Natural Language Processing in Action

Understanding, analyzing, and generating text with Python

HOBSON LANE
COLE HOWARD
HANNES MAX HAPKE

MANNING
SHELTER ISLAND

Table of Contents

Front matter

- Foreword
- Preface
- Acknowledgments
- About this Book
- About the Authors
- About the cover Illustration

Part 1. Wordy machines

1 Packets of thought (NLP overview)

- 1.1 Natural language vs. programming language
- 1.2 The magic
  - 1.2.1 Machines that converse
  - 1.2.2 The math
- 1.3 Practical applications
- 1.4 Language through a computer's "eyes"
  - 1.4.1 The language of locks
  - 1.4.2 Regular expressions
  - 1.4.3 A simple chatbot
  - 1.4.4 Another way
- 1.5 A brief overflight of hyperspace
- 1.6 Word order and grammar

1.7 A chatbot natural language pipeline

1.8 Processing in depth

1.9 Natural language IQ

Summary

2 Build your vocabulary (word tokenization)

2.1 Challenges (a preview of stemming)

2.2 Building your vocabulary with a tokenizer

2.2.1 Dot product

2.2.2 Measuring bag-of-words overlap

2.2.3 A token improvement

2.2.4 Extending your vocabulary with n-grams

2.2.5 Normalizing your vocabulary

2.3 Sentiment

2.3.1 VADER—A rule-based sentiment analyzer

2.3.2 Naive Bayes

Summary

3 Math with words (TF-IDF vectors)

3.1 Bag of words

3.2 Vectorizing

3.2.1 Vector spaces

3.3 Zipf's Law

3.4 Topic modeling

3.4.1 Return of Zipf

3.4.2 Relevance ranking

3.4.3 Tools

3.4.4 Alternatives

3.4.5 Okapi BM25

3.4.6 What's next
Summary
4 Finding meaning in word counts (semantic analysis)
4.1 From word counts to topic scores
4.1.1 TF-IDF vectors and lemmatization
4.1.2 Topic vectors
4.1.3 Thought experiment
4.1.4 An algorithm for scoring topics
4.1.5 An LDA classifier
4.2 Latent semantic analysis
4.2.1 Your thought experiment made real
4.3 Singular value decomposition
4.3.1 U—left singular vectors
4.3.2 S—singular values
4.3.3 VT—right singular vectors
4.3.4 SVD matrix orientation
4.3.5 Truncating the topics
4.4 Principal component analysis
4.4.1 PCA on 3D vectors
4.4.2 Stop horsing around and get back to NLP
4.4.3 Using PCA for SMS message semantic analysis
4.4.4 Using truncated SVD for SMS message semantic analysis
4.4.5 How well does LSA work for spam classification?
4.5 Latent Dirichlet allocation (LDiA)
4.5.1 The LDiA idea
4.5.2 LDiA topic model for SMS messages
4.5.3 LDiA + LDA = spam classifier

[illegible]

4.5.4 A fairer comparison: 32 LDIA topics

4.6 Distance and similarity

4.7 Steering with feedback

4.7.1 Linear discriminant analysis

4.8 Topic vector power

4.8.1 Semantic search

4.8.2 Improvements

Summary

Part 2. Deeper learning (neural networks)

5 Baby steps with neural networks (perceptrons and backpropagation)

5.1 Neural networks, the ingredient list

5.1.1 Perceptron

5.1.2 A numerical perceptron

5.1.3 Detour through bias

5.1.4 Let's go skiing—the error surface

5.1.5 Off the chair lift, onto the slope

5.1.6 Let's shake things up a bit

5.1.7 Keras: Neural networks in Python

5.1.8 Onward and deepward

5.1.9 Normalization: input with style

Summary

6 Reasoning with word vectors (Word2vec)

6.1 Semantic queries and analogies

6.1.1 Analogy questions

6.2 Word vectors

6.2.1 Vector-oriented reasoning

Material protegido por copyright

6.2.2 How to compute Word2vec representations

6.2.3 How to use the gensim.word2vec module

6.2.4 How to generate your own word vector representations

6.2.5 Word2vec vs. GloVe (Global Vectors)

6.2.6 fastText

6.2.7 Word2vec vs. LSA

6.2.8 Visualizing word relationships

6.2.9 Unnatural words

6.2.10. Document similarity with Doc2vec

Summary

7 Getting words in order with convolutional neural networks (CNNs)

7.1 Learning meaning

7.2 Toolkit

7.3 Convolutional neural nets

7.3.1 Building blocks

7.3.2 Step size (stride)

7.3.3 Filter composition

7.3.4 Padding

7.3.5 Learning

7.4 Narrow windows indeed

7.4.1 Implementation in Keras: prepping the data

7.4.2 Convolutional neural network architecture

7.4.3 Pooling

7.4.4 Dropout

7.4.5 The cherry on the sundae

7.4.6 Let's get to learning (training)

7.4.7 Using the model in a pipeline

Material protected by copyright

7.4.8 Where do you go from here?

Summary

8 *Loopy (recurrent) neural networks (RNNs)*

8.1 Remembering with recurrent networks

8.1.1 Backpropagation through time

8.1.2 When do we update what?

8.1.3 Recap

8.1.4 There's always a catch

8.1.5 Recurrent neural net with Keras

8.2 Putting things together

8.3 Let's get to learning our past selves

8.4 Hyperparameters

8.5 Predicting

8.5.1 Statefulness

8.5.2 Two-way street

8.5.3 What is this thing?

Summary

9 *Improving retention with long short-term memory networks*

9.1 LSTM

9.1.1 Backpropagation through time

9.1.2 Where does the rubber hit the road?

9.1.3 Dirty data

9.1.4 Back to the dirty data

9.1.5 Words are hard. Letters are easier.

9.1.6 My turn to chat

9.1.7 My turn to speak more clearly

9.1.8 Learned how to say, but not yet what

9.1.9 Other kinds of memory

9.1.10. Going deeper

Summary

10 Sequence-to-sequence models and attention

10.1 Encoder-decoder architecture

10.1.1 Decoding thought

10.1.2 Look familiar?

10.1.3 Sequence-to-sequence conversation

10.1.4 LSTM review

10.2 Assembling a sequence-to-sequence pipeline

10.2.1 Preparing your dataset for the sequence-to-sequence training

10.2.2 Sequence-to-sequence model in Keras

10.2.3 Sequence encoder

10.2.4 Thought decoder

10.2.5 Assembling the sequence-to-sequence network

10.3 Training the sequence-to-sequence network

10.3.1 Generate output sequences

10.4 Building a chatbot using sequence-to-sequence networks

10.4.1 Preparing the corpus for your training

10.4.2 Building your character dictionary

10.4.3 Generate one-hot encoded training sets

10.4.4 Train your sequence-to-sequence chatbot

10.4.5 Assemble the model for sequence generation

10.4.6 Predicting a sequence

10.4.7 Generating a response

10.4.8 Converse with your chatbot

10.5 Enhancements

Material protegido por copyright

10.5.1 Reduce training complexity with bucketing

10.5.2 Paying attention

10.6 In the real world

Summary

Part 3. Getting real (real-world NLP challenges)

***11** Information extraction (named entity extraction and question answering)*

11.1 Named entities and relations

11.1.1 A knowledge base

11.1.2 Information extraction

11.2 Regular patterns

11.2.1 Regular expressions

11.2.2 Information extraction as ML feature extraction

11.3 Information worth extracting

11.3.1 Extracting GPS locations

11.3.2 Extracting dates

11.4 Extracting relationships (relations)

11.4.1 Part-of-speech (POS) tagging

11.4.2 Entity name normalization

11.4.3 Relation normalization and extraction

11.4.4 Word patterns

11.4.5 Segmentation

11.4.6 Why won't split('.!?') work?

11.4.7 Sentence segmentation with regular expressions

11.5 In the real world

Summary

***12** Getting chatty (dialog engines)*

12.1 Language skill
12.1.1 Modern approaches
12.1.2 A hybrid approach
12.2 Pattern-matching approach
12.2.1 A pattern-matching chatbot with AIML
12.2.2 A network view of pattern matching
12.3 Grounding
12.4 Retrieval (search)
12.4.1 The context challenge
12.4.2 Example retrieval-based chatbot
12.4.3 A search-based chatbot
12.5 Generative models
12.5.1 Chat about NLPIA
12.5.2 Pros and cons of each approach
12.6 Four-wheel drive
12.6.1 The Will to succeed
12.7 Design process
12.8 Trickery
12.8.1 Ask questions with predictable answers
12.8.2 Be entertaining
12.8.3 When all else fails, search
12.8.4 Being popular
12.8.5 Be a connector
12.8.6 Getting emotional
12.9 In the real world
Summary
13 *Scaling up (optimization, parallelization, and batch processing)*

Material protegido por copyright

13.1 Too much of a good thing (data)
13.2 Optimizing NLP algorithms
13.2.1 Indexing
13.2.2 Advanced indexing
13.2.3 Advanced indexing with Annoy
13.2.4 Why use approximate indexes at all?
13.2.5 An indexing workaround: discretizing
13.3 Constant RAM algorithms
13.3.1 Gensim
13.3.2 Graph computing
13.4 Parallelizing your NLP computations
13.4.1 Training NLP models on GPUs
13.4.2 Renting vs. buying
13.4.3 GPU rental options
13.4.4 Tensor processing units
13.5 Reducing the memory footprint during model training
13.6 Gaining model insights with TensorBoard
13.6.1 How to visualize word embeddings
Summary

Appendix A. Your NLP tools
A.1 Anaconda3
A.2 Install NLPIA
A.3 IDE
A.4 Ubuntu package manager
A.5 Mac
A.5.1 A Mac package manager

A.5.2 Some packages

A.5.3 Tuneups

A.6 Windows

A.6.1 Get Virtual

A.7 NLPIA automagic

Appendix B. *Playful Python and regular expressions*

B.1 Working with strings

B.1.1 String types (str and bytes)

B.1.2 Templates in Python (.format())

B.2 Mapping in Python (dict and OrderedDict)

B.3 Regular expressions

B.3.1 |—OR

B.3.2 ()—Groups

B.3.3 []—Character classes

B.4 Style

B.5 Mastery

Appendix C. *Vectors and matrices (linear algebra fundamentals)*

C.1 Vectors

C.1.1 Distances

Appendix D. *Machine learning tools and techniques*

D.1 Data selection and avoiding bias

D.2 How fit is fit?

D.3 Knowing is half the battle

D.4 Cross-fit training

D.5 Holding your model back

D.5.1 Regularization

D.5.2 Dropout

D.5.3 Batch normalization
D.6 Imbalanced training sets
D.6.1 Oversampling
D.6.2 Undersampling
D.6.3 Augmenting your data
D.7 Performance metrics
D.7.1 Measuring classifier performance
D.7.2 Measuring regressor performance
D.8 Pro tips
Appendix E. *Setting up your AWS GPU*
E.1 Steps to create your AWS GPU instance
E.1.1 Cost control
Appendix F. *Locality sensitive hashing*
F.1 High-dimensional vectors are different
F.1.1 Vector space indexes and hashes
F.1.2 High-dimensional thinking
F.2 High-dimensional indexing
F.2.1 Locality sensitive hashing
F.2.2 Approximate nearest neighbors
F.3 "Like" prediction

Resources
Applications and project ideas
Courses and tutorials
Tools and packages
Research papers and talks
Vector space models and semantic search

Material protected by copyright

Finance
Question answering systems
Deep learning
LSTMs and RNNs
Competitions and awards
Datasets
Search engines
Search algorithms
Open source search engines
Open source full-text indexers
Manipulative search engines
Less manipulative search engines
Distributed search engines
Glossary
Acronyms
Terms
Index
List of Figures
List of Tables
List of Listings