

KOCAELİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

DERİN ÖĞRENME MODELLERİ İLE SOSYAL MEDYA  
ÜZERİNDE DUYGU ANALİZİ

HAZAL GİZEM DÖNMEZ

KOCAELİ 2024

**KOCAELİ ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**YÜKSEK LİSANS TEZİ**

**DERİN ÖĞRENME MODELLERİ İLE SOSYAL MEDYA**  
**ÜZERİNDE DUYGU ANALİZİ**

**HAZAL GİZEM DÖNMEZ**

**Prof. Dr. Yaşar BECEKLİ**

**Danışman, Kocaeli Üniversitesi**

.....

**Doç. Dr. Orhan AKBULUT**

**Jüri Üyesi, Kocaeli Üniversitesi**

.....

**Prof. Dr. Nevcihan DURU**

**Jüri Üyesi, Kocaeli Sağlık ve Teknoloji Üniversitesi**

.....

**Tezin Savunulduğu Tarih: 29.01.2024**

## ETİK BEYAN VE ARAŞTIRMA FONU DESTEĞİ

Kocaeli Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- Bu tezin bana ait, özgün bir çalışma olduğunu,
- Çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı,
- Bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi,
- Bu çalışmanın Kocaeli Üniversitesi'nin abone olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun olduğunu,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı,

beyan ederim.

Bu tez çalışmasının herhangi bir aşaması hiçbir kurum/kuruluş tarafından maddi/alt yapı desteği ile desteklenmemiştir.

Bu tez çalışması kapsamında üretilen veri ve bilgiler ..... tarafından ..... no'lu proje kapsamında maddi/alt yapı desteği alınarak gerçekleştirilmiştir.

Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Hazal Gizem DÖNMEZ

## YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI

Fen Bilimleri Enstitüsü tarafından onaylanan lisansüstü tezimin tamamını veya herhangi bir kısmını, basılı ve elektronik formatta arşivleme ve aşağıda belirtilen koşullarla kullanıma açma izninin Kocaeli Üniversitesi'ne verdiğimi beyan ederim. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet hakları bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki makale, kitap, tebliğ, lisans, patent gibi çalışmalarda kullanımı, danışmanımın isim hakkı saklı kalmak koşuluyla ve her iki tarafın bilgisi dâhilinde bana ait olacaktır.

Tezin kendi özgün çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim kurulu tarafından yayınlanan **“Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge”** kapsamında tezim aşağıda belirtilen koşullar haricinde YÖK Ulusal Tez Merkezi/ Kocaeli Üniversitesi Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- Enstitü yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihinden itibaren 2 yıl ertelenmiştir.
- Enstitü yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihinden itibaren 6 ay ertelenmiştir.
- Tezim ile ilgili gizlilik kararı verilmemiştir.

Hazal Gizem DÖNMEZ

## ÖNSÖZ VE TEŞEKKÜR

Bu tez çalışmasında, derin öğrenme ve transformatör modelleri ile duygu analizi çalışması yapılmıştır.

Tez çalışmamın ilerleyişinde bana yol gösteren, her aşamasında destekleyen ve anlayan değerli danışman hocam Prof. Dr. Yaşar BECERİKLİ'ye teşekkür ederim. Akademik çalışmalarım için beni teşvik eden Dr. Öğr. Üyesi Elif BOZKAYA'ya ve Recep Doğa SİYİLİ'ye teşekkür ederim.

Yüksek lisans çalışmam boyunca bana güç veren, destekleyen ve her zaman yanımda olan eşim Mesut Can DÖNMEZ ve abim Sercan Ozan NUR başta olmak üzere aileme teşekkürlerimi sunarım.

Ocak – 2024

Hazal Gizem DÖNMEZ

## İÇİNDEKİLER

ETİK BEYAN VE ARAŞTIRMA FONU DESTEĞİ.....	i
YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI .....	ii
ÖNSÖZ VE TEŞEKKÜR.....	iii
İÇİNDEKİLER.....	iv
ŞEKİLLER DİZİNİ .....	vi
TABLolar DİZİNİ.....	viii
SİMGELER VE KISALTMALAR DİZİNİ .....	ix
ÖZET .....	xi
ABSTRACT .....	xii
1. GİRİŞ.....	1
1.1. Tezin Kapsamı .....	2
1.2. Literatür Taraması.....	3
1.3. Tezin Yapısı .....	7
2. DUYGU ANALİZİNDE TEMEL KAVRAMLAR .....	8
2.1. Duygu Analizi .....	8
2.2. Sosyal Medya ve Twitter (X).....	10
2.3. Python Programlama ve Çerçevesi.....	11
2.3.1. Python Programlama.....	11
2.3.2. Tensorflow Çerçevesi .....	13
2.3.3. Keras Çerçevesi .....	13
2.3.4. Transformer Çerçevesi.....	14
2.4. Doğal Dil İşleme .....	15
2.5. Makine Öğrenmesi.....	17
2.6. Derin Öğrenme Modelleri.....	19
2.6.1. RNN (Recurrent Neural Network, Özyinelemeli Sinir Ağı).....	21
2.6.2. LSTM (Long Short-Term Memory, Uzun-Kısa Vadeli Bellek).....	22
2.6.3. ConNN (Convolutional Neural Networks, Evrişimli Sinir Ağı) .....	25
2.7. Dönüştürücü (Transformer) Modeller.....	27
2.7.1. BERT .....	29
2.8. Optimizasyon Algoritmaları .....	32
2.8.1. Gradyan İnişi.....	32
2.8.2. Stokastik Gradyan İnişi.....	32
2.8.3. Momentum.....	32
2.8.4. Adagrad.....	33
2.8.5. RMSProp .....	34
2.8.6. Adam.....	34
2.9. Aktivasyon Fonksiyonları .....	35
2.9.1. Doğrusal Fonksiyon .....	35
2.9.2. Sigmoid Fonksiyonu .....	36
2.9.3. Tanh Fonksiyonu .....	37
2.9.4. ReLu Fonksiyonu.....	38
2.9.5. Softmax Fonksiyonu .....	38
2.10. Değerlendirme Metrikleri .....	39
2.10.1. Doğruluk .....	42
2.10.2. Dengeli Doğruluk .....	43
2.11. Kayıp Fonksiyonları.....	43

2.11.1. İkili Çapraz Kayıp Fonksiyonu.....	43
2.11.2. Çapraz Kategorik Kayıp Fonksiyonu .....	44
3. MATERYAL VE YÖNTEM .....	45
3.1. Veri Kümesi .....	45
3.2. LSTM Modeli .....	47
3.3. ConNN -LSTM Hibrit Modeli .....	48
3.4. BERT Modeli.....	51
3.5. Model Derlemeleri .....	51
4. BULGULAR VE TARTIŞMA .....	52
4.1. Türkçe Tweet Veri Kümesinde Bulgular .....	52
4.2. Türkçe Mağaza Yorumları Veri Kümesinde Bulgular.....	56
4.3. Tartışma .....	60
5. SONUÇ VE ÖNERİLER .....	63
KAYNAKLAR.....	65
KİŞİSEL YAYIN VE ESERLER.....	71
ÖZGEÇMİŞ.....	72

## ŞEKİLLER DİZİNİ

Şekil 2.1.	Tweet Örneği .....	11
Şekil 2.2.	Standart Derin Öğrenme Çerçeve Yapısı.....	12
Şekil 2.3.	Dönüştürücü Çerçevesi Yapısı.....	14
Şekil 2.4.	Makine Öğrenmesi Yapısı .....	18
Şekil 2.5.	RNN Mimarisi .....	21
Şekil 2.6.	RNN Hücre Yapısı.....	22
Şekil 2.7.	LSTM Hücre Yapısı.....	23
Şekil 2.8.	ConNN Mimarisi .....	25
Şekil 2.9.	Dikkat Mekanizması.....	27
Şekil 2.10.	BERT Modeli .....	29
Şekil 2.11.	Maskeli Dil Modelleme .....	30
Şekil 2.12.	Sonraki Cümle Tahmini.....	31
Şekil 2.13.	Momentum Uygulanmamış Stokastik Gradyan İnişi .....	33
Şekil 2.14.	Momentum Uygulanmış Stokastik Gradyan İnişi .....	33
Şekil 2.15.	Doğrusal Fonksiyon Grafiği.....	35
Şekil 2.16.	Sigmoid Fonksiyonu Grafiği.....	36
Şekil 2.17.	Tanh Fonksiyonu Grafiği.....	37
Şekil 2.18.	ReLu Fonksiyonu Grafiği.....	38
Şekil 3.1.	Türkçe Tweet Veri Kümesi .....	45
Şekil 3.2.	Türkçe Mağaza Yorumları Veri Kümesi.....	46
Şekil 3.3.	Türkçe Mağaza Yorum Veri Kümesi .....	46
Şekil 3.4.	Kelime Sözlüğü Oluşturulması .....	47
Şekil 3.5.	Metinlerin Sayı Dizisine Dönüştürülmesi .....	47
Şekil 3.6.	LSTM Modeli Katmanları.....	47
Şekil 3.7.	ConNN -LSTM Hibrit Modeli Katmanları .....	49
Şekil 3.8.	Konvolüsyon Gösterimi .....	49
Şekil 3.9.	Maxpooling Gösterimi .....	50
Şekil 4.1.	Türkçe Tweet Veri Kümesi LSTM Yaklaşımları .....	52
Şekil 4.2.	Türkçe Tweet Veri Kümesinde LSTM Doğruluk Grafiği.....	52
Şekil 4.3.	Türkçe Tweet Veri Kümesinde LSTM Kayıp Grafiği .....	53
Şekil 4.4.	Türkçe Tweet Veri Kümesi ConNN-LSTM Hibrit Modelinin Yaklaşımları .....	53
Şekil 4.5.	Türkçe Tweet Veri Kümesi ConNN-LSTM Modelinin Doğruluk Grafiği.....	53
Şekil 4.6.	Türkçe Tweet Veri Kümesinde ConNN-LSTM Hibrit Modelinin Kayıp Grafiği.....	54
Şekil 4.7.	Türkçe Tweet Veri Kümesi BERT Modeli Yaklaşımları .....	54
Şekil 4.8.	Türkçe Tweet Veri Kümesinde BERT Modeli Doğruluk Grafiği.....	55
Şekil 4.9.	Türkçe Tweet Veri Kümesinde BERT Modeli Kayıp Grafiği .....	55
Şekil 4.10.	Türkçe Mağaza Yorumları Veri Kümesi LSTM Modeli Yaklaşımları .....	56
Şekil 4.11.	Türkçe Mağaza Yorumları Veri Kümesi LSTM Modeli Doğruluk Grafiği.....	56
Şekil 4.12.	Türkçe Mağaza Yorumları Veri Kümesi LSTM Modeli Kayıp Grafiği.....	57

Şekil 4.13.	Türkçe Mağaza Yorumları Veri Kümesi ConNN-LSTM Hibrit Modeli Yaklaşımları .....	57
Şekil 4.14.	Türkçe Mağaza Yorumları Veri Kümesi ConNN-LSTM Hibrit Modeli Doğruluk Grafiği.....	58
Şekil 4.15.	Türkçe Mağaza Yorumları Veri Kümesi ConNN-LSTM Hibrit Modeli Kayıp Grafiği .....	58
Şekil 4.16.	Türkçe Mağaza Yorumları Veri Kümesi Bert Modeli Yaklaşımları .....	59
Şekil 4.17.	Türkçe Mağaza Yorumları Veri Kümesi Bert Modeli Doğruluk Grafiği.....	59
Şekil 4.18.	Türkçe Mağaza Yorumları Veri Kümesi Bert Modeli Kayıp Grafiği.....	59



## TABLolar DİZİNİ

Tablo 2.1.	Karmaşıklık Matrisi.....	39
Tablo 4.1.	Türkçe Tweet Veri Kümesinde Model Performansları .....	56
Tablo 4.2.	Türkçe Mağaza Yorumları Veri Kümesinde Model Performansları .....	60



## SİMGELER VE KISALTMALAR DİZİNİ

$a$	: Gizli Durum
$b$	: Bias
$C$	: Hücre Durumu
$d$	: Anahtar Boyutu
$D$	: Doğru
$e$	: Euler sayısı
$E$	: Gömme Vektörü
$f$	: Unutma Geçidi
$g$	: Aktivasyon Fonksiyonu
$h$	: Hücre Bilgisi
$i$	: Giriş Geçidi
$K$	: Anahtar
$L$	: Kayıp
$m$	: Momentum
$o$	: Çıktı Geçidi
$p$	: Tahmini Değer
$P$	: Olasılık
$r$	: Gerçek Değer
$s$	: Özellik Haritası
$t$	: Zaman
$v$	: Gradyan Karelerinin Üssel Olarak Ağırlıklandırılmış Ortalamaları
$V$	: Öz-dikkat Mekanizması Değerleri
$y$	: Çıktı
$Y$	: Yanlış
$\beta$	: Momentum Hiperparametresi
$\tilde{C}$	: Yeni Hücre Durumu
$\epsilon$	: Stabilizasyon Terimi
$w$	: Çekirdek
$W$	: Ağırlık
$Q$	: Sorgu
$x$	: Girdi
$\hat{y}_i$	: Çıktı Vektörü
$\sigma$	: Sigmoid

### Kısaltmalar

ANN-YSA	: Artificial Neural Networks (Yapay Sinir Ağları)
API	: Application Programming Interface (Uygulama Programlama Arayüzü)
BERT	: Bidirectional Encoder Representations from Transformers Dönüştürücülerden Çift Yönlü Kodlayıcı Temsilleri)
Bi-LSTM	: Bidirectional Long Short Term Memory (Çift Yönlü Uzun Kısa Süreli Bellek)
ConNN	: Convolutional Neural Networks (Evrşimli Sinir Ağı)

CNTK	: Microsoft Cognitive Toolkit (Microsoft Bilişsel Araç Takımı)
CPU-MİB	: Central Processing Unit (Merkezi İşlemci Birimi)
DBN	: Deep Belief Networks (Derin İnşa Edilebilir Ağlar)
DTR	: Decision Tree Regression (Karar Ağacı Regresyonu)
DQN	: Deep Q-Networks (Derin Q-Ağları)
GAN	: Generative Adversarial Networks (Çekişmeli Üretici Ağ)
GPT	: Generative Pre-trained Transformer (Üretken Ön İşlemeli Dönüştürücü)
GPU-GİB	: Graphics Processing Unit (Grafik İşlemci Birimi)
GRU	: Gated Recurrent Unit (Kapılı Tekrarlayan Birim)
KNN	: K-Nearest Neighbors (K-En Yakın Komşu)
LR	: Logistic Regression (Lojistik Regresyon)
LSTM	: Long Short-Term Memory (Uzun-Kısa Vadeli Bellek)
MLM	: Masked Language Modeling (Maskeli Dil Modelleme)
MLP	: Multilayer Perceptron (Çok Katmanlı İleri Beslemeli Yapay Sinir Ağı)
NB	: Naive Bayes (Saf Bayes)
NLP	: Natural Language Processing (Doğal Dil İşleme)
NSP	: Next Sentence Prediction (Sonraki Cümle Tahmini)
RFR	: Random Forest Regression (Rassal Orman Regresyonu)
RNN	: Recurrent Neural Network (Özyinelemeli Sinir Ağı)
RoBERTa	: Robustly Optimized BERT Approach (Sağlam Optimize Edilmiş BERT Yaklaşımı)
SES	: Simple Exponential Smoothing (Basit Üstel Düzeltme)
SMA	: Simple Moving Average (Basit Hareketli Ortalama)
SVM	: Support Vector Machine (Destek Vektör Makinesi)
SVR	: Support Vector Regression (Destek Vektör Regresyon)
T5	: Text-To-Text Transfer Transformer (Metinden Metine Aktarma Dönüştürücüsü)
ULMFIT	: Universal Language Model Fine-tuning (Evrensel Dil Modeli İnce Ayar)

# DERİN ÖĞRENME MODELLERİ İLE SOSYAL MEDYA ÜZERİNDE DUYGU ANALİZİ

## ÖZET

Duygu analizi, metinlerdeki duyguları belirleyen doğal dil işleme teknolojilerinin bir uygulamasıdır. Hizmet kalitesinin değerlendirilmesi için müşteri memnuniyetinin ölçülmesi, müşterilerin ürün tanıtımına nasıl tepki verdiklerinin saptanması, halkın seçim kampanyalarına nasıl tepki verdiğinin belirlenmesi gibi birçok konuda ihtiyaç duyulduğu için önemlidir. Sosyal medyanın giderek yaygınlaşması, finanstan tıbbı, medyadan e-ticarete, eğitimden siyasete kadar pek çok alanda kullanımını beraberinde getirmiştir.

Tez çalışması kapsamında, Türkçe metinler üzerinde, BERT (Bidirectional Encoder Representations from Transformers, Dönüştürücülerden Çift Yönlü Kodlayıcı Temsilleri), LSTM (Long Short-Term Memory, Uzun-Kısa Vadeli Bellek) ve ConNN (Convolutional Neural Networks, Evrişimli Sinir Ağı)-LSTM Hibrit modelleri ile duygu analizi yapılarak modellerin performansları karşılaştırılmıştır. Mutlu, kızgın, korku, şaşkın ve üzgün ile olumlu, olumsuz ve tarafsız olarak etiketli iki ayrı veri kümesi Kaggle aracılığıyla elde edilmiştir.

ConNN -LSTM Hibrit modelinde, metin verilerini sayısal vektörlere dönüştürmek için bir gömme, verilerin analizi ile özellik çıkarma için konvolüsyon ve havuzlama, zaman bağımlılıklarını modellemek için LSTM son olarak sınıflandırma ve çıkış katmanı olan yoğun katmanı yer almaktadır. LSTM modelinde ise gömme, LSTM ve yoğun katmanı kullanılmıştır. BERT modelinde önceden eğitilmiş bir dil modeli kullanılarak metinlerin dizi vektörlerine dönüştürülmesini ve dil özelliklerini öğrenmesini sağlayan kodlayıcı ve bu vektörleri kullanarak metinlerin eğitilmesini sağlayan çözücü kullanılmıştır.

Üç model veri setleri üzerinde eğitilmiş ve uygun optimizasyon algoritması, kayıp fonksiyonu ile doğruluk metriği kullanılmıştır. Eğitim ve test verilerinin beklenen sonuçlarla kıyaslanması ile modellerin performanslarına ulaşılmış ve performansların karşılaştırılması yapılmıştır.

Bu tez çalışması, Türkçe veri kümesi üzerinde gerçekleştirilen duygu analiziyle farklı öğrenme modellerinin performanslarının karşılaştırılması sayesinde duygu analizi alanına önemli ölçüde katkı sağlamıştır.

**Anahtar Kelimeler:** BERT, ConNN, Derin Öğrenme, Duygu Analizi, LSTM.

# SENTIMENT ANALYSIS ON SOCIAL MEDIA USING DEEP LEARNING MODELS

## ABSTRACT

Sentiment analysis is a technique of natural language processing that identifies emotions in texts. It is important because with the widespread use of social media, its use in many areas such as finance, medicine, e-commerce media, education and politics.

Within the scope of this thesis, sentiment analysis was performed on Turkish texts using BERT (Bidirectional Encoder Representations from Transformers), LSTM (Long Short-Term Memory) and ConNN (Convolutional Neural Networks)-LSTM Hybrid models and the performances of the models were compared. Two separate datasets labeled as happy, angry, fearful, confused, sad and positive, negative, neutral were obtained through Kaggle.

In the ConNN -LSTM model, there is an embedding to convert text into digital vectors, convolution and pooling for feature extraction by analyzing the data, LSTM to model time dependencies, the dense layer for classification and output. LSTM model, the embedding, LSTM and dense layer are used. The BERT model uses a pre-trained language model, an encoder to convert texts into sequence vectors and learn language features, and a decoder to train texts.

The three models were trained on the datasets and the appropriate optimization algorithm, loss function and accuracy metric were used. The performances of the models were obtained by comparing the training and test data with the expected results and the were compared.

This thesis has made a significant contribution to the field of sentiment analysis by This thesis has significantly contributed to the field of this by comparing the performance of different learning models on Turkish datasets.

**Keywords:** BERT, ConNN, Deep Learning, Sentiment Analysis, LSTM.

## 1. GİRİŞ

Günümüzde sosyal medya platformları arasında etkileşim ve topluluk oluşturma konusunda Twitter'ın önemi büyüktür. Twitter, olayların gerçek zamanlı takibini sağlama konusunda büyük bir etkiye sahiptir. Anlık haberler ve popüler konular hakkında güncel bilgilere kolayca ulaşmak mümkündür. Kullanıcılar, dünyadaki gelişmeleri hızlı bir şekilde öğrenirken finans, tıp, medya, e-ticaret, eğitim, politika, teknoloji, sanat, spor ve daha birçok konuda tartışabilmekte, görüşlerini paylaşabilmekte ve diğer kullanıcılarla etkileşimde bulunabilmektedir.

Twitter, duygu analizi açısından son derece etkili bir platformdur. Kullanıcılar, Tweetler aracılığıyla duygusal tepkilerini ifade etme eğiliminde olduğu için büyük miktarda veri sağlamaktadır. Bu veriler, duygu analizi araçları ve modelleri ile incelenerek, toplumun genel duygusal durumunu anlama amacıyla değerlendirilebilmektedir.

Twitter, kullanıcılara gerçek zamanlı olarak duygusal tepkilerini paylaşma imkânı tanıdığından, anlık ve güncel duygu durumunu yansıtabilme potansiyeline sahiptir. Örneğin, bir spor etkinliği sırasında maçla ilgili Tweetlerdeki coşku ve heyecan, kullanıcıların mutluluğunu ve sevincini yansıtmaktadır. Benzer şekilde, toplumsal olaylar, siyasi tartışmalar veya acil durumlar gibi durumlar da Twitter üzerinde geniş çaplı duygusal tepkiler tetiklenebilmektedir.

Teknolojinin ilerlemesi ile internet alışverişleri yaygınlaşmıştır. Ürün fiyatlarının karşılaştırılması, müşteri memnuniyeti gibi pek çok faktör alışverişlerde etkili hale gelmiştir.

Duygu analizi, metinlerdeki dilbilimsel ipuçlarını ve duygusal ifadeleri algılamak için doğal dil işleme ve makine öğrenimi tekniklerini kullanmaktadır. Twitter ve alışveriş siteleri üzerindeki büyük veri kümeleri, bu tekniklerin uygulanması için zengin bir kaynak sağlamaktadır. Duygu analizi araçları, metinlerdeki çeşitli etiketler ile duygusal ifadeler tanımlanarak genel bir duygu durumu analizi yapılabilmektedir.

Bu analiz, ürün veya hizmetlerin algılanan memnuniyetini ölçmek, kampanyaların etkisini değerlendirmek veya olayların toplumdaki duygusal tepkisini anlamak gibi birçok alanda kullanılabilir.

Duygu analizi çalışmaları günümüzde popülerlik kazanmış olsa da mevcut modeller ve çerçeveler İngilizcenin dünyada konuşulan ortak dil olması sebebi ile büyük ölçüde İngilizce verilere odaklanmış durumdadır. Türkçe duygu analizi halen geliştirilmeye ihtiyaç duyulan bir alan olduğu için bu çalışmada, Türkçe duygu analizi literatürünün incelenmesi amaçlanmaktadır. Bu çalışmada “Türkçe metinler üzerinde hangi model daha başarılı performans göstermektedir?” sorusuna cevap aranmaktadır. Bu soruya yanıt bulabilmek için metin sınıflandırma için bilinen modeller ile Türkçe metinler üzerinde duygu analizi gerçekleştirilerek performanslarını karşılaştırmak ve en üst düzey performansa ulaşan modeli bularak hem akademik alanda çalışacak olan araştırmacılara rehberlik etmesi hem de ileride yapılacak araştırmalar için bir temel oluşturulması planlanmaktadır.

### **1.1. Tezin Kapsamı**

Bu tez çalışmasında, Türkçe metinler üzerinde duygu analizi yapmak için iki ayrı veri kümesi üzerinde LSTM, BERT ve ConNN<sup>1</sup>-LSTM hibrit modelleri kullanılmış ve performansları karşılaştırılmıştır. Bu modeller, dil işleme ve metin sınıflandırma alanında yaygın olarak kullanılan ve başarılı sonuçlar veren yöntemlerdir.

Kullanılan veri kümeleri, Kaggle aracılığıyla elde edilmiştir (Güven, 2021; Bilen, 2020). Mutlu, kızgın, korku, şaşkın ve üzgün duygularını içeren ilk veri kümesi, tweet örneklerini içermektedir. Olumlu, olumsuz ve tarafsız duygu etiketlerini içeren ikinci veri kümesi mağaza yorumlarını içermektedir. Veri kümeleri, duygusal etiketlemelerin doğru ve güvenilir bir şekilde yapılması için incelenmiştir. Mağaza yorumlarını içeren veri kümesinde tarafsız olan etiketlerin güvenilir sonuç vermediği tespit edildiği için veri kümesinden çıkarılarak veri olumlu ve olumsuz olarak değerlendirilmiştir.

---

<sup>1</sup> Bu tez çalışmasında Evrişimli Sinir Ağı (Convolutional Neural Networks) kısaltması ConNN olarak verilmiştir. CNN veya CoNN olarak kısaltılmamıştır. CNN uzun süredir Hücreli Sinir Ağları (Cellular Neural Network) tanımının kısaltılması olarak ve CoNN uzun süredir İşbirlikçi Sinir Ağları (Cooperative Neural Networks) tanımının kısaltılması olarak literatürde yer almaktadır.

## 1.2. Literatür Taraması

Güven ve arkadaşları tarafından 2020 yılında yapılan duygu analizi çalışmasında kızgın, korku, mutlu, üzgün ve şaşkın olmak üzere 5 farklı duyguya ait 4000 tweetten oluşan veri kümesi ve kızgın, korku, mutlu olmak üzere 3 farklı duyguya ait 2400 tweetten oluşan veri kümesi üzerinde LDA (Latent Dirichlet Allocation, Gizli Dirichlet Ayırımı), LSA (Latent Semantic Analysis, Gizli Anlamsal Analiz) ve PLSA (Probabilistic-Latent Semantic Analysis, Olasılıksal-Gizli Anlamsal Analiz) modellerini karşılaştırmıştır. 5 sınıflı veri kümesinde Klasik LDA'da 60,37%, 2 aşamalı LDA'da 70,5%, 3 aşamalı LDA'da 76,37%, LSA'da 87%, PLSA'da 63,12% sonuçlarını elde etmiştir. 3 sınıflı veri kümesinde ise Klasik LDA'da 65,83%, 2 aşamalı LDA'da 80,83%, 3 aşamalı LDA'da 81,5%, LSA'da 95,4, PLSA'da 81,45% sonuçlarını elde etmiştir. 3 sınıflı veri kümesinde başarı oranının 5 sınıflı veri kümesine göre daha fazla olduğu gözlemlenmiştir (Güven ve diğ., 2020).

Sargın, kullandığım Türkçe mağaza yorumları veri kümesi üzerinde Kaggle platformunda yaptığı duygu analizi çalışmasında n-gram, kelime düzeyi, sayım vektörü ile karakter düzeyi modellerinde LR (Logistic Regression, Lojistik Regresyon), NB (Naive Bayes), RFR (Random Forest Regression, Rassal Orman Regresyonu) ve XGBoost modellerini kullanmıştır (Sargın, 2022). LR modelinde sayım vektöründe 88.13%, kelime düzeyinde 87,86%, n-gram da 77,82%, karakter düzeyinde 86,06% başarı elde etmiştir. NB modelinde sayım vektöründe 89,21%, kelime düzeyinde 89,11%, n-gram da 76,54%, karakter düzeyinde 84,31% başarı elde etmiştir. RFR modelinde sayım vektöründe 85,86%, kelime düzeyinde 85,54%, n-gram da 75,22%, karakter düzeyinde 82,95% başarı elde etmiştir. XGBoost modelinde sayım vektöründe 84,45%, kelime düzeyinde 84,22%, n-gram da 69,81%, karakter düzeyinde 84,69% başarı elde etmiştir.

Dorukhan Afacan tarafından Türkçe mağaza yorumları veri kümesi üzerinde Kaggle platformunda yapılan çalışmada RFR modeli kullanılmıştır (Afacan, 2020). 86,8% başarı oranı elde edilmiştir.

Umut Akter tarafından Türkçe mağaza yorumları veri kümesi üzerinde Kaggle platformunda LR, DTR (Decision Tree Regression, Karar Ağacı Regresyonu), RFR ve NB modelleri kullanılarak sınıflandırma çalışması yapılmıştır (Akter, 2021). Başarı

oranları LR, DTR, RFR ve NB modellerinde sırasıyla 87%, 79%, 85% ve 74% elde edilmiştir.

Görgel ve Kavlak tarafından 2020 yılında, rüzgâr hızı veri kümesi ile rüzgâr enerjisi üretim tahmini üzerine çalışma yapılmıştır. Çalışma kapsamında Sarimax, Prophet, GRU, ConNN, LSTM ve ConNN -LSTM hibrit metotları karşılaştırılmış ve ConNN -LSTM hibrit modelinin daha başarılı olduğu tespit edilmiştir (Görgel & Kavlak, 2020).

Yılmaz ve arkadaşı, Covid-19 pandemisi ile ilgili Twitter veri kümesi ile duygu analizi çalışması yapmıştır. Bunun için, olumlu ve olumsuz düşünceleri kategorilere ayırarak derin öğrenme modeli olan LSTM ile eğitim yapmış ve 97% doğruluk oranına ulaşmışlardır (Yılmaz & Orman, 2021).

To ve arkadaşları, BERT, Bi-LSTM (Bidirectional Long Short Term Memory, Çift Yönlü Uzun Kısa Süreli Bellek), SVM ve NB sınıflandırma modellerini kullanarak aşı karşıtı Tweet'lerin tespiti için çalışma yapmıştır. Doğruluk oranları, BERT modelinde 91,6%, Bi-LSTM modelinde 89,8%, SVM modelinde 92,3% ve NB modelinde 88,8% elde edilmiştir (To ve diğ., 2021).

Küçüktaş, Uysal, Hardalaç ve Biri 2021 yılında, BERT modelini kullanarak covid-19 aşısı ile ilgili Tweetlerden oluşan veri kümesini haber, anlamsızlar, aşı karşıtlığı ve aşı destekleyenler olmak üzere 4 farklı kategoride eğitmiş ve kategorilerde sırasıyla 99%, 91%, 92%, 92% doğruluk elde etmiştir (Küçüktaş ve diğ., 2021).

Öztürk ve Pashaei 2021 yılında, konuşmalardaki duygunun tespiti üzerine çalışma yapmıştır. ConNN -LSTM hibrit modeli kullanılarak başka çalışmalarda yapılan LSTM, ConNN ile diğer modellerde alınan doğruluk oranları ile karşılaştırılmış ve diğer modellere kıyasla daha başarılı sonuç verdiği gözlemlenmiştir (Öztürk & Pashaei, 2021).

Er ve Işık 2021 yılında, LSTM-ConNN hibrit modelini kullanarak diyabet hastalığının tahmini için çalışma yapmıştır. Önerilen modeli karşılaştırmak için literatürde kullanılmış olan pozitif ve negatif olarak etiketli Pima Indians Diabetes veri kümesi kullanılmıştır. Sınıflandırma yöntemlerinden en başarılı olanın 86,45% ile ConNN -LSTM modeli olduğu gözlemlenmiştir (Er & Işık, 2021).

Çakmak ve arkadaşı, ConNN, RNN (Recurrent Neural Network, Özyinelemeli Sinir Ağı), LSTM, GRU (Gated Recurrent Unit, Kapılı Tekrarlayan Birim) modelleri ile protein ikincil yapı tahmini çalışması gerçekleştirmiştir. ConNN modelinde 82,54%, RNN modelinde 82,06%, LSTM modelinde 81,1% ve GRU modelinde 81,48% doğruluk oranlarına ulaşılmıştır (Çakmak & Selvi, 2022).

Aslan 2022 yılında Covid-19 Aşılı ile ilgili Twitter verileri ile duygu analizi çalışması yapmıştır (Aslan, 2022). Pozitif ve negatif olarak etiketli veri kümesi ile Bi-GRU (Bidirectional Gated Recurrent Unit, Çift-yönlü Geçitli Tekrarlayan Ünite) modeli önerilmiş ve önerilen model SVM, LR, RFR modelleri ile karşılaştırılmıştır. Önerilen modelin daha iyi performans gösterdiği gözlemlenmiştir.

Göğüş 2022 yılında, “Ataletsel Navigasyon Sistemlerinde Kestirim İçin Farklı Yöntemlerin Karşılaştırılması” isimli yaptığı çalışmada LSTM, Bi-LSTM ve GRU derin öğrenme modelleri kullanılmıştır. En başarılı sonucu GRU modelinin verdiği gözlemlenmiştir (Göğüş, 2022).

Yurtsever ve Çiçekli, satış tahmini için LSTM, ARIMA ve LSTM-ARIMA modelleri kullanarak 2022 yılında bir çalışma yapmıştır. Modeller incelendiğinde en düşük başarı oranı ile LSTM-ARIMA modelinin en başarılı sonuç verdiği gözlemlenmiştir (Yurtsever & Çiçekli, 2022).

Özbay 2022’de yaptığı çalışmada saldırgan ve saldırgan olmayan olarak etiketlenen veri kümesine üç makine öğrenme; NB, SVM (Support Vector Machine, Destek Vektör Makinesi), KNN (K-Nearest Neighbors, K-En Yakın Komşu), üç derin öğrenme; ConNN, LSTM ve GRU ve üç transformatör tabanlı; BERT, XLNet ve ULMFIT (Universal Language Model Fine-tuning, Evrensel Dil Modeli İnce Ayar) modelleri uygulanmıştır. LMTweets adında bir kodlayıcı veri kümesinin özelliklerinin çıkarılması için 20001 adet Tweet üzerinden eğitilmiştir. Makine öğrenmesi modelleri olan NB için 68%, SVM için 67%, KNN için 53% doğruluk elde edilmiştir. SVM ve NB modelinin karşılaştırılabilir şekilde performans gösterdiği, KNN modelinin düşük performans gösterdiği gözlemlenmiştir. Derin öğrenme modellerinden ConNN için 83%, LSTM için 85%, GRU için 88% doğruluk değerleri elde edilmiştir. LSTM ve ConNN modellerinin, karşılaştırmalı olarak neredeyse eşit performans gösterdiği tespit edilmiştir.

Transformatör modellerinden BERT için 83%, XLNet için 85% ULMFIT için 79% doğruluk oranları elde edilmiştir. BERT modeli ile ULMFIT modeli yaklaşık olarak eşit performans göstermiştir. BERT-tabanlı mimariye dayalı bir LMTweets ile ConNN hibrit modelinin ise 96% doğruluk oranına sahip olduğu ve tüm makine öğrenmesi, derin öğrenme, transformatör modellerinden daha iyi performans gösterdiği belirlenmiştir (Özbay, 2022).

Suncak ve Aktaş 2022 yılında yaptığı çalışmada, Türkçe cümlelerdeki anlatım bozukluklarının tespiti için LSTM- ConNN hibrit modelini önermiştir. Önerdikleri yöntemin karşılaştırmasını SVM ve RFR modelleri ile gerçekleştirmiştir. ConNN -LSTM hibrit modelinin daha başarılı olduğu sonucuna varılmıştır (Suncak & Aktaş, 2022).

Uca ve arkadaşları 2022 yılında, “Sağlık Alanındaki Etkileşimlerde Duygu Analizi Yaklaşımları ve Analizi” adlı yaptığı çalışmada pozitif, negatif ve nötr olarak sınıflandırılan veri kümesi üzerinde LSTM modelini kullanmıştır. Eğitim sonucunda 94% başarı elde edilmiştir. Literatürde inceledikleri makale ve çalışmalar ile karşılattırdıklarında modelin başarılı sonuç verdiğini gözlemişlerdir (Uca ve diğ., 2022).

2022 yılında Çetiner, derin öğrenme modelleri olan LSTM, ConNN ve ConNN-LSTM hibrit modelleri ile Toksik, Şiddetli Toksik, Müstehcen, Tehdit, Hakaret, Kimlik Nefreti sınıflandırmalarını kullanarak tahmin çalışması gerçekleştirmiştir. En başarılı sonucu 99,41% ile ConNN -LSTM Hibrit modelinde elde etmiştir (Çetiner, 2022). Çetiner 2023 yılında yaptığı başka bir çalışmada ise, yine aynı derin öğrenme modelleri ile mahsul verim tahmini çalışması gerçekleştirmiştir. 0 ve 1 olarak etiketli veri kümesi üzerinde LSTM, ConNN ve ConNN -LSTM hibrit derin öğrenme modellerinde başarı oranları sırası ile 88,34%, 86,24%, 89,71% olarak elde edilmiştir (Çetiner, 2023)

2023 yılında yapılan bir çalışmada önemli petrol endeksleri ve hisse senedi fiyatlarını içeren bir veri kümesini kullanılarak RFR, LSTM ve ConNN -LSTM modelleri ile fiyat tahmini çalışması gerçekleştirmiştir. Kullanılan modellerin karşılaştırılması yapıldığında en başarı sonuç veren modelin LSTM olduğu tespit edilmiştir (Toprak, 2023).

Daşgın 2023 yılında Udemy sitesinde yer alan kurslara yapılan yorumlar ile olumlu, olumsuz olarak ayrılmış bir veri kümesi kullanarak GRU ve ConNN -LSTM hibrit modeli

ile duygu analizi çalışması yapmıştır. Eğitim tamamlandıktan sonra 95,67% doğruluk oranı ile GRU modelinden daha başarılı sonuç elde edilmiştir (Daşgın & Adem, 2023).

Pekoz ve İnkaya 2022 yılında COVID-19 döneminde vaka analizi amacıyla Üçüncü parti lojistik firması için LSTM- ConNN hibrit modelini kullanarak bir çalışma yapmıştır. Çalışmada, SMA (Simple Moving Average, Basit Hareketli Ortalama), SES (Simple Exponential Smoothing, Basit Üstel Düzeltme), LR, ARIMA kesin tahmin, KNN, Regresyon Ağacı, RFR, SVR (Support Vector Regression, Destek Vektör Regresyon) makine öğrenmesi ve MLP (Multilayer Perceptron, Çok Katmanlı İleri Beslemeli Yapay Sinir Ağı), ConNN, LSTM, ConNN -LSTM derin öğrenme yöntemlerinin karşılaştırılmasını yapılmıştır. Yapılan karşılaştırmalara göre ConNN -LSTM yönteminin diğer modellere göre daha başarılı olduğu tespit edilmiştir (Zeybel Peköz & İnkaya, 2023).

### **1.3. Tezin Yapısı**

Çalışma şu şekilde düzenlenmiştir; birinci kısımda çalışma hakkında bilgi verilerek literatürdeki duygu analizi çalışmaları incelenmiş, ikinci kısımda tez çalışmasına ait temel kavramlar irdelenmiştir. Üçüncü kısımda çalışmada kullanılan veri kümelerine ve duygu analizi için kullanılan yöntemlere, dördüncü kısımda kullanılan yöntemlerden elde edilen bulgulara yer verilmiştir. Beşinci kısımda ise eğitilen modellerin başarı oranlarından ve ileride yapılacak çalışmalar için önerilerden söz edilmiştir.

## 2. DUYGU ANALİZİNDE TEMEL KAVRAMLAR

Tez çalışmasının bu kısmında duygu analizinin kullanım alanları ve önemi, kullanılan modeller ve modellerin yazıldığı programlama dili açıklanmıştır. Modellerin mimarileri, modellerde kullanılan optimizasyon algoritmaları, aktivasyon fonksiyonları, değerlendirme metrikleri ile kayıp fonksiyonları incelenmiştir.

### 2.1. Duygu Analizi

Duygu analizi, doğal dil işleme alanında kullanılan bir teknoloji ve yapay zekâ yöntemidir. Metin verilerini otomatik olarak analiz ederek içerdikleri duygusal eğilimleri belirlemeyi amaçlamaktadır. Bu sayede insanların metinlerdeki duygusal tepkilerini anlayarak, bilgi çıkarmaya ve değerli içgörüler elde etmeye yardımcı olmaktadır (Aslan, 2022).

Duygu analizi, farklı sektörlerde ve uygulamalarda geniş bir kullanım alanına sahiptir:

**Sosyal Medya ve Pazarlama:** Şirketler, marka ve ürünlerine yönelik sosyal medya platformlarında yapılan paylaşımları izleyerek müşterilerinin duygusal tepkilerini değerlendirebilirler. Bu bilgiler, pazarlama stratejilerinin ve reklam kampanyalarının geliştirilmesinde önemli bir rol oynamaktadır.

**Müşteri Deneyimi:** Duygu analizi, müşteri geri bildirimlerini inceleyerek ürün ve hizmetlerin müşteriler üzerindeki duygusal etkisini anlamak için kullanılmaktadır. Şirketler, müşteri memnuniyetini artırmak ve eksiklikleri gidermek için bu bilgileri değerlendirmektedir.

**Siyaset ve Kamuoyu Araştırmaları:** Siyasi liderler, politikacılar ve anket yapımcıları, halkın siyasi konulardaki duygusal eğilimlerini anlamak için duygu analizini kullanmaktadır. Bu, politik stratejilerin belirlenmesinde ve seçmen davranışının analizinde önemli bir rol oynamaktadır.

**Finans ve Yatırımlar:** Duygu analizi, finansal piyasalardaki duygusal eğilimleri değerlendirmede kullanılmaktadır. Yatırımcılar, sosyal medya ve haber kaynaklarından gelen duygu analizi verilerini kullanarak daha bilinçli yatırım kararları alabilirler.

Duygu analizi, metin tabanlı verilerin duygusal içeriklerini anlama ve değerlendirme sürecinde büyük öneme sahiptir. Gelişen teknoloji ve daha zengin veri kümeleri, duygu analizinin etkinliğini artırmakta ve daha geniş uygulama alanlarına olanak tanımaktadır.

Duygu analizinin önemli özellikleri:

**Metin Sınıflandırması:** Duygu analizi, metinleri otomatik olarak belirli duygu sınıflarına ayırmayı sağlamaktadır. Bu temel özellik sayesinde, metinlerdeki duygu yoğunluğunu ve eğilimini anlamak mümkün hale gelmektedir.

**Dil ve Kültürel Uyarlamalar:** Duygu analizi modelleri, farklı dillerdeki metinleri anlamak ve kültürel farklılıkları göz önünde bulundurmak için dil çözümlemelerini ve dil özelliklerini kullanmaktadır. Bu sayede, çeşitli dil ve kültürel gruplardan gelen metinleri analiz etmek mümkün olmaktadır.

**Metin Öznitelikleri:** Duygu analizi modelleri, metinleri incelemek için çeşitli öznitelikler kullanmaktadır. Bu öznitelikler, kelime frekansları, duygusal yüklü terimler, cümle yapıları gibi metin içeriğiyle ilgili verileri içermektedir. Özniteliklerin doğru bir şekilde seçilmesi, analiz sonuçlarının kalitesini etkilemektedir.

**Eğitim Verileri ve Makine Öğrenimi:** Duygu analizi modelleri, büyük miktarda eğitim verisine dayalı olarak makine öğrenimi yöntemleriyle eğitilmektedir. Bu süreçte model, çeşitli metinlerin duygusal etiketlerini kullanarak belirli duygu sınıflarını tanımayı öğrenmektedir.

**Duygu Yoğunluğu Ölçümü:** Bazı duygu analizi modelleri, metinlerin sadece duygu sınıfını belirlemekle kalmaz, aynı zamanda metindeki duygu yoğunluğunu da ölçmektedir. Yani, metindeki duygu ifadelerinin ne kadar güçlü veya hafif olduğunu değerlendirebilir.

**Anlam Karmaşıklığı:** Metinlerdeki duygu ifadeleri bazen karmaşık ve dolaylı olabilir. Duygu analizi, metin içerisindeki ince ayrımları ve mecazları anlayabilmek için doğal dil işleme algoritmalarını kullanılmaktadır.

Gerçek Zamanlı Analiz: Gelişen teknoloji sayesinde, bazı duygu analizi sistemleri gerçek zamanlı analiz sunmaktadır. Yani, metinler anında değerlendirilmekte ve duygusal eğilimler hemen ortaya çıkmaktadır.

Uygulama Alanları Çeşitliliği: Duygu analizi, sosyal medya analitiği, müşteri deneyimi değerlendirmesi, pazarlama stratejileri belirleme, finansal piyasa analizi ve siyasi analiz gibi farklı alanlarda kullanılabilir. Bu, çeşitli sektörlerdeki karar vericilere değerli içgörüler sunmaktadır.

Gelişen yapay zekâ algoritmaları ve büyüyen veri kümeleriyle birlikte, duygu analizinin kullanım alanı ve etkinliği de artmaktadır.

## **2.2. Sosyal Medya ve Twitter (X)**

Sosyal medya, kişilerin, grupların veya organizasyonların platformlar aracılığıyla içerik paylaştığı ve iletişim kurduğu ortamlardır. Kullanıcılar, sosyal medya aracılığıyla bağlantı kurabilir, düşüncelerini paylaşabilir, güncel olaylar hakkında bilgi alabilir ve çeşitli konularda etkileşimde bulunabilirler.

Twitter, 2006 yılında Evan Williams, Biz Stone ve Jack Dorsey tarafından kurulmuştur (Picard, 2011). Günümüz dijital dünyasının en tanınmış sosyal medya mecralarından biri olan ve 2023 yılında X adını alan Twitter, dünya çapında milyonlarca insanın bağlantı kurma, bilgi paylaşma ve anlık iletişim sağlama amacıyla kullandığı güçlü bir sosyal medya ağıdır. Kullanıcılarına kısa ve öz metin tabanlı mesajlar olan tweetler gönderme imkânı sağlamaktadır.

Twitter, haberleri öğrenmek, güncel olaylar hakkında tartışmak, ilgi alanlarına göre içerik takip etmek ve farklı insanlarla etkileşim kurmak için kullanıcılarına geniş bir platform sunmaktadır. Ünlüler, politikacılar ve diğer kamu çalışanları da Twitter'ı kullanarak takipçileriyle doğrudan iletişim kurmakta ve düşüncelerini paylaşmaktadır.

Tweet, en fazla 280 karakter uzunluğunda olan metin mesajlarıdır ve kullanıcılar tarafından oluşturulup paylaşılmaktadır (Gligorić ve diğ., 2018). Twitter kullanıcıları, takip ettikleri kişilerin Tweetlerini ana sayfalarında görmekte ve bu Tweetlere beğeni veya paylaşım yapabilmektedir. Şekil 2.1'de Tweet örneği verilmiştir.



Şekil 2.1. Tweet Örneği (Pham ve diğ., 2019)

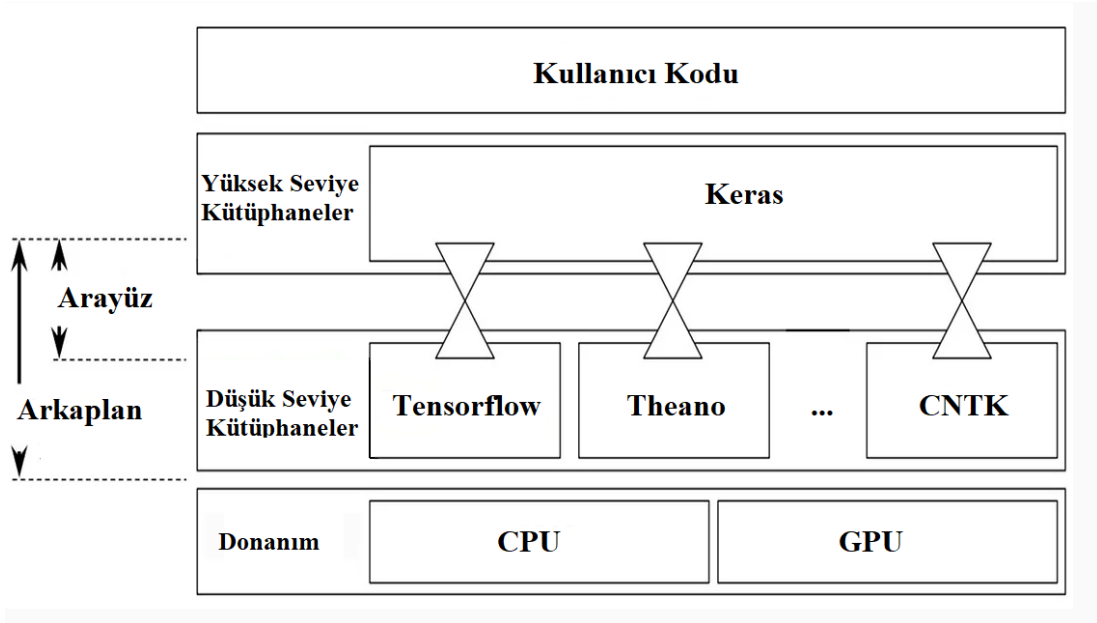
Platform, metin dışında resim, video ve bağlantı gibi medya içeriklerinin de paylaşılmasına olanak tanımaktadır. Ayrıca, etiket (#) kullanılarak kullanıcılar tweetleri belirli konularla ilişkilendirebilmekte ve diğer kullanıcıların ilgili konulardaki tweetleri bulmasını kolaylaştırmaktadır.

## 2.3. Python Programlama ve Çerçeveseler

### 2.3.1. Python Programlama

Python, derin öğrenme alanında oldukça önemli bir role sahip olan popüler bir programlama dilidir. Derin öğrenme, yapay sinir ağları üzerine kurulan ve büyük miktardaki veriyi kullanarak karmaşık problemleri çözmeye odaklanan bir makine öğrenmesi alt alanıdır. Python, bu alanda kullanılan çeşitli derin öğrenme çerçeveleri sayesinde geliştiricilerin tercih ettiği bir dil haline gelmiştir.

Python'un derin öğrenme ile olan ilişkisi, bu alanda kullanılan çeşitli güçlü çerçeveler üzerinden ortaya çıkmaktadır. TensorFlow, Keras, PyTorch ve Scikit-learn gibi derin öğrenme için özel olarak tasarlanmış çerçeveler bulunmaktadır. Şekil 2.2'de standart Derin Öğrenme çerçevelerinin yapısı gösterilmektedir. Bu çerçeveler, geliştiricilere derin sinir ağlarını oluşturma, eğitme ve değerlendirmeye yönelik özel araçlar ve fonksiyonlar sunmaktadır.



Şekil 2.2. Standart Derin Öğrenme Çerçeve Yapısı (Pham ve diğ., 2019)

TensorFlow, Google tarafından geliştirilen açık kaynaklı bir derin öğrenme çerçevesidir. Python ile uyumlu bir şekilde kullanılarak, yüksek performanslı hesaplamalar ve karmaşık model yapıları sağlamaktadır.

Keras, yine Python ile uyumlu olan bir derin öğrenme çerçevesidir ve TensorFlow, CNTK (Microsoft Cognitive Toolkit, Microsoft Bilişsel Araç Takımı) ve Theano gibi çeşitli arka uçları desteklemektedir. Yüksek seviyeli bir API (Application Programming Interface, Uygulama Programlama Arayüzü) olması, hızlı prototipleme yapmak isteyen geliştiriciler için ideal bir seçenektir.

PyTorch, Facebook tarafından geliştirilen bir derin öğrenme çerçevesidir ve Python odaklıdır. Esnek yapısı sayesinde araştırmacılar ve geliştiriciler tarafından tercih edilmektedir.

Scikit-learn ise Python dilinde popüler olan bir makine öğrenmesi çerçevesidir. Sınıflandırma, regresyon, kümeleme gibi görevler için yaygın olarak kullanılmaktadır.

Python'un derin öğrenme ile ilişkisi, bu çerçeveler sayesinde güçlenmektedir. Python'un esnek, kullanıcı dostu ve kolay anlaşılır yapısı, derin öğrenme geliştiricilerinin verimli bir şekilde çalışmasına olanak tanımaktadır. Gelişen derin öğrenme teknolojileri ile Python dilinin bu alandaki önemi ve etkisi giderek artmaktadır.

### 2.3.2. Tensorflow Çerçevesi

TensorFlow çerçevesi, ikinci nesil bir sistem olan ekibi olan Google Brain Ekibi tarafından geliştirilmiştir (Bilgin, 2021). TensorFlow, esas olarak makine öğrenimi için kullanılan açık kaynaklı bir platformdur. Veri bilimcilerin makine öğreniminde en son teknolojiyi kullanmalarını sağlamaktadır. Geliştiriciler TensorFlow’u kullanarak makine öğrenimi destekli sistemleri hızla oluşturabilmekte ve dağıtabilmektedir.

TensorFlow birden fazla CPU (Central Processing Unit, Merkezi İşlemci Birimi) veya GPU (Graphics Processing Unit, Grafik İşlemci Birimi) üzerinde çalışabilmektedir. TensorFlow, son yıllarda yaygın bir derin öğrenme çerçevesi haline gelmiştir. Basit bir yapay sinir ağından ConNN veya RNN’ye kadar herhangi bir derin öğrenme yapısı oluşturmak için kullanılabilir. TensorFlow çerçevesi, ConNN veya RNN gibi derin öğrenme mimarisini desteklemek için farklı uygulama programlama arayüzlerini entegre etmektedir (Konasani & Kadre, 2021).

### 2.3.3. Keras Çerçevesi

Keras, Python’da yazılmış, TensorFlow ve diğer alt düzey çerçevelerin üzerinde çalışabilen yüksek düzeyli bir sinir ağı API’sidir. Hızlı denemelere izin vermeye ve basit ve hızlı prototip oluşturmaya odaklanarak tasarlanmıştır (Chicho & Sallow, 2021).

Keras, evrişimli katmanlar, tekrarlayan katmanlar, bırakma katmanları gibi çeşitli sinir ağı bileşenlerini kabul ederek varyasyonlarını desteklemektedir.

Kod, CPU ve GPU gibi araçları dinamik olarak işleyerek bunların en iyi şekilde kullanılmasını sağlamaktadır. Ayrıca aktivasyon fonksiyonları, optimize ediciler, metrik formüller ve eğitim oturumlarını kolaylıkla idare etmek için gerekli prosedürlerin uygulamalarına sahiptir (Bohrer ve diğ., 2020).

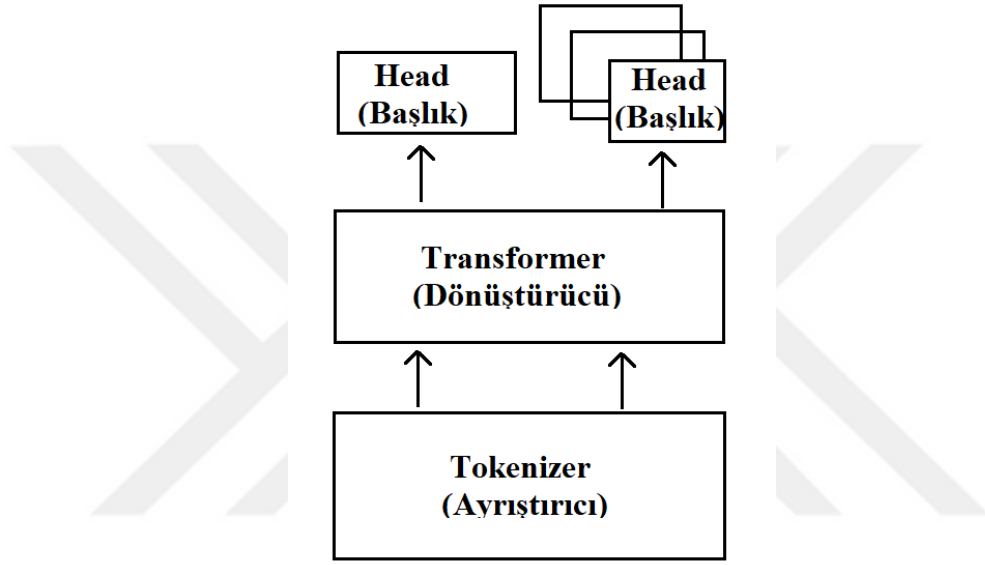
Bir Derin Öğrenme eklentisi olan Keras çerçevelerindeki işlevselliği entegre etmekte ve bu da TensorFlow’daki işlevleri Python’a entegre etmektedir.

Keras çerçevesi Tokenizer sınıfına sahiptir. Bu sınıfın fit\_on\_texts ve texts\_to\_sequences fonksiyonları vardır. fit\_on\_texts fonksiyonu ile metin verileri işlenerek kelime sözlüğü

oluşturulmasına olanak sağlamaktadır. `texts_to_sequences` fonksiyonu ile de oluşturulan kelime sözlüğü kullanılarak metin verileri sayı dizilerine dönüştürülmektedir.

#### 2.3.4. Transformer Çerçevesi

Transformer (Dönüştürücü), NLP (Natural Language Processing, Doğal Dil İşleme) alanında önemli bir ilerleme kaydetmek için geliştirilen açık kaynaklı bir Python çerçevesidir.



Şekil 2.3. Dönüştürücü Çerçevesi Yapısı (Wolf ve diğ., 2020)

Her model Şekil 2.3'te görüldüğü gibi ayrıştırıcı, dönüştürücü ve başlıklardan oluşmaktadır. Model sabit bir başlıkla ön eğitime tabi tutularak farklı görevler için alternatif başlıklara ince ayar yapabilmektedir.

Transformer, bir dizi önceden eğitilmiş modeli içermekte ve bu modellerde, büyük miktarda veri kullanılarak dilin temel yapılarını ve bağlamlarını anlamak için önceden eğitilmektedir. Önceden eğitilmiş modeller, sınıflandırma, çeviri, anlam çıkarma, doldurma, metin üretimi gibi çeşitli NLP görevlerini çözmek için kullanılabilir.

Bu çerçeve, popüler derin öğrenme mimarileri olan BERT, GPT (Generative Pre-trained Transformer, Üretken Ön İşlemeli Dönüştürücü), RoBERTa (Robustly Optimized BERT Approach, Sağlam Optimize Edilmiş BERT Yaklaşımı), XLNet gibi modelleri

içermektedir. Bu modeller, büyük miktarda metin verilerini analiz ederek dildeki ilişkileri, bağlamları ve anlamları anlamak için benzersiz bir yeteneğe sahiptir.

Transformer çerçevesi, dil çapraz transferi sağlayabilme özelliğine sahiptir. Yani, aynı modeli farklı diller için kullanabilmek mümkündür. Bu, önceden eğitilmiş modellerin farklı dillere uyarlanabilmesi ve dil işleme görevlerinin çeşitli dillerde çözülebilmesi anlamına gelmektedir.

Ayrıca, Transformer çerçevesi ile Hugging Face, Model Hub adında bir platform sunmaktadır. Bu platform sayesinde araştırmacılar ve geliştiriciler, kendi önceden eğitilmiş modellerini paylaşabilir ve topluluk üyeleriyle iş birliği yapabilirler. Aynı zamanda, kullanıcılar da Model Hub'dan hazır modelleri indirerek NLP projelerini hızlandırabilir ve özelleştirebilirler.

Transformer çerçevesi, doğal dil işleme alanında derin öğrenme modellerini kullanmak isteyenler için güçlü bir kaynaktır. Önceden eğitilmiş modellerin gücü ve dil çapraz transfer yeteneği sayesinde, NLP görevlerini daha az eğitim verisiyle etkili bir şekilde çözmek mümkün hale gelmektedir. Bu çerçeve, dil işleme alanındaki gelişmelerin hızlanmasına katkıda bulunan ve araştırmacılar için büyük bir kolaylık sağlayan önemli bir araçtır.

#### **2.4. Doğal Dil İşleme**

Doğal Dil İşleme, bilgisayarlar tarafından insan dilinin anlaşılmasını, yorumlanmasını, üretilmesini ve işlenmesini sağlayan bir disiplindir. Metin tabanlı verilerin analiz edilmesini ve dilin anlamının çıkarılmasını hedeflemektedir. Bu sayede, bilgisayarlar metinler üzerinde anlam üretme, dilbilgisi yapılarını tanıma, metinleri sınıflandırma, çeviri yapma, metin tabanlı sorulara yanıt verme gibi dil tabanlı görevleri gerçekleştirebilmektedir.

Doğal Dil İşlemenin kullanım alanları oldukça geniş olmakla beraber aşağıda örnekleri verilmiştir:

**Metin Sınıflandırma:** Doğal Dil İşleme, metinleri belirli kategorilere veya sınıflara ayırma görevlerinde kullanılmaktadır. Bir e-postanın spam veya önemli olduğunun belirlenmesi,

bir Tweetin pozitif veya negatif olduğunun sınıflandırması gibi görevlerde kullanılabilir.

**Duygu Analizi:** Doğal Dil İşleme, metinlerdeki duygusal ifadeleri tanımlama ve sınıflandırmada kullanılmaktadır. Bir ürün yorumunun olumlu, olumsuz veya nötr olduğunu belirlemede kullanılabilir.

**Metin Üretimi:** Doğal Dil İşleme, metin tabanlı otomatik özetleme, makale oluşturma veya diyalog sistemleri gibi görevlerde kullanılmaktadır. Bu sayede, bir metin temel alınarak yeni metinler oluşturulabilir veya makine-insan etkileşiminde doğal dilde yanıtlar üretilebilir.

**Makine Çevirisi:** Doğal Dil İşleme, bir dildeki metni başka bir dile çevirme görevinde kullanılmaktadır. Bu, metin tabanlı çeviri hizmetlerinde veya çok dilli iletişimde önemli bir rol oynamaktadır.

**Metin Tabanlı Soru-Cevap:** Doğal Dil İşleme, metin tabanlı sorulara yanıt bulma görevinde kullanılmaktadır. Bir metin belgesindeki bir sorunun yanıtını bulma veya bir metin tabanlı soruya cevap verme gibi görevlerde kullanılabilir.

**Metin Özetleme:** Haber makalelerini veya belgeleri özetlemek için Doğal Dil İşleme teknikleri kullanılabilir. Bu, kullanıcılara içeriği hızlıca anlama imkânı sağlamaktadır.

**İşlemsel Dil İşleme:** Doğal Dil İşleme, metin tabanlı işlemleri gerçekleştiren otomasyon sistemlerinde kullanılmaktadır. Bankacılık sektöründe müşteri taleplerini otomatik olarak işlemek veya müşteri hizmetlerinde sık sorulan sorulara otomatik yanıtlar üretmek için Doğal Dil İşleme kullanılabilir.

**Metin Madenciliği:** Doğal Dil İşleme, büyük metin veri kümelerini analiz ederek anlamlı bilgiler çıkarmayı sağlamaktadır. Bu, metin tabanlı verilerde trendleri, kalıpları veya ilişkileri keşfetmek için kullanılabilir. Sosyal medya verilerinde marka algısı veya tüketici eğilimlerini anlamak için Doğal Dil İşleme teknikleri kullanılabilir.

Konuşma Tanıma: Ses kayıtlarını veya canlı konuşmayı yazıya dönüştürmek için Doğal Dil İşleme teknikleri kullanılabilir. Sesli komutları algılamak veya transkriptler oluşturmak gibi görevlerde kullanılmaktadır.

Anlamsal Analiz: Doğal Dil İşleme, metinlerdeki anlam yapılarını anlama ve analiz etme görevinde kullanılmaktadır. Bu, anlatılan hikayelerdeki ilişkileri veya anlatı akışını anlamak için kullanılabilir. Bir romanın karakter ilişkilerini analiz etmek veya bir film incelemesindeki ana temaları belirlemek için Doğal Dil İşleme teknikleri kullanılabilir.

Bilgi Çıkarma: Metinlerden yapısal bilgileri çıkarma görevinde kullanılmaktadır. Belirli bilgi öğelerini (isimler, tarihler, yerler gibi) tanımlamak ve etiketlemek için kullanılabilir. Bir makaleden kişi adlarını veya ürünlerin fiyatlarını çıkarmak gibi görevlerde kullanılabilir.

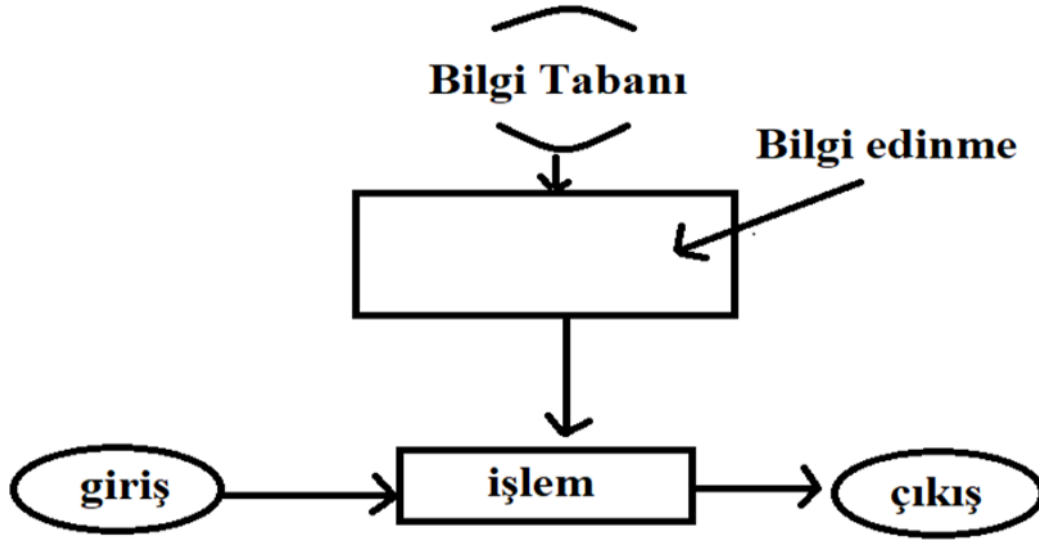
Dil Modellemesi: Dilin yapısal özelliklerini ve kullanımını modelleme görevinde kullanılmaktadır. Dil modelleri, metin üretimi, otomatik düzeltme veya kelime sıralamalarını modelleme gibi görevlerde kullanılabilir.

Doğal Dil İşleme, bilgisayarların metinleri işleyebilme yeteneğini artırarak, otomatikleştirmeye, anlamı çıkarmaya ve insan-makine etkileşimini geliştirmeye olanak tanımaktadır.

## **2.5. Makine Öğrenmesi**

Makine öğrenmesi, bilgisayar sistemlerinin belli bir görevi icra etmek için verilerden öğrenmesini ve deneyimlerini kullanarak performanslarını geliştirmesini sağlayan bir yapay zekâ alanıdır. Geleneksel programlama yaklaşımlarının aksine, makine öğrenmesi, bilgisayar programcısının adım adım talimatlar vermesi yerine, verilere dayanarak modeller oluşturması ve öğrenmesi ile sorunların çözülmesini hedeflemektedir.

Makine öğrenmesi, büyük miktarda ve karmaşık verilerin olduğu veya insanların tanımlamasının güç olduğu problemler için etkili bir çözüm sunmaktadır. Bu veri analizi yöntemi, işlem süreçlerini optimize etmek, öngörüler yapmak, desenleri tanımak, sınıflandırmak ve daha fazlası için kullanılabilir.



Şekil 2.4. Makine Öğrenmesi Yapısı (Zecevic & Glisovic, 2008)

Makine öğrenme sistemlerinin tasarımı temelde çok basittir. Şekil 2.4'te görüldüğü gibi; bilgi tabanı, bu bilginin sentezlendiği modül ve çıktı temsilinin elde edilebilmesi için girdi temsili üzerinden işleme modülü içeri alınmalıdır.

Makine öğrenmesi temelde üç ana kategoride incelenmektedir:

**Gözetimli Öğrenme:** Bu yöntemde, makine eğitim verilerindeki girdi ile çıktılar arasındaki ilişkileri öğrenmekte ve yeni girdilerle sonuçları tahmin etmeye çalışmaktadır. Eğitim verileri, girdi ve çıktıları içeren etiketli verilerden oluşmaktadır. Sınıflandırma ve regresyon, gözetimli öğrenmenin iki temel alt dalıdır. Sınıflandırma, verileri farklı kategorilere ayırmak için kullanılırken, regresyon, veriler arasındaki ilişkiyi tahmin etmek için kullanılmaktadır.

**Gözetimsiz Öğrenme:** Bu yöntemde, makine, etiketlenmemiş verilerdeki desenleri ve yapıları keşfetmeye çalışmaktadır. Eğitim verileri, sadece girdileri içermekte ve çıktılar etiketlenmemektedir. Kümeleme ve boyut indirgeme, gözetimsiz öğrenmenin iki temel alt alanıdır. Kümeleme, verileri benzer özelliklere sahip gruplara ayırırken, boyut indirgeme, verileri daha az boyutta temsil etmek için kullanılmaktadır.

**Pekiştirmeli Öğrenme:** Bu yöntemde, bir ajan, belirli bir görevi gerçekleştirmek için çevresiyle etkileşime girmekte ve stratejisini optimize etmeye çalışmaktadır. Bu

eylemlerin sonuçlarına göre ödülleri veya cezalar almaktadır. Zamanla daha iyi kararlar almayı öğrenmektedir.

Makine öğrenmesi birçok alanda kullanılmaktadır:

Doğal Dil İşleme: Metin verilerinin analizi, çeviri, duygu analizi gibi Doğal Dil İşleme problemlerinde makine öğrenmesi teknikleri kullanılmaktadır.

Görüntü İşleme: Nesne tanıma, yüz tanıma, görüntü sınıflandırması gibi alanlarda makine öğrenmesi modelleri başarıyla kullanılmaktadır.

Sağlık ve Tıp: Hastalık teşhisi, ilaç geliştirme, tedavi planlaması ve genetik analizler gibi birçok sağlık alanında makine öğrenmesi önemli rol oynamaktadır.

Otomotiv: Sürücüsüz araçlar, trafik yönetimi gibi otomotiv endüstrisinde makine öğrenmesi büyük bir potansiyele sahiptir.

Finans: Hisselerin fiyat tahmini, kredi değerlendirme, dolandırıcılık tespiti gibi finansal analizlerde makine öğrenmesi kullanılmaktadır.

Perakende ve E-ticaret: Ürün önerileri, talep tahminleri, fiyat optimizasyonu gibi alanlarda makine öğrenmesi ile verimlilik artırılmaktadır.

Makine öğrenmesi modelleri, geniş bir yelpazede bulunmaktadır. SVM, DTR, KNN, RFR, derin öğrenme, dönüştürücü gibi modeller farklı problemleri çözmek için kullanılmaktadır. Model seçimi, veri türüne, problemin karmaşıklığına ve performans gereksinimlerine göre değişmektedir.

## **2.6. Derin Öğrenme Modelleri**

Derin öğrenme, makine öğrenmesinin bir dalıdır ve yapay sinir ağlarına dayanmaktadır. Büyük miktardaki verilerden otomatik olarak öğrenme yapabilen bir yapay zekâ yöntemidir.

Derin öğrenme, karmaşık ilişkileri keşfetmek amacıyla verileri işlemek için çok katmanlı yapıları kullanmaktadır. Bu yapılar, girdi katmanı, gizli katmanlar ve çıktı katmanından oluşmaktadır. Her katman, verileri işleyerek bir sonraki katmana iletmektedir. Bu şekilde,

derin öğrenme, verilerdeki önemli özellikleri otomatik olarak çıkarabilmekte ve temsil edebilmektedir.

Derin öğrenme, ağırlıkların otomatik olarak ayarlanmasıyla gerçekleşmektedir. Büyük veri kümeleri üzerinde tekrarlayan işlemlerle ağırlıkların güncellenmektedir. Bu süreç, geri yayılım algoritması kullanılarak gerçekleştirilmektedir. Ağırlık hata miktarını azaltmak ve performansı iyileştirmek için ağırlıklar ayarlanmaktadır.

Derin öğrenme teknikleri, geniş bir yelpazede etkileyici sonuçlar elde etmektedir. Özellikle görüntü işleme, ses işleme, doğal dil işleme ve tahminleme gibi alanlarda etkili kullanılmaktadır. Büyük veri kümeleri üzerinde yüksek performans sağlama ve karmaşık problemleri çözebilme yeteneği ile derin öğrenme, güçlü bir araçtır. Ayrıca, derin öğrenme modelleri, insan gibi algılama, tanıma ve anlama yeteneklerini keşfetmek amacıyla da kullanılmaktadır.

Derin öğrenme, makine öğrenmesi alanında büyük bir ilerleme sağlamış ve birçok alanda başarılı uygulamalara imkân tanımıştır. Yapay zekâ alanında önemli bir rol oynayan derin öğrenme yüksek hesaplama gücü gerektiren büyük veri kümelerini işleme kapasitesine sahiptir.

Derin öğrenme, çeşitli modelleri içeren geniş bir alandır. Derin öğrenme alanında sık kullanılan bazı modeller:

ANN (Artificial Neural Networks, Yapay Sinir Ağları): Derin öğrenmenin temelini oluşturan modellerdir. Girdi, gizli ve çıktı katmanlarından oluşan bir yapıya sahiptir ve ağırlıkların otomatik olarak ayarlandığı bir öğrenme süreci ile çalışmaktadır.

ConNN: Evrişimli sinir ağları, özellikle görüntü işleme alanında büyük başarı elde eden derin öğrenme modelleridir. Evrişim ve havuzlama katmanları gibi özel katmanlardan oluşmaktadır.

RNN: Tekrarlayan sinir ağları, sıralı verilerin veya zaman serilerinin işlenmesi için kullanılan bir derin öğrenme modelidir. Bu ağlar, geçmiş bilgiyi hatırlama yeteneği sayesinde zaman bağımlılıklarını ve sıralı yapıları modellendirme yeteneğine sahiptir.

DBN (Deep Belief Networks, Derin İnşa Edilebilir Ağlar): Gözetimsiz makine öğrenme için kullanılan derin öğrenme modelleridir. Katman eğitim yaklaşımı ile daha karmaşık özelliklerin öğrenilmesini sağlamaktadır.

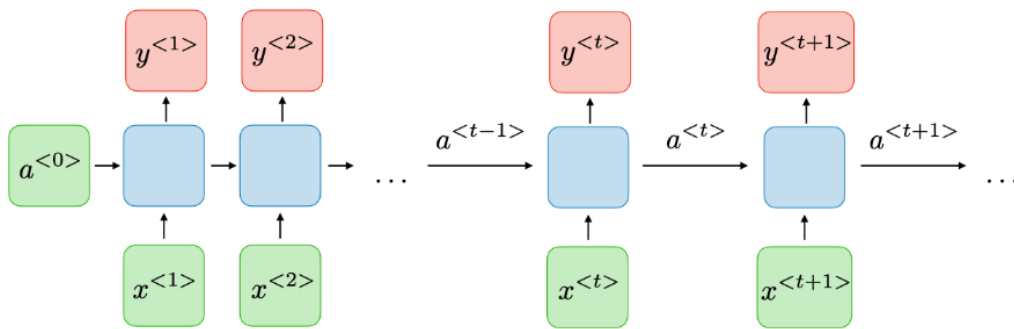
DQN (Deep Q-Networks, Derin Q-Ağları): Pekiştirmeli öğrenme alanında kullanılan bir derin öğrenme modelidir. Ağın değerini tahmin etmek ve en iyi eylemi seçmek için kullanılmaktadır.

GAN (Generative Adversarial Networks, Çekişmeli Üretici Ağ): Gerçekçi veri örnekleri üretmek için kullanılan bir derin öğrenme modelidir. Çekişmeli ağ ve ayrıştırıcı ağ olmak üzere iki bileşenden oluşmaktadır ve birbirleriyle rekabet ederek öğrenme gerçekleştirmektedir.

Her bir modelin kendine özgü özellikleri ve kullanım alanları bulunmaktadır. Derin öğrenme alanı hızla gelişmekte olup yeni modeller sürekli olarak keşfedilmektedir.

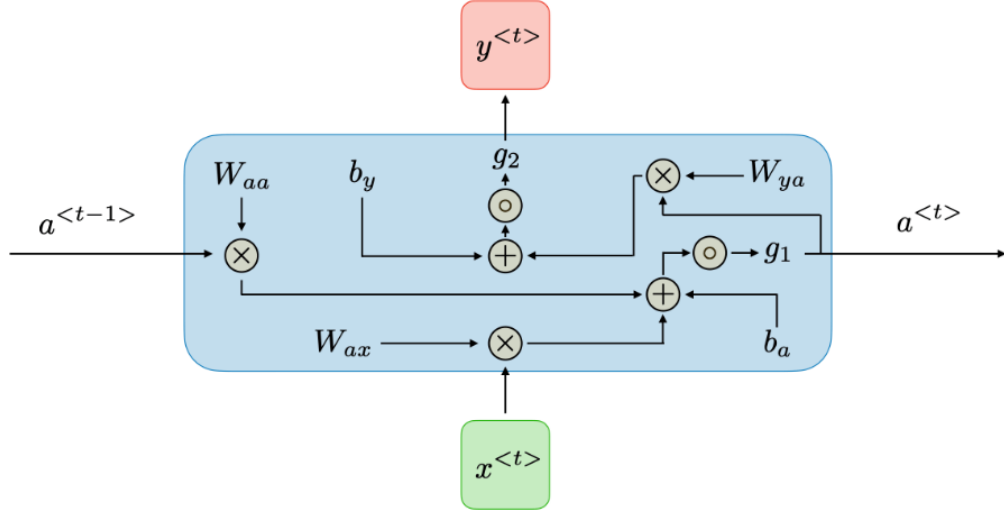
### 2.6.1. RNN (Recurrent Neural Network, Özyinelemeli Sinir Ağı)

RNN, ardışık verilerin işlenmesi için tasarlanmış bir yapay sinir ağı mimarisidir. RNN, veri akışını bir hücre durumu aracılığıyla gerçekleştirmektedir ve bu hücre durumu, geçmiş bilgilerin gelecekteki çıktılara olan etkisini yakalamak için geri besleme döngüsü kullanılmaktadır (Wang ve diğ., 2021).



Şekil 2.5. RNN Mimarisi (Amidi & Amidi, 2018)

Şekil 2.5'te gösterilen RNN mimarisinin hücre yapısı Şekil 2.6'da gösterilmiştir.



Şekil 2.6. RNN Hücre Yapısı (Amidi & Amidi, 2018)

RNN mimarisine göre her bir  $t$  zamanında,  $a^{<t>}$  aktivasyonu ve  $y^{<t>}$  çıktısı Denklem (2.1) ve Denklem (2.2)'deki gibi;

$$a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a) \quad (2.1)$$

$$y^{<t>} = g_2(W_{ya}a^{<t>} + b_y) \quad (2.2)$$

şeklinde olacaktır.

Temel bir RNN hücresi şimdiki girdi olan  $x^{<t>}$  ve geçmişten gelen bilgileri içeren önceki  $a^{<t-1>}$  gizli durumunu girdi olarak alarak bir sonraki RNN hücresine verilen ve aynı zamanda  $y^{<t>}$  çıktısını tahmin etmek için kullanılan  $a^{<t>}$ 'yi çıktı olarak vermektedir.  $W_{aa}, W_{ax}, W_{ya}, b_a, b_y$  katsayılar ve  $g_1$  ile  $g_2$  aktivasyon fonksiyonlarıdır.

Geleneksel RNN'lerde uzun dönem bağımlılıkları ele almak zor olabilmektedir. Bu durum, uzun bir veri akışı boyunca hafıza kaybına ve gradyan problemlerine yol açabilmektedir.

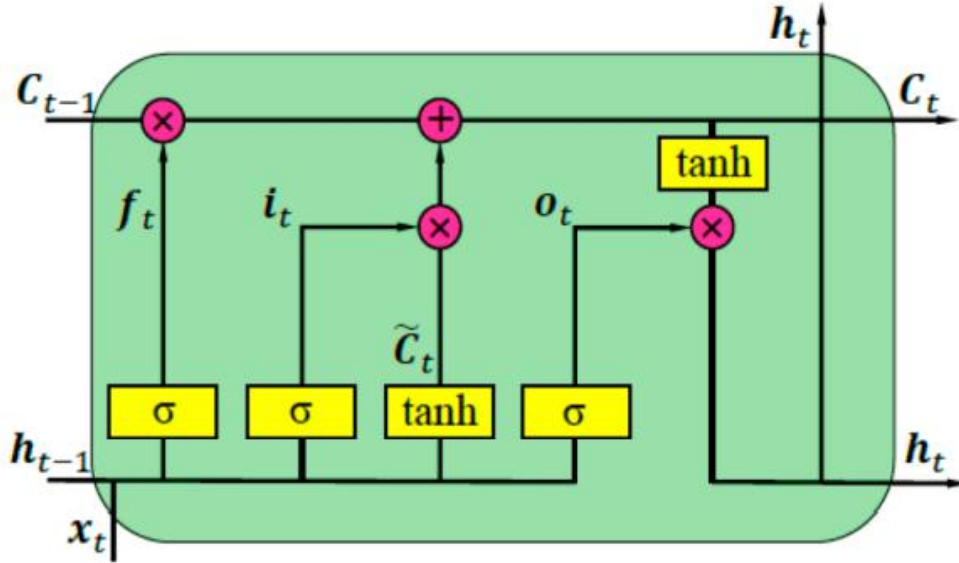
## 2.6.2. LSTM (Long Short-Term Memory, Uzun-Kısa Vadeli Bellek)

LSTM, RNN'nin bir türüdür ve RNN'nin yaşadığı sorunları çözmek için özel olarak tasarlanmıştır. LSTM, hücre durumunu kullanarak bilgiyi uzun bir süre boyunca saklama

yeteneğine sahiptir. Hücre durumu, RNN'ye kıyasla daha gelişmiş bir hafıza yapısıdır ve veri akışını daha etkili bir şekilde yönetmektedir.

LSTM, unutmama, giriş ve çıkış kapısı adı verilen üç önemli kapı içermektedir. Bu kapılar, bilginin hafıza durumunda nasıl akıp gideceğini kontrol etmektedir. Unutmama kapısı, geçmiş bilgilerin ne kadarının unutulacağını belirlerken, giriş kapısı yeni bilgilerin ne kadarının hafıza durumuna ekleneceğini belirlemektedir. Çıkış kapısı ise hafıza durumundaki bilgilerin ne kadarının kullanılacağını ve son çıktılara aktarılacağını yönetmektedir.

Bu farklı kapılar sayesinde LSTM, uzun dönem bağımlılıkları daha etkili bir şekilde ele alabilmektedir. Bu özelliği, dil işleme, metin sınıflandırma ve zaman serisi analizi gibi problemlerde daha iyi performans sağlamaktadır. LSTM, RNN'ye kıyasla daha derin ve karmaşık bir yapıya sahip, ancak bu ek bileşenler sayesinde daha güçlü bir hafıza ve daha etkili bilgi akışı sağlamaktadır.



Şekil 2.7. LSTM Hücre Yapısı (Kara, 2019)

Şekil 2.7'de gösterildiği gibi bir LSTM hücresinde sigmoid ve tanh katmanları bulunmaktadır.

İlk sigmoid katmanında, mevcut bilgi  $x(t)$  ve önceki hücreden gelen bilgi  $h(t-1)$  değerlerine bakılarak  $C(t-1)$  hücre durumu 0-1 çıktısını vermektedir. 0-1 çıktısı giriş

bilgisinin hangisinin unutulacağına hangisinin tutulacağına karar vermektedir. 1 değeri bilginin tutulması anlamına gelmekte, 0 değeri bilginin unutulacağı anlamına gelmektedir (Greff ve diğ., 2017).

$f_t$  unutma geçidinin hesaplanması Denklem (2.3)'deki gibi;

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.3)$$

şeklinde olacaktır.

Bir sonraki sigmoid katmanında hangi değerlerin güncelleneceğine karar verilmektedir. Bu işlemi sağlayan giriş geçidi  $i(t)$  Denklem (2.4)'teki gibi;

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2.4)$$

şeklinde olacaktır.

Tanh katmanında ise yeni olası değerler için vektör oluşturularak hücre durumu  $C(t)$  yeni duruma eklenmektedir. Yeni olası değerler için  $\tilde{C}$  hesaplaması, hücre durumu  $C(t)$  Denklem (2.5) ve Denklem (2.6)'daki gibi;

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (2.5)$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (2.6)$$

şeklinde olacaktır.

Çıkışa hangi bilginin yollanacağına karar verilmesi için sigmoid katmanı ve tanh fonksiyonu kullanılmaktadır. Bu sayede sadece karar verilen parçalar hücreden çıkmaktadır. Çıkış geçidi  $o(t)$ , hücre bilgisinin çıkışı  $h(t)$  hesaplaması Denklem (2.7) ve Denklem (2.8)'deki gibi;

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2.7)$$

$$h_t = o_t \times \tanh(C_t) \quad (2.8)$$

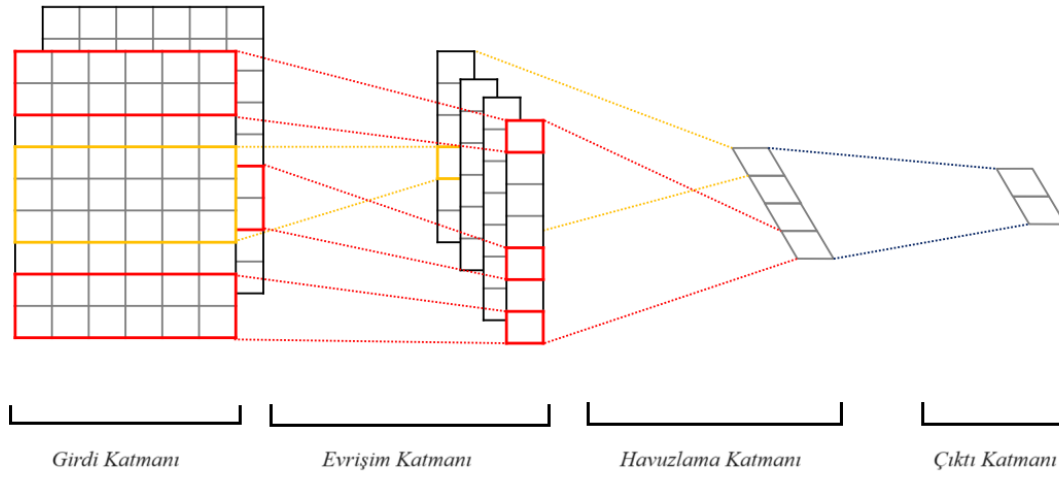
şeklinde olacaktır.

### 2.6.3. ConNN (Convolutional Neural Networks, Evrişimli Sinir Ağı)

ConNN, görüntü, ses, metin gibi verilerin analizinde kullanılan etkili bir yapay sinir ağı mimarisidir. Özellikle sınıflandırma, nesne tanıma ve öznitelik çıkarımı gibi veri analizi problemlerinde başarılı sonuçlar elde etmektedir (Li ve diğ., 2018).

ConNN verinin özelliklerini otomatik olarak çıkarabilmek için öğrenilebilir filtreleri kullanmaktadır. Bu filtreler, evrişim ve havuzlama katmanlarından oluşan yapı içinde kullanılmaktadır (Zhang ve diğ., 2021).

Şekil 2.8’de ConNN mimarisinin katman yapısı gösterilmektedir.



Şekil 2.8. ConNN Mimarisi (Kim, 2014)

Evrişim katmanı, filtreler veri üzerinde hareket ettirilerek, veri üzerinde belirli alanlarda özelliklerin varlığını tespit etmek için kullanılmaktadır. Bu işlem özellik haritalarının oluşturulmasını sağlamaktadır. Her bir özellik haritası, farklı filtreler tarafından yakalanmış farklı özellikleri temsil etmektedir.

Evrişim işleminin matematiksel gösterimi Denklem (2.9)’daki gibi, ayrık fonksiyon gösterimi Denklem (2.10)’daki gibi;

$$s(t) = (x * w)(t) = \int x(a)w(t - a)da \quad (2.9)$$

$$s(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a) \quad (2.10)$$

şeklinde olacaktır. Bu gösterimde  $x$  girdi verisi,  $w$  çekirdek (kernel) ve  $s$  özellik haritasıdır (Gummeson, 2016).

Evrişim katmanının işleyişi aşağıdaki adımlardan oluşur:

1. Evrişim Filtreleri: Her bir filtre, belirli bir boyutta veri dilimlerini kullanarak evrişim işlemini uygulamaktadır. Filtreler, farklı özelliklerin veya desenlerin belirlenmesi için verinin farklı bölgelerini analiz etmektedir.
2. Evrişim İşlemi: Evrişim filtreleri, verinin üzerinde kaydırma işlemi yaparak evrişim işlemini gerçekleştirmektedir. Bu işlem, filtrelerin verinin farklı bölgelerinde özelliklerin varlığını tespit etmesini sağlamaktadır. Evrişim işlemi, filtrelerin konumsal bilgiyi yakalayarak öznitelik haritalarını oluşturmasını sağlamaktadır.
3. Öznitelik Haritaları: Evrişim işlemi sonucunda, her bir filtre tarafından yakalanan özelliklerden oluşan öznitelik haritaları elde edilmektedir. Öznitelik haritaları, verideki farklı özelliklerin temsilini sağlamaktadır. Her bir öznitelik haritası, evrişim filtrelerinden biriyle ilgili özellikleri yansıtmaktadır.
4. Aktivasyon Fonksiyonu: Evrişim katmanı genellikle, öznitelik haritalarındaki önemli özellikleri vurgulayan bir aktivasyon fonksiyonu ile kullanılmaktadır. Bu fonksiyon, öznitelik haritalarındaki değerleri sıfır veya pozitif değerlere dönüştürerek daha belirgin özelliklerin çıkarılmasına yardımcı olmaktadır.

Havuzlama katmanı, özellik haritalarının boyutunu azaltmak ve özniteliklerin özetlenmesini sağlamak için kullanılmaktadır. Bu katman, özellik haritalarının alt örneklerini alarak boyutu küçültmektedir. Bu sayede, modelin daha hızlı ve etkili bir şekilde öznitelikleri tanımasına ve önemli bilgilerin korumasına yardımcı olmaktadır.

Havuzlama katmanının işleyişi aşağıdaki adımlardan oluşur:

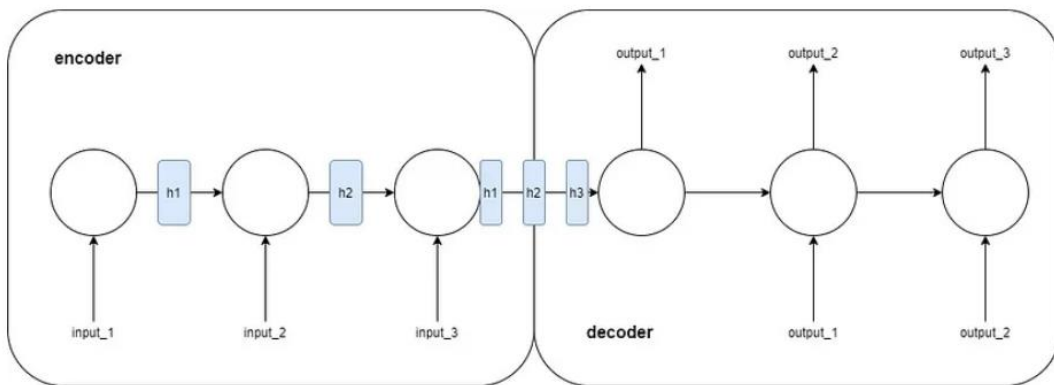
1. Girdi Verisi: Havuzlama katmanı, evrişim katmanından elde edilen öznitelik haritalarını belirlenen havuzlama katmanının boyutuna göre almaktadır.
2. Pencere ve Kaydırma: Özetleme işlemi için belirli bir pencere boyutu ve kaydırma adımı kullanılmaktadır. Pencere boyutu, özetlemenin uygulanacağı dilim boyutunu belirtmektedir. Kaydırma adımı ise pencerenin veri üzerinde nasıl kaydırılacağını belirlemektedir.

3. Özetleme İşlemi: Her pencere bölgesinde en büyük ya da ortalama değerini seçerek özetleme işlemini gerçekleştirmektedir. Bu işlem, her pencere bölgesindeki en belirgin özelliği yakalamayı sağlamaktadır. Özetleme işlemi sonucunda, öznitelik haritalarının boyutu küçülmekte ve özniteliklerin özetlenmiş temsilleri elde edilmektedir.
4. Boyut Azaltma: Özetleme işlemiyle birlikte boyut azaltma sağlanmaktadır. Bu, daha az parametre ve hesaplama gerektiren bir yapı oluştururken önemli özelliklerin de korunmasını sağlamaktadır.

## 2.7. Dönüştürücü (Transformer) Modeller

Dönüştürücü modeller, özellikle NLP alanında kullanılan bir makine öğrenmesi modelidir. Bu modeller, dil modellerini eğitmek ve metinleri işlemek için özel olarak tasarlanmıştır. Geleneksel RNN ve ConNN tabanlı modellerin yerine geliştirilen dönüştürücü modeli, dil işleme görevlerinde yüksek performans sağlamaktadır.

Dönüştürücü modeller, metindeki kelimelerin birbirleriyle olan ilişkisini hesaplamak için "dikkat (attention)" mekanizmasını kullanmaktadır. Bu mekanizma, her kelimenin metindeki diğer kelimelerle olan bağlantısını belirlemekte ve dikkate alınması gereken kelimeleri ağırlıklarla hesaplamaktadır. Bu sayede, metindeki bağımlılıklar ve ilişkiler daha iyi yakalanmaktadır.



Şekil 2.9. Dikkat Mekanizması (Uçar, 2022)

Dönüştürücü modeller Şekil 2.9'da gösterilen dikkat mekanizmasına göre, kodlayıcı ve çözücü adı verilen iki bileşenden oluşmaktadır. Kodlayıcı, girdi metnini bir dizi vektör

temsiline dönüştürmektedir. Bu vektör temsili, her bir kelimenin anlamını ve bağlamını yakalamak için kullanılmaktadır. Çözücü ise vektör temsilini kullanarak metni çıktıya dönüştürmektedir.

Dil modelleri, metin çevirisi, metin sınıflandırma, metin üretimi gibi birçok NLP görevinde başarılı bir şekilde kullanılmaktadır. Paralel hesaplamaları daha etkili bir şekilde kullanması ve dikkat mekanizmasının avantajları sayesinde diğer derin öğrenme modellerine kıyasla daha iyi sonuçlar elde etmektedir. Ayrıca, uzun metinlerdeki bağımlılıkları daha iyi yakalayabilme yeteneği ile bilinmektedir.

Dönüştürücü modeller, 2017 yılında yayımlanan "Attention is All You Need" başlıklı makalede tanıtılmış olup, o zamandan beri NLP alanında birçok uygulamada kullanılmaktadır (Vaswani ve diğ., 2023).

Aşağıda, bazı önemli dönüştürücü model çeşitleri verilmiştir:

**Transformer:** Bu, temel dönüştürücü modelinin orijinal versiyonudur. Kodlayıcı ve çözücü bloklarından oluşmakta ve dil işleme görevlerinde etkili bir şekilde kullanılmaktadır.

**BERT:** Çift yönlü kodlayıcı temsillerinden oluşmaktadır. Bu model, geniş bir dil öğrenme süreciyle eğitilmekte ve kelime düzeyindeki bağlamları yakalama yeteneğiyle tanınmaktadır. Özellikle doğal dil işleme görevlerinde yaygın olarak kullanılmaktadır.

**GPT:** Geniş bir metin veri kümesi kullanılarak önceden eğitilmekte ve ardından metin üretme ve dil anlama gibi görevlerde kullanılabilir.

**XLNet:** Sıralı dil modellemeyi kullanarak önceden eğitilmekte ve farklı dil işleme görevlerinde kullanılabilir. Özellikle sıralı bilgiye dayalı bağımsızlık ve dikkat mekanizmasının geliştirilmesiyle bilinmektedir.

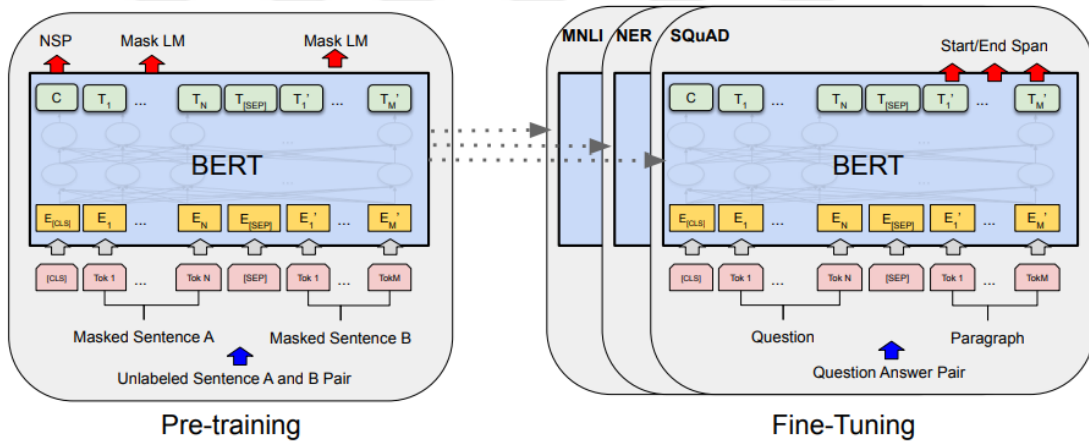
**RoBERTa:** RoBERTa, BERT modeline dayalı olarak geliştirilen bir dil modelidir. Daha geniş bir öğrenme veri kümesi ve gelişmiş bir optimizasyon yöntemiyle daha iyi performans sağlayabilmektedir.

T5 (Text-To-Text Transfer Transformer, Metinden Metine Aktarma Dönüştürücüsü): Metin tabanlı görevler için kullanılabilen bir dönüştürücü modelidir. Metni girdi olarak metin tabanlı görevlerde çıktı üretmektedir.

Bu modeller, dönüştürücü modellerin çeşitlerinden bazılarıdır. Her biri farklı yapı ve özelliklere sahiptir. Derin öğrenme ve doğal dil işleme alanında sürekli olarak yeni dönüştürücü model çeşitleri ve iyileştirmeleri geliştirilmektedir.

### 2.7.1. BERT

BERT, doğal dil işleme alanında duygu analizi gibi görevlerde kullanılan etkili bir modeldir. Bu model, Dönüştürücü modele dayanarak kelime seviyesinde anlamı yakalama ve kelime ilişkilerini anlama konusunda geliştirilmiştir.



Şekil 2.10. BERT Modeli (Devlin ve diğ., 2019)

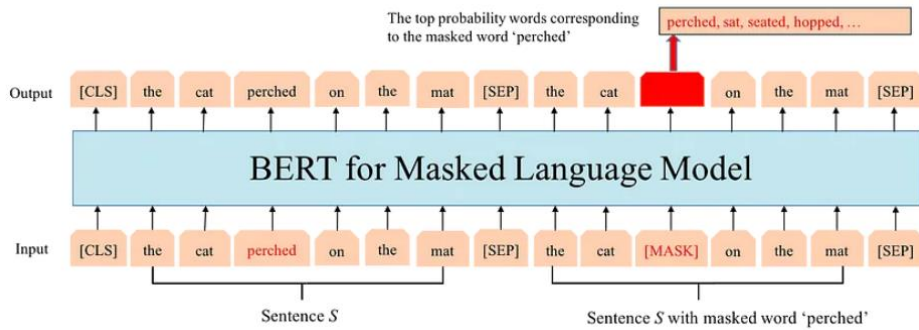
BERT modelinin işleyişi aşağıdaki adımlarla açıklanabilir:

1. Ayırma: Türkçe için önceden eğitilmiş Bert modeline uygun girişlerin elde edilmesini sağlamaktadır. Metinleri küçük parçalara ayırarak modelin anlayabileceği sayısal temsilcilere dönüştürmek için kullanılmaktadır.
2. Önceden Eğitim: BERT, büyük bir metin veri kümesi üzerinde önceden eğitilmektedir. Maskeli Dil Modelleme (Masked Language Modeling, MLM) ve Sonraki Cümle Tahmini (Next Sentence Prediction, NSP) adı verilen iki teknikle eğitim sağlanmaktadır. Maskeli Dil Modellemede cümle içerisindeki kelimeler

arasındaki ilişki üzerinde durulurken, Sonraki Cümle Tahmininde cümleler arasındaki ilişki kurulmaktadır. Şekil 2.10'da Önceden Eğitim (Pre-Training) verilmiştir.

3. BERT Modeli: BERT modeli, Dönüştürücü mimarisini kullanmaktadır. Bu mimari, birçok öz-dikkat mekanizması ve katmanlardan oluşmaktadır. Öz-dikkat mekanizması, her kelimenin diğer kelimelerle olan ilişkisini dikkate alarak anlamı daha iyi kavramak için kullanılmaktadır. Bu sayede, BERT, metindeki kelime ilişkilerini daha iyi anlayabilmekte ve kelime seviyesinde anlamı yakalayabilmektedir.
4. Düzenleme: Duygu analizi için, BERT modeli, girdi metnini belirli bir biçime dönüştürmektedir. Metinlerin sayısal vektörlerini içeren bir liste ve etiketlerini içeren ayrı bir liste oluşturmakta ve bu listeleri birleştirerek yeni bir veri kümesi oluşturmaktadır.
5. İnce Ayar: Önceden eğitilmiş BERT modeli, duygu analizi gibi belirli bir görev için ince ayar yapmak üzere kullanılmaktadır. Bu aşamada, belirli bir veri kümesiyle BERT modeli yeniden eğitilmektedir. İnce ayar sürecinde, model, veri kümesindeki örnekleri kullanarak belirli bir duyguyu sınıflandırmak için öğrenme yapmaktadır. İnce ayar sayesinde model, hangi etiketi tahmin etmesi gerektiğini duygu analizi yaparken belirleyebilmektedir. Şekil 2.10'da İnce Ayar (Fine-Tuning) verilmiştir.

Maskeli Dil Modelleme (Devlin ve diğ., 2019): Veri kümesindeki cümleler, BERT'e giriş olarak verilmeden önce, her cümlenin 15%'indeki kelimeler rastgele seçilerek [MASK] belirteci ile değiştirilir. Ardından model, maskelenmiş kelimelerin orijinal değerini tahmin etmek üzere, bağlamın maskelenmemiş diğer kelimeleri tarafından sağlanan bilgileri kullanarak öğrenmeye çalışır. Şekil 2.11'de gösterilmiştir.



Şekil 2.11. Maskeli Dil Modelleme (Haliloglu, 2020)

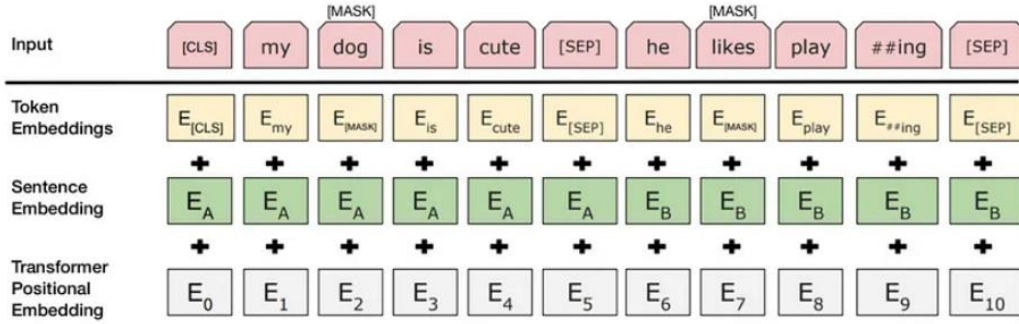
Maskeli dil modellemeye göre her kelimenin olasılığının hesaplanması Denklem (2.11)'deki gibi;

$$P(y_i|x) = \text{Softmax}(\hat{y}_i \cdot E^T) \quad (2.11)$$

şeklinde olacaktır.

Kodlayıcıdan çıkan  $\hat{y}_i$  çıktı vektörleri ile gömme vektörleri ile çarpılarak kelime boyutuna dönüştürülür. Ardından Softmax aktivasyon fonksiyonu ile her kelimenin olasılığı hesaplanır.

Sonraki Cümle Tahmini: Eğitim sırasında, BERT, iki cümleyi özel bir [SEP] simgesi ile ayırarak alır. Eğitim örneklerinde ikinci cümlelerin yarısı, ilk cümlelerin ardından geldiği için yarısı rastgele seçilmiş bir cümleden oluşur. BERT, daha sonra ikinci cümlelerin rastgele seçilen bir cümle olup olmadığını tahmin etmek üzere eğitilir.



Şekil 2.12. Sonraki Cümle Tahmini (Devlin ve diğ., 2019)

Öz-Dikkat Mekanizması: Aynı cümle içindeki tüm kelimelerin özelliklerinin ağırlıklı toplamını hesaplamaktadır. Girişe sorgular için Q, anahtarlar için K, değerler için V ve anahtarların boyutu olan  $d_k$  değeri verildiğinde, ölçeklendirilmiş nokta çarpımı dikkati Denklem (2.12)'deki gibi;

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.12)$$

şeklinde olacaktır (Park & Ahn, 2019).

BERT modeli, özellikle duygu analizi gibi doğal dil işleme konusunda etkili bir şekilde kullanılabilir. Önceden eğitilmiş olması, büyük bir veri kümesindeki dil yapılarını ve anlam ilişkilerini öğrenmesini sağlamaktadır. İnce ayar süreciyle belirli bir görev için özelleştirildikten sonra, BERT, daha küçük bir etiketli veri kümesi üzerinde duygu analizi yapabilmektedir.

BERT, kelime seviyesinde anlamı yakalama yeteneği ve önceden eğitimle kazandığı dilbilgisi temelindedir. Bu sayede, doğal dildeki kelimelerin anlamını daha iyi kavrayabilmekte ve duygu analizinde yüksek performans sağlayabilmektedir.

## **2.8. Optimizasyon Algoritmaları**

Optimizasyon kavramı, belirli sınırlara ve parametrelere tabi tutulan bir kayıp/maliyet fonksiyonunun en üst düzeye veya alt düzeye indirgenmesi anlamına gelmektedir.

### **2.8.1. Gradyan İnişi**

Sinir ağı eğitimi, temelde konveks olmayan bir optimizasyon problemi olan geri yayılım ve gradyan iniş algoritması yoluyla kaybı optimize etme sürecidir (Wang ve diğ., 2022). Gradyan inişi, sinir ağlarında optimizasyon gerçekleştirmek için en popüler algoritmalarından biridir. Gradyan inişinin, hedef fonksiyonun gradyanının hesaplanması için kullanılan veri miktarına bağlı olarak değişiklik gösteren üç çeşidi bulunmaktadır: Stokastik gradyan inişi, Toplu gradyan inişi, Mini toplu gradyan inişi. (Ruder, 2017).

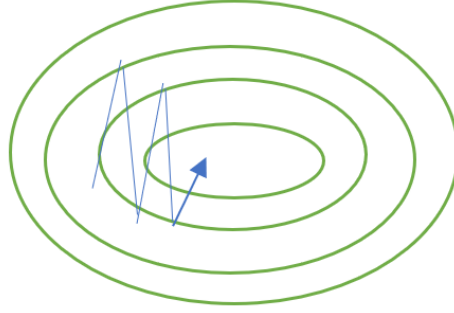
### **2.8.2. Stokastik Gradyan İnişi**

Stokastik gradyan iniş yöntemi, sinir ağlarının eğitiminde kayıp fonksiyonunu optimize etmek için yaygın olarak kullanılmaktadır. Ağ parametrelerini bir seferde mini toplu örneklerle güncellemektedir. Bu güncelleme hesaplama karmaşıklığını azaltmakta ve Toplu Gradyan İnişi ile karşılaştırıldığında genelleme performansını artırmaktadır (Wang ve diğ., 2022).

### **2.8.3. Momentum**

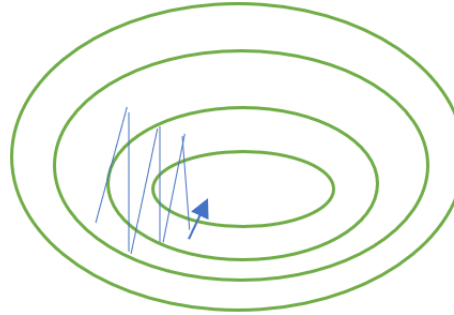
Momentum, Stokastik Gradyan İniş yöntemini hızlandıran ve salınımları azaltma yönünde etki gösteren bir yöntem olarak önerilmiştir (Qian, 1999).

Şekil 2.13 ve Şekil 2.14'te salınımlar temsili olarak gösterilmiştir.



Şekil 2.13. Momentum Uygulanmamış Stokastik Gradyan İnişi (Sutton, 1986)

Stokastik gradyan inişinin, yerel optimuma doğru yalnızca dipte ve tereddütlü bir ilerleme sağladığı Şekil 2.13'te gösterilmektedir.



Şekil 2.14. Momentum Uygulanmış Stokastik Gradyan İnişi (Sutton, 1986)

Momentumun ilgili yönde hızlandırmaya yardımcı olan ve salınımları azaltan bir yöntem olduğu Şekil 2.14'te görülmektedir.

#### 2.8.4. Adagrad

AdaGrad gibi Uyarlanabilir Stokastik Gradyan İniş yöntemleri son birkaç yılda yaygın olarak kullanılmaktadır. Her adımda her parametre için değişen öğrenme katsayıları kullanılarak, her bir parametrenin güncellenmesi farklı bir şekilde gerçekleştirilmektedir. Bu sayede, öğrenme katsayısının manuel olarak ayarlama gerekliliği ortadan kalkmaktadır. Adagrad algoritmasında, her bir parametrenin kendine özgü bir öğrenme hızı bulunmaktadır ve güncelleme işlemi sırasında öğrenme katsayısı değerinin bölüldüğü terimin zamanla artmaya devam etmesi nedeniyle öğrenme katsayısı aşırı derecede küçülmektedir (Çarkacı, 2020).

### 2.8.5. RMSProp

Adagrad algoritmasındaki öğrenme katsayısının aşırı küçültme sorununa bir çözüm yolu olarak geliştirilmiştir. Geçmiş gradyanların karelerinin tamamını kullanmak yerine, değer miktarını belli bir çerçeve boyutu ile kısıtlamaktadır (Ruder, 2017).

### 2.8.6. Adam

Adam algoritması her parametre için uyarlanabilir öğrenme hızlarını hesaplayan başka bir yöntemdir. Bu yöntemde, Rmsprop yönteminde olduğu gibi, önceki gradyan karelerinin üssel olarak ağırlıklandırılmış ortalamaları ( $v_t$ ) kullanılırken, aynı zamanda momentum değişiklikleri ( $m_t$ ) de önbellekte saklanmaktadır. ( $v_t$ ) ve ( $m_t$ ) Denklem (2.13) ile Denklem (2.14)'teki gibi;

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (2.13)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (2.14)$$

şeklinde olacaktır.

Gradyanlar hesaplanırken, Denklem (2.13)'te gradyanların kendisinin, Denklem (2.14)'te önceki gradyanların karelerinin dikkate alındığı görülmektedir (Özkaya, 2020).

Denklem (2.15)'te verilen  $\beta_1$  Katsayısı, önceki gradyanın hesaplanmasında gradyanın kendi büyüklüğüne olan etkisini belirtmektedir. Denklem (2.16)'da verilen  $\beta_2$  Katsayısı ise önceki gradyanın hesaplanmasında gradyanın karesinin etkisini belirtmektedir.

$$m'_t = \frac{m_t}{(1-\beta_1)^t} \quad (2.15)$$

$$v'_t = \frac{v_t}{(1-\beta_2)^t} \quad (2.16)$$

Adam güncelleme kuralı Denklem (2.17)'deki gibi;

$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{v_t + \epsilon}} m'_t \quad (2.17)$$

şeklinde olacaktır.

## 2.9. Aktivasyon Fonksiyonları

Aktivasyon fonksiyonları olmadan bir Sinir Ağı, genellikle sınırlı performans ve kapasite ile temelde bir Doğrusal Regresyon Modeli gibi işlev göstermektedir. Sinir ağlarının rolü yalnızca doğrusal fonksiyonları hesaplamakla sınırlı olmayıp, aynı zamanda görüntülerden videolara, seslerden konuşmalara ve metinlere kadar çeşitli karmaşık veri türlerini modellemek gibi daha sofistike görevleri başarıyla gerçekleştirmesi beklenmektedir. (Sharma ve diğ., 2020).

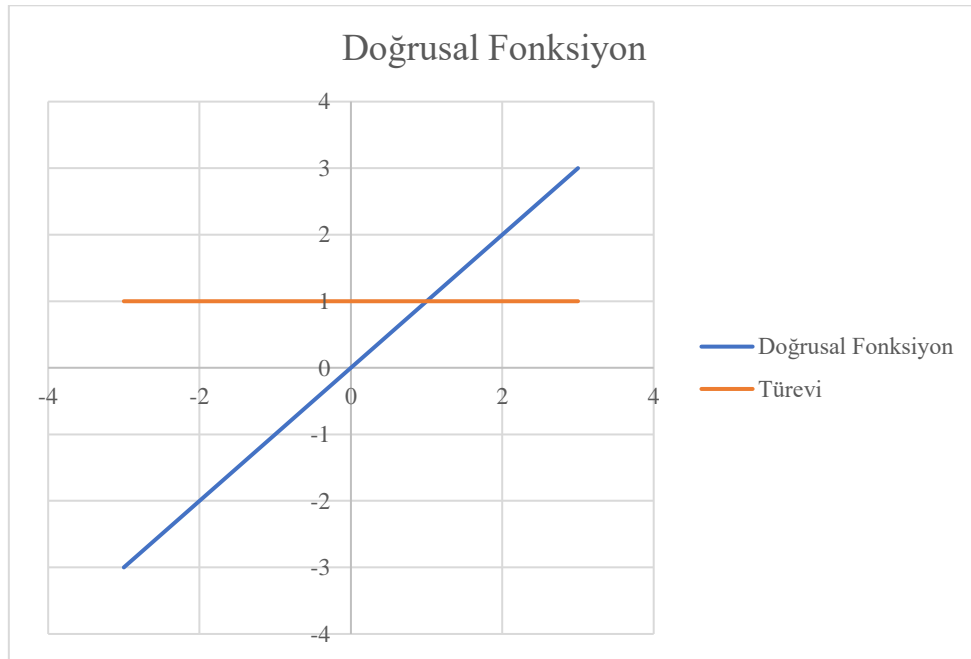
### 2.9.1. Doğrusal Fonksiyon

Doğrusal aktivasyon fonksiyonu giriş ile doğru orantılıdır (Sharma ve diğ., 2020). Doğrusal fonksiyon formülü Denklem (2.18)'deki gibi;

$$f(x) = x \quad (2.18)$$

şeklinde olacaktır.

Burada  $f(x)$  fonksiyonunu kullanılan sabitin değerine eşittir. Gradyan  $x$  giriş değerinden bağımsız sabit bir değerdir, bu da güncelleme faktörü aynı olmasına rağmen ağırlıkların ve önyargıların geri yayılım adımı sırasında güncelleneceği anlamına gelmektedir.



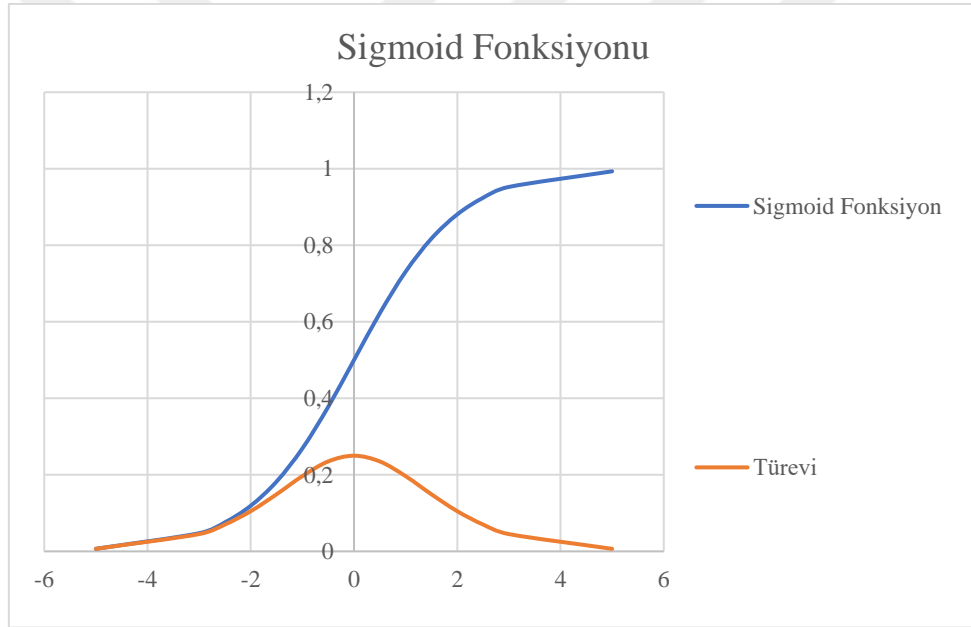
Şekil 2.15. Doğrusal Fonksiyon Grafiği

Şekil 2.15'te fonksiyon grafiği verilmiştir.

Doğrusal fonksiyon kullanmanın pek bir faydası yoktur çünkü sinir ağı her iterasyon için aynı gradyan değeri nedeniyle hatayı iyileştirmeyecektir. Ayrıca, ağ verilerden karmaşık örüntüleri tanımlayamayacaktır. Bu nedenle, doğrusal fonksiyonlar yorumlanabilirliğin gerekli olduğu durumlarda ve basit görevler için idealdir (Sharma ve diğ., 2020).

### 2.9.2. Sigmoid Fonksiyonu

Sigmoid Fonksiyonu pozitif negatif gibi ikili sınıflandırmalarda kullanılmaktadır. Gelen girdi değerini (0,1) aralığında sıkıştırılmaktadır (Apicella ve diğ., 2021). Sigmoid fonksiyonu türevlenebilir olduğu için kullanımında öğrenme sağlanacaktır.



Şekil 2.16. Sigmoid Fonksiyonu Grafiği

Şekil 2.16 incelendiğinde fonksiyonda x değeri arttıkça y değerindeki değişiklik azalmaktadır. Buda türev değerlerinin çok küçük olduğunu ve 0'a yakınsandığını göstermektedir. Bu problem kaybolan gradyan olarak adlandırılmaktadır. Öğrenme olayının minimum düzeyde gerçekleşmesine sebep olmaktadır.

Sigmoid fonksiyonu formülü Denklem (2.19)'daki gibi;

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad (2.19)$$

şeklinde olacaktır.

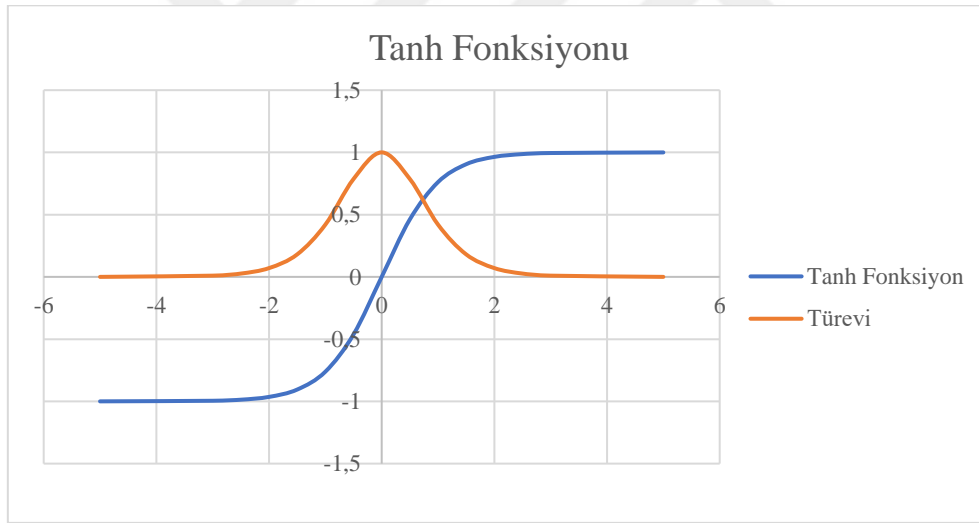
### 2.9.3. Tanh Fonksiyonu

Sigmoid fonksiyonuna benzemektedir, ancak sigmoid fonksiyonu (0,1) aralığında çıktı üretirken Tanh fonksiyonu (-1,1) aralığında çıktı üretmektedir (Nwankpa ve diğ., 2018). Önceki katmanlardan gelen çıktılar, farklı işaretler aracılığıyla belirginleşerek bir sonraki katmana iletilir, bu da farklı sonuçların ortaya çıkmasını sağlamaktadır (Sharma ve diğ., 2020).

Tanh fonksiyonu formülü Denklem (2.20)'deki gibi;

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.20)$$

şeklinde olacaktır.



Şekil 2.17. Tanh Fonksiyonu Grafiği

Şekil 2.17'de verilen tanh fonksiyonu grafiği incelendiğinde, yüksek girdilerde çıktı 1'e yaklaşırken düşük girdilerde çıktı -1'e yaklaşmaktadır.

Tanh fonksiyonu, çok katmanlı sinir ağları için daha iyi eğitim performansı sağlaması nedeniyle sigmoid fonksiyonuna kıyasla tercih edilen fonksiyon olmaktadır. Bununla birlikte, tanh fonksiyonu sigmoid fonksiyonların yaşadığı kaybolan gradyan sorununu da çözememiştir. Fonksiyonun sağladığı temel avantaj, sıfır merkezli çıktı üretmesi ve böylece geri yayılım sürecine yardımcı olmasıdır (Nwankpa ve diğ., 2018).

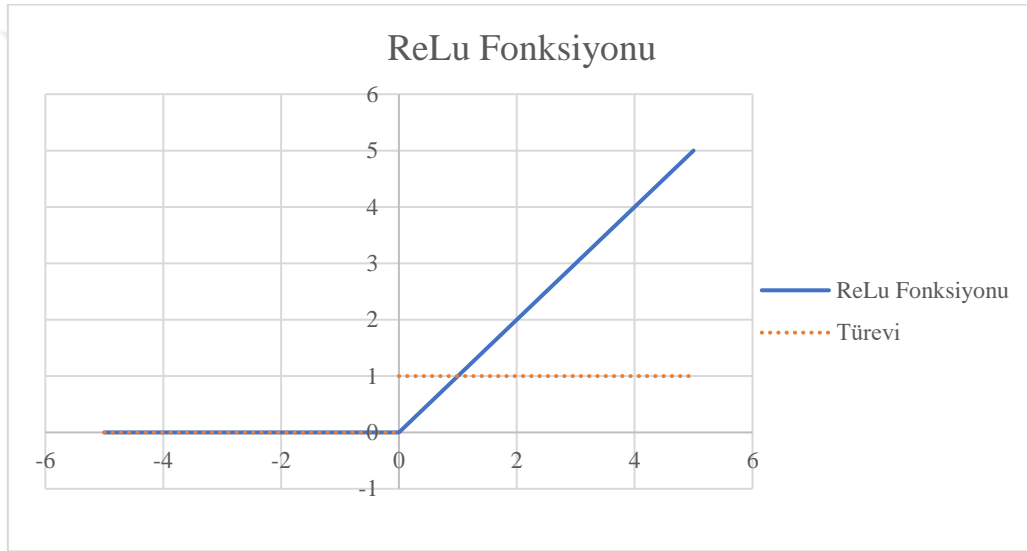
#### 2.9.4. ReLu Fonksiyonu

ReLu fonksiyonu son birkaç yılda çok popüler hale gelmiştir, yapay sinir ağlarının gizli katmanlarında, özellikle de Evrişimli Sinir Ağlarında veya Derin Öğrenmede uygulanmaktadır.

ReLu fonksiyonu formülü Denklem (2.21)'deki gibi;

$$f(x) = \max(x, 0) \quad (2.21)$$

şeklinde olacaktır (Ghorbani ve diğ., 2020).



Şekil 2.18. ReLu Fonksiyonu Grafiği

Şekil 2.18'de görüldüğü gibi doğrusal gibi görünmektedir ancak aslında doğrusal değildir. Çünkü x, 0'dan küçük olduğunda çıkış 0'dır ve kaybolan gradyan problemini önlemekte ve düzeltmektedir. Sigmoid ve Tanh'a göre hesaplama açısından daha az maliyetlidir (Szandała, 2021).

#### 2.9.5. Softmax Fonksiyonu

Softmax fonksiyonu, derin öğrenme sınıflandırma işlemlerinde yaygın kullanılmaktadır (Banerjee ve diğ., 2020). Gerçek sayılardan oluşan bir vektörden olasılık dağılımını hesaplamak için kullanılmaktadır. 0 ile 1 aralığında bir değer olacak şekilde olasılıkların toplamının 1'e eşit olduğu çıktı üretmektedir. Bu sebeple çok etiketli sınıflandırma

işlevlerinde tercih edilmektedir (Nwankpa ve diğ., 2018). Softmax fonksiyonu formülü Denklem (2.22)'deki gibi;

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad \text{for } j = 1, 2, \dots, K. \quad (2.22)$$

şeklinde olacaktır (Dunne & Campbell, 1997).

## 2.10. Değerlendirme Metrikleri

Sınıflandırma modellerinin değerlendirme metrikleri olarak doğruluk, duyarlılık, dengeli doğruluk, kesinlik ve f-ölçütü gibi metrikler kullanılmaktadır. Ancak kesinlik, duyarlılık, f-ölçütü gibi metrikler pozitif ve negatif gibi ikili sınıflandırmalar için kullanılmaktadır. Çok sınıflı sınıflandırma işlemlerinde doğruluk, dengeli doğruluk gibi metrikler kullanılmaktadır. Bu metriklerin değerlendirilmesi için Tablo 2.1'deki karmaşıklık Matrisi kullanılmaktadır (Grandini ve diğ., 2020).

Tablo 2.1. Karmaşıklık Matrisi

	Tahmin Edilen Sınıf						Toplam
	Sınıflar	A	B	C	D	E	
Gerçek Sınıf	A	D <sub>A</sub>	Y <sub>AB</sub>	Y <sub>AC</sub>	Y <sub>AD</sub>	Y <sub>AE</sub>	A <sub>sat</sub>
	B	Y <sub>BA</sub>	D <sub>B</sub>	Y <sub>BC</sub>	Y <sub>BD</sub>	Y <sub>BE</sub>	B <sub>sat</sub>
	C	Y <sub>CA</sub>	Y <sub>CB</sub>	D <sub>C</sub>	Y <sub>CD</sub>	Y <sub>CE</sub>	C <sub>sat</sub>
	D	Y <sub>DA</sub>	Y <sub>DB</sub>	Y <sub>DC</sub>	D <sub>D</sub>	Y <sub>DE</sub>	D <sub>sat</sub>
	E	Y <sub>EA</sub>	Y <sub>EB</sub>	Y <sub>EC</sub>	Y <sub>ED</sub>	D <sub>E</sub>	E <sub>sat</sub>
Toplam	A <sub>sut</sub>	B <sub>sut</sub>	C <sub>sut</sub>	D <sub>sut</sub>	E <sub>sut</sub>	Toplam	

Tablo 2.1'de verilen karmaşıklık Matrisinin ifadeler şu şekildedir:

- D<sub>A, B, C, D, E</sub>: A, B, C, D ve E sınıflarında doğru tahmin edilen veri sayısı.
- Y<sub>AB</sub>: Gerçekte A sınıfına ait olan verinin B sınıfı olarak tahmin edildiği veri sayısı.
- Y<sub>AC</sub>: Gerçekte A sınıfına ait olan verinin C sınıfı olarak tahmin edildiği veri sayısı.

- $Y_{AD}$ : Gerçekte A sınıfına ait olan verinin D sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{AE}$ : Gerçekte A sınıfına ait olan verinin E sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{BA}$ : Gerçekte B sınıfına ait olan verinin A sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{BC}$ : Gerçekte B sınıfına ait olan verinin C sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{BD}$ : Gerçekte B sınıfına ait olan verinin D sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{BE}$ : Gerçekte B sınıfına ait olan verinin E sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{CA}$ : Gerçekte C sınıfına ait olan verinin A sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{CB}$ : Gerçekte C sınıfına ait olan verinin B sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{CD}$ : Gerçekte C sınıfına ait olan verinin D sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{CE}$ : Gerçekte C sınıfına ait olan verinin E sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{DA}$ : Gerçekte D sınıfına ait olan verinin A sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{DB}$ : Gerçekte D sınıfına ait olan verinin B sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{DC}$ : Gerçekte D sınıfına ait olan verinin C sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{DE}$ : Gerçekte D sınıfına ait olan verinin E sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{EA}$ : Gerçekte E sınıfına ait olan verinin A sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{EB}$ : Gerçekte E sınıfına ait olan verinin B sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{EC}$ : Gerçekte E sınıfına ait olan verinin C sınıfı olarak tahmin edildiği veri sayısı.
- $Y_{ED}$ : Gerçekte E sınıfına ait olan verinin D sınıfı olarak tahmin edildiği veri sayısı.
- $A_{sat}$ : A sınıfının verileri için doğru ve yanlış tahmin edilen verilerin toplam sayısını vermektedir.  $A_{sat}$  formülü Denklem (2.23)'teki gibi;

$$A_{sat} = D_A + Y_{AB} + Y_{AC} + Y_{AD} + Y_{AE} \quad (2.23)$$

şeklinde olacaktır.

- $B_{sat}$ : B sınıfının verileri için doğru ve yanlış tahmin edilen verilerin toplam sayısını vermektedir.  $B_{sat}$  formülü Denklem (2.24)'teki gibi;

$$B_{sat} = Y_{BA} + D_B + Y_{BC} + Y_{BD} + Y_{BE} \quad (2.24)$$

şeklinde olacaktır.

- $C_{sat}$ : C sınıfının verileri için doğru ve yanlış tahmin edilen verilerin toplam sayısını vermektedir.  $C_{sat}$  formülü Denklem (2.25)'teki gibi;

$$C_{sat} = Y_{CA} + Y_{CB} + D_C + Y_{CD} + Y_{CE} \quad (2.25)$$

şeklinde olacaktır.

- $D_{sat}$ : D sınıfının verileri için doğru ve yanlış tahmin edilen verilerin toplam sayısını vermektedir.  $D_{sat}$  formülü Denklem (2.26)'daki gibi;

$$D_{sat} = Y_{DA} + Y_{DB} + Y_{DC} + D_D + Y_{DE} \quad (2.26)$$

şeklinde olacaktır.

- $E_{sat}$ : E sınıfının verileri için doğru ve yanlış tahmin edilen verilerin toplam sayısını vermektedir.  $E_{sat}$  formülü Denklem (2.27)'deki gibi;

$$E_{sat} = Y_{EA} + Y_{EB} + Y_{EC} + Y_{ED} + D_E \quad (2.27)$$

şeklinde olacaktır.

- $A_{sut}$ : A sınıfı olarak tahmin edilen verilerin toplam sayısını vermektedir.  $A_{sut}$  formülü Denklem (2.28)'deki gibi;

$$A_{sut} = D_A + Y_{BA} + Y_{CA} + Y_{DA} + Y_{EA} \quad (2.28)$$

şeklinde olacaktır.

- $B_{sut}$ : B sınıfı olarak tahmin edilen verilerin toplam sayısını vermektedir.  $B_{sut}$  formülü Denklem (2.29)'daki gibi;

$$B_{sut} = Y_{AB} + D_B + Y_{CB} + Y_{DB} + Y_{EB} \quad (2.29)$$

şeklinde olacaktır.

- $C_{sut}$ : C sınıfı olarak tahmin edilen verilerin toplam sayısını vermektedir.  $C_{sut}$  formülü Denklem (2.30)'daki gibi;

$$C_{sut} = Y_{AC} + Y_{BC} + D_C + Y_{DC} + Y_{EC} \quad (2.30)$$

şeklinde olacaktır.

- $D_{sut}$ : D sınıfı olarak tahmin edilen verilerin toplam sayısını vermektedir.  $D_{sut}$  formülü Denklem (2.31)'deki gibi;

$$D_{sut} = Y_{AD} + Y_{BD} + Y_{CD} + D_D + Y_{ED} \quad (2.31)$$

şeklinde olacaktır.

- $E_{sut}$ : E sınıfı olarak tahmin edilen verilerin toplam sayısını vermektedir.  $E_{sut}$  formülü Denklem (2.32)'deki gibi;

$$E_{sut} = Y_{AE} + Y_{BE} + Y_{CE} + Y_{DE} + D_E \quad (2.32)$$

şeklinde olacaktır.

- Toplam: A, B, C, D, E sınıflarına ait doğru ve yanlış tahmin edilen bütün veri sayılarının toplamıdır. Veri kümesinin 2 katını vermektedir. Toplam formülü Denklem (2.33)'teki gibi;

$$Toplam = A_{sat} + B_{sat} + C_{sat} + D_{sat} + E_{sat} + A_{sut} + B_{sut} + C_{sut} + D_{sut} + E_{sut} \quad (2.33)$$

şeklinde olacaktır.

### 2.10.1. Doğruluk

Doğruluk, çok sınıflı sınıflandırmada en popüler ölçütlerden biridir ve doğrudan karışıklık matrisinden hesaplanmaktadır. Doğruluk, modelin tüm veri kümesi üzerinde ne kadar doğru tahmin yaptığının genel ölçütünü vermektedir (Grandini ve diğ., 2020). Hesaplama Formülü Denklem (2.34)'teki gibi;

$$Doğruluk = \frac{D_A + D_B + D_C + D_D + D_E}{Toplam} \quad (2.34)$$

şeklinde olacaktır.

Doğruluk formülü, payda Gerçek sınıfların toplamını ve paydada karışıklık matrisinin tüm girişlerinin toplamını dikkate almaktadır.

Metrik çok sezgiseldir ve anlaşılması kolaydır. Doğruluk 0 ile 1 arasında değerler alırken, 1'e ulaşmak için eksik olan miktar Yanlış Sınıflandırma Oranı olarak adlandırılmaktadır (Grandini ve diğ., 2020).

### 2.10.2. Dengeli Doğruluk

Dengeli Doğruluk hem ikili hem de çok sınıflı sınıflandırmada iyi bilinen bir başka metriktir ve karışıklık matrisinden hesaplanmaktadır. Hesaplama Formülü Denklem (2.35)'teki gibi;

$$\text{Dengeli Doğruluk} = \frac{\frac{D_A}{A_{sat}} + \frac{D_B}{B_{sat}} + \frac{D_C}{C_{sat}} + \frac{D_D}{D_{sat}} + \frac{D_E}{E_{sat}}}{\text{Toplam Sınıf Sayısı}} \quad (2.35)$$

şeklinde olacaktır.

Dengeli Doğruluk formülü temelde geri çağırımların bir ortalamasıdır. Önce her bir sınıf için Geri Çağırma değeri hesaplanmaktadır. Ardından Dengeli Doğruluk değerini elde etmek için değerlerin ortalaması alınmaktadır. Her bir sınıf için Geri Çağırma değeri "o sınıftaki bir verinin doğru sınıflandırılma olasılığı nedir?" sorusuna cevap vermektedir. Dolayısıyla, Dengeli Doğruluk, farklı sınıflar arasında bu kavramın ortalama bir ölçüsünü sağlamaktadır (Grandini ve diğ., 2020).

### 2.11. Kayıp Fonksiyonları

Kayıp fonksiyonları, makine öğrenimi modelinin eğitim sürecinde gerçek etiketler ile modelin tahminleri arasındaki farkı ölçmektedir. Model, kayıp fonksiyonunu minimize etmek için iteratif olarak modelin ağırlıklarını güncellemekte ve gerçek sonuçları daha iyi tahmin etmeye çalışmaktadır. Sınıflandırma problemlerinde kategorik kayıp fonksiyonu, ikili çapraz kayıp fonksiyonu, çapraz kategorik kayıp fonksiyonu tercih edilen kayıp fonksiyonlarıdır.

#### 2.11.1. İkili Çapraz Kayıp Fonksiyonu

İkili çapraz kayıp fonksiyonu, tahmin değerinin 1 veya 0 olduğu özel bir çapraz kayıp fonksiyonudur. Derin sinir ağlarını kullanarak tahmin için genellikle sigmoid aktivasyonunu kullanmaktadır (Ruby & Yendapalli, 2020).

N sınıf için ikili çapraz kayıp fonksiyonu Denklem (2.36)'daki gibi;

$$L = -\frac{1}{n} [\sum_{i=1}^n [r_i \log(p_i) + (1 - r_i) \log(1 - p_i)]] \quad (2.36)$$

şeklinde olacaktır. Formülde r gerçek değerleri p ise tahmin değerlerini göstermektedir (Koech, 2022).

### 2.11.2. Çapraz Kategorik Kayıp Fonksiyonu

Çok kategorili sınıflandırma için kullanılmaktadır. Çok kategorili sınıflandırmada, sinir ağının işlenmemiş çıktıları softmax aktivasyonundan geçirilerek girdi sınıfları üzerinde tahmin edilen olasılıkların bir vektörünü çıkarmaktadır. Evrişimli Sinir Ağı her veri için çok sınıf üzerinden bir olasılık çıktısı verecek şekilde eğitilmektedir.

Kategorik Çapraz Kayıp fonksiyonu softmax Kaybı olarak da bilinmektedir. Çok kategorili sınıflandırmada, sinir ağının işlenmemiş çıktıları softmax aktivasyonundan geçirilerek girdi sınıfları üzerinde tahmin edilen olasılıkların bir vektörünü çıkarmaktadır. Tahmin edilen her sınıf olasılığı, 0 veya 1 olarak istenen gerçek sınıf çıktısıyla karşılaştırılmakta ve gerçek değerden sapmanın büyüklüğüne bağlı olarak, olasılık değerine ceza uygulayan bir kayıp hesaplanmaktadır. Bu logaritmik ceza, 1'e yakınsayan fazla farklar için büyük değer ve 0'a yaklaşan düşük farklar için düşük değer şeklinde uygulanmaktadır.

Kategorik çapraz kaybı, eğitim sırasında model ağırlıklarını ayarlarken kullanılmaktadır. Temel amaç kaybı minimize etmektir, bu nedenle kayıp değeri ne kadar küçük olursa, model o kadar başarılı kabul edilmektedir. Kusursuz bir modelin çapraz kaybı 0 olarak ifade edilmektedir. (Koech, 2022).

N sınıf için çapraz kategorik kayıp fonksiyonu Denklem (2.37)'deki gibi;

$$L_{CE} = -\sum_{i=1}^n r_i \log(p_i) \quad (2.37)$$

şeklinde olacaktır. Formülde t gerçek değerleri p ise tahmin değerlerini göstermektedir.

### 3. MATERYAL VE YÖNTEM

#### 3.1. Veri Kümesi

Veri toplama sürecinde, Türkçe Tweetler içeren etiketli bir Excel veri kümesi Kaggle platformu aracılığıyla temin edilmiştir. Şekil 3.1'de elde edilen veri kümesi gösterilmiştir.

	Tweet	Etiket
0	Allah'ım çıldıracağım. Yemin ederim çıldıracağ...	kızgın
1	site ne zaman çalıştıda ürün stokları bitti di...	kızgın
2	Sebebi neydi ki diye bağıracağım şimdi az kaldı	kızgın
3	Black friday ne amk bu yıl çıkardınız bu bokud...	kızgın
4	kısa kısa cevaplar vericeksen niye yazıyorsun ...	kızgın
...	...	...
3995	Aynı şeyleri yapmaktan sıkıldım	üzgün
3996	Olacağı varsa oluyor ya hani, biteceği varsa d...	üzgün
3997	Çok şey istiyorum ama sonu hep fiyasko MUTSUZUM	üzgün
3998	Garip bir şekilde canım çok yanıyor moralim bozuk	üzgün
3999	Üzgünüm birazdan takipten çıkarıp hesabımı giz...	üzgün

[4000 rows x 2 columns]

Şekil 3.1. Türkçe Tweet Veri Kümesi (Güven, 2020)

Toplamda 4000 adet örneğe sahip bu veri kümesi, "mutlu", "kızgın", "korku", "şaşkın" ve "üzgün" olmak üzere beş duygu etiketini içermektedir.

Türkçe tweet veri kümesi 80% eğitim ve 20% test verisi olarak bölünmüştür. Eğitim veri kümesi içerisinde 20%'lik doğrulama verisi ayrılmıştır.

Türkçe mağaza yorumlarını içeren etiketli ikinci veri kümesi Kaggle platformu aracılığıyla temin edilmiştir.

Toplamda 11429 adet örneğe sahip "olumlu", "olumsuz" ve "tarafsız" olmak üzere üç duygu etiketini içeren veri kümesi Şekil 3.2'de gösterilmiştir. Ancak veri kümesi incelendiğinde tarafsız etiketli verilerin düzgün etiket yapılmadığı, olumlu ve olumsuz etikete sahip olması gereken verinin tarafsız etiketlendiği sahip olduğu gözlenmiştir. Bu sebeple tarafsız veri kümesi veriden çıkarılarak çalışma yapılmıştır.

	Görüş	Durum
5291	Cihaz oldukça zarif içerisine bazı mağazalar s...	Tarafsız
3533	Ürün mükemmel tam istediğim gibi geldi kargo c...	Olumlu
10279	amazzon'un çalışma sisteminden çok memnun olma...	Olumsuz
6645	Bir kaç gün sergilenmiş kutusuna konulmuştur y...	Olumsuz
2320	Görseli güzel şık duruyor ama motor sesi yükse...	Tarafsız
...	...	...
3689	Ürün beklenen tarihten önce geldi...Ürün tüm ö...	Olumlu
2812	sayın tedarikçi kusura bakmayın ama yurt içi k...	Olumsuz
3682	Üzerine 1000 lira verip gt2 alırım daha iyi. A...	Olumsuz
7484	pazar aksami siparis verdim carsamba geldi bas...	Tarafsız
11197	teslim esnasında fatura verimedi bu birinci ek...	Olumsuz

[11415 rows x 2 columns]

Şekil 3.2. Türkçe Mağaza Yorumları Veri Kümesi (Bilen, 2020)

Şekil 3.3'te tarafsız etiket çıkarıldıktan sonraki veri kümesi verilmiştir.

	Görüş	Durum
0	Zamanında geldi sorunsuz şekilde teslim aldım...	Olumlu
4	7.5 aylık kullanımın ardından bu akşam parçala...	Olumsuz
5	Kargo çok hızlı ve sorunsuz geldi. Ertesi gün ...	Olumlu
6	ürün kaliteli tereddüt etmeden alabilirsiniz.	Olumlu
7	Ben bu ürünü 1 ay önce sipariş verdim ve hala ...	Olumsuz
...	...	...
11375	Sipariş 1 gün sonra teslim edildi. Fakat paket...	Olumsuz
11376	Fiyatı ve performansı olarak çok güzel bir ürü...	Olumlu
11378	1 yıldır kullanıyorum hiç sorun yaşamadım gaye...	Olumlu
11379	Kargo söylenen günde geldi. Getiren uğur yetki...	Olumlu
11383	Kesinlikle çok kullanışlı. pahalı markalara hi...	Olumlu

[8462 rows x 2 columns]

Şekil 3.3. Türkçe Mağaza Yorum Veri Kümesi (Olumlu, Olumsuz)

Türkçe mağaza yorumları veri kümesi 70% eğitim ve 30% test verisi olarak bölünmüştür. Eğitim veri kümesi içerisinde 30%'luk doğrulama verisi ayrılmıştır.

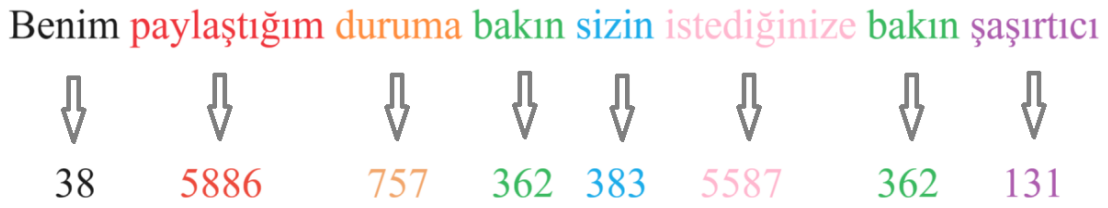
Eğitim aşamasında; LSTM, ConNN-LSTM hibrit derin öğrenme modelleri ve dönüştürücü model olan BERT modeli eğitilmiştir.

LSTM ve ConNN-LSTM hibrit derin öğrenme modellerine veri gönderilmeden önce metin verileri keras çerçevesinin fit\_on\_texts fonksiyonu ile Şekil 3.4'te gösterildiği gibi veri kümesinde kelime sözlüğü oluşturulmuştur.

da': 342, 'geldim': 343, 'fark': 344, '7': 345, 'insanların': 346, 'bazen': 347, 'gol': 348, 'buldum': 349, 'bozucu': 350, 'ciden': 351, 'şeye': 352, 'size': 353, 'iptal': 354, 'salak': 355, 'yüzden': 356, 'birini': 357, 'geçti': 358, 'aralar': 359, 'başladı': 360, 'bize': 361, 'bakın': 362, 'yorgunum': 363, 'beklemiyordum': 364, 'uyuyamıyorum': 365, 'bok': 366, 'erkek': 367, 'çabuk': 368, 'üzüldüm': 369, 'olsa': 370, 'müthiş': 371, 'herşey': 372, 'hem': 373, '20': 374, 'yanlış': 375, 'kaç': 376, 'olmaz': 377, 'gelen': 378, 'bizim': 379, 'dilerim': 380, 'inanmıyorum': 381, 'gitti': 382, 'sizin': 383, 'hangi': 384, 'babam': 385, 'para': 386, 'umarım': 387, 'sövüyorum': 388, 'arkadaş': 389, 'hepsi': 390, 'saatte': 391, 'diyor': 392, 'şarkı': 393, 'gelsin': 394, 'musun': 395, 'şeylere': 396, 'inanılmaz': 397, 'alıp': 398, 'dertlendim': 399, 'acayip': 400, 'film': 401, 'gözlerime': 402, 'inanamadım': 403, 'tekrar': 404, 'acıp': 405, 'geçiyor': 406, 'bulamıyorum': 407, 'görmek': 408, 'çıktım': 409, 'giden': 410, 'yalnız': 411, 'onu': 412, 'sizi': 413, 'olunca': 414, 'ona': 415, 'takip': 416, 'atmış': 417, 'kendini': 418, 'gidiyorum': 419, 'yakın': 420, 'göre': 421, 'uykum': 422, 'gibiymiş': 423, 'gidiyor': 424, 'uç': 425, 'neye': 426, 'nefes': 427, 'olduğu': 428, 'gözüm': 429, 'güne': 430, 'yerine': 431, 'açtım': 432, 'dolayı': 433, 'geç': 434, 'rüyamda': 435, 'hayretler': 436, 'mükemmel': 437, 'efsane': 438, 'sağolsun': 439, 'varmış': 440, 'aslında': 441, 'devam': 442, 'yoktur': 443, 'kızım': 444, 'dünyanın': 445, 'tamam': 446, 'gayet': 447, 'kabul': 448, 'edici': 449, 'çocuklar': 450, 'izledim': 451, 'ters': 452, 'izliyorum':

Şekil 3.4. Kelime Sözlüğü Oluşturulması

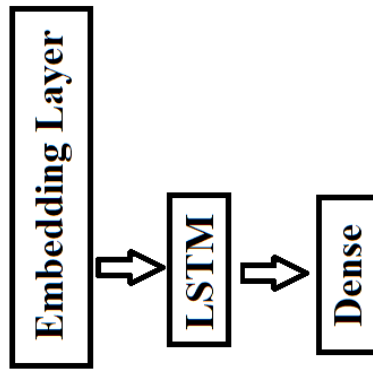
Oluşturulan kelime sözlüğü kullanılarak `texts_to_sequences` fonksiyonu ile her metin sayı dizisine dönüştürülmüştür. Şekil 3.5'te gösterilmiştir.



Şekil 3.5. Metinlerin Sayı Dizisine Dönüştürülmesi

### 3.2. LSTM Modeli

LSTM Modeli oluşturulurken üç katman tercih edilmiştir. Kullanılan Embedding (Gömme), LSTM ve Dense (Yoğun) katmanları Şekil 3.6 ile gösterilmiştir.



Şekil 3.6. LSTM Modeli Katmanları

Gömme katmanı her kelimeyi başlangıçta rastgele bir vektör olarak göstermektedir. Her kelimenin aldığı değer kelimelerin birbirleri ile ilişkisine göre belirlenmektedir. Eğitim sırasında güncellenerek birbiri ile ilişkili kelimeler birbirine yakın konumlandırılmakta ilişkisiz kelimelerin olduğu vektörler uzayda farklı konumlara aktarılmaktadır.

Gömme katmanının alığı parametreler şu şekildedir:

gömme\_katmanı(sözlük\_boyutu, çıktı\_boyutu, girdi\_boyutu).

Türkçe tweet veri kümesinde farklı toplam kelime sayısı hesaplanarak sözlük boyutu 11275 olarak bulunmuştur. Çıktı boyutu katmandan çıkan verinin boyutunun belirlendiği parametredir. Çıktı boyutu girdi boyutu ile aynı belirlenmiştir. 11275 tane kelimenin 21 boyutlu bir vektörle temsil edildiği anlamına gelmektedir. Her girdinin kelime uzunluğu hesaplanmış ve maksimum kelime uzunluğu 21 olarak bulunmuştur. Girdi boyutunun gömme katmanına sabit uzunlukta verilebilmesi için bütün girdiler ekleme (padding) yöntemi ile aynı boyuta getirilmiştir. Bu sebeple giriş boyutu 21 olarak belirlenmiştir.

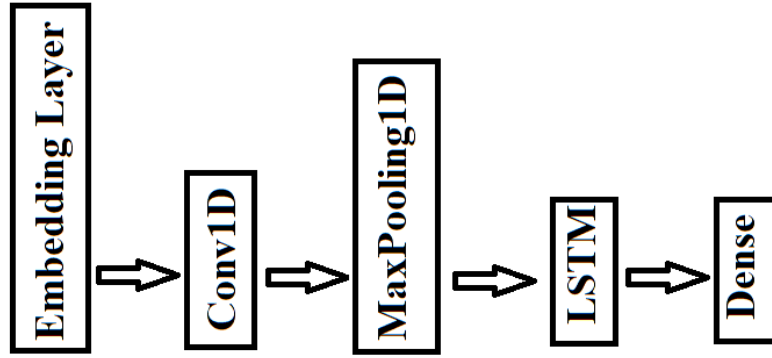
Türkçe mağaza yorumları veri kümesinde sözlük boyutu 15095, girdi boyutu 37 ve çıktı boyutuda girdi boyutu ile aynı belirlenmiştir.

LSTM katmanı, giriş bilgisinin hangisinin unutulacağına hangisinin tutulacağına karar vermektedir. Veriler arasındaki bağlantıların ve desenlerin öğrenilmesi için birim değeri yani LSTM hücre sayısı 128 olarak seçilmiştir. Boyutun büyüklüğünü indirgemek ve aşırı öğrenmeyi engellemek için her adımda ve katmandan sonra dropout kullanılmıştır.

Dense katmanı çıkış katmanıdır. Bu katman, LSTM katmanından aldığı çıktıyı aktivasyon fonksiyonuna sokarak çıktı üretmektedir. Genellikle LSTM modelinde sigmoid aktivasyon fonksiyonu kullanılmaktadır. Ancak sınıflandırmamız çok sınıflı olduğu için softmax aktivasyon fonksiyonu kullanılmıştır (Dunne & Campbell, 1997).

### **3.3. ConNN -LSTM Hibrit Modeli**

Bu çalışmada ConNN modeli ile LSTM modellerinin birleşimini kullanılarak zaman serisi ve ardışık veriler üzerinde daha etkili bir analiz yapılması hedeflenmiştir. Bu hibrit model, veri üzerinde ConNN'in evrişim ve havuzlama katmanlarından elde edilen özellik haritalarını LSTM'in hafıza yapısıyla birleştirmektedir. Böylece, hem LSTM'in uzun dönem bağımlılıkları yakalama yeteneğinden yararlanılmış, hem de ConNN'in özellik çıkarım gücü ile verideki önemli özellikler tanımlanmıştır. Şekil 3.7'de ConNN -LSTM hibrit modelinde kullanılan katmanlar gösterilmiştir.

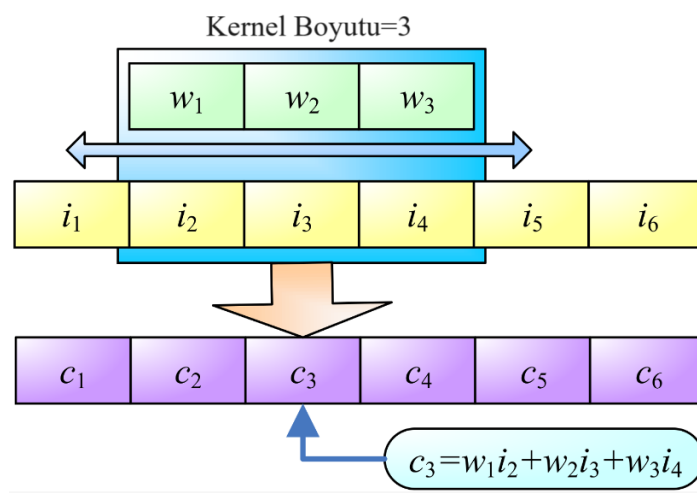


Şekil 3.7. ConNN -LSTM Hibrit Modeli Katmanları

Gömme katmanı LSTM modelinde olduğu gibi tüm elemanların boyutunu eşitleyerek verileri sayı vektörlerine dönüştürmektedir. Gömme katmanının parametreleri LSTM katmanı ile aynı belirlenmiştir.

Konvolüsyon katmanında veri kümesi metin olduğu için bir boyutlu evrişimi yalnızca bir boyutta gerçekleştiren Conv1D katmanı kullanılmıştır. Konvolüsyon katmanının aldığı parametreler şu şekildedir:

Conv1D(evrişim\_filtresi, kernel\_sayısı, aktivasyon\_fonksiyonu).



Şekil 3.8. Konvolüsyon Gösterimi (Kuo & Huang, 2018)

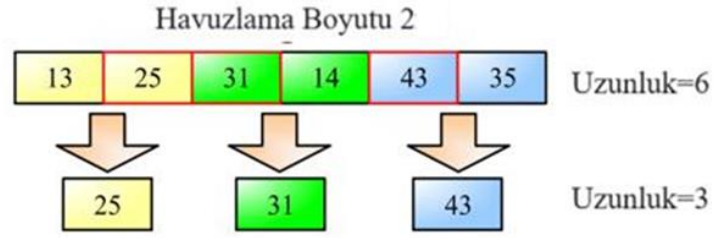
Şekil 3.8’da gösterilen Conv1D konvolüsyon katmanı örneğinde kernel parametresi 3 olarak belirlenmiştir. Ağırlık parametreleri  $w_1$ ,  $w_2$ ,  $w_3$ , girdi parametleri  $i_1$ ,  $i_2$ ,  $i_3$ ,  $i_4$ ,  $i_5$ ,  $i_6$

ve çıktı parametreleri  $c_1, c_2, c_3, c_4, c_5, c_6$  olarak verilmiştir. Çıktı değerleri havuzlama katmanına gönderilerek öznitelik haritaları oluşturulmaktadır.

Türkçe tweet veri kümesinde Conv1D katmanında, 4 boyutlu kernel ve 128 evrişim filtresi kullanılmıştır. Aktivasyon fonksiyonu olarak ReLu fonksiyonu kullanılmıştır.

Türkçe mağaza yorumları veri kümesinde Conv1D katmanında, 2 boyutlu Kernel ve 64 evrişim filtresi kullanılmıştır. Aktivasyon fonksiyonu olarak ReLu fonksiyonu kullanılmıştır.

Havuzlama katmanı olarak maxpooling1D kullanılmıştır. Maxpooling1D katmanı evrişim katmanından elde edilen veriler ile öznitelik haritaları oluşturmaktadır. Bu adımda iki veri kümesinde de pencere boyutu 2 olarak belirlenmiştir. Şekil 3.9’de gösterildiği gibi 1x2 boyutundaki matrisler içerisinde maksimum değer seçilerek öznitelik haritaları oluşturulmaktadır.



Şekil 3.9. Maxpooling Gösterimi (Kuo & Huang, 2018)

LSTM katmanı, elde edilen özellik haritaları ile beslenmektedir. Havuzlama katmanından gelen özelliklerin ardışık ilişkilerini ve zaman bağımlılıklarını modellemek için kullanılmaktadır. LSTM modelinin hafıza yapısı, verinin geçmiş bilgisini koruyarak gelecekteki çıktıları tahmin etmeye yardımcı olmaktadır. Türkçe tweet veri kümesinde 128 LSTM hücresi, Türkçe mağaza yorumları veri kümesinde 32 LSTM hücresi kullanılmıştır.

Dense katmanı yani yoğun katman özelliklerin daha derinlemesine analizini yaparak sınıflandırma veya tahmin gibi çıktıları üretmek için kullanılmaktadır. Elde edilen verilerin softmax aktivasyon fonksiyonuna tabii tutulması ile çıktılar üretilmektedir.

### 3.4. BERT Modeli

Bert modeli ayırma, önceden eğitim, model yükleme, düzenleme ve ince ayar aşamalarından oluşmaktadır.

Ayrırma: Girdileri önceden eğitilmiş Bert Modeline uygun parçalara ayırmak ve ayrılan her parçayı belirli bir sayısal indekse dönüştürmek için Autokenizer modülü kullanılmıştır.

Önceden Eğitim ve Model Yükleme: Önceden eğitim sürecinde, Hugging Face platformu tarafından sağlanan bir dil modeli olan "dbmdz/bert-base-turkish-uncased" kullanılmıştır. Bu model, Türkçe Bert modelini ve sınıflandırma katmanlarını yüklemektedir. Girdileri dizi olarak alarak bu dizilerin sınıflandırmasını yapmaktadır. Modelin çıktısı, her girdi için tahmin edilen sınıf etiketlerini içerecektir.

Düzenleme: Eğitim, test ve doğrulama olarak ayrılan veri setleri kelimelere ayrılarak modelin anlayabileceği sayı vektörlerine dönüştürülmüştür. Bu sayısal vektörleri içeren liste ve etiketlerini içeren liste birleştirilerek Bert girişine uygun yeni eğitim, doğrulama ve test veri setleri oluşturulmuştur.

İnce Ayar: Eğitim veri kümesi 1000 örneklilik toplu gruplar halinde karıştırılmış ve Türkçe tweet veri kümesinde 64, Türkçe mağaza yorumları veri kümesinde 256 örneklilik toplu gruplar ile modelin beslemesi sağlanmıştır. Eğitim verilerini farklı düzenlemelerle karıştırmak ve küçük toplu veri grupları halinde modeli beslemek, modelin daha iyi genelleştirme yeteneği kazanmasına yardımcı olmaktadır.

### 3.5. Model Derlemeleri

Modelleri eğitilebilir hale getirmek için gerekli derlemeler yapılmıştır. Bu derleme aşamasında, optimizasyon algoritması, kayıp fonksiyonu ve metriklerle modeller hazırlanmıştır. Eğitim sırasında modellerin ağırlıklarını güncellemek ve performanslarını artırmak için Adam optimizasyon algoritması kullanılmıştır. Modellerin tahminlerinin gerçek etiketlerden ne kadar uzak olduğunu ölçmek için çapraz kategorik kayıp fonksiyonu kullanılmıştır. Eğitim ilerlemesini değerlendirmek için ise doğruluk metriği kullanılmıştır.

## 4. BULGULAR VE TARTIŞMA

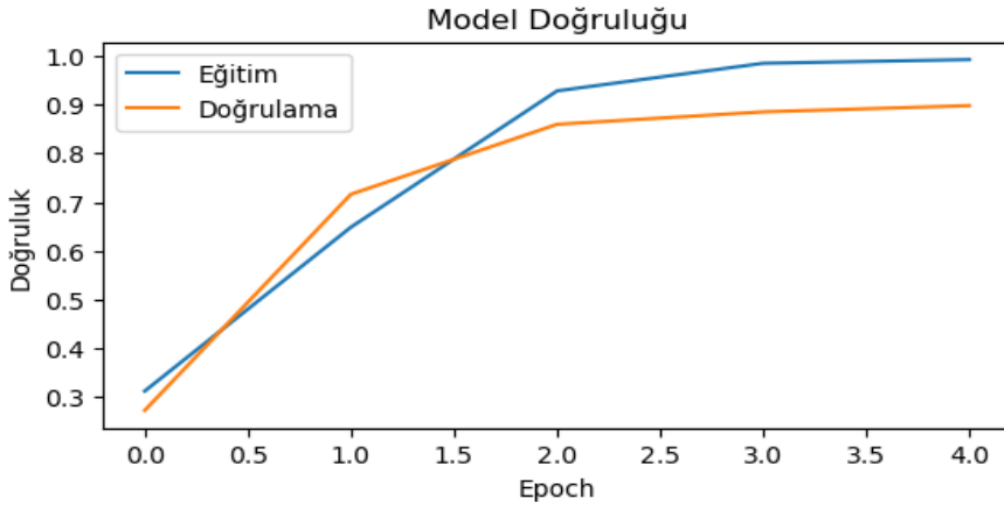
### 4.1. Türkçe Tweet Veri Kümesinde Bulgular

Türkçe tweet veri kümesi üzerinde uygulanan LSTM modelinin eğitim sürecinde 5 yaklaşım kullanılmıştır. Şekil 4.1’de her yaklaşımda eğitim ve doğrulama veri kümesi üzerinde elde edilen kayıp ve doğruluk değerleri verilmiştir.

```
Epoch 1/5  
79/79 [=====] - 6s 51ms/step - loss: 1.5874 - accuracy: 0.3123 - val_loss: 1.5089 - val_accuracy: 0.2727  
Epoch 2/5  
79/79 [=====] - 4s 45ms/step - loss: 1.0600 - accuracy: 0.6486 - val_loss: 0.7784 - val_accuracy: 0.7161  
Epoch 3/5  
79/79 [=====] - 4s 45ms/step - loss: 0.3091 - accuracy: 0.9282 - val_loss: 0.4229 - val_accuracy: 0.8596  
Epoch 4/5  
79/79 [=====] - 4s 46ms/step - loss: 0.0738 - accuracy: 0.9848 - val_loss: 0.3210 - val_accuracy: 0.8852  
Epoch 5/5  
79/79 [=====] - 4s 45ms/step - loss: 0.0402 - accuracy: 0.9924 - val_loss: 0.3176 - val_accuracy: 0.8979
```

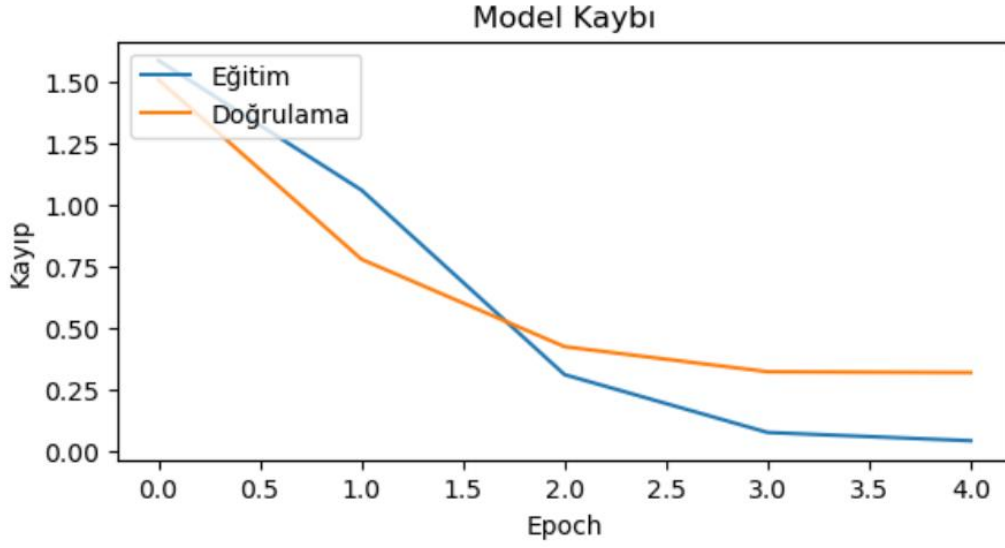
Şekil 4.1. Türkçe Tweet Veri Kümesi LSTM Yaklaşımları

Şekil 4.2’de LSTM modelinde eğitim ve doğrulama veri kümesinde 5 yaklaşım boyunca elde edilen doğruluk değerlerinin grafiği verilmiştir.



Şekil 4.2. Türkçe Tweet Veri Kümesinde LSTM Doğruluk Grafiği

Şekil 4.3’te LSTM modelinde eğitim ve doğrulama veri kümesinde 5 yaklaşım boyunca elde edilen kayıp değerlerinin grafiği verilmiştir.



Şekil 4.3. Türkçe Tweet Veri Kümesinde LSTM Kayıp Grafiği

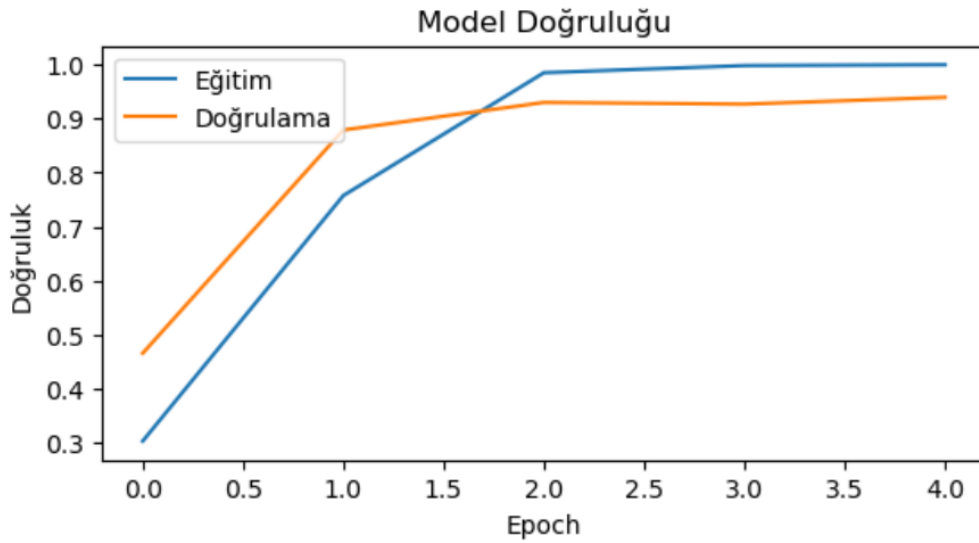
Türkçe tweet veri kümesi üzerinde uygulanan ConNN-LSTM Hibrit modelinin eğitim sürecinde 5 yaklaşım kullanılmıştır. Şekil 4.4'te her yaklaşımda eğitim ve doğrulama veri kümesi üzerinde elde edilen kayıp ve doğruluk değerleri verilmiştir.

```

Epoch 1/5
79/79 [=====] - 6s 50ms/step - loss: 1.5984 - accuracy: 0.3032 - val_loss: 1.4193 - val_accuracy: 0.4657
Epoch 2/5
79/79 [=====] - 4s 46ms/step - loss: 0.6984 - accuracy: 0.7575 - val_loss: 0.3953 - val_accuracy: 0.8788
Epoch 3/5
79/79 [=====] - 3s 44ms/step - loss: 0.0670 - accuracy: 0.9848 - val_loss: 0.2271 - val_accuracy: 0.9298
Epoch 4/5
79/79 [=====] - 3s 42ms/step - loss: 0.0108 - accuracy: 0.9980 - val_loss: 0.2181 - val_accuracy: 0.9266
Epoch 5/5
79/79 [=====] - 3s 43ms/step - loss: 0.0044 - accuracy: 0.9996 - val_loss: 0.2178 - val_accuracy: 0.9394

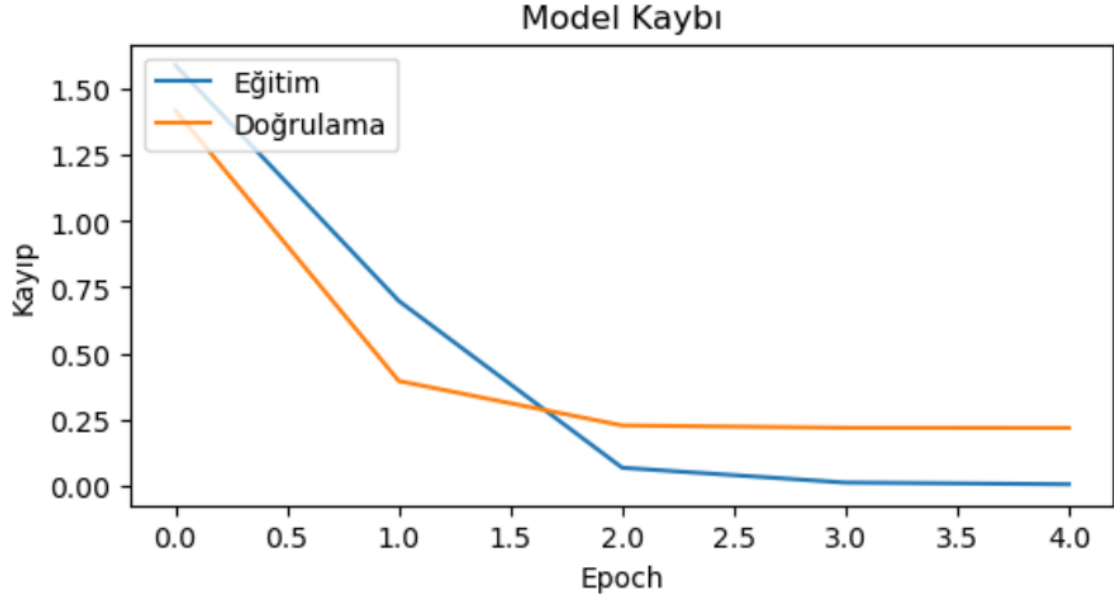
```

Şekil 4.4. Türkçe Tweet Veri Kümesi ConNN-LSTM Hibrit Modelinin Yaklaşımları



Şekil 4.5. Türkçe Tweet Veri Kümesi ConNN-LSTM Modelinin Doğruluk Grafiği

Şekil 4.5'te ConNN-LSTM hibrit modelinde eğitim ve doğrulama veri kümesinde 5 yaklaşımda elde edilen doğruluk değerlerinin grafiği gösterilmiştir.



Şekil 4.6. Türkçe Tweet Veri Kümesinde ConNN-LSTM Hibrit Modelinin Kayıp Grafiği

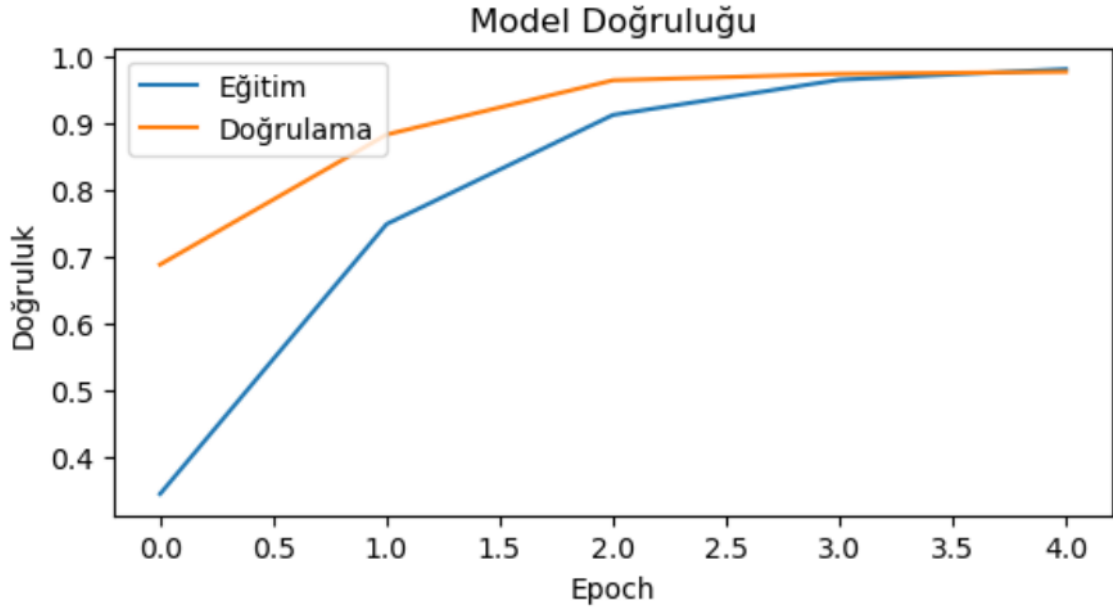
Şekil 4.6'da ConNN-LSTM hibrit modelinde eğitim ve doğrulama veri kümesinde 5 yaklaşımda elde edilen kayıp değerlerinin grafiği gösterilmiştir.

Türkçe tweet veri kümesi üzerinde uygulanan BERT modelinin eğitim sürecinde LSTM ve ConNN-LSTM Hibrit modelinin başarı oranlarını karşılaştırabilmek için 5 yaklaşım kullanılmıştır. Her yaklaşımda elde edilen kayıp ve doğruluk değerleri Şekil 4.7'de gösterilmiştir.

```
Epoch 1/5
40/40 [=====] - 138s 3s/step - loss: 1.5404 - accuracy: 0.3454 - val_loss: 1.3377 - val_accuracy: 0.6890
Epoch 2/5
40/40 [=====] - 115s 3s/step - loss: 1.0918 - accuracy: 0.7495 - val_loss: 0.6201 - val_accuracy: 0.8836
Epoch 3/5
40/40 [=====] - 109s 3s/step - loss: 0.4560 - accuracy: 0.9130 - val_loss: 0.2038 - val_accuracy: 0.9649
Epoch 4/5
40/40 [=====] - 110s 3s/step - loss: 0.1938 - accuracy: 0.9657 - val_loss: 0.1214 - val_accuracy: 0.9745
Epoch 5/5
40/40 [=====] - 111s 3s/step - loss: 0.1122 - accuracy: 0.9817 - val_loss: 0.1031 - val_accuracy: 0.9777
```

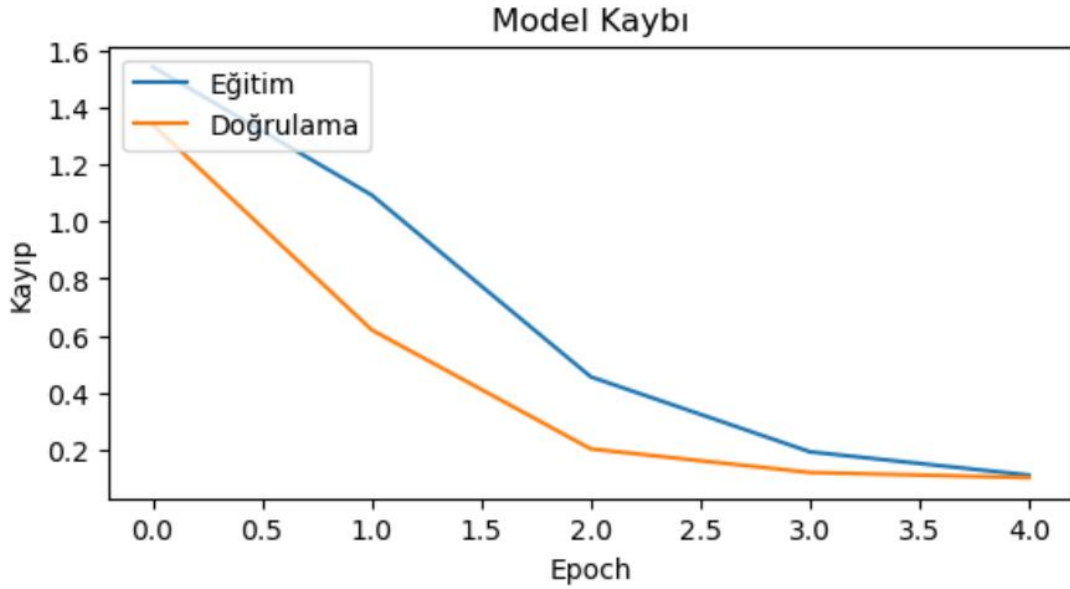
Şekil 4.7. Türkçe Tweet Veri Kümesi BERT Modeli Yaklaşımları

Şekil 4.8'te BERT modelinde eğitim ve doğrulama veri kümesinde 5 yaklaşımda elde edilen doğruluk değerlerinin grafiği gösterilmiştir.



Şekil 4.8. Türkçe Tweet Veri Kümesinde BERT Modeli Doğruluk Grafiği

Şekil 4.9’da BERT modelinde eğitim ve doğrulama veri kümesinde 5 yaklaşımda elde edilen kayıp değerlerinin grafiği gösterilmiştir.



Şekil 4.9. Türkçe Tweet Veri Kümesinde BERT Modeli Kayıp Grafiği

Türkçe tweet veri kümesi üzerinde eğitilen LSTM, ConNN-LSTM Hibrit ve BERT modellerinin test veri kümeleri üzerindeki performansları Tablo 4.1’de gösterilmiştir.

Tablo 4.1. Türkçe Tweet Veri Kümesinde Model Performansları

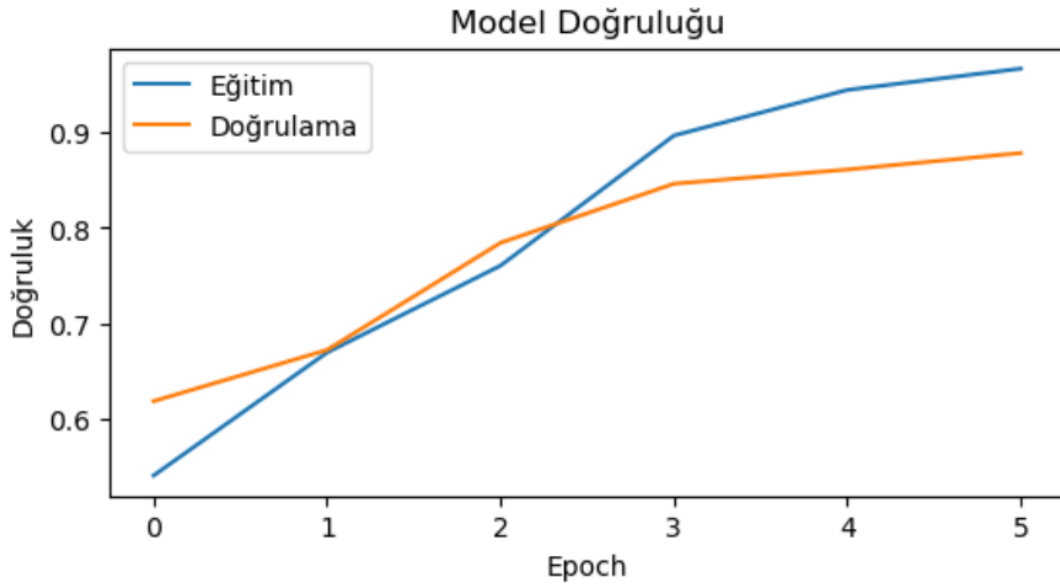
Modeller	Doğruluk Oranı
LSTM Modeli	90,69%
ConNN-LSTM Hibrit Modeli	92,60%
BERT Modeli	96,81%

#### 4.2. Türkçe Mağaza Yorumları Veri Kümesinde Bulgular

Türkçe mağaza yorumları veri kümesi üzerinde uygulanan LSTM ve ConNN-LSTM modellerinin eğitim sürecinde 6, Bert modelinin eğitim sürecinde 5 yaklaşım kullanılmıştır. LSTM Modelinin 6 yaklaşımında elde edilen kayıp ve doğruluk değerleri Şekil 4.10'da gösterilmiştir.

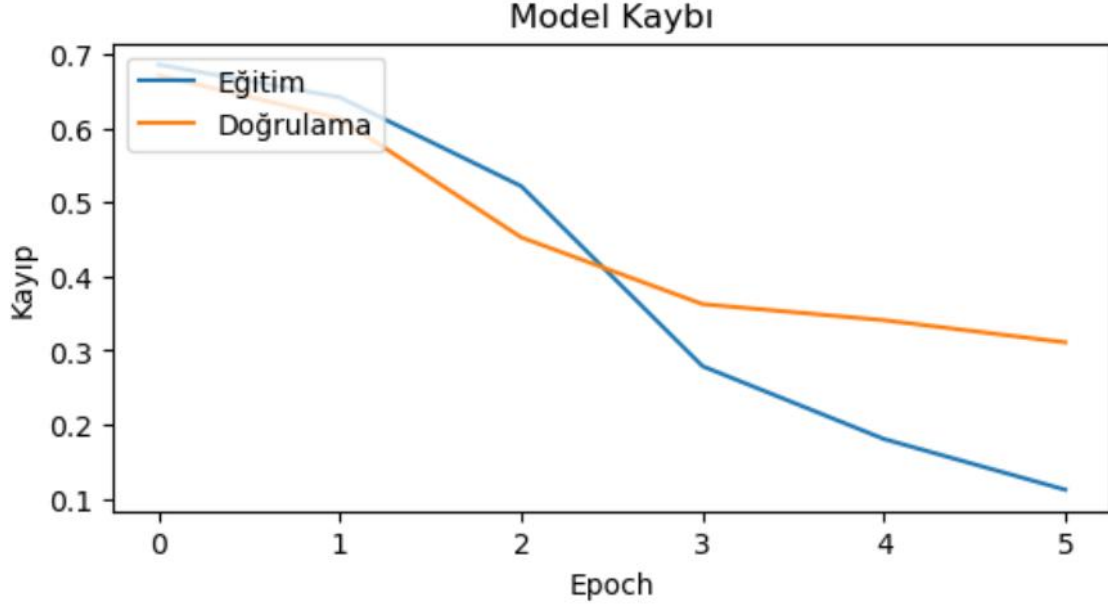
```
Epoch 1/6
12/12 [=====] - 4s 199ms/step - loss: 0.6854 - accuracy: 0.5406 - val_loss: 0.6711 - val_accuracy: 0.6185
Epoch 2/6
12/12 [=====] - 2s 146ms/step - loss: 0.6412 - accuracy: 0.6695 - val_loss: 0.6126 - val_accuracy: 0.6722
Epoch 3/6
12/12 [=====] - 2s 149ms/step - loss: 0.5215 - accuracy: 0.7605 - val_loss: 0.4524 - val_accuracy: 0.7845
Epoch 4/6
12/12 [=====] - 2s 152ms/step - loss: 0.2788 - accuracy: 0.8969 - val_loss: 0.3625 - val_accuracy: 0.8464
Epoch 5/6
12/12 [=====] - 2s 151ms/step - loss: 0.1806 - accuracy: 0.9447 - val_loss: 0.3408 - val_accuracy: 0.8613
Epoch 6/6
12/12 [=====] - 2s 149ms/step - loss: 0.1122 - accuracy: 0.9671 - val_loss: 0.3109 - val_accuracy: 0.8786
```

Şekil 4.10. Türkçe Mağaza Yorumları Veri Kümesi LSTM Modeli Yaklaşımları



Şekil 4.11. Türkçe Mağaza Yorumları Veri Kümesi LSTM Modeli Doğruluk Grafiği

Şekil 4.11’de LSTM modelinde eğitim ve doğrulama veri kümesinde 6 yaklaşım boyunca elde edilen doğruluk değerlerinin grafiği verilmiştir.



Şekil 4.12. Türkçe Mağaza Yorumları Veri Kümesi LSTM Modeli Kayıp Grafiği

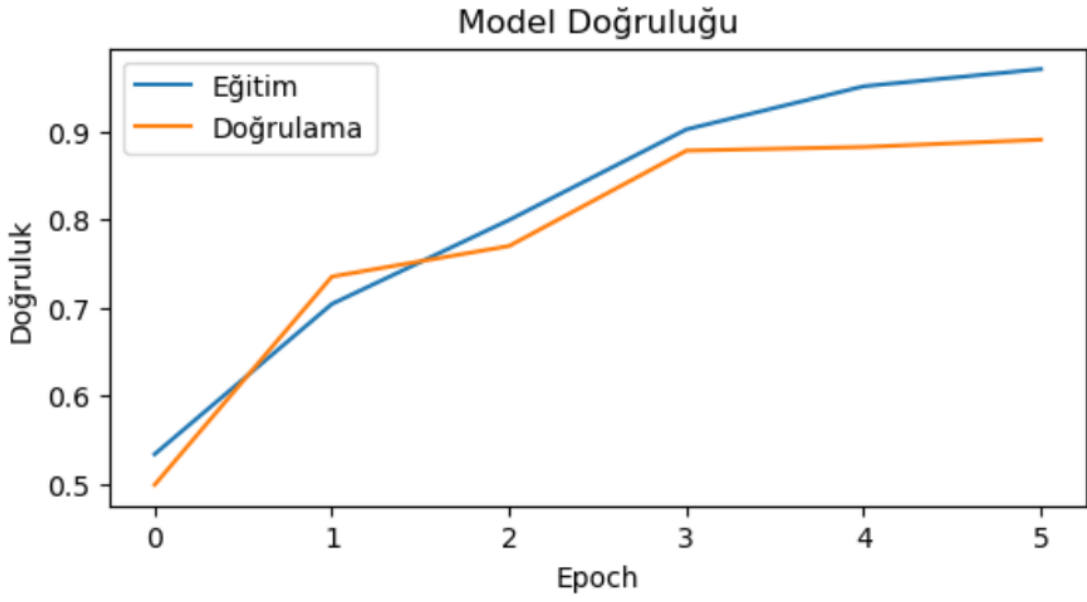
Şekil 4.12’de LSTM modelinde eğitim ve doğrulama veri kümesinde 10 yaklaşımda elde edilen kayıp değerlerinin grafiği gösterilmiştir.

Türkçe mağaza yorumları veri kümesi üzerinde uygulanan ConNN-LSTM hibrit modelinin 6 yaklaşımında elde edilen kayıp ve doğruluk değerleri Şekil 4.13’te gösterilmiştir.

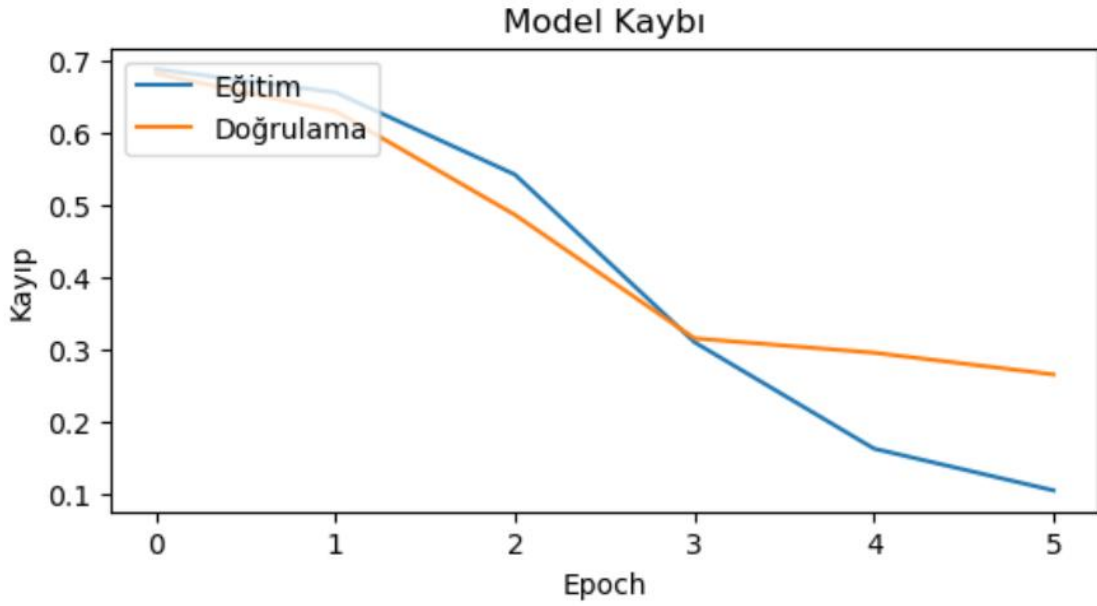
```
Epoch 1/6  
12/12 [=====] - 3s 96ms/step - loss: 0.6877 - accuracy: 0.5335 - val_loss: 0.6815 - val_accuracy: 0.4988  
Epoch 2/6  
12/12 [=====] - 0s 33ms/step - loss: 0.6555 - accuracy: 0.7039 - val_loss: 0.6294 - val_accuracy: 0.7349  
Epoch 3/6  
12/12 [=====] - 0s 33ms/step - loss: 0.5416 - accuracy: 0.7991 - val_loss: 0.4858 - val_accuracy: 0.7696  
Epoch 4/6  
12/12 [=====] - 0s 30ms/step - loss: 0.3094 - accuracy: 0.9019 - val_loss: 0.3147 - val_accuracy: 0.8778  
Epoch 5/6  
12/12 [=====] - 0s 41ms/step - loss: 0.1619 - accuracy: 0.9508 - val_loss: 0.2948 - val_accuracy: 0.8819  
Epoch 6/6  
12/12 [=====] - 0s 29ms/step - loss: 0.1040 - accuracy: 0.9702 - val_loss: 0.2647 - val_accuracy: 0.8902
```

Şekil 4.13. Türkçe Mağaza Yorumları Veri Kümesi ConNN-LSTM Hibrit Modeli Yaklaşımları

Şekil 4.14’te ConNN-LSTM hibrit modelinde eğitim ve doğrulama veri kümesinde 6 yaklaşım boyunca elde edilen doğruluk değerlerinin grafiği verilmiştir.



Şekil 4.14. Türkçe Mağaza Yorumları Veri Kümesi ConNN-LSTM Hibrit Modeli Doğruluk Grafiği



Şekil 4.15. Türkçe Mağaza Yorumları Veri Kümesi ConNN-LSTM Hibrit Modeli Kayıp Grafiği

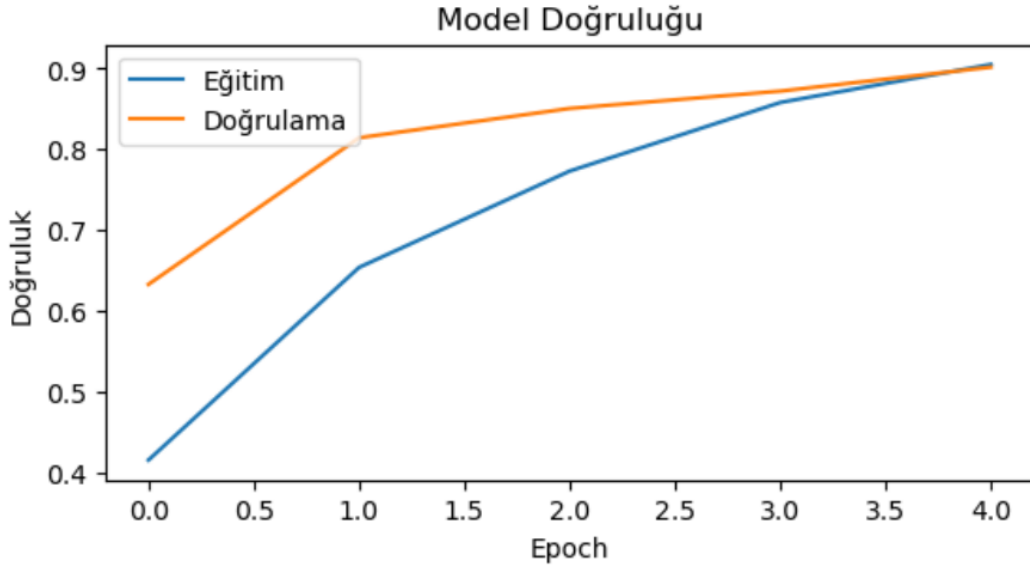
Şekil 4.15'te ConNN-LSTM hibrit modelinde eğitim ve doğrulama veri kümesinde 6 yaklaşım boyunca elde edilen kayıp değerlerinin grafiği verilmiştir.

Türkçe mağaza yorumları veri kümesi üzerinde uygulanan Bert modelinin 5 yaklaşımında elde edilen kayıp ve doğruluk değerleri Şekil 4.16'da gösterilmiştir.

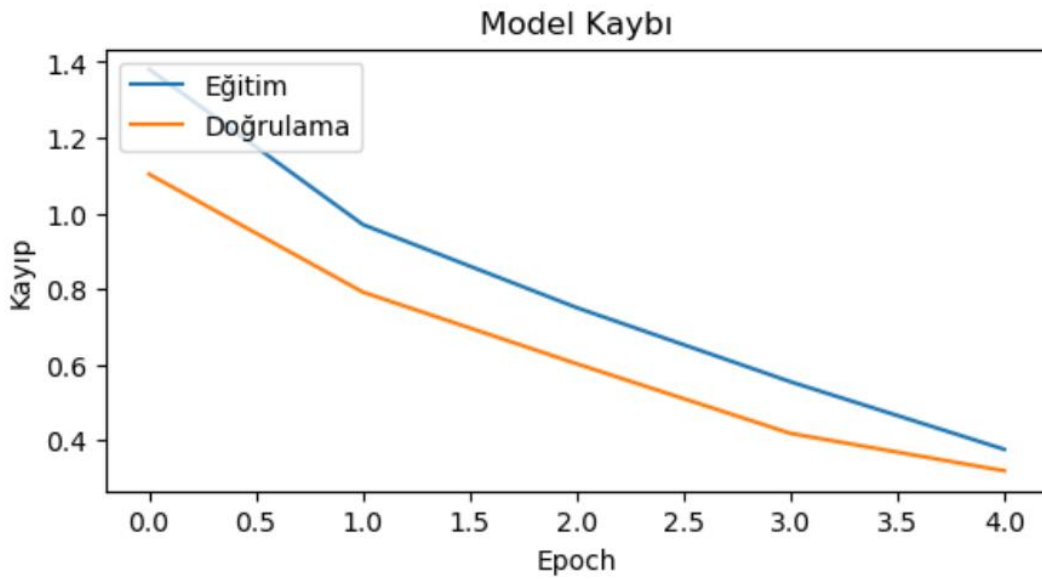
```
Epoch 1/5
12/12 [=====] - 328s 21s/step - loss: 1.3808 - accuracy: 0.4148 - val_loss: 1.1031 - val_accuracy: 0.6325
Epoch 2/5
12/12 [=====] - 227s 19s/step - loss: 0.9699 - accuracy: 0.6536 - val_loss: 0.7912 - val_accuracy: 0.8142
Epoch 3/5
12/12 [=====] - 223s 19s/step - loss: 0.7502 - accuracy: 0.7729 - val_loss: 0.6017 - val_accuracy: 0.8505
Epoch 4/5
12/12 [=====] - 224s 19s/step - loss: 0.5542 - accuracy: 0.8580 - val_loss: 0.4176 - val_accuracy: 0.8720
Epoch 5/5
12/12 [=====] - 228s 19s/step - loss: 0.3755 - accuracy: 0.9051 - val_loss: 0.3187 - val_accuracy: 0.9017
```

Şekil 4.16. Türkçe Mağaza Yorumları Veri Kümesi Bert Modeli Yaklaşımları

Şekil 4.17’de Bert modelinin eğitim ve doğrulama veri kümesinde 5 yaklaşım boyunca elde edilen doğruluk değerlerinin grafiği gösterilmiştir.



Şekil 4.17. Türkçe Mağaza Yorumları Veri Kümesi Bert Modeli Doğruluk Grafiği



Şekil 4.18. Türkçe Mağaza Yorumları Veri Kümesi Bert Modeli Kayıp Grafiği

Şekil 4.18’de Bert modelinin eğitim ve doğrulama veri kümesinde 5 yaklaşım boyunca elde edilen kayıp değerlerinin grafiği gösterilmiştir.

Türkçe mağaza yorumları eğitim veri kümesi üzerinde eğitilen LSTM, ConNN-LSTM Hibrit ve BERT modellerinin test veri kümesi üzerindeki performansları Tablo 4.2’de gösterilmiştir.

Tablo 4.2. Türkçe Mağaza Yorumları Veri Kümesinde Model Performansları

<b>Modeller</b>	<b>Doğruluk Oranı</b>
LSTM Modeli	88,27%
ConNN-LSTM Hibrit Modeli	88,32%
BERT Modeli	91,21%

### 4.3. Tartışma

Literatürde, ConNN-LSTM hibrit modellerinin kullanıldığı çalışmalar incelendiğinde kullanılan bu hibrit modelin diğer modellere göre daha başarılı sonuçlar verdiği gözlemlenmiştir (Çetiner, 2022, 2023; Er & Işık, 2021; Görgel & Kavlak, 2020; Özbay, 2022; Öztürk & Pashaei, 2021; Suncak & Aktaş, 2022; Toprak, 2023; Zeybel Peköz & İnkaya, 2023). Bu çalışmada derin öğrenme modelleri olan LSTM ve ConNN-LSTM hibrit modelleri karşılaştırıldığında Türkçe tweet veri kümesi üzerinde ConNN-LSTM hibrit modeli LSTM modeline göre daha başarılı iken Türkçe mağaza yorumları veri kümesi üzerinde LSTM modeli, ConNN-LSTM hibrit modeline göre daha başarılı sonuç verdiği gözlemlenmiştir.

Makine öğrenmesi, derin öğrenme ve transformatör modellerin karşılaştırıldığı başka bir çalışmada ise transformatör modellerin daha başarılı sonuç verdiği gözlemlenmiştir (Özbay, 2022). Bu iki veri kümesi üzerinde yapılan çalışmada, literatürde yapılan çalışma desteklenerek transformatör model olan BERT modelinin derin öğrenme modelleri olan LSTM ve ConNN-LSTM modellerinden daha başarılı olduğu gözlemlenmiştir.

Kullandığım Türkçe tweet veri kümesi üzerinde yapılan çalışmada konu modelleme algoritmalarından LDA, LSA ve PLSA kullanılmıştır. 3 sınıf ve 5 sınıf olmak üzere veri kümesini ikiye ayırarak çalışmayı gerçekleştirmiştir. 3 Sınıflı duygu analizinde en yüksek başarıyı 95,4% ile 5 sınıflı duygu analizinde en yüksek başarıyı 87% ile LSA yönteminde elde etmişlerdir (Güven ve diğ., 2020). 3 Sınıflı duygu analizinde: LSTM ve ConNN-LSTM hibrit modeli, LSA modelinden aldıkları sonuca göre daha düşük başarı oranı elde etse de Klasik LDA, 2 aşamalı LDA, 3 aşamalı LDA ve PLSA modellerinden daha başarılı olduğu sonuç vermiştir. 5 Sınıflı duygu analizinde, LSTM ve ConNN-LSTM hibrit modeli bütün çalışmalardan daha başarılı sonuç vermiştir. 3 ve 5 sınıflı duygu analizinde LSA modelinin başarı oranının LSTM ve ConNN-LSTM hibrit modellerinin başarı oranından fazla ya da düşük olmasının veri kümesinin boyutunun değişmesinden kaynaklandığı düşünülmektedir. Veri kümesi küçüldüğünde LSA modelinin başarıları artarken veri kümesi büyüdükçe LSTM ve ConNN-LSTM hibrit modellerinin başarılarının arttığı görülmektedir. Uzun vadeli bağımlılıkları yakalamada daha başarılı olduğu için BERT modelinin çalıştıkları bütün modellerden daha başarılı sonuç verdiği düşünülmektedir.

Kullandığım Türkçe mağaza yorumları veri kümesi üzerinde yapılan duygu analizi çalışmasında LR, NB, RFR ve XGBoost modellerini kullanmıştır (Sargın, 2022). LR modelindeki en başarılı sonuç 88,13%, NB modelinde 89,21%, RFR modelinde 85,86%, XGBoost modelinde 84,69% olduğu tespit edilmiştir. LSTM ve ConNN-LSTM hibrit modelleri LR, RFR ve XGBoost modellerinden daha başarılı olurken NB modeline göre daha başarısız oldukları görülmüştür. NB modeli küçük veri kümelerinde basit ama etkili bir yöntemdir. Aşırı öğrenme probleminde daha etkili olduğu için LSTM ve ConNN-LSTM hibrit modellerinden daha başarılı olduğu düşünülmektedir. BERT modelinin ise önceden eğitim yapılmış bir yapıyı kullanarak kelimenin anlamını, cümlenin içindeki diğer kelimelerle ilişkisini anlamadaki başarıları sebebi ile yapılan çalışmadaki modellerden daha başarılı olduğu düşünülmektedir.

Kullandığım Türkçe mağaza yorumları veri kümesi üzerinde Doruk AFACAN tarafından yapılan çalışmada RFR modeli ile 86,8% başarı oranı elde edilmiştir (Afacan, 2020). LSTM, BERT ve ConNN-LSTM hibrit modellerinde alınan başarı oranı RFR modelinde alınan başarı oranından daha yüksektir. Karar ağaçları, özellikleri bölerek ve

sınıflandırarak genel bir model oluşturur. Bu nedenle, dilin karmaşıklığını daha etkili bir şekilde kavrayabilen ve önceki katmanlarda öğrendikleri bilgileri kullanabilen LSTM, BERT ve CNN-LSTM hibrit modellerine göre daha düşük performans sergileme eğilimindedir.

Türkçe mağaza yorumları veri kümesi üzerinde Kaggle platformunda LR, DTR (Decision Tree Regression, Karar Ağacı Regresyonu), RFR ve NB modelleri kullanılarak Umut AKTER tarafından yapılan sınıflandırma çalışmasında 87% ile LR en başarılı sonucu vermiştir. (Akter, 2021). LSTM, BERT ve ConNN-LSTM hibrit modellerinde alınan başarı oranının çalışmadaki modellerden daha yüksek olduğu görülmüştür. Lineer bir model olan LR modeli, her kelimeyi bağımsız olarak değerlendirdiği için kelime sırası ve cümle içindeki bağlamı anlamada özyinelemeli model olan LSTM ve dikkat mekanizmasını kullanan BERT modeline göre daha başarısız olduğu düşünülmektedir.

## 5. SONUÇ VE ÖNERİLER

Duygu analizi toplumun belirli olaylara karşı verdiği tepkilerin anlaşılması için önemlidir. Bu tepkilerin toplanabilmesi için Twitter önemli bir araçtır. Ürün satışlarının güvenilir ve gerçek tepkili olarak yapılabilmesi içinde çevrimiçi alışveriş sitelerindeki yorumlar önemli faktörlerdir. Literatürdeki çalışmalarda duygu analizi yapılırken makine öğrenmesi, derin öğrenme, transformatör modellerin birçok çeşidinin kullanıldığı görülmüştür. Çalışmamızda kullandığımız yöntemler literatürdeki yöntemlerin bir araya getirilmesi ile oluşturulmuştur.

Bu tez çalışmasında etiketli Türkçe metinler içeren iki Excel veri kümesi kullanılmıştır. Bu veri kümeleri üzerinde kullanılacak modellere uygun olarak dönüşümler gerçekleştirilmiştir. Derlemeler tamamlandıktan sonra LSTM, ConNN-LSTM ve BERT modelleri ile eğitim gerçekleştirilmiştir. Gerçekleştirilen eğitim sonrasında modellerin tahmin ettiği değerler ile gerçek değerler karşılaştırılarak modellerin performansları elde edilmiştir. Türkçe Tweet veri kümesi üzerinde doğruluk performansları, LSTM modelinde 90,69%, ConNN-LSTM hibrit modelinde 92,6% ve BERT modelinde 96,81% olarak tespit edilmiştir. Türkçe mağaza yorumları veri kümesi üzerinde doğruluk performansları, LSTM modelinde 88,27%, ConNN-LSTM hibrit modelinde 88,32% ve BERT modelinde 91,21% olarak tespit edilmiştir. Üç modelinde Türkçe Tweet veri kümesi üzerinde daha başarılı olduğu gözlemlenmiştir. İki veri kümesinde de ve ConNN-LSTM hibrit modelinin LSTM modeline göre daha başarılı olduğu, BERT modelinin ise bu iki modelden daha başarılı olduğu tespit edilmiştir.

Türkçe Tweet ve Türkçe mağaza yorumları veri kümesi üzerinde literatürde yapılan çalışmalar incelendiğinde LSTM ile ConNN-LSTM hibrit modellerinin başarı oranının modellere ve veri kümesine göre değişkenlik gösterdiği, BERT modelinin ise çalışılan bütün modellerden daha başarılı sonuç verdiği gözlemlenmiştir.

Literatürde yapılan incelemelere göre derin öğrenme, makine öğrenmesi ve transformatör modellerin karşılaştırmasına ve hibrit model kullanımına yer verilmediği, veri kümesi etiketlerinin genellikle 0-1 gibi ikili sınıftan oluşturulduğu gözlemlenmiştir. İleride yapılacak çalışmalarda daha büyük veri kümesi ve daha fazla etiket ile derin öğrenme,

makine öğrenmesi ve transformatör modeller ile hibrit model karşılaştırılmaları yapılabilir.

Bu tez çalışmasının ileride gerçekleştirilecek duygu analizi arařtırmalarına rehberlik edebileceđi düşünölmektedir.



## KAYNAKLAR

- Afacan, D. (2020). *NLP Workshop—BoW*. Kaggle. <https://kaggle.com/code/afajohn/nlp-workshop-bow>, (Ziyaret Tarihi: 19 Kasım 2023).
- Akter, U. (2021). *Turkish BOW*. Kaggle. <https://kaggle.com/code/umutakter/turkish-bow>, (Ziyaret Tarihi: 10 Kasım 2023).
- Amidi, S., & Amidi, A. (2018). Derin Öğrenme El Kitabı. *Stanford University*, <https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks>.
- Apicella, A., Donnarumma, F., Isgrò, F., & Prevete, R. (2021). A survey on modern trainable activation functions. *Neural Networks*, 138, 14-32. <https://doi.org/10.1016/j.neunet.2021.01.026>.
- Aslan, S. (2022). BiGRU-CNN Tabanlı Derin Öğrenme Modeliyle Türkiye'deki Covid-19 Aşılara Yönelik Twitter Duygu Analizi. *International Journal of Pure and Applied Sciences*, 8(2), 312-330. <https://doi.org/10.29132/ijpas.1087486>.
- Banerjee, K., C, V. P., Gupta, R. R., Vyas, K., H, A., & Mishra, B. (2020). *Exploring Alternatives to Softmax Function* (arXiv:2011.11538). arXiv. <http://arxiv.org/abs/2011.11538>.
- Bilen, B. (2020). *Duygu Analizi Veri Seti (Olumlu/Olumsuz/Tarafsız)*. <https://www.kaggle.com/datasets/burhanbilenn/duygu-analizi-icin-urun-yorumlari>, (Ziyaret Tarihi: 19 Ağustos 2023).
- Bilgin, A. (2021). TensorFlow ve Keras Nedir? *Medium*. <https://aylablgn.medium.com/tensorflow-ve-keras-nedir-4d68e1c378d4>, (Ziyaret Tarihi: 19 Kasım 2023).
- Bohrer, J. da S., Grisci, B. I., & Dorn, M. (2020). *Neuroevolution of Neural Network Architectures Using CoDeepNEAT and Keras* (arXiv:2002.04634). arXiv. <http://arxiv.org/abs/2002.04634>.
- Chicho, B. T., & Sallow, A. B. (2021). A Comprehensive Survey of Deep Learning Models Based on Keras Framework. *Journal of Soft Computing and Data Mining*, 2(2), Article 2.
- Çakmak, E., & Selvi, İ. (2022). Derin Öğrenme (CNN, RNN, LSTM, GRU) Kullanarak Protein İkincil Yapı Tahmini. *Acta Infologica*, 6(1), 43-52. <https://doi.org/10.26650/acin.1008075>.
- Çarkacı, N. (2020). Derin Öğrenme Uygulamalarında En Sık Kullanılan Hiper-parametreler. *Medium*. <https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-en-sik-kullanilan-hiper-parametreler-ece8e9125c4>, (Ziyaret Tarihi: 19 Şubat 2023).

- Çetiner, H. (2022). Multi-Label Text Analysis with a CNN and LSTM Based Hybrid Deep Learning Model. *Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi*, 9(17), 447-457. <https://doi.org/10.54365/adyumbd.1106981>.
- Çetiner, H. (2023). Hybrid Deep Learning Implementation for Crop Yield Prediction. *Afyon Kocatepe University Journal of Sciences and Engineering*, 23(3), 648-660. <https://doi.org/10.35414/akufemubid.1116187>.
- Daşgın, R., & Adem, K. (2023). Kitleleş Çevrimiçi Ders Platformlarında Kurslara Yapılan Yorumların Metin Madenciliği Kullanılarak Duygu Analizi. *Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi*. <https://doi.org/10.29137/umagd.1266187>.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (arXiv:1810.04805). arXiv. <http://arxiv.org/abs/1810.04805>.
- Dönmez, H. G., & Becerikli, Y. (2023). Türkçe Twitter Verileri Üzerinde Duygu Analizi: LSTM, CNN-LSTM, BERT Algoritmalarının Karşılaştırılması. *International Journal of Advanced Natural Sciences and Engineering Researches*, 7(6), 132-138. <https://doi.org/10.59287/ijanser.1146>.
- Dunne, R. A., & Campbell, N. A. (1997). *On the Pairing of the Softmax Activation and Cross-Entropy Penalty Functions and the Derivation of the Softmax Activation Function*. 181-185.
- Er, M. B., & Işık, İ. (2021). LSTM Tabanlı Derin Ağlar Kullanılarak Diyabet Hastalığı Tahmini. *Türk Doğa ve Fen Dergisi*, 10(1), 68-74. <https://doi.org/10.46810/tdfd.818528>.
- Ghorbani, M., Bahaghighat, M., Xin, Q., & Özen, F. (2020). ConvLSTMConv Network: A Deep Learning Approach for Sentiment Analysis in Cloud Computing. *Journal of Cloud Computing: Advances, Systems and Applications*, 9(1), 16. <https://doi.org/10.1186/s13677-020-00162-1>.
- Gligorić, K., Anderson, A., & West, R. (2018). How Constraints Affect Content: The Case of Twitter's Switch from 140 to 280 Characters. *Proceedings of the International AAAI Conference on Web and Social Media*, 12(1), Article 1. <https://doi.org/10.1609/icwsm.v12i1.15079>.
- Göğüş, B. (2022). *Ataletsel Navigasyon Sistemlerinde Kestirim için Farklı Yöntemlerin Performanslarının Karşılaştırılması*. Yüksek Lisans Tezi, Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 757223.
- Görgel, P., & Kavlak, E. (2020). Uzun Kısa Süreli Hafıza ve Evrimsel Sinir Ağları ile Rüzgar Enerjisi Üretim Tahmini. *DÜMF Mühendislik Dergisi*, 11(1), 69-80. <https://doi.org/10.24012/dumf.596533>.
- Grandini, M., Bagli, E., & Visani, G. (2020). *Metrics for Multi-Class Classification: An Overview* (arXiv:2008.05756). arXiv. <http://arxiv.org/abs/2008.05756>.

- Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10), 2222-2232. <https://doi.org/10.1109/TNNLS.2016.2582924>
- Gummesson, A. (2016). *Prostate Cancer Classification using Convolutional Neural Networks*. Lund University, Faculty of Engineering, Sweden.
- Güven, A. (2021). *Turkish Tweets Dataset*. Kaggle. <https://www.kaggle.com/datasets/anil1055/turkish-tweet-dataset>.
- Güven, Z. A. (2020). *Turkish Tweets Dataset*. Kaggle. <https://www.kaggle.com/datasets/anil1055/turkish-tweet-dataset>, (Ziyaret Tarihi: 15 Ocak 2023).
- Güven, Z. A., Diri, B., & Çakaloğlu, T. (2020). Duygu analizi için n-aşamalı Gizli Dirichlet Ayırımı ile diğer konu modelleme yöntemlerinin karşılaştırılması. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 35(4), 2135-2146. <https://doi.org/10.17341/gazimmfd.556104>.
- Haliloglu, M. (2020). *BERT*. Medium. <https://medium.com/@mehmethaliloglu/bert-979e28cd08b7>, (Ziyaret Tarihi: 08 Mart 2023).
- Kara, A. (2019). Uzun-Kısa Süreli Bellek Ağı Kullanarak Global Güneş Işınımı Zaman Serileri Tahmini. *Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji*, 7(4), 882-892. <https://doi.org/10.29109/gujsc.571831>.
- Kim, Y. (2014). *Convolutional Neural Networks for Sentence Classification* (arXiv:1408.5882). arXiv. <http://arxiv.org/abs/1408.5882>.
- Koeh, K. E. (2022). *Cross-Entropy Loss Function*. Medium. <https://towardsdatascience.com/cross-entropy-loss-function-f38c4ec8643e>, (Ziyaret Tarihi: 10 Nisan 2023).
- Konasani, V. R., & Kadre, S. (2021). *Machine Learning and Deep Learning Using Python and Tensorflow* (1. bs). McGraw-Hill.
- Kuo, P.-H., & Huang, C.-J. (2018). A Green Energy Application in Energy Management Systems by an Artificial Intelligence-Based Solar Radiation Forecasting Model. *Energies*, 11(4), Article 4. <https://doi.org/10.3390/en11040819>.
- Küçüktaş, Ü. T., Uysal, F., Hardalaç, F., & Biri, İ. (2021). BERT Modeli ile Covid-19 Aşısı için Aşı Karşıtı Tweetlerin Arttığı Zaman Aralıklarının Tespiti. *1. Uluslararası Yapay Zekâ ve Veri Bilimi Kongresi*, 19-24.
- Li, P., Wang, D., Wang, L., & Lu, H. (2018). Deep Visual Tracking: Review and Experimental Comparison. *Pattern Recognition*, 76, 323-338. <https://doi.org/10.1016/j.patcog.2017.11.007>.

- Nwankpa, C., Ijomah, W., Gachagan, A., & Marshall, S. (2018). *Activation Functions: Comparison of trends in Practice and Research for Deep Learning* (arXiv:1811.03378). arXiv. <http://arxiv.org/abs/1811.03378>.
- Özbay, E. (2022). Transformatör-Tabanlı Evrişimli Sinir Ağı Modeli Kullanarak Twitter Verisinde Saldırganlık Tespiti. *Konya Journal of Engineering Sciences*, 10(4), 986-1001. <https://doi.org/10.36306/konjes.1061807>.
- Özkaya, U. (2020). *Derin Öğrenme Yöntemleri ile GPR B Tarama Görüntülerinin Analizi*. Doktora Tezi, Konya Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Konya, 637564.
- Öztürk, Ö. F., & Pashaei, E. (2021). Konuşmalardaki Duygunun Evrişimsel LSTM Modeli ile Tespiti. *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, 581-589. <https://doi.org/10.24012/dumf.1001914>.
- Park, D., & Ahn, C. W. (2019). Self-Supervised Contextual Data Augmentation for Natural Language Processing. *Symmetry*, 11(11), Article 11. <https://doi.org/10.3390/sym11111393>.
- Pham, H. V., Lutellier, T., Qi, W., & Tan, L. (2019). CRADLE: Cross-Backend Validation to Detect and Localize Bugs in Deep Learning Libraries. *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, 1027-1038. <https://doi.org/10.1109/ICSE.2019.00107>.
- Picard, R. G. (2011). *The Economics and Financing of Media Companies* (2. bs). Fordham Univ Press.
- Qian, N. (1999). On the Momentum Term in Gradient Descent Learning Algorithms. *Neural Networks*, 12(1), 145-151. [https://doi.org/10.1016/S0893-6080\(98\)00116-6](https://doi.org/10.1016/S0893-6080(98)00116-6).
- Ruby, U., & Yendapalli, V. (2020). Binary Cross Entropy with Deep Learning Technique for Image Classification. *International Journal of Advanced Trends in Computer Science and Engineering*, 9. <https://doi.org/10.30534/ijatcse/2020/175942020>.
- Ruder, S. (2017). *An Overview of Gradient Descent Optimization Algorithms* (arXiv:1609.04747). arXiv. <http://arxiv.org/abs/1609.04747>.
- Sargın, A. F. (2022). *Mağaza Duygu Analizi*. Kaggle. <https://kaggle.com/code/ahmetfaruksargin/magza-duygu-anal-z>, (Ziyaret Tarihi: 18 Kasım 2023).
- Sharma, S., Sharma, S., & Athaiya, A. (2020). Activation Function in Neural Networks. *International Journal of Engineering Applied Sciences and Technology*, 04(12), 310-316. <https://doi.org/10.33564/IJEAST.2020.v04i12.054>.
- Suncak, A., & Aktaş, Ö. (2022). Hibrit bir Derin Öğrenme Yöntemi Kullanarak Türkçe Cümlelerdeki Anlatım Bozukluklarının Tespiti. *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, 24(72), 825-834.

- Sutton, R. S. (1986). Two problems with backpropagation and other steepest-descent learning procedures for networks. *Proc. of Eighth Annual Conference of the Cognitive Science Society*, 823-831.
- Szandała, T. (2021). Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks. İçinde A. K. Bhoi, P. K. Mallick, C.-M. Liu, & V. E. Balas (Ed.), *Bio-inspired Neurocomputing* (C. 903, ss. 203-224). Springer Singapore. [https://doi.org/10.1007/978-981-15-5495-7\\_11](https://doi.org/10.1007/978-981-15-5495-7_11).
- To, Q. G., To, K. G., Huynh, V.-A. N., Nguyen, N. T. Q., Ngo, D. T. N., Alley, S. J., Tran, A. N. Q., Tran, A. N. P., Pham, N. T. T., Bui, T. X., & Vandelanotte, C. (2021). Applying Machine Learning to Identify Anti-Vaccination Tweets during the Covid-19 Pandemic. *International Journal of Environmental Research and Public Health*, 18(8), Article 8. <https://doi.org/10.3390/ijerph18084069>.
- Toprak, Ş. (2023). *Makine Öğrenmesi Yöntemlerini Kullanarak bir Petrokimya Firmasının Hisse Senedi Fiyat Tahmini*. Yüksek Lisans Tezi, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Sakarya, 787766.
- Uca, E., Yılmaz, S., Kızıllan, H., & Karaköse, M. (2022). Sağlık Alanındaki Etkileşimlerde Duygu Analizi Yaklaşımları ve Analizi. *Fırat Üniversitesi Uzay ve Savunma Teknolojileri Dergisi*, 1(1), 465-470.
- Uçar, K. T. (2022, Ocak 4). BERT Modeli ile Türkçe Metinlerde Sınıflandırma Yapmak. *Medium*. <https://medium.com/@ktoprakucar/bert-modeli-ile-t%C3%BCrk%C3%A7e-metinlerde-s%C4%B1n%C4%B1fland%C4%B1rma-yapmak-260f15a65611>, (Ziyaret Tarihi: 17 Mart 2023).
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). *Attention Is All You Need* (arXiv:1706.03762). arXiv. <https://arxiv.org/abs/1706.03762>.
- Wang, K., Dou, Y., Sun, T., Qiao, P., & Wen, D. (2022). An Automatic Learning Rate Decay Strategy for Stochastic Gradient Descent Optimization Methods in Neural Networks. *International Journal of Intelligent Systems*, 37(10), 7334-7355. <https://doi.org/10.1002/int.22883>.
- Wang, Y., Zhao, Y., & Addepalli, S. (2021). Practical Options for Adopting Recurrent Neural Network and Its Variants on Remaining Useful Life Prediction. *Chinese Journal of Mechanical Engineering*, 34(1), 69. <https://doi.org/10.1186/s10033-021-00588-x>.
- Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., ... Rush, A. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38-45. <https://doi.org/10.18653/v1/2020.emnlp-demos.6>.

- Yılmaz, M., & Orman, Z. (2021). LSTM Derin Öğrenme Yaklaşımı ile Covid-19 Pandemi Sürecinde Twitter Verilerinden Duygu Analizi. *Acta Infologica*, 5(2), 359-372. <https://doi.org/10.26650/acin.947747>.
- Yurtsever, M., & Çiçekli, U. (2022). *A Novel Hybrid Model Using Arima and LSTM for Sales Forecasting*. 762-776. [https://www.researchgate.net/publication/363058664\\_a\\_novel\\_hybrid\\_model\\_using\\_arima\\_and\\_lstm\\_for\\_sales\\_forecasting](https://www.researchgate.net/publication/363058664_a_novel_hybrid_model_using_arima_and_lstm_for_sales_forecasting).
- Zecevic, A., & Glisovic, N. (2008). Conversion of texts by applying the memory based learning. *2008 6th International Symposium on Intelligent Systems and Informatics*, 1-3. <https://doi.org/10.1109/SISY.2008.4664932>.
- Zeybel Peköz, A., & İnkaya, T. (2023). Derin Öğrenme ile Talep Tahmini: Bir Üçüncü Parti Lojistik Firması için Covid-19 Döneminde Vaka Analizi. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 29(4), 331-339. <https://doi.org/0.5505/pajes.2022.73537>.
- Zhang, K., Ying, H., Dai, H.-N., Li, L., Peng, Y., Guo, K., & Yu, H. (2021). Compacting Deep Neural Networks for Internet of Things: Methods and Applications. *IEEE Internet of Things Journal*, 8(15), 11935-11959. <https://doi.org/10.1109/JIOT.2021.3063497>.

## KİŞİSEL YAYIN VE ESERLER

**Dönmez, H. G., & Becerikli, Y. (2023).** Türkçe Twitter Verileri Üzerinde Duygu Analizi: LSTM, CNN-LSTM, BERT Algoritmalarının Karşılaştırılması, *International Journal of Advanced Natural Sciences and Engineering Researches*, 7(6), 132-138. <https://doi.org/10.59287/ijanser.1146>.



## ÖZGEÇMİŞ

Karabük Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünden 2016 yılında mezun olmuştur. 2017 yılında özel sektörde görev yapmıştır. 2019 yılı itibari ile Milli Savunma Üniversitesi Deniz Harp Okulu Dekanlığı Bilgisayar Mühendisliği Bölüm Başkanlığı'nda Araştırma görevlisi olarak görev yapmaktadır. 2020 yılında Kocaeli Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalında Yüksek Lisans eğitime başlamıştır.

