INDEX

A

abbreviations, 6
access tokens, 156
accusative form, 188
acomp label, 26
add_info() function, 165
AI ecosystem, xvi
amod label, 26
antecedents, 122–125
API keys, 156
application logic, 130–132
authentication tokens, 156
aux label, 26
auxiliary verbs, 188

B

bind variables, 137
bot channels, 156–166
BotFather, 157–158

C

callback functions, 159
cancel() function, 165
categorization tasks, 70–71
cdef functions, 45
chatbots
  context and, 91–95
  database-powered, 132–139
  deploying, 156–166
  questions for, 56–60
  testing, 181–182
  word sequence patterns in, 83–86
children, 186
chunks, 33–35
Clarifai, 176–183
clausal complements, 56
C-level data structures, 42–46
CNNs (convolutional neural networks), 11
coarse-grained parts of speech tags, 22, 49
'colors' option, 104–106
compact mode, 103
compound label, 26
compounding() method, 147
condensing text, 89–91
conjuncts, 114
constituent-based grammars, 185–187
constituents, 185–186
container objects, 31–37
context, 13, 24, 91–93
ConversationHandler object, 162–163
convolutional neural networks (CNNs), 11
coordinates, 64–65
cosine similarity, 3
currency, 49–50
Cython, 42–46

D

databases, 127–139
dative label, 26
dep_pattern function, 78, 85
dependencies, 186
dependency arcs, 27
dependency grammars, 185–187
dependency lables, 24–26. *See also* syntactic dependencies
  acomp, 26
  amod, 26
  aux, 26
  compound, 26
  dative, 26
  det, 26
  dobj, 25–26
  nsubj, 26
  pobj, 26–28
  ROOT, 26–28

dependency parsing
  convolutional neural networks (CNNs) and, 11
  customizing, 149–153
  syntactic dependency parsing, 3, 24
  visualizing, 100–101
dependency tree structures, 5, 86–91
dependent words, 186
descriptive tags, 176–178
det label, 26
determine_question_type function, 60
dimensions, 65–66
direct objects, 112–117, 187
discourse, 122–125
displaCy dependency visualizer, 98–99
displaCy named entity visualizer, 99–100
displaCy web servers, 100–101
displacy.render() method, 106–107
displacy.serve() method, 106–107
DISTRICT label, 40
dobj label, 25–26
Doc objects, 18, 32
doc.noun_chunks containers, 35
doc.sents containers, 33–35

E

echo() function, 159
EntityRecognizer class, 40
'ents' option, 104–106
errors
  missing INTO keyword, 7
  training process and, 142
Explosion AI website, 97
extract_intent() function, 161, 164

F

Facebook Messenger, 157
fastText, 65
financial data, 49–50
fine-grained parts of speech tags, 22, 49
for loops, 34, 53–54, 59
from_disk() method, 42

G

generate_question function, 60
Gensim, xvi
get_utterance function, 86
governors, 186

H

handlers, 156
head/child syntactic relations, 25, 87, 186–187
helper functions, 178–180
hex color codes, 106

I

idle() method, 160
images, 175–178
inflections, 6
information questions, 56–57
INSERT statements, 137, 138
intent identifiers, 112
interchange formats, 129–130
  JSON, 129–133
  XML, 129
inversion, 52

J

JJ tag, 22
JSON, 129–133

K

keyphrase() function, 178–180
keywords, 12, 67–68, 86–91

L

leftward syntactic dependencies, 33
lemmas, 18–21
linguistics, 185–188
loanwords, 6
loss, 142

M

machine learning, xvii, 3–8
main() function, 165–166
Matcher, 79–80
meaning transition, 13
meanings, 65–66
messaging apps, 157

minibatch() method, 147
missing INTO keyword error, 7
modal auxiliary verbs, 188
model training, 4
modifiers, 94–95
money, 49–50
multiple intents, 113–114
MySQL databases, 135–138

N

named entities, 29, 72–74
named entity recognition, 4, 29, 143–144
natural language processing (NLP), xv–xvi, 1–2
Natural Language Toolkit (NLTK), xvi
ner component, 42
ner.add_label() method, 41
neural networks, 9–11
nlp.begin_training() method, 147
nlp.meta attribute, 39
NN tag, 22
NNS tag, 22
nominative form, 188
noun chunks, 33–35
nsubj label, 26
numbers, 133–135
numeric tags, 48–49

O

objects of a preposition, 188
objects vs. subjects, 55–56
optimizers, 147
options argument, 103–106
ORG label, 40
Oxford English Dictionary, 6

P

part-of-speech tags, 3–4, 21–24, 48–54
- JJ, 22
- NN, 22
- NNS, 22
- PRP, 22, 51–54
- PRP$, 22, 51
- VB, 22, 23
- VBD, 22
- VBG, 22, 23
- VBZ, 56

patterns, 76–86
personal pronouns, 52, 188
photo_tags() function, 179–180
photo() function, 181
phrase structure grammars, 185–187
pipelines, 17–18, 37–42, 146
pobj label, 26–28
pos_pattern function, 81, 83–85
possessive pronouns, 52
postmodifiers, 94
predefined lists, 117–119
premodifiers, 94
prepositional objects, 187–188
preprocessing, 71–72, 128
present tense progressive aspect, 22
probability distributions, 8
processing pipelines, 17–18, 37–42
proforms, 122–125
pron_pattern function, 82–85
pronouns, 51–54, 82–83, 188
- personal, 52, 188
- possessive, 52
- reflexive, 188

PRP tag, 22, 51–54
PRP$ tag, 22, 51
punctuation tags, 48–49
python-telegram-bot library, 158–159
.pyx files, 43

Q

question_type variable, 58–59
questions, 56–60, 171–175

R

reflexive pronouns, 188
relational databases, 127–139
rendering, 100
ROOT label, 26–28
row-and-column databases, 127–139

S

scientific terms, 6
semantic relations, 149–153
semantic similarity, 3, 120–121

sentence-by-sentence visualizations, 102–103
shredding the discourse, 28
similarity method, 63, 66–67
slang, 6
spaCy library, xvi–xvii
spacy.explain() function, 48, 143
spacy.load() function, 38
spacy.util.get_lang_class() function, 40
Span objects, 36
span.merge() function, 36
special cases, 21
split infinitives, 6
SQL, 131
start() function, 164, 180
start_polling() method, 160
statistical models, 8–11, 16–17
structured data, 128–132
subjects vs. objects, 55–56
substitutes, 123–124
symbolic tags, 48–49
synonyms, 3, 117–119
syntactic dependencies
  overview, 24–28
  keywords and, 86–91
  leftward, 33
  parsing, 3
  syntactic dependency labels, 24–26. *See also* dependency labels
  text processing, 55–60

T

Telegram, 157–166, 178–183
text content, 18
third-party word vectors, 68–69
to_disk() method, 42, 148
token.conjuncts, 113–114
tokens, 18, 32
training examples, 142–144
transitive verbs, 112–117, 187

U

unstructured data, 128–132
user's intent, 12
utterances, 77–79

V

VB tag, 22, 23
VBD tag, 22
VBG tag, 22, 23
VBZ tag, 56
vectors, 64–65
verbs
  auxiliary, 188
  modal auxiliary, 188
  transitive, 112–117, 187
visualizations
  customizing, 103–106
  dependency parsing, 100–101
  displaCy dependency visualizer, 98–99
  displaCy named entity visualizer, 99–100
  exporting, 106–107
  sentence-by-sentence, 102–103
vocab objects, 32

W

web color names, 106
while loops, 50
wiki() function, 180
Wikipedia, 170–175, 178–183
word embedding, 2–3
word lists, 115–117
word sequence patterns, 76–86
word vectors
  overview, 64–68
  calculating similarity using, 69–74
  installing, 68–69
  third-party word vectors, 68–69
word2int() function, 134
word2vec algorithm, xvi
word-based dependency grammars, 186
working environment set up, 16

X

XML, 129

Y

yes/no questions, 56–57