

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/375744928>

Natural Language Processing : A Textbook with Python Implementation

Book · November 2023

DOI: 10.1007/978-981-99-1999-4

CITATIONS

0

READS

485

1 author:



[Raymond Lee](#)

United International College

119 PUBLICATIONS **1,006** CITATIONS

[SEE PROFILE](#)

Raymond S. T. Lee

Natural Language Processing

A Textbook with Python
Implementations

 Springer

Natural Language Processing

Raymond S. T. Lee

Natural Language Processing

A Textbook with Python Implementation



Springer

Raymond S. T. Lee
United International College
Beijing Normal University-Hong Kong Baptist University
Zhuhai, China

ISBN 978-981-99-1998-7 ISBN 978-981-99-1999-4 (eBook)
<https://doi.org/10.1007/978-981-99-1999-4>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2024

This work is subject to copyright. All rights are solely and exclusively licensed 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 Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

This book is dedicated to all readers and students taking my undergraduate and postgraduate courses in Natural Language Processing, your enthusiasm in seeking knowledge incited me to write this book.

Preface

Motivation of This Book

Natural Language Processing (NLP) and its related applications become part of daily life with exponential growth of Artificial Intelligence (AI) in past decades. NLP applications including Information Retrieval (IR) systems, Text Summarization System, and Question-and-Answering (Chatbot) System became one of the prevalent topics in both industry and academia that had evolved routines and benefited immensely to a wide array of day-to-day services.

The objective of this book is to provide NLP concepts and knowledge to readers with a 14-h 7 step-by-step workshops to practice various core Python-based NLP tools: NLTK, spaCy, TensorFlow Keras, Transformer, and BERT Technology to construct NLP applications.

Organization and Structure of This Book

This book consists of two parts:

Part I Concepts and Technology (Chaps. 1–9)

Discuss concepts and technology related to NLP including: Introduction, N-gram Language Model, Part-of-Speech Tagging, Syntax and Parsing, Meaning Representation, Semantic Analysis, Pragmatic Analysis, Transfer Learning and Transformer Technology, Major NLP Applications.

Part II Natural Language Processing Workshops with Python Implementation (Chaps. 10–16)

7 Python workshops to provide step-by-step Python implementation tools including: NLTK, spaCy, TensorFlow Keras, Transformer, and BERT Technology.

This book is organized and structured as follows:

Part I: Concepts and Technology

- Chapter 1: Introduction to Natural Language Processing

This introductory chapter begins with human language and intelligence constituting six levels of linguistics followed by a brief history of NLP with major components and applications. It serves as the cornerstone to the NLP concepts and technology discussed in the following chapters. This chapter also serves as the conceptual basis for Workshop#1: Basics of Natural Language Toolkit (NLTK) in Chap. 10.

- Chapter 2: N-gram Language Model

Language model is the foundation of NLP. This chapter introduces N-gram language model and Markov Chains using classical literature *The Adventures of Sherlock Holmes* by Sir Conan Doyle (1859–1930) to illustrate how N-gram model works that form NLP basics in text analysis followed by Shannon’s model and text generation with evaluation schemes. This chapter also serves as the conceptual basis for Workshop#2 on N-gram modelling with NLTK in Chap. 11.

- Chapter 3: Part-of-Speech Tagging

Part-of-Speech (POS) Tagging is the foundation of text processing in NLP. This chapter describes how it relates to NLP and Natural Language Understanding (NLU). There are types and algorithms for POS Tagging including Rule-based POS Tagging, Stochastic POS Tagging, and Hybrid POS Tagging with Brill Tagger and evaluation schemes. This chapter also serves as the conceptual basis for Workshop#3: Part-of-Speech using Natural Language Toolkit in Chap. 12.

- Chapter 4—Syntax and Parsing

As another major component of Natural Language Understanding (NLU), this chapter explores syntax analysis and introduces different types of constituents in English language followed by the main concept of context-free grammar (CFG) and CFG parsing. It also studies different major parsing techniques, including lexical and probabilistic parsing with live examples for illustration.

- Chapter 5: Meaning Representation

Before the study of Semantic Analysis, this chapter explores meaning representation, a vital component in NLP. It studies four major meaning representation techniques which include: first-order predicate calculus (FOPC), semantic net, conceptual dependency diagram (CDD), and frame-based representation. After that it explores canonical form and introduces Fillmore’s theory of universal cases followed by predicate logic and inference work using FOPC with live examples.

- Chapter 6: Semantic Analysis

This chapter studies Semantic Analysis, one of the core concepts for learning NLP. First, it studies the two basic schemes of semantic analysis: lexical and compositional semantic analysis. After that it explores word senses and six commonly used lexical semantics followed by word sense disambiguation (WSD) and various WSD schemes. Further, it also studies WordNet and online thesauri for word similarity and various distributed similarity measurement including Point-wise Mutual Information (PMI) and Positive Point-wise Mutual information (PPMI) models with live examples for illustration. Chapters 4 and 5 also

serve as the conceptual basis for Workshop#4: Semantic Analysis and Word Vectors using spaCy in Chap. 13.

- Chapter 7: Pragmatic Analysis

After the discussion of semantic meaning and analysis, this chapter explores pragmatic analysis in linguistics and discourse phenomena. It also studies coherence and coreference as the key components of pragmatics and discourse critical to NLP, followed by discourse segmentation with different algorithms on Co-reference Resolution including Hobbs Algorithm, Centering Algorithm, Log-Linear Model, the latest machine learning methods, and evaluation schemes. This chapter also serves as the conceptual basis for Workshop#5: Sentiment Analysis and Text Classification in Chap. 14.

- Chapter 8: Transfer Learning and Transformer Technology

Transfer learning is a commonly used deep learning model to minimize computational resources. This chapter explores: (1) Transfer Learning (TL) against traditional Machine Learning (ML); (2) Recurrent Neural Networks (RNN), a significant component of transfer learning with core technologies such as Long Short-Term Memory (LSTM) Network and Bidirectional Recurrent Neural Networks (BRNNs) in NLP applications, and (3) Transformer technology architecture, Bidirectional Encoder Representation from Transformers (BERT) Model, and related technologies including Transformer-XL and ALBERT technologies. This chapter also serves as the conceptual basis for Workshop#6: Transformers with spaCy and Tensorflow in Chap. 15.

- Chapter 9: Major Natural Language Processing Applications

This is a summary of Part I with three core NLP applications: Information Retrieval (IR) systems, Text Summarization (TS) systems, and Question-and-Answering (Q&A) chatbot systems, how they work and related R&D in building NLP applications. This chapter also serves as the conceptual basis for Workshop#7: Building Chatbot with TensorFlow and Transformer Technology in Chap. 16.

Part II: Natural Language Processing Workshops with Python Implementation in 14 h

- Chapter 10: Workshop#1 Basics of Natural Language Toolkit (Hour 1–2)

With the basic NLP concept being learnt in Chap. 1, this introductory workshop gives a NLTK overview and system installation procedures are the foundations of Python NLP development tool used for text processing which include simple text analysis, text analysis with lexical dispersion plot, text tokenization, and basic statistical tools in NLP.

- Chapter 11: Workshop#2 N-grams Modelling with Natural Language Toolkit (Hour 3–4)

This is a coherent workshop of Chap. 2 using NLTK technology for N-gram generation and statistics. This workshop consists of two parts. Part I introduces N-gram language model using NLTK in Python and N-grams class to generate N-gram statistics on any sentence, text objects, whole document, literature to

provide a foundation technique for text analysis, parsing and semantic analysis in subsequent workshops. Part II introduces spaCy, the second important NLP Python implementation tools not only for teaching and learning (like NLTK) but also widely used for NLP applications including text summarization, information extraction, and Q&A chatbot. It is a critical mass to integrate with Transformer Technology in subsequent workshops.

- Chapter 12: Workshop#3 Part-of-Speech Tagging with Natural Language Toolkit (Hour 5–6)

In Chap. 3, we studied basic concepts and theories related to Part-of-Speech (POS) and various POS tagging techniques. This workshop explores how to implement POS tagging by using NLTK starting from a simple recap on tokenization techniques and two fundamental processes in word-level progressing: stemming and stop-word removal, which will introduce two types of stemming techniques: Porter Stemmer and Snowball Stemmer that can be integrated with WordCloud commonly used in data visualization followed by the main theme of this workshop with the introduction of PENN Treebank Tagset and to create your own POS tagger.

- Chapter 13: Workshop#4 Semantic Analysis and Word Vectors using spaCy (Hour 7–8)

In Chaps. 5 and 6, we studied the basic concepts and theories related to meaning representation and semantic analysis. This workshop explores how to use spaCy technology to perform semantic analysis starting from a revisit on word vectors concept, implement and pre-train them followed by the study of similarity method and other advanced semantic analysis.

- Chapter 14: Workshop#5 Sentiment Analysis and Text Classification (Hour 9–10)

This is a coherent workshop of Chap. 7, this workshop explores how to position NLP implementation techniques into two important NLP applications: text classification and sentiment analysis. TensorFlow and Kera are two vital components to implement Long Short-Term Memory networks (LSTM networks), a commonly used Recurrent Neural Networks (RNN) on machine learning especially in NLP applications.

- Chapter 15: Workshop#6 Transformers with spaCy and TensorFlow (Hour 11–12)

In Chap. 8, the basic concept about Transfer Learning, its motivation and related background knowledge such as Recurrent Neural Networks (RNN) with Transformer Technology and BERT model are introduced. This workshop explores how to put these concepts and theories into practice. More importantly, is to implement Transformers, BERT Technology with the integration of spaCy's Transformer Pipeline Technology and TensorFlow. First, it gives an overview and summation on Transformer and BERT Technology. Second, it explores Transformer implementation with TensorFlow by revisiting Text Classification using BERT model as example. Third, it introduces spaCy's Transformer Pipeline Technology and how to implement Sentiment Analysis and Text Classification system using Transformer Technology.

- Chapter 16: Workshop#7 Building Chatbot with TensorFlow and Transformer Technology (Hour 13–14)

In previous six NLP workshops, we studied NLP implementation tools and techniques ranging from tokenization, N-gram generation to semantic and sentiment analysis with various key NLP Python enabling technologies: NLTK, spaCy, TensorFlow and contemporary Transformer Technology. This final workshop explores how to integrate them for the design and implementation of a live domain-based chatbot system on a movie domain. First, it explores the basis of chatbot system and introduce a knowledge domain—the Cornell Large Movie Conversation Dataset. Second, it conducts a step-by-step implementation of movie chatbot system which involves dialog preprocessing, model construction, attention learning implementation, system integration, and performance evaluation followed by live tests. Finally, it introduces a mini project for this workshop and present related chatbot datasets with resources in summary.

Readers of This Book

This book is both an NLP textbook and NLP Python implementation book tailored for:

- Undergraduates and postgraduates of various disciplines including AI, Computer Science, IT, Data Science, etc.
- Lecturers and tutors teaching NLP or related AI courses.
- NLP, AI scientists and developers who would like to learn NLP basic concepts, practice and implement via Python workshops.
- Readers who would like to learn NLP concepts, practice Python-based NLP workshops using various NLP implementation tools such as NLTK, spaCy, TensorFlow Keras, BERT, and Transformer technology.

How to Use This book?

This book can be served as a textbook for undergraduates and postgraduate courses on Natural Language Processing, and a reference book for general readers who would like to learn key technologies and implement NLP applications with contemporary implementation tools such as NLTK, spaCy, TensorFlow, BERT, and Transformer technology.

Part I (Chaps. 1–9) covers the main course materials of basic concepts and key technologies which include N-gram Language Model, Part-of-Speech Tagging, Syntax and Parsing, Meaning Representation, Semantic Analysis, Pragmatic

Analysis, Transfer Learning and Transformer Technology, and Major NLP Applications. Part II (Chaps. 10–16) provides materials for a 14-h, step-by-step Python-based NLP implementation in 7 workshops.

For readers and AI scientists, this book can be served as both reference in learning NLP and Python implementation toolbox on NLP applications by using the latest Python-based NLP development tools, platforms, and libraries.

For seven NLP Workshops in Part II (Chaps. 10–16), readers can download all JupyterNB files and data files from my NLP GitHub directory: <https://github.com/raymondshtlee/nlp/>. For any query, please feel free to contact me via email: raymondshtlee@uic.edu.cn.

Zhuhai, China

Raymond S. T. Lee

Acknowledgements

I would like to express my gratitude:

To my wife Iris for her patience, encouragement, and understanding, especially during my time spent on research and writing in the past 30 years.

To Ms. Celine Cheng, executive editor of Springer NATURE and her professional editorial and book production team for their support, valuable comments, and advice.

To Prof. Tang Tao, President of UIC, for the provision of excellent environment for research, teaching, and writing this book.

To Prof. Weijia Jia, Vice President (Research and Development) of UIC for their supports for R&D of NLP and related AI projects.

To Prof. Jianxin Pan, Dean of Faculty of Science and Technology of UIC, and Prof. Weifeng Su, Head of Department of Computer Science of UIC for their continuous supports for AI and NLP courses.

To research assistant Mr. Zihao Huang for the help of NLP workshops preparation. To research student Ms. Clarissa Shi and student helpers Ms. Siqi Liu, Mr. Mingjie Wang, and Ms. Jie Lie to help with literature review on major NLP applications and Transformer technology, and Mr. Zhuohui Chen to help for bugs fixing and version update for the workshop programs.

To UIC for the prominent support in part by the Guangdong Provincial Key Laboratory IRADS (2022B1212010006, R0400001-22), Key Laboratory for Artificial Intelligence and Multi-Model Data Processing of Department of Education of Guangdong Province and Guangdong Province F1 project grant on Curriculum Development and Teaching Enhancement on course development UICR0400050-21 CTL for the provision of an excellent environment and computer facilities for the preparation of this book.

Dr. Raymond Lee

December 2022

Beijing Normal University-Hong Kong Baptist University United
International College
Zhuhai
China

About the Book

This textbook presents an up-to-date and comprehensive overview of Natural Language Processing (NLP) from basic concepts to core algorithms and key applications. It contains 7 step-by-step workshops (total 14 h) to practice essential Python tools like NLTK, spaCy, TensorFlow Keras, Transformer, and BERT.

The objective of this book is to provide readers with fundamental knowledge, core technologies, and enable to build their own applications (e.g. Chatbot systems) using Python-based NLP tools. It is both a textbook and toolbook intended for undergraduate students from various disciplines who want to learn, lecturers and tutors who want to teach courses or tutorials for undergraduate/graduate students on the subject and related AI topics, and readers with various backgrounds who want to learn and build practicable applications after completing 14 h Python-based workshops.

Contents

Part I Concepts and Technology

1	Natural Language Processing	3
1.1	Introduction	3
1.2	Human Language and Intelligence	4
1.3	Linguistic Levels of Human Language	6
1.4	Human Language Ambiguity	7
1.5	A Brief History of NLP	8
1.5.1	First Stage: Machine Translation (Before 1960s)	8
1.5.2	Second Stage: Early AI on NLP from 1960s to 1970s	8
1.5.3	Third Stage: Grammatical Logic on NLP (1970s–1980s)	9
1.5.4	Fourth Stage: AI and Machine Learning (1980s–2000s)	9
1.5.5	Fifth Stage: AI, Big Data, and Deep Networks (2010s–Present)	10
1.6	NLP and AI	10
1.7	Main Components of NLP	11
1.8	Natural Language Understanding (NLU)	12
1.8.1	Speech Recognition	13
1.8.2	Syntax Analysis	13
1.8.3	Semantic Analysis	13
1.8.4	Pragmatic Analysis	13
1.9	Potential Applications of NLP	14
1.9.1	Machine Translation (MT)	14
1.9.2	Information Extraction (IE)	15
1.9.3	Information Retrieval (IR)	15
1.9.4	Sentiment Analysis	15
1.9.5	Question-Answering (Q&A) Chatbots	16
	References	16

2	N-Gram Language Model	19
2.1	Introduction	19
2.2	N-Gram Language Model	21
2.2.1	Basic NLP Terminology	22
2.2.2	Language Modeling and Chain Rule	24
2.3	Markov Chain in N-Gram Model	26
2.4	Live Example: The Adventures of Sherlock Holmes	27
2.5	Shannon's Method in N-Gram Model	31
2.6	Language Model Evaluation and Smoothing Techniques	34
2.6.1	Perplexity	34
2.6.2	Extrinsic Evaluation Scheme	35
2.6.3	Zero Counts Problems	35
2.6.4	Smoothing Techniques	36
2.6.5	Laplace (Add-One) Smoothing	36
2.6.6	Add-k Smoothing	38
2.6.7	Backoff and Interpolation Smoothing	39
2.6.8	Good Turing Smoothing	40
	References	41
3	Part-of-Speech (POS) Tagging	43
3.1	What Is Part-of-Speech (POS)?	43
3.1.1	Nine Major POS in English Language	43
3.2	POS Tagging	44
3.2.1	What Is POS Tagging in Linguistics?	44
3.2.2	What Is POS Tagging in NLP?	45
3.2.3	POS Tags Used in the PENN Treebank Project	45
3.2.4	Why Do We Care About POS in NLP?	46
3.3	Major Components in NLU	48
3.3.1	Computational Linguistics and POS	48
3.3.2	POS and Semantic Meaning	49
3.3.3	Morphological and Syntactic Definition of POS	49
3.4	9 Key POS in English	50
3.4.1	English Word Classes	51
3.4.2	What Is a Preposition?	51
3.4.3	What Is a Conjunction?	52
3.4.4	What Is a Pronoun?	53
3.4.5	What Is a Verb?	53
3.5	Different Types of POS Tagset	56
3.5.1	What Is Tagset?	56
3.5.2	Ambiguous in POS Tags	57
3.5.3	POS Tagging Using Knowledge	58
3.6	Approaches for POS Tagging	58
3.6.1	Rule-Based Approach POS Tagging	58
3.6.2	Example of Rule-Based POS Tagging	59

3.6.3	Example of Stochastic-Based POS Tagging	60
3.6.4	Hybrid Approach for POS Tagging Using Brill Taggers.	61
3.7	Taggers Evaluations.	63
3.7.1	How Good Is an POS Tagging Algorithm?	64
	References.	65
4	Syntax and Parsing	67
4.1	Introduction and Motivation	67
4.2	Syntax Analysis	68
4.2.1	What Is Syntax.	68
4.2.2	Syntactic Rules.	68
4.2.3	Common Syntactic Patterns.	69
4.2.4	Importance of Syntax and Parsing in NLP	70
4.3	Types of Constituents in Sentences	70
4.3.1	What Is Constituent?	70
4.3.2	Kinds of Constituents.	72
4.3.3	Noun-Phrase (NP)	72
4.3.4	Verb-Phrase (VP)	72
4.3.5	Complexity on Simple Constituents	73
4.3.6	Verb Phrase Subcategorization	74
4.3.7	The Role of Lexicon in Parsing	75
4.3.8	Recursion in Grammar Rules.	76
4.4	Context-Free Grammar (CFG).	76
4.4.1	What Is Context-Free Language (CFL)?	76
4.4.2	What Is Context-Free Grammar (CFG)?	77
4.4.3	Major Components of CFG	77
4.4.4	Derivations Using CFG	78
4.5	CFG Parsing.	79
4.5.1	Morphological Parsing.	79
4.5.2	Phonological Parsing	79
4.5.3	Syntactic Parsing	79
4.5.4	Parsing as a Kind of Tree Searching	80
4.5.5	CFG for Fragment of English	80
4.5.6	Parse Tree for “Play the Piano” for Prior CFG	80
4.5.7	Top-Down Parser	81
4.5.8	Bottom-Up Parser	82
4.5.9	Control of Parsing	84
4.5.10	Pros and Cons of Top-Down vs. Bottom-Up Parsing	84
4.6	Lexical and Probabilistic Parsing.	85
4.6.1	Why Using Probabilities in Parsing?	85
4.6.2	Semantics with Parsing	86
4.6.3	What Is PCFG?	87
4.6.4	A Simple Example of PCFG	87

4.6.5	Using Probabilities for Language Modeling	90
4.6.6	Limitations for PCFG	90
4.6.7	The Fix: Lexicalized Parsing	91
	References	94
5	Meaning Representation	95
5.1	Introduction	95
5.2	What Is Meaning?	95
5.3	Meaning Representations	96
5.4	Semantic Processing	97
5.5	Common Meaning Representation	98
5.5.1	First-Order Predicate Calculus (FOPC)	98
5.5.2	Semantic Networks	98
5.5.3	Conceptual Dependency Diagram (CDD)	99
5.5.4	Frame-Based Representation	99
5.6	Requirements for Meaning Representation	100
5.6.1	Verifiability	100
5.6.2	Ambiguity	100
5.6.3	Vagueness	101
5.6.4	Canonical Forms	101
5.7	Inference	102
5.7.1	What Is Inference?	102
5.7.2	Example of Inferencing with FOPC	103
5.8	Fillmore's Theory of Universal Cases	103
5.8.1	What Is Fillmore's Theory of Universal Cases?	104
5.8.2	Major Case Roles in Fillmore's Theory	105
5.8.3	Complications in Case Roles	106
5.9	First-Order Predicate Calculus	107
5.9.1	FOPC Representation Scheme	107
5.9.2	Major Elements of FOPC	107
5.9.3	Predicate-Argument Structure of FOPC	108
5.9.4	Meaning Representation Problems in FOPC	110
5.9.5	Inferencing Using FOPC	111
	References	113
6	Semantic Analysis	115
6.1	Introduction	115
6.1.1	What Is Semantic Analysis?	115
6.1.2	The Importance of Semantic Analysis in NLP	116
6.1.3	How Human Is Good in Semantic Analysis?	116
6.2	Lexical Vs Compositional Semantic Analysis	117
6.2.1	What Is Lexical Semantic Analysis?	117
6.2.2	What Is Compositional Semantic Analysis?	117
6.3	Word Senses and Relations	118
6.3.1	What Is Word Sense?	118
6.3.2	Types of Lexical Semantics	119

6.4	Word Sense Disambiguation	123
6.4.1	What Is Word Sense Disambiguation (WSD)?	123
6.4.2	Difficulties in Word Sense Disambiguation.	123
6.4.3	Method for Word Sense Disambiguation.	124
6.5	WordNet and Online Thesauri	126
6.5.1	What Is WordNet?	126
6.5.2	What Is Synsets?	126
6.5.3	Knowledge Structure of WordNet	127
6.5.4	What Are Major Lexical Relations Captured in WordNet?	129
6.5.5	Applications of WordNet and Thesauri?	129
6.6	Other Online Thesauri: MeSH	130
6.6.1	What Is MeSH?	130
6.6.2	Uses of the MeSH Ontology	131
6.7	Word Similarity and Thesaurus Methods.	131
6.8	Introduction	131
6.8.1	Path-based Similarity	132
6.8.2	Problems with Path-based Similarity.	133
6.8.3	Information Content Similarity	134
6.8.4	The Resnik Method	135
6.8.5	The Dekang Lin Method	135
6.8.6	The (Extended) Lesk Algorithm	136
6.9	Distributed Similarity.	137
6.9.1	Distributional Models of Meaning.	137
6.9.2	Word Vectors	137
6.9.3	Term-Document Matrix	137
6.9.4	Point-wise Mutual Information (PMI).	139
6.9.5	Example of Computing PPMI on a Term-Context Matrix.	140
6.9.6	Weighing PMI Techniques.	141
6.9.7	K-Smoothing in PMI Computation	142
6.9.8	Context and Word Similarity Measurement.	144
6.9.9	Evaluating Similarity	145
	References.	146
7	Pragmatic Analysis and Discourse	149
7.1	Introduction	149
7.2	Discourse Phenomena	149
7.2.1	Coreference Resolution	150
7.2.2	Why Is it Important?	150
7.2.3	Coherence and Coreference.	151
7.2.4	Importance of Coreference Relations	152
7.2.5	Entity-Based Coherence.	153
7.3	Discourse Segmentation.	154
7.3.1	What Is Discourse Segmentation?	154

7.3.2	Unsupervised Discourse Segmentation	154
7.3.3	Hearst's TextTiling Method	155
7.3.4	TextTiling Algorithm	157
7.3.5	Supervised Discourse Segmentation	158
7.4	Discourse Coherence	158
7.4.1	What Makes a Text Coherent?	158
7.4.2	What Is Coherence Relation?	159
7.4.3	Types of Coherence Relations	159
7.4.4	Hierarchical Structure of Discourse Coherence.	160
7.4.5	Types of Referring Expressions	161
7.4.6	Features for Filtering Potential Referents	162
7.4.7	Preferences in Pronoun Interpretation	162
7.5	Algorithms for Coreference Resolution.	163
7.5.1	Introduction	163
7.5.2	Hobbs Algorithm	163
7.5.3	Centering Algorithm	166
7.5.4	Machine Learning Method.	169
7.6	Evaluation	171
	References.	172
8	Transfer Learning and Transformer Technology.	175
8.1	What Is Transfer Learning?	175
8.2	Motivation of Transfer Learning	176
8.2.1	Categories of Transfer Learning	176
8.3	Solutions of Transfer Learning	178
8.4	Recurrent Neural Network (RNN).	180
8.4.1	What Is RNN?	180
8.4.2	Motivation of RNN	180
8.4.3	RNN Architecture	181
8.4.4	Long Short-Term Memory (LSTM) Network	183
8.4.5	Gate Recurrent Unit (GRU).	185
8.4.6	Bidirectional Recurrent Neural Networks (BRNNs).	186
8.5	Transformer Technology	188
8.5.1	What Is Transformer?	188
8.5.2	Transformer Architecture.	188
8.5.3	Deep Into Encoder	189
8.6	BERT	192
8.6.1	What Is BERT?	192
8.6.2	Architecture of BERT	192
8.6.3	Training of BERT.	192
8.7	Other Related Transformer Technology.	194
8.7.1	Transformer-XL.	194
8.7.2	ALBERT	195
	References.	196

9	Major NLP Applications	199
9.1	Introduction	199
9.2	Information Retrieval Systems	199
9.2.1	Introduction to IR Systems	199
9.2.2	Vector Space Model in IR	200
9.2.3	Term Distribution Models in IR	202
9.2.4	Latent Semantic Indexing in IR	207
9.2.5	Discourse Segmentation in IR	208
9.3	Text Summarization Systems	212
9.3.1	Introduction to Text Summarization Systems	212
9.3.2	Text Summarization Datasets	214
9.3.3	Types of Summarization Systems	214
9.3.4	Query-Focused Vs Generic Summarization Systems	215
9.3.5	Single and Multiple Document Summarization	217
9.3.6	Contemporary Text Summarization Systems	218
9.4	Question-and-Answering Systems	224
9.4.1	QA System and AI	224
9.4.2	Overview of Industrial QA Systems	228
	References	236

Part II Natural Language Processing Workshops with Python Implementation in 14 Hours

10	Workshop#1 Basics of Natural Language Toolkit (Hour 1–2)	243
10.1	Introduction	243
10.2	What Is Natural Language Toolkit (NLTK)?	243
10.3	A Simple Text Tokenization Example Using NLTK	244
10.4	How to Install NLTK?	245
10.5	Why Using Python for NLP?	246
10.6	NLTK with Basic Text Processing in NLP	248
10.7	Simple Text Analysis with NLTK	249
10.8	Text Analysis Using Lexical Dispersion Plot	253
10.8.1	What Is a Lexical Dispersion Plot?	253
10.8.2	Lexical Dispersion Plot Over Context Using Sense and Sensibility	253
10.8.3	Lexical Dispersion Plot Over Time Using Inaugural Address Corpus	254
10.9	Tokenization in NLP with NLTK	255
10.9.1	What Is Tokenization in NLP?	255
10.9.2	Different Between Tokenize() vs Split()	256
10.9.3	Count Distinct Tokens	257
10.9.4	Lexical Diversity	258
10.10	Basic Statistical Tools in NLTK	260
10.10.1	Frequency Distribution: FreqDist()	260

10.10.2	Rare Words: Hapax	262
10.10.3	Collocations	263
	References	265
11	Workshop#2 N-grams in NLTK and Tokenization in SpaCy (Hour 3–4)	267
11.1	Introduction	267
11.2	What Is N-Gram?	267
11.3	Applications of N-Grams in NLP	268
11.4	Generation of N-Grams in NLTK	268
11.5	Generation of N-Grams Statistics	270
11.6	spaCy in NLP	276
11.6.1	What Is spaCy?	276
11.7	How to Install spaCy?	277
11.8	Tokenization using spaCy	278
11.8.1	Step 1: Import spaCy Module	278
11.8.2	Step 2: Load spaCy Module "en_core_web_sm"	278
11.8.3	Step 3: Open and Read Text File "Adventures_Holmes.txt" Into file_handler "fholmes"	278
11.8.4	Step 4: Read Adventures of Sherlock Holmes	278
11.8.5	Step 5: Replace All Newline Symbols	279
11.8.6	Step 6: Simple Counting	279
11.8.7	Step 7: Invoke nlp() Method in spaCy	280
11.8.8	Step 8: Convert Text Document Into Sentence Object	280
11.8.9	Step 9: Directly Tokenize Text Document	282
	References	284
12	Workshop#3 POS Tagging Using NLTK (Hour 5–6)	285
12.1	Introduction	285
12.2	A Revisit on Tokenization with NLTK	285
12.3	Stemming Using NLTK	288
12.3.1	What Is Stemming?	288
12.3.2	Why Stemming?	289
12.3.3	How to Perform Stemming?	289
12.3.4	Porter Stemmer	289
12.3.5	Snowball Stemmer	291
12.4	Stop-Words Removal with NLTK	292
12.4.1	What Are Stop-Words?	292
12.4.2	NLTK Stop-Words List	292
12.4.3	Try Some Texts	294
12.4.4	Create Your Own Stop-Words	295
12.5	Text Analysis with NLTK	296

12.6	Integration with WordCloud	299
12.6.1	What Is WordCloud?	299
12.7	POS Tagging with NLTK	301
12.7.1	What Is POS Tagging?	301
12.7.2	Universal POS Tagset	301
12.7.3	PENN Treebank Tagset (English and Chinese)	302
12.7.4	Applications of POS Tagging	303
12.8	Create Own POS Tagger with NLTK	306
	References	312
13	Workshop#4 Semantic Analysis and Word Vectors Using spaCy	
	(Hour 7–8)	313
13.1	Introduction	313
13.2	What Are Word Vectors?	313
13.3	Understanding Word Vectors	314
13.3.1	Example: A Simple Word Vector	314
13.4	A Taste of Word Vectors	316
13.5	Analogies and Vector Operations	319
13.6	How to Create Word Vectors?	320
13.7	spaCy Pre-trained Word Vectors	320
13.8	Similarity Method in Semantic Analysis	323
13.9	Advanced Semantic Similarity Methods with spaCy	326
13.9.1	Understanding Semantic Similarity	326
13.9.2	Euclidian Distance	326
13.9.3	Cosine Distance and Cosine Similarity	327
13.9.4	Categorizing Text with Semantic Similarity	329
13.9.5	Extracting Key Phrases	330
13.9.6	Extracting and Comparing Named Entities	331
	References	333
14	Workshop#5 Sentiment Analysis and Text Classification	
	with LSTM Using spaCy (Hour 9–10)	335
14.1	Introduction	335
14.2	Text Classification with spaCy and LSTM Technology	335
14.3	Technical Requirements	336
14.4	Text Classification in a Nutshell	336
14.4.1	What Is Text Classification?	336
14.4.2	Text Classification as AI Applications	337
14.5	Text Classifier with spaCy NLP Pipeline	338
14.5.1	TextCategorizer Class	339
14.5.2	Formatting Training Data for the TextCategorizer	340
14.5.3	System Training	344
14.5.4	System Testing	346
14.5.5	Training TextCategorizer for Multi-Label Classification	347

14.6	Sentiment Analysis with spaCy	351
14.6.1	IMDB Large Movie Review Dataset	351
14.6.2	Explore the Dataset	351
14.6.3	Training the TextClassifier	355
14.7	Artificial Neural Network in a Nutshell.	357
14.8	An Overview of TensorFlow and Keras.	358
14.9	Sequential Modeling with LSTM Technology.	358
14.10	Keras Tokenizer in NLP.	359
14.10.1	Embedding Words	363
14.11	Movie Sentiment Analysis with LSTM Using Keras and spaCy.	364
14.11.1	Step 1: Dataset	365
14.11.2	Step 2: Data and Vocabulary Preparation.	366
14.11.3	Step 3: Implement the Input Layer	368
14.11.4	Step 4: Implement the Embedding Layer	368
14.11.5	Step 5: Implement the LSTM Layer	368
14.11.6	Step 6: Implement the Output Layer	369
14.11.7	Step 7: System Compilation	369
14.11.8	Step 8: Model Fitting and Experiment Evaluation	370
	References.	371
15	Workshop#6 Transformers with spaCy and TensorFlow (Hour 11–12)	373
15.1	Introduction	373
15.2	Technical Requirements.	373
15.3	Transformers and Transfer Learning in a Nutshell	374
15.4	Why Transformers?	375
15.5	An Overview of BERT Technology.	377
15.5.1	What Is BERT?	377
15.5.2	BERT Architecture.	378
15.5.3	BERT Input Format	378
15.5.4	How to Train BERT?	380
15.6	Transformers with TensorFlow	382
15.6.1	HuggingFace Transformers	382
15.6.2	Using the BERT Tokenizer	383
15.6.3	Word Vectors in BERT.	386
15.7	Revisit Text Classification Using BERT	388
15.7.1	Data Preparation.	388
15.7.2	Start the BERT Model Construction	389
15.8	Transformer Pipeline Technology	392
15.8.1	Transformer Pipeline for Sentiment Analysis	393
15.8.2	Transformer Pipeline for QA System	393
15.9	Transformer and spaCy	394
	References.	398

16 Workshop#7 Building Chatbot with TensorFlow and Transformer Technology (Hour 13–14)	401
16.1 Introduction	401
16.2 Technical Requirements	401
16.3 AI Chatbot in a Nutshell	402
16.3.1 What Is a Chatbot?	402
16.3.2 What Is a Wake Word in Chatbot?	403
16.3.3 NLP Components in a Chatbot	404
16.4 Building Movie Chatbot by Using TensorFlow and Transformer Technology	404
16.4.1 The Chatbot Dataset	405
16.4.2 Movie Dialog Preprocessing	405
16.4.3 Tokenization of Movie Conversation	407
16.4.4 Filtering and Padding Process	408
16.4.5 Creation of TensorFlow Movie Dataset Object (mDS)	409
16.4.6 Calculate Attention Learning Weights	410
16.4.7 Multi-Head-Attention (MHAttention)	411
16.4.8 System Implementation	412
16.5 Related Works	430
References	431
Index	433

About the Author

Raymond Lee is the founder of the Quantum Finance Forecast System (QFFC) (<https://qffc.uic.edu.cn>) and currently an Associate Professor at United International College (UIC) with 25+ years' experience in AI research and consultancy, Chaotic Neural Networks, NLP, Intelligent Fintech Systems, Quantum Finance, and Intelligent E-Commerce Systems. He has published over 100 publications and authored 8 textbooks in the fields of AI, chaotic neural networks, AI-based fintech systems, intelligent agent technology, chaotic cryptosystems, ontological agents, neural oscillators, biometrics, and weather simulation and forecasting systems. Upon completion of the QFFC project, in 2018 he joined United International College (UIC), China, to pursue further R&D work on AI-Fintech and to share his expertise in AI-Fintech, chaotic neural networks, and related intelligent systems with fellow students and the community. His three latest textbooks, Quantum Finance: Intelligent Forecast and Trading Systems (2019), Artificial Intelligence in Daily Life (2020), and this NLP book have been adopted as the main textbooks for various AI courses in UIC.

Abbreviations

AI	Artificial intelligence
ASR	Automatic speech recognition
BERT	Bidirectional encoder representations from transformers
BRNN	Bidirectional recurrent neural networks
CDD	Conceptual dependency diagram
CFG	Context-free grammar
CFL	Context-free language
CNN	Convolutional neural networks
CR	Coreference resolution
DNN	Deep neural networks
DT	Determiner
FOPC	First-order predicate calculus
GRU	Gate recurrent unit
HMM	Hidden Markov model
IE	Information extraction
IR	Information retrieval
KAI	Knowledge acquisition and inferencing
LSTM	Long short-term memory
MEMM	Maximum entropy Markov model
MeSH	Medical subject thesaurus
ML	Machine learning
NER	Named entity recognition
NLP	Natural language processing
NLTK	Natural language toolkit
NLU	Natural language understanding
NN	Noun
NNP	Proper noun
Nom	Nominal
NP	Noun phrase
PCFG	Probabilistic context-free grammar
PMI	Pointwise mutual information

POS	Part-of-speech
POST	Part-of-speech tagging
PPMI	Positive pointwise mutual information
Q&A	Question-and-answering
RNN	Recurrent neural networks
TBL	Transformation-based learning
VB	Verb
VP	Verb phrase
WSD	Word sense disambiguation