Index

_

(phrasal) projections *(3.2)*

A

accuracy *(3.2)*
address *(4.5)*
adjectives *(2.6)*
adverbs *(2.6)*
agreement *(1.1)*
alphabetic variants *(4.2)*
anaphora resolution *(5.2)*
anaphoric antecedent *(5.1)*
antecedent *(5.2)*
antonymy *(5.3)*
appending *(2.1)*
appropriate *(5)*
argument *(2)*
arity *(3.3)*
articles *(2.6)*
assignment *(3.5)*
associative array *(3)*
assumption *(2)*
atomic *(1.3)*
attribute value matrix *(1.3)*
auxiliaries *(3.3)*
auxiliary *(1.3)*

B

backoff *(4.3)*
backtracks *(4.1)*
base case *(4.7.1)*
basic types *(3.1)*
bigrams *(3.3)*
binary predicate *(3.1)*
bind *(3.1)*
binding operators *(4.5)*
binning *(5.3)*
BIO Format *(8)*
Bold *(X)*
boolean *(1.3)*
boolean operators *(2)*
bottom-up *(4.7.3)*
bottom-up parsing *(4.1)*
bound *(3.1)*
bound *(3.1)*
breakpoints *(4.6.4)*

C

call-by-value *(4.4.2)*
call *(1.4)*
call structure *(4.7.3)*
Catalan numbers *(6.2)*
characteristic function *(3.4)*
chart *(4.4)*
chart parsing *(4.4)*
child *(4.2)*
chink *(2.5)*
chink *(8)*
chunk grammar *(2.1)*
chunk *(2)*
chunking *(2)*
class label *(1)*
Classification *(1)*
closed class *(7.4)*
closed *(3.1)*
closures *(3.4.2)*
code point *(3.3.1)*
coindex *(2)*
collocation *(3.3)*
comparative wordlist *(4.3)*
complements *(5.1)*
complete *(9)*
complex *(1.3)*
complex types *(3.1)*
components *(5.5)*
concatenation *(3.2.1)*
conclusion *(2)*
conditional expression *(4.3)*
conditional frequency distribution *(2)*
conditional *(6.3)*
confusion matrix *(3.4)*
consecutive classification *(1.6)*
consistent *(1.2)*
constituent *(2.1)*
constituent structure *(2.1)*
control *(4)*
control structure *(4.3)*
Cooper storage *(4.5)*
coordinate structure *(2.1)*
copy *(4.1.1)*
coreferential *(3.1)*
corpora *(0)*
Corpus Linguistics *(7)*
cross-validation *(3.5)*

D

data intensive *(I)*
debugger *(4.6.4)*

decision nodes *(4)*
decision stump *(4)*
decision tree *(4)*
decoding *(3.3.1)*
defensive programming *(4.4.4)*
dependents *(5)*
determiners *(2.6)*
dev-test *(1.2)*
development set *(1.2)*
dialogue acts *(2.2)*
dictionary *(3)*
dictionary *(3.2)*
directed acyclic graphs *(2)*
discourse *(5)*
discourse referents *(5.1)*
discourse representation structure *(5.1)*
dispersion plot *(1.3)*
divide-and-conquer *(4.7)*
docstring *(4.4)*
doctest block *(4.4.6)*
domain *(3.4)*
DRS conditions *(5.1)*
duck typing *(4.3)*
dynamic programming *(4.4)*

E

empiricism *(II)*
encode *(1.1)*
encoding *(3.3.1)*
entails *(5.3)*
equivalent *(2)*
error analysis *(1.2)*
evaluation set *(3.1)*
existential quantifier *(3.1)*
Expected Likelihood Estimation *(5.2)*
export *(3.9.2)*

F

F-Measure *(3.3)*
F-Score *(3.3)*
f-structure *(5)*
False negatives *(3.3)*
False positives *(3.3)*
feature extractor *(1.1)*
feature *(1.2)*
feature path *(2)*
feature set *(1.1)*
feature structures *(1)*
features *(1.1)*
fields *(4.2.2)*
filler *(3.4)*

folds *(3.5)*
formal language theory *(I)*
format string *(3.9.2)*
free *(3.1)*
frequency distribution *(3.1)*
function *(3.2)*

G

gaps *(3.4)*
gazetteer *(5)*
generalized quantifiers *(8)*
generative grammar *(8)*
generative *(6.3)*
generator expression *(4.2.3)*
gerund *(7.1)*
Glue semantics *(7)*
glyphs *(3.3.1)*
gold standard *(4.4)*
graphs *(4.8.2)*
greedy sequence classification *(1.6)*

H

hapaxes *(3.1)*
hash array *(3)*
head features *(6)*
head *(5)*
headword *(4)*
Heldout Estimation *(5.2)*
Hidden Markov Models *(1.7)*
Hole semantics *(7)*
holonyms *(5.3)*
homonyms *(4)*
hyponyms *(5.2)*

I

idealism *(II)*
identifiers *(2.3)*
immediate constituents *(2.1)*
immutable *(4.2.2)*
inconsistent *(1.2)*
indented code block *(1.4)*
independence assumption *(5.1)*
index *(2.2)*
inference *(2)*
Information Extraction *(1)*
information gain *(4.1)*
Inline annotation *(3.5)*
interpreter *(1.1)*
IOB tags *(2.6)*
iterative optimization *(6)*

J

joint classifier *(1.6)*

joint-feature *(6.1)*

K

Kappa *(2.2)*
key *(3.2)*
keyword arguments *(4.5.4)*
Kleene closures *(3.4.2)*

L

lambda expressions *(4.5.1)*
latent semantic analysis *(4.8.4)*
leaf nodes *(4)*
left-corner *(4.3)*
left-corner parser *(4.3)*
left-recursive *(3.3)*
lemma *(4)*
letter trie *(4.7.1)*
lexical acquisition *(9)*
lexical categories *(0)*
lexical entry *(4)*
lexical relations *(5.3)*
lexicon *(10)*
LGB rule *(4.4.3)*
library *(3.3)*
licensed *(3.4)*
likelihood ratios *(1.1)*
Linear-Chain Conditional Random Field Models *(1.7)*
list *(2.1)*
local variables *(3.2)*
logical constants *(3.1)*
logical form *(2)*

M

machine translation *(5.3)*
mapping *(3)*
maximal projection *(3.2)*
Maximum Entropy *(6)*
Maximum Entropy Markov Models *(1.7)*
Maximum Entropy principle *(6.2)*
meronyms *(5.3)*
methods *(3.2)*
modals *(2.6)*
model checking *(3.5)*
model *(1.2)*
models *(7)*
module *(3.3)*
morpho-syntactic *(7.5)*
morphological analysis *(7.5)*
multiword expression *(3.11)*
mutable *(4.2.2)*

N

n-gram tagger *(5.3)*

naive Bayes assumption *(5.1)*
naive Bayes *(5)*
named entity detection *(1.1)*
named entity recognition *(5)*
newlines *(3.1.5)*
NLTK Data Repository *(6.3)*
non-logical constants *(3.1)*
non-standard words *(3.6.2)*
normalized *(3.6)*
noun phrase chunking *(2.1)*
noun phrase *(II)*
NP-chunking *(2.1)*

O

objective function *(3.8.2)*
open class *(7.4)*
open formula *(3.1)*
out-of-vocabulary *(5.5)*
overfit *(4.1)*
overfitting *(1.2)*

P

package *(3.3)*
parameter *(1.4)*
parameters *(5.5)*
parent *(4.2)*
parser *(4)*
part-of-speech tagging *(0)*
partial information *(2.1)*
parts of speech *(0)*
personal pronouns *(2.6)*
phonology *(I)*
phrasal level *(3.2)*
POS-tagger *(1)*
POS-tagging *(0)*
pre-sort *(4.7)*
Precision *(3.3)*
precision/recall trade-off *(5.3)*
predicates *(3.1)*
prepositional phrase attachment ambiguity *(3.1)*
prepositional phrase *(2.1)*
present participle *(7.1)*
principle of compositionality *(I)*
prior probability *(5)*
probabilistic context free grammar *(6.3)*
productions *(1.1)*
projective *(5)*
proof goal *(3.2)*
Propositional logic *(2)*
propositional symbols *(2)*
prune *(4.1)*

Q

question answering *(5.3)*

R

rationalism *(III)*
raw string *(3.4.2)*
realism *(III)*
Recall *(3.3)*
recognizing *(4.4)*
record *(4.2.2)*
recursion *(4.7.1)*
recursive *(3.3)*
reduce *(4.2)*
reentrancy *(2)*
refactor *(4.4.5)*
regression testing *(4.6.5)*
relation detection *(1.1)*
relational operators *(4.1)*
replacement field *(3.9.2)*
return value *(3.2)*
root element *(4.3)*
root node *(4)*
runtime error *(2.2)*

S

S-Retrieval *(4.5)*
satisfies *(3.5)*
scope *(3.7)*
segmentation *(3.8)*
semantic role labeling *(5.2)*
sequence classifier *(1.6)*
sequence *(3.2.6)*
shift *(4.2)*
shift-reduce parser *(4.2)*
siblings *(4.2)*
signature *(3.1)*
slash categories *(3.4)*
slicing *(2.2)*
smoothing *(5.2)*
stack trace *(4.6.4)*
standoff annotation *(2.3)*
standoff annotation *(3.5)*
start-symbol *(3.1)*
stopwords *(4.1)*
string formatting *(3.9.2)*
string *(3.2)*
strings *(2.4)*
structurally ambiguous *(3.1)*
structure sharing *(2)*
structured data *(1)*
stylistics *(1.3)*

subcategorized *(5.1)*
subsumption *(2.1)*
subtype *(5)*
supervised *(1)*
Swadesh wordlists *(4.3)*
symbolic logic *(I)*
synonyms *(5.1)*
synset *(5.1)*
syntax error *(1.1)*

T

T9 *(3.4.2)*
tag *(2)*
tag patterns *(2.2)*
tagged *(2.2)*
tagging *(0)*
tagset *(0)*
terms *(3.1)*
test set *(1.1)*
test set *(3.1)*
text alignment *(5.4)*
textonyms *(3.4.2)*
token *(1.4)*
tokenization *(3.1.1)*
top-down *(4.7.3)*
top-down parsing *(4.1)*
total likelihood *(6)*
train *(5.1)*
training *(5.1)*
training set *(1.1)*
training set *(1.2)*
transitive verbs *(5.1)*
tree *(4.2)*
True negatives *(3.3)*
True positives *(3.3)*
truth-conditions *(2)*
tuple *(4.2)*
Turing Test *(5.5)*
Type I errors *(3.3)*
Type II errors *(3.3)*
type-raising *(4.3)*
Typed feature structures *(5)*
types *(3.1)*

U

unary predicate *(3.1)*
unbounded dependency construction *(3.4)*
underspecified *(1.2)*
unification *(2.1)*
unique beginners *(5.2)*
universal quantifier *(3.1)*

unseen *(7)*
unstructured data *(1)*

V

valencies *(5.1)*
valid *(2)*
validity *(4.1)*
valuation function *(3.4)*
value *(3.2)*
variable *(2.3)*
verb phrase *(2.1)*

W

weights *(5.5)*
well formed formulas *(2)*
well formed *(4.1)*
well-formed substring table *(4.4)*
wildcard *(3.4.1)*
word classes *(0)*
word sense disambiguation *(5.1)*
word type *(1.4)*
WordNet *(5)*

X

XML attribute *(4.1)*
XML element *(4.1)*

Z

zero projection *(3.2)*

A

α-conversion *(4.2)*
α equivalents *(4.2)*

B

β-reduction *(4.2)*

Λ

λ abstraction *(4.2)*
λ operator *(4.2)*

About this document...

UPDATED FOR NLTK 3.0. This is a chapter from *Natural Language Processing with Python*, by Steven Bird, Ewan Klein and Edward Loper, Copyright © 2014 the authors. It is distributed with the *Natural Language Toolkit* [http://nltk.org/], Version 3.0, under the terms of the *Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 United States License* [http://creativecommons.org/licenses/by-nc-nd/3.0/us/].

This document was built on Wed 1 Jul 2015 12:30:05 AEST