

PyTorch ile Derin Öğrenme: Kavramdan Uygulamaya

Yazar
Yılmaz KAYA



© Copyright 2025

Bu kitabın, basım, yayın ve satış hakları Akademisyen Kitabevi A.Ş.'ne aittir. Anılan kuruluşun izni alınmadan kitabın tümü ya da bölümleri mekanik, elektronik, fotokopi, manyetik kayıt ve/veya başka yöntemlerle çoğaltılamaz, basılamaz, dağıtılamaz. Tablo, şekil ve grafikler izin alınmadan, ticari amaçlı kullanılamaz. Bu kitap T.C. Kültür Bakanlığı bandrolü ile satılmaktadır.

ISBN 978-625-375-506-5	Sayfa ve Kapak Tasarımı Akademisyen Dizgi Ünitesi
Kitap Adı PyTorch ile Derin Öğrenme: Kavramdan Uygulamaya	Yayıncı Sertifika No 47518
Yazar Yılmaz KAYA ORCID iD: 0000-0001-5167-1101	Baskı ve Cilt Vadi Matbaacılık
Yayın Koordinatörü Yasin DİLMEN	Bisac Code EDU039000
	DOI 10.37609/akya.3678

Kütüphane Kimlik Kartı

Kaya, Yılmaz.

PyTorch ile Derin Öğrenme: Kavramdan Uygulamaya / Yılmaz Kaya.

Ankara : Akademisyen Yayınevi Kitabevi, 2025.

359 s. : tablo. ; 160x235 mm.

Kaynakça var.

ISBN 9786253755065

GENEL DAĞITIM
Akademisyen Kitabevi A.Ş.

Halk Sokak 5 / A Yenışehir / Ankara

Tel: 0312 431 16 33

siparis@akademisyen.com

www.akademisyen.com

ÖNSÖZ

Bu kitap, yalnızca bir derin öğrenme eğitimi sunmayı değil; sizi fikirden uygulamaya, koddan projeye, modelden ürüne uzanan bütünlüklü bir yolculuğa çıkarmayı amaçlıyor.

Yapay zekâ artık bir araştırma konusu olmaktan çıkıp gündelik hayatın merkezine yerleşti. Görüntü tanıma, ses analizi, metin yorumlama, hastalık tahmini, üretim hattı kontrolü gibi pek çok alanda derin öğrenme modelleri karar veriyor, yönlendiriyor ve tahmin ediyor. Ancak bu sistemleri sadece kullanmakla kalmayıp sıfırdan kurabilmek, doğru yöntemle eğitebilmek ve gerçek problemlere uygulayabilmek, günümüz mühendislerinin ve araştırmacılarının sahip olması gereken temel bir beceri hâline geldi.

İşte bu kitap, derin öğrenmeyi öğrenmek isteyen herkes için yalın, sistematik ve pratik bir yaklaşım sunmak amacıyla hazırlandı. Konular, en temel kavramlardan başlayarak adım adım yükselen bir yapıda işlenmiştir. Yapay sinir ağlarının nasıl çalıştığını anlamakla başlayıp, gerçek veri kümeleri üzerinde CNN, RNN, Transformer gibi modellerin nasıl eğitileceğini, değerlendirileceğini ve optimize edileceğini öğreniyorsunuz.

Her bölümde yalnızca teori anlatılmıyor; PyTorch kütüphanesi ile kod örnekleri, veri setleriyle uygulamalar ve gerçek dünyadan ilham alınmış problemler yer alıyor. Özellikle transfer öğrenme, hiperparametre optimizasyonu, dağıtık eğitim, model sunumu ve endüstri uygulamaları gibi ileri konular, kitap boyunca örnek projelerle desteklenerek ele alınıyor.

Son bölümde ise, tüm öğrendiklerinizi bir araya getirerek sıfırdan bir bitirme projesi geliştirmeniz teşvik ediliyor. Bu bölümde proje fikrinden veri keşfine, model seçiminden performans değerlendirmeye ve model dağıtımına kadar uçtan uca bir yapay zekâ çözümünün nasıl geliştirileceği gösteriliyor.

Bu kitabı yazarken hedefim; teknik kavramları ezberletmek değil, her okuyucunun bir projeyi kendi başına geliştirebileceği zihinsel altyapıyı oluşturmaya yardımcı olmaktır. PyTorch gibi güçlü bir açık kaynak aracın desteğiyle, siz de artık sadece öğrenen değil, üreten, deneyen ve paylaşan bir yapay zekâ geliştiricisine dönüşebilirsiniz.

İster bir öğrenci, ister bir araştırmacı, ister bir endüstri çalışanı olun; bu kitap, derin öğrenmeye sağlam bir temel atmak ve gerçek dünyaya yönelik çözümler geliştirmek isteyen herkes için yazıldı. Umarım ilham verir, cesaretlendirir ve yol gösterir.

Sevgiyle, emekle ve umutla yazıldı.

Doç. Dr. Yılmaz KAYA

TEŞEKKÜR

Bu kitabın yazılmasında her türlü anlayışı gösteren sevgili eşime, çocuklarıma, beni bu günlere getiren annem ve babama gösterdikleri her türlü sabır, anlayış ve destekten dolayı sonsuz teşekkür ederim.

Ayrıca bu kitabın hazırlanmasında desteklerini eksik etmeyen Nobel Yayıncılık Eğitim çalışanlarına çok teşekkür ederim.

Doç.Dr. Yılmaz KAYA
Batman Üniversitesi

İÇİNDEKİLER

BÖLÜM 1	Derin Öğrenmenin Temelleri ve PyTorch'a Giriş	1
1.1.	Yapay Zekâ (AI), Makine Öğrenmesi (ML) ve Derin Öğrenme (DL) Nedir?	1
1.2.	AI → ML → DL Arasındaki Hiyerarşik İlişki	3
1.3.	Veriye Dayalı Öğrenme Türleri: Denetimli, Denetimsiz, Yarı Denetimli ve Pekiştirmeli Öğrenme	4
1.4.	Makine Öğrenmesinde Model, Hipotez ve Fonksiyon Kavramları	6
1.5.	Parametre ve Hiperparametre Arasındaki Fark: Kavramsal ve İşlevsel Bir Ayrım	7
1.6.	Bias, Variance ve Generalization Nedir?	9
1.7.	Aşırı Öğrenme (Overfitting), Yetersiz Öğrenme (Underfitting)	10
1.8.	Derin Öğrenmeye Neden İhtiyaç Duyuldu?	13
1.9.	Gerçek Dünya Uygulama Alanları: Görüntü, Metin, Ses, Sağlık, Finans	14
BÖLÜM 2	Sinir Ağlarının Yapısı ve Bileşenleri	19
2.1.	Yapay nöron (Perceptron) ve çok katmanlı algılayıcı (MLP)	19
2.2.	İleri Beslemeli Ağlar (Feedforward Networks)	22
2.3.	Katman Türleri: Dense, Dropout, BatchNorm, Aktivasyon	24
2.4.	Aktivasyon Fonksiyonları: Sigmoid, Tanh, Relu, Leakyrelu, GELU	25
2.5.	Hesaplama Grafiği Ve Otomatik Türev	26
2.6.	Parametre Güncelleme: Gradient Descent ve Türevleri (SGD, Adam, RMSprop)	28
2.7.	Kayıp (Loss) ve Maliyet (Cost) Fonksiyonları	30
2.8.	PyTorch Mimarisi: Modül, Katman, Tensör, Optimizer, Loss Function	31
BÖLÜM 3	Tensörler ve Pytorch Temelleri	35
3.1.	Tensor Nedir? Vektör, Matris ve Yüksek Boyutlu Tensör Farkları	36
3.2.	PyTorch tensör işlemleri	37
3.3.	GPU Kullanımı ve CUDA Mimarisi	41
3.4.	requires_grad, backward() ve .grad: Otomatik Türev Mekanizması (Autograd)	43
3.5.	PyTorch ile NumPy Arasındaki Farklar	45
3.5.	PyTorch'ta Random Seed Kontrolü ve Tekrar Üretilbilirlik (Reproducibility)	47
3.7.	Basit Örnek: Tensör Tabanlı Lineer Regresyon (PyTorch)	48

BÖLÜM 4	PyTorch ile Eğitim Süreci ve Model Değerlendirme	53
4.1.	Eğitim (Training) Ve Test (Inference) Aşamaları	53
4.2.	Eğitim Döngüsü Nasıl Kurulur?	55
4.3.	nn.Module ve forward() Fonksiyonu Nedir?	56
4.4.	torch.optim ile Optimizer Kullanımı	58
4.5.	Batch, Mini-Batch ve Epoch Kavramları	60
4.6.	Öğrenme Oranı (Learning Rate) Seçimi ve Planlaması	63
4.7.	model.train() ve model.eval() Modları	65
4.8.	Early Stopping ve Model Checkpointing: Aşırı Öğrenmeye Karşı Stratejiler	68
BÖLÜM 5	Veri Hazırlama ve PyTorch Dataset API	71
5.1.	PyTorch'ta torch.utils.data.Dataset ve DataLoader Sınıfları	71
5.2.	Görüntü, Metin ve Zaman Serisi Verilerinin İşlenmesi	73
5.3.	torchvision.transforms ve Custom Transform Yazımı	76
5.4.	Data Augmentation Teknikleri: Random Crop, Flip, Noise, Blur	79
5.5.	Dönüştürme İşlemleri: ToTensor, Normalize ve Özel Ön İşleyiciler	81
5.6.	Batch Shuffle, num_workers ve Bellek Yönetimi	83
5.7.	Custom Dataset Sınıfı Oluşturma	84
BÖLÜM 6	Görüntü Tanıma için CNN (Convolutional Neural Networks)	87
6.1.	Konvolüsyon işlemi nedir?	87
6.2.	Filtreler, Kernel, Padding, Stride Kavramları	89
6.3.	Pooling ve Pooling Yöntemleri (Max, Average, Global Pooling)	90
6.4.	Softmax ve Flatten Katmanları	92
6.5.	Sınıflandırma Metrikleri	92
6.6.	CNN Mimarisi ve bir Sınıflandırma Örneği	94
6.7.	Feature Map Nedir?	98
6.8.	nn.Sequential ile Model Yazımı	101
6.9.	GPU ile Eğitim Örneği – Derin Öğrenmede Donanımın Gücünü Kullanmak	103
BÖLÜM 7	İleri Seviye CNN Yapıları ve Transfer Öğrenme	107
7.1.	VGG, ResNet, DenseNet, EfficientNet Transfer Öğrenme Metotları	107
7.2.	Residual Connection ve Bottleneck Yapısı	116
7.3.	Pre-trained Model Kullanımı ve Fine-Tuning	120
7.4.	Feature Extractor Olarak CNN Kullanımı	122
7.5.	Transfer Öğrenme ile Küçük Veri Setlerinde Başarı Artırma	127
BÖLÜM 8	Düzenleme ve Aşırı Öğrenmeyi Önleme Teknikleri	129
8.1.	Dropout ve SpatialDropout	129

8.2.	L1 ve L2 Düzenleme (Weight Decay).....	131
8.3.	Batch Normalization	133
8.4.	Early Stopping ve Learning Rate Scheduler.....	135
8.5.	Model Karmaşıklığı Kontrolü (Parametre Sayısı)	138
BÖLÜM 9	RNN, LSTM, TCN ve GRU	145
9.1.	Zaman Serisi Ve Doğal Dil İşleme Temelleri.....	145
9.2.	nn.RNN, nn.LSTM ve nn.GRU Yapıları ve Farkları	147
9.3.	Gizli Durum (Hidden State) ve Hücre Durumu (Cell State).....	151
9.4.	Sequence Veri Hazırlama (Padding, Pack_Padded)	154
9.5.	Uygulama: Metin Üretimi, Duygu Analizi, Zaman Serisi Tahmini... 156	
BÖLÜM 10	Transformers ve Attention Mekanizmaları	171
10.1.	Attention Nedir, Neden Kullanılır?	171
10.2.	Self-Attention ve Multi-Head Attention	173
10.3.	Transformer Mimarisinin Katmanları	176
10.4.	BERT, GPT Gibi Modellerin Temelleri.....	182
10.5.	Huggingface Transformers Ile Hazır Model Kullanımı.....	185
BÖLÜM 11	Doğal Dil İşleme (NLP) Uygulamaları.....	189
11.1.	Tokenization Ve Embedding Kavramı.....	190
11.2.	Word2Vec, Glove, Fasttext Kullanımı.....	191
11.3.	Embedding Katmanı Ve Sequence Modelleme	194
11.4.	Text Classification ve Sentiment Analysis.....	197
11.5.	Sequence-To-Sequence Modeli ile Çeviri Örneği	200
BÖLÜM 12	Otomatik Kodlayıcılar (Autoencoders).....	205
12.1.	Encoder-Decoder Mantığı	206
12.2.	Lineer ve Konvolüsyonel Autoencoder	210
12.3.	Gürültülü Autoencoder (Denoising AE)	215
12.4.	Latent Space Görselleştirme.....	218
12.5.	VAE (Variational Autoencoder).....	221
BÖLÜM 13	Generative Adversarial Networks (GAN)	229
13.1.	GAN Mimarisi: Generator vs Discriminator	230
13.2.	Gerçek vs Sahte Örnek Üretimi	231
13.3.	GAN Eğitim Dengesizlikleri ve Çözüm Yolları.....	234
13.4.	Basit bir GAN ile Çiçek Üretimi.....	235
13.5.	DCGAN, CycleGAN, StyleGAN Modeller	238
BÖLÜM 14	Model Kaydetme, Yükleme ve Servis Etme	241
14.1.	torch.save() ve torch.load() Kullanımı	242
14.2.	state_dict Mantığı	245
14.3.	Eğitim Sonrası Modeli Inference (Çıkarım) İçin Hazırlama.....	246

14.4. Torchscript ve ONNX İle Model Taşınabilirliği	252
14.5. Model Versiyonlama Ve Üretim Ortamı İçin En İyi Uygulamalar	254
BÖLÜM 15 Performans İzleme ve Model Takibi	257
15.1. torch.utils.tensorboard, matplotlib	257
15.2. Eğitim Eğrileri: Loss, Accuracy, Precision-Recall	262
15.3. AUC-ROC ve Confusion Matrix Görselleştirmeleri	267
15.4. Model Loglama ve Sürüm Takibi	272
BÖLÜM 16 Hiperparametre Optimizasyonu	275
16.1. Öğrenme Oranı, Dropout Oranı, Batch Size	276
16.2. Grid Search, Random Search, Bayesian Optimization	279
16.3. Optuna ve Ray Tune Tanıtımı	283
16.4. Otomatik Deney Yönetimi	285
BÖLÜM 17 PyTorch Lightning ile Temiz Eğitim	291
17.1. PyTorch Lightning nedir?	292
17.2. LightningModule ve Trainer Sınıfı	296
17.3. Kodun Sadeleşmesi, Yeniden Kullanılabilirlik	299
17.4. Callback Yapıları İle Erken Durdurma ve Model Kaydı	302
17.5. Dağıtık Eğitim Desteği	306
BÖLÜM 18 Çoklu GPU ve Dağıtık Eğitim	309
18.1. DataParallel vs DistributedDataParallel	310
18.2. GPU Bellek Yönetimi	312
18.3. Batch Büyüklüğü Optimizasyonu	316
18.4. Eğitim Hızlandırma Teknikleri (Mixed Precision, AMP)	318
BÖLÜM 19 Gerçek Dünya Projeleri ve Endüstri Uygulamaları	321
19.1. Görüntü sınıflandırma	321
19.2. Sinyal Sınıflandırma (EKG)	326
19.3. NLP (metin analizleri, spam tespiti)	331
19.4. Anomali tespiti (finans, IoT, siber güvenlik)	335
BÖLÜM 20 Bitirme Projesi – Sıfırdan Model Geliştirme Süreci	341
20.1. Proje Fikri Oluşturma	342
20.2. Veri Keşfi ve Ön İşleme	343
20.4. Performans Değerlendirme	345
20.5. Sunum ve Model Deployment	346
Kaynaklar	346

KAYNAKLAR

1. Goldberg, Y. (2017). *Neural network methods in natural language processing*. Morgan & Claypool Publishers.
2. Rothman, D. (2022). *Transformers for Natural Language Processing: Build, train, and fine-tune deep neural network architectures for NLP with Python, Hugging Face, and OpenAI's GPT-3, ChatGPT, and GPT-4*. Packt Publishing Ltd.
3. Manning, C., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press.
4. Jain, S. M. (2022). Hugging face. In *Introduction to transformers for NLP: With the hugging face library and models to solve problems* (pp. 51-67). Berkeley, CA: Apress.
5. Kelleher, J. D. (2019). *Deep learning*. MIT press.
6. Stevens, E., Antiga, L., & Viehmann, T. (2020). *Deep Learning with PyTorch: Build, train, and tune neural networks using Python tools*. Manning.
7. Liu, M. (2024). *Learn Generative AI with PyTorch*. Simon and Schuster.
8. Chollet, F., & Chollet, F. (2021). *Deep learning with Python*. Simon and Schuster.