

Przedmowa	11
1. Rozproszone uczenie maszynowe. Terminologia i pojęcia	19
Etapy przepływu pracy uczenia maszynowego	22
Narzędzia i technologie w potoku uczenia maszynowego	24
Modele przetwarzania rozproszonego	25
Modele uniwersalne	26
Dedykowane modele przetwarzania rozproszonego	28
Wprowadzenie do architektury systemów rozproszonych	28
Systemy scentralizowane a zdecentralizowane	29
Modele interakcji	30
Komunikacja w środowisku rozproszonym	31
Wprowadzenie do metod uczenia zespołowego	32
Wysoka i niska stronniczość	32
Rodzaje metod zespołowych	33
Topologie szkolenia rozproszonego learner	34
Wyzwania związane z rozproszonymi systemami uczenia maszynowego	36
Wydajność	36
Zarządzanie zasobami	39
Odporność na błędy	40
Prywatność	41
Przenośność	42
Konfiguracja środowiska lokalnego	42
Środowisko samouczków z rozdziałów 2. – 6.	42
Środowisko samouczków z rozdziałów 7. – 10.	44
Podsumowanie	45
2. Wprowadzenie do Sparka i PySparka	46
Architektura Apache Spark	46
Wprowadzenie do PySparka	49

Podstawy Apache Spark	50
Architektura oprogramowania	50
PySpark a programowanie funkcyjne	56
Uruchamianie kodu PySparka	57
Ramki DataFrame biblioteki pandas kontra ramki DataFrame systemu Spark	57
Scikit-Learn kontra MLlib	58
Podsumowanie	59
3. Zarządzanie cyklem życia eksperymentu uczenia maszynowego za pomocą MLflow	60
Wymagania dotyczące zarządzania cyklem życia uczenia maszynowego	61
Czym jest MLflow?	62
Komponenty oprogramowania platformy MLflow	63
Użytkownicy platformy MLflow	64
Komponenty platformy MLflow	65
MLflow Tracking	65
MLflow Projects	68
MLflow Models	69
MLflow Model Registry	70
Korzystanie z platformy MLflow w rozwiązaniach dużej skali	71
Podsumowanie	74
4. Pozyskiwanie danych, wstępne przetwarzanie i statystyki opisowe	75
Pozyskiwanie danych za pomocą Sparka	76
Przetwarzanie obrazów	77
Przetwarzanie danych tabelarycznych	79
Wstępne przetwarzanie danych	79
Przetwarzanie wstępne a właściwe	80
Po co wstępnie przetwarzać dane?	80
Struktury danych	81
Typy danych MLlib	81
Przetwarzanie wstępne z wykorzystaniem transformatorów MLlib	83
Wstępne przetwarzanie danych obrazów	89
Zapisywanie danych i unikanie problemu małych plików	92
Statystyki opisowe: poznawanie danych	94
Obliczanie statystyk	95
Statystyki opisowe z wykorzystaniem obiektu Summarizer Sparka	95
Skośność danych	98
Korelacja	98
Podsumowanie	102

5. Inżynieria cech	103
Cechy i ich wpływ na modele uczenia maszynowego	104
Narzędzia do cechowania w bibliotece MLlib	107
Ekstraktory	108
Selektory	108
Przykład: Word2Vec	109
Proces cechowania obrazów	111
Wykonywanie działań na obrazach	112
Wyodrębnianie cech za pomocą API Sparka	114
Proces cechowania tekstu	119
Worek słów	120
TF-IDF	121
n-gramy	122
Techniki dodatkowe	122
Wzbogacanie zbioru danych	123
Podsumowanie	124
6. Szkolenie modeli za pomocą biblioteki MLlib platformy Spark	125
Algorytmy	125
Nadzorowane uczenie maszynowe	127
Klasyfikacja	127
Regresja	131
Nienadzorowane uczenie maszynowe	136
Wydobywanie częstych wzorców	136
Klasteryzacja	136
Ocena	139
Ewaluatory nadzorowane	140
Ewaluatory nienadzorowane	142
Hiperparametry i eksperymenty dostrajania	143
Budowanie siatki parametrów	143
Podział danych na zbiory szkoleniowe i testowe	144
Walidacja krzyżowa: lepszy sposób testowania modeli	145
Potoki uczenia maszynowego	146
Budowa potoku	148
Jak działa podział dla API Pipeline?	148
Utrwalanie	149
Podsumowanie	149
7. Łączenie Sparka z frameworkami uczenia głębokiego	150
Podejście oparte na danych i dwóch klastrach	153
Implementacja dedykowanej warstwy dostępu do danych	155
Cechy DAL	155
Wybór warstwy DAL	157

Czym jest Petastorm?	157
SparkDatasetConverter	159
Petastorm jako magazyn Parquet	163
Projekt Hydrogen	164
Barierowy tryb wykonania	165
Harmonogramowanie z uwzględnieniem akceleratorów	166
Wprowadzenie do API Horovod Estimator	167
Podsumowanie	168
8. Rozproszone uczenie maszynowe z wykorzystaniem TensorFlow	170
Przegląd podstawowych wywołań API biblioteki TensorFlow	171
Czym jest sieć neuronowa?	173
Role i obowiązki w procesie klastra TensorFlow	174
Ładowanie danych Parquet do zbioru danych TensorFlow	175
Strategie rozproszonego uczenia maszynowego TensorFlow	177
ParameterServerStrategy	179
CentralStorageStrategy: jedna maszyna, wiele procesorów	181
MirroredStrategy: jedna maszyna, wiele procesorów, lokalna kopia	181
MultiWorkerMirroredStrategy: wiele maszyn, tryb synchroniczny	182
TPUStrategy	186
Co się zmienia po zmianie strategii?	186
Szkoleniowe interfejsy API	187
API Keras	187
Niestandardowa pętla szkoleniowa	191
API Estimator	193
Połączmy kropki	194
Rozwiązywanie problemów	196
Podsumowanie	197
9. Rozproszone uczenie maszynowe z wykorzystaniem frameworka PyTorch	198
Przegląd podstaw frameworka PyTorch	199
Graf obliczeniowy	199
Mechanika frameworka PyTorch i związane z nim pojęcia	201
Strategie rozproszonego szkolenia modeli frameworka PyTorch	204
Wprowadzenie do podejścia rozproszonego wykorzystywanego przez framework PyTorch	205
Rozproszone i równoległe szkolenie danych (DDP)	206
Szkolenie rozproszone oparte na RPC	207
Topologie komunikacji frameworka PyTorch (c10d)	215
Do czego można wykorzystać niskopoziomowe wywołania API frameworka PyTorch?	223

Ładowanie danych za pomocą frameworka PyTorch i biblioteki Petastorm	224
Rozwiązywanie problemów podczas korzystania z biblioteki Petastorm i frameworka PyTorch w środowisku rozproszonym	227
Enigma niedopasowanych typów danych	227
Tajemnica marudnych węzłów roboczych	228
Czym PyTorch różni się od TensorFlow?	229
Podsumowanie	230
10. Wzorce wdrażania modeli uczenia maszynowego	231
Wzorce wdrażania	232
Wzorzec 1. Prognozy zbiorcze	232
Wzorzec 2. Model w ramach usługi	233
Wzorzec 3. Model jako usługa	234
Decydowanie o wykorzystywanym wzorcu	235
Wymagania dotyczące oprogramowania produkcyjnego	236
Monitorowanie modeli uczenia maszynowego w produkcji	239
Dryf danych	240
Dryf modelu, dryf koncepcji	243
Przesunięcie dziedziny rozkładu (długi ogon)	244
Jakie wskaźniki należy monitorować w produkcji?	244
W jaki sposób wykorzystać system monitorowania do mierzenia zmian?	245
Jak to wygląda w systemie produkcyjnym?	247
Produkcyjna pętla sprzężenia zwrotnego	248
Wdrażanie z wykorzystaniem biblioteki MLlib	248
Produkcyjne potoki uczenia maszynowego ze strukturalnym przesyłaniem strumieniowym	249
Wdrażanie z wykorzystaniem biblioteki MLflow	251
Definiowanie wrappera MLflow	251
Wdrażanie modelu jako mikrouslugi	254
Ładowanie modelu jako funkcji UDF platformy Spark	255
Jak pracować nad systemem w sposób iteracyjny?	255
Podsumowanie	256
Skorowidz	259