He is a recipient of an IBM Corporate Award (2003) for his work on bio-terrorist threat detection in data streams, a recipient of the IBM Outstanding Innovation Award (2008) for his scientific contributions to privacy technology, and a recipient of two IBM Outstanding Technical Achievement Awards (2009, 2015) for his work on data streams/high-dimensional data. He received the EDBT 2014 Test of Time Award for his work on condensation-based privacy-preserving data mining. He is also a recipient of the IEEE ICDM Research Contributions Award (2015), which is one of the two highest awards for influential research contributions in the field of data mining.

He has served as the general co-chair of the IEEE Big Data Conference (2014) and as the program co-chair of the ACM CIKM Conference (2015), the IEEE ICDM Conference (2015), and the ACM KDD Conference (2016). He served as an associate editor of the IEEE Transactions on Knowledge and Data Engineering from 2004 to 2008. He is an associate editor of the IEEE Transactions on Big Data, an action editor of the Data Mining and Knowledge Discovery Journal, and an associate editor of the Knowledge and Information Systems Journal. He serves as the editor-in-chief of the ACM Transactions on Knowledge Discovery from Data as well as the ACM SIGKDD Explorations. He serves on the advisory board of the Lecture Notes on Social Networks, a publication by Springer. He has served as the vice-president of the SIAM Activity Group on Data Mining and is a member of the SIAM industry committee. He is a fellow of the SIAM, ACM, and the IEEE, for "contributions to knowledge discovery and data mining algorithms."