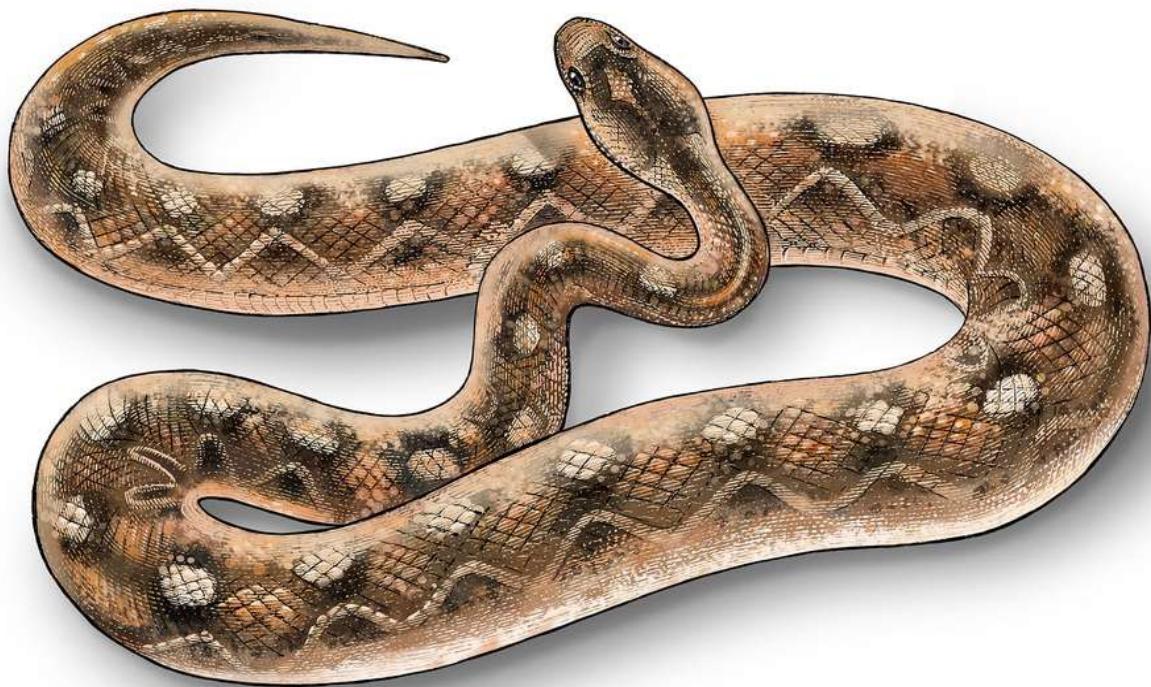


O'REILLY®

Blueprints for Text Analytics Using Python

Machine Learning-Based Solutions for
Common Real World (NLP) Applications



Jens Albrecht,
Sidharth Ramachandran
& Christian Winkler

1. Preface

1. Approach of the Book
2. Prerequisites
3. Some Important Libraries to Know
4. Books to Read
5. Conventions Used in This Book
6. Using Code Examples
7. O'Reilly Online Learning
8. How to Contact Us
9. Acknowledgments

2. 1. Gaining Early Insights from Textual Data

1. What You'll Learn and What We'll Build
2. Exploratory Data Analysis
3. Introducing the Dataset
4. Blueprint: Getting an Overview of the Data with Pandas

1. Calculating Summary Statistics for Columns
2. Checking for Missing Data

3. Plotting Value Distributions
4. Comparing Value Distributions Across Categories
5. Visualizing Developments Over Time
5. Blueprint: Building a Simple Text Preprocessing Pipeline
 1. Performing Tokenization with Regular Expressions
 2. Treating Stop Words
 3. Processing a Pipeline with One Line of Code
6. Blueprints for Word Frequency Analysis
 1. Blueprint: Counting Words with a Counter
 2. Blueprint: Creating a Frequency Diagram
 3. Blueprint: Creating Word Clouds
 4. Blueprint: Ranking with TF-IDF
7. Blueprint: Finding a Keyword-in-Context
8. Blueprint: Analyzing N-Grams
9. Blueprint: Comparing Frequencies Across Time Intervals and Categories

1. Creating Frequency Timelines
 2. Creating Frequency Heatmaps
10. Closing Remarks
3. 2. Extracting Textual Insights with APIs
 1. What You'll Learn and What We'll Build
 2. Application Programming Interfaces
 3. Blueprint: Extracting Data from an API Using the Requests Module
 1. Pagination
 2. Rate Limiting
 4. Blueprint: Extracting Twitter Data with Tweepy
 1. Obtaining Credentials
 2. Installing and Configuring Tweepy
 3. Extracting Data from the Search API
 4. Extracting Data from a User's Timeline

5. Extracting Data from the Streaming API

5. Closing Remarks

4. 3. Scraping Websites and Extracting Data

1. What You'll Learn and What We'll Build
2. Scraping and Data Extraction
3. Introducing the Reuters News Archive
4. URL Generation
5. Blueprint: Downloading and Interpreting robots.txt
6. Blueprint: Finding URLs from sitemap.xml
7. Blueprint: Finding URLs from RSS
8. Downloading Data
9. Blueprint: Downloading HTML Pages with Python
10. Blueprint: Downloading HTML Pages with wget
11. Extracting Semistructured Data
12. Blueprint: Extracting Data with Regular Expressions
13. Blueprint: Using an HTML Parser for Extraction
14. Blueprint: Spidering

1. Introducing the Use Case

2. Error Handling and Production-Quality Software

15. Density-Based Text Extraction

1. Extracting Reuters Content with Readability

2. Summary Density-Based Text Extraction

16. All-in-One Approach

17. Blueprint: Scraping the Reuters Archive with Scrapy

18. Possible Problems with Scraping

19. Closing Remarks and Recommendation

5. 4. Preparing Textual Data for Statistics and Machine Learning

1. What You'll Learn and What We'll Build

2. A Data Preprocessing Pipeline

3. Introducing the Dataset: Reddit Self-Posts

1. Loading Data Into Pandas

2. Blueprint: Standardizing Attribute Names
3. Saving and Loading a DataFrame
4. Cleaning Text Data

1. Blueprint: Identify Noise with Regular Expressions
2. Blueprint: Removing Noise with Regular Expressions
3. Blueprint: Character Normalization with textacy
4. Blueprint: Pattern-Based Data Masking with textacy

5. Tokenization

1. Blueprint: Tokenization with Regular Expressions
2. Tokenization with NLTK
3. Recommendations for Tokenization

6. Linguistic Processing with spaCy

1. Instantiating a Pipeline
2. Processing Text

3. Blueprint: Customizing Tokenization
4. Blueprint: Working with Stop Words
5. Blueprint: Extracting Lemmas Based on Part of Speech
6. Blueprint: Extracting Noun Phrases
7. Blueprint: Extracting Named Entities

7. Feature Extraction on a Large Dataset

1. Blueprint: Creating One Function to Get It All
2. Blueprint: Using spaCy on a Large Dataset
3. Persisting the Result
4. A Note on Execution Time

8. There Is More

1. Language Detection
2. Spell-Checking
3. Token Normalization

9. Closing Remarks and Recommendations

6. 5. Feature Engineering and Syntactic Similarity

1. What You'll Learn and What We'll Build
2. A Toy Dataset for Experimentation
3. Blueprint: Building Your Own Vectorizer

1. Enumerating the Vocabulary
 2. Vectorizing Documents
 3. The Document-Term Matrix
 4. The Similarity Matrix
4. Bag-of-Words Models

1. Blueprint: Using scikit-learn's CountVectorizer
 2. Blueprint: Calculating Similarities
5. TF-IDF Models

1. Optimized Document Vectors with TfidfTransformer
2. Introducing the ABC Dataset
3. Blueprint: Reducing Feature Dimensions
4. Blueprint: Improving Features by Making Them More Specific

5. Blueprint: Using Lemmas Instead of Words for Vectorizing Documents
6. Blueprint: Limit Word Types
7. Blueprint: Remove Most Common Words
8. Blueprint: Adding Context via N-Grams

6. Syntactic Similarity in the ABC Dataset

1. Blueprint: Finding Most Similar Headlines to a Made-up Headline
2. Blueprint: Finding the Two Most Similar Documents in a Large Corpus (Much More Difficult)
3. Blueprint: Finding Related Words
4. Tips for Long-Running Programs like Syntactic Similarity

7. Summary and Conclusion

7. 6. Text Classification Algorithms

1. What You'll Learn and What We'll Build
2. Introducing the Java Development Tools Bug Dataset
3. Blueprint: Building a Text Classification System

1. Step 1: Data Preparation
 2. Step 2: Train-Test Split
 3. Step 3: Training the Machine Learning Model
 4. Step 4: Model Evaluation
 4. Final Blueprint for Text Classification
 5. Blueprint: Using Cross-Validation to Estimate Realistic Accuracy Metrics
 6. Blueprint: Performing Hyperparameter Tuning with Grid Search
 7. Blueprint Recap and Conclusion
 8. Closing Remarks
 9. Further Reading
8. 7. How to Explain a Text Classifier
1. What You'll Learn and What We'll Build
 2. Blueprint: Determining Classification Confidence Using Prediction Probability
 3. Blueprint: Measuring Feature Importance of Predictive Models
 4. Blueprint: Using LIME to Explain the Classification Results
 5. Blueprint: Using ELI5 to Explain the Classification Results

6. Blueprint: Using Anchor to Explain the Classification Results

1. Using the Distribution with Masked Words

2. Working with Real Words

7. Closing Remarks

9. 8. Unsupervised Methods: Topic Modeling and Clustering

1. What You'll Learn and What We'll Build

2. Our Dataset: UN General Debates

1. Checking Statistics of the Corpus

2. Preparations

3. Nonnegative Matrix Factorization (NMF)

1. Blueprint: Creating a Topic Model Using NMF for Documents

2. Blueprint: Creating a Topic Model for Paragraphs Using NMF

4. Latent Semantic Analysis/Indexing

1. Blueprint: Creating a Topic Model for Paragraphs with SVD

5. Latent Dirichlet Allocation

1. Blueprint: Creating a Topic Model for Paragraphs with LDA
2. Blueprint: Visualizing LDA Results
6. Blueprint: Using Word Clouds to Display and Compare Topic Models
7. Blueprint: Calculating Topic Distribution of Documents and Time Evolution
8. Using Gensim for Topic Modeling
 1. Blueprint: Preparing Data for Gensim
 2. Blueprint: Performing Nonnegative Matrix Factorization with Gensim
 3. Blueprint: Using LDA with Gensim
 4. Blueprint: Calculating Coherence Scores
 5. Blueprint: Finding the Optimal Number of Topics
 6. Blueprint: Creating a Hierarchical Dirichlet Process with Gensim
9. Blueprint: Using Clustering to Uncover the Structure of Text Data

10. Further Ideas

11. Summary and Recommendation

12. Conclusion

10. 9. Text Summarization

1. What You'll Learn and What We'll Build

2. Text Summarization

1. Extractive Methods

2. Data Preprocessing

3. Blueprint: Summarizing Text Using Topic Representation

1. Identifying Important Words with TF-IDF Values

2. LSA Algorithm

4. Blueprint: Summarizing Text Using an Indicator Representation

5. Measuring the Performance of Text Summarization Methods

6. Blueprint: Summarizing Text Using Machine Learning

1. Step 1: Creating Target Labels

2. Step 2: Adding Features to Assist Model Prediction

3. Step 3: Build a Machine Learning Model

7. Closing Remarks

8. Further Reading

11. 10. Exploring Semantic Relationships with Word Embeddings

1. What You'll Learn and What We'll Build

2. The Case for Semantic Embeddings

1. Word Embeddings

2. Analogy Reasoning with Word Embeddings

3. Types of Embeddings

3. Blueprint: Using Similarity Queries on Pretrained Models

1. Loading a Pretrained Model

2. Similarity Queries

4. Blueprints for Training and Evaluating Your Own Embeddings

1. Data Preparation

2. Blueprint: Training Models with Gensim

3. Blueprint: Evaluating Different Models

5. Blueprints for Visualizing Embeddings

1. Blueprint: Applying Dimensionality Reduction

2. Blueprint: Using the TensorFlow Embedding Projector

3. Blueprint: Constructing a Similarity Tree

6. Closing Remarks

7. Further Reading

12. 11. Performing Sentiment Analysis on Text Data

1. What You'll Learn and What We'll Build

2. Sentiment Analysis

3. Introducing the Amazon Customer Reviews Dataset

4. Blueprint: Performing Sentiment Analysis Using Lexicon-Based Approaches

1. Bing Liu Lexicon

2. Disadvantages of a Lexicon-Based Approach

5. Supervised Learning Approaches

1. Preparing Data for a Supervised Learning Approach

6. Blueprint: Vectorizing Text Data and Applying a Supervised Machine Learning Algorithm

1. Step 1: Data Preparation

2. Step 2: Train-Test Split

3. Step 3: Text Vectorization

4. Step 4: Training the Machine Learning Model

7. Pretrained Language Models Using Deep Learning

1. Deep Learning and Transfer Learning

8. Blueprint: Using the Transfer Learning Technique and a Pretrained Language Model

1. Step 1: Loading Models and Tokenization

- 2. Step 2: Model Training
 - 3. Step 3: Model Evaluation
- 9. Closing Remarks
- 10. Further Reading
- 13. 12. Building a Knowledge Graph
 - 1. What You'll Learn and What We'll Build
 - 2. Knowledge Graphs
 - 1. Information Extraction
 - 3. Introducing the Dataset
 - 4. Named-Entity Recognition
 - 1. Blueprint: Using Rule-Based Named-Entity Recognition
 - 2. Blueprint: Normalizing Named Entities
 - 3. Merging Entity Tokens
 - 5. Coreference Resolution
 - 1. Blueprint: Using spaCy's Token Extensions
 - 2. Blueprint: Performing Alias Resolution

3. Blueprint: Resolving Name Variations
4. Blueprint: Performing Anaphora Resolution with NeuralCoref
5. Name Normalization
6. Entity Linking
6. Blueprint: Creating a Co-Occurrence Graph
 1. Extracting Co-Occurrences from a Document
 2. Visualizing the Graph with Gephi
7. Relation Extraction
 1. Blueprint: Extracting Relations Using Phrase Matching
 2. Blueprint: Extracting Relations Using Dependency Trees
8. Creating the Knowledge Graph
 1. Don't Blindly Trust the Results
9. Closing Remarks
10. Further Reading
14. 13. Using Text Analytics in Production

1. What You'll Learn and What We'll Build
2. Blueprint: Using Conda to Create Reproducible Python Environments
3. Blueprint: Using Containers to Create Reproducible Environments
4. Blueprint: Creating a REST API for Your Text Analytics Model
5. Blueprint: Deploying and Scaling Your API Using a Cloud Provider
6. Blueprint: Automatically Versioning and Deploying Builds
7. Closing Remarks
8. Further Reading

15. Index