

# Machine Learning for Text

Charu C. Aggarwal

# Machine Learning for Text

Second Edition

 Springer

Charu C. Aggarwal  
Mohegan Lake  
NY, USA

A Solution Manual to this book can be downloaded from <https://link.springer.com/book/10.1007/978-3-030-96623-2>

ISBN 978-3-030-96622-5      ISBN 978-3-030-96623-2 (eBook)  
<https://doi.org/10.1007/978-3-030-96623-2>

1<sup>st</sup> edition: © Springer International Publishing AG, part of Springer Nature 2018

2<sup>nd</sup> edition: © Springer Nature Switzerland AG 2022, corrected publication 2023

This work is subject to copyright. All rights are reserved 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

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 models*: 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 16 will discuss these advanced topics like sequence embedding, deep learning, transformers, pre-trained language models, information extraction, knowledge graphs, summarization, question-answering, 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 Chapters 10

through 16 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  $\bar{X}$  or  $\bar{y}$ . 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  $\bar{X} \cdot \bar{Y}$ . 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  $\bar{y}$  of class variables of  $n$  data points.

## What Is New in the Second Edition

The second edition of the book emphasizes deep learning and natural language processing. Chapter 10 on deep learning has been significantly enhanced with discussions on different types of neural networks as well as language models like ELMo. Chapter 11 is entirely new, and it discusses transformers and pre-trained language models. Deep learning methods have been added to the chapter on text summarization with a special focus on abstractive summarization (Chapter 12). The information extraction chapter has now been updated to an integrated chapter on information extraction and knowledge graphs. The addition of knowledge graphs also lays the ground for a completely new chapter on question-answering (Chapter 14). Deep learning methods for sentiment analysis are also introduced in the book.

Mohegan Lake, NY, USA

Charu C. Aggarwal

---

---

# Acknowledgments

---

---

## Acknowledgements for the First Edition

---

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.

## Acknowledgements for the Second Edition

---

I would like to thank Roy Lee, Chandan Reddy, and Jiliang Tang for their feedback on the second edition of the book.

---

---

# Contents

---

---

<b>1</b>	<b>An Introduction to Text Analytics</b>	<b>1</b>
1.1	Introduction	1
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	Transformers and Pretrained Language Models	14
1.3.9	Text Summarization	14
1.3.10	Information Extraction	15
1.3.11	Question Answering	15
1.3.12	Opinion Mining and Sentiment Analysis	15
1.3.13	Text Segmentation and Event Detection	16
1.4	Summary	16
1.5	Bibliographic Notes	16
1.5.1	Software Resources	17
1.6	Exercises	17



<b>2</b>	<b>Text Preparation and Similarity Computation</b>	<b>19</b>
2.1	Introduction . . . . .	19
2.2	Raw Text Extraction and Tokenization . . . . .	20
2.2.1	Web-Specific Issues in Text Extraction . . . . .	23
2.3	Extracting Terms from Tokens . . . . .	23
2.3.1	Stop-Word Removal . . . . .	24
2.3.2	Hyphens . . . . .	24
2.3.3	Case Folding . . . . .	25
2.3.4	Usage-Based Consolidation . . . . .	25
2.3.5	Stemming . . . . .	25
2.4	Vector Space Representation and Normalization . . . . .	26
2.5	Similarity Computation in Text . . . . .	28
2.5.1	Is idf Normalization and Stemming Always Useful? . . . . .	30
2.6	Summary . . . . .	31
2.7	Bibliographic Notes . . . . .	31
2.7.1	Software Resources . . . . .	32
2.8	Exercises . . . . .	32
<b>3</b>	<b>Matrix Factorization and Topic Modeling</b>	<b>33</b>
3.1	Introduction . . . . .	33
3.1.1	Normalizing a Two-Way Factorization into a Standardized Three-Way Factorization . . . . .	35
3.2	Singular Value Decomposition . . . . .	37
3.2.1	Example of SVD . . . . .	39
3.2.2	The Power Method of Implementing SVD . . . . .	41
3.2.3	Applications of SVD/LSA . . . . .	41
3.2.4	Advantages and Disadvantages of SVD/LSA . . . . .	42
3.3	Nonnegative Matrix Factorization . . . . .	43
3.3.1	Interpretability of Nonnegative Matrix Factorization . . . . .	45
3.3.2	Example of Nonnegative Matrix Factorization . . . . .	45
3.3.3	Folding in New Documents . . . . .	47
3.3.4	Advantages and Disadvantages of Nonnegative Matrix Factorization . . . . .	48
3.4	Probabilistic Latent Semantic Analysis . . . . .	48
3.4.1	Connections with Nonnegative Matrix Factorization . . . . .	52
3.4.2	Comparison with SVD . . . . .	52
3.4.3	Example of PLSA . . . . .	53
3.4.4	Advantages and Disadvantages of PLSA . . . . .	53
3.5	A Bird's Eye View of Latent Dirichlet Allocation . . . . .	54
3.5.1	Simplified LDA Model . . . . .	54
3.5.2	Smoothed LDA Model . . . . .	57
3.6	Nonlinear Transformations and Feature Engineering . . . . .	58
3.6.1	Choosing a Similarity Function . . . . .	61
3.6.1.1	Traditional Kernel Similarity Functions . . . . .	61
3.6.1.2	Generalizing Bag-of-Words to $N$ -Grams . . . . .	64
3.6.1.3	String Subsequence Kernels . . . . .	64
3.6.1.4	Speeding Up the Recursion . . . . .	67
3.6.1.5	Language-Dependent Kernels . . . . .	68

3.6.2	Nyström Approximation . . . . .	68
3.6.3	Partial Availability of the Similarity Matrix . . . . .	70
3.7	Summary . . . . .	71
3.8	Bibliographic Notes . . . . .	72
3.8.1	Software Resources . . . . .	72
3.9	Exercises . . . . .	73
<b>4</b>	<b>Text Clustering</b>	<b>75</b>
4.1	Introduction . . . . .	75
4.2	Feature Selection and Engineering . . . . .	77
4.2.1	Feature Selection . . . . .	77
4.2.1.1	Term Strength . . . . .	77
4.2.1.2	Supervised Modeling for Unsupervised Feature Selection . . . . .	78
4.2.1.3	Unsupervised Wrappers with Supervised Feature Selection . . . . .	78
4.2.2	Feature Engineering . . . . .	79
4.2.2.1	Matrix Factorization Methods . . . . .	79
4.2.2.2	Nonlinear Dimensionality Reduction . . . . .	80
4.3	Topic Modeling and Matrix Factorization . . . . .	81
4.3.1	Mixed Membership Models and Overlapping Clusters . . . . .	81
4.3.2	Non-overlapping Clusters and Co-clustering: A Matrix Factorization View . . . . .	81
4.3.2.1	Co-clustering by Bipartite Graph Partitioning . . . . .	84
4.4	Generative Mixture Models for Clustering . . . . .	86
4.4.1	The Bernoulli Model . . . . .	86
4.4.2	The Multinomial Model . . . . .	88
4.4.3	Comparison with Mixed Membership Topic Models . . . . .	89
4.4.4	Connections with Naïve Bayes Model for Classification . . . . .	90
4.5	The $k$ -Means Algorithm . . . . .	90
4.5.1	Convergence and Initialization . . . . .	93
4.5.2	Computational Complexity . . . . .	93
4.5.3	Connection with Probabilistic Models . . . . .	93
4.6	Hierarchical Clustering Algorithms . . . . .	94
4.6.1	Efficient Implementation and Computational Complexity . . . . .	96
4.6.2	The Natural Marriage with $k$ -Means . . . . .	98
4.7	Clustering Ensembles . . . . .	99
4.7.1	Choosing the Ensemble Component . . . . .	99
4.7.2	Combining the Results from Different Components . . . . .	100
4.8	Clustering Text as Sequences . . . . .	100
4.8.1	Kernel Methods for Clustering . . . . .	101
4.8.1.1	Kernel $k$ -Means . . . . .	101
4.8.1.2	Explicit Feature Engineering . . . . .	102
4.8.1.3	Kernel Trick or Explicit Feature Engineering? . . . . .	103
4.8.2	Data-Dependent Kernels: Spectral Clustering . . . . .	104
4.9	Transforming Clustering into Supervised Learning . . . . .	106
4.10	Clustering Evaluation . . . . .	107
4.10.1	The Pitfalls of Internal Validity Measures . . . . .	107
4.10.2	External Validity Measures . . . . .	107

	4.10.2.1 Relationship of Clustering Evaluation to Supervised Learning . . . . .	111
	4.10.2.2 Common Mistakes in Evaluation . . . . .	111
4.11	Summary . . . . .	112
4.12	Bibliographic Notes . . . . .	112
	4.12.1 Software Resources . . . . .	113
4.13	Exercises . . . . .	113
<b>5</b>	<b>Text Classification: Basic Models</b>	<b>115</b>
5.1	Introduction . . . . .	115
	5.1.1 Types of Labels and Regression Modeling . . . . .	116
	5.1.2 Training and Testing . . . . .	117
	5.1.3 Inductive, Transductive, and Deductive Learners . . . . .	118
	5.1.4 The Basic Models . . . . .	119
	5.1.5 Text-Specific Challenges in Classifiers . . . . .	119
5.2	Feature Selection and Engineering . . . . .	119
	5.2.1 Gini Index . . . . .	120
	5.2.2 Conditional Entropy . . . . .	121
	5.2.3 Pointwise Mutual Information . . . . .	121
	5.2.4 Closely Related Measures . . . . .	121
	5.2.5 The $\chi^2$ -Statistic . . . . .	122
	5.2.6 Embedded Feature Selection Models . . . . .	124
	5.2.7 Feature Engineering Tricks . . . . .	124
5.3	The Naïve Bayes Model . . . . .	125
	5.3.1 The Bernoulli Model . . . . .	125
	5.3.2 Multinomial Model . . . . .	128
	5.3.3 Practical Observations . . . . .	129
	5.3.4 Ranking Outputs with Naïve Bayes . . . . .	129
	5.3.5 Example of Naïve Bayes . . . . .	130
	5.3.5.1 Bernoulli Model . . . . .	130
	5.3.5.2 Multinomial Model . . . . .	132
	5.3.6 Semi-Supervised Naïve Bayes . . . . .	133
5.4	Nearest Neighbor Classifier . . . . .	135
	5.4.1 Properties of 1-Nearest Neighbor Classifiers . . . . .	136
	5.4.2 Rocchio and Nearest Centroid Classification . . . . .	138
	5.4.3 Weighted Nearest Neighbors . . . . .	140
	5.4.3.1 Bagged and Subsampled 1-Nearest Neighbors as Weighted Nearest Neighbor Classifiers . . . . .	141
	5.4.4 Adaptive Nearest Neighbors: A Powerful Family . . . . .	142
5.5	Decision Trees and Random Forests . . . . .	144
	5.5.1 Basic Procedure for Decision Tree Construction . . . . .	144
	5.5.2 Splitting a Node . . . . .	145
	5.5.3 Multivariate Splits . . . . .	146
	5.5.4 Problematic Issues with Decision Trees in Text Classification . . . . .	147
	5.5.5 Random Forests . . . . .	148
	5.5.6 Random Forests as Adaptive Nearest Neighbor Methods . . . . .	149
5.6	Rule-Based Classifiers . . . . .	150
	5.6.1 Sequential Covering Algorithms . . . . .	150
	5.6.1.1 Learn-One-Rule . . . . .	151

5.6.2	Generating Rules from Decision Trees . . . . .	152
5.6.3	Associative Classifiers . . . . .	153
5.7	Summary . . . . .	154
5.8	Bibliographic Notes . . . . .	155
5.8.1	Software Resources . . . . .	156
5.9	Exercises . . . . .	156
<b>6</b>	<b>Linear Models for Classification and Regression</b>	<b>159</b>
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
6.2	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	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
6.3.4	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 De- scent 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 . . . . .	194
6.5.2	Why do Conventional Kernels Promote Linear Separability? . . .	195
6.5.3	Strengths and Weaknesses of Different Kernels . . . . .	197

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
<b>7</b>	<b>Classifier Performance and Evaluation</b>	<b>207</b>
7.1	Introduction . . . . .	207
7.2	The Bias-Variance Trade-Off . . . . .	208
7.2.1	A Formal View . . . . .	209
7.2.2	Telltale Signs of Bias and Variance . . . . .	213
7.3	Implications of Bias-Variance Trade-Off on Performance . . . . .	213
7.3.1	Impact of Training Data Size . . . . .	213
7.3.2	Impact of Data Dimensionality . . . . .	215
7.3.3	Implications for Model Choice in Text . . . . .	215
7.4	Systematic Performance Enhancement with Ensembles . . . . .	216
7.4.1	Bagging and Subsampling . . . . .	216
7.4.2	Boosting . . . . .	218
7.5	Classifier Evaluation . . . . .	220
7.5.1	Segmenting into Training and Testing Portions . . . . .	221
7.5.1.1	Hold-Out . . . . .	221
7.5.1.2	Cross-Validation . . . . .	222
7.5.2	Absolute Accuracy Measures . . . . .	222
7.5.2.1	Accuracy of Classification . . . . .	222
7.5.2.2	Accuracy of Regression . . . . .	223
7.5.3	Ranking Measures for Classification and Information Retrieval . . . . .	224
7.5.3.1	Receiver Operating Characteristic . . . . .	225
7.5.3.2	Top-Heavy Measures for Ranked Lists . . . . .	229
7.6	Summary . . . . .	230
7.7	Bibliographic Notes . . . . .	230
7.7.1	Software Resources . . . . .	231
7.7.2	Data Sets for Evaluation . . . . .	231
7.8	Exercises . . . . .	232
<b>8</b>	<b>Joint Text Mining with Heterogeneous Data</b>	<b>233</b>
8.1	Introduction . . . . .	233
8.2	The Shared Matrix Factorization Trick . . . . .	235
8.2.1	The Factorization Graph . . . . .	235
8.2.2	Application: Shared Factorization with Text and Web Links . . . . .	236
8.2.2.1	Solving the Optimization Problem . . . . .	238
8.2.2.2	Supervised Embeddings . . . . .	239
8.2.3	Application: Text with Undirected Social Networks . . . . .	240
8.2.3.1	Application to Link Prediction with Text Content . . . . .	241
8.2.4	Application: Transfer Learning in Images with Text . . . . .	241
8.2.4.1	Transfer Learning with Unlabeled Text . . . . .	242
8.2.4.2	Transfer Learning with Labeled Text . . . . .	243
8.2.5	Application: Recommender Systems with Ratings and Text . . . . .	244
8.2.6	Application: Cross-Lingual Text Mining . . . . .	246

8.3	Factorization Machines . . . . .	247
8.4	Joint Probabilistic Modeling Techniques . . . . .	250
8.4.1	Joint Probabilistic Models for Clustering . . . . .	251
8.4.2	Naïve Bayes Classifier . . . . .	252
8.5	Transformation to Graph Mining Techniques . . . . .	252
8.6	Summary . . . . .	255
8.7	Bibliographic Notes . . . . .	255
8.7.1	Software Resources . . . . .	256
8.8	Exercises . . . . .	256
<b>9</b>	<b>Information Retrieval and Search Engines</b>	<b>257</b>
9.1	Introduction . . . . .	257
9.2	Indexing and Query Processing . . . . .	258
9.2.1	Dictionary Data Structures . . . . .	259
9.2.2	Inverted Index . . . . .	261
9.2.3	Linear Time Index Construction . . . . .	262
9.2.4	Query Processing . . . . .	264
9.2.4.1	Boolean Retrieval . . . . .	264
9.2.4.2	Ranked Retrieval . . . . .	265
9.2.4.3	Positional Queries . . . . .	269
9.2.4.4	Zoned Scoring . . . . .	270
9.2.4.5	Machine Learning in Information Retrieval . . . . .	271
9.2.4.6	Ranking Support Vector Machines . . . . .	272
9.2.5	Efficiency Optimizations . . . . .	274
9.2.5.1	Skip Pointers . . . . .	274
9.2.5.2	Champion Lists and Tiered Indexes . . . . .	275
9.2.5.3	Caching Tricks . . . . .	275
9.2.5.4	Compression Tricks . . . . .	276
9.3	Scoring with Information Retrieval Models . . . . .	278
9.3.1	Vector Space Models with tf-idf . . . . .	278
9.3.2	The Binary Independence Model . . . . .	279
9.3.3	The BM25 Model with Term Frequencies . . . . .	281
9.3.4	Statistical Language Models in Information Retrieval . . . . .	283
9.3.4.1	Query Likelihood Models . . . . .	283
9.4	Web Crawling and Resource Discovery . . . . .	285
9.4.1	A Basic Crawler Algorithm . . . . .	285
9.4.2	Preferential Crawlers . . . . .	287
9.4.3	Multiple Threads . . . . .	288
9.4.4	Combatting Spider Traps . . . . .	288
9.4.5	Shingling for Near Duplicate Detection . . . . .	289
9.5	Query Processing in Search Engines . . . . .	289
9.5.1	Distributed Index Construction . . . . .	290
9.5.2	Dynamic Index Updates . . . . .	291
9.5.3	Query Processing . . . . .	291
9.5.4	The Importance of Reputation . . . . .	292

9.6	Link-Based Ranking Algorithms . . . . .	293
9.6.1	PageRank . . . . .	293
9.6.1.1	Topic-Sensitive PageRank . . . . .	296
9.6.1.2	SimRank . . . . .	297
9.6.2	HITS . . . . .	298
9.7	Summary . . . . .	300
9.8	Bibliographic Notes . . . . .	300
9.8.1	Software Resources . . . . .	301
9.9	Exercises . . . . .	302
<b>10</b>	<b>Language Modeling and Deep Learning</b>	<b>303</b>
10.1	Introduction . . . . .	303
10.2	Statistical Language Models . . . . .	306
10.2.1	Skip-Gram Models . . . . .	308
10.2.2	Relationship with Embeddings . . . . .	310
10.2.3	Evaluating Language Models with Perplexity . . . . .	311
10.3	Kernel Methods for Sequence-Centric Learning . . . . .	312
10.4	Word-Context Matrix Factorization Models . . . . .	313
10.4.1	Matrix Factorization with Counts . . . . .	313
10.4.2	The GloVe Embedding . . . . .	315
10.4.3	PPMI Matrix Factorization . . . . .	316
10.4.4	Shifted PPMI Matrix Factorization . . . . .	317
10.4.5	Incorporating Syntactic and Other Features . . . . .	317
10.5	Graphical Representations of Word Distances . . . . .	317
10.6	Neural Networks and Word Embeddings . . . . .	319
10.6.1	Neural Networks: A Gentle Introduction . . . . .	319
10.6.1.1	Single Computational Layer: The Perceptron . . . . .	320
10.6.1.2	Multilayer Neural Networks . . . . .	325
10.6.2	Neural Embedding with Word2vec . . . . .	329
10.6.2.1	Neural Embedding with Continuous Bag of Words . . . . .	330
10.6.2.2	Neural Embedding with Skip-Gram Model . . . . .	332
10.6.2.3	Skip-Gram with Negative Sampling . . . . .	335
10.6.2.4	What Is the Actual Neural Architecture of SGNS? . . . . .	336
10.6.3	Word2vec (SGNS) Is Logistic Matrix Factorization . . . . .	337
10.6.4	Beyond Words: Embedding Paragraphs with Doc2vec . . . . .	339
10.7	Recurrent Neural Networks . . . . .	341
10.7.1	Language Modeling Example of RNN . . . . .	343
10.7.1.1	Generating a Language Sample . . . . .	344
10.7.2	Backpropagation Through Time . . . . .	345
10.7.3	Bidirectional Recurrent Networks . . . . .	348
10.7.4	Multilayer Recurrent Networks . . . . .	350
10.7.5	Long Short-Term Memory (LSTM) . . . . .	350
10.7.6	Gated Recurrent Units (GRUs) . . . . .	353
10.7.7	Layer Normalization . . . . .	355
10.8	Applications of Recurrent Neural Networks . . . . .	356
10.8.1	Contextual Word Embeddings with ELMo . . . . .	356
10.8.2	Application to Automatic Image Captioning . . . . .	357
10.8.3	Sequence-to-Sequence Learning and Machine Translation . . . . .	358
10.8.3.1	BLEU Score for Evaluating Machine Translation . . . . .	361

10.8.4	Application to Sentence-Level Classification . . . . .	362
10.8.5	Token-Level Classification with Linguistic Features . . . . .	362
10.9	Convolutional Neural Networks for Text . . . . .	364
10.10	Summary . . . . .	365
10.11	Bibliographic Notes . . . . .	366
10.11.1	Software Resources . . . . .	367
10.12	Exercises . . . . .	367
<b>11</b>	<b>Attention Mechanisms and Transformers</b>	<b>369</b>
11.1	Introduction . . . . .	369
11.2	Attention Mechanisms for Machine Translation . . . . .	371
11.2.1	The Luong Attention Model . . . . .	371
11.2.2	Variations and Comparison with Bahdanau Attention . . . . .	373
11.3	Transformer Networks . . . . .	375
11.3.1	How Self Attention Helps . . . . .	375
11.3.2	The Self-Attention Module . . . . .	376
11.3.3	Incorporating Positional Information . . . . .	378
11.3.4	The Sequence-to-Sequence Transformer . . . . .	379
11.3.5	Multihead Attention . . . . .	380
11.4	Transformer-Based Pre-trained Language Models . . . . .	380
11.4.1	GPT-n . . . . .	381
11.4.2	BERT . . . . .	382
11.4.3	T5 . . . . .	384
11.5	Natural Language Processing Applications . . . . .	385
11.5.1	The GLUE and SuperGLUE Benchmarks . . . . .	386
11.5.2	The Corpus of Linguistic Acceptability (CoLA) . . . . .	386
11.5.3	Sentiment Analysis . . . . .	387
11.5.4	Token-Level Classification . . . . .	387
11.5.5	Machine Translation and Summarization . . . . .	388
11.5.6	Textual Entailment . . . . .	389
11.5.7	Semantic Textual Similarity . . . . .	389
11.5.8	Word Sense Disambiguation . . . . .	389
11.5.9	Co-Reference Resolution . . . . .	390
11.5.10	Question Answering . . . . .	390
11.6	Summary . . . . .	390
11.7	Bibliographic Notes . . . . .	391
11.7.1	Software Resources . . . . .	391
11.8	Exercises . . . . .	391
<b>12</b>	<b>Text Summarization</b>	<b>393</b>
12.1	Introduction . . . . .	393
12.1.1	Extractive and Abstractive Summarization . . . . .	394
12.1.2	Key Steps in Extractive Summarization . . . . .	395
12.1.3	The Segmentation Phase in Extractive Summarization . . . . .	395
12.2	Topic Word Methods for Extractive Summarization . . . . .	396
12.2.1	Word Probabilities . . . . .	396
12.2.2	Normalized Frequency Weights . . . . .	397
12.2.3	Topic Signatures . . . . .	398
12.2.4	Sentence Selection Methods . . . . .	400



12.3	Latent Methods for Extractive Summarization . . . . .	401
12.3.1	Latent Semantic Analysis . . . . .	401
12.3.2	Lexical Chains . . . . .	402
12.3.2.1	Short Description of WordNet . . . . .	402
12.3.2.2	Leveraging WordNet for Lexical Chains . . . . .	403
12.3.3	Graph-Based Methods . . . . .	404
12.3.4	Centroid Summarization . . . . .	405
12.4	Traditional Machine Learning for Extractive Summarization . . . . .	406
12.4.1	Feature Extraction . . . . .	406
12.4.2	Which Classifiers to Use? . . . . .	407
12.5	Deep Learning for Extractive Summarization . . . . .	407
12.5.1	Recurrent Neural Networks . . . . .	407
12.5.2	Using Pre-Trained Language Models with Transformers . . . . .	409
12.6	Multi-Document Summarization . . . . .	410
12.6.1	Centroid-Based Summarization . . . . .	410
12.6.2	Graph-Based Methods . . . . .	412
12.7	Abstractive Summarization . . . . .	412
12.7.1	Sentence Compression . . . . .	413
12.7.2	Information Fusion . . . . .	413
12.7.3	Information Ordering . . . . .	414
12.7.4	Recurrent Neural Networks for Summarization . . . . .	414
12.7.5	Abstractive Summarization with Transformers . . . . .	415
12.8	Summary . . . . .	416
12.9	Bibliographic Notes . . . . .	417
12.9.1	Software Resources . . . . .	417
12.10	Exercises . . . . .	418
<b>13</b>	<b>Information Extraction and Knowledge Graphs</b>	<b>419</b>
13.1	Introduction . . . . .	419
13.1.1	Historical Evolution . . . . .	421
13.1.2	The Role of Natural Language Processing . . . . .	422
13.2	Named Entity Recognition . . . . .	424
13.2.1	Rule-Based Methods . . . . .	425
13.2.1.1	Training Algorithms for Rule-Based Systems . . . . .	427
13.2.2	Transformation to Token-Level Classification . . . . .	429
13.2.3	Hidden Markov Models . . . . .	430
13.2.3.1	Training . . . . .	432
13.2.3.2	Prediction for Test Segment . . . . .	433
13.2.3.3	Incorporating Extracted Features . . . . .	433
13.2.3.4	Variations and Enhancements . . . . .	433
13.2.4	Maximum Entropy Markov Models . . . . .	434
13.2.5	Conditional Random Fields . . . . .	436
13.2.6	Deep Learning for Entity Extraction . . . . .	437
13.2.6.1	Recurrent Neural Networks for Named Entity Recognition . . . . .	437
13.2.6.2	Use of Pretrained Language Models with Transformers . . . . .	439
13.3	Relationship Extraction . . . . .	439
13.3.1	Transformation to Classification . . . . .	440

13.3.2	Relationship Prediction with Explicit Feature Engineering . . . .	441
13.3.3	Relationship Prediction with Implicit Feature Engineering: Kernel Methods . . . . .	444
13.3.3.1	Kernels from Dependency Graphs . . . . .	445
13.3.3.2	Subsequence-Based Kernels . . . . .	446
13.3.3.3	Convolution Tree-Based Kernels . . . . .	447
13.3.4	Relationship Extraction with Pretrained Language Models . . .	449
13.4	Knowledge Graphs . . . . .	450
13.4.1	Constructing a Knowledge Graph . . . . .	456
13.4.2	Knowledge Graphs in Search . . . . .	458
13.5	Summary . . . . .	460
13.6	Bibliographic Notes . . . . .	461
13.6.1	Weakly Supervised Learning Methods . . . . .	461
13.6.2	Unsupervised and Open Information Extraction . . . . .	462
13.6.3	Software Resources . . . . .	462
13.7	Exercises . . . . .	463
<b>14</b>	<b>Question Answering</b>	<b>465</b>
14.1	Introduction . . . . .	465
14.2	The Reading Comprehension Task . . . . .	469
14.2.1	Using Recurrent Neural Networks with Attention . . . . .	471
14.2.2	Leveraging Pretrained Language Models . . . . .	474
14.3	Retrieval for Open-Domain Question Answering . . . . .	476
14.3.1	Dense Retrieval in Open Retriever Question Answering . . . . .	477
14.3.2	Salient Span Masking . . . . .	480
14.4	Closed Book Systems with Pretrained Language Models . . . . .	480
14.5	Question Answering with Knowledge Graphs . . . . .	482
14.5.1	Leveraging Query Translation . . . . .	483
14.5.2	Fusing Text and Structured Data . . . . .	483
14.5.3	Knowledge Graph to Corpus Translation . . . . .	485
14.6	Challenges of Long-Form Question Answering . . . . .	486
14.7	Summary . . . . .	487
14.8	Bibliographic Notes . . . . .	488
14.8.1	Data Sets for Evaluation . . . . .	489
14.8.2	Software Resources . . . . .	489
14.9	Exercises . . . . .	489
<b>15</b>	<b>Opinion Mining and Sentiment Analysis</b>	<b>491</b>
15.1	Introduction . . . . .	491
15.1.1	The Opinion Lexicon . . . . .	493
15.2	Document-Level Sentiment Classification . . . . .	496
15.2.1	Unsupervised Approaches to Classification . . . . .	498
15.3	Phrase- and Sentence-Level Sentiment Classification . . . . .	499
15.3.1	Applications of Sentence- and Phrase-Level Analysis . . . . .	500
15.3.2	Reduction of Subjectivity Classification to Minimum Cut Problem . . . . .	501
15.3.3	Context in Sentence- and Phrase-Level Polarity Analysis . . . . .	501
15.3.4	Sentiment Analysis with Deep Learning . . . . .	502
15.3.4.1	Recurrent Neural Networks . . . . .	502

15.3.4.2	Leveraging Pretrained Language Models with Transformers . . . . .	503
15.4	Aspect-Based Opinion Mining as Information Extraction . . . . .	503
15.4.1	Hu and Liu's Unsupervised Approach . . . . .	504
15.4.2	OPINE: An Unsupervised Approach . . . . .	505
15.4.3	Supervised Opinion Extraction as Token-Level Classification . . . . .	506
15.5	Opinion Spam . . . . .	508
15.5.1	Supervised Methods for Spam Detection . . . . .	508
15.5.1.1	Labeling Deceptive Spam . . . . .	509
15.5.1.2	Feature Extraction . . . . .	509
15.5.2	Unsupervised Methods for Spammer Detection . . . . .	510
15.6	Opinion Summarization . . . . .	511
15.7	Summary . . . . .	512
15.8	Bibliographic Notes . . . . .	513
15.8.1	Software Resources . . . . .	514
15.9	Exercises . . . . .	514
<b>16</b>	<b>Text Segmentation and Event Detection</b>	<b>515</b>
16.1	Introduction . . . . .	515
16.1.1	Relationship with Topic Detection and Tracking . . . . .	516
16.2	Text Segmentation . . . . .	516
16.2.1	TextTiling . . . . .	517
16.2.2	The C99 Approach . . . . .	518
16.2.3	Supervised Segmentation with Off-the-Shelf Classifiers . . . . .	519
16.2.4	Supervised Segmentation with Markovian Models . . . . .	521
16.3	Mining Text Streams . . . . .	523
16.3.1	Streaming Text Clustering . . . . .	523
16.3.2	Application to First Story Detection . . . . .	524
16.4	Event Detection . . . . .	525
16.4.1	Unsupervised Event Detection . . . . .	525
16.4.1.1	Window-Based Nearest-Neighbor Method . . . . .	525
16.4.1.2	Leveraging Generative Models . . . . .	526
16.4.1.3	Event Detection in Social Streams . . . . .	527
16.4.2	Supervised Event Detection as Supervised Segmentation . . . . .	527
16.4.3	Event Detection as an Information Extraction Problem . . . . .	528
16.4.3.1	Transformation to Token-Level Classification . . . . .	528
16.4.3.2	Open Domain Event Extraction . . . . .	529
16.5	Summary . . . . .	531
16.6	Bibliographic Notes . . . . .	531
16.6.1	Software Resources . . . . .	531
16.7	Exercises . . . . .	532
	<b>Correction to: Machine Learning for Text</b>	<b>C1</b>
	<b>Index . . . . .</b>	<b>561</b>

---

# 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 400 papers in refereed conferences and journals and authored over 80 patents. He is the author or editor of 20 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 a recipient of the IEEE ICDM Research Contributions Award (2015) and ACM SIGKDD Innovation Award, which are the two most prestigious awards for influential research contributions in the field of data mining. He is also a recipient of the W. Wallace McDowell Award, which is the highest award given solely by the IEEE Computer Society across the field of Computer Science.

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 has served or currently serves as the editor-in-chief of the ACM Transactions on Knowledge Discovery from Data as well as the ACM SIGKDD Explorations. He is also an editor-in-chief of ACM Books. 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.”