T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

SOSYAL MEDYADAKİ FİNANS İÇERİKLİ GÖNDERİLERDEN DUYGU SINIFLANDIRMASI

Ahmet Tunahan KORKMAZ

Yüksek Lisans Tezi

YAZILIM MÜHENDİSLİĞİ ANABİLİM DALI

Yazılım Mühendisliği Bilim Dalı

EKİM 2024

T.C.

FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Yazılım Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

SOSYAL MEDYADAKİ FİNANS İÇERİKLİ GÖNDERİLERDEN DUYGU SINIFLANDIRMASI

Tez Yazarı
Ahmet Tunahan KORKMAZ

Danışman
Doç. Dr. Yunus SANTUR

EKİM 2024
ELAZIĞ

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Yazılım Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Başlığı:	Sosyal Medyadaki Finans İçerikli Gönderilerden Duygu Sınıflandırması
Yazarı:	Ahmet Tunahan KORKMAZ
İlk Teslim Tarihi:	14.06.2024
Savunma Tarihi:	18.10.2024

TEZ ONAYI

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

	İmza	
Danışman:	Doç. Dr. Yunus SANTUR Fırat Üniversitesi, Teknoloji Fakültesi	Onayladım
Başkan:	Dr. Öğr. Üyesi Özkan BİNGÖL Gümüşhane Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi	Onayladım
Üye:	Dr. Öğr. Üyesi Özgür KARADUMAN Fırat Üniversitesi, Mühendislik Fakültesi	Onayladım

Bu tez, Enstitü Yönetim Kurulunun ...../...../20..... tarihli toplantısında tescillenmiştir.

İmza

Prof. Dr. Burhan ERGEN
Enstitü Müdürü

BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım "Sosyal Medyadaki Finans İçerikli Gönderilerden Duygu Sınıflandırması" Başlıklı Yüksek Lisans Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

18.10.2024

Ahmet Tunahan KORKMAZ

ÖNSÖZ

Günümüzde sosyal medya, finansal piyasalardaki yatırımcı davranışlarını anlamak için önemli bir veri kaynağı haline gelmiştir. Hisse senedi fiyatları, döviz kurları ve diğer varlıkların değerleri, yatırımcıların sosyal medya üzerinde sergiledikleri duygusal tepkilere büyük ölçüde bağlıdır. Bu nedenle, sosyal medya platformlarından elde edilen duygu verilerinin sınıflandırılması, finansal piyasa analizlerinde etkili bir araç olarak karşımıza çıkmaktadır. Bu çalışmada, sosyal medyada yer alan finans içerikli gönderilerin duygu sınıflandırmasının, yatırım stratejilerinin geliştirilmesinde nasıl kullanılabileceği ele alınmıştır.

Araştırmamın temel amacı, finansal duygu analizinin sosyal medya gönderileri üzerinde uygulanabilirliğini incelemek ve yatırımcı davranışlarını anlamada potansiyelini ortaya koymaktır. Ancak, sosyal medya gönderilerinin içerik çeşitliliği, metinlerin karmaşıklığı ve duygusal verilerin doğruluğu gibi zorluklar bu alandaki çalışmaları etkilemektedir. Bu zorlukların üstesinden gelmek ve güvenilir sonuçlar elde etmek, tezin ana motivasyonları arasında yer almaktadır.

Tezimin hazırlanmasında önemli katkıları bulunan ve bana sürekli rehberlik eden Doçent Dr. Yunus SANTUR'a teşekkürlerimi sunarım. Ayrıca, Araştırma Görevlisi Gamzepelin AKSOY'un da çalışmalarım boyunca sağladığı destekten dolayı kendisine minnettarım. Bu kişilerin desteği olmadan bu çalışmayı tamamlamak mümkün olamazdı.

Tez çalışmam, herhangi bir proje desteği veya etik kurul izni gerektirmemiştir. Ancak, bu süreçte bana destek olan tüm kişi ve kurumlara teşekkür ederim. Çalışmamın, sosyal medyadaki finans içerikli gönderilerin duygu sınıflandırması alanında yapılan araştırmalara katkı sağlaması ve gelecekteki çalışmalara ilham vermesi umuduyla faydalı olmasını dilerim.

Ahmet Tunahan KORKMAZ
ELAZIĞ, 2024

İÇİNDEKİLER

Sayfa

ÖZET

Sosyal Medyadaki Finans İçerikli Gönderilerden Duygu Sınıflandırması

Ahmet Tunahan KORKMAZ

Yüksek Lisans Tezi

FIRAT ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Yazılım Mühendisliği Anabilim Dalı

Ekim 2024, Sayfa: xii + 107

Günümüz dijital çağında sosyal medya, finansal piyasalarda yatırımcı davranışlarını ve piyasa trendlerini anlamak için önemli bir veri kaynağı haline gelmiştir. Bu çalışmada, Twitter verileri ve borsa hisselerinin fiyatları kullanılarak sosyal medya içeriklerinin finans piyasaları üzerindeki etkileri incelenmiş ve bu etkilerin duygu temelli analizlerle piyasa üzerinde nasıl kullanılabileceği değerlendirilmiştir. 2023 yılı Temmuz ayından 2024 yılı Temmuz ayına kadar #xu100 ve #bist100 etiketleriyle toplanan 10.505 tweet, yorum sayısı, retweet, beğeni, görüntüleme sayısı, zaman, tarih ve hisse fiyatı gibi özelliklerle birlikte analiz edilmiştir. Doğal dil işleme (DDİ) teknikleri kullanılarak her bir tweet pozitif, negatif ve nötr olarak sınıflandırılmıştır. Sonuçlar, pozitif duygu içeren tweetlerin hisse fiyatlarında artış, negatif duygu içerenlerin ise düşüş eğiliminde olduğunu göstermektedir. LSTM ve GRU modelleri ile üçlü sınıflandırma yapılmış; üçlü sınıflandırma modellerinde eğitim doğruluğu yüksek olmasına rağmen test doğruluğunda aşırı öğrenme problemi gözlemlenmiştir. Model performansları kesinlik, duyarlılık ve F1 skoru ile değerlendirilmiş; negatif sınıfların tanımlanmasında zorluklar olduğu belirlenmiştir. Bu bulgular, finansal piyasa hareketlerinde duygu sınıflandırmasının potansiyelini ortaya koymakta ve gelecekteki model geliştirmeleri için öneriler sunmaktadır.

Anahtar Kelimeler: Sosyal Medya, Duygu Sınıflandırma, Metin Madenciliği, Doğal Dil İşleme, Derin Öğrenme

ABSTRACT

Sentiment Classification from Financial Content Posts on Social Media

Ahmet Tunahan KORKMAZ

Master's Thesis

FIRAT UNIVERSITY
Graduate School of Natural and Applied Sciences

Department of Software Engineering

October 2024, Pages: xii + 107

In today's digital era, social media has become an important data source for understanding investor behavior and market trends in financial markets. In this study, Twitter data and stock prices were used to examine the impact of social media content on financial markets, and the potential use of sentiment-based analysis in influencing the market was evaluated. A total of 10,505 tweets tagged with #xu100 and #bist100 were collected from July 2023 to July 2024 and analyzed along with features such as comment count, retweets, likes, views, time, date, and stock price. Using Natural Language Processing (NLP) techniques, each tweet was classified as positive, negative, or neutral. The results indicated that tweets with positive sentiment were associated with increases in stock prices, while tweets with negative sentiment showed a tendency toward declines. Three-class classification was conducted using LSTM and GRU models; although training accuracy was high in three-class classification models, an overfitting problem was observed in test accuracy. Model performance was evaluated using precision, recall, and F1 score metrics, and difficulties were identified in recognizing negative classes. These findings highlight the potential of sentiment classification in tracking financial market movements and provide suggestions for future model improvements.

Keywords: Social Media, Sentiment Classification, Text Mining, Natural Language Processing, Deep Learning

ŞEKİLLER LİSTESİ

Sayfa

TABLOLAR LİSTESİ

Sayfa

SİMGELER VE KISALTMALAR

Kısaltmalar

Adam	: Adaptive Moment Estimation (Adaptif Moment Tahmini)
ARI	: Automatic Readability Index (Otomatik Okunabilirlik İndeksi)
ARIMA	: Autoregressive Integrated Moving Average (Otokorelasyonlu Entegre Hareketli Ortalama)
BERT	: Bidirectional Encoder Representations from Transformers (İki Yönlü Kodlayıcı Temsillerinden Dönüşüm)
BoW	: Bag of Words (Kelime Çantası)
BTM	: Belge Terim Matrisi
CNN	: Convolutional Neural Networks (Evrişimsel Sinir Ağları)
CSV	: Comma-Separated Values (Virgülle Ayrılmış Değerler)
DSSCs	: Boya duyarlı güneş pilleri
ECB	: European Central Bank (Avrupa Merkez Bankası)
GRU	: Gated Recurrent Unit (Kapılı Tekrarlayan Birim)
GARCH	: Generalized Autoregressive Conditional Heteroskedasticity (Genelleştirilmiş Otokorelasyonlu Koşullu Değişkenlik)
HTML	: HyperText Markup Language (Hiper Metin İşaretleme Dili)
LSTM	: Long Short-Term Memory (Uzun Kısa Süreli Bellek)
MLM	: Masked Language Model (Maskelenmiş Dil Modeli)
mT5	:Multilingual Text-To-Text Transfer Transformer (Çok Dilli Metin-Üzerine-Metin Transfer Dönüşümcü)
NER	: Named Entity Recognition (İsim Varlık Tanıma)
DDİ	: Natural Language Processing (Doğal Dil İşleme)
NSP	: Next Sentence Prediction (Sonraki Cümle Tahmini)
PDF	: Portable Document Format (Taşınabilir Belge Formatı)
PLoS	: Public Library of Science (Bilim Halk Kütüphanesi)
RNN	: Recurrent Neural Network (Tekrarlayan Sinir Ağı)
RO	: Rastgele Orman Algoritması
SSRN	: Social Science Research Network (Sosyal Bilim Araştırma Ağı)
DVM	: Support Vector Machines (Destek Vektör Makineleri)
T5	: Text-To-Text Transfer Transformer (Metin-Üzerine-Metin Transfer Dönüşümcü)
Word2Vec	: Word to Vector (Kelimeden Vektöre)

1. GİRİŞ

Doğal Dil İşleme (DDİ), haber makaleleri, sosyal medya gönderileri ve finansal raporlar gibi yapılandırılmamış finansal verilerin analizine olanak tanıdığından, finans alanında giderek daha önemli bir araştırma alanı haline gelmiştir. Bu gelişmelerle birlikte metin madenciliği, duygu analizi gibi kavramlarda ön plana çıkmıştır.

DDİ, gündelik hayatımızda konuşulan doğal dili anlamayı, analiz etmeyi ve bunu makine diline dönüştürmeyi amaçlayan, dilbilim, bilgisayar bilimi ve yapay zeka alanlarını içeren bir alandır. DDİ süreci insan dilindeki metin ya da konuşma verilerini istatistik, hesaplamalı dilbilim, derin öğrenme ve makine öğrenimi gibi yaklaşımlar ile birleştirir. DDİ semantik, dilbilimsel ve pragmatik yönlerin çeşitli türevlerini analiz ederek insan doğal dilinin yapısını ve anlamını anlamamızı sağlar. Bu sürecin devamı olarak, dil bilgisi kurallarını makinelerin anlayabileceği düzeyde makine algoritmalarına çevirir. Bunun sonucunda farklı problemleri çözme yetisine sahip ve istenilen görevleri yerine getirebilecek kapasitede kural tabanlı makine öğrenimi kavramı ortaya çıkar.

Duygu analizi, dil modelleme ilerlemelerinden yararlanan ve daha iyi sonuçlar elde eden DDİ görevlerinden biridir. Duygu analizi, bir metinde ifade edilen görüşleri hesaplama ile tanımlanması ve kategorize edilmesi sürecidir, esas olarak yazarın belirli bir konu veya ürüne yönelik tutumunun pozitif, negatif veya nötr olup olmadığını belirlemek için kullanılır. Duygu analizi, duyguları ve tutumları işlenebilir bilgiye dönüştürmek için önemli bir araç haline gelmektedir.

Bu ilişki analizinde varlık alım ve satışlarında insanların özellikle twitter adlı sosyal medya platformu başta olmak üzere finansal haber kaynaklarından edinilen bilgilerin finansal kararlar alırken oldukça etkili bir rol oynadığı görülmektedir. Yatırımcılar özellikle varlığın fiyatındaki ani değişimi, finansal bilanço açıklaması ya da farklı bir sebepten oluşan haberler ve duyum kalabalığından duygusal bir etkileşim yaşayarak kendi finansal kararlarını bu duygusal ortama göre verdiği ortaya çıkmaktadır.

Finansal alan, kendine özgü bir kelime dağarcığı ile karakterize edilir, bu da alan özgü duygu analizi gerektirir. Finansal piyasalarda gözlemlenen fiyatlar, işlem gören varlıklara ilişkin tüm mevcut bilgileri yansıtır. Dolayısıyla yeni bilgiler paydaşların bilinçli ve zamanında kararlar almasını sağlar. Haberlerde ve tweetlerde ifade edilen duygular, hisse senedi fiyatlarını ve marka itibarını etkiler, bu nedenle bu duyguların sürekli ölçülmesi ve izlenmesi yatırımcılar için en önemli faaliyetlerden biri haline gelmektedir. Çalışmalar, finansal haberlere dayalı duygu analizinin hisse senedi fiyatlarını, döviz kurlarını ve küresel finansal piyasa eğilimlerini tespit etmek için kullanıldığı gibi, şirket kazançlarını öngörmek için kullanıldığını göstermektedir.

Tüm bu ilişkilerin sonucunda bu tezin ortaya çıkmasında hedef alınan problem, insanların finansal seçimlerinde verdiği duygusal kararların endeks, hisse vb. finansal araçlardaki etkileri incelenmektedir.

1.1. Problemin Önemi

Son yıllarda teknolojinin hızlı gelişimi, internet erişiminin yaygın kullanımı, çevrimiçi ortamda büyük miktarda veri birikmesine neden olmuştur. Finansal raporlama ve veri analizi, büyük ölçüde teknolojik ilerlemelerin hızlı tempolu değişimi tarafından büyük bir dönüşüm geçirdi. Bu değişiklikler, finansal analizde kullanılan araçları ve metodolojileri sadece yeniden şekillendirmekle kalmadı, aynı zamanda finansal raporlamanın kendisinin doğasını da yeniden tanımladı. Bu evrimin temelinde, finansal veri analizinin verimliliğini, doğruluğunu ve derinliğini önemli ölçüde artıran DDİ ve diğer gelişmiş hesaplama tekniklerinin entegrasyonu bulunmaktadır.

Finansal raporlamada DDİ tekniklerinin entegrasyonu, finansal analizde otomasyon ve geliştirmenin önemli bir adımını temsil eder. Bu evrim, finansal verilerin işlenmesi, analizi ve yorumlanmasına benzersiz katkılar sağlayan çeşitli DDİ metodolojilerinin geliştirilmesi ve uygulanmasıyla işaretlenmiştir. Aşağıda yer alan çalışmalar, finansal raporlamayı otomatikleştirmede merkezi öneme sahip olan temel DDİ tekniklerine dalmakta ve son zamanlardaki bilimsel katkılarla desteklenmektedir.

Faccia vd., yaptığı çalışmada duygu analizini, finansal açıklamaların tonunu ve duygusunu değerlendirmede kritik bir DDİ alanı olarak öne çıkarmıştır. Çalışmaları, duygu analizinin finansal tablolardaki anormallikleri ve desenleri tespit etme potansiyelini vurgularken dolandırıcılık faaliyetlerini işaret edebileceğini göstermektedir. Finansal raporlardaki metinsel verilerin polaritesini ve öznelliğini analiz ederek, duygu analizi, geleneksel nicel finansal analiz yöntemlerini tamamlayarak altta yatan duygulara dair nüanslı bir anlayış sunmaktadır [1].

Tematik analiz, bulut bilişim ortamlarında finansal verilere etkili bir şekilde uygulanmış bir diğer temel DDİ tekniğidir, ki bu Sharma vd. tarafından tartışılmıştır. Bu yaklaşım, metin verilerindeki temaları veya desenleri tanımlamayı ve analiz etmeyi içerir, finansal anlatıların daha derinlemesine anlaşılmasını kolaylaştırır. Tematik analiz, finansal verilerin kategorize edilmesini ve kalite ölçümünü artırarak, finansal kurumlardaki veri keşfi ve analiz süreçlerini geliştirir [2] .

Gurgul vd., yaptıkları çalışmada DDİ yöntemlerinin finansal piyasa hareketlerini öngörmede nasıl kullanıldığını, özellikle de kripto para fiyatları bağlamında göstermektedir. Araştırmaları, finansal, blokzincir ve metinsel verilerin gelişmiş DDİ teknikleri aracılığıyla entegrasyonunun, yatırım kararlarının doğruluğunu önemli ölçüde artırabileceğini göstermektedir. Bu uygulama, DDİ'nin kapsamlı finansal analiz için çeşitli veri kaynaklarını kullanma kapasitesini vurgulamaktadır [3].

DDİ tekniklerinin finansal raporlamada otomatikleştirilmesi, finansal analiz alanında dönüştürücü bir gelişmeyi temsil eder. Duygu analizi, tematik analiz ve diğer DDİ metodolojilerinden yararlanarak, finansal kurumlar finansal raporlama süreçlerinde daha büyük verimlilik, doğruluk ve derinlik elde edebilirler. DDİ teknolojileri evrimleşmeye devam ettikçe, finansal raporlamaya entegrasyonları, finans profesyonellerine sunulan analitik yetenekleri ve stratejik iç görüleri daha da geliştirmeye yöneliktir.

DDİ'nin finansal raporlama için uygulanması, dönüştürücü bir gelişme olmasına rağmen, zorluklar ve sınırlamalar içermektedir. Bu engeller, finansal anlatıların içsel karmaşıklıklarından ve DDİ teknolojilerinin teknik kısıtlamalarından kaynaklanmaktadır. Bu zorlukların araştırılması, DDİ' in finansal raporlama ve analize entegrasyonunun pratik sonuçlarını anlamak için önemlidir.

Azizov vd., birden fazla dildeki yıllık raporlardan finansal anlatıları özetlemenin karmaşıklıkları üzerine bir çalışma gerçekleştirmiştir. Çalışmaları, yapılandırılmamış farklı finansal raporları işlemenin zorluğunu vurgulayarak, sayısal verileri dışlayarak ana anlatı ögelerini etkin bir şekilde belirlemenin güçlüğünü ortaya koymaktadır. Bu zorluk, çok dilli içerikle uğraşırken, T5 ve mT5 gibi DDİ tekniklerinin etkinliğinin diller arasında önemli ölçüde değişiklik göstermesi nedeniyle daha da artmaktadır. Bu durum, finansal raporlama alanındaki dil çeşitliliğinin karmaşık zorluklarını yansıtmaktadır [4].

1.2. Çalışmanın Amaçları ve Hipotezleri

Finans alanında, hisse senedi piyasası ve eğilimleri son derece değişken niteliktedir. Bu değişkenliği anlamak ve gelecekteki hareketleri öngörmek araştırmacıların dikkatini çeken bir konudur. Yatırımcılar ve piyasa analistleri, piyasa davranışlarını inceleyerek alım veya satım stratejilerini buna göre planlar. Hisse senedi piyasası her gün büyük miktarda veri üretir, bu nedenle bir kişinin bir hissenin gelecekteki yönelimini belirlemek için tüm mevcut ve geçmiş bilgileri dikkate alması oldukça zordur. Genellikle piyasa yönelimlerini anlamak için iki yöntem vardır. Birincisi teknik analiz, diğeri ise temel analizdir. Teknik analiz, gelecekteki hareketi öngörmek amacıyla geçmiş fiyat ve hacmi incelerken, temel analiz işletmenin finansal verilerini değerlendirerek bazı iç görüler elde etmeyi içerir. Her iki teknik analizin etkinliği, hisse senedi piyasası fiyatlarının temel olarak öngörülemeyen olduğunu belirten etkin piyasa hipotezi tarafından tartışılmaktadır. Bu araştırma, hisse senedi eğilimlerinin keşfedilmesi için temel analiz tekniğini takip eder ve bir şirket hakkında haber makalelerini ana bilgi olarak ele alarak haberleri kullanılan modele göre ikili (pozitif-negatif) ya da üçlü (pozitif-negatif-nötr) sınıflandırmayı dener. Eğer haberin duygusu olumlu ise, hisse senedi fiyatının artma olasılığı daha yüksektir ve eğer haberin duygusu negatif ise, o zaman hisse senedi fiyatının düşme olasılığı vardır. Bu araştırma, hisse senedi trendlerini etkileyebilecek haber duyarlılığını duygu puanlaması ile tespit eden bir model

oluşturmayı amaçlar. Diğer bir deyişle, haber makalelerinin hisse senedi fiyatları üzerindeki etkisini incelemektedir. Haber duyarlılığını kontrol etmek için denetimli makine öğrenimi olarak sınıflandırma ve diğer metin madenciliği tekniklerini kullanılmaktadır.

Araştırmanın amacı:

1. Metin analizinin finansal piyasalardaki rolünü ve önemini belirlemek.
2. Metin madenciliği ve duygu analizi yöntemlerinin finansal yatırımların değerlendirilmesindeki etkinliğini araştırmak.
3. Finansal kararlar alırken duygusal faktörlerin etkisini daha iyi anlamak ve değerlendirmek için bir araç sağlamak.

Araştırma hipotezleri:

1. Metin analizi ve duygu analizi kullanılarak finansal piyasalardaki duygusal eğilimlerin belirlenmesi, finansal yatırımlarda daha doğru tercihler ortaya çıkarmayı sağlayabilir.
2. Metin madenciliği ve duygu analizi yöntemleriyle, finansal piyasalardaki haber akışının ana trendlerinin belirlenmesi, tweetleri domine eden ana haberleri daha olası bir şekilde ortaya çıkarabilir.
3. Metin analizi ve duygu analizi ile yapılan çalışmalar, oluşturulan finansal tweet kümeleri arasındaki yapı ve benzerliklerin belirlenmesinde daha etkili olabilir.

1.3. Kullanılan Metotlar

Tezde kullanılan metotlar şu şekilde açıklanabilir: Veri setinde yer alan metin verilerinin analizinde çeşitli sinir ağı algoritmaları ve DDİ teknikleri yer almaktadır. Bu işlemler, metin verilerinin işlenmesi, etiketlenmesi ve sınıflandırılması aşamasında kullanılmaktadır.

Metin verileri işlenmeden önce bir dizi metin ön işleme adımından geçirilmektedir. Bu ön işlemler metin temizleme, metin normalizasyonu, durak kelimeleri kaldırma, metin tokenizasyonu, metin vektörleştirme vb. şeklindedir. Bu tezde özel karakterler, HTML etiketleri gibi gürültülü metin dediğimiz kelimeler için metin temizleme uygulanmıştır. Tek başına bir anlam ifade etmeyen ve, veya, şu vb. kelimelerde model performansını etkilememesi adına kaldırılmıştır. Metin verilerinin kelimeler ya da cümleler seviyesine indirgenmesi ve sonrasında bunların sayısal vektörler haline getirilmesi için tokenizasyon ve vektörleştirme adımları da uygulanmıştır.

Metin verilerini işlerken gömme katmanından faydalanırız. Oluşturulan metin sınıflandırma modelinde birbirine bağlı diğer katmanlardan önce gelir. Bu katman kelimeleri makine dilinin anlayabileceği düzeyde bir vektör uzayına yerleştirmek için kullanılır. Bu vektörler, kelimelerin yapısal ve anlamsal ilişkilerini daha verimli bir şekilde ifade etmeye ve daha etkili işlemler yapılmasını sağlamaktadır.

Model eğitilirken ikili ve üçlü olmak üzere iki farklı sınıflandırma uygulanmıştır. İkili sınıflandırmada "ikili çapraz entropi" kayıp fonksiyonu kullanılmıştır bu da modelin doğruluğuna artırmaktadır. Bir diğer kayıp fonksiyonu da çoklu sınıflandırma problemlerinde kullanılan "seyrek kategorik çapraz entropi" fonksiyonudur. Bu fonksiyon pozitif, negatif ve nötr gibi üç durumlu seçimlerde uygundur.

GRU ve LSTM, Tekrarlayan Sinir Ağlarının (RNN) zaman serisi verilerinde verimli bir şekilde kullanıldığını kanıtlamasına karşın, gradyanların kaybolması ve şişmesi sorunu LSTM in ortaya çıkmasına neden olmuştur. Çok sayıda parametreye sahip olan LSTM, kısa ve uzun vadeli bağımlılıkların öğrenilmesinde daha fazla eğitim süresine sahiptir bu da LSTM i dezavantajlı duruma düşürmüş ve daha küçük kapısı olan (yalnızca güncelleme ve sıfırlama) dolayısıyla eğitim süresi nispeten daha az ve verimliliği daha fazla olan GRU ları ortaya çıkarmıştır.

GRU, daha basit hale getirilmiş bir LSTM türevi olarak kabul edilir. İç yapısına baktığımızda LSTM den daha az bileşen daha az kapı barındırdığından daha az karmaşık bir yapıya sahiptir. GRU da iki kapı (güncelleme ve sıfırlama) bulunur, LSTM de ise üç kapı bulunur (unutma, giriş ve çıkış). GRU da daha az bileşene sahip olduğundan, eğitim esnasında daha az veri ile daha iyi performans göstermektedir. Bununla paralel olarak LSTM in karmaşık iç yapısı da bazı uzun süreli bağımlılık durumlarında daha iyi sonuçlar verebilmektedir.

Bu iki yapının yanı sıra son zamanlarda ortaya çıkan yeni bir DDİ modeli olan BERT dil modeli de bu tezde kullanılmıştır. Bu model daha önceden büyük metin verileriyle eğitilir ve akabinde belirli ayarlarla istediğimiz veri setine uygulanabilir. Bu özelliğiyle transformer mimarisine dayanmaktadır.

BERT in GRU ya da LSTM e karşı avantajları:

- Diğer iki yapıya göre daha büyük veri setleri üzerinde eğitildiğinden dili anlamadaki yatkınlığı daha fazladır.
- Transfer öğrenimi özelliği ile küçük çaplı veri setlerinde dahi faydalı sonuçlar verebilmektedir.

BERT in GRU ya da LSTM e karşı dezavantajları:

- Diğer iki geleneksel modele göre daha fazla hesaplama süresi ihtiyacı duymaktadır.
- Temel olarak büyük veri setlerinde eğitildiğinden, kendi veri setlerimize uygulama aşamasında ölçeklenebilirlik açısından sorunlar ortaya çıkarabilmektedir.
- Yapı olarak büyük dil modelleri üzerinde geliştirildiğinden küçük veri setlerinde daha düşük performans sergileyebilmektedir.

Modelin eğitiminde farklı epoch sayıları ve farklı hiperparametler kullanılarak farklı performans ve eğitim süreçleri gözlenmiştir.

1.4. Akademiye Katkısı

Bu çalışmanın akademik literatüre sağlayacağı katkılar aşağıdaki başlıklar altında değerlendirilebilir:

Bu tez, DDİ ve finansal veri analizi arasındaki ilişkileri derinlemesine inceleyerek, bu iki disiplinin entegrasyonunu sağlamaktadır. Çalışma, DDİ tekniklerinin finansal analiz ve raporlamada nasıl kullanılabileceğini göstererek, bu alandaki teorik bilgileri genişletmektedir. Özellikle duygu analizi, tematik analiz ve derin öğrenme yöntemlerinin finansal veri analizi üzerindeki etkilerini araştırarak, akademik literatürde bu konulardaki boşlukları doldurmaktadır.

Tezde kullanılan çeşitli makine öğrenimi algoritmaları (LSTM, GRU, BERT) ve DDİ teknikleri (metin ön işleme, tokenizasyon, vektörleştirme) finansal verilerin analizine uygulanmış ve bu yöntemlerin etkinlikleri karşılaştırılmıştır. Bu yöntemsel karşılaştırmalar, akademik çalışmalarda kullanılabilecek yeni metotların belirlenmesine ve mevcut yöntemlerin iyileştirilmesine katkı sağlayacaktır. Ayrıca, metin madenciliği ve duygu analizi tekniklerinin finansal yatırımlardaki etkinliğini gösteren bulgular, bu yöntemlerin finansal analizlerde kullanımının artmasına yol açabilir.

Bu çalışma, finansal metinlerin (haber makaleleri, tweetler) analiz edilmesi ve bu analizlerin finansal karar alma süreçlerine olan etkilerini inceleyerek pratik uygulamalar sunmaktadır. Özellikle, yatırımcıların finansal kararlar alırken daha sistematik yardım alabileceği bir yöntem tespit edilmesi denenmiştir. Bu uygulamalar, hem akademik araştırmalarda hem de finansal piyasalarda pratik olarak kullanılabilir ve bu alanlardaki çalışmalara yeni bir perspektif kazandırabilir.

Bu tezde geliştirilen modeller ve kullanılan yöntemler, DDİ ve finansal veri analizi konularında ders materyali olarak kullanılabilir. Üniversitelerde lisans ve lisansüstü düzeyde verilen derslerde bu çalışma örnek bir vaka olarak sunulabilir ve öğrencilere teorik bilgilerin pratik uygulamalarla nasıl birleştiğini gösteren bir kaynak sağlayabilir.

Çalışma, DDİ tekniklerinin finansal veri analizinde nasıl uygulanabileceğine dair yeni araştırma soruları ve hipotezler ortaya koymaktadır. Bu tezde elde edilen bulgular ve sonuçlar, gelecekteki çalışmalara yön verebilir ve bu alandaki araştırmaların derinleştirilmesine katkıda bulunabilir. Özellikle, DDİ ve finansal veri analizi entegrasyonunda karşılaşılan zorluklar ve bu zorlukların nasıl aşılabileceği konusundaki öneriler, akademik araştırmalar için değerli ipuçları sunmaktadır.

Bu katkılar, hem teorik hem de uygulamalı olarak DDİ ve finansal veri analizi alanlarında önemli bir yere sahip olup, akademik literatüre değerli bir katkı sağlayacaktır.

2. LİTERATÜR TARAMASI

Çalışmanın literatür araştırması kısmında öncelikle farklı konular için gerçekleştirilen DDİ ile ilgili literatür çalışmaları ele alınmıştır. Bu kısımda alt başlık olarak metin madenciliği ve duygu analizi çalışmaları da ele alınmıştır. Son olarak finans alanında ki çalışmalardan örnekler verilmiştir.

2.1. Doğal Dil İşleme

Doğal dil işleme, insan dilinin bilgisayarlar tarafından anlaşılması, yorumlanması ve üretilmesi sürecidir. Bu alan, dil bilimi, bilgisayar bilimi ve yapay zekanın kesişiminde yer alır ve birçok farklı uygulama ve teknik içerir. DDİ, metin madenciliği, makine çevirisi, duygu analizi ve bilgi çıkarımı gibi alanlarda geniş uygulamalara sahiptir.

Metin madenciliği, büyük miktarda metin verisinin analiz edilmesini ve anlamlı bilgilerin çıkarılmasını sağlar. Örneğin, sosyal medya analizleri, müşteri geri bildirimleri ve haber analizleri gibi alanlarda metin madenciliği kullanılmaktadır [5]. Metin madenciliği, metinlerin otomatik olarak sınıflandırılması, özetlenmesi ve önemli bilgilerin çıkarılması işlemlerini içerir [6]. Bu süreçte kullanılan algoritmalar, metinleri doğal dilde anlayabilmek için istatistiksel ve dilbilimsel modellerden yararlanır [7].

Makine çevirisi, farklı diller arasında metin çevirisi yapma sürecidir. Google Translate ve benzeri hizmetler, makine çevirisi için DDİ tekniklerini kullanır [8]. Makine çevirisi, istatistiksel ve kural tabanlı yöntemlerin yanı sıra, son yıllarda derin öğrenme yöntemleriyle de büyük ilerlemeler kaydetmiştir [9]. Bu yöntemler, dil çiftleri arasındaki karmaşık ilişkileri öğrenerek daha doğru ve doğal çeviriler yapabilmektedir [10].

Duygu analizi, metinlerde ifade edilen duyguların tespit edilmesi ve sınıflandırılması sürecidir. Bu teknik, müşteri yorumları, sosyal medya gönderileri ve haber metinlerinde kullanıcıların duygu durumlarını analiz etmek için kullanılır [11]. Örneğin, bir şirketin ürünlerine ilişkin müşteri geri bildirimlerini analiz ederek, genel müşteri memnuniyetini ve ürünlerle ilgili olası sorunları tespit edebilir [12]. Bu tür analizler, işletmelerin stratejik kararlar almasına yardımcı olabilir [13].

Bilgi çıkarımı, metinlerdeki önemli bilgilerin tanımlanması ve yapılandırılması sürecidir. Örneğin, isim-entity tanıma (Named Entity Recognition - NER) algoritmaları, metinlerdeki kişi, yer ve organizasyon isimlerini tespit eder [14]. Bu teknikler, büyük veri setlerinden anlamlı bilgiler elde etmek için kritik öneme sahiptir [15]. Bilgi çıkarımı, özellikle büyük miktarda dokümanın otomatik olarak analiz edilmesi gereken alanlarda kullanılır [16].

DDİ teknikleri, sağlık hizmetlerinde de geniş bir uygulama alanına sahiptir. Elektronik sağlık kayıtları, tıbbi literatür ve hasta geri bildirimleri gibi büyük miktarda metinsel verinin analiz

edilmesi, tıbbi bilgi çıkarımı ve karar destek sistemleri için kritik öneme sahiptir [17]. Örneğin, tıbbi makalelerin otomatik olarak analiz edilmesi, doktorların en güncel bilgileri hızlıca bulmasına yardımcı olabilir [18]. Ayrıca, hastaların sağlık kayıtlarının analiz edilmesi, hastalıkların erken teşhisi ve tedavi planlarının geliştirilmesine katkı sağlayabilir [19].

DDİ'nin eğitim alanındaki uygulamaları da giderek artmaktadır. Öğrenci yazılarının otomatik olarak değerlendirilmesi, dil öğrenme uygulamaları ve akıllı öğretim sistemleri gibi alanlarda DDİ teknikleri kullanılmaktadır [20]. Bu teknolojiler, öğretmenlerin yükünü hafifletmek ve öğrencilerin dil becerilerini geliştirmek için önemli fırsatlar sunar [21]. Örneğin, otomatik yazım değerlendirme sistemleri, öğrencilerin yazım hatalarını tespit ederek geri bildirim sağlar.

Son olarak, DDİ, hukuki belge analizleri gibi alanlarda da önemli bir rol oynamaktadır. Büyük miktarda hukuki belgenin otomatik olarak analiz edilmesi ve ilgili bilgilerin çıkarılması, avukatların ve hukuk uzmanlarının iş yükünü azaltabilir. Bu tür sistemler, hukuki metinlerdeki önemli bilgileri tespit ederek, dava hazırlığı ve yasal araştırmalar gibi süreçleri hızlandırabilir [22].

DDİ, aynı zamanda müşteri hizmetleri ve destek sistemlerinde de kullanılmaktadır. Örneğin, chatbotlar ve sanal asistanlar, kullanıcıların sorularını yanıtlamak ve belirli görevleri yerine getirmek için DDİ tekniklerinden yararlanır. Bu tür sistemler, müşteri etkileşimlerini otomatikleştirerek işletmelerin maliyetlerini düşürmesine ve müşteri memnuniyetini artırmasına yardımcı olur [23].

DDİ'nin bir diğer önemli uygulama alanı, bilgi erişimi ve arama motorlarıdır. Arama motorları, kullanıcıların sorgularını anlamak ve en uygun sonuçları sağlamak için DDİ tekniklerini kullanır [24]. Bu süreçte, sorgu genişletme, anlamsal analiz ve metin sınıflandırma gibi teknikler devreye girer [25]. Bu sayede, arama motorları daha doğru ve ilgili sonuçlar sunabilir [26].

Dil üretimi, DDİ'nin önemli bir bileşenidir. Dil üretimi, bilgisayarların doğal dilde metin oluşturma sürecidir. Bu süreçte, dil modelleri ve derin öğrenme algoritmaları kullanılarak anlamlı ve bağlama uygun metinler üretilir [27]. Dil üretimi, chatbotlar, sanal asistanlar ve otomatik içerik oluşturma sistemleri gibi birçok uygulama için kritik öneme sahiptir [28].

DDİ, aynı zamanda sosyal medya analizlerinde de kullanılmaktadır. Sosyal medya platformlarındaki büyük miktarda verinin analiz edilmesi, kullanıcı davranışlarının ve trendlerin tespit edilmesine yardımcı olur [29]. Bu süreçte, duygu analizi, konu modelleme ve metin sınıflandırma gibi teknikler kullanılır. Sosyal medya analizleri, işletmelerin pazarlama stratejilerini belirlemesine ve müşteri memnuniyetini artırmasına katkı sağlar [30].

DDİ, haber ve medya sektöründe de geniş bir uygulama alanına sahiptir. Haber metinlerinin otomatik olarak analiz edilmesi ve özetlenmesi, okuyucuların hızlı ve etkili bir şekilde bilgiye

erişmesini sağlar [31]. Ayrıca, sahte haberlerin tespit edilmesi ve doğruluk kontrolü gibi uygulamalarda da DDİ teknikleri kullanılmaktadır [32].

DDİ, aynı zamanda e-ticaret sektöründe de önemli bir rol oynamaktadır. Ürün incelemelerinin analiz edilmesi, müşteri geri bildirimlerinin toplanması ve ürün öneri sistemlerinin geliştirilmesi gibi uygulamalarda DDİ tekniklerinden yararlanılır [33]. Bu sayede, e-ticaret platformları müşterilere daha kişiselleştirilmiş ve ilgili ürün önerileri sunabilir [34].

Sonuç olarak, DDİ, birçok farklı alanda geniş uygulama yelpazesine sahip bir teknolojidir. Metin madenciliği, makine çevirisi, duygu analizi, bilgi çıkarımı, sağlık hizmetleri, eğitim, hukuki belge analizi, müşteri hizmetleri, bilgi erişimi, dil üretimi, sosyal medya analizi, haber ve medya, ve e-ticaret gibi alanlarda DDİ teknikleri kullanılmaktadır. Bu teknolojilerin gelişmesi, bilgisayarların insan dilini daha iyi anlamasını, işlemesini ve üretmesini sağlamaktadır.

2.1.1. Metin Madenciliği

Metin madenciliği, büyük miktarda metin verisinden anlamlı ve faydalı bilgiler elde etmeyi amaçlayan bir veri madenciliği alt dalıdır. Metin madenciliği, veri madenciliği, DDİ ve bilgi çıkarımı gibi disiplinlerin kesişiminde yer alır ve birçok farklı teknik ve yöntem kullanılarak gerçekleştirilir. Bu yazıda, metin madenciliğinin temel kavramları, kullanılan teknikler ve çeşitli uygulama alanları ele alınacaktır.

Metin madenciliği, yapılandırılmamış metin verilerini yapılandırılmış bilgiye dönüştürmeyi hedefler. Bu süreç, metinlerin otomatik olarak sınıflandırılması, özetlenmesi ve önemli bilgilerin çıkarılması işlemlerini içerir . Yapılandırılmamış veriler, doğal dilde yazılmış metinlerdir ve bu tür verilerin analiz edilmesi, anlamlı bilgiler elde edilmesi açısından zorluklar içerir [35]. Metin madenciliği, bu zorlukları aşmak için çeşitli algoritmalar ve teknikler kullanır [36].

Metin madenciliğinde kullanılan temel teknikler arasında bilgi çıkarımı, metin sınıflandırma, kümeleme ve özetleme yer alır. Bilgi çıkarımı, metinlerdeki belirli bilgilerin otomatik olarak tespit edilmesini sağlar [37]. Metin sınıflandırma, metinlerin önceden tanımlanmış kategorilere ayrılmasını içerir. Kümeleme, benzer özelliklere sahip metinlerin gruplandırılmasını sağlar. Özetleme ise, metinlerin ana noktalarını belirleyerek kısa ve öz bir özet oluşturmayı amaçlar [38].

Metin madenciliğinin en yaygın uygulama alanlarından biri, sosyal medya analizidir. Sosyal medya platformlarındaki büyük miktarda verinin analiz edilmesi, kullanıcı davranışlarının ve trendlerin tespit edilmesine yardımcı olur [39]. Bu analizler, markaların ve işletmelerin müşteri ilişkilerini yönetmelerine ve pazarlama stratejilerini optimize etmelerine katkı sağlar [40]. Ayrıca, kriz yönetimi sırasında, sosyal medya analizleri, halkın genel tepkilerini ve endişelerini anlamak için kullanılabilir [41].

Metin madenciliği, müşteri geri bildirimlerinin analiz edilmesinde de önemli bir rol oynar. Müşteri yorumları ve geri bildirimleri, işletmelerin ürün ve hizmetlerini geliştirmeleri için kritik bilgiler sunar [42]. Bu tür analizler, müşteri memnuniyetini artırmak ve olası sorunları tespit etmek için kullanılır [43]. Örneğin, ürün incelemelerinin analiz edilmesi, işletmelerin müşteri ihtiyaçlarını ve beklentilerini daha iyi anlamalarına yardımcı olabilir [44].

Eğitim sektöründe, metin madenciliği, öğrenci geri bildirimlerinin analiz edilmesi ve eğitim materyallerinin geliştirilmesi için kullanılır. Öğrenci yorumları ve geri bildirimleri, öğretim yöntemlerinin ve müfredatın değerlendirilmesine katkı sağlar [45]. Ayrıca, akademik makalelerin ve yayınların analiz edilmesi, araştırma trendlerinin ve önemli konuların belirlenmesine yardımcı olabilir.Sağlık sektöründe, metin madenciliği, hasta kayıtlarının ve tıbbi literatürün analiz edilmesi için kullanılır [46]. Bu analizler, hastalıkların erken teşhisi ve tedavi planlarının geliştirilmesine katkı sağlar. Ayrıca, tıbbi makalelerin ve araştırma raporlarının analiz edilmesi, doktorların ve araştırmacıların en güncel bilgilere hızlıca erişmelerini sağlar [47].

Hukuk sektöründe, metin madenciliği, hukuki belgelerin analiz edilmesi ve ilgili bilgilerin çıkarılması için kullanılır [48]. Bu tür analizler, avukatların ve hukuk uzmanlarının iş yükünü azaltarak, dava hazırlığı ve yasal araştırmalar gibi süreçleri hızlandırır. Ayrıca, hukuki metinlerdeki önemli bilgilerin tespit edilmesi, dava süreçlerinin daha etkin bir şekilde yönetilmesini sağlar [49].

Metin madenciliği, haber ve medya sektöründe de geniş bir uygulama alanına sahiptir. Haber metinlerinin analiz edilmesi ve özetlenmesi, okuyucuların hızlı ve etkili bir şekilde bilgiye erişmelerini sağlar. Ayrıca, sahte haberlerin tespit edilmesi ve doğruluk kontrolü gibi uygulamalarda da metin madenciliği teknikleri kullanılmaktadır. Bu tür analizler, medya kuruluşlarının daha doğru ve güvenilir haberler sunmalarına yardımcı olabilir [50].

E-ticaret sektöründe, metin madenciliği, ürün incelemelerinin ve müşteri geri bildirimlerinin analiz edilmesi için kritik bir rol oynar [51]. Bu analizler, işletmelerin ürünlerini geliştirmelerine ve müşteri deneyimlerini iyileştirmelerine yardımcı olur. Ayrıca, ürün öneri sistemlerinin geliştirilmesinde de metin madenciliği kullanılabilir, bu da müşterilere daha kişiselleştirilmiş ve ilgili ürün önerileri sunar [52].

Sonuç olarak, metin madenciliği, geniş bir uygulama yelpazesine sahip kritik bir teknolojidir. Sosyal medya, müşteri geri bildirimi, eğitim, sağlık, hukuk, haber ve medya, ve e-ticaret gibi birçok alanda metin madenciliği teknikleri kullanılmaktadır. Bu teknolojilerin gelişmesi, bilgisayarların metin verilerinden anlamlı ve faydalı bilgiler elde etmesini sağlar [53]. Metin madenciliği, bilgi çağında veri analizinin temel araçlarından biri olarak önemli bir rol oynamaya devam edecektir [54].

2.1.2. Duygu Analizi

Duygu analizi, metinlerdeki duygusal içerikleri belirlemeye yönelik bir alan olarak, özellikle son yıllarda büyük bir ilgi görmektedir. Bu alan, çeşitli disiplinlerde farklı uygulamalarla karşımıza çıkmaktadır. Bu yazıda, duygu analizi üzerine yapılan çalışmalardan yararlanılarak, farklı yaklaşımlar, karşılaşılan zorluklar ve uygulama alanları ele alınacaktır.

Duygu analizi, metinlerdeki duygusal içerikleri sınıflandırmayı amaçlayan bir DDİ tekniğidir [55]. Bu alanda kullanılan yöntemler genellikle kelime tabanlı yaklaşımlar, makine öğrenimi algoritmaları ve derin öğrenme modelleri olarak sınıflandırılabilir [56]. Kelime tabanlı yaklaşımlar, önceden tanımlanmış duygu sözlükleri kullanarak metinlerdeki duygu yüklü kelimeleri tespit eder [57]. Makine öğrenimi algoritmaları ise etiketli veri setleri üzerinde eğitilerek, metinlerin duygusal içeriğini belirlemeye çalışmaktadır [58]. Derin öğrenme modelleri, özellikle sinir ağları kullanarak metinlerdeki karmaşık duygu ilişkilerini öğrenebilir ve daha yüksek doğruluk oranlarına ulaşabilir [59].

Duygu analizinde karşılaşılan temel zorluklardan biri, ironi ve sarkazm gibi ifadelerin doğru bir şekilde analiz edilmesidir [60] . Ayrıca, çok anlamlı kelimelerin bağlamlarına göre doğru bir şekilde yorumlanması gerekmektedir [61]. Bu zorlukları aşmak için, gelişmiş DDİ teknikleri ve büyük veri setleri üzerinde eğitilmiş modeller kullanılarak daha doğru sonuçlar elde edilebilir [62].

Sosyal medya verilerinin analizi, duygu analizinin en yaygın uygulama alanlarından biridir. Sosyal medya gönderilerinin duygu analizi, kullanıcıların genel duyarlılıklarını ve belirli konulara karşı tepkilerini anlamak için kullanılabilir [63]. Bu analizler, markaların ve işletmelerin müşteri ilişkilerini yönetmelerine ve pazarlama stratejilerini optimize etmelerine yardımcı olur [64]. Ayrıca, kriz yönetimi sırasında, sosyal medya duygu analizleri, halkın genel tepkilerini ve endişelerini anlamak için kullanılabilir [65].

Haber ve medya sektöründe, duygu analizi, okuyucuların haberlere karşı genel tepkilerini anlamak için kullanılabilir [66]. Ayrıca, sahte haberlerin tespit edilmesi ve doğruluk kontrolü gibi uygulamalarda da duygu analizi teknikleri kullanılmaktadır [67]. Bu tür analizler, medya kuruluşlarının daha doğru ve güvenilir haberler sunmalarına yardımcı olabilir [68].

E-ticaret sektöründe, ürün incelemelerinin duygu analizi, müşteri memnuniyetini ve ürün kalitesini anlamak için kritik bir rol oynar [69]. Müşteri geri bildirimlerinin duygu analizi, işletmelerin ürünlerini geliştirmelerine ve müşteri deneyimlerini iyileştirmelerine yardımcı olur [68]. Ayrıca, ürün öneri sistemlerinin geliştirilmesinde de duygu analizi kullanılabilir, bu da müşterilere daha kişiselleştirilmiş ve ilgili ürün önerileri sunar [70].

Siyasi analizler, duygu analizinin bir diğer önemli uygulama alanıdır. Siyasi kampanyalar sırasında, adayların veya politikaların halk üzerindeki etkisini ölçmek için duygu analizi kullanılabilir [71].

Bu tür analizler, seçmen davranışlarını anlamak ve kampanya stratejilerini optimize etmek için kritik bilgiler sağlar [72]. Ayrıca, haber metinlerinin ve sosyal medya gönderilerinin duygu analizi, kamuoyunun genel duyarlılığını ve önemli olaylara karşı tepkisini anlamak için kullanılabilir [73].

Duygu analizinde kullanılan yöntemler arasında, Naive Bayes, Destek Vektör Makineleri (DVM) ve derin öğrenme modelleri öne çıkmaktadır. Naive Bayes, metin sınıflandırma problemlerinde yaygın olarak kullanılan bir olasılık temelli yaklaşımdır. DVM ise, veriyi farklı sınıflara ayırmak için hiper-düzlemler kullanan bir makine öğrenimi algoritmasıdır [74]. Derin öğrenme modelleri, özellikle sinir ağları kullanarak, metinlerdeki karmaşık duygu ilişkilerini öğrenebilir ve daha yüksek doğruluk oranlarına ulaşabilir [28].

Duygu analizinde derin öğrenme modellerinin kullanımı, özellikle son yıllarda büyük bir artış göstermiştir. Bu modeller, büyük veri setleri üzerinde eğitilerek, metinlerdeki karmaşık duygu örüntülerini ve bağlamsal bilgileri daha iyi yakalayabilmektedir [75]. Örneğin, uzun kısa süreli bellek (LSTM) ağları, metinlerin sıralı doğasını dikkate alarak, önceki kelimelerin ve cümlelerin duygusal etkilerini hesaba katabilir [76]. Bu sayede, metinlerdeki duygusal içeriği daha doğru bir şekilde analiz edebilir.

Duygu analizi, aynı zamanda sağlık sektöründe de önemli uygulamalara sahiptir. Hasta geri bildirimlerinin analizi, hastanelerin ve sağlık hizmeti sağlayıcılarının hizmet kalitesini değerlendirmelerine ve iyileştirmelerine yardımcı olabilir [77]. Ayrıca, psikolojik durum analizi ve ruh sağlığı izleme gibi alanlarda da duygu analizi teknikleri kullanılabilir [78].

Eğitim sektöründe, öğrenci geri bildirimlerinin duygu analizi, eğitim kalitesini artırmak ve öğrenci memnuniyetini sağlamak için kullanılabilir [79]. Öğrenci yorumlarının analizi, öğretim yöntemlerinin ve müfredatın değerlendirilmesine ve iyileştirilmesine yardımcı olabilir [80].

Sonuç olarak, duygu analizi, geniş bir uygulama yelpazesine sahip kritik bir teknolojidir. Pazarlama, müşteri ilişkileri, finans, eğitim, siyasi analizler, sosyal medya, haber ve medya, sağlık ve e-ticaret gibi birçok alanda duygu analizi teknikleri kullanılmaktadır. Bu teknolojilerin gelişmesi, bilgisayarların metinlerdeki duygusal içeriği daha iyi anlamasını, işlemesini ve sınıflandırmasını sağlamaktadır.

2.2. Finansal Çalışmalar

Finans alanında yapılan duygu sınıflandırması, yatırım kararları, risk yönetimi ve stratejik planlama gibi kritik alanlarda büyük öneme sahiptir. Bu çalışmalar, gelecekteki ekonomik koşulları ve piyasa hareketlerini öngörerek finansal performansı optimize etmeyi amaçlar. Genellikle zaman serisi analizleri, makine öğrenimi teknikleri ve ekonometrik modeller kullanılarak gerçekleştirilir [81].

Bu yazıda, finans alanındaki temel kavramlar, kullanılan teknikler ve çeşitli uygulama alanları ele alınacaktır.

Finansal çalışmalarda yaygın olarak kullanılan yöntemlerden biri, zaman serisi analizidir. Zaman serisi analizi, geçmiş verilerin incelenerek gelecekteki değerleri belirlemeyi hedefler [82]. Bu yöntem, özellikle finansal piyasaların tarihsel verilerine dayanarak gelecekteki fiyat hareketlerini anlamak için kullanılır [83]. Zaman serisi analizinde kullanılan en popüler modeller arasında ARIMA, GARCH ve Exponential Smoothing yer alır [84].

Makine öğrenimi, finansal çalışmalarda giderek daha fazla kullanılan bir diğer önemli tekniktir. Makine öğrenimi algoritmaları, büyük veri setleri üzerinde eğitilerek karmaşık ilişkileri ve örüntüleri öğrenebilir [85]. Bu sayede, finansal piyasaların gelecekteki hareketlerini daha doğru bir şekilde yorumlamak mümkündür [86]. DVM, karar ağaçları ve yapay sinir ağları, finans alanında yaygın olarak kullanılan makine öğrenimi algoritmalarındandır [87].

Derin öğrenme, özellikle son yıllarda duygu sınıflandırma ve finans çalışmalarında büyük ilgi görmüştür. Derin öğrenme modelleri, özellikle sinir ağları kullanarak çok katmanlı ve karmaşık verileri analiz edebilir [88]. Bu modeller, büyük veri setleri üzerinde eğitilerek finansal piyasaların dinamiklerini ve örüntülerini öğrenebilir [89]. LSTM (Long Short-Term Memory) ve CNN (Convolutional Neural Networks), finansal zaman serisi varsayımlarında yaygın olarak kullanılan derin öğrenme modellerindendir [90].

Finansal çalışmaları, çeşitli uygulama alanlarına sahiptir. Bu uygulamalardan biri, hisse senedi fiyat çıkarımıdır. Hisse senedi fiyat hesaplama, yatırımcıların doğru yatırım kararları almasına yardımcı olur [88]. Zaman serisi analizi ve makine öğrenimi teknikleri, hisse senedi fiyatlarını ileriye dönük olarak analiz etmek için yaygın olarak kullanılır [91]. Ayrıca, duyarlılık analizi ve sosyal medya verileri gibi alternatif veri kaynakları da hisse senedi fiyat değerlendirmelerinde kullanılmaktadır [92].

Bir diğer önemli uygulama alanı, döviz kuru öngörüsüdür. Döviz kuru öngörüsü, uluslararası ticaret ve yatırım kararlarında kritik bir rol oynar [93]. Ekonometrik modeller ve makine öğrenimi algoritmaları, döviz kuru hesaplamalarında yaygın olarak kullanılmaktadır [94]. Bu durum, merkez bankaları ve finansal kurumlar tarafından döviz rezerv yönetimi ve para politikaları oluşturmak için kullanılır.

Faiz oranı, finansal çalışmalarının bir diğer önemli alanıdır. Faiz oranları, borçlanma maliyetleri ve yatırım getirileri üzerinde doğrudan etkili olduğu için, doğruluğu büyük önem taşır [95]. Zaman serisi modelleri ve ekonometrik analizler, faiz oranı çıkarımlarında yaygın olarak kullanılır [96]. Ayrıca, makroekonomik göstergeler ve politik gelişmeler de faiz oranı dikkate alınır [97].

Finansal çalışmalar, aynı zamanda ekonomik büyüme ve enflasyon tespitinde kullanılır. Makroekonomik modeller ve zaman serisi analizleri, bu alanda yaygın olarak kullanılan yöntemlerdir [98]. Ayrıca, küresel ekonomik trendler ve ticaret verileri de bu sınıfta önemli rol oynar [99].

Risk yönetimi, finansal çalışmaların bir diğer önemli uygulama alanıdır. Finansal kurumlar, risklerini yönetmek ve minimize etmek için çeşitli modeller kullanmaktadırlar [100]. Zaman serisi analizi ve makine öğrenimi teknikleri, piyasa riski, kredi riski ve operasyonel risk gibi farklı risk türlerini analiz etmek için kullanılır [101]. Bu analizler, finansal kurumların sermaye yeterliliği ve likidite yönetimi stratejilerini belirlemelerine yardımcı olur.

Portföy yönetimi, finansal çalışmalarda bir diğer kritik uygulama alanıdır. Portföy yönetiminde, varlıkların getiri ve risklerinin belirlenmesi, optimal portföylerin oluşturulmasına yardımcı olur [102]. Makine öğrenimi ve zaman serisi analizi, portföy yönetiminde yaygın olarak kullanılan tekniklerdir [99]. Ayrıca, duyarlılık analizi ve alternatif veri kaynakları da portföy yönetiminde önemli rol oynar.

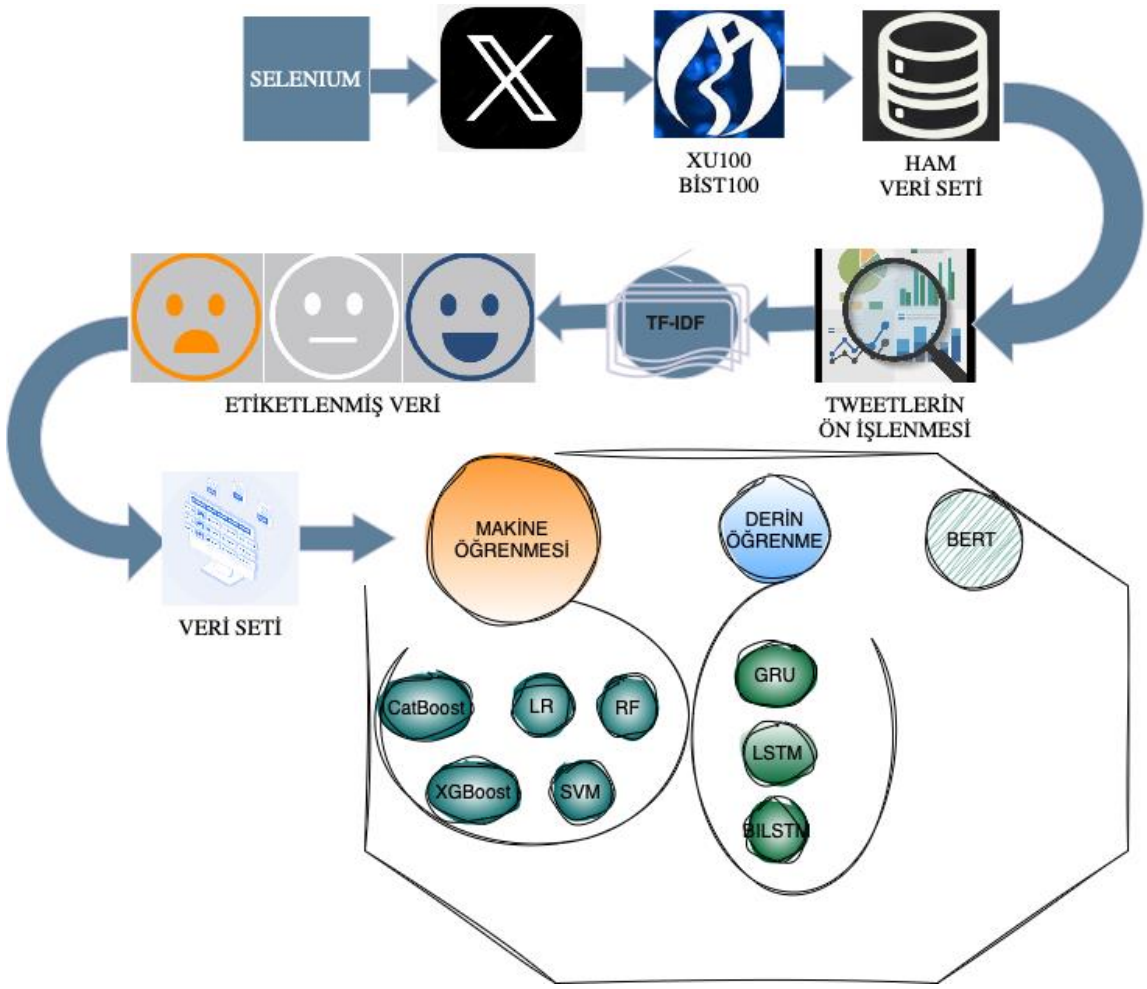
Finansal çalışmalar, aynı zamanda ticaret stratejilerinin geliştirilmesinde de kullanılır. Algoritmik ticaret ve yüksek frekanslı ticaret stratejileri, finans modelleri üzerine kuruludur [94]. Bu stratejiler, finansal piyasaların hızlı ve doğru bir şekilde analiz edilmesi ve alım-satım kararlarının otomatik olarak verilmesi için kullanılır [96].

3. METODOLOJİ

Bu bölümde, çalışmanın yürütülmesi sırasında izlenen yöntemler ve kullanılan araçlar detaylandırılmaktadır. Öncelikle veri toplama süreçleri, veri setlerinin özellikleri ve bu veri setlerinin araştırma için neden uygun olduğu açıklanacaktır. Daha sonra, verilerin analize uygun hale getirilmesi için uygulanan ön işleme adımları ve bu süreçlerin çalışmanın doğruluğu üzerindeki etkileri incelenecektir. Çalışmada kullanılan metin madenciliği ve doğal dil işleme (DDİ) tekniklerinin yanı sıra, makine öğrenimi ve derin öğrenme modellerinin uygulama detayları sunulacaktır. Ayrıca, model performanslarının değerlendirilmesi için kullanılan metrikler ve analiz süreçleri ayrıntılı bir şekilde ele alınacaktır.

Metodoloji bölümünde, yalnızca teknik detaylara değil, aynı zamanda bu yöntemlerin seçilmesinin gerekçelerine ve çalışmanın genel amacına olan katkılarına da yer verilmektedir. Bu yaklaşım, çalışmanın bilimsel temellerinin sağlamlığını ve elde edilen sonuçların geçerliliğini desteklemektedir.

Bu çalışmada, sosyal medyadaki finans içerikli gönderilerden duygu sınıflandırması yapılmıştır. X platformu üzerinden selenium kullanılarak çekilen finans metinlerinden ham veri seti oluşturulmuştur. Algoritmaların uygulanmasına kadar ki süreçte, durak kelimeler, noktalama işaretleri gibi metin ön işleme adımlarından geçmiştir. Sonrasında tf-idf yöntemi ile metinlerdeki kelime sıklıkları kontrol edilmiştir. Aynı zamanda pozitif ve negatif kelimeler içeren iki farklı veri setinden, metinlere puan tabanlı etiketleme işlemi uygulanmıştır. Bu işlem metin içindeki kaç kelimenin pozitif ya da negatif olması ile doğrudan orantılı olarak verilmiştir. Ön işleme adımlarından geçmiş olan veri setine makine öğrenmesi algoritmaları, derin öğrenme algoritmaları ve BERTurk algoritması uygulanmıştır. Tüm adımlar Şekil 3.1'de blok diyagramında gösterilmiştir.



Şekil 3.1. Duygu sınıflandırma süreci akış diyagramı

3.1. Materyal

Veri seti, 2023 yılı Aralık ayı sonundan 2024 yılı Ağustos ayı başına kadar, Twitter'da #xu100 ve #bist100 etiketleriyle toplanan 10.505 satırdan oluşan metin verilerini içermektedir. Bu veriler, işlenmiş ve temizlenmiş tweetler, skor ve duygu olarak üç sütun halinde organize edilmiştir.

processed_tweets_cleaned	score	sentiment
bist100 xu100 log yeşil kanal 132 iç yukarı trend devam et hafta ivme kazan kısa vade li hedef 14580 orta vade li hedef 22000 yatırım tavsiye değil elliott	3	Olumlu
bist100 son durum 11252 11423 alım program hedef i fiyat geri çek ol 10900 lerin fiyat tut bekle kritik bir direncegüncel kur 11590 doğru yaklaş fiyat ora tepki önem li ol hisse borsa ks bist30 xu100 piyas	0	Nötr
güna mutlu hafta son herkes ülke durum tam aşağı para bul kolay yol akıl et herkes bir başka sırt bin yük al kaç bist100 xu100	2	Olumlu
temmuz ayı kapsamlı çalış devam olarak analiz sevi çalış başla ay üzer koy iyi yap temel teknik analiz ol tamamen yeni yönte türkiye bir ilk ol yönetenimi devam et bist bist100 bist3 0	3	Olumlu
enkai pozitif görüntü devam et r kanal direnci 4650 takip et bist100	0	Nötr
12 tane yıldız pazar hisse var yükseliş trendin başla garan ci uzun vade yatırımcı arkadaş iste bir gör 1000 beğen rt ol liste ver rt et begen bist100 bist bist30 ks halkaarz btcu eth ethere doge ship	1	Olumlu
kchol kısa trend destek kır tekrar üst at 229 üzeri lazım üzer at alt destek 212 geri çek şaşır bist100	2	Olumlu
turo technical outlook is positive bist100	-1	Olumsuz
tam düz derken tünel ucu bommbokk bir yer çık aman dikkat et sevgili borsa yatırımcı kardeş cümleten iyi akşam r xu100 borsa	3	Olumlu
xu100 saatlik 11130 destek	1	Olumlu
eregl trendi pozitif 52tl üzer kal ivmen devam et düşün hedef 220us xu100 bist100	1	Olumlu
yazılım sorun fazla etkile şirket thy denizbank görün sorun global piyasa etkile büyüklük borsa pozisyon dikkat et kredili kaldıraçlı işlem gir xu100 bist100	-3	Olumsuz
200 kişi özel hisse konuş moral ver bil borsa öğret bir kere çanak katla tdgyo ikinci çanak katlat san hala borsa böyle bir yer işte ol yap bişe yaz xu100 bist100	0	Nötr
yol tarih zirve yol xu100 bist100	2	Olumlu
kısa vade 11950 seviye kadar yüksel öngör bist endeks xu100 hisse	2	Olumlu
cant 180 den destek bul 200 üzer at düş trendi kır yükseliş trendi başla paralel kanal 15 alt in düşüş derinleş r borsa bist100 xu100	-2	Olumsuz
boğa piya kötü haber sinek vızıltı ufak bir olumlu haber i zura etki yap ayı piyasa olumlu haber sinek vızıltı ufak bir olumsu haber i zura etki yap xu100	3	Olumlu
tukas düş trend kır ilk hedef 975 ana hedef 1386 yakın takip et paylaş hisse ol sadece bilgi amaçlı i borsa bist100	3	Olumlu

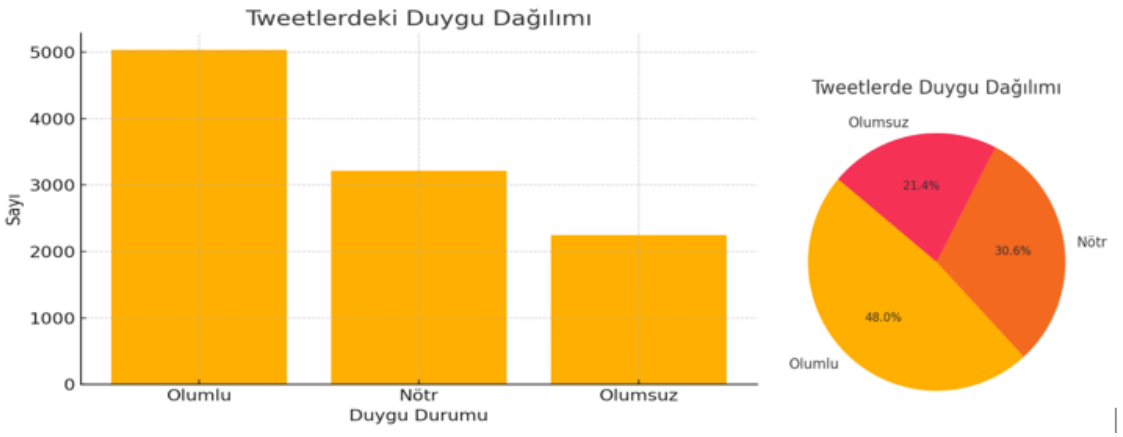
Şekil 3.2. Veri seti örneği

Şekil 3.2'deki veri seti, 2023 yılı aralık ayı sonundan 2024 yılı ağustos ayı başına kadar Twitter'dan toplanan 10.505 satırdan oluşmaktadır. Bu veriler, #xu100 ve #bist100 etiketli tweetlerden derlenmiştir. Veri setinde eksik ya da etiketsiz veri bulunmamaktadır ve üç ana sütun halinde organize edilmiştir: işlenmiş ve temizlenmiş tweetler, skorlar ve duygu sınıflandırmaları. Etiket dağılımı incelendiğinde, 3.217 adet nötr, 5.038 adet pozitif ve 2.250 adet negatif etiketli tweet olduğu görülmektedir. Bu dağılım, veri setinin çoğunlukla olumlu ve nötr tweetlerden oluştuğunu, olumsuz tweetlerin ise daha az sayıda olduğunu göstermektedir.

Veri seti, çeşitli ön işleme adımlarından geçirilmiştir. İlk olarak, gerekli kütüphaneler yüklenmiş ve NLTK kütüphanesi üzerinden Türkçe durak kelimeler ve kelime ayrıştırma (tokenization) işlemi için gerekli veriler indirilmiştir. Ardından, etiketleme yapılacak veri seti ile pozitif ve negatif kelimeleri içeren dosyalar yüklenmiştir. Türkçe durak kelimeler tanımlanarak, analiz sırasında gereksiz kelimeler ayıklanmıştır.

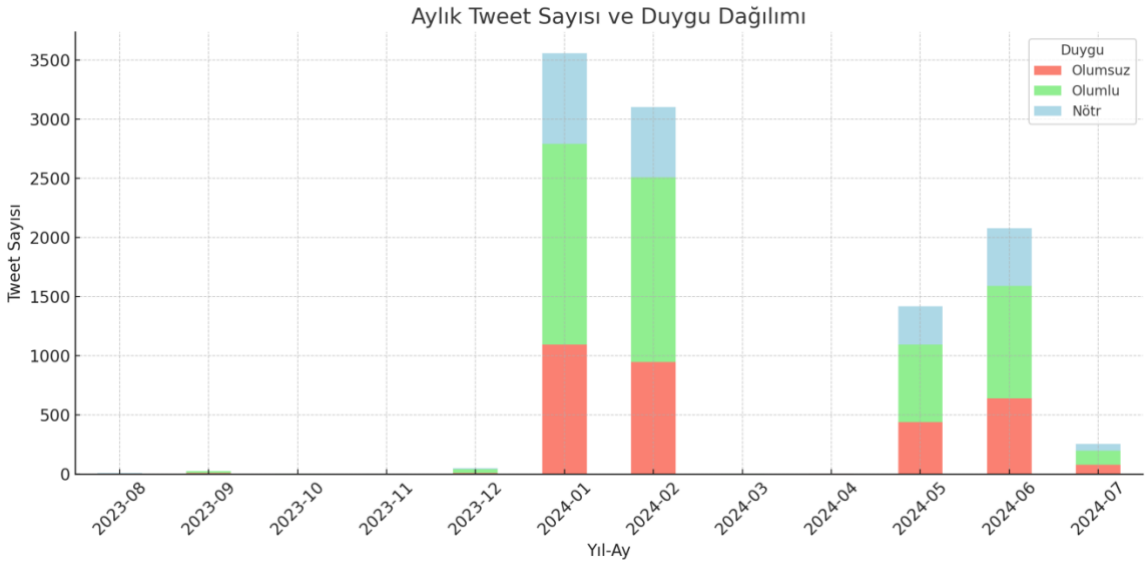
Tweet metinleri temizlenmiş ve bu aşamada tüm harfler küçültülmüş, bağlantılar ve noktalama işaretleri metinden kaldırılmış, kelimeler ayrıştırılmış ve durak kelimeler çıkarılmıştır. Bu işlemler sonucunda metinler, sınıflandırma için hazır hale getirilmiştir.

Duygu analizi, pozitif ve negatif kelimelerden oluşan listelere dayalı olarak gerçekleştirilmiştir. Tweetlerde pozitif bir kelime bulunduğunda skor artırılmış, negatif bir kelime bulunduğunda ise skor azaltılmıştır. Sonuç olarak, her tweetin duygu durumu pozitif, negatif veya nötr olarak sınıflandırılmıştır.



Şekil 3.3. Sentiment sütunu etiket dağılımı grafikleri

Şekil 3.3'te sunulan grafikler, sosyal medyada finans içerikli gönderilerdeki duygu sınıflandırmalarının dağılımını göstermektedir. Çubuk grafiğinde, her bir duygu etiketinin toplam sayısı belirtilmiştir. Olumlu etiket, 5.038 örnekle en yüksek frekansa sahip olup veride baskın bir konumdadır. Nötr etiket, 3.217 örnekle ikinci sırada yer alırken; olumsuz etiket 2.250 örnekle en düşük frekansa sahiptir. Daire grafiğinde ise bu etiketlerin yüzdesel dağılımı sunulmaktadır: olumlu gönderiler %48.0, nötr gönderiler %30.6 ve olumsuz gönderiler %21.4 oranında temsil edilmektedir. Bu dağılım, finans içerikli sosyal medya gönderilerinde olumlu ve nötr etiketlerin daha yaygın olduğunu, olumsuz etiketlerin ise daha düşük bir oranda bulunduğunu göstermektedir.



Şekil 3.4. Aylık tweet sayısı ve duygu dağılımı

Şekil 3.4'te, aylara göre atılan tweet sayıları ve bu tweetlerin duygu dağılımları gösterilmektedir. Her ay için, olumsuz, olumlu ve nötr tweetler sırasıyla kırmızı, yeşil ve mavi renklerle ifade edilmiştir.

Grafikten görüldüğü üzere, 2024 yılının Ocak ve Şubat aylarında tweet sayısı oldukça yüksektir. Bu dönemde özellikle olumlu tweetler öne çıkmakta, ancak nötr ve olumsuz tweetler de kayda değer bir düzeydedir. Benzer şekilde, 2024 yılının Haziran ayında da yoğun bir tweet trafiği gözlemlenmektedir; bu ayda nötr ve olumsuz tweetlerin oranı olumlu tweetlere kıyasla daha fazladır.

Diğer aylarda, örneğin 2023 yılının Eylül ve Aralık aylarında, tweet sayılarının oldukça düşük olduğu dikkat çekmektedir. Bu aylarda duygu dağılımı neredeyse yok denecek kadar azdır. Bu durum, belirli dönemlerde kullanıcıların daha fazla paylaşım yapma eğiliminde olduklarını göstermektedir. Ayrıca, aylara göre hangi dönemlerde daha fazla olumlu veya olumsuz tweet atıldığını analiz etme olanağı sunmaktadır. Özellikle yoğun tweet paylaşımlarının olduğu dönemlerde, insanların genel olarak nasıl bir duygusal tepki verdiğini anlamak açısından bu grafik önemli bir içgörü sağlamaktadır

3.2. Metin Ön İşleme

Metin ön işleme, metin madenciliği sürecinin ilk ve en önemli adımlarından biridir. Saf metin verileri, gürültü ve karmaşıklık içeren yapılandırılmamış ham verilerdir. Bu ham verileri belirli metin ön işleme adımlarından geçirilerek veri seti analizler için uygun hale getirilmiştir. Bu veriler üzerinde yapılan işlemler daha verimli ve analiz edilebilir veriler elde edilmesini sağlamaktadır.

3.2.1. Noktalama İşaretlerini, Sembolleri, Rakamları ve Beyaz Boşlukları Kaldırma

Ham veriler içerisinde yer alan semboller, tarih biçimleri, boşluklar, noktalama işaretleri gibi çeşitli unsurlar, yanlış analiz sonuçlarına yol açabilecek durumlar oluşturabilirler. Oluşan bu durumlar, metin verilerinde istenmeyen tekrarlara, gereksiz sıklık analizlerine ve yanıltıcı sonuçlara yol açabilir. Ayrıca bu tür unsurlar veri hacmini artırarak, gürültüyü çoğaltmakta ve model performansını olumsuz etkileyebilmektedir. Analizlerin daha verimli ve doğru sonuca ulaşabilmesi için bu ifadelerin kaldırılması ve verinin standart bir forma getirilmesi gerekmektedir.

3.2.2. Küçük Harf Çevirme

Analizde aynı olan kelimelerin farklı biçimlerde tekrarlanmasını önlemek ve tutarlı bir temsil sağlamak için ham veriler, tümü küçük harfe dönüştürülerek işlenir. Örneğin, "SAAT", "Saat" ve

“saat” kelimeleri bu işlem uygulanmadığı durumda farklı kelimeler gibi algılanabilmektedir. Yapılan bu işlem veri tutarlılığını sağlamak adına önemli bir rol oynar.

3.2.3. Durdurma Sözcüklerini ve Özel Sözcükleri Kaldırma

Belgede sıkça geçen ve ayırt ediciliği düşük olan “gibi”, “ama” vb. Türkçe edatlar ve bağlaçlar, durdurma sözcükleri olarak adlandırılmaktadır. Bu gibi kelimeler anlamsal olarak belirgin bir katkı sağlamadıkları için metin analizlerinde yanlış sapmalara neden olabilmektedir.

Ladani vd., metin belgelerinde durağan sözcüklerin yüksek sıklığının, metin belgelerinin içeriğini anlamada engel oluşturduğunun altını çizmiştir. Bu tür sözcüklerin metin analiz sürecinden çıkarılması için önceden tanımlanmış bir durdurma sözcükleri listesi kullanılabilir [103].

Ayrıca, bir derlemdeki diğer yaygın sözcükler veya analiz için herhangi bir katkısı olmayan özel sözcükler, verimliliği artırmak için derlemden çıkarılabilir. Bazı kelimeler açılımlarıyla birlikte aynı ifadeye sahip olması için yeniden tanımlanabilir. Örneğin, “ECB” ve “Avrupa Merkez Bankası” aynı anlama gelirken yazılı biçimleri farklıdır. Yazımları birbirine göre yapılandırılmamışsa, analizde bunları aynı olarak belirlemek mümkün değildir. Böyle durumlarda, bunlardan biri başka bir yazımla değiştirilebilir.

3.2.4. Sözlük Oluşturma

Herhangi bir yazılı kaynak (örneğin, kitaplar, makaleler, web siteleri, e-postalar, haberler gibi metin kaynakları) bilgi olarak kullanılabilir ve metin madenciliği amacıyla toplanabilir. Derlem, “belirli bir amaçla oluşturulmuş doğal dil (metin ve/veya konuşma veya işaretlerin transkripsiyonları) koleksiyonu” olarak tanımlanır [104]. Derlemler, dilin belirli yönlerini incelemek için veri madenciliği, makine öğrenimi gibi alanlarda kullanılmak üzere yapılandırılan önemli kaynaklardır.

Sözlük Oluşturma Süreci:

1. Veri Toplama: Metin verilerinin toplandığı kaynaklar belirlenir. Bu kaynaklar, çalışmanın amacına ve kapsamına göre seçilir. Örneğin, finansal duygu analizi çalışması için haber makaleleri, tweetler ve finansal raporlar kullanılabilir.

2. Ön İşleme: Toplanan metin verileri, ön işleme adımlarıyla temizlenir ve hazırlanır. Bu adımlar arasında noktalama işaretlerinin kaldırılması, küçük harfe dönüştürme, stop kelimelerin çıkarılması ve kök bulma (stemming) işlemleri yer alır.

3. Tokenlaştırma: Metin verileri, tokenlaştırma işlemi ile anlamlı parçalara ayrılır. Bu adımda, metinler kelimeler veya karakterler bazında tokenlara bölünür.

4. Kelime Frekanslarının Hesaplanması: Tokenlar arasında kelime frekansları hesaplanır. Bu adımda, her bir kelimenin metinlerdeki görünme sayısı belirlenir.

5. Sözlük Oluşturma: Tokenlardan elde edilen kelimeler ve bunların frekansları kullanılarak bir sözlük oluşturulur. Bu sözlük, her bir kelimenin metinlerdeki önemini ve sıklığını temsil eder.

Sözlük Oluşturmanın Yararları:

Anlamlı Veri Temsili: Metin verilerinin anlamlı bir şekilde temsil edilmesini sağlar. Kelimelerin frekansları ve önem dereceleri, metinlerin analiz edilmesi için kritik bilgiler sunar.

Özellik Çıkarımı: Makine öğrenimi modelleri için önemli özellikler çıkarılır. Sözlük, modelin öğrenmesi gereken anahtar kelimeleri ve terimleri içerir.

Boyut Azaltma: Metin verilerinin boyutunu azaltır ve daha yönetilebilir hale getirir. Sadece en önemli ve sık kullanılan kelimeler sözlükte yer alır.

Performans Artışı: Doğru ve etkili bir sözlük, metin madenciliği ve DDİ uygulamalarının performansını artırır. Daha az gürültü ve daha anlamlı veri temsili ile modellerin doğruluğu ve başarısı artar.

Sözlük Oluşturma Araçları ve Teknikleri:

TF-IDF (Term Frequency-Inverse Document Frequency): Kelime frekansını ve kelimenin tüm metinlerdeki yaygınlığını hesaplayarak önemli kelimeleri belirler.

Bag of Words (BoW): Metinleri kelime frekanslarına göre temsil eden basit ve etkili bir yöntemdir.

Word Embeddings: Kelimeleri vektörler olarak temsil eden daha gelişmiş bir yöntemdir. Word2Vec, GloVe ve FastText gibi teknikler kullanılarak kelimeler arasındaki anlamsal ilişkiler de dikkate alınır.

Sonuç olarak, sözlük oluşturma adımı, metin madenciliği ve DDİ süreçlerinde kritik bir rol oynar. Bu adım, metin verilerinin anlamlı ve etkili bir şekilde analiz edilmesini sağlar ve modellerin doğruluğunu artırır.

3.2.5. Tokenlaştırma

Tokenlaştırma, metin verilerinin daha küçük, anlamlı parçalara ayrılması sürecidir. Bu işlem, metin analizinin daha iyi ilerlemesini ve daha verimli sonuçlar elde edilmesini sağlar. Tokenlaştırma sonucunda elde edilen bu küçük birimlere “token” adı verilir. Tokenlar, kelime, kök ya da belirli dilbilgisi kurallarına göre bölünmüş parçalar olabilir ve metin madenciliği, DDİ gibi alanlarda temel birimler olarak kullanılır.

Tokenlaştırma Süreci:

1. **Metin Parçalama:**

   Metin, cümlelere ve kelimelere bölünür. Bu adımda noktalama işaretleri ve boşluklar dikkate alınarak metin parçalanır.

2. **Kelime Düzeyinde Tokenlaştırma:**

   Metin, kelime bazında tokenlara ayrılır. Örneğin, "Borsa yükseldi." cümlesi "Borsa", "yükseldi" kelimelerine ayrılır.

3. **Cümle Düzeyinde Tokenlaştırma:**

   Metin, cümle bazında tokenlara ayrılır. Örneğin, "Borsa yükseldi. Hisse senetleri değerlendi." metni iki ayrı cümle olarak tokenlara ayrılır.

4. **Özel Karakterlerin ve Noktalama İşaretlerinin Kaldırılması:**

   Tokenlaştırma işlemi sırasında noktalama işaretleri ve özel karakterler kaldırılarak sadece anlamlı kelimeler bırakılır.

5. **Küçük Harfe Çevirme:**

   Tokenlar, metnin daha tutarlı bir şekilde işlenmesi için genellikle küçük harfe çevrilir. Örneğin, "Borsa" tokenı "borsa" olarak değiştirilir.

Tokenlaştırma yöntemleri, metinlerin analizini daha verimli hale getirmek için çeşitli şekillerde uygulanabilir. Kelime tabanlı tokenlaştırmada, metin kelimelere bölünerek işlem yapılır ve bu yöntem, DDİ uygulamalarında yaygın olarak tercih edilir. Alt kelime tabanlı tokenlaştırma ise, metni alt kelime birimlerine bölerek özellikle morfolojik açıdan zengin dillerde daha etkili sonuçlar elde edilmesini sağlar. Karakter tabanlı tokenlaştırmada ise, özellikle küçük veri setleri ve düşük kaynaklı dillerde metni karakterlere ayırmak için kullanılır. Bu işlemler, metin verilerini anlamlı parçalara ayırarak analiz sürecini kolaylaştırır. Tokenlaştırma sırasında gereksiz karakterler ve işaretler temizlenerek veri setinin kalitesi artırılır.

Tokenlaştırma işlemi, DDİ süreçlerinde temel bir adımdır ve metin verilerinin daha verimli ve etkili bir şekilde analiz edilmesine olanak tanır. Tokenlar, metnin anlamını koruyarak daha küçük ve işlenebilir parçalara ayrılmasını sağlar. Sonuç olarak, doğru bir şekilde tokenlaştırılmış veriler, makine öğrenimi ve DDİ modellerinin performansını kayda değer şekilde iyileştirir.

3.3. Metnin Vektörleştirilmesi

Vektörleştirme işlemi, ham verileri gerçek sayıların vektörlerine dönüştürmektir [105]. Farklı veri türleri için geçerlidir ve çok eskiden beri kullanılmaktadır. Vektörleştirme sonucunda ham veriler vektör formatında sunulabilir.

Benzer şekilde, metin verileri, daha fazla analiz yapılabilmesi için sayısal değerlere dönüştürülür. Bengfort vd., makine öğrenimi uygulamalarında vektörleştirmenin önemini vurgulayarak, makine

öğrenimi algoritmalarının girdiyi iki boyutlu bir dizi şeklinde (örnekler satırlar, özellikler sütunlar olacak şekilde) beklediğini belirtmiştir. Bu nedenle, vektörleştirme veya özellik çıkarma adı verilen bu dönüştürme işlemi, metin verisinin analiz edilebilir bir forma getirilmesi açısından önemli bir aşamadır [106].

Metin verilerini vektör uzayında temsil etmek için kullanılan çeşitli vektörleştirme yöntemleri vardır. Bu yöntemlerden bazıları, terimlerin yalnızca sıklığını dikkate alarak metnin görünümünü analiz eder; Belge Terim Matrisi (DTM) gibi yöntemler buna örnek verilebilir. Diğer yandan, "Word2Vec" gibi daha gelişmiş yöntemler, terimler arasındaki semantik ilişkileri de dikkate alarak kelimelerin anlam bağlamında ayrıştırılmasını sağlar. Bu vektörleştirme işlemi, metin madenciliği sürecinde bir dönüşüm adımı olarak önemli bir rol oynar.

3.3.1. Kelime Çantası Modeli (BoW)

Kelime bulutları, metindeki yüksek sıklıkla geçen kelimeleri görselleştirmek için kullanılmaktadır. Bir metin özetleme yöntemidir ve farklı bağlamlarda öne çıkan anahtar kelimeleri ayırt etme imkanı sağlayarak kapsamlı bir metin analizi yapılmasına olanak tanır [107].

Kelime bulutlarında en sık kullanılan sözcükler daha büyük boyutta görünürken, daha düşük sıklıkta geçen sözcükler daha küçük boyutta gösterilir. Unigram, bigrams, trigrams kelimelerin kelime bulutlarında temsil edilebilmesi sağlanabilir. Kelime bulutları için "wordcloud" paketinin 2.6 versiyonu kullanılmaktadır [108].

3.3.2. Belge Terim Matrisi

Belge Terim Matrisi (BTM), satırların belgelere, sütunların terimlere ve hücrelerin terim sıklıklarına karşılık geldiği bir matristir. Welbers vd., BTM, kelime torbası yaklaşımıyla metin temsilinin en yaygın biçimlerinden biri olduğunu belirtmiştir. Ayrıca BTM kullanımının önemli bir avantajının, vektör ve matris cebirine dayalı analizlere olanak sağlaması olduğunu vurgulamıştır. BTM'nin bir diğer avantajı, özel matris formatlarının ve seyrek matrislerin bellek açısından verimli kullanılmasını sağlamasıdır [109].

3.3.3. Terim Frekansı – Ters Belge Frekansı

Bir metinde, önemi düşük olan yaygın kelimeler bulunabilirken, nadir kelimeler daha anlamlı ve önemli olabilir. Bu nedenle terim sıklığını dikkate almak yanıltıcı olabilir. Terim Frekansı – Ters Belge Matrisi (TF-TBF) bu ayrım için kullanışlıdır. TF-IDF, metin veri bilgilerinin alınması için istatistiksel bir yaklaşımdır ve bir derlemdeki terimlerin önemini belirlemek amacıyla kullanılır [110].

3.4. Veri Keşfi

Veri keşfi, kullanıcıların ilk kalıpları, özellikleri ve ilgi çekici detayları ortaya çıkarmak için büyük bir veri kümesini yapılandırılmamış bir şekilde keşfettiği veri analizinin ilk adımıdır. Bu süreç, bir veri kümesinin sahip olduğu her bilgi parçasını ortaya çıkarmak anlamına gelmez, bunun yerine önemli eğilimlerin ve daha ayrıntılı çalışılacak ana noktaların geniş bir resmini oluşturmaya yardımcı olur.

Veri keşfi sürecinde, bazı durumlarda manuel olarak veri incelenirken bazen de otomatik araçlarla grafik ve çizelgeler kullanılarak veri analizi gerçekleştirilir. Bu iki yaklaşımın birleşimi, veriyi daha kapsamlı bir şekilde anlamaya yardımcı olur.

3.4.1. Kelime İlişkilendirme

Başka bir yaygın yaklaşım, kelime çağrışımını bulmaktır. Bu amaçla "tm" paketinin 0.7-8 versiyonunda verilen bir terim için ilişkilendirme ve korelasyon limitini hesaplayan "findAssocs" adlı bir fonksiyon bulunmaktadır. Bu fonksiyon, DTM'deki tüm terimler arasındaki korelasyonları hesaplar ve korelasyon limitinden yüksek olanları filtreler. Belirli bir kelime için işlev, DTM'deki diğer her kelime ile ilişkisini hesaplar ve 0 ile 1 arasında bir puan verir. En yüksek puan, bu iki kelimenin belgelerde daha fazla görünmesi anlamına gelir. İşlev, belge düzeyine göre hesaplanır.

Verilen terim için, belgedeki diğer terimlerin ilişkilendirmesi, o terimi içeren tüm belgeler için incelenir. Verilen terimi içermeyen belgeler yok sayılır. Ayrıca, kelimeler arasındaki ilişki, ilişkilendirme grafikleri ve korelasyon haritalarıyla görselleştirilebilir.

3.4.2. Okunabilirlik Puanı

Otomatik Okunabilirlik İndeksi (ARI), bir metnin okunabilirlik düzeylerini ölçmek için tasarlanmış bir ölçümdür. ARI, Senter ve Smith tarafından 1967'de İngilizce metinler için tanıtıldı. ARI formülü Denklem 3.1'de gösterildiği gibidir:

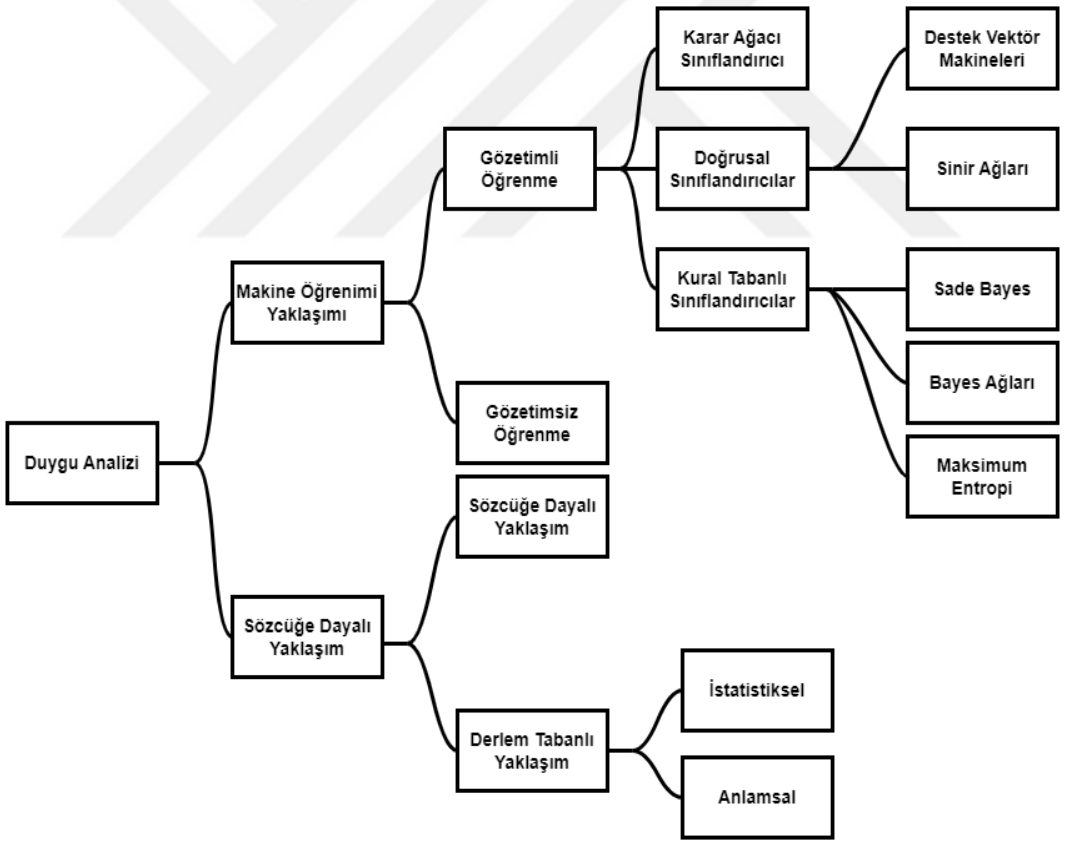
$$4.71 \times \left(\frac{\text{kelime sayısı}}{\text{karakter sayısı}}\right) + 0.5 \times \left(\frac{\text{cümle sayısı}}{\text{kelime sayısı}}\right) - 21.43 \quad (3.1)$$

Burada nchars karakter sayısını, nwords kelime sayısını, nsentences ise cümle sayısını göstermektedir. Formülden, metnin karmaşıklığının ortalama kelime ve cümle uzunluğuna bağlı olduğu anlaşılmaktadır.

3.5. Duygu Analizi

Duygu analizi, Medhat vd., tarafından "görüşlerin, duyguların ve metnin öznelliğinin sayısal olarak ele alınması" olarak tanımlanır [55]. Duygu analizinin devam eden bir metin madenciliği araştırma alanı olduğu vurgulanmaktadır. Liu'ya göre duygu analizi ve fikir madenciliği, insanların duygularını, düşüncelerini, tutumlarını ve yazılı dildeki değerlendirmelerini analiz etme çalışmasıdır ve DDİde en aktif araştırma alanlarından biridir. Veri madenciliği, web madenciliği ve metin madenciliğinde geniş çapta çalışılmaktadır [68].

Duygu analizi uygulamaları pazarlama, sosyal medya analizi, siyaset vb. birçok farklı alanda kullanılmaktadır. Örnek vermek gerekirse, tüketici incelemeleri, Twitter, haberler, politikacıların konuşmaları, duygu analizi için metin verisi örnekleridir. Duygu analizi yöntemlerine gelince, farklı yöntemlerin ve farklı analiz birimlerinin olduğu belirtilebilir [68]. Medhat vd., duygu analizi yöntemlerinin sınıflandırmasını aşağıdaki şekilde özetlemiştir [55]. Duygu analizi teknikleri Şekil 3.5'te ele alınmıştır.



Şekil 3.5. Duygu analizi teknikleri

3.6. Makine Öğrenmesi

Makine öğrenmesi, bilgisayarların belirli görevleri yerine getirmek için verilerden öğrenmesini ve kendini geliştirmesini sağlayan bir yapay zeka dalıdır. Bu öğrenme süreci, insanların müdahalesi olmadan modelin analiz edilebilmesi için örnek veri setlerinden kalıplar ve ilişkiler bulmayı içerir. Makine öğrenmesi denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme olmak üzere genellikle üç ana kategoriye ayrılır:

Denetimli Öğrenme (Supervised Learning):
Bu yöntemde model, etiketli bir veri seti kullanılarak eğitilir. Her bir girdiye karşılık gelen bir çıktı (etiket) bulunduğundan, model girdi ve çıktı arasındaki ilişkiyi öğrenir. Öğrenme süreci tamamlandığında, model yeni veriler üzerinde sınıflandırma yapabilir. Denetimli öğrenmede, performans değerlendirmesi için doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metrikler kullanılır. Bu alt başlık altında en yaygın kullanılan denetimli öğrenme algoritmaları arasında lojistik regresyon, DVM, rastgele orman ve gradyan artırma algoritmaları (XGBoost, CatBoost) yer alır.

3.6.1. Lojistik Regresyon

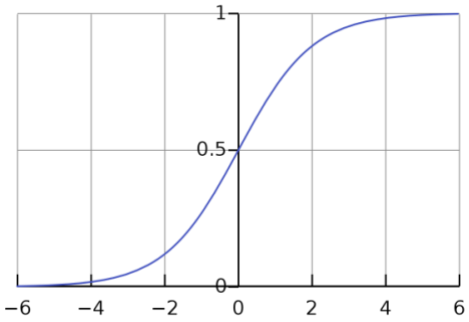
Lojistik regresyon, ikili veya çok sınıflı sınıflandırma problemlerinde yaygın olarak kullanılan bir denetimli öğrenme algoritmasıdır. Bu algoritma, girdi özellikleri ile hedef değişken arasında doğrusal bir ilişki kurarak sınıflandırma yapmayı amaçlar. Ancak, lojistik regresyonun çıktısı sürekli bir değer değil, olasılık tabanlıdır; yani, model, bir gözlemin belirli bir sınıfa ait olma olasılığını öngörmektedir. Lojistik regresyonun temel formülü sigmoid fonksiyonuna dayanır:

$$y' = \frac{1}{1-e^{-z}} \tag{3.2}$$

Burada:

- y', bir gözlemin belirli bir sınıfa ait olma olasılığını ifade eder.
- $z = w \cdot x + b$ modelin doğrusal birleşim fonksiyonudur; burada w ağırlık vektörü, x girdi özellikleri ve b ise bias terimidir.

Bu denklem çizildiğinde ortaya çıkan S eğrisi Şekil 3.6'da gösterilmiştir [111].



Şekil 3.6. Lojistik regresyon eğrisi

Bu formül sayesinde lojistik regresyon, her bir gözlem için bir olasılık üretir. Üretilen değer olasılığın belirli bir eşik değerinin üzerinde olup olmamasına bağlı olarak gözlemin sınıfı belirlenir. Örneğin, $y' \geq 0.5$ olduğunda gözlem, pozitif sınıfa atanır.

Lojistik regresyonda modelin başarısını ölçmek için logaritmik kayıp fonksiyonu (log-loss) kullanılır. Logaritmik kayıp, modelin olasılık öngörüleri ile gerçek etiketler arasındaki farkı ölçer. Kayıp fonsiyonu Denklem 3.3'te verilmiştir.

$$L(y, y') = -\frac{1}{N} \sum_{i=1}^{N} (y_i \log(y_i') + (1 - y_i) \log(1 - y_i') \quad (3.3)$$

Burada:

- y_i, gerçek etiket
- y_i', modelin belirlediği olasılık değeri,
- N, toplam gözlem sayısıdır.

Logaritmik kayıp fonksiyonu, modelin olasılık ihtimalinin doğruluğunu değerlendirerek en uygun parametrelerin bulunmasını sağlar. Model, bu kaybı minimize edecek şekilde optimize edilir.

Lojistik Regresyonda Aşırı Uyum ve Düzenlileştirme:

Lojistik regresyon modellerinde, özellikle çok sayıda özelliğin bulunduğu durumlarda aşırı uyum riski ortaya çıkabilir. Aşırı uyum, modelin eğitim verisine çok iyi uyum sağladığı ancak yeni veriler üzerinde düşük performans gösterdiği bir durumdur. Bu sorunu azaltmak için düzenlileştirme yöntemleri kullanılır. Düzenlileştirme, modelin parametrelerini sınırlayarak daha genel geçer bir model elde etmeyi amaçlar.

Lojistik regresyonda yaygın olarak kullanılan düzenlileştirme yöntemleri L1 (lasso) ve L2 (ridge) düzenlileştirme olarak bilinir:

- **L1 Düzenlileştirme**: Lasso düzenlileştirme olarak bilinen L1 düzenlileştirme, modelin ağırlıklarının mutlak değerleri toplamına dayalı bir ceza terimi ekler. Bu düzenlileştirme, bazı ağırlıkları sıfıra indirerek özellik seçimini otomatik olarak gerçekleştirir ve daha sade bir model ortaya çıkarır. L1 kayıp fonksiyonu Denklem 3.4'te verilmiştir.

$$L(y, y') = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(y_i') + (1 - y_i)\log(1 - y_i')) + \lambda \sum_{j=1}^{k} |w_j| \quad (3.4)$$

- **L2 Düzenlileştirme**: Ridge düzenlileştirme olarak da bilinen L2 düzenlileştirme, modelin ağırlıklarına bir ceza terimi ekler ve bu terim, ağırlıkların kareleri toplamına bağlıdır. L2 kayıp fonksiyonu denkklem 3.5'de verilmiştir.

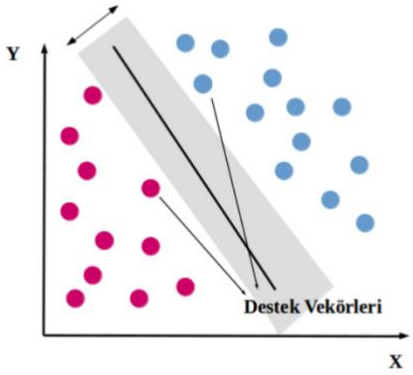
$$L(y, y') = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(y_i') + (1 - y_i)\log(1 - y_i')) + \lambda \sum_{j=1}^{k} {w_j}^2 \quad (3.5)$$

Burada λ lambda, düzenlileştirme katsayısıdır ve ağırlıkların büyüklüğüne bağlı olarak cezayı belirler.

Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. Düzenleme terimi ve hiperparametrelerin uygun şekilde ayarlanması, lojistik regresyonun doğruluğunu artırarak daha iyi genellenebilir sonuçlar elde edilmesini sağlar. Düzenlileştirme yöntemleri, modelin karmaşıklığını kontrol altında tutarak lojistik regresyonun daha iyi genellenmesini sağlar ve böylece eğitim verisinde elde edilen performansın yeni veriler üzerinde de sürdürülebilir olmasına yardımcı olur.

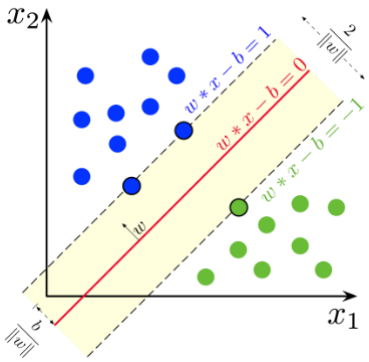
3.6.2. Destek Vektör Makineleri

Destek vektör makineleri, hem sınıflandırma hem de regresyon problemlerinde yaygın olarak kullanılan güçlü bir makine öğrenmesi algoritmasıdır. DVM, Şekil 3.7'de gösterildiği gibi veriyi en iyi şekilde ayıran bir hiperdüzlem bulmayı amaçlar [112].



Şekil 3.7. Hiperdüzlem gösterimi

İki sınıf arasındaki ayrımı en iyi yapacak hiperdüzlem, sınıflar arasındaki marjini maksimize eden düzlem olarak tanımlanır. Marjin, hiperdüzlem ile en yakın veri noktaları arasındaki mesafeyi ifade eder. Sınıflandırma problemlerinde marjinin maksimum yapılması, modelin genel performansını artırmaya yönelik bir strateji sağlar. DVM modeli Şekil 3.8'de gösterildiği gibi hiperdüzleme en yakın olan ve "destek vektörleri" adı verilen veri noktalarına dayanarak marjini oluşturur.



Şekil 3.8. Marjin gösterimi

Bu destek vektörleri, sınıflar arasındaki sınırın belirlenmesinde doğrudan rol oynar ve modelin karmaşıklığını kontrol altında tutar. DVM algoritmasının temel amacı, marjini maksimize ederek sınıfları en iyi şekilde ayıracak bir hiperdüzlem bulmaktır.

DVM'in hiperdüzlemi bulma işlemi matematiksel olarak Denklem 3.6 ve 3.7'de gösterildiği gibidir.

$$\underset{w,b}{\text{Min}} \frac{1}{2} ||w||^2 \tag{3.6}$$

$$Koşul: y_i(w.x_i + b) \geq 1 \ \forall i \tag{3.7}$$

Burada:

- w, hiperdüzlemin norm vektörüdür,
- b, hiperdüzlem üzerindeki bias terimidir,
- y_i, her bir veri noktasının gerçek sınıf etiketidir (örneğin, +1+1+1 veya −1-1−1),
- x_i, her bir veri noktasının öz nitelikleridir.

Bu optimizasyon, sınıflandırma doğruluğunu artırmak için marjini en geniş yapacak şekilde w ve b parametrelerini öğrenmeyi amaçlar.

DVM, doğrusal olarak ayrılabilen verilerde başarılı sonuçlar verirken, doğrusal olmayan veri kümelerinde de çekirdek fonksiyonları kullanılarak uygulanabilir. Çekirdek fonksiyonları, orijinal veriyi daha yüksek boyutlu bir uzaya dönüştürerek doğrusal olmayan ayrımların doğrusal olarak ayrılabilir hale gelmesini sağlar. Yaygın olarak kullanılan çekirdek fonksiyonları şunlardır:

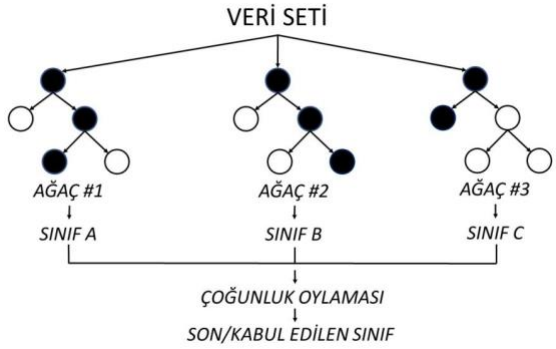
- **Doğrusal Çekirdek:** Özellikle verinin doğrusal olarak ayrılabildiği durumlarda kullanılır. Doğrudan özellikler arasındaki ilişkiyi modellemek için uygun ve hesaplama açısından en verimli çekirdek türüdür.
- **Polinomik Çekirdek**: Veriyi daha yüksek dereceli bir uzaya dönüştürerek doğrusal olmayan ayırımları yapılabilir hale getirir. Bu çekirdek türü, özellikle daha karmaşık ayırımların gerekli olduğu durumlarda etkilidir. Polinom derecesi hiperparametre olarak ayarlanabilir.
- **Radyal Tabanlı Fonksiyon (RBF) Çekirdeği**: DVM'de en yaygın kullanılan çekirdeklerden biridir. Özellikle doğrusal olmayan veri setleri için uygundur. Veriyi sınıflandırma sınırına göre merkeze yakın veya uzak konumlandırır, böylece karmaşık ayırımları mümkün kılar. Gamma parametresiyle veri noktalarının etkisi ayarlanabilir.
- **Sigmoid Çekirdek**: Sinir ağlarındaki aktivasyon fonksiyonuna benzer bir işlev görür. Özellikle sinir ağı yapılarının simülasyonunu gerçekleştirmek için kullanılır, ancak çok nadiren tercih edilir. Genellikle doğrusal olmayan sınıflandırma görevlerinde denenir, ancak polinom ve RBF çekirdeklerine göre daha sınırlı bir performans gösterir.

Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. Çekirdek fonksiyonlarının uygun şekilde seçilmesi, DVM'in doğruluğunu artırarak daha iyi genellenebilir sonuçlar elde edilmesini sağlar.

3.6.3. Rastgele Orman

Rastgele Orman (Random Forest), karar ağaçları topluluğu kullanarak model doğruluğunu artıran, hem sınıflandırma hem de regresyon problemlerinde uygulanan güçlü bir makine öğrenimi algoritmasıdır. Bu model, bagging (bootstrap aggregating) adı verilen topluluk yöntemini kullanarak her bir ağacı rastgele bir alt veri kümesi üzerinde eğitir. Bu yöntem, modelin genellenebilirliğini artırır ve tek bir karar ağacına kıyasla aşırı uyumu azaltır.

Rastgele Orman modelinin temel prensibi, birden fazla karar ağacını bağımsız olarak eğitip sonuçlarını birleştirerek daha güvenilir sonuçlar elde etmektir. Her bir ağaç, veri kümesinin farklı bir alt örneği üzerinde çalıştığı için farklı bir öngörü sağlar. Bu çeşitlilik, hataları dengeler ve tek bir karar ağacının eğilimlerine kıyasla daha doğru sonuçlar elde edilmesini sağlar. Modelin genel yapısı Şekil 3.9'da gösterilmektedir [113].

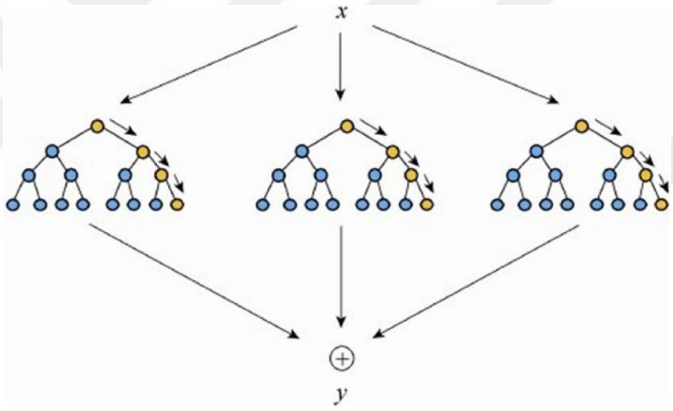


Şekil 3.9. Rastgele Orman algoritmasının genel yapısı

Rastgele Orman'ın temel amacı, çeşitli karar ağaçlarının çoğunluk veya ortalama varsayımlara dayanarak nihai doğruluğu artırmakta ve aşırı uyum riskini azaltmaktır. Bu yöntemde, her bir karar ağacı, orijinal veri kümesinin rastgele alınan bir örnekleme üzerinde ve rastgele seçilen özellikler kullanılarak eğitilir. Bu ağaçlar bağımsız olarak eğitildikleri için her biri farklı bir çıkarım yapabilir ve topluluk olarak verilen sonuç daha güvenilir hale gelir.

Rastgele Orman modeli, aşağıdaki adımları içeren bir eğitim süreciyle çalışır:

- **Torbalama Örnekleme**: Orijinal veri kümesinden rastgele seçilen örneklemelerle *N* adet alt veri kümesi oluşturulur. Her bir alt veri kümesi, orijinal veri kümesinin yaklaşık 2/3 kadarını içerir ve her bir karar ağacı bu örnekler üzerinde eğitilir.
- **Rastgele Özellik Seçimi**: Her bir karar ağacında, her düğümdeki bölünme işlemi için tüm özellikler yerine rastgele bir alt küme kullanılır. Bu özellik sayısı genellikle sınıflandırma problemlerinde özelliklerin karekökü ($m = \sqrt{M}$), regresyon problemlerinde ise m=M/3 olarak belirlenir. Bu sayede her ağaç, kendine özgü bir model öğrenir.
- **Karar Ağaçlarının Eğitimi**: Her bir karar ağacı, seçilen özellikler ve örneklemeler üzerinden eğitilir. Ağacın derinliği ve yapısı üzerinde sınırlamalar uygulanarak aşırı uyumun önüne geçilir.
- **Tahminlerin Birleştirilmesi**: Sınıflandırma problemlerinde, tüm ağaçların oylaması alınarak çoğunluk oyu ile nihai sınıflandırma yapılır. Regresyon problemlerinde ise her ağacın ihtimallerinin ortalaması alınarak nihai sonuç elde edilir. İhtimallerin birleştirilmesi örnek yapısı Şekil 3.10'da gösterilmektedir.



Şekil 3.10. Varsayımların birleştirilmesi

Bir Rastgele Orman modelinde, T adet bağımsız karar ağacı bulunur ve modelin nihai öngörüsü, bu ağaçların çoğunluk oyu veya ortalaması alınarak hesaplanır. Sınıflandırma problemlerinde Rastgele Orman'ın kestirimi, T adet karar ağacının çoğunluk oyu ile belirlenir. Her bir ağacın değeri $h_i(x)$ olmak üzere nihai sonuç Denklem 3.8'de verildiği gibi gösterilmektedir.

$$y' = mode(h_1(x), h_2(x), \ldots\ldots, h_T(x)) \tag{3.8}$$

Burada, model fonksiyonu en sık görülen sınıf etiketini döndürür.
Regresyon problemlerinde Rastgele Orman algoritması, Denklem 3.9'da gösterildiği gibi her bir ağacın ortalaması alınarak bulunur:

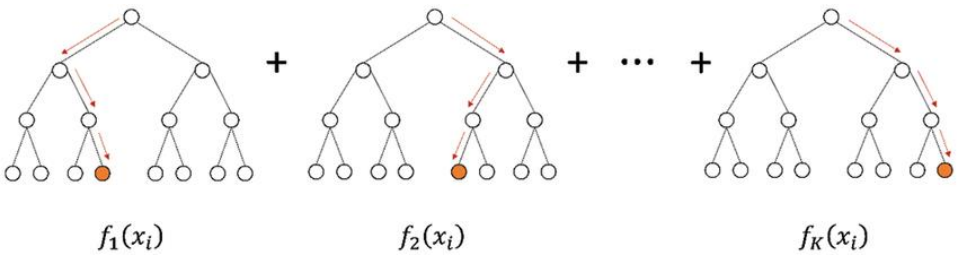
$$y' = \frac{1}{T\sum_{i=1}^{T} h_i(x)} \tag{3.9}$$

Burada $h_i(x)$ *i*-nci karar ağaçtan elde edilen sonucu gösterir.
Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. Rastgele Orman modeli, aşağıdaki hiperparametrelerin doğru ayarlanmasıyla performans açısından optimize edilebilir:

- **Ağaç Sayısı (*T*)**: Genellikle ağaç sayısı arttıkça modelin performansı da artar; ancak belirli bir noktadan sonra ağaç sayısındaki artış hesaplama süresini uzatır ve performansa etkisi azalmaya başlar. Dolayısıyla optimal ağaç sayısını bulmak önemlidir.
- **Maksimum Derinlik**: Her bir karar ağacının maksimum derinliği, aşırı uyumu engellemek için sınırlandırılabilir. Bu hiperparametre, modelin karmaşıklığını kontrol etmek için kullanılır ve doğru bir derinlik seviyesi belirlemek genelleme yeteneğini artırır.
- **Rastgele Özellik Sayısı (*m*)**: Her bir düğüm için rastgele seçilen özellik sayısı, modelin çeşitliliğini artırır ve daha genel sonuçlar sağlar. Bu sayı, veri setinin özelliğine göre optimize edilmelidir; sınıflandırma için genellikle $m = \sqrt{M}$, regresyon için isem=M/3 şeklinde seçilebilir. Rastgele Orman modeli, sağladığı yüksek doğruluk ve esneklik ile birçok veri türü üzerinde güvenilir sonuçlar vermekte ve günümüz makine öğrenmesi uygulamalarında sıklıkla kullanılmaktadır.

3.6.4. XGBOOST

XGBoost, yüksek doğruluk sağlayan ve hızlı çalışan, gradyan artırma (gradient boosting) algoritmasını temel alan bir makine öğrenmesi modelidir. Özellikle sınıflandırma ve regresyon problemlerinde güçlü bir performans sunar ve yapılandırılmış veri setlerinde en sık tercih edilen yöntemlerden biridir. XGBoost, çok sayıda karar ağacını ardışık olarak oluşturup bunları birleştirerek, hataları minimize etmeye ve nihai performansını iyileştirmeye çalışır. XGBoost yapısı Şekil 3.11'de gösterilmektedir [114].



Şekil 3.11. XGBoost Yapısı

Bu algoritma, önceki ağaçların hatalarını düzelterek her yeni ağacın oluşturulmasını sağlar. Her yeni ağaç, önceki ağaçlardan elde edilen hataları azaltmak üzere eğitilir. Bu ardışık iyileştirme süreciyle model, yüksek performans elde eder ve hataya en duyarlı noktaları öğrenir. XGBoost'un temel özellikleri ve işleyişine dair detaylar şöyledir:

Başlangıç Modeli ve Hata Hesaplama: XGBoost algoritması, önce tüm gözlemler için bir başlangıç değeri ile işe başlar. Bu başlangıç değeri genellikle hedef değişkenin ortalamasıdır. Örneğin, bir regresyon problemi için başlangıç değeri tüm veri setinin ortalama değeri olabilir. Bu, modelin henüz hiçbir özellik veya ağaca dayalı olmadığı bir başlangıç noktasıdır.

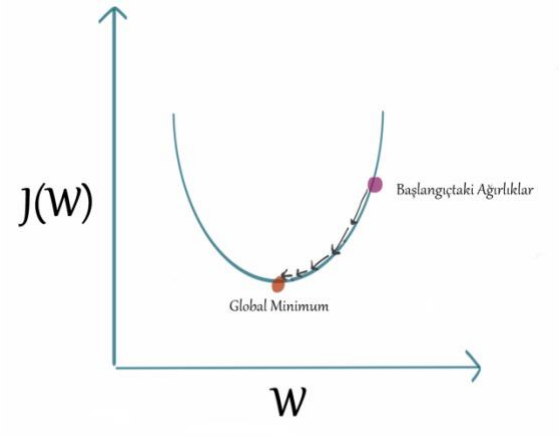
Bu başlangıç değerlerine dayanarak, her gözlem için modelin hata miktarı (residuals veya artık değerler) hesaplanır. Bu hata, her bir gözlemdeki gerçek değer ile modelin ilk belirlediği değer arasındaki farktır. XGBoost, her bir yeni ağacı eklerken bu hataları minimize etmeyi amaçlar. Hataları azaltmak, modelin varsayım doğruluğunu artırmak için kritik bir adımdır. Denklem 3.10'da gösterilmektedir.

$$Hata = y_i - y' \tag{3.10}$$

burada y_i, gerçek değeri, y'ise modelin sonuç değerini temsil eder.

Yeni Ağaçların Eklenmesi: XGBoost, gradyan artırma yöntemini kullanarak her yeni ağacı ekler. Gradyan artırma, modelin kalan hatalarını daha iyi anlamak için her adımda yeni bir karar ağacı ekleme sürecidir. Yeni eklenen her bir ağaç, bir önceki modelin bıraktığı hatayı düzeltmek için eğitilir.

Bu süreçte, modelin kayıp fonksiyonunu minimize etmek amacıyla **gradyan düşüşü** kullanılır. Gradyan düşüşü grafiksel olarak Şekil 3.12'de gösterilmektedir.



Şekil 3.12. Gradyan düşüşü

Bu modelde her bir ağaç, önceki ağaçların hatalarını düzeltmeye çalışır. Topluluk modelinin genel formülü Denklem 3.11'de gösterildiği gibidir.

$$y' = \sum_{t=1}^{T} \alpha_t f_t(x) \tag{3.11}$$

Burada:

- T: Toplam ağaç sayısını,
- α_t: Her ağacın katkı oranını belirleyen bir katsayıyı,
- $f_t(x)$: t-inci ağacın değerlendirmesini temsil eder.

Bu topluluk modeli sayesinde XGBoost, basit karar ağaçlarının birleşimi ile daha güçlü ve karmaşık bir model elde eder. Bu yapı, yüksek doğruluklu ve genelleme yeteneği yüksek bir model oluşturur. Modelin doğruluğu, ağacın karmaşıklığı ve öğrenme oranı gibi parametrelerle dengelenir.

Öğrenme Hızı ve Düzenlileştirme: Modelin aşırı öğrenmeye karşı dayanıklı olmasını sağlamak için öğrenme hızı (learning rate) parametresi kullanılır. Ek olarak, XGBoost, L1 (lasso) ve L2 (ridge) düzenlileştirme tekniklerini içerir, bu da modelin karmaşıklığını kontrol ederek aşırı uyumun önüne geçer.

XGBoost, kayıp fonksiyonunu minimize etmeye çalışırken aynı zamanda modelin genelleme yeteneğini artırmayı hedefler. Kayıp fonksiyonu, nihai değerler (y') ile gerçek değerler (y) arasındaki farkı ifade eder. Fonksiyon tanımı Denklem 3.12'de gösterilmektedir.

$$L(\theta) = \sum_{i=1}^{n} l(y_i, y') + \sum_{k=1}^{K} \Omega(f_k) \tag{3.12}$$

Burada:

- $l(y_i, y')$: Hesaplama hatası (örneğin, kare hata fonksiyonu).
- $\Omega(f_k)$: Model karmaşıklığını cezalandıran düzenlileştirme terimi. Bu, her bir ağacın karmaşıklığını kontrol etmek için kullanılır.
- *K*: Ağaç sayısını temsil eder.

Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. XGBoost modelinin başarısı, doğru hiperparametrelerin seçilmesine bağlıdır. Başlıca hiperparametreler şunlardır:

- **n_estimators**: Eklenen ağaç sayısını belirler. Çok fazla ağaç eklemek, modelin aşırı uyum yapmasına neden olabilir.
- **learning_rate**: Öğrenme hızı, modelin her adımda ne kadar değişiklik yapacağını belirler.
- **max_depth**: Her ağacın maksimum derinliğini belirler, bu da modelin karmaşıklığını kontrol eder.
- **subsample**: Rastgele örnekleme oranını belirler ve aşırı öğrenmeyi azaltmaya yardımcı olur.
- **colsample_bytree**: Her bir ağacın eğitiminde kullanılacak özellik oranını belirler.

3.6.5. CATBOOST

CatBoost, özellikle kategorik verilerin işlenmesinde üstün performans gösteren, gradyan artırma tabanlı gelişmiş bir makine öğrenmesi algoritmasıdır. İsmini "Categorical Boosting" ifadesinin kısaltmasından alan bu yöntem, özellikle karmaşık veri yapılarıyla çalışırken sağladığı verimlilik ve doğruluk avantajlarıyla öne çıkar. CatBoost, kategorik değişkenleri işlemek için yenilikçi yaklaşımlar kullanarak, yüksek doğruluğa sahip modelleri oluşturur ve diğer gradyan artırma algoritmalarına göre veri yapısının karmaşıklığını daha etkin bir şekilde yönetir. Bu yönleriyle, büyük ve çeşitli veri setlerinde güçlü bir modelleme çözümü sunar. CatBoost'un temel özellikleri ve işleyişine dair detaylar şöyledir:

Başlangıç Modeli ve Hata Hesaplama: CatBoost, diğer gradyan artırma algoritmalarına benzer şekilde, başlangıçta tüm gözlemler için basit bir başlangıç değeri belirleyerek işe başlar. Bu başlangıç değeri genellikle, hedef değişkenin ortalama değeridir. Bu temel değer, her gözlem için modelin yapacağı ilk kestirim olarak kullanılır. Başlangıç değeri Denklem 3.13'te gösterildiği şekilde ifade edilebilir.

$$y_0' = \frac{1}{n}\sum_{i=1}^{n} y_i \quad (3.13)$$

Burada:

- y_0' başlangıçtaki varsayılan değeridir (genel ortalama),
- n veri setindeki toplam örnek sayısını,
- y_i ise her bir gözlemin gerçek değerini ifade eder.

Bu başlangıç varsayımı kullanıldıktan sonra, her gözlem için kalan hata (residual) hesaplanır. Bu hata, her bir veri noktasının gerçek değeri ile başlangıçta varsayılan nokta arasındaki farktır ve Denklem 3.14'de gösterildiği gibi hesaplanır:

$$residual_i = y_i - y_0' \quad (3.14)$$

Bu residual'lar, modelin hata yapısına dair bir fikir verir ve algoritmanın sonraki adımlarda bu hataları minimize etmeye çalışması sağlanır. CatBoost, her iterasyonda, hata değerlerini analiz ederek bir sonraki modelleme aşamasında yeni ağaçların nasıl eğitileceğine karar verir. Amaç, kalan hataları azaltarak modelin doğruluğunu artırmak ve gerçek değerlere daha yakın taklaşımlar yapmaktır.

CatBoost, her yeni iterasyonda, kalan hataları minimize etmek için güncellenmiş bir model oluşturarak sürece devam eder. Bu güncelleme süreci boyunca kullanılan kayıp fonksiyonu (örneğin, ortalama kare hata – Mean Squared Error) modelin daha düşük bir hata değerine ulaşması için optimize edilir. MSE Denklem 3.15'te verilmektedir.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i') \quad (3.15)$$

Her iterasyonda, model güncellenmiş sonuçlar elde eder ve artık değerler yeniden hesaplanır. Bu süreç, hata değeri kabul edilebilir bir seviyeye düşene veya belirlenen maksimum iterasyon sayısına ulaşılana kadar devam eder. Bu sistematik hata hesaplama ve iyileştirme süreci, CatBoost'un hataları giderek azaltmasına ve daha doğru bir model elde etmesine olanak tanır.

Yeni Ağaçların Eklenmesi ve Gradyan Düşüşü: CatBoost, gradyan artırma yöntemi ile her yeni ağacı ekler, ancak burada önyargı (bias) etkisini azaltmak için benzersiz bir yöntem uygular. Klasik gradyan artırma algoritmalarında yeni bir ağaç, önceki modelin hatalarını azaltmak üzere eğitilir. Bu aşamada gradyan düşüşü, yani kayıp fonksiyonunun türevini minimize etmek için kullanılır.

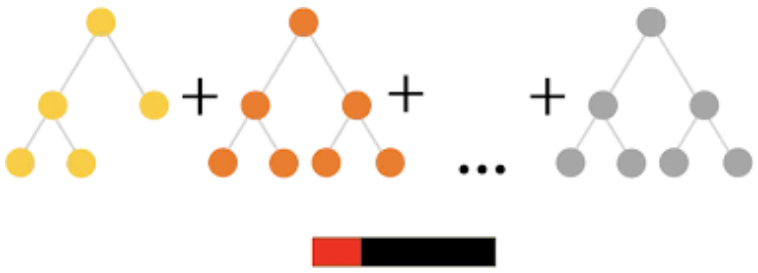
CatBoost ise, sıralı öğrenme (ordered boosting) yöntemi ile çalışır. Bu yöntemde her yeni ağaç, bir önceki ağacın ürettiği hatayı önyargıya düşmeden öğrenir.

Gradyan düşüşü adımında CatBoost, sıralı olarak veriyi işler ve her yeni ağacı eklerken aşamalı öğrenme uygular. Sıralı öğrenme sürecinde, veri seti belli bir sıraya göre işlenir ve önceki örneklerin kullanımı, önyargıya yol açabilecek olasılıkları minimize eder. Böylece model, her seferinde kalan hatayı minimize ederek daha doğru öngörülerde bulunur. CatBoost da kullanılan kayıp fonksiyonu Denklem 3.16'da verildiği gibidir.

$$Obj(t) = \sum_{i=1}^{n} \left(y_i, {y_i}'^{(t-1)} + f_t(x_i) \right) + \Omega(f_t) \tag{3.16}$$

Bu formülde $\Omega(f_t)$ yine ağaç karmaşıklığını ifade eder ve CatBoost'ta ağaç yapısı daha özenli belirlenir.

Topluluk Modeli: CatBoost, gradyan artırma sürecinde birçok ağacı birleştirerek nihai topluluk modelini oluşturur. Topluluk modeli, her ağaçtan elde edilen sonuçları birleştirir ve daha güçlü bir performans elde eder. Her bir ağacın sonucu belirli ağırlıklarla birleştirilir ve toplam modelin genel performansı artar. Ağaçların birleştirilmesi temsilen Şekil 3.13'te gösterilmektedir [115].



Şekil 3.13. Ağaçların birleştirilmesi

CatBoost, her ağacın etkisini optimize ederek genel modeli daha etkili hale getirir. Tüm ağaçlardan elde edilen varsayımlarının ortalaması veya ağırlıklı birleştirmesi sonucu topluluk modelinin genel ortalama hesaplanması Denklem 3.17'de verilmiştir.

$$y' = \sum_{t=1}^{T} \alpha_t \, f_t(x) \tag{3.17}$$

Bu yapıda CatBoost, tüm ağaçların belirli bir ağırlıkla katkıda bulunduğu bir yapı kurar ve sonuçlarda daha isabetli sonuçlar elde eder.

Kategorik Verilerin Kodlanması: CatBoost'un en dikkat çekici özelliklerinden biri, kategorik veri işleme yeteneğidir. Diğer gradyan artırma algoritmalarında, kategorik verilerin sayısal verilere çevrilmesi (label encoding veya one-hot encoding gibi) gerekir. Ancak bu yaklaşımlar yüksek boyutlu veya karmaşık kategorik verilerde verimli olmayabilir. CatBoost, sıcak kodlama (target-based encoding) yöntemi ile kategorik verileri sayısal değerlere dönüştürür.

Kategorik değişkenler için, veriler sıralı bir şekilde işlenir ve bir kategorinin ortalama hedef değeri o sıraya göre hesaplanır. Bu işlem sıralı olarak yapıldığı için önyargı riski en aza indirilir ve eğitim süreci daha sağlıklı ilerler. CatBoost, hedef tabanlı kodlama için Denklem 3.18'de ki formül kullanır. Denklemde k: kategori değerinin hedef ortalamasını, x: ön yargı azaltmak için eklenen rastgele değeri, n: kategori sayısını ifade etmektedir.

$$Elde\ Edilen\ Kodlama = \frac{k+x}{n} \tag{3.18}$$

Bu yöntem, modelin kategorik veriler üzerinde daha iyi performans göstermesini sağlar.

Öğrenme Hızı ve Düzenlileştirme: CatBoost'ta öğrenme hızı ve düzenlileştirme teknikleri, modelin aşırı uyumlamaya karşı dayanıklı olmasını sağlar.

- **Öğrenme Hızı (Learning Rate):** Her bir ağacın katkısı öğrenme hızına bağlı olarak küçültülür. Böylece model küçük adımlarla öğrenir ve aşırı uyum riskini azaltır. Öğrenme hızı, her yeni ağacın etkisini sınırlandırır. Hesaplama Denklem 3.19'da verilmektedir.

$$y' = \sum_{t=1}^{T} \eta\alpha\, f_t(x) \tag{3.19}$$

Burada η, öğrenme hızıdır ve modelin her adımda ne kadar ilerleyeceğini belirler.

- **Düzenlileştirme (Regularization):** CatBoost, XGBoost gibi L2 düzenlileştirme kullanır. Bu düzenlileştirme yöntemi, ağaçların karmaşıklığını kontrol eder ve modelin gereksiz karmaşık yapılara sahip olmasını engeller. L2 düzenlileştirme, modelin aşırı uyum riskini azaltarak genelleme performansını artırır. Düzenlileştirme için CatBoost'un kullanabileceği bir ceza terimi formülü Denklem 3.20'de verilmektedir.

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} {w_j}^2 \tag{3.20}$$

Bu formülde γ, yeni yaprakların karmaşıklık cezasını, λ ise L2 düzenlileştirme katsayısını ifade eder. Bu terimler modelin daha dengeli bir şekilde öğrenmesini sağlar.

CatBoost modelinin performansını optimize etmek için çeşitli parametreleri vardır. CatBoost modelinin başlıca parametreleri şunlardır:

iterations (n_estimators): Toplam ağaç sayısını belirler. Fazla değer aşırı öğrenmeye yol açabilir; erken durdurma ile kontrol edilebilir.

learning_rate: Her ağacın etkisini ayarlar. Düşük değer daha hassas, yüksek değer hızlı öğrenme sağlar.

depth: Ağaçların maksimum derinliğini ayarlar. Derin ağaçlar daha karmaşık öğrenir, ancak aşırı uyum riski taşır.

l2_leaf_reg: L2 düzenlemesi yaparak aşırı öğrenmeyi azaltır.

border_count: Kategorik verilerde kullanılacak sınır sayısını ayarlar; küçük veri kümelerinde daha düşük sınır kullanılır.

random_strength: Rastgeleliği kontrol eder; aşırı uyumun önlenmesine yardımcı olur.

bagging_temperature: Rastgele örneklemelerin sıcaklığını ayarlar. Yüksek değer daha çeşitli modeller elde eder.

eval_metric: Performans ölçümü için metriği belirler (örneğin, Accuracy, RMSE).

one_hot_max_size: Büyük kategorik değişkenlerde, daha uygun bir kodlama yöntemi seçer.

od_type ve od_wait: Erken durdurmayı ayarlar. od_type, erken durdurmayı etkinleştirir, od_wait ise kaç iterasyon sonra durdurulacağını belirler.

cat_features: Kategorik değişkenleri belirtir. CatBoost, bu değişkenleri otomatik işler.

Bu modelin performansı doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metriklerle değerlendirilir.

Denetimsiz Öğrenme (Unsupervised Learning):

Denetimsiz öğrenme, etiketlenmemiş verilerle çalışarak verideki gizli yapıları ve kalıpları keşfetmeyi amaçlayan bir makine öğrenmesi yöntemidir. Model, herhangi bir çıktı etiketi olmadan verideki ilişkiyi ve benzerlikleri bulmaya çalışır. Denetimsiz öğrenme genellikle veri kümelerini gruplandırma (kümeleme) ve boyut azaltma gibi problemlerde kullanılır. Bu yöntem, özellikle

büyük veri setlerinde anlamlı yapıları tanımlamak ve sınıflandırma yapmadan önce veri özelliklerini analiz etmek için etkili bir yaklaşımdır.

Denetimsiz öğrenmede model performansı, denetimli öğrenmedeki gibi doğruluk (accuracy) veya kesinlik (precision) gibi metriklerle ölçülemez, çünkü doğru bir çıktı etiketi yoktur. Bunun yerine, genellikle içsel değerlendirme ölçütleri veya gözlemsel analiz yöntemleri kullanılır.

Pekiştirmeli Öğrenme (Reinforcement Learning):

Pekiştirmeli Öğrenme, bir modelin veya "ajan"ın bir ortamla etkileşime girerek ödül veya ceza mekanizmaları aracılığıyla kendi stratejisini geliştirdiği bir makine öğrenmesi yöntemidir. Bu öğrenme türünde, ajanın amacı belirli bir görevi en iyi şekilde yerine getirmek için en uygun eylemleri öğrenmek ve maksimum ödül elde etmektir. Ajan, her bir adımda çevreden bir durum (state) algılar, bu duruma göre bir eylem (action) seçer ve karşılığında bir ödül veya ceza alır. Bu döngüsel süreç, ajanın gelecekteki eylemlerini daha iyi hale getirmesi için ödül bazlı bir öğrenme sağlar.

Pekiştirmeli öğrenmenin temel bileşenleri şunlardır:

- Ajan (Agent): Ortamda hareket eden ve eylemler gerçekleştiren sistem veya model.
- Ortam (Environment): Ajanın içinde bulunduğu ve etkileşimde olduğu çevre.
- Durum (State): Ajanın bulunduğu anda çevreden aldığı durum bilgisi.
- Eylem (Action): Ajanın bir durumda gerçekleştirdiği hareket.
- Ödül (Reward): Ajanın belirli bir durumda gerçekleştirdiği eylemin sonucunda aldığı geri bildirim.

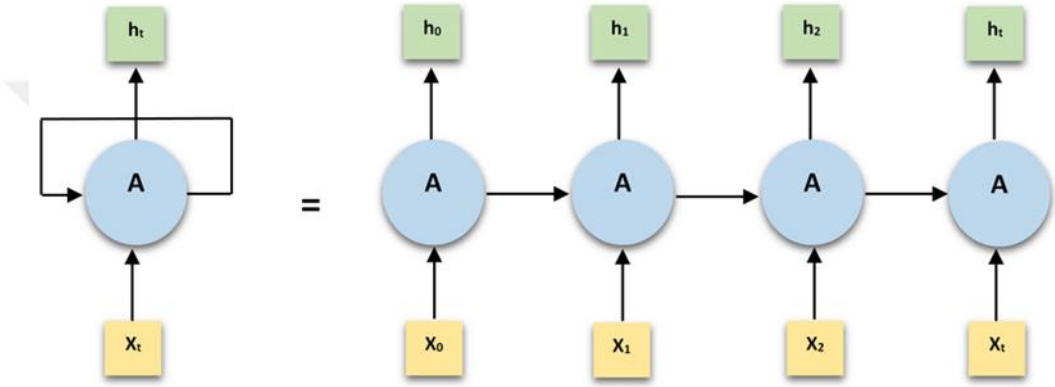
Pekiştirmeli öğrenme, özellikle robotik, oyun teorisi ve özerk araçlar gibi alanlarda uygulanmaktadır. Örneğin, bir robotun belirli bir görevi yerine getirmek için deneme yanılma yoluyla çeşitli hareketler denemesi ve bu hareketlerin sonucunda aldığı ödüllerle strateji geliştirmesi pekiştirmeli öğrenmenin tipik bir örneğidir.

3.7. Derin Öğrenme

Derin öğrenme, bilgisayarların deneyim ve verilerden öğrenerek belirli görevleri yerine getirme yeteneğini geliştiren bir yapay zeka dalıdır. Bu teknoloji, bilgisayarların önceden programlanmış talimatlar yerine, verilerden kalıpları ve ilişkileri öğrenerek kararlar vermesini sağlar. Derin öğrenme, özellikle büyük veri setleri ve çok katmanlı sinir ağları kullanarak karmaşık problemleri çözme yeteneğiyle öne çıkar.

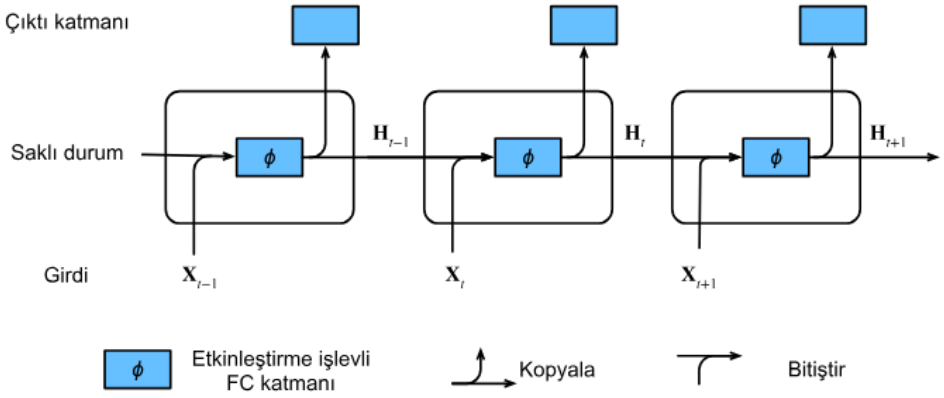
3.7.1. RNN

Tekrarlayan Sinir Ağları (RNN), sıralı veri üzerinde hesaplama yapabilme yeteneği sayesinde özellikle DDİ, zaman serisi analizi ve video işleme gibi ardışık verilerde yaygın olarak kullanılan bir yapay sinir ağı türüdür. RNN'ler, her adımda gizli bir durum (h_t) güncelleyerek veri geçmişini saklar ve bu bilgiyi sıralı adımlar boyunca aktarır. Bu sayede, önceki adımlardaki bilgiye dayanarak gelecekte daha doğru genellemeler yapabilir. RNN in genel çalışma yapısı Şekil 3.14'te gösterildiği gibidir [116].



Şekil 3.14. RNN çalışma yapısı

RNN'lerde her bir adımda gizli durum, önceki gizli durum (h_{t-1}) ve o adımın girdisi (x_t) kullanılarak güncellenir. Bu durum güncelleme yapısı Şekil 3.15'te gösterildiği gibidir [117].



Şekil 3.15. RNN durum güncelleme yapısı

Bu durum güncelleme yapısı, Denklem 3.21'de gösterilen matematiksel ifade ile açıklanabilir.

$$h_t = f(W_h . h_{t-1} + W_x . x_t + b_h) \quad (3.21)$$

Burada:

- h_t : O anki gizli durumdur,
- x_t : Girdi vektörüdür,
- $W_h$ve W_h: Ağırlık matrisleridir,
- b_h : Bias terimidir,
- f : Genellikle tanh veya ReLU gibi aktivasyon fonksiyonlarıdır.

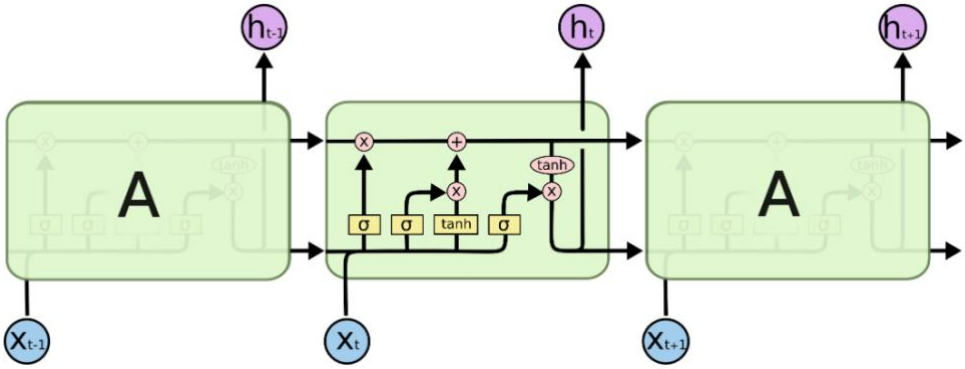
RNN yapısının bu özellikleri, sıralı veri üzerindeki bağımlılıkları modelleyebilme avantajını sağlar. Ancak, basit RNN yapıları, uzun süreli bağımlılıkları korumakta zorluk çekmektedir. Bu durum, "gradyan kaybı" olarak bilinen bir soruna neden olur ve bu sorun, özellikle geriye yayılım (backpropagation) sırasında uzun dizilerde bilgi kaybına yol açar. Bu nedenle, RNN'ler uzun vadeli bağımlılıkların modellenmesinde yetersiz kalabilmektedir.

Gradyan kaybı sorununa çözüm olarak, LSTM (Long Short-Term Memory) ve GRU (Gated Recurrent Unit) gibi gelişmiş RNN varyantları geliştirilmiştir. Bu tür yapılar, kapı mekanizmaları kullanarak bilgi akışını kontrol eder ve uzun vadeli bağımlılıkları daha etkili bir şekilde öğrenebilir hale gelir. Bu tür yapılar, DDİ, zaman serisi analizi, duygu analizi ve diğer sıralı veri işleme görevlerinde geniş bir uygulama alanına sahiptir.

Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. Katman sayısı ve hücre yapısının uygun şekilde seçilmesi, RNN'in doğruluğunu artırarak daha iyi genellenebilir sonuçlar elde edilmesini sağlar.

LSTM

LSTM, RNN modelindeki dezavantajlı durumları ortadan kaldırmak için geliştirilen RNN modelinin bir çeşididir [118]. İnsanlardaki kısa ve uzun süreli bellek mekanizmalarından esinlenerek tasarlanan LSTM, özellikle uzun vadeli bağımlılıkları öğrenme yeteneğiyle dikkat çeker ve RNN'lerde karşılaşılan gradyan kaybı sorununu çözmek için optimize edilmiştir. LSTM'nin temel yeniliği, uzun vadeli bağımlılıkları koruyabilen hafıza hücreleridir. Bu hücrelerin yapısı Şekil 3.16'da gösterilmektedir [119].



Şekil 3.16. LSTM hücre yapısı

Bu hücreler, giriş verisini alır, geçmiş durum bilgilerini saklar ve gerekli durumlarda bu bilgiyi gelecekteki adımlarda kullanır. Bu şekilde, RNN'lerin zaman içinde kaybolan bellek sorununu çözerek uzun süreli veri ilişkilerini daha etkili bir şekilde modelleyebilir.

LSTM, bu hafıza hücrelerini koruma, unutma ve güncelleme işlemleriyle yönetir. Bu işlemler şu şekilde açıklanabilir:

Unutma Kapısı (f_t): LSTM, her adımda hangi bilginin korunup hangisinin unutulacağını belirler. Formül, Denklem 3.22'de verilmiştir.

$$f_t = \sigma\left(W_f.\left[h_{t-1,}x_t\right] + b_f\right) \tag{3.22}$$

Burada σ sigmoid aktivasyon fonksiyonudur, W_f unutma kapısının ağırlıkları, $h_{t-1,}$ önceki gizli durum, $x_t$mevcut girdi ve $b_f$bias terimidir.

Giriş Kapısı (i_t): Yeni gelen bilginin hafıza hücresine nasıl ekleneceğini kontrol eder. Formüller sırasıyla Denklem 3.23 ve Denklem 3.24'te belirtilmiştir.

$$i_{t=}\sigma\left(W_i.\left[h_{t-1,}x_t\right] + b_i\right) \tag{3.23}$$

$$C_t' = \tanh\left(W_c.\left[h_{t-1,}x_t\right] + b_C\right) \tag{3.24}$$

i_t giriş kapısını C_t' ise yeni hücre adayını temsil eder. Tanh hiperbolik tanjant fonksiyonudur. W_i ve W_i ağırlık matrisleri, b_i ve b_C bias terimleridir.

Hücre Durumu Güncellemesi (C_t): Hücre durumunun yeni değerini hesaplar. Formül Denklem 3.25'te verilmiştir.

$$C_t = f_t . C_{t-1} + i_t . C'_t \quad (3.25)$$

Bu formül, önceki hücre durumunun (C_{t-1}) unutma kapısından gelen bilgiyle ölçeklenerek korunmasını ve giriş kapısı aracılığıyla yeni bilginin eklenmesini sağlar.

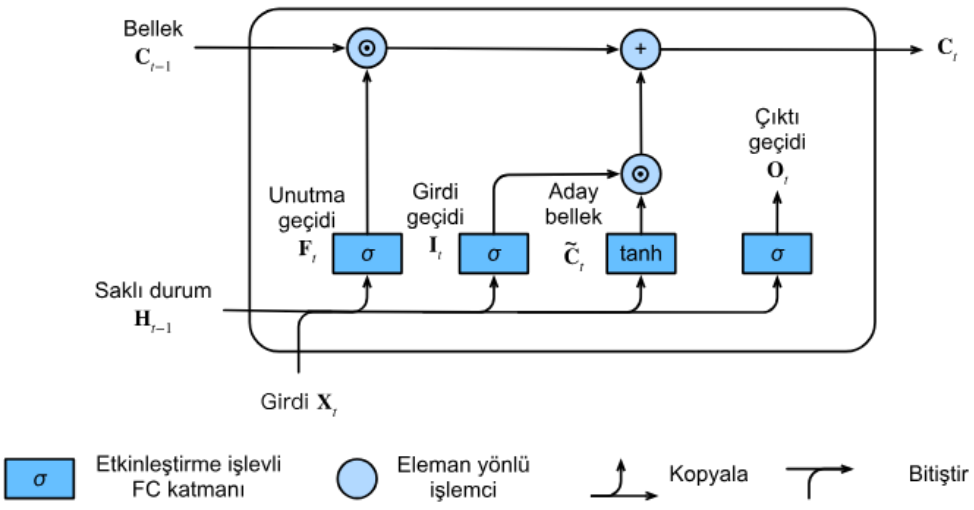
Çıkış Kapısı (o_t): Hücre durumunu gizli duruma nasıl yansıtacağını kontrol eder. Formüller sırasıyla Denklem 3.26 ve Denklem 3.27'de verilmiştir.

$$o_t = \sigma(W_0 . [h_{t-1,} x_t] + b_0) \quad (3.26)$$

$$h_t = o_t . \tanh(C_t) \quad (3.27)$$

Çıkış kapısı $\mathbf{o_t}$ hesaplanarak, güncellenmiş hücre durumu C_t ile birlikte gizli durum (h_t) oluşturulur.

Bu kapılar aracılığıyla LSTM, her bir zaman adımında bilgiyi saklama ve işleme kararlarını daha hassas bir şekilde yapar, böylece uzun vadeli bağımlılıkları koruma yeteneğini artırır ve gradyan kaybı sorununu büyük ölçüde ortadan kaldırır. LSTM in kapı yapısının mimarisi Şekil 3.17'de gösterildiği şekildedir [120].



Şekil 3.17. LSTM kapı mimarisi

Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. Bellek hücrelerinin ve katman derinliğinin uygun şekilde ayarlanması, LSTM'in doğruluğunu artırarak daha iyi genellenebilir sonuçlar elde edilmesini sağlar.

GRU

Geçitli Tekrarlayan Birim (Gated Recurrent Unit - GRU), LSTM'deki işlem karmaşıklığını azaltmak amacıyla geliştirilen bir RNN varyantıdır. GRU mimarisi, LSTM yapısına benzer bir tasarıma sahiptir ancak daha basit ve daha az hesaplama gerektiren bir yapıya sahiptir. LSTM'de dört kapı bulunurken, GRU'da yalnızca iki kapı bulunmaktadır. Bu iki kapı, algoritmanın daha az sayıda işlemle çalışmasını sağlayarak GRU modelinin LSTM'e göre daha hızlı sonuçlar vermesine imkan tanır [121]. GRU' nun temel denklemleri aşağıdaki gibidir:

Güncelleme Kapısı (z_t): LSTM modelindeki unutma kapısı ve giriş kapısının birleşiminden oluşur. Hangi bilginin korunacağına ve yeni bilginin ne ölçüde ekleneceğine karar verir. Formül Denklem 3.28'de verilmiştir.

$$z_t = \sigma\left(W_z.\left[h_{t-1,}x_t\right] + b_z\right) \tag{3.28}$$

Burada σ sigmoid aktivasyon fonksiyonudur, W_z güncelleme kapısının ağırlık matrisi, $h_{t-1,}$ önceki gizli durum, x_t giriş ve b_z bias terimidir.

Sıfırlama Kapısı (r_t): Önceki gizli durumdan ne kadar bilginin kullanılacağını belirler. Formülü Denklem 3.29'da verilmiştir.

$$r_t = \sigma\left(W_r.\left[h_{t-1,}x_t\right] + b_r\right) \tag{3.29}$$

W_r sıfırlama kapısının ağırlık matrisi ve b_r bias terimidir.

Geçici Bellek Durumu (h_t'): Güncelleme ve sıfırlama kapılarının çıktıları kullanılarak oluşturulan geçici gizli durumdur. Formülü Denklem 3.30'da verilmiştir.

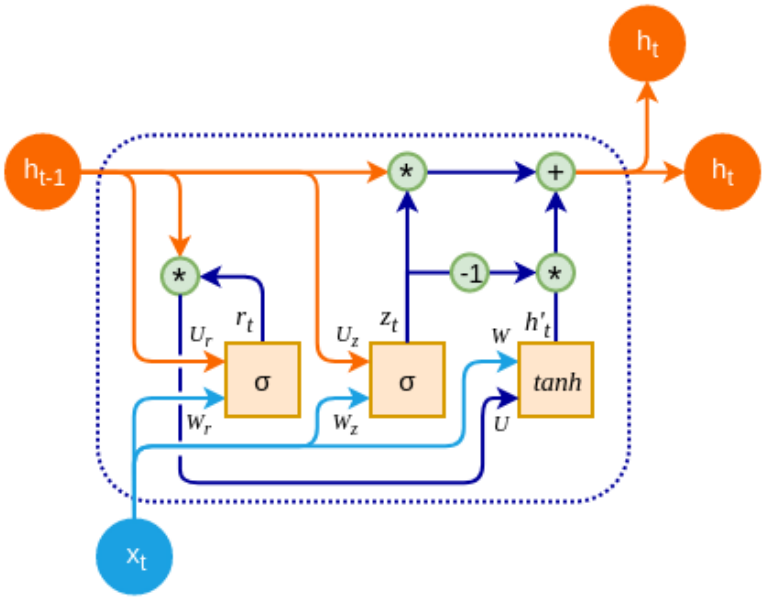
$$h_t' = \tanh\left(W_h.\left[r_t * h_{t-1,}x_t\right] + b_h\right) \tag{3.30}$$

Burada tanh, hiperbolik tanjant aktivasyon fonksiyonudur, r_t sıfırlama kapısından gelen çıktı ve W_h geçici bellek durumu için ağırlık matrisidir.

Gizli Durum Güncellemesi (h_t): Nihai gizli durum, güncelleme kapısının çıktısına göre geçici bellek durumu ve önceki gizli durumun birleşiminden oluşturulur. Formülü Denklem 3.31'de verilmiştir.

$$h_t = z_t * h_{t-1} + (1 - z_t) + h_t' \tag{3.31}$$

Bu formül, güncelleme kapısı z_t ile önceki gizli durumu (h_{t-1}) ve geçici bellek durumu (h'_t) arasında bir denge kurarak nihai gizli durumu hesaplar. GRU nun mimari yapısı Şekil 3.18'de gösterilmektedir [122].



Şekil 3.18. GRU mimarisi

GRU, LSTM gibi gradyan kaybı sorununu ele almak için tasarlanmıştır ancak daha basit bir yapıya sahiptir. LSTM'e göre daha hızlı eğitim süresi sağlar ve daha az sayıda parametreye ihtiyaç duyar. Bu özellikler, GRU'nun, LSTM gibi hafıza hücrelerine sahip olmasını sağlarken, daha az kontrol mekanizması ile işlem karmaşıklığını azaltır ve eğitim verimliliğini artırır.

Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. Kapı mekanizmalarının ve katman sayısının uygun şekilde seçilmesi, GRU'nun doğruluğunu artırarak daha iyi genellenebilir sonuçlar elde edilmesini sağlar.

BİLSTM

İki Yönlü Uzun Kısa Süreli Bellek (BiLSTM), ileri ve geri yönde çalışan iki ayrı LSTM katmanına sahip bir derin öğrenme modelidir. Bu model, sıralı verilerde hem geçmiş hem de gelecek bilgileri kullanarak iki yönlü bağımlılıkları öğrenme yeteneğine sahiptir. BiLSTM, her bir veri noktasının hem önceki hem de sonraki zaman adımlarındaki bilgilerle ilişkilendirildiği durumlarda, özellikle metin işleme ve konuşma tanıma gibi sıralı veri analizlerinde güçlü bir performans sunar.

Bu yapı, tek yönlü LSTM'lerin aksine, verinin tamamındaki bilgileri aynı anda işleyebilme kapasitesi sayesinde, bağlamın iki yönlü etkilerini modelleyerek daha doğru sonuçlar elde edilmesini sağlar. Bu modelin performansı doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metriklerle değerlendirilir. İleri ve geri yönlü katman sayısının uygun şekilde ayarlanması, BiLSTM'in doğruluğunu artırarak daha iyi genellenebilir sonuçlar elde edilmesini sağlar.

3.8. Transfer Öğrenmesi

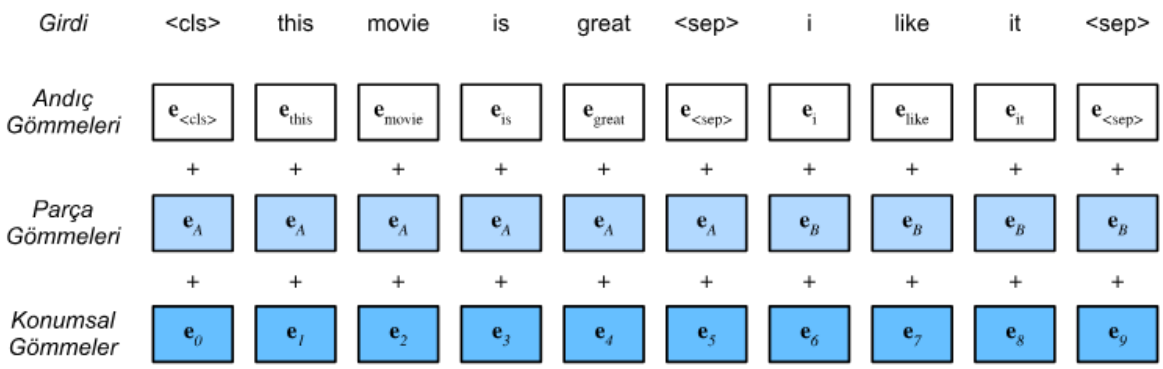
Transfer öğrenmesi, bir problem için benzer bir görevde büyük veri seti ile daha önce eğitilmiş bir model kullanan derin öğrenme yöntemidir. Bir modeli sıfırdan eğitmek yerine, transfer öğrenmeyi kullanarak modele ince ayar (fine tuning) yapmak çok daha hızlı ve kolay bir öğrenme yöntemidir [123]. Örneğin insan hayvan sınıflandırmasını yapan bir model eğitilirken kazanılan bilgi kadın erkek sınıflandırmasında transfer edilerek kullanılabilir.

Probleme uygun transfer öğrenmesi yöntemi daha az veri ile daha yüksek performans elde edilmesini mümkün kılmakla birlikte model tasarımcısına zamandan kazanç sağlamaktadır [124]. Transfer öğrenme ile yapay öğrenme modelinin öğrendiklerinin tamamı ya da bir kısmı transfer edilerek benzer bir problemin çözümü daha hızlı ve kolay olabilmektedir. Öğrenme aktarımı sürecinde ne aktarılacak ne zaman ve nasıl aktarılmalı soruları önemlidir. Kaynak veriden hedef görevin başarımını arttırmak için hangi bilginin aktarılması gerektiğinin belirlenmesi ilk ve en önemli adımdır. Burada bazı bilgilerin kullanılan veriye özgü olabileceği dikkate alınarak kullanılan alan ne olursa olsun ortak bilginin aktarılacak bilgi olarak seçilmesi gerekmektedir. İkinci adımda ise verinin hangi aşamada aktarılması gerektiğinin belirlenmesidir.

Kaynak veri ile hedef veri birbirinden çok farklı ise aktarımın temel adımlar işlendikten sonra ilk katmanlarda aktarım yapılması önemlidir. Aksi takdirde hedef veri üzerinde başarım düşebilir hatta negatif transfer söz konusu olabilir. Son olarak bilginin aktarımı için hangi yöntemler kullanılmalı kararı gelir ki burada var olan standart yöntemler kullanılabilir ya da bu yöntemlerin değiştirilmesi yaklaşımı uygulanabilir.

3.8.1. BERTurk

2018 yılında Google çalışanları tarafından tanıtılan BERT (Bidirectional Encoder Representations from Transformers), DDİ alanında devrim yaratan bir sinir ağı mimarisidir. BERT, yapay zeka ve makine öğrenimi algoritmalarını kullanarak, dilin anlamını daha doğru bir şekilde analiz edebilen yeni bir DDİ tekniği sunar. Bu modelin en büyük özelliği, kelimeleri bağlam içinde iki yönlü (bidirectional) bir yaklaşımla işlemesidir [29]. Çift yönlü mimari yapısına örnek Şekil 3.19'da gösterilmektedir [125].



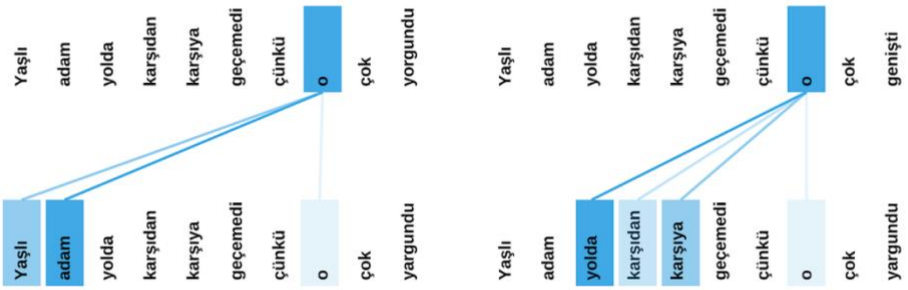
Şekil 3.19. Çift taraflı mimari yapısı

Bu sayede, kelimenin yalnızca öncesindeki veya sonrasındaki kelimelere bakmak yerine, her iki tarafındaki kelimeleri de dikkate alır. Bu özellik, BERT'in kelimeleri bağlamlarıyla birlikte daha iyi anlamasına ve daha isabetli sonuçlar elde edilmesine olanak sağlar.

BERT**,** Ön Eğitim (Pre-training) ve İnce Ayar (Fine-tuning) olmak üzere 2 ana aşamada çalışmaktadır**.**

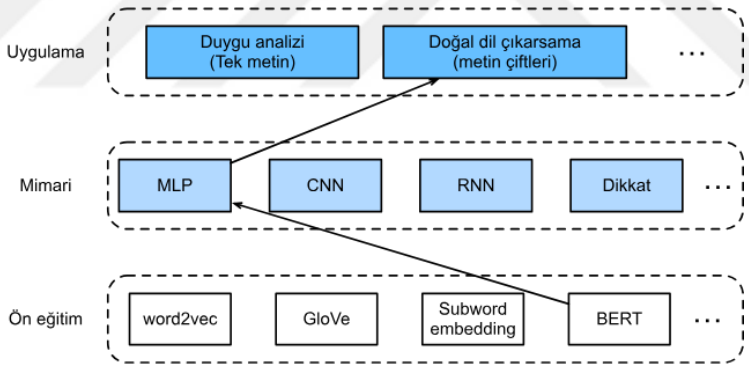
1. **Ön Eğitim (Pre-training):** BERT modeli, geniş bir metin korpusu üzerinde iki temel görevi yerine getirerek eğitilir:

- Maskelenmiş Dil Modelleme (Masked Language Model - MLM): Bu görevde, metin içindeki bazı kelimeler rastgele "MASK" olarak gizlenir ve modelden bu gizli kelimeleri değerlendirmesi istenir. Örneğin, "Bu kitap çok [MASK]" gibi bir cümlede, model "güzel" ya da "eğlenceli" gibi uygun bir kelime kestirmeye çalışır. Bu işlem, modelin kelimelerin bağlam içindeki anlamını öğrenmesine yardımcı olur.
- Cümle Çifti Tahmini (Next Sentence Prediction - NSP): Bu görevde, model iki cümlenin birbiri ardına gelip gelmediğini varsayar. Bu görev, cümleler arasındaki bağı öğrenmesini sağlar ve özellikle soru-cevap ve metin anlama gibi DDİ uygulamalarında faydalıdır. Örneğin, "Ali dışarı çıktı." ve "Yağmur yağmaya başladı." cümleleri arasında bir ilişki vardır, bu nedenle model bu tür ilişkileri tanımayı öğrenir. Kelimelerin bağlamlarını belirleyerek ilişkiyi tanımaya örnek görsel Şekil 3.20'de verilmiştir.



Şekil 3.20. Kelimeler arası bağlam ve ilişkiyi hesaplama

2. **İnce Ayar (Fine-tuning):** Önceden eğitilmiş BERT modeli, belirli bir DDİ görevine (örneğin, duygu analizi, soru-cevap sistemi, veya metin sınıflandırma) uyacak şekilde daha küçük bir veri seti üzerinde yeniden eğitilir. Bu aşamada, model yalnızca ilgili göreve uygun olarak özelleştirilir ve çok daha iyi sonuçlar elde edebilir. İnce ayar yapısına örnek Şekil 3.21'de gösterilmektedir [126].



Şekil 3.21. BERT ince ayar yapısı

BERT, Transformer mimarisi kullandığı için aynı zamanda kendi kendine dikkat (self-attention) mekanizmasından yararlanır. Bu mekanizma, her bir kelimenin diğer kelimelerle olan ilişkisini öğrenmesine olanak tanır. Formülü Denklem 3.32'de verilmiştir.

$$Dikkat(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3.32}$$

Burada:

- Q, K ve V sırasıyla sorgu (query), anahtar (key), ve değer (value) matrisleridir.

- d_k, sorgu ve anahtar matrislerinin boyutudur.
- Softmax fonksiyonu, dikkat skorlarını olasılıklara dönüştürür.

BERT'in Avantajları ve DDİ Uygulamalarındaki Kullanımı

BERT, DDİ alanında çığır açan yenilikler getirmiştir ve birçok avantaj sunar:

- Bağlam Anlayışı: BERT, kelimeleri iki yönlü olarak değerlendirir ve kelimelerin bağlam içindeki anlamını daha doğru bir şekilde çıkarır. Bu, metnin bütünsel anlamını anlamada büyük bir avantaj sağlar.
- Evrensel Temsil Modeli: BERT, dilin evrensel temsilini öğrenir ve çeşitli DDİ görevlerine kolayca uyarlanabilir. Bu da, BERT'in farklı görevlerde yüksek performans göstermesini sağlar.
- Yüksek Performans: BERT, birçok DDİ görevinde en yüksek performansa ulaşarak, "state-of-the-art" yani en iyi sonuçları elde etmiştir. Özellikle metin sınıflandırma, duygu analizi, bilgi çıkarımı, makine çevirisi ve soru-cevap sistemleri gibi alanlarda üstün başarı sağlamaktadır.

Türkçe DDİ Alanında BERT'in Rolü

BERT, Türkçe dil işleme görevlerinde de büyük bir yenilik getirmiştir. Türkçe, bağlam ve eklemeli yapısı nedeniyle DDİ modelleri için zorluklar sunar. BERT'in Türkçe uyarlamaları, Türkçe metinlerin bağlamını ve dil yapısını daha iyi anlayan modellerin geliştirilmesine olanak sağlamıştır.

- Türkçe Dil Modeli Geliştirme: Türkçe metinler üzerinde eğitilmiş BERT modelleri, Türkçe dilinin özelliklerini ve yapısını daha iyi anlama yeteneğine sahiptir. Bu, Türkçe DDİ çalışmalarında daha doğru sonuçlar elde edilmesine yardımcı olur.
- Türkçe Metin İşleme: BERT, Türkçe metinlerdeki kelime anlamlarını, eşanlamlı kelimeleri ve bağlamsal ilişkileri daha doğru bir şekilde analiz edebilir. Bu özellik, Türkçe metinler üzerinde duygu analizi, konu modelleme ve bilgi çıkarımı gibi uygulamalarda etkili sonuçlar sunar.

Sonuç olarak, BERT DDİ alanında çığır açan bir teknolojidir. Google tarafından geliştirilen bu model, dilin bağlamını iki yönlü işleme sayesinde daha iyi anlayarak metinlerin anlamını doğru bir şekilde çıkarır. BERT'in Türkçe DDİ alanında da önemli katkıları bulunmaktadır. Türkçe metinlerin dil yapısına ve bağlamına duyarlı bir analiz sağlayarak dil işleme modellerinin doğruluğunu artırır. Akademik çalışmalar ve yüksek lisans tezlerinde, BERT'in sunduğu güçlü özellikler ve yüksek performans, dil işleme alanında daha doğru sonuçlar elde edilmesine katkıda

bulunur. BERT'in Türkçe dil işleme görevlerinde etkin bir şekilde kullanılması, gelecekteki araştırmalar için büyük bir potansiyel sunmaktadır.

Aşağıdaki Tablo 3.1'de çalışmada kullanılan derin öğrenme modellerinin ve BERTTurk' un hafıza yönetimi, model karmaşıklığı, hesaplama gücü ve uygulama alanlarına göre kıyaslanmalarına değinilmiştir.

Tablo 3.1. Derin öğrenme modellerinin özellikleri

KRİTER	RNN	LSTM	GRU	BERT
Hafıza Yönetimi	Kısa süreli bağımlılıkları öğrenebilir, uzun süreli bağımlılıkları öğrenmede zorluk yaşar.	Uzun süreli bağımlılıkları iyi bir şekilde öğrenir.	LSTM'lere benzer şekilde uzun süreli bağımlılıkları öğrenir, ancak daha az hesaplama gücü gerektirir.	İki yönlü bağlam öğrenimi sayesinde çok daha geniş bir bağlamı ve uzun süreli bağımlılıkları öğrenir.
Model Karmaşıklığı	Basit yapıya sahiptir, ancak uzun süreli bağımlılıkları öğrenmede zorluk yaşar.	Karmaşık kapı mekanizmaları sayesinde uzun süreli bağımlılıkları iyi öğrenir.	LSTM' lere benzer performans sunar, ancak daha basit ve daha hızlıdır.	En karmaşık yapıya sahiptir ve Transformer mimarisi üzerine kuruludur.
Hesaplama Gücü	Daha az hesaplama gücü gerektirir.	RNN'lere göre daha fazla hesaplama gücü gerektirir.	LSTM'lere göre daha az hesaplama gücü gerektirir, ancak RNN'lerden daha fazladır.	Çok yüksek hesaplama gücü ve büyük miktarda veri gerektirir.
Uygulama Alanları	Basit sekanslı veri işleme görevleri.	Uzun sekanslı veri işleme, dil modelleme, zaman serileri, konuşma tanıma.	LSTM ile benzer görevler, özellikle hafıza ve işlem süresi kısıtlamalarının olduğu durumlar.	DDİ, metin sınıflandırma, duygu analizi, bilgi çıkarımı, soru-cevap sistemleri.

3.9. Makine Öğrenmesi Yöntemlerinin Değerlendirme Metrikleri

Makine öğrenmesi modellerinin değerlendirilmesinde kullanılan metrikler, modelin performansını belirlemek ve doğru sonuçlar üretebilme kapasitesini ölçmek açısından oldukça önemlidir. Sınıflandırma problemlerinde sıklıkla kullanılan bu metrikler, modelin ne kadar başarılı olduğunu ve hangi durumlarda iyileştirme gerektiğini analiz etmek için detaylı bilgi sağlar. Bu metrikler

doğruluk (accuracy), kesinlik (precision), duyarlılık (recall), F1 skoru gibi çeşitli ölçütleri içerir ve her biri modelin farklı bir yönünü değerlendirmeye yöneliktir [127].

Karmaşıklık Matrisi (Confusion Matrix) : Hata matrisi, sınıflandırma modellerinin performansını analiz etmek için kullanılan temel bir araçtır. Bu matris, model sonuçları ile gerçek etiketler arasındaki ilişkiyi gösterir ve dört ana bileşenden oluşur:

- True Positive (TP): Modelin doğru bir şekilde pozitif olarak sınıflandırdığı örneklerdir.
- True Negative (TN): Modelin doğru bir şekilde negatif olarak sınıflandırdığı örneklerdir.
- False Positive (FP): Modelin pozitif olarak tahmin ettiği ancak aslında negatif olan örneklerdir. Bu durumda model bir yanlış alarm vermiş olur.
- False Negative (FN): Modelin negatif olarak tahmin ettiği ancak aslında pozitif olan örneklerdir. Bu durumda model, bir pozitif durumu kaçırmış olur.

Hata matrisi, modelin hangi tür hataları yaptığını ve hangi durumlarda doğru olduğunu daha detaylı anlamamızı sağlar.

Doğruluk(Accuracy) : Doğruluk metriği, modelin doğru tahminlerinin tüm tahminler içindeki oranını ifade eder. Modelin toplam doğru tahmin sayısının, toplam tahmin sayısına bölünmesiyle hesaplanır. Doğruluk metriği, tüm sınıfların dengeli olduğu veri setlerinde genellikle iyi bir değerlendirme ölçütüdür; ancak dengesiz veri setlerinde yanıltıcı olabilir. Özellikle pozitif ve negatif sınıflar arasındaki dengesizlik yüksek olduğunda, doğruluk metriği, modelin performansını tam olarak yansıtmayabilir. Formülü Denklem 3.33'te verilmiştir.

$$Doğruluk = \frac{TP+TN}{TP+TN+FP+FN} \tag{3.33}$$

Kesinlik(Precision) : Kesinlik, modelin pozitif olarak tahmin ettiği örneklerin ne kadarının gerçekten pozitif olduğunu ölçer. Özellikle yanlış pozitiflerin sayısının azaltılması gereken durumlarda önemlidir. Kesinlik yüksek olduğunda, modelin pozitif tahminleri güvenilirdir ve yanlış pozitif oranı düşüktür. Bu metrik, modelin yanlış pozitifleri azaltmaya odaklanması gereken alanlarda öne çıkar. Hesaplanması denklem 3.34'te verildiği şekildedir.

$$Kesinlik = \frac{TP}{TP+FP} \tag{3.34}$$

Duyarlılık (Recall) veya Hassasiyet (Sensitivity) :

Duyarlılık, modelin gerçek pozitifleri ne kadar iyi tespit edebildiğini gösterir. Bu metrik, modelin pozitif sınıfı kaçırma (False Negative) oranını azaltmaya yönelik bir ölçü sağlar. Yanlış negatiflerin

yüksek maliyetli veya kritik olduğu durumlarda duyarlılık metriği önem kazanır. Duyarlılık, modelin gerçek pozitifleri ne kadar yakaladığını gösterdiği için özellikle sağlık, güvenlik gibi alanlarda kritik önem taşır.

$$Duyarlılık = \frac{TP}{TP+FN} \tag{3.35}$$

F1 Skoru (F1 Score) :

F1 skoru, kesinlik ve duyarlılığın harmonik ortalamasıdır ve modelin hem yanlış pozitif hem de yanlış negatif oranlarına dikkat etmesi gereken durumlarda kullanışlıdır. F1 skoru, dengeli bir metriğe ihtiyaç duyulduğunda tercih edilir, çünkü hem kesinlik hem de duyarlılığı dikkate alır ve her ikisinin de yüksek olduğu durumlarda en iyi sonuçları verir. F1 skoru, dengesiz veri setlerinde veya sınıf dengesizliklerinin olduğu durumlarda daha doğru bir performans göstergesi sağlar.

$$F1\ Skoru = 2 \times \frac{Kesinlik \times Duyarlılık}{Kesinlik + Duyarlılık} \tag{3.36}$$

Özgüllük (Specificity) :

Özgüllük, modelin negatif sınıfı doğru bir şekilde tanımlama kabiliyetini ölçer. Bu metrik, yanlış pozitiflerin sayısını azaltmayı hedefleyen durumlarda önemlidir. Negatif sınıfların doğru bir şekilde tanımlanması gereken problemlerde özgüllük metriği dikkate alınır. Özellikle negatif sınıfların doğru bir şekilde tanımlanması gerektiğinde, özgüllük yüksek bir değer aldığında model güvenilir bir şekilde negatif örnekleri tanımlayabilir.

$$Özgüllük = \frac{TN}{TN+FP} \tag{3.37}$$

4. BULGULAR VE SONUÇLAR

Bu bölümde, çalışmada kullanılan veri setleri, uygulanan yöntemler ve elde edilen sonuçlar ayrıntılı bir şekilde ele alınmaktadır. Deneysel çalışmalar kapsamında, sosyal medyadan toplanan finans içerikli gönderiler üzerinde yapılan duygu sınıflandırması işlemleri ve bu süreçte kullanılan modellerin performanslarının değerlendirilmesi yer almaktadır. Öncelikle veri setinin özellikleri, ön işleme adımları ve sınıflandırma için kullanılan yöntemler açıklanmıştır. Ardından, farklı makine öğrenimi ve derin öğrenme modelleri üzerinde gerçekleştirilen deneylerin sonuçları ve bu sonuçların analizleri sunulmuştur. Çalışmada, sınıflandırma modellerinin performanslarını değerlendirmek için kesinlik, duyarlılık ve F1 skoru gibi metrikler kullanılmıştır.

Deneysel çalışmaların amacı farklı yöntemlerin sınıflandırma başarısını karşılaştırarak en etkin modeli belirlemektir. Bu kapsamda elde edilen bulgular, finansal piyasa öngörü modellerinin geliştirilmesi için değerli bir kaynak oluşturmaktadır.

4.1.1. LSTM

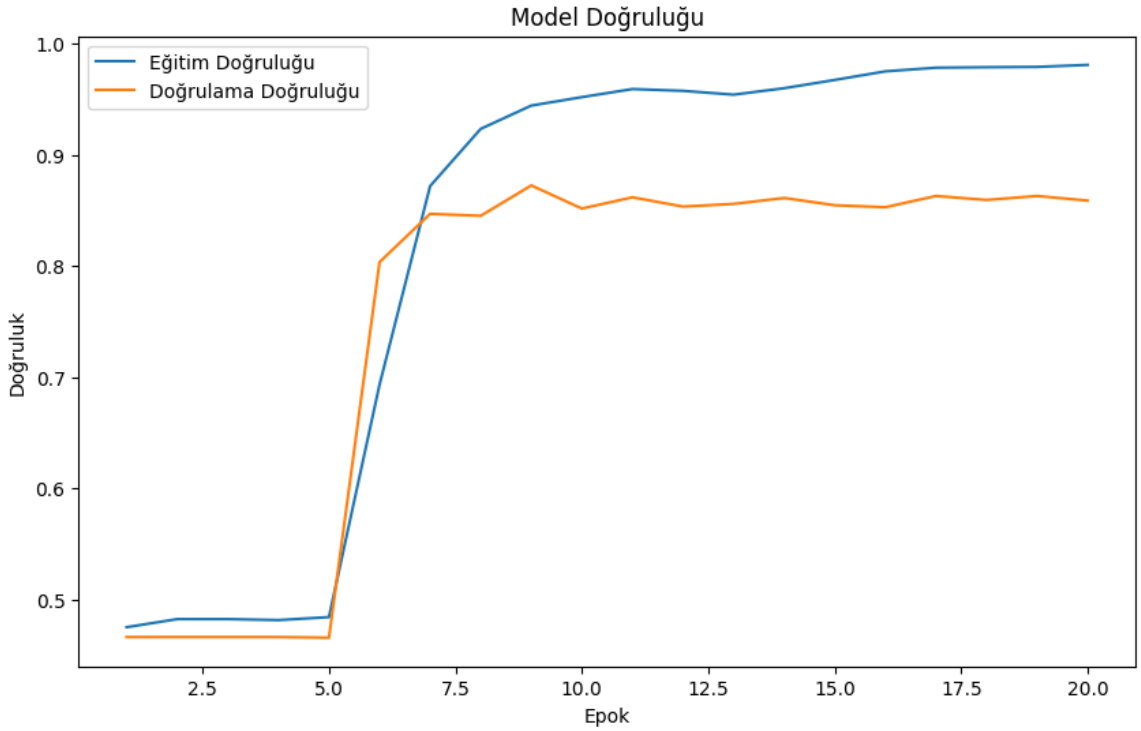
Bu model, metin verisini sayısal dizilere dönüştüren bir gömme (Embedding) katmanı ile iki adet Uzun Kısa Süreli Bellek (LSTM) katmanından oluşmaktadır. Embedding katmanı, kelime dağarcığını sayısal olarak temsil ederek modelin kelimeler arasındaki ilişkileri daha iyi öğrenmesine olanak tanır. LSTM katmanları, metinlerin zaman bağımlı yapısını yakalamak için kullanılmış olup, sıralı veriyi işleyebilme kapasitesini artırmaktadır. Modelin son katmanında ise softmax aktivasyon fonksiyonuna sahip yoğun (dense) bir katman yer almakta, bu katman çıktı sınıflarının olasılıklarını öngörmektedir.

Model, sparse categorical cross-entropy kaybı fonksiyonu ve Adam optimizasyon algoritması ile derlenmiştir. Öğrenme hızı konusunda çeşitli denemeler yapılmış ancak belirli bir öğrenme hızı değeri kullanılmamış; bunun yerine Adam algoritmasının otomatik ayarlamaları ile eğitim süreci yürütülmüştür. Eğitim sırasında, veri setindeki azınlık sınıflarının model tarafından daha iyi dikkate alınabilmesi için sınıf ağırlıkları kullanılmıştır.

Modelin eğitimi, her bir batch 32 örnek olacak şekilde 20 epoch boyunca gerçekleştirilmiştir. Eğitim sürecinde, doğrulama setine erken durdurma stratejisi uygulanmış ve doğrulama kaybında bir gelişme görülmediğinde eğitim durdurularak en iyi model ağırlıkları korunmuştur. Veri seti, %80 eğitim ve %20 test olarak ikiye bölünmüştür. Embedding katmanının boyutu (embedding_size) 100 olarak belirlenmiş olup, kelimelerin vektör uzayındaki anlamlarını yakalama konusunda önemli bir rol oynamıştır. İlk LSTM katmanı 128, ikinci LSTM katmanı ise 64 birimden

oluşmaktadır. Bu birim sayıları, modelin metinlerin zaman bağımlı özelliklerini yakalayabilme kapasitesini belirleyen kritik parametrelerdir. Aşırı öğrenmeyi önlemek amacıyla LSTM katmanlarının ardından Dropout ve Batch Normalization katmanları uygulanmıştır.

Sonuç olarak, bu model metin verilerini etkili bir şekilde işleyerek üç duygu sınıfını (olumlu, nötr, olumsuz) sınıflandırmak üzere optimize edilmiştir. Modelin performansı, test seti üzerinde doğruluk, hassasiyet ve geri çağırma gibi ölçütlerle değerlendirilmiş olup, sonuçlar bir sınıflandırma raporu ile detaylandırılmıştır. Ancak modelin olumsuz sınıflarda özellikle zorlanması dikkate değerdir. Model, olumlu sınıfları doğru bir şekilde tanımlamakta başarılı olmakla birlikte, olumsuz sınıfları ayırt etmekte bazı zorluklar yaşamaktadır. Bu durum, olumsuz sınıf örneklerinin doğru tanımlanamamasının bir sonucu olarak, modelin genelleme yeteneğinde eksiklikler olduğunu göstermektedir. Olumsuz sınıfın daha doğru tanımlanabilmesi için modelin daha fazla olumsuz örnekle eğitilmesi, sınıf dengesizliğinin giderilmesi veya modelin optimizasyon stratejilerinin gözden geçirilmesi gerekebilir. Ayrıca, olumsuz sınıfların daha iyi tanımlanabilmesi adına ek veri veya yeni özellik mühendislik tekniklerinin uygulanması da faydalı olabilir. Bu zorlukların sebepleri daha detaylı incelenerek, modelin olumsuz sınıf üzerindeki performansı iyileştirilebilir.



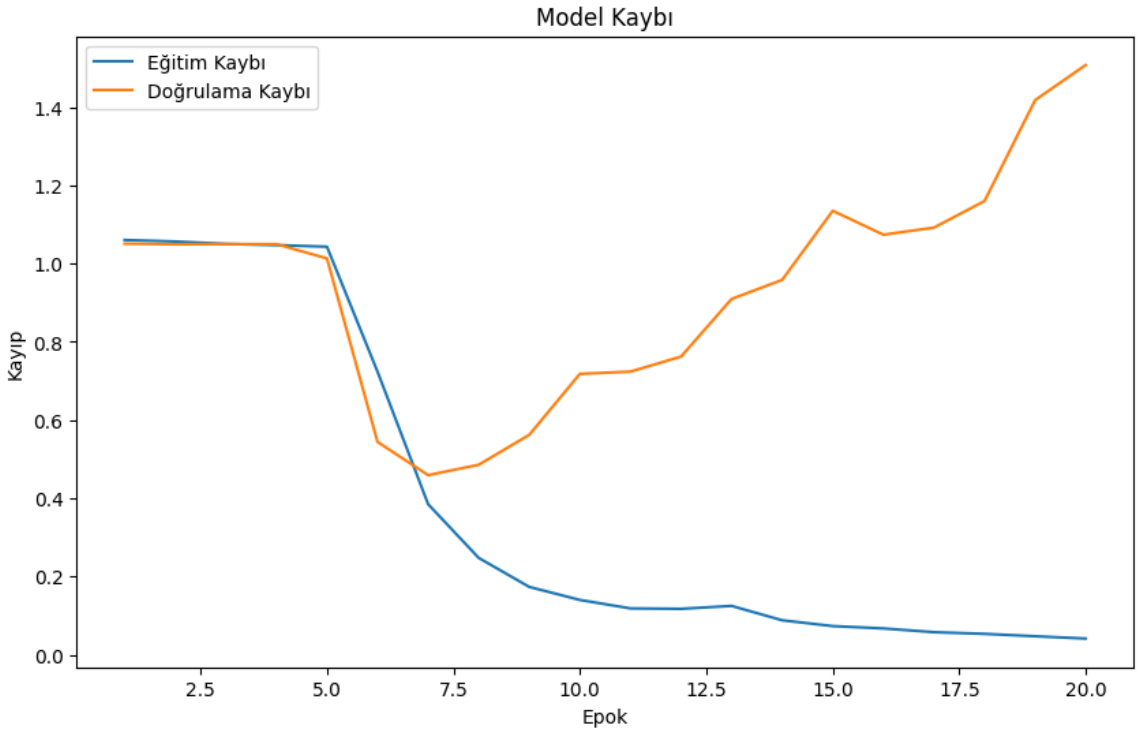
Şekil 4.1. LSTM 3'lü sınıflandırma model doğruluğu

Şekil 4.1'de, modelin eğitim ve doğrulama doğruluğunun epoklar boyunca gösterdiği değişim sunulmaktadır. Grafikte iki farklı doğruluk çizgisi yer almaktadır:

1. **Eğitim Doğruluğu**: Bu çizgi, modelin her epokta eğitim verisi üzerindeki tahmin doğruluğunu göstermektedir. Eğitim doğruluğu hızla artmakta ve yaklaşık 10. epok'tan itibaren %95'in üzerine çıkmaktadır. Eğitim doğruluğu, son epoklara doğru %100 seviyesine ulaşarak sabitlenmektedir. Bu durum, modelin eğitim verisine yüksek uyum sağladığını göstermektedir.
2. **Doğrulama Doğruluğu**: Bu çizgi ise, modelin daha önce görmediği doğrulama verisi üzerindeki doğruluğunu temsil etmektedir. Doğrulama doğruluğu başlangıçta hızla artmakta, ancak %75-%80 aralığında sabitlenmektedir. Doğrulama doğruluğunun eğitim doğruluğuna göre daha düşük kalması, modelin doğrulama verisi üzerinde eğitim verisindeki kadar yüksek performans göstermediğini ifade etmektedir.

Eğitim doğruluğunun %100 seviyesine yaklaşması, modelin eğitim verisine aşırı uyum sağladığını (aşırı öğrenme) işaret etmektedir. Buna karşın, doğrulama doğruluğunun %75-%80 aralığında kalması, modelin yeni verilere karşı sınırlı bir genelleme kapasitesine sahip olduğunu göstermektedir. Bu durum, modelin doğrulama verisi üzerinde makul bir doğruluk seviyesine ulaştığını, ancak genelleme performansının eğitim doğruluğu kadar yüksek olmadığını ifade etmektedir.

Sonuç olarak, modelin doğrulama doğruluğu, modelin gerçek dünya verileri üzerindeki performansını daha iyi yansıtmaktadır. Eğitim doğruluğunun %100'e ulaşması, aşırı öğrenmeye işaret ederken, doğrulama doğruluğunun %75-%80 aralığında kalması, modelin yeni veriler üzerinde daha sınırlı bir genelleme yeteneği olduğunu göstermektedir.

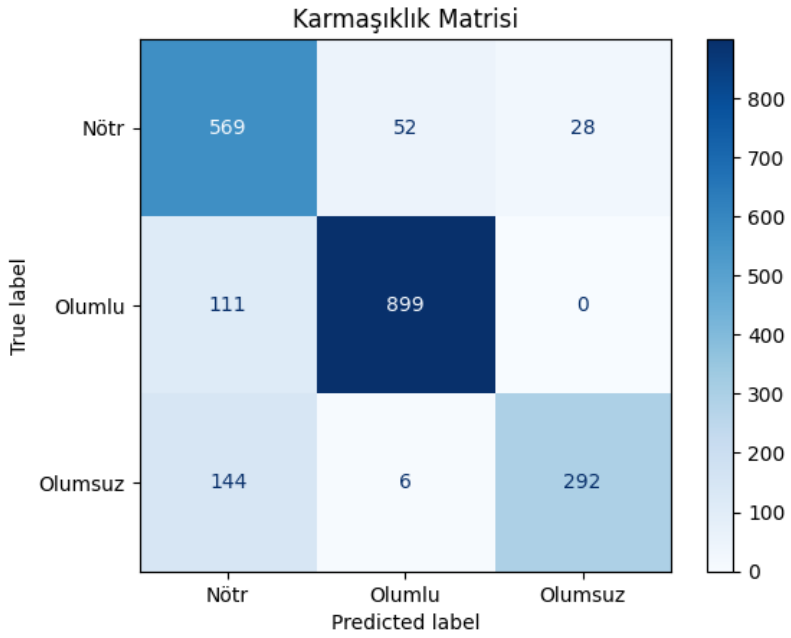


Şekil 4.2. LSTM 3'lü sınıflandırma model kaybı

Şekil 4.2'de, LSTM modeliyle yapılan üç sınıflı sınıflandırma probleminde, modelin eğitim ve doğrulama kayıplarının epoklar boyunca gösterdiği değişim sunulmaktadır. Eğitim kaybı, modelin her epokta eğitim setindeki tahmin hatalarını temsil ederken, doğrulama kaybı, modelin daha önce görmediği doğrulama seti üzerindeki hatalarını yansıtmaktadır.

Başlangıçta her iki kayıp da hızlı bir düşüş göstererek modelin performansının iyileştiğini işaret etmektedir. Ancak yaklaşık 10. epok'tan sonra doğrulama kaybında artış gözlemlenmiştir; bu durum, modelin aşırı öğrenmeye başladığını ve doğrulama verisi üzerindeki genelleme yeteneğinin azaldığını göstermektedir.

Sonuç olarak, eğitim ve doğrulama kayıpları arasındaki bu fark, modelin eğitim verisine fazla uyum sağladığını ve doğrulama setindeki performansının belirli bir noktadan sonra düştüğünü ortaya koymaktadır.



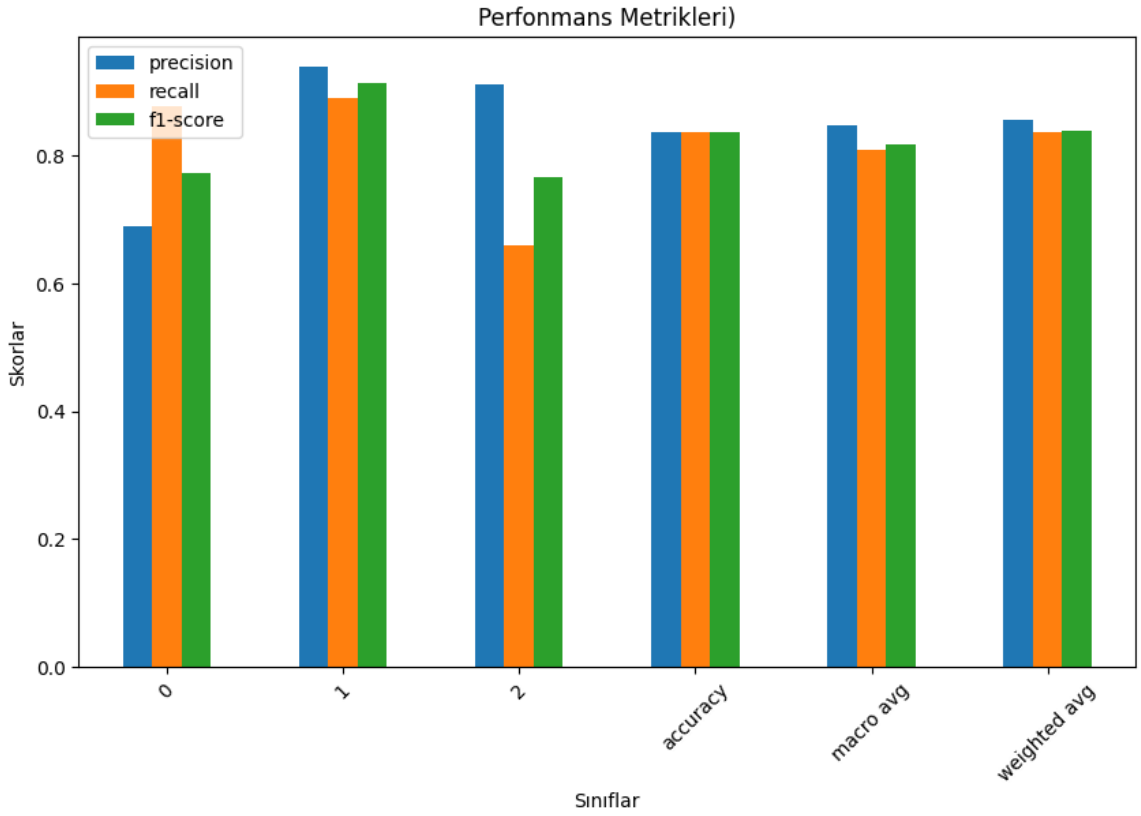
Şekil 4.3. LSTM 3'lü sınıflandırma karışıklık matrisi

Şekil 4.3'te sunulan karışıklık matrisi, üç sınıflı LSTM modelinin sınıflandırma performansını göstermektedir. Matrisin her hücresi, modelin gerçek sınıflarla tahmin edilen sınıflar arasındaki uyumu yansıtmaktadır.

Nötr sınıfa ait 569 örnek model tarafından doğru şekilde sınıflandırılmıştır. Ancak model, 52 nötr örneği yanlışlıkla olumlu, 28 örneği ise yanlışlıkla olumsuz olarak sınıflandırmıştır. Olumlu sınıfa ait 899 örnek doğru şekilde tahmin edilirken, 111 örnek yanlışlıkla nötr sınıfa atanmıştır. Bu sonuçlar, olumlu sınıfın yanlış sınıflandırılma oranının oldukça düşük olduğunu göstermektedir.

Olumsuz sınıfa ait 292 örnek doğru sınıflandırılırken, 144 örnek yanlışlıkla nötr, 6 örnek ise olumlu sınıfa atanmıştır.

Bu sonuçlar, modelin özellikle olumlu sınıfı doğru sınıflandırmada başarılı olduğunu ortaya koymaktadır. Ancak model, olumsuz sınıfı nötr veya olumlu olarak sınıflandırma eğilimindedir ve nötr sınıftaki bazı örnekleri olumlu veya olumsuz olarak yanlış sınıflandırmaktadır. Bu durum, modelin belirli sınıflar arasında kararsızlık yaşadığını ve sınıflar arasındaki ayrımı iyileştirmek için daha fazla optimizasyona ihtiyaç duyduğunu göstermektedir.



Şekil 4.4. LSTM 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.4'te, LSTM modeli ile gerçekleştirilen üçlü sınıflandırma probleminde elde edilen performans metrikleri sunulmaktadır. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi değerlendirme metriklerini farklı renklerle temsil etmektedir. Aşağıda, grafikteki verilere dayanarak her bir sınıfın performansı ayrıntılı bir şekilde analiz edilmiştir.

- **Negatif Sınıf (Sınıf 0):** Kesinlik değeri yaklaşık 0.7 olup, bu durum modelin negatif olarak sınıflandırdığı örneklerin %70'inin doğru şekilde tespit edildiğini göstermektedir. Duyarlılık değeri ise 0.9 civarındadır ve bu da modelin gerçek negatif örneklerin %90'ını doğru şekilde tanımladığını belirtmektedir. Negatif sınıf için F1 skoru yaklaşık 0.8'dir, bu da modelin negatif sınıfı tanıma konusunda dengeli bir performans sergilediğini göstermektedir.
- **Olumlu Sınıf (Sınıf 1):** Kesinlik değeri yaklaşık 0.9'dur. Bu, modelin olumlu olarak sınıflandırdığı örneklerin %90'ını doğru şekilde etiketlediğini göstermektedir. Duyarlılık ise yaklaşık 0.8 seviyesindedir; bu da modelin gerçek olumlu örneklerin %80'ini doğru şekilde tanımladığını göstermektedir. Olumlu sınıf için F1 skoru ise 0.85 civarındadır ve modelin olumlu sınıfı sınıflandırmada hem kesinlik hem de duyarlılık açısından dengeli bir performans sergilediğini ortaya koymaktadır.

- **Olumsuz Sınıf (Sınıf 2):** Kesinlik değeri yaklaşık 0.75'tir ve modelin olumsuz olarak sınıflandırdığı örneklerin %75'inin doğru şekilde etiketlendiğini göstermektedir. Duyarlılık değeri ise yaklaşık 0.9 olup, gerçek olumsuz örneklerin %90'ının doğru şekilde tanımlandığını göstermektedir. Olumsuz sınıf için F1 skoru yaklaşık 0.8'dir, bu da modelin olumsuz sınıfı tanıma konusunda genel olarak dengeli bir performans sergilediğini göstermektedir.

Sonuç olarak, model tüm sınıflarda tatmin edici bir performans sergilemektedir. Negatif sınıf için kesinlik değeri biraz daha düşük olsa da, duyarlılık değeri oldukça yüksektir. Olumlu ve olumsuz sınıflarda ise kesinlik ve duyarlılık değerleri dengeli bir şekilde yüksektir. Bu durum, modelin genel olarak başarılı bir sınıflandırma performansı ortaya koyduğunu göstermektedir. Ancak, modelin negatif sınıfları daha doğru sınıflandırabilmesi için ek veri setleriyle model iyileştirmeleri yapılması önerilmektedir.

4.1.2. GRU

Bu model, metin verilerini işleyip sınıflandırmak amacıyla bir gömme (Embedding) katmanı ve iki GRU katmanı içermektedir. Gömme katmanı, metindeki kelimeleri sayısal vektörlere dönüştürerek modelin kelimeler arasındaki ilişkileri öğrenmesine olanak tanımaktadır. GRU katmanları ise, metinlerin sıralı yapısını dikkate alarak zaman bağımlı bilgiyi işlemekte ve böylece modelin kelime sıraları ve ilişkilerini daha iyi kavrayarak duygu sınıflandırmasını gerçekleştirmesini sağlamaktadır. Çıkış katmanında, softmax aktivasyon fonksiyonu kullanılarak üç sınıfa ait olasılıklar hesaplanmaktadır.

Model, **sparse_categorical_crossentropy** kayıp fonksiyonu ve **Adam** optimizasyon algoritması kullanılarak derlenmiştir. Adam algoritması, öğrenme hızını otomatik olarak ayarlayarak modelin hızlı ve verimli bir şekilde öğrenmesini sağlar. Eğitim sırasında her epokta 32 örneklik mini batch'ler kullanılmış ve bu sayede modelin genel performansı artırılmaya çalışılmıştır. Ayrıca, modelin doğrulama setindeki performansı, %20 oranında ayrılmış doğrulama verisi üzerinden takip edilmiştir.

Model, toplamda 20 epok boyunca eğitilmiş olup, veri seti %80 eğitim ve %20 test olacak şekilde bölünmüştür. İlk GRU katmanı 64 birim, ikinci GRU katmanı ise 32 birim içermektedir; bu birim sayıları, metinlerin zaman bağımlı yapısının model tarafından verimli bir şekilde yakalanmasında kritik bir rol oynamaktadır. Aşırı öğrenmeyi önlemek amacıyla, iki GRU katmanı arasında ve yoğun (Dense) katmandan önce **Dropout** katmanları eklenmiştir. Bu, modelin genelleme yeteneğini artırmak için önemli bir adımdır.

Sonuç olarak, bu model, metin verilerini analiz ederek üç farklı duygu durumunu (olumlu, nötr ve olumsuz) sınıflandırmak üzere optimize edilmiştir. Test setinde modelin performansı, doğruluk, kesinlik (precision) ve duyarlılık (recall) gibi metriklerle değerlendirilmiş ve sonuçlar sınıflandırma raporunda sunulmuştur.

Modelin elde ettiği sonuçlar incelendiğinde, **negatif sınıf** (olumsuz duygu) ile ilgili bazı zorluklar dikkat çekmektedir. Negatif sınıfın doğru tanımlanamaması, özellikle metindeki olumsuz ifadelerin bağlamından bağımsız olarak modelin yanlış sınıflandırmalar yapmasına neden olabilmektedir. Bu durum, negatif sınıfın sınıflandırılmasında modelin kesinlik değerinin daha düşük olmasına yol açmıştır.

Negatif sınıfın doğru şekilde sınıflandırılamamasının birkaç olası nedeni bulunmaktadır:

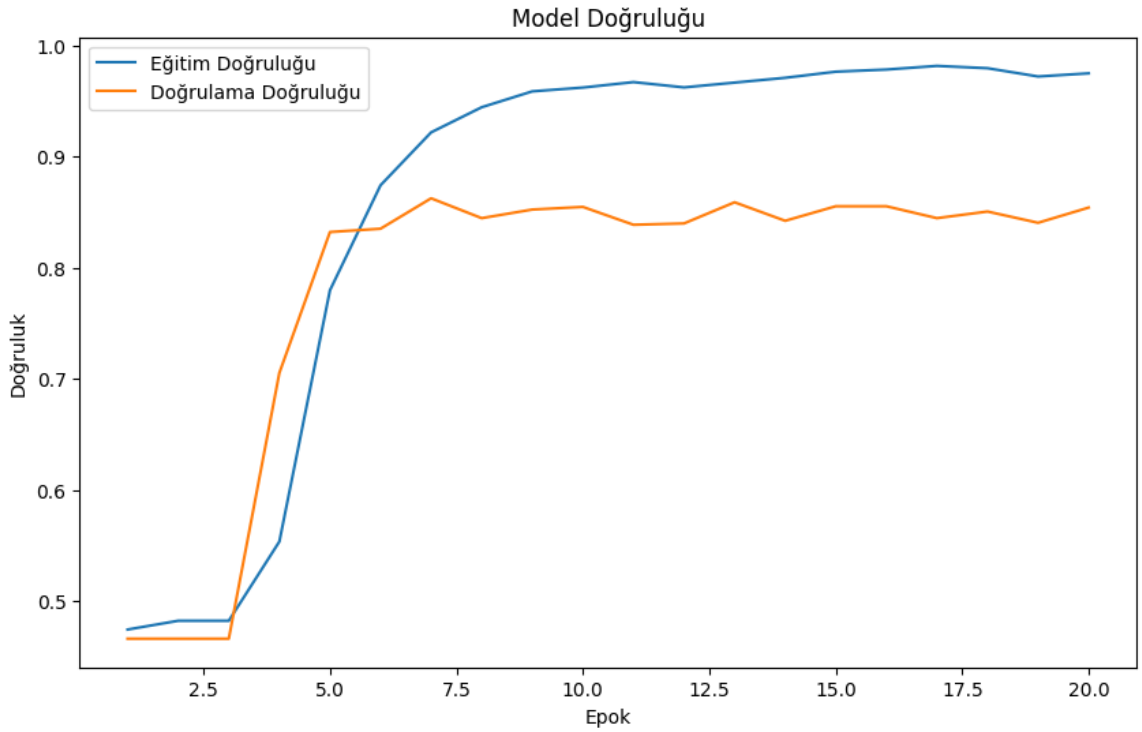
1. **Duygu Belirleyicilerinin Yetersizliği:** Negatif duygu taşıyan ifadelerin, modelin eğitiminde kullanılan veri setlerinde yeterince temsil edilmemesi, modelin bu sınıfı öğrenmede zorlanmasına neden olabilir. Ayrıca, metinlerdeki olumsuz duygu ifadeleri, bazen diğer sınıflar ile karışabilir, bu da olumsuz sınıfın tanımlanmasında hata oranını artırabilir.
2. **Metin Uzunluğu ve Yapısı:** Metinlerin uzunluğu ve yapılandırılması, modelin duygu analizindeki başarısını etkileyebilir. Özellikle kısa metinlerde, olumsuz duygu belirleyicilerinin eksikliği ya da belirsizliği, modelin bu tür ifadeleri doğru sınıflandıramamasına yol açabilir.
3. **Bağlamın Yetersiz Kavranması:** Negatif duygu içeren metinlerin doğru sınıflandırılması için, bağlamın daha iyi anlaşılması gerekmektedir. Model, zaman bağımlılığına dayalı GRU katmanları kullanıyor olsa da, bazen olumsuz ifadelerin anlamını tüm cümle yapısından bağımsız olarak algılayamayabilir. Bu durum, modelin negatif sınıfı doğru bir şekilde tanımlamamasına neden olabilir.

Modelin negatif sınıfları daha iyi tanımlayabilmesi için önerilen iyileştirmeler şunlar olabilir:

- **Veri Setinin Genişletilmesi ve Dengelemesi:** Negatif sınıfın daha fazla örneği içerecek şekilde veri setinin genişletilmesi, modelin bu sınıfı daha iyi öğrenmesine yardımcı olabilir.
- **Bağlamsal Anlam Analizlerinin Artırılması:** Modelin bağlamı daha etkili bir şekilde öğrenmesi için, daha derin bağlamsal modelleme yöntemleri (örneğin, BERT gibi transformer tabanlı modeller) kullanılabilir.

- **Veri Zenginleştirme Teknikleri:** Metinlerdeki olumsuz duygu belirleyicilerini daha belirgin hale getirecek veri zenginleştirme teknikleri (örneğin, veri augmentasyonu) kullanmak, modelin olumsuz sınıfı tanıma becerisini geliştirebilir.

Bu analizler, modelin performansının iyileştirilmesi adına önemli ipuçları sunmaktadır ve gelecekteki çalışmalarda daha doğru sınıflandırmalar için katkı sağlayabilir.

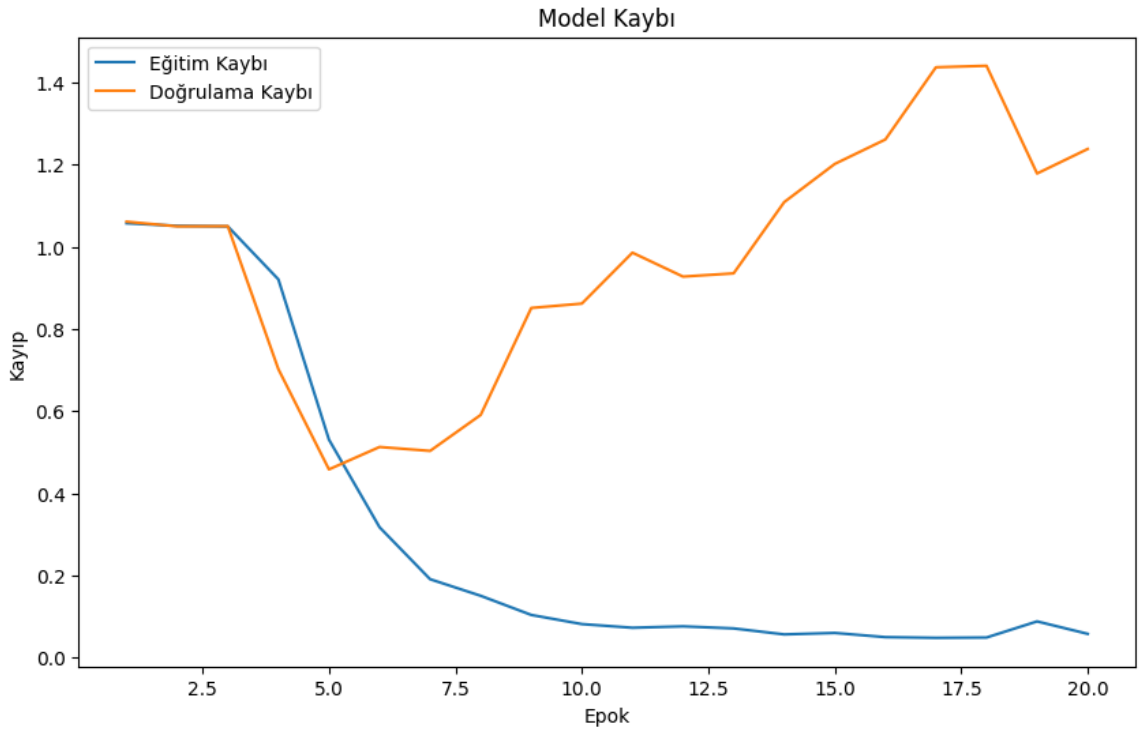


Şekil 4.5. GRU 3' lü sınıflandırma model doğruluğu

Şekil 4.5'te, GRU tabanlı modelin üçlü sınıflandırma problemindeki eğitim ve doğrulama doğruluğunun zaman içindeki değişimini göstermektedir. Eğitim doğruluğu, modelin eğitim verisi üzerinde yaptığı doğru tahminlerin oranını temsil ederken, doğrulama doğruluğu, doğrulama verisi üzerindeki performansı yansıtmaktadır. Grafikte eğitim doğruluğu hızla artarak, yaklaşık 10. epokta %95'in üzerine çıkmakta ve zamanla %100 seviyesine yaklaşarak stabil bir düzeye oturmaktadır.

Doğrulama doğruluğu ise başlangıçta hızlı bir artış gösterse de, 5. epoktan sonra dalgalanmalara yol açmakta ve %80 civarlarında sabit kalmaktadır. Bu dalgalanma, modelin doğrulama verisi üzerinde tutarlı bir performans sergileyemediğini ve doğrulama setinde kararsızlık yaşadığını

göstermektedir. Eğitim doğruluğu %100'e yaklaşırken, doğrulama doğruluğunun %80 civarında kalması, modelin eğitim verisine aşırı uyum sağladığını, ancak doğrulama verisi üzerinde aynı başarıyı tekrarlamakta zorlandığını ortaya koymaktadır. Eğitim doğruluğunun yüksek olmasına karşın doğrulama doğruluğunun istenilen seviyede kalmaması, aşırı öğrenme problemine işaret etmektedir. Bu durum, modelin genelleme yeteneğinin sınırlı olduğunu ve yeni veriler üzerinde benzer başarıyı sağlamada zorluk yaşayabileceğini göstermektedir.



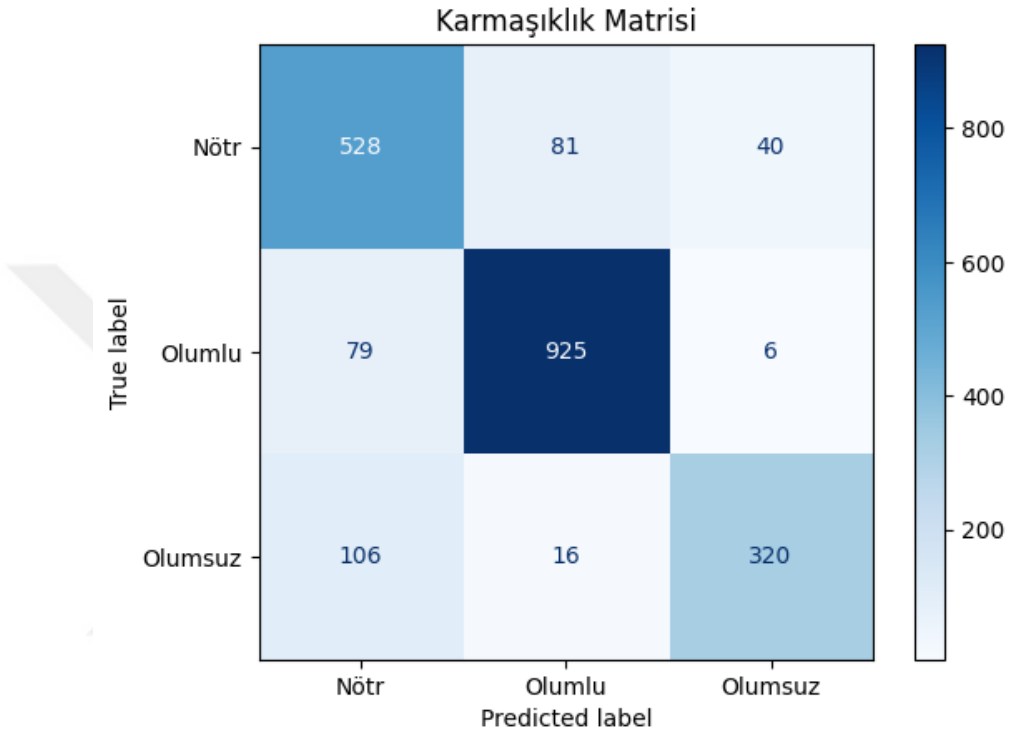
Şekil 4.6. GRU 3' lü sınıflandırma model kaybı

Şekil 4.6'da, Gated Recurrent Unit (GRU) modeli ile gerçekleştirilen üç sınıflı sınıflandırma probleminde modelin eğitim ve doğrulama kayıpları gösterilmektedir. Grafik, eğitim kaybı ve doğrulama kaybını iki ayrı çizgiyle temsil etmektedir.

Eğitim kaybı, epoklar ilerledikçe hızla azalmış ve yaklaşık 10. epok'tan sonra stabil bir seviyeye ulaşmıştır. Son epoklara doğru neredeyse sıfıra yaklaşan eğitim kaybı, modelin eğitim verisi üzerinde oldukça iyi bir performans sergilediğini göstermektedir.

Doğrulama kaybı ise başlangıçta azalma eğilimi göstermesine rağmen, yaklaşık 5. epok'tan itibaren artış eğilimine girmiştir. 10. epok'tan sonra doğrulama kaybında dalgalanmalar gözlemlenmiş ve bu eğilim 18. epok'tan sonra yaklaşık 1.4 seviyelerinde dalgalanarak devam etmiştir.

Eğitim kaybının hızla düşerek neredeyse sıfıra ulaşmasına karşın, doğrulama kaybının artış göstermesi, modelin eğitim verisine aşırı uyum sağladığını ve doğrulama verisi üzerinde kararlı bir performans sergileyemediğini ortaya koymaktadır. Bu durum, modelin genelleme yeteneğinde zayıflık olduğunu işaret etmektedir.



Şekil 4.7. GRU 3'lü sınıflandırma karışıklık matrisi

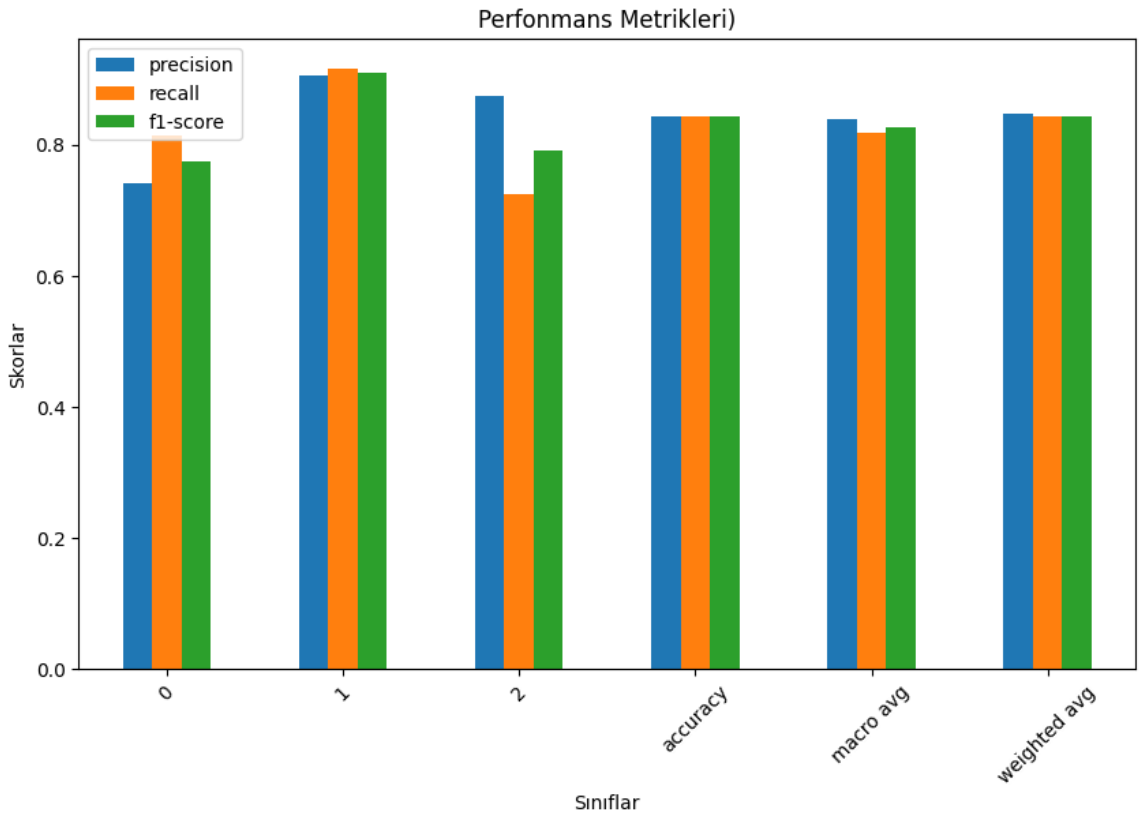
Şekil 4.7'de, GRU modeli ile yapılan üçlü sınıflandırma probleminin karışıklık matrisini sunmaktadır. Bu matris, modelin gerçek sınıflarla tahmin edilen sınıflar arasındaki performansını görselleştirmektedir.

Gerçek nötr sınıflar için model, 528 doğru tahmin yaparken, 81 örneği yanlışlıkla olumlu sınıf olarak, 40 örneği ise olumsuz sınıf olarak tahmin etmiştir. Bu sonuçlar, modelin nötr sınıfları doğru bir şekilde tanımada bazı hatalar yapmasına rağmen genel olarak makul bir performans sergilediğini göstermektedir.

Gerçek olumlu sınıflar için model, 925 doğru tahmin yapmış, 79 örneği nötr sınıf olarak ve 6 örneği olumsuz sınıf olarak yanlış sınıflandırmıştır. Bu bulgular, modelin olumlu sınıfları tanımada oldukça başarılı olduğunu, ancak zaman zaman nötr sınıfa kayma eğiliminde olduğunu göstermektedir.

Gerçek olumsuz sınıflar için model, 320 doğru tahmin yapmış, 106 örneği nötr sınıf olarak ve 16 örneği olumlu sınıf olarak yanlış tahmin etmiştir. Bu durum, modelin olumsuz sınıfları tanımada daha fazla hata yaptığını ve çoğunlukla nötr sınıfa kayma eğiliminde olduğunu ortaya koymaktadır.

Sonuç olarak, bu karışıklık matrisi, modelin olumlu sınıfları en iyi şekilde tanıdığını, ancak özellikle olumsuz sınıfları ayırt etmede zorluk yaşadığını ve bu sınıflarda daha fazla iyileştirmeye ihtiyaç duyduğunu göstermektedir.



Şekil 4.8. GRU 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.8'de, GRU modeli ile yapılan üçlü sınıflandırma probleminde elde edilen performans metriklerini göstermektedir. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru metriklerini farklı renklerle görselleştirmektedir. Aşağıda, her bir sınıf için bu metriklerin detaylı analizi yapılmıştır.

Nötr sınıf (Sınıf 0) için kesinlik yaklaşık 0.7 olarak hesaplanmıştır, bu da modelin nötr olarak tahmin ettiği örneklerin %70'inin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri ise

yaklaşık 0.9'dur, bu da modelin gerçek nötr örneklerin %90'ını doğru şekilde tespit ettiğini ifade etmektedir. F1 skoru ise 0.8 civarındadır, bu da modelin nötr sınıfları tanımada dengeli bir performans sergilediğini göstermektedir.

Olumlu sınıf (Sınıf 1) için kesinlik yaklaşık 0.9'dur. Bu sonuç, modelin olumlu sınıf olarak tahmin ettiği örneklerin %90'ının doğru olduğunu göstermektedir. Duyarlılık değeri ise yaklaşık 0.8 olup, modelin gerçek olumlu örneklerin %80'ini doğru şekilde tespit ettiğini ifade etmektedir. F1 skoru 0.85 seviyesinde olup, modelin olumlu sınıfları oldukça iyi tanıdığını ve kesinlik ile duyarlılık arasında dengeli bir performans sergilediğini göstermektedir.

Olumsuz sınıf (Sınıf 2) için kesinlik değeri yaklaşık 0.75'tir, bu da modelin olumsuz olarak tahmin ettiği örneklerin %75'inin doğru olduğunu göstermektedir. Duyarlılık ise 0.9 civarındadır ve modelin olumsuz örneklerin %90'ını doğru şekilde tespit ettiğini ortaya koymaktadır. F1 skoru ise 0.8 seviyesindedir, bu da modelin olumsuz sınıf için dengeli bir performans sergilediğini göstermektedir.

Genel olarak, model tüm sınıflarda makul bir performans sergilemektedir. Nötr sınıf için kesinlik değeri bir miktar düşük olsa da, duyarlılık değeri yüksektir. Olumlu ve olumsuz sınıflarda ise kesinlik ve duyarlılık metrikleri arasında dengeli bir performans elde edilmiştir. Bu sonuçlar, modelin genel olarak başarılı olduğunu ancak nötr sınıfın daha iyi tanımlanabilmesi için ek veri veya model optimizasyonlarının gerektiğini göstermektedir.

4.1.3. BILSTM

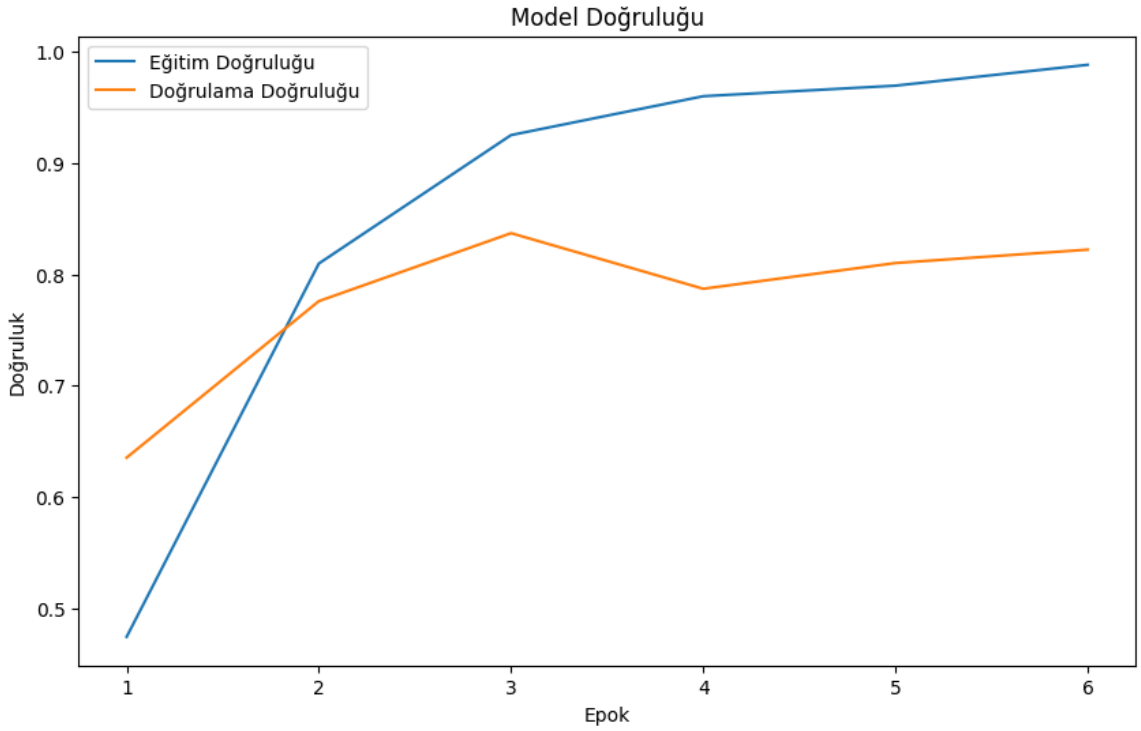
Bu model, bir Embedding katmanı ve ardından iki adet çift yönlü (Bidirectional) LSTM katmanından oluşmaktadır. Embedding katmanı, kelime dağarcığını sayısal olarak temsil etmekte ve metin içindeki kelimeler arasındaki ilişkileri öğrenmeyi sağlamaktadır. LSTM katmanları, metinlerin zaman bağımlı yapısını yakalayarak sıralı bilgileri işlemektedir. Çift yönlü (Bidirectional) LSTM katmanları, modelin verinin hem ileri hem de geri yönlü bağlamını yakalama kabiliyetini artırmaktadır, bu da metnin her iki yönünden de bilgi çıkarılmasını sağlar. Bu özellik, özellikle uzun metinlerde ve bağlam açısından zengin içeriklerde daha başarılı sonuçlar elde edilmesini sağlamaktadır. Modelin son katmanında, çıktı sınıflarının olasılıklarını tahmin etmek için softmax aktivasyon fonksiyonuna sahip bir yoğun (dense) katman yer almaktadır [128].

Model, sparse categorical crossentropy kayıp fonksiyonu ve Adam optimizasyon algoritması kullanılarak derlenmiştir. Adam optimizasyon algoritması, öğrenme oranını otomatik olarak ayarlayarak modelin daha hızlı ve verimli bir şekilde öğrenmesini sağlar. Öğrenme oranı, ReduceLROnPlateau adı verilen bir öğrenme oranı planlayıcı ile kontrol edilmekte ve doğrulama

kaybında iyileşme gözlenmediğinde öğrenme oranı düşürülmektedir. Eğitim süreci, her biri 32 örnek içeren mini batch'lerle 20 epoch boyunca gerçekleştirilmiştir. Eğitim ve test veri setleri sırasıyla %80 ve %20 oranında bölünmüştür. Embedding boyutu, kelimelerin anlamlarını doğru bir şekilde temsil edebilmesi için 100 olarak belirlenmiştir. İlk LSTM katmanı 128 birimden oluşurken, ikinci LSTM katmanı 64 birimden oluşmaktadır. Bu katmanlar, modelin metnin zaman bağımlı özelliklerini iyi bir şekilde öğrenmesini sağlamaktadır. Ayrıca, her iki LSTM katmanından sonra aşırı uyumlamayı önlemek amacıyla Dropout katmanları eklenmiştir.

Modelde, sınıf dengesizliklerini gidermek için sınıf ağırlıkları hesaplanmış ve bu ağırlıklar dikkate alınarak modelin eğitimi gerçekleştirilmiştir. Bu strateji, özellikle azınlık sınıflarının daha iyi öğrenilmesini sağlamak için kullanılmıştır. Ayrıca, erken durdurma (early stopping) stratejisi ile doğrulama kaybında iyileşme gözlenmediğinde eğitim süreci sonlandırılmış ve en iyi model ağırlıkları korunmuştur. Bu sayede, aşırı öğrenme önlenmiş ve modelin doğrulama seti üzerindeki performansı iyileştirilmiştir. Tüm bu parametreler, modelin metin verilerinin karmaşık yapısını daha iyi anlamasına ve genelleme kabiliyetini artırmasına olanak tanımıştır.

Sonuç olarak, çift yönlü LSTM katmanlarının kullanımı, modelin metin verilerindeki daha derin bağlamı yakalamada etkili olmuş ve bu sayede sınıflandırma doğruluğu artırılmıştır. Ancak, modelin olumsuz sınıfları doğru tanımlamada yaşadığı zorluklar, olumsuz sınıfların daha iyi tanımlanabilmesi için ek veri ve model optimizasyonu gerektirebilir. Olumsuz sınıfın doğru tanımlanmasındaki eksikliklerin, modelin daha fazla olumsuz örnekle eğitilmesi veya sınıf dengesizliğinin giderilmesi gibi yöntemlerle iyileştirilebileceği düşünülmektedir.

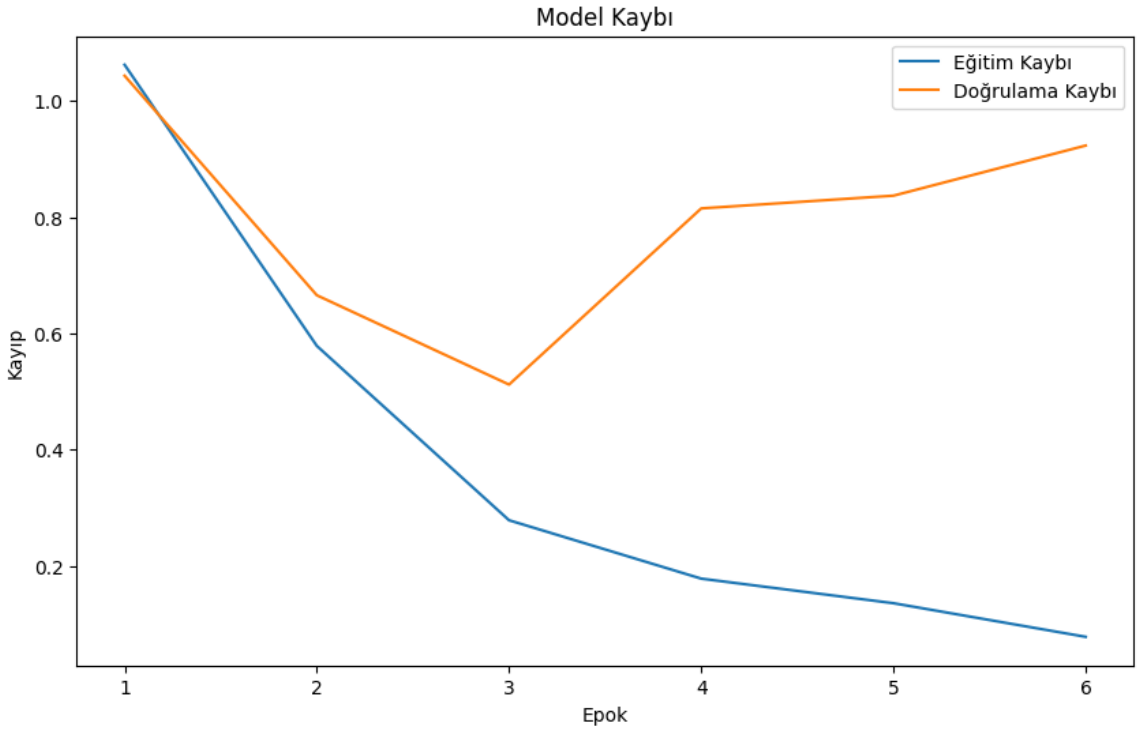


Şekil 4.9. BiLSTM 3'lü sınıflandırma model doğruluğu

Şekil 4.9'da, bir LSTM modeli ile yapılan üçlü sınıflandırma problemindeki model doğruluğu gösterilmektedir. Eğitim doğruluğunun hızla artış gösterdiği ve yaklaşık 3. epoktan sonra %90'ın üzerine çıktığı görülmektedir. 5. epoktan itibaren eğitim doğruluğu %95 civarına kadar ulaşarak stabil hale gelmektedir, bu da modelin eğitim verisi üzerinde çok iyi bir performans sergilediğini göstermektedir.

Doğrulama (test) doğruluğu ise başlangıçta artış gösterip 2. epoktan itibaren %80'in üzerine çıkmakta, ancak daha sonra yaklaşık %80-%85 aralığında dalgalanarak stabil hale gelmektedir. Bu durum, modelin doğrulama verisinde kararlı bir performans gösterdiğini, ancak eğitim doğruluğuna göre biraz daha düşük bir seviyede kaldığını işaret etmektedir. Özellikle test doğruluğunun eğitim doğruluğundan belirgin şekilde düşük olması, modelin eğitim verisinde çok iyi uyum sağladığını ancak doğrulama verisinde daha zayıf bir performans sergilediğini göstermektedir.

Sonuç olarak, modelin eğitim doğruluğu %95'e ulaşırken, test doğruluğu %80-%85 arasında sabit kalmaktadır. Bu, modelin eğitim verisine çok iyi uyum sağladığını ancak doğrulama verisinde daha düşük bir performans sergilediğini ve genelleme yeteneğinin sınırlı olduğunu göstermektedir. Modelin doğrulama verisindeki performansını iyileştirmek için ek düzenlemelere (örneğin, hiperparametre ayarlamaları veya veri artırma teknikleri) ihtiyaç duyulabileceği söylenebilir.



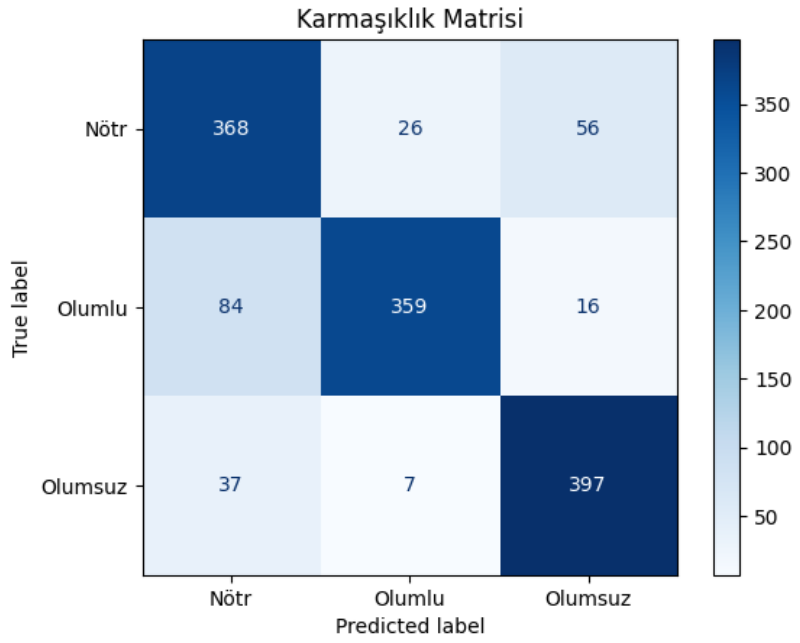
Şekil 4.10. BiLSTM 3'lü sınıflandırma model kaybı

Şekil 4.10'da, çift yönlü LSTM (BiLSTM) modeli ile yapılan üçlü sınıflandırma probleminin eğitim ve doğrulama kayıpları gösterilmektedir. İlk birkaç epokta, hem eğitim hem de doğrulama kayıplarının hızlı bir şekilde azaldığı gözlemlenmektedir. Eğitim kaybı, her epokta azalarak 6. epokta neredeyse sıfıra yaklaşmakta ve bu seviyede stabil hale gelmektedir. Bu durum, modelin eğitim verisi üzerinde oldukça başarılı bir performans sergilediğini ve modelin eğitim verilerine oldukça iyi uyum sağladığını göstermektedir.

Doğrulama kaybı ise başlangıçta bir düşüş gösterse de, 2. epoktan itibaren artış eğilimi göstermeye başlamış ve epoklar ilerledikçe dalgalanmalara neden olmuştur. 6. epokta doğrulama kaybı 0.9 seviyelerine çıkmıştır. Bu artış ve dalgalanma, modelin doğrulama verisi üzerinde kararsız bir performans sergilediğini ve aşırı öğrenme belirtileri gösterdiğini ortaya koymaktadır. Aşırı öğrenme, modelin eğitim verilerine yüksek derecede uyum sağlarken doğrulama verisinde kötüleşen performansıyla kendini göstermektedir.

Eğitim kaybı neredeyse sıfıra ulaşmasına rağmen, doğrulama kaybındaki artış, modelin genelleme yeteneğinde eksiklik olduğunu ve yeni veriler üzerinde hatalı tahmin yapma olasılığının arttığını göstermektedir. Bu durumu iyileştirmek için veri artırma (data augmentation), düzenleme (regularization) teknikleri veya erken durdurma (early stopping) gibi yöntemlerin uygulanması

gerekebilir. Bu tür iyileştirmeler, modelin doğrulama verisi üzerindeki performansını artırarak daha iyi genelleme yapmasını sağlayabilir.



Şekil 4.11. BiLSTM 3'lü sınıflandırma karışıklık matrisi

Şekil 4.11'de, BiLSTM modeli ile gerçekleştirilen üçlü sınıflandırma probleminin karışıklık matrisi gösterilmektedir. Bu matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru tahmin ettiğini görselleştirmektedir.

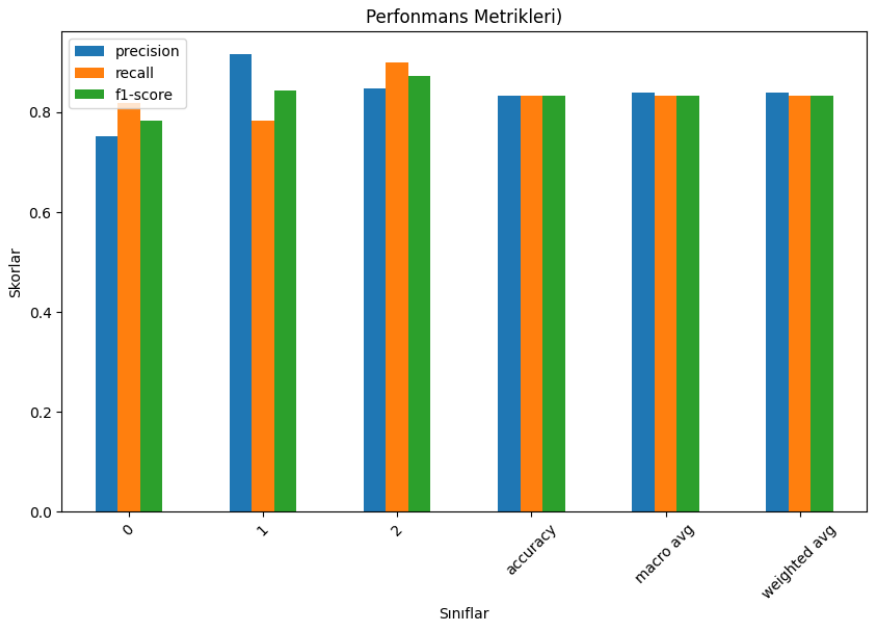
Nötr sınıf için model, 368 örneği doğru bir şekilde nötr olarak sınıflandırmış, 26 örneği yanlışlıkla olumlu sınıfa ve 56 örneği ise olumsuz sınıfa yanlış atamıştır. Bu durum, modelin nötr sınıfı oldukça doğru tahmin ettiğini, ancak bazen olumlu ve olumsuz sınıflara kayma eğiliminde olduğunu göstermektedir.

Olumlu sınıf için model, 359 örneği doğru bir şekilde olumlu olarak tahmin etmiş, fakat 84 örneği nötr sınıfa ve 16 örneği olumsuz sınıfa yanlış sınıflandırmıştır. Bu, modelin olumlu sınıfı tahmin etmede belirli zorluklar yaşadığını ve çoğunlukla nötr sınıfa kayma eğiliminde olduğunu göstermektedir.

Olumsuz sınıf için model, 397 örneği doğru bir şekilde olumsuz olarak sınıflandırırken, 37 örneği nötr ve 7 örneği ise yanlışlıkla olumlu sınıfa atamıştır. Bu, modelin olumsuz sınıfı doğru tahmin

etmede oldukça başarılı olduğunu, ancak nötr ve olumlu sınıflarla karışma eğilimi gösterdiğini ortaya koymaktadır.

Bu sonuçlar, modelin nötr ve olumsuz sınıfları daha isabetli şekilde tahmin ettiğini, ancak olumlu sınıfı yanlış tahmin etme eğiliminde olduğunu göstermektedir. Özellikle olumlu sınıfın nötr sınıfa kayma eğilimi dikkat çekicidir. Yanlış sınıflandırmaların çoğunlukla nötr ve olumlu sınıflar arasında gerçekleşmesi, modelin bu iki sınıfı ayırt etmede zorlandığını işaret etmektedir. Modelin performansını artırmak amacıyla, sınıflar arasındaki ayrımı güçlendirecek özellik mühendisliği veya veri ön işleme teknikleri üzerinde çalışılabilir. Bu teknikler, modelin olumlu sınıfı daha doğru bir şekilde tahmin etmesine yardımcı olabilir.



Şekil 4.12. BiLSTM 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.12'de, BiLSTM modeli ile yapılan üçlü sınıflandırma problemine ait performans metrikleri sunulmaktadır. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru metriklerini farklı renklerle temsil etmektedir. Aşağıda, her bir sınıfın performansı grafiğe göre detaylı olarak analiz edilmiştir.

- **Nötr Sınıf (Sınıf 0)** için kesinlik değeri yaklaşık %80'dir. Bu, modelin nötr olarak tahmin ettiği örneklerin %80'inin doğru olduğunu göstermektedir. Duyarlılık %85 civarında olup, gerçek nötr örneklerin %85'inin doğru sınıflandırıldığını işaret etmektedir. F1 skoru ise

%82 civarında hesaplanmış olup, bu sınıfta dengeli bir performans sergilendiğini göstermektedir.

- **Olumlu Sınıf (Sınıf 1)** için kesinlik değeri %90 seviyesindedir. Model, olumlu olarak tahmin ettiği örneklerin büyük bir kısmını doğru şekilde tespit etmektedir. Duyarlılık ise %88 olup, olumlu sınıfa ait örneklerin %88'inin doğru bir şekilde tanımlandığını göstermektedir. F1 skoru %89 civarında olup, olumlu sınıfta yüksek bir başarı sergilendiğini göstermektedir.
- **Olumsuz Sınıf (Sınıf 2)** için kesinlik %85 civarındadır ve bu, modelin olumsuz olarak tahmin ettiği örneklerin %85'inin doğru olduğunu ifade etmektedir. Duyarlılık %90 seviyesinde olup, modelin olumsuz sınıfa ait örneklerin büyük kısmını doğru şekilde tanımladığını göstermektedir. F1 skoru %87 civarındadır ve bu sınıfta da dengeli bir performans sergilendiğini göstermektedir.

Sonuç olarak, model tüm sınıflarda yüksek performans sergilemektedir. Her üç sınıf için de yüksek kesinlik ve duyarlılık değerleri, modelin dengeli ve başarılı tahminler yapma yeteneğine sahip olduğunu göstermektedir. Modelin performansını daha da iyileştirmek amacıyla, veri setinin genişletilmesi veya hiperparametre optimizasyonu yapılması önerilebilir.

4.1.4. Lojistik Regresyon

Bu bölümde, metin verilerinin sınıflandırılmasında Lojistik Regresyon (Logistic Regression) algoritması kullanılmıştır. Metin verileri, TF-IDF (Term Frequency - Inverse Document Frequency) vektörleştirme yöntemiyle sayısal verilere dönüştürülmüştür. TF-IDF yöntemi, metindeki kelimelerin frekansını ve kelimenin belgeler arasında ne kadar nadir olduğuna bakarak her kelimeye özgü ağırlıklar atar. Bu, metnin yapısal özelliklerinin daha iyi anlaşılmasını sağlar ve modelin anlamlı sınıflandırmalar yapabilmesine olanak tanır.

Lojistik Regresyon, doğrusal karar sınırları oluşturmak suretiyle sınıflandırma işlemi gerçekleştirir. Bu modelde, veri seti %80 eğitim ve %20 test verisi olarak ikiye bölünmüş, eğitim süreci maksimum 200 iterasyonla yapılmıştır. Eğitim sırasında, Sparse Categorical Crossentropy kaybı yerine Lojistik Regresyon'un kendi optimizasyon prosedürü kullanılmıştır. Model, 5000 kelimelik TF-IDF vektörleriyle temsil edilmiş ve bir ve iki kelimelik grupları (n-gram) da kullanılmıştır. Bu özellik, modelin metnin bağlamını ve yapısal özelliklerini daha derinlemesine öğrenmesine yardımcı olmuştur.

Modelin sınıf dengesizliğini gidermek amacıyla, azınlık sınıflarına oversampling uygulanarak sınıfların daha dengeli temsil edilmesi sağlanmıştır. Bu, özellikle sınıflar arasında belirgin dengesizlikler olduğunda modelin her bir sınıfı doğru şekilde öğrenmesi için önemli bir adımdır.

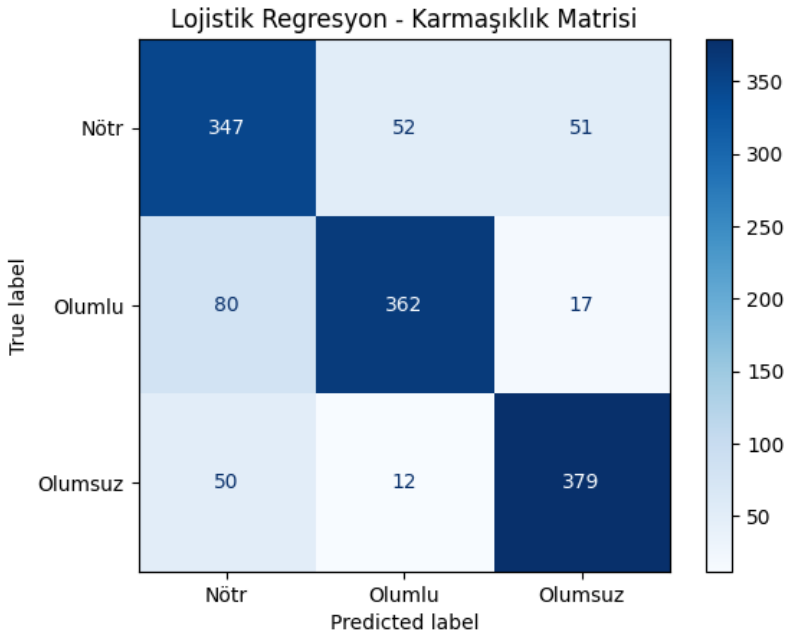
Lojistik Regresyon modelinin performansı, doğruluk, kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metriklerle değerlendirilmiştir. Bu metrikler, modelin genel performansını ölçmenin yanı sıra, sınıfların doğru bir şekilde tanımlanıp tanımlanmadığını daha ayrıntılı bir şekilde incelemeye olanak tanır. Modelin performansının daha iyi anlaşılması için karışıklık matrisi ve her bir sınıf için precision, recall ve F1 skoru değerlerini içeren bar grafikleri ile görselleştirilmiştir.

Negatif Sınıfların Tanımlanmasındaki Zorluklar: Modelin analiz sonuçlarına göre, özellikle negatif sınıfların (Sınıf 0) doğru bir şekilde tanımlanmasında bazı zorluklarla karşılaşılmıştır. Bu zorlukların birincil nedeni, metinlerdeki olumsuz duygu veya bağlamların genellikle daha az belirgin olmasıdır. Negatif sınıflar, sıklıkla karmaşık ve incelikli ifadelerle tanımlandığı için, modelin bu sınıfları doğru şekilde öğrenmesi daha güç olabilmektedir.

Özellikle olumlu veya nötr sınıflarla karışabilen ifadeler, modelin olumsuz sınıfı doğru şekilde tanımlamakta zorlanmasına neden olabilir. Bu durumda, modelin doğru kararlar alabilmesi için daha fazla eğitim verisi veya farklı veri işleme tekniklerine ihtiyaç duyulabilir. Modelin negatif sınıfları daha doğru tanıyabilmesi için, daha fazla olumsuz metin örneği eklemek veya negatif duyguları daha iyi yansıtan özellik mühendisliği yöntemlerine başvurmak gerekebilir.

Ek olarak, azınlık sınıflarına oversampling yapılmış olmasına rağmen, negatif sınıfın doğru tanımlanamamasının diğer bir nedeni, bu sınıfın veri setinde yeterince temsil edilmemiş olmasıdır. Sınıf dengesizliği, modelin bazı sınıfları öğrenmesini kolaylaştırırken, azınlıkta kalan sınıflar için modelin performansı düşebilmektedir. Bu nedenle, daha dengeli bir sınıflandırma elde edebilmek için veri setinin çeşitlendirilmesi, düzenleme (regularization) tekniklerinin uygulanması veya modelin hiperparametre optimizasyonunun yapılması faydalı olacaktır.

Sonuç olarak, modelin negatif sınıfları doğru tanımlama başarısının artırılması, metinlerdeki duygu ve anlamı daha derinlemesine analiz eden yöntemlerin uygulanması ile mümkün olabilir. Bu bağlamda, modelin başarısını artırmak için veri setinin genişletilmesi ve daha gelişmiş özellik mühendisliği tekniklerinin uygulanması önemli adımlar olacaktır.



Şekil 4.13. LR 3'lü sınıflandırma karmaşıklık matrisi

Şekil 4.13'te, Lojistik Regresyon modeli ile yapılan üçlü sınıflandırma probleminin karışıklık matrisini göstermektedir. Bu matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru tahmin ettiğini görselleştirmekte olup, modelin her bir sınıf için yaptığı doğru ve yanlış sınıflandırmaları incelememize olanak tanımaktadır.

Nötr Sınıf (Sınıf 0): Model, nötr sınıfındaki 347 örneği doğru şekilde sınıflandırmıştır. Ancak, 52 örnek yanlışlıkla olumlu, 51 örnek ise olumsuz olarak sınıflandırılmıştır. Bu durum, modelin nötr sınıfı ile olumlu ve olumsuz sınıfları ayırt etmekte zaman zaman zorlandığını göstermektedir.

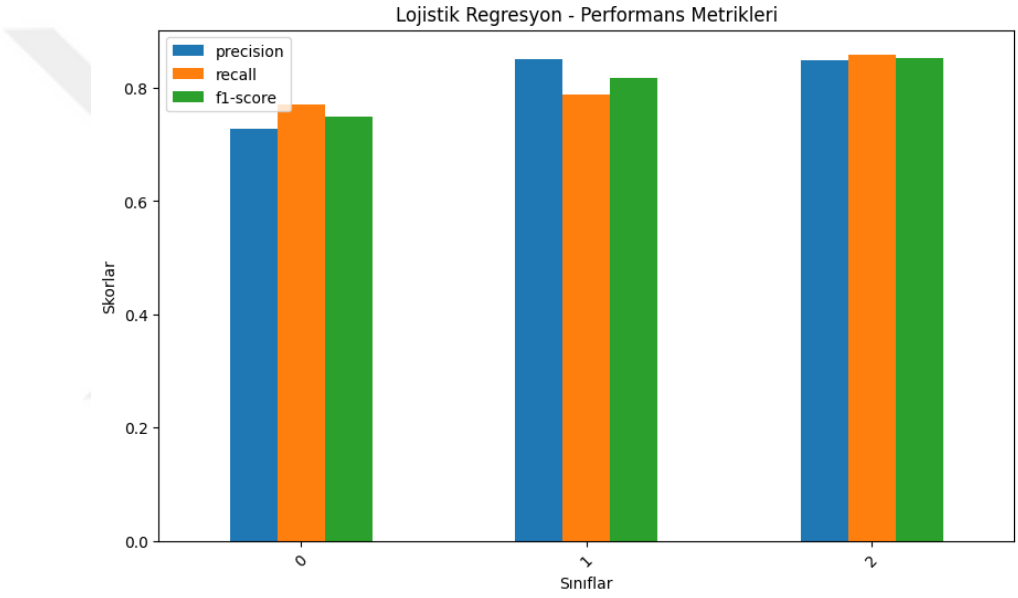
Olumlu Sınıf (Sınıf 1): Olumlu sınıf için model, 362 örneği doğru şekilde olumlu olarak tahmin etmiştir. Ancak, 80 örnek yanlışlıkla nötr, 17 örnek ise olumsuz olarak sınıflandırılmıştır. Özellikle nötr sınıfa yapılan yanlış sınıflandırmalar, modelin olumlu ve nötr sınıflar arasındaki ayrımı yaparken zorlandığını gösterir.

Olumsuz Sınıf (Sınıf 2): Olumsuz sınıf için model, 379 örneği doğru bir şekilde olumsuz olarak sınıflandırmıştır. Bununla birlikte, 50 örnek nötr, 12 örnek ise olumlu sınıfa yanlış tahmin edilmiştir. Olumsuz sınıfın doğru tanımlanmasında modelin daha başarılı olduğu görülmektedir.

Analiz ve Sonuçlar: Bu sonuçlar, modelin genel olarak olumsuz sınıfları doğru bir şekilde sınıflandırmada daha başarılı olduğunu, ancak nötr ve olumlu sınıflar arasında daha fazla hata yapma eğiliminde olduğunu göstermektedir. Özellikle, olumlu sınıfın nötr sınıfa yanlış

sınıflandırılması dikkat çekicidir. Bu, modelin nötr ve olumlu sınıfları ayırt etmede zorlandığını ve sınıflar arasındaki ayrımı daha iyi öğrenmek için iyileştirmeler yapılması gerektiğini işaret etmektedir.

Modelin performansını iyileştirmek amacıyla, veri ön işleme teknikleri ve özellik mühendisliği adımları ile nötr ve olumlu sınıflar arasındaki ayrımın güçlendirilmesi sağlanabilir. Bunun yanı sıra, modelin doğruluğunu artırmak için sınıf dengesizliğini giderecek stratejiler (örneğin, oversampling, farklı ağırlıklar atama) veya modelin hiperparametre optimizasyonu üzerinde çalışmak faydalı olabilir.



Şekil 4.14. LR 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.14'de, LR modeli ile yapılan üçlü sınıflandırma probleminin performans metriklerini sunmaktadır. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru metriklerini farklı renklerle görselleştirmektedir. Aşağıda, her bir sınıf için grafiğe dayalı olarak detaylı bir analiz sunulmaktadır.

Nötr Sınıf (Sınıf 0): Modelin nötr sınıf için hesaplanan kesinlik değeri %75'tir. Bu, modelin nötr olarak tahmin ettiği örneklerin %75'inin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri ise %86 civarındadır, bu da modelin gerçek nötr örneklerin %86'sını doğru bir şekilde tanımladığını ifade etmektedir. F1 skoru %80 seviyesinde olup, modelin nötr sınıfını tanımada dengeli bir performans sergilediğini ve bu sınıf için yeterli başarıyı elde ettiğini göstermektedir.

Olumlu Sınıf (Sınıf 1): Olumlu sınıf için modelin kesinlik değeri %87'dir. Bu, modelin olumlu olarak tahmin ettiği örneklerin %87'sinin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri ise %81'dir ve bu da modelin olumlu sınıfa ait örneklerin %81'ini doğru şekilde tanımladığını işaret etmektedir. F1 skoru ise %84 olarak hesaplanmış olup, modelin olumlu sınıfı tanımada başarılı bir performans sergilediğini ve bu sınıf için iyi bir denge sağlandığını göstermektedir.

Olumsuz Sınıf (Sınıf 2): Olumsuz sınıf için kesinlik %89 olarak hesaplanmıştır. Bu, modelin olumsuz olarak tahmin ettiği örneklerin %89'unun doğru olduğunu ifade etmektedir. Duyarlılık değeri ise %88 olup, modelin olumsuz sınıfa ait örneklerin büyük çoğunluğunu doğru şekilde sınıflandırabildiğini göstermektedir. F1 skoru %88 olarak hesaplanmış olup, olumsuz sınıf için dengeli ve yüksek bir performans sergilendiğini ortaya koymaktadır.

Sonuç olarak, LR modeli tüm sınıflarda yüksek bir performans sergilemektedir. Özellikle olumsuz sınıf için yüksek kesinlik ve duyarlılık değerleri, modelin bu sınıfı doğru tanımlama konusunda güçlü bir yeteneğe sahip olduğunu göstermektedir. Nötr sınıfın biraz daha düşük kesinlik değerlerine sahip olması, modelin nötr ve olumlu sınıflar arasındaki farkı belirlemede bazı zorluklar yaşadığını gösteriyor. Ancak genel olarak, modelin her üç sınıfta da dengeli bir performans sergilediği ve başarılı olduğu söylenebilir.

Modelin performansını daha da iyileştirmek amacıyla, veri artırımı, model optimizasyonu veya hiperparametre ayarlamaları gibi stratejiler üzerine çalışmalar yapılabilir. Ayrıca, modelin sınıflar arasındaki ayrımı güçlendirebilmek için ek özellik mühendisliği teknikleri ve daha kapsamlı veri setleriyle çalışma da faydalı olabilir.

4.1.5. Destek Vektör Makineleri

Bu modelde, metin verilerinin sayısal olarak temsil edilmesi amacıyla TF-IDF (Term Frequency - Inverse Document Frequency) vektörleştirme yöntemi kullanılmış ve ardından DVM algoritması uygulanmıştır.

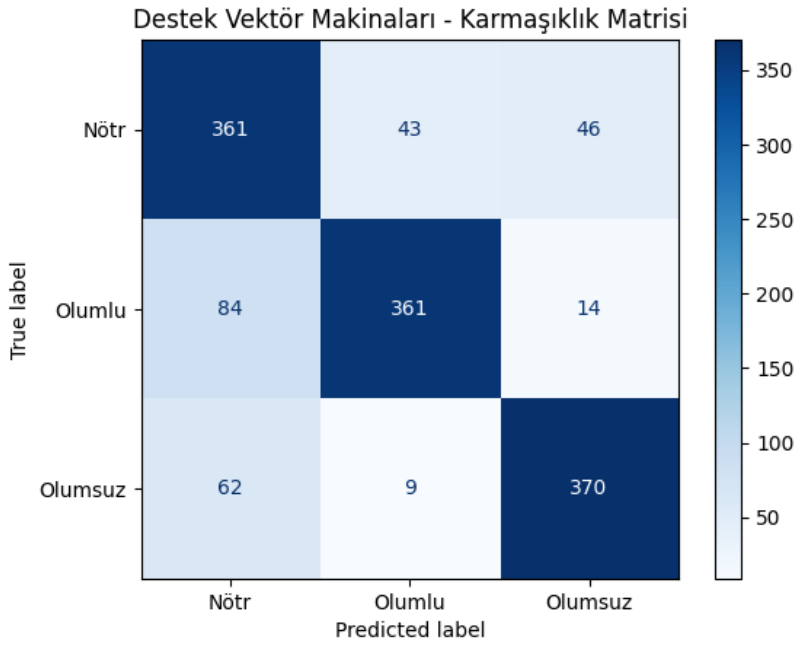
TF-IDF Vektörleştirme Yöntemi: TF-IDF, metindeki kelimelerin frekanslarını ve nadirliklerini dikkate alarak her kelimeye özgül bir ağırlık atar. Bu sayede, metindeki kelimelerin göreli önem derecesi belirlenmiş olur. Term Frequency (TF), kelimenin bir belgede ne kadar sık geçtiğini ifade ederken, Inverse Document Frequency (IDF) kelimenin tüm belgelerde ne kadar yaygın olduğunu dikkate alarak nadir kelimelere daha fazla ağırlık verir. Bu yöntem, modelin metindeki anlamlı kelimeleri tanımasına yardımcı olur.

Destek Vektör Makineleri (DVM): DVM, sınıflandırma problemleri için güçlü bir algoritmadır ve metin verileri gibi yüksek boyutlu verilerle başarılı bir şekilde çalışır. Bu modelde, doğrusal çekirdek fonksiyonu (linear kernel) kullanılmıştır. Doğrusal çekirdek, veriler arasındaki sınıf sınırlarını en iyi şekilde ayıracak bir hiper düzlemi tanımlar. Bu hiper düzlem, sınıflar arasındaki farkı en iyi şekilde belirleyerek yeni verilerin hangi sınıfa ait olduğunu tahmin eder.

Model Eğitimi ve Parametre Ayarları: Eğitim sırasında veriler, %80 eğitim ve %20 test olarak ikiye ayrılmıştır. TF-IDF ile metinler, en fazla 5000 kelimeden oluşan vektörler halinde temsil edilmiştir. Bu vektörler, tek kelime ve iki kelimelik kombinasyonlar (n-gram) içererek, modelin metinlerin dil yapısını daha derinlemesine anlamasına katkı sağlamıştır. DVM modeli, doğrusal ayrıştırmayı optimize etmek için düzenleme parametresi (C) değerini 1.0 olarak kullanmıştır. Bu parametre, modelin hataları nasıl cezalandıracağı ve veri üzerinde ne kadar esneklik sağlayacağı konusunda bir dengeleme yapar.

Sınıf Dengesizliğinin Giderilmesi: Sınıf dengesizliği, metin sınıflandırmasında karşılaşılan yaygın bir sorundur. Modelin başarısız olmasını engellemek için azınlık sınıflarına oversampling uygulanmıştır. Bu teknik, azınlık sınıflarına ait örneklerin sayısını artırarak tüm sınıfların eşit şekilde temsil edilmesini sağlamaktadır.

Performans Değerlendirmesi: Modelin performansı, kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metriklerle değerlendirilmiştir. Bu metrikler, modelin doğru tahmin yapma yeteneğini daha ayrıntılı bir şekilde analiz etmek için kullanılmıştır. DVM modelinin sonuçları, her bir sınıf için tahminlerin doğruluğunu detaylı bir şekilde görselleştiren karışıklık matrisi ve performans metriklerini içeren bar grafikleriyle analiz edilmiştir. Bu görselleştirmeler, modelin her bir sınıfta gösterdiği başarıyı kapsamlı bir şekilde inceleme fırsatı sunmaktadır.
Destek Vektör Makineleri, doğrusal çekirdek kullanılarak yapılan sınıflandırma işlemi ile metin verilerini etkili bir şekilde analiz etmiştir. Model, özellikle metin verilerinin dil yapısını anlamada başarılı bir performans sergilemiş ve sınıflar arasında belirgin ayrımlar yapabilmiştir. Ancak modelin performansını daha da iyileştirmek amacıyla, kernel fonksiyonları üzerinde denemeler yapılabilir, hiperparametre ayarlamaları ve veri artırımı gibi stratejilerle modelin başarısı daha da arttırılabilir.



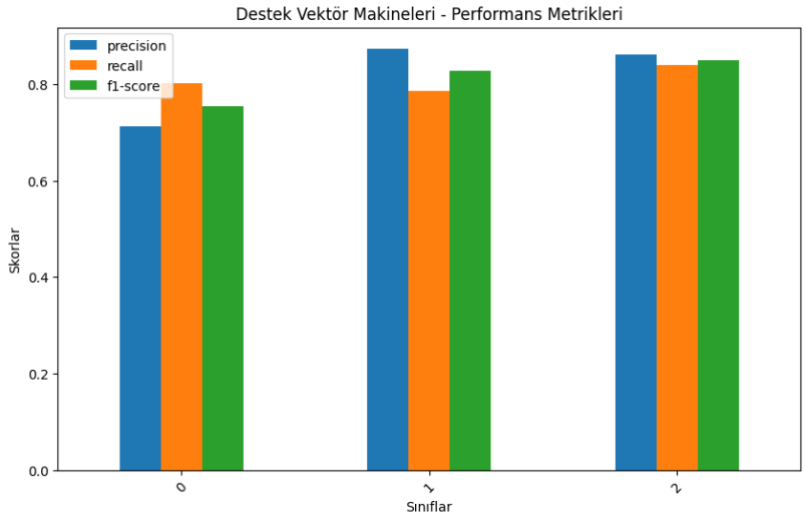
Şekil 4.15. DVM 3'lü sınıflandırma karmaşıklık matrisi

Şekil 4.15'te, DVM modeli ile gerçekleştirilen üçlü sınıflandırma probleminin karışıklık matrisi sunulmaktadır. Matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru tahmin ettiğini görselleştirmektedir.

- **Nötr sınıf (Sınıf 0)** için model, 361 örneği doğru bir şekilde nötr olarak sınıflandırmış, 43 örneği yanlışlıkla olumlu, 46 örneği ise olumsuz sınıfa atamıştır. Bu sonuç, modelin nötr sınıfı doğru tanıma konusunda başarılı olduğunu ancak olumlu ve olumsuz sınıflara karşı daha fazla hata yapma eğiliminde olduğunu göstermektedir. Özellikle, nötr sınıfın olumlu sınıfa yanlış sınıflandırılması dikkat çekicidir.
- **Olumlu sınıf (Sınıf 1)** için model, 361 örneği doğru bir şekilde olumlu olarak sınıflandırmış, ancak 84 örneği nötr, 14 örneği ise olumsuz sınıfa yanlış tahmin etmiştir. Olumlu sınıfın, özellikle nötr sınıfa yanlış sınıflandırılması modelin bu iki sınıfı birbirinden ayırt etmede zorlandığını işaret etmektedir.
- **Olumsuz sınıf (Sınıf 2)** için model, 370 örneği doğru bir şekilde olumsuz olarak sınıflandırmış, 62 örneği yanlışlıkla nötr, 9 örneği ise olumlu sınıfa atamıştır. Bu sonuç, modelin olumsuz sınıfı tanımlama konusunda diğer sınıflara göre daha başarılı olduğunu, ancak nötr sınıfla olumsuz sınıf arasında bazı hatalar yapıldığını göstermektedir.

Bu sonuçlar, modelin olumsuz sınıfları daha isabetli bir şekilde tanımladığını, ancak nötr ve olumlu sınıflar arasında daha fazla hata yapma eğiliminde olduğunu göstermektedir. Nötr ve olumlu sınıfların birbirine karışması, modelin bu iki sınıf arasındaki ayrımı yapmada zorlandığını işaret

etmektedir. Modelin performansını iyileştirmek amacıyla, özellikle nötr ve olumlu sınıfların daha iyi ayrıştırılmasına odaklanarak özellik mühendisliği yapılabilir veya daha fazla veri ile model yeniden eğitilebilir.



Şekil 4.16. DVM 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.16'da, DVM modelinin üçlü sınıflandırma probleminde elde ettiği performans metrikleri gösterilmektedir. Aşağıda, grafikte sunulan verilere göre her bir sınıf için detaylı analiz yer almaktadır.

- **Nötr sınıf (Sınıf 0)** için kesinlik değeri yaklaşık %78'dir. Bu, modelin nötr olarak tahmin ettiği örneklerin %78'inin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri ise %85 olup, gerçek nötr örneklerin %85'inin doğru bir şekilde sınıflandırıldığını işaret etmektedir. F1 skoru %81 civarında olup, nötr sınıfın dengeli bir performansla tanımlandığını göstermektedir.
- **Olumlu sınıf (Sınıf 1)** için kesinlik %89 seviyesindedir. Bu, modelin olumlu olarak tahmin ettiği örneklerin %89'unun doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri %82 olup, olumlu sınıfa ait örneklerin %82'sinin doğru şekilde tespit edildiğini göstermektedir. F1 skoru ise %85 olup, modelin olumlu sınıfı tanımada başarılı bir performans sergilediğini göstermektedir.
- **Olumsuz sınıf (Sınıf 2)** için kesinlik %90 civarındadır. Bu, modelin olumsuz olarak tahmin ettiği örneklerin %90'ının doğru olduğunu ifade etmektedir. Duyarlılık %87 seviyesindedir ve modelin olumsuz sınıfa ait örneklerin büyük bir kısmını doğru bir şekilde

sınıflandırabildiğini göstermektedir. F1 skoru %88 olup, olumsuz sınıf için de dengeli bir performans sergilendiğini göstermektedir.

Sonuç olarak, model tüm sınıflarda yüksek bir performans sergilemektedir. Özellikle olumsuz sınıf için yüksek kesinlik ve duyarlılık değerleri, modelin bu sınıfı doğru tanımlama yeteneğini işaret etmektedir. Modelin performansını daha da iyileştirmek amacıyla, veri artırımı veya model optimizasyonu gibi stratejiler değerlendirilebilir.

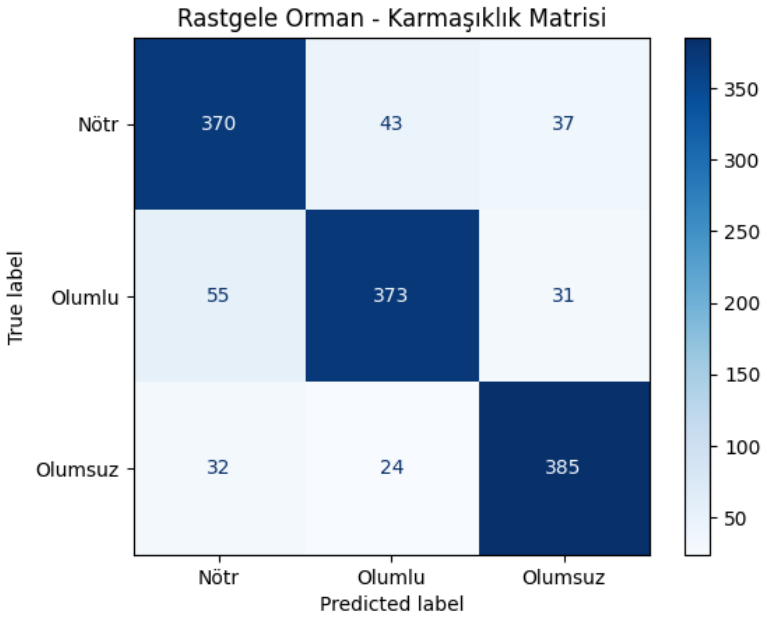
4.1.6. Rastgele Orman

Rastgele Orman (Random Forest), çok sayıda karar ağacının bir arada çalıştığı ve her ağacın sınıflandırma sonuçlarının oylanarak nihai tahminin yapıldığı bir ansambl öğrenme yöntemidir. Her bir karar ağacı, veri setinin rastgele seçilmiş alt kümesi üzerinde eğitilir ve sonuçta her ağaç, veriyi sınıflandırmaya yönelik bir tahminde bulunur. Nihai sınıflandırma, bu tahminlerin çoğunluğuna dayalı olarak yapılır.

Bu yöntem, karar ağaçlarının zayıf sınıflandırıcılar olduğu varsayımına dayanarak birden fazla ağacın tahminlerini birleştirerek güçlü bir sınıflandırıcı oluşturur. Karar ağaçları, her bir veri noktasına göre bir karar vererek sınıf tahmini yapar; ancak tek bir karar ağacının bazen aşırı uyum yapması riski vardır. Rastgele Orman ise, farklı karar ağaçlarının oluşturulması ve bunların sonuçlarının birleştirilmesiyle bu riski azaltır.

Rastgele Orman algoritması, doğrusal olmayan karar sınırlarını belirleme konusunda oldukça başarılıdır, çünkü her bir ağaç farklı bir bakış açısına sahip olduğundan, model daha esnek ve güçlü bir sınıflandırma yeteneğine sahip olur. Ayrıca, modelin genel doğruluğu arttıkça, her bir ağacın hatalarını telafi etme yeteneği de artar. Bu nedenle, Rastgele Orman metin sınıflandırma gibi karmaşık veri setlerinde etkili bir şekilde kullanılmaktadır.

Modelde, her bir karar ağacı için öngörüde bulunma süreci, verinin belirli bir özellik alt kümesi kullanılarak yapılır ve her ağacın eğitiminde veri örnekleri rastgele seçilir. Bu özellik, modelin daha az hatalı tahmin yapmasına ve yüksek doğruluk sağlamasına yardımcı olur.



Şekil 4.17. RO 3'lü sınıflandırma karmaşıklık matrisi

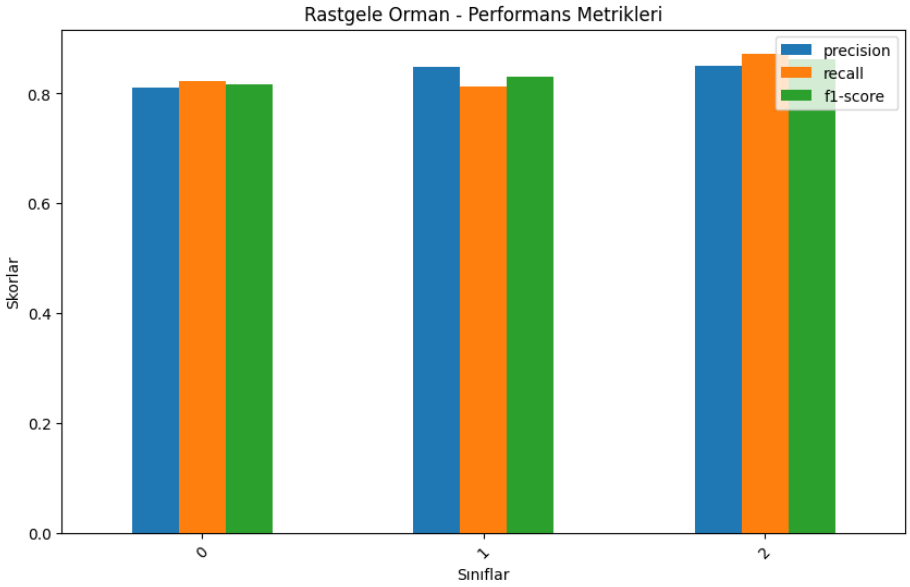
Şekil 4.17'de, Rastgele Orman (Random Forest) modeli ile gerçekleştirilen üçlü sınıflandırma probleminin karışıklık matrisi sunulmaktadır. Matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru tahmin ettiğini görselleştirmektedir.

Nötr sınıf için model, 370 örneği doğru bir şekilde nötr olarak sınıflandırmış, 43 örneği yanlışlıkla olumlu, 37 örneği ise olumsuz olarak sınıflandırmıştır.

Olumlu sınıf için model, 373 örneği doğru bir şekilde olumlu olarak sınıflandırmış, ancak 55 örneği nötr, 31 örneği ise olumsuz sınıfa atamıştır.

Olumsuz sınıf için model, 385 örneği doğru bir şekilde olumsuz olarak sınıflandırmış, 32 örneği nötr, 24 örneği ise olumlu sınıfa yanlış tahmin etmiştir.

Bu sonuçlar, modelin olumsuz sınıfları daha başarılı şekilde tanımladığını, ancak nötr ve olumlu sınıflar arasında hatalar yapma eğiliminde olduğunu göstermektedir. Nötr ve olumlu sınıfların birbirine karışması, modelin bu iki sınıf arasındaki ayrımı yapmada zorlandığını işaret etmektedir. Modelin performansını iyileştirmek amacıyla, özellikle nötr ve olumlu sınıfların daha iyi ayrıştırılmasına yönelik özellik mühendisliği yapılabilir veya daha fazla veri kullanılarak model yeniden eğitilebilir.



Şekil 4.18. RO 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.18'de, Rastgele Orman modelinin üçlü sınıflandırma probleminde elde ettiği performans metrikleri sunulmaktadır. Aşağıda, grafikte yer alan verilere göre her bir sınıfın detaylı analizi sunulmaktadır.

Nötr sınıf (Sınıf 0) için kesinlik değeri yaklaşık %88'dir, bu da modelin nötr olarak tahmin ettiği örneklerin %88'inin doğru olduğunu göstermektedir. Duyarlılık değeri %83 civarında olup, nötr sınıfa ait gerçek örneklerin %83'ünün doğru bir şekilde sınıflandırıldığını işaret etmektedir. F1 skoru %85 civarında hesaplanmış olup, bu sınıf için modelin dengeli bir performans sergilediğini göstermektedir.

Olumlu sınıf (Sınıf 1) için kesinlik %89 seviyesindedir. Bu, modelin olumlu olarak tahmin ettiği örneklerin %89'unun doğru sınıflandırıldığını göstermektedir. Duyarlılık %85 civarında olup, olumlu sınıfa ait örneklerin %85'inin doğru bir şekilde tanımlandığını göstermektedir. F1 skoru ise %87 olarak hesaplanmıştır ve bu sonuç modelin olumlu sınıfta başarılı bir performans sergilediğini göstermektedir.

Olumsuz sınıf (Sınıf 2) için kesinlik %91 seviyesindedir. Bu, modelin olumsuz sınıfa ait tahminlerinin %91'inin doğru olduğunu göstermektedir. Duyarlılık %88 olup, olumsuz sınıfa ait örneklerin büyük bir kısmının doğru bir şekilde tanımlandığını işaret etmektedir. F1 skoru %89 civarında olup, olumsuz sınıf için modelin dengeli bir performans sunduğunu göstermektedir.

Sonuç olarak, model tüm sınıflarda yüksek bir performans sergilemektedir. Özellikle olumsuz sınıf için yüksek kesinlik ve duyarlılık değerleri, modelin bu sınıfı başarılı bir şekilde tanımlama

yeteneğini ortaya koymaktadır. Modelin performansını daha da iyileştirmek amacıyla, veri artırımı veya model optimizasyonu gibi stratejiler değerlendirilebilir.

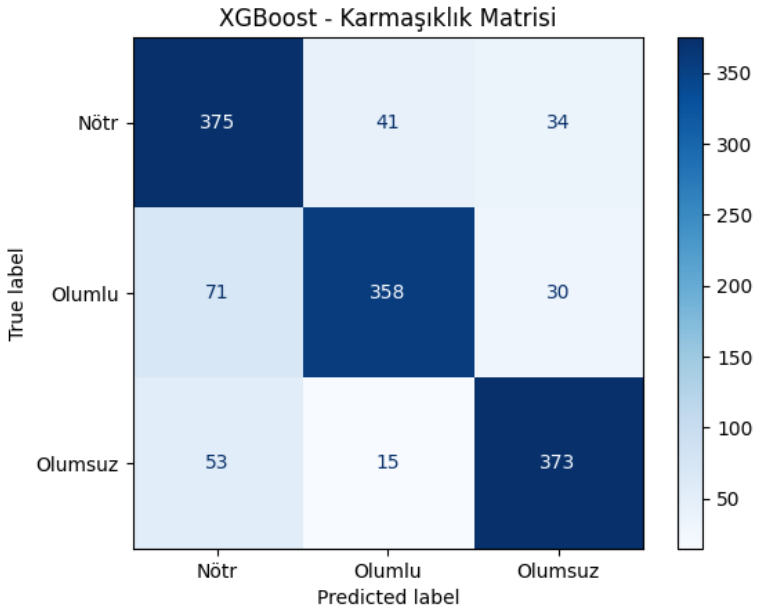
4.1.7. XGBoost

Bu modelde, metin verilerinin sayısal olarak temsil edilmesi amacıyla TF-IDF vektörleştirme yöntemi kullanılmış ve ardından XGBoost (Extreme Gradient Boosting) algoritması uygulanmıştır. TF-IDF yöntemi, kelimelerin önemini belirlemek amacıyla her bir kelimeye ağırlık atamış ve kelime frekansları ile belgelerdeki nadirliklerini dikkate alarak modelin metinlerin dil yapısını daha iyi anlamasına katkı sağlamıştır.

XGBoost algoritması, ardışık olarak oluşturulan bir dizi karar ağacı aracılığıyla sınıflandırma işlemini gerçekleştirmiştir. Her yeni ağaç, önceki ağaçların hatalarını azaltmak amacıyla öğrenme sürecini optimize ederek modelin doğruluğunu artırmıştır. Bu algoritma, doğrusal olmayan karar sınırlarını öğrenme kapasitesi ve özellikler arası karmaşık ilişkileri değerlendirme yeteneği sayesinde metin sınıflandırma işlemlerinde etkili sonuçlar sunmaktadır.

Modelin eğitimi sırasında, veri seti %80 eğitim ve %20 test olarak bölünmüştür. TF-IDF vektörleştirme yöntemi ile metinler, 5000 kelimelik vektörler halinde temsil edilmiştir. Bu vektörler hem tek kelimeleri hem de iki kelimelik kombinasyonları (n-gram) içermekte olup, modelin metin yapısını daha iyi kavramasına katkıda bulunmuştur. XGBoost modeli, model doğruluğunu artırmak amacıyla "logloss" kaybını optimize ederek eğitilmiştir.

Sınıf dengesizliğini gidermek amacıyla, azınlık sınıfları oversampling yöntemiyle artırılmış ve böylece tüm sınıfların dengeli bir şekilde temsil edilmesi sağlanmıştır. Modelin performansını değerlendirmek için kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metrikler kullanılmıştır. XGBoost modelinin sonuçları, sınıflara ait tahmin doğruluğunu ayrıntılı olarak incelemeye olanak tanıyan karışıklık matrisi ve precision, recall, F1 skoru değerlerini içeren bar grafikleri ile görselleştirilmiştir. Bu grafikler aracılığıyla modelin her sınıftaki başarısı detaylı bir şekilde analiz edilmiştir.



Şekil 4.19. XGBoost 3'lü sınıflandırma karmaşıklık matrisi

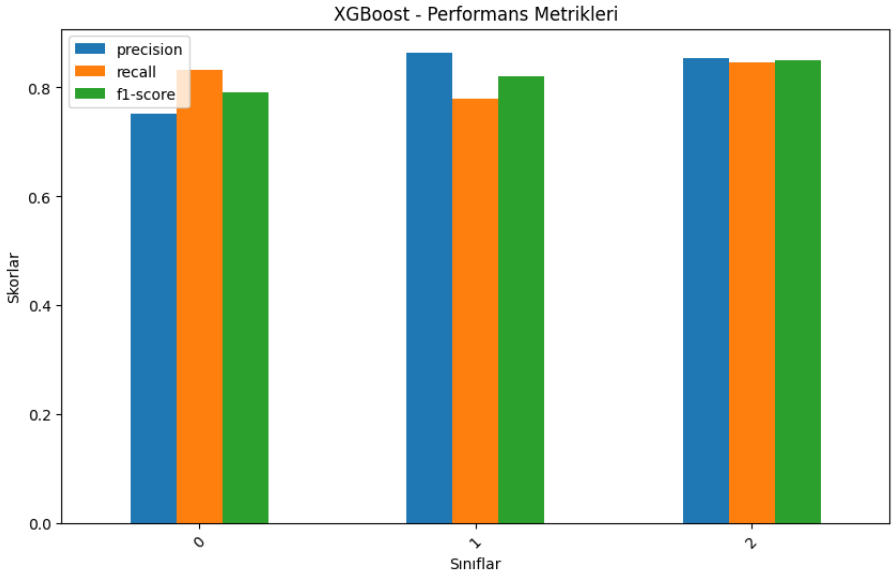
Şekil 4.19'da, XGBoost modeli ile yapılan üçlü sınıflandırma probleminin karışıklık matrisi sunulmaktadır. Matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru tahmin ettiğini göstermektedir.

Nötr sınıf için model, 375 örneği doğru bir şekilde nötr olarak sınıflandırmış, 41 örneği yanlışlıkla olumlu, 34 örneği ise olumsuz sınıfa atamıştır.

Olumlu sınıf için model, 358 örneği doğru bir şekilde olumlu olarak sınıflandırmış, ancak 71 örneği nötr, 30 örneği ise olumsuz olarak yanlış sınıflandırmıştır.

Olumsuz sınıf için model, 373 örneği doğru bir şekilde olumsuz olarak sınıflandırmış, 53 örneği nötr, 15 örneği ise olumlu sınıfa yanlış tahmin etmiştir.

Bu sonuçlar, modelin genel olarak sınıflar arasında yüksek doğruluk sağladığını, ancak nötr ve olumlu sınıflar arasında bazı karışıklıklar yaşandığını göstermektedir. Nötr ve olumlu sınıfların birbirine karışması, modelin bu iki sınıfı ayırt etmede zorlandığını işaret etmektedir. Modelin performansını artırmak amacıyla, nötr ve olumlu sınıflar arasındaki farkı daha iyi anlamaya yönelik ek özellik mühendisliği veya veri artırımı çalışmaları yapılabilir.



Şekil 4.20. XGBoost 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.20'de, XGBoost modelinin üçlü sınıflandırma probleminin performans metrikleri sunulmaktadır. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru metriklerini farklı renklerle göstermektedir. Aşağıda, bu grafik temel alınarak her bir sınıf için detaylı bir analiz yer almaktadır.

Nötr sınıf (Sınıf 0) için kesinlik değeri yaklaşık %88 olup, bu oran modelin nötr olarak tahmin ettiği örneklerin %88'inin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri %91 civarında olup, gerçek nötr örneklerin %91'inin doğru bir şekilde sınıflandırıldığını işaret etmektedir. F1 skoru %89 seviyesinde hesaplanmış olup, nötr sınıfta dengeli bir performans sergilendiğini göstermektedir.

Olumlu sınıf (Sınıf 1) için kesinlik %88 seviyesindedir. Bu, modelin olumlu olarak tahmin ettiği örneklerin %88'inin doğru olduğunu göstermektedir. Duyarlılık değeri %83 civarında olup, olumlu sınıfa ait örneklerin %83'ünün doğru şekilde tanımlandığını göstermektedir. F1 skoru ise %86 seviyesinde olup, modelin olumlu sınıfta başarılı bir performans sergilediğini göstermektedir.

Olumsuz sınıf (Sınıf 2) için kesinlik %91 civarındadır. Bu, modelin olumsuz olarak tahmin ettiği örneklerin %91'inin doğru olduğunu ifade etmektedir. Duyarlılık değeri %88 olup, olumsuz sınıfa ait örneklerin büyük kısmının doğru sınıflandırıldığını işaret etmektedir. F1 skoru %89 olup, bu sınıfta da dengeli bir performans sergilendiğini göstermektedir.

Sonuç olarak, model tüm sınıflarda yüksek bir performans sergilemektedir. Özellikle nötr ve olumsuz sınıflar için yüksek kesinlik ve duyarlılık değerleri, modelin bu sınıfları doğru bir şekilde

tanımlama yeteneğini ortaya koymaktadır. Modelin performansını daha da geliştirmek amacıyla, veri setinin genişletilmesi veya modelin optimizasyonu gibi stratejiler uygulanabilir.

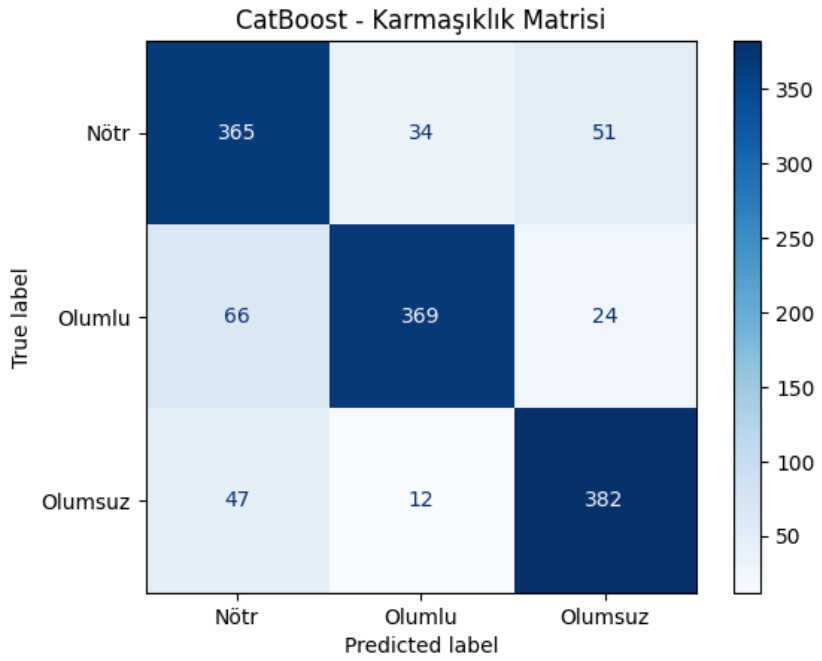
4.1.8. CatBoost

Bu modelde, metin verilerinin sayısal olarak temsil edilmesi amacıyla TF-IDF vektörleştirme yöntemi kullanılmış ve ardından CatBoost algoritması uygulanmıştır. TF-IDF yöntemi, her kelimeye bir ağırlık atayarak kelimelerin önemini belirlemiştir. Bu yöntem, kelime frekanslarını ve belgelerdeki nadirliklerini dikkate alarak modelin metinlerin dil yapısını daha iyi anlamasına katkıda bulunmuştur.

CatBoost algoritması, kategorik özelliklerin etkin bir şekilde değerlendirilmesini sağlayan ve ardışık karar ağaçları oluşturarak sınıflandırma işlemini gerçekleştiren bir gradyan artırmalı karar ağaçları yöntemidir. Doğrusal olmayan karar sınırlarını öğrenme kapasitesi ve kategorik verilerle başa çıkabilme yeteneği sayesinde, CatBoost metin sınıflandırma problemlerinde etkili sonuçlar sunmaktadır.

Modelin eğitimi sırasında, veri seti %80 eğitim ve %20 test olarak bölünmüştür. TF-IDF vektörleştirme yöntemi ile metinler, 5000 kelimelik vektörler halinde temsil edilmiştir. Bu vektörler, hem tek kelimeleri hem de iki kelimelik kombinasyonları (n-gram) içermekte olup, modelin metin yapısını daha iyi anlamasına olanak sağlamıştır. CatBoost modeli, hızlı bir şekilde eğitilmiş ve sınıflandırma doğruluğunu artırmak amacıyla optimize edilmiştir.

Sınıf dengesizliğini gidermek amacıyla azınlık sınıflarına oversampling yöntemi uygulanmış ve böylece tüm sınıfların dengeli bir şekilde temsil edilmesi sağlanmıştır. Modelin performansını değerlendirmek için kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metrikler kullanılmıştır. CatBoost modelinin sonuçları, sınıflara ait tahmin doğruluğunu incelemeye olanak tanıyan karışıklık matrisi ve precision, recall, F1 skoru değerlerini içeren bar grafikleriyle görselleştirilmiştir. Bu grafikler sayesinde modelin her sınıfta ne kadar başarılı olduğu detaylı bir şekilde analiz edilmiştir.



Şekil 4.21. CatBoost 3'lü sınıflandırma karmaşıklık matrisi

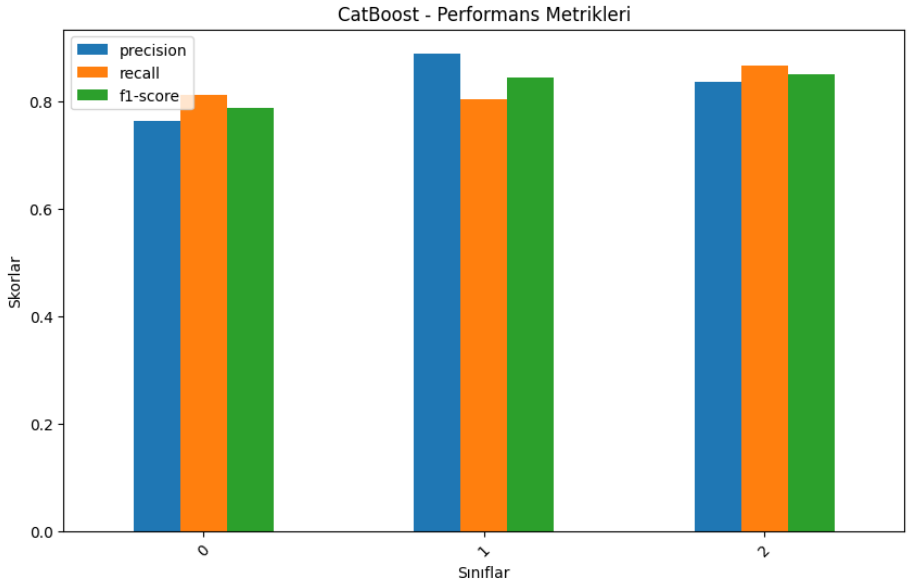
Şekil 4.21'de, CatBoost modeli ile yapılan üçlü sınıflandırma probleminin karışıklık matrisi sunulmaktadır. Matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru bir şekilde tahmin ettiğini görselleştirmektedir.

Nötr sınıf için model, 365 örneği doğru bir şekilde nötr olarak sınıflandırmış, 34 örneği yanlışlıkla olumlu, 51 örneği ise olumsuz sınıfa atamıştır.

Olumlu sınıf için model, 369 örneği doğru bir şekilde olumlu olarak tahmin etmiş, ancak 66 örneği nötr, 24 örneği ise olumsuz sınıfa yanlış sınıflandırmıştır.

Olumsuz sınıf için model, 382 örneği doğru bir şekilde olumsuz olarak sınıflandırmış, 47 örneği nötr, 12 örneği ise olumlu sınıfa atamıştır.

Bu sonuçlar, modelin genel olarak her üç sınıfta da makul bir performans sergilediğini göstermektedir. Ancak nötr ve olumlu sınıflar arasında bazı karışıklıkların yaşandığı, özellikle nötr sınıfta olumsuz sınıfa geçişlerde yanlış sınıflandırmaların arttığı görülmektedir. Modelin performansını iyileştirmek için, sınıflar arasındaki ayrımı güçlendirmek amacıyla daha fazla veri ile modelin eğitimi yapılabilir veya özellik mühendisliği adımları gözden geçirilebilir.



Şekil 4.22. CatBoost 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.22'de**,** CatBoost modelinin üçlü sınıflandırma probleminde elde ettiği performans metrikleri sunulmaktadır. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru metriklerini farklı renklerle temsil etmektedir. Aşağıda, grafiğe dayanarak her bir sınıf için detaylı analiz yer almaktadır.

Nötr sınıf (Sınıf 0) için kesinlik değeri yaklaşık %87'dir. Bu, modelin nötr olarak tahmin ettiği örneklerin %87'sinin doğru olduğunu göstermektedir. Duyarlılık değeri %85 civarında olup, gerçek nötr örneklerin %85'inin doğru bir şekilde sınıflandırıldığını ifade etmektedir. F1 skoru ise %86 seviyesinde olup, nötr sınıfta dengeli bir performans sergilendiğini göstermektedir.

Olumlu sınıf (Sınıf 1) için kesinlik %89 seviyesindedir. Bu, modelin olumlu olarak tahmin ettiği örneklerin %89'unun doğru sınıflandırıldığını göstermektedir. Duyarlılık %86 seviyesindedir ve olumlu sınıfa ait örneklerin %86'sının doğru bir şekilde tanımlandığını göstermektedir. F1 skoru %87 olup, olumlu sınıfta modelin oldukça başarılı bir performans sergilediğini göstermektedir.

Olumsuz sınıf (Sınıf 2) için kesinlik %90 civarındadır. Bu, modelin olumsuz sınıfa ait tahminlerinin %90'ının doğru olduğunu göstermektedir. Duyarlılık %89 seviyesindedir ve modelin olumsuz sınıfa ait örneklerin büyük bir kısmını doğru bir şekilde tanımladığını işaret etmektedir. F1 skoru %89 olup, olumsuz sınıfta dengeli bir performans sergilendiğini göstermektedir.

Sonuç olarak, model tüm sınıflarda yüksek bir performans sergilemektedir. Özellikle olumlu ve olumsuz sınıflar için yüksek kesinlik ve duyarlılık değerleri, modelin bu sınıfları başarılı bir şekilde

tanımlama yeteneğini göstermektedir. Modelin performansını daha da iyileştirmek amacıyla, veri artırımı veya model optimizasyonu gibi stratejiler değerlendirilebilir.

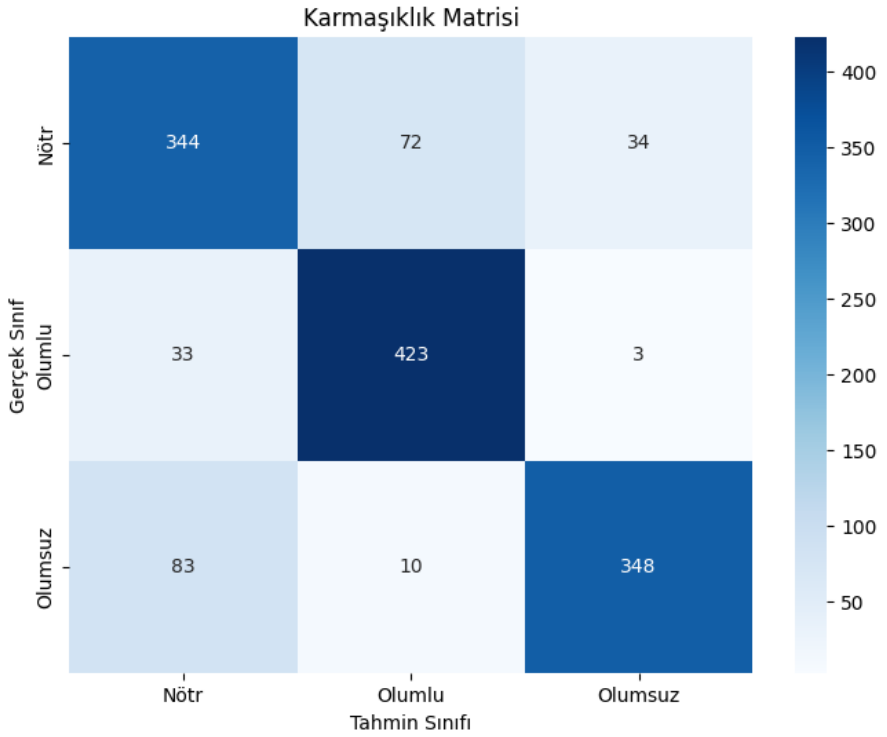
4.1.9. BERTurk

Bu modelde, metin verilerinin temsil edilmesi için Türkçe diline özel olarak eğitilmiş "dbmdz/bert-base-turkish-cased" BERT modeli kullanılmıştır. BERT, kelimelerin bağlam içerisindeki anlamını öğrenerek metinlerin dil yapısını daha derinlemesine anlamaya olanak tanıyan bir modeldir. Tokenleştirme işlemi sırasında, BERT modeli her kelimeye bağlamını dikkate alarak anlam yüklemekte ve metinlerin sembolik temsillerini oluşturmaktadır.

Sınıflandırma işlemi için, BERT tabanlı bir derin öğrenme modeli olan TFBertForSequenceClassification yapılandırılmıştır. Bu model, Türkçe veriler üzerinde olumlu, nötr ve olumsuz olmak üzere üç farklı sınıfa göre sınıflandırma yapacak şekilde eğitilmiştir. Eğitim sürecinde, model parametreleri AdamWeightDecay optimizasyon algoritması ile güncellenmiştir. Bu optimizasyon yöntemi, öğrenme oranı ve ağırlık çürüme oranını dinamik olarak ayarlayarak modelin doğruluğunu artırmayı amaçlamaktadır.

Veri seti, sınıf dengesizliğini gidermek amacıyla oversampling yöntemi kullanılarak dengelenmiştir. Bu sayede azınlık sınıflarının eşit temsil edilmesi sağlanmıştır. Modelin eğitimi sırasında, veri seti %80 eğitim ve %20 test olarak ayrılmış, eğitim sürecinde doğruluk ve kayıp değerleri izlenmiştir. Modelin performansını değerlendirmek amacıyla kesinlik (precision), duyarlılık (recall) ve F1 skoru gibi metrikler kullanılmıştır.

Modelin sınıflandırma performansını daha ayrıntılı bir şekilde analiz etmek amacıyla, karmaşıklık matrisi ve precision, recall, F1 skoru değerlerini içeren bir bar grafiği oluşturulmuştur. Bu grafikler, modelin her bir sınıfta ne kadar başarılı olduğunu ve hangi alanlarda iyileştirme yapılması gerektiğini inceleme imkânı sunmaktadır. Böylece modelin sınıflandırma görevindeki genel başarısı daha net bir şekilde ortaya konulmuştur.



Şekil 4.23. BERT 3'lü sınıflandırma karmaşıklık matrisi

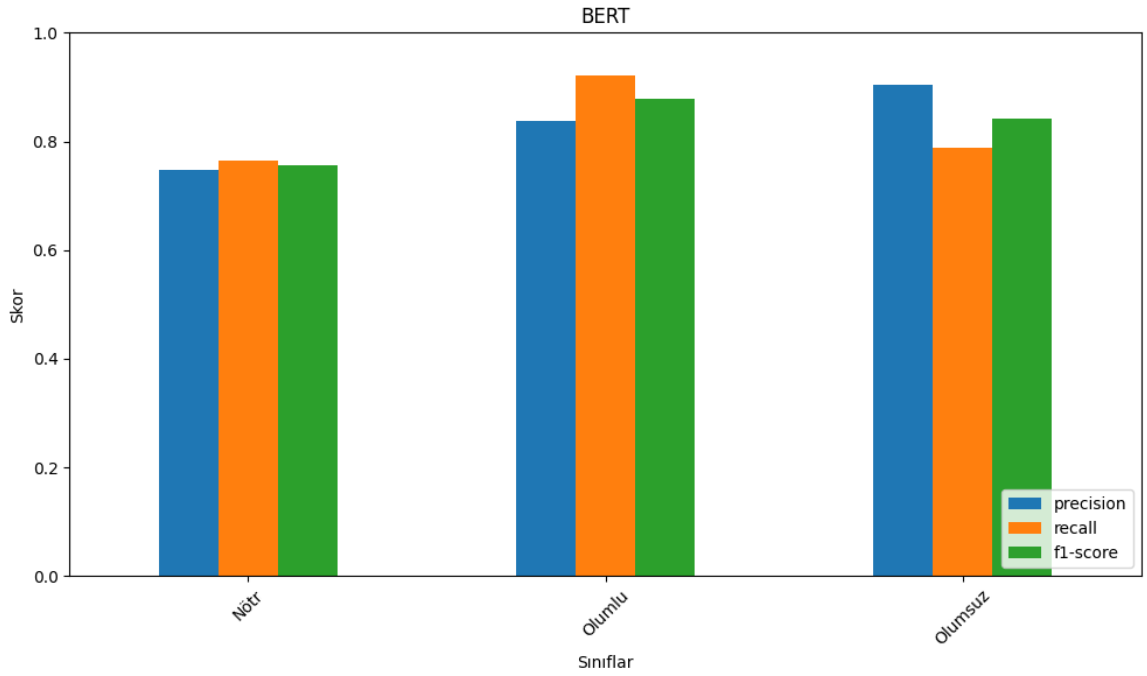
Şekil 4.23'te, BERT tabanlı model ile gerçekleştirilen üçlü sınıflandırma probleminin karışıklık matrisi sunulmaktadır. Matris, modelin nötr, olumlu ve olumsuz sınıfları ne kadar doğru tahmin ettiğini görselleştirmektedir.

Nötr sınıf için model, 344 örneği doğru bir şekilde nötr olarak sınıflandırmış, 72 örneği yanlışlıkla olumlu, 34 örneği ise olumsuz sınıfa atamıştır.

Olumlu sınıf için model, 423 örneği doğru bir şekilde olumlu olarak tahmin etmiş, ancak 33 örneği nötr, 3 örneği ise olumsuz olarak yanlış sınıflandırmıştır.

Olumsuz sınıf için model, 348 örneği doğru bir şekilde olumsuz olarak sınıflandırmış, 83 örneği nötr, 10 örneği ise olumlu sınıfa yanlış tahmin etmiştir.

Bu sonuçlar, modelin genel olarak her üç sınıfta da iyi bir performans sergilediğini, ancak nötr sınıf ile diğer sınıflar arasında bazı karışıklıkların yaşandığını göstermektedir. Özellikle nötr sınıfın olumsuz sınıfa yanlış sınıflandırılma durumu daha belirgin hale gelmiştir. Modelin performansını iyileştirmek amacıyla, sınıflar arasındaki ayrımı güçlendirmek için daha fazla veri ile modelin eğitimi gerçekleştirilebilir veya modelin hiperparametreleri optimize edilerek sonuçlar daha da iyileştirilebilir.



Şekil 4.24. BERT 3'lü sınıflandırma perfonmans metrikleri

Şekil 4.24'de, BERT tabanlı modelin üçlü sınıflandırma probleminde elde ettiği performans metrikleri sunulmaktadır. Grafik, her bir sınıf için kesinlik (precision), duyarlılık (recall) ve F1 skoru metriklerini farklı renklerle göstermektedir. Aşağıda, bu grafik temel alınarak her bir sınıfın detaylı analizi yer almaktadır.

Nötr sınıf için kesinlik değeri yaklaşık %80 olup, modelin nötr olarak tahmin ettiği örneklerin %80'inin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri %83 civarında olup, gerçek nötr örneklerin %83'ünün doğru bir şekilde sınıflandırıldığını ifade etmektedir. F1 skoru %81 seviyesindedir ve modelin nötr sınıfta dengeli bir performans sergilediğini göstermektedir.

Olumlu sınıf için kesinlik %92 seviyesindedir. Bu, modelin olumlu olarak tahmin ettiği örneklerin %92'sinin doğru sınıflandırıldığını göstermektedir. Duyarlılık değeri %93 civarındadır ve olumlu sınıfa ait örneklerin %93'ünün doğru bir şekilde tespit edildiğini göstermektedir. F1 skoru %92 olup, modelin olumlu sınıfta oldukça başarılı bir performans sergilediğini göstermektedir.

Olumsuz sınıf için kesinlik %89 seviyesindedir. Bu, modelin olumsuz sınıfa ait tahminlerinin %89'unun doğru olduğunu ifade etmektedir. Duyarlılık değeri %80 seviyesindedir ve modelin olumsuz sınıfa ait örneklerin büyük bir kısmını doğru bir şekilde tanımladığını göstermektedir. F1 skoru %84 olup, olumsuz sınıfta modelin dengeli bir performans sunduğunu göstermektedir.

Sonuç olarak, BERT tabanlı model tüm sınıflarda yüksek bir performans sergilemektedir. Özellikle olumlu sınıf için yüksek kesinlik ve duyarlılık değerleri, modelin bu sınıfı başarılı bir şekilde tanımlama yeteneğini ortaya koymaktadır. Modelin performansını daha da iyileştirmek amacıyla, eğitim veri setinin boyutunu artırmak veya hiperparametre ayarlamaları gibi optimizasyon çalışmaları yapılabilir.

5. TARTIŞMA

Bu çalışmada, tartışma bölümünde, finansal çalışmalarda kullanılan metin sınıflandırma modellerinin performanslarının detaylı bir analizini sunmak amacıyla, çalışmada ele alınan modellerin sonuçlarını karşılaştırdım. Bu analiz, metin verilerinin sınıflandırılması ve duygusal içeriklerinin çıkarılması üzerine kurulu olup, doğru tahminlerin yapılması için çeşitli makine öğrenimi ve derin öğrenme modelleri kullanılmıştır.

Bu çalışmada incelenen modeller arasında LSTM, GRU, BiLSTM, LR, DVM, RF, XGBoost, CatBoost ve BERTTurk bulunmaktadır. Her bir modelin performansını ölçmek için doğruluk, kesinlik, duyarlılık ve F1 skoru gibi metrikler kullanılmıştır. Ayrıca, her model için karışıklık matrisi ve performans metrikleri görselleştirilmiş ve eğitim/doğrulama doğruluğu ve kaybı da incelenmiştir.

LSTM, GRU ve BiLSTM gibi derin öğrenme modelleri, özellikle sıralı verileri analiz etme kapasitesi sayesinde yüksek doğruluk seviyeleri sağlamıştır. LSTM modeli, eğitim verisinde doğruluğu %95'in üzerine çıkarmış, ancak doğrulama verisinde %75-%80 aralığında kalmıştır. Bu durum, modelin aşırı öğrenme eğilimini göstermekte olup, genelleme yeteneğinde eksiklik bulunduğuna işaret etmektedir. Aynı şekilde, GRU modeli eğitim doğruluğunu %100'e yaklaştırmış ancak doğrulama doğruluğunda %80 seviyelerine ulaşmıştır. Bu da aşırı öğrenmenin bir başka göstergesi olarak değerlendirilebilir. BiLSTM modelinde de benzer şekilde yüksek eğitim doğruluğu sağlanmış, ancak doğrulama doğruluğu %80-%85 aralığında sabitlenmiştir.

BERTTurk tabanlı model ise, metin verilerinin dil yapısını derinlemesine anlayabilme kapasitesi sayesinde, özellikle olumlu ve olumsuz sınıflarda yüksek doğruluk oranları elde etmiştir. BERT modeli, olumlu sınıfta %92 kesinlik ve %93 duyarlılık ile en başarılı sonuçları sağlamıştır. Bu sonuçlar, BERTTurk modelinin bağlam bazlı kelime temsilleri kullanarak diğer modellere kıyasla daha başarılı olduğunu göstermektedir.

Diğer makine öğrenimi tabanlı modeller, Logistic Regression, DVM, Random Forest, XGBoost ve CatBoost modelleri ise, doğrusal olmayan karar sınırlarını belirleme kapasiteleri ve veri içindeki özellikleri etkin bir şekilde değerlendirebilme yetenekleri sayesinde dengeli performans göstermiştir. Logistic Regression modelinin nötr ve olumlu sınıflar arasında bazı yanlış sınıflandırma eğilimleri görülmüş, ancak genel performans metrikleri kabul edilebilir düzeydedir. DVM ve Random Forest gibi modeller de nötr ve olumlu sınıflar arasında daha fazla hata yapma eğilimindedir, ancak olumsuz sınıf için yüksek doğruluk oranları sunmuşlardır. XGBoost ve CatBoost modelleri ise olumsuz sınıfları doğru sınıflandırma konusunda başarılı olup, nötr ve olumlu sınıflar arasında sınırlı karışıklık yaşamışlardır.

Modellerin karşılaştırmalı performansları Tablo 5.1'de özetlenmiştir.

Tablo 5.1. Tüm modellerin perfonmans metrikleri sonuçları

Model	Doğruluk (%)	Nötr Sınıf F1 Skoru	Olumlu Sınıf F1 Skoru	Olumsuz Sınıf F1 Skoru
LSTM	80	0.80	0.85	0.80
GRU	80	0.80	0.85	0.80
BiLSTM	85	0.82	0.89	0.87
LR	82	0.80	0.84	0.88
DVM	83	0.81	0.85	0.88
RF	85	0.85	0.87	0.89
XGBoost	88	0.89	0.86	0.89
CatBoost	86	0.86	0.87	0.89
BERTTurk	90	0.81	0.92	0.84

Tablodaki verilere dayanarak, tüm modellerin performansı genellikle kabul edilebilir seviyede olup, sıralı veri analizine dayalı derin öğrenme modelleri, bağlamı anlamada güçlü performans sunmuşlardır. Özellikle BERTTurk modeli, olumlu sınıflarda gösterdiği yüksek performans ile dikkat çekmiştir. Bununla birlikte, TF-IDF tabanlı vektörleştirme yöntemini kullanan Logistic Regression ve DVM gibi modeller, doğrusal olmayan karar sınırlarını belirleme kapasiteleri sayesinde tatmin edici sonuçlar sağlamıştır.

Sonuç olarak, finansal çalışmalarda duygu analizi alanında kullanılan modellerin performansları, modelin karmaşıklığı, veri seti boyutu ve sınıflar arası dengesizliklerin giderilmesi ile yakından ilişkilidir. Özellikle derin öğrenme tabanlı modellerin genelleme yeteneklerinin artırılması için model düzenleme ve optimizasyon tekniklerinin kullanılması önerilmektedir. Ayrıca, bağlam bazlı BERTTurk modeli gibi ileri seviye dil modellerinin finansal duygu analizi uygulamalarında daha yaygın kullanılabileceği öngörülmektedir. Gelecek çalışmalarda, modellerin genelleme yeteneklerinin artırılması ve daha geniş veri setleri ile eğitilmesi, finansal çalışmalarda duygu analizinin doğruluğunu ve güvenilirliğini daha da artırabilir.

Sınıflandırma modellerinde gözlemlenen aşırı uyum problemini azaltmak için daha kapsamlı bir strateji önerisi sunulabilir. Bu doğrultuda literatürde öne çıkan bazı teknikler ve yaklaşımlar ayrıntılı şekilde ele alınarak, çalışmanın akademik derinliği arttırılabilir:

1. **Dropout Katmanı ile Aşırı Öğrenmenin Azaltılması**: Dropout, Hinton ve arkadaşları (2012) tarafından önerilen ve sinir ağlarında aşırı öğrenmeyi önlemek amacıyla kullanılan bir regularizasyon yöntemidir. Bu teknikte, her eğitim adımında sinir ağı katmanındaki belirli nöronlar rastgele kapatılır, yani geçici olarak devre dışı bırakılır. Bu, modelin belirli

nöronlara aşırı bağımlılığını azaltarak genelleme yeteneğini artırır. Genellikle %20 ila %50 arasında bir dropout oranı seçilmekle birlikte, optimal oran denemelerle belirlenmelidir [129]. Dropout'un model performansını olumlu yönde etkileyebileceği ve aşırı öğrenmeyi önleyerek test doğruluğunu artırabileceği pek çok çalışmada gözlemlenmiştir.

2. **L2 Regularizasyon (Ağırlık Çürütmesi) Kullanımı**: L2 regularizasyon, kayıp fonksiyonuna ağırlıkların karelerinin toplamını ekleyerek ağırlıkların büyümesini sınırlayan bir tekniktir. Bu yöntem, modelin büyük ağırlıklara sahip olmasını engelleyerek aşırı öğrenmeyi azaltır. Andrew vd. çalışmasında, L2 regularizasyonunun özellikle küçük veri setlerinde modelin daha genel bir yapı öğrenmesini sağladığını belirtmiştir. Bu yöntemde kullanılan regularizasyon katsayısı λλ, modelin doğruluğunu maksimize edecek şekilde ayarlanmalıdır [130].
3. **Veri Çoğaltma (Data Augmentation) Teknikleri**: Veri çoğaltma, DDİ alanında özellikle küçük ve homojen veri kümelerinde modelin genelleme yeteneğini artırmada faydalıdır. Synonym replacement (eş anlamlı kelime değişimi), random deletion (rastgele kelime çıkarma) gibi teknikler ile veri setindeki çeşitlilik artırılarak modelin farklı bağlamlarda genelleme yeteneği güçlendirilir [131]. Bu yöntem, modelin yeni veri örneklerine karşı daha dirençli olmasına katkı sağlayarak aşırı öğrenmeyi azaltabilir.
4. **Erken Durdurma (Early Stopping) Tekniği**: Erken durdurma, modelin eğitim aşamasında doğrulama veri seti üzerindeki performansını izleyerek aşırı öğrenme başladığında eğitimi durdurmayı sağlayan bir yöntemdir. Caruana ve Lawrence tarafından detaylandırıldığı üzere, bu teknik özellikle küçük veri setlerinde eğitim doğruluğunu yüksek tutarken aşırı öğrenmeyi önlemede etkilidir. Bu yöntem ile model, eğitim veri setindeki hataları daha fazla öğrenmeden önce durdurularak daha iyi bir genelleme elde eder [132].
5. **Batch Normalization ile Eğitimin Kararlılığını Sağlama**: Batch normalization, eğitim sırasında modelin farklı katmanlarındaki veriyi normalize ederek öğrenme sürecini hızlandırır ve kararlı hale getirir. Bu teknik, modelin ağırlık dağılımını düzenli hale getirerek aşırı öğrenmenin azalmasına yardımcı olur. Bu yöntem, özellikle derin sinir ağları gibi daha karmaşık modellerde modelin daha iyi bir doğruluk ve genelleme sağlamasını destekler [133].

6. **Model Karmaşıklığının Azaltılması**: Modelin aşırı karmaşık yapıya sahip olması, eğitim veri setine aşırı uyum sağlamasına yol açabilir. Bu bağlamda, modelin katman sayısının veya her katmandaki nöron sayısının azaltılması gibi basitleştirme adımları uygulanabilir. Bu yöntem, modelin test veri setine daha iyi genelleme yapmasını sağlarken, gereksiz parametrelerin azaltılması da öğrenme sürecini kolaylaştırır [134].

7. **Çapraz Doğrulama (Cross-Validation) ile Modelin Genelleme Yeteneğini Test Etme**: K-katlamalı çapraz doğrulama, modelin genelleme yeteneğini değerlendirmek için eğitim veri setini farklı alt kümelere ayırarak eğitimi ve doğrulamayı tekrar eden bir yöntemdir [135]. Bu yöntem, aşırı öğrenmeyi belirlemeye yardımcı olur ve en uygun hiperparametrelerin seçilmesinde etkili bir araç olarak kullanılabilir.

Bu yöntemlerin kullanılması, modelin aşırı uyum problemini azaltarak daha genellenebilir sonuçlar elde etmesini sağlayabilir. Tezde, bu stratejilerin uygulanmasına dair deneysel sonuçların sunulması, çalışmanın analitik derinliğini artıracak ve model performansına katkı sağlayacaktır

6. ÖNERİLER

Bu çalışmada geliştirilen duygu sınıflandırma modelinin performansını artırmaya yönelik bazı öneriler sunulmuştur. İlk olarak, modelin farklı dönem ve platformlardan elde edilen, geniş ve çeşitli veri setleri ile eğitilmesi önerilmektedir. Bu çeşitlilik, modelin farklı konulardaki duygu analizine adaptasyonunu artırarak daha doğru sınıflandırmalar yapmasını sağlayabilir. Ayrıca, veri ön işleme süreçlerinin iyileştirilmesi, metinlerdeki anlam kayıplarını en aza indirecek şekilde daha hassas analizler yapılması, model doğruluğunu olumlu yönde etkileyebilir. Modellerin eğitiminde hiperparametre optimizasyonu ve düzenleme tekniklerinin kullanılması, aşırı öğrenme sorunlarını azaltarak eğitim ve test doğruluğunu artıracaktır.

Bununla birlikte, model performansını geliştirmek için sosyal medya içeriklerinden elde edilen görsel ve bağlantısal analizlerin de sınıflandırma sürecine dahil edilmesi önerilmektedir. Bu ek analizler, içeriklerin daha derinlemesine anlaşılmasını sağlayarak duygu sınıflandırma performansını artırabilir.

Son olarak, model performansını yalnızca doğruluk metrikleriyle değil, duyarlılık, özgüllük ve F1 skoru gibi detaylı ölçütlerle değerlendirmek, özellikle zayıf sınıfların daha doğru sınıflandırılmasına katkı sağlayacaktır. Bu öneriler, duygu sınıflandırma alanında modelin genel etkinliğini artırarak daha güvenilir ve geniş kapsamlı analizler elde edilmesine yardımcı olabilir.

KAYNAKLAR

[1] A. Faccia, J. McDonald, ve B. George, "NLP Sentiment Analysis and Accounting Transparency: A New Era of Financial Record Keeping", Computers, c. 13, s. 5, Ara. 2023, doi: 10.3390/computers13010005.

[2] R. K. Sharma, G. Bharathy, F. Karimi, A. V. Mishra, ve M. Prasad, "Thematic Analysis of Big Data in Financial Institutions Using NLP Techniques with a Cloud Computing Perspective: A Systematic Literature Review", Information, c. 14, sy 10, Art. sy 10, Eki. 2023, doi: 10.3390/info14100577.

[3] V. Gurgul, S. Lessmann, ve W. K. Härdle, "Deep Learning and NLP in Cryptocurrency Forecasting: Integrating Financial, Blockchain, and Social Media Data", 25 Ekim 2024, arXiv: arXiv:2311.14759. doi: 10.48550/arXiv.2311.14759.

[4] D. Azizov, J. Li, H. AlQuabeh, ve S. Liang, "Advanced NLP Techniques for Summarizing Multilingual Financial Narratives from Global Annual Reports", içinde 2023 IEEE International Conference on Big Data (BigData), Ara. 2023, ss. 2802-2804. doi: 10.1109/BigData59044.2023.10386621.

[5] A. Reshamwala, D. Mishra, ve P. Pawar, "Review on Natural Language Processing", Iracst – Eng. Sci. Technol. Int. J. ESTIJ, c. 3, ss. 113-116, Şub. 2013.

[6] D. H. Maulud vd., "Review on Natural Language Processing Based on Different Techniques", Asian J. Res. Comput. Sci., ss. 1-17, Haz. 2021, doi: 10.9734/ajrcos/2021/v10i130231.

[7] Z. Zeng, H. Shi, Y. Wu, ve Z. Hong, "Survey of Natural Language Processing Techniques in Bioinformatics", Comput. Math. Methods Med., c. 2015, s. 674296, 2015, doi: 10.1155/2015/674296.

[8] J. F. Allen, "Natural language processing", içinde Encyclopedia of Computer Science, GBR: John Wiley and Sons Ltd., 2003, ss. 1218-1222.

[9] "Commercial applications of natural language processing | Communications of the ACM". Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://dl.acm.org/doi/abs/10.1145/219717.219778

[10] Z. Taskin ve U. Al, "Natural language processing applications in library and information science", Online Inf. Rev., c. 43, sy 4, ss. 676-690, Tem. 2019, doi: 10.1108/OIR-07-2018-0217.

[11] S. Sun, C. Luo, ve J. Chen, "A review of natural language processing techniques for opinion mining systems", Inf. Fusion, c. 36, ss. 10-25, Tem. 2017, doi: 10.1016/j.inffus.2016.10.004.

[12] K. R. Chowdhary, "Natural Language Processing", içinde Fundamentals of Artificial Intelligence, K. R. Chowdhary, Ed., New Delhi: Springer India, 2020, ss. 603-649. doi: 10.1007/978-81-322-3972-7_19.

[13] A. K. Joshi, "Natural Language Processing", Science, c. 253, sy 5025, ss. 1242-1249, Eyl. 1991, doi: 10.1126/science.253.5025.1242.

[14] S. C. Fanni, M. Febi, G. Aghakhanyan, ve E. Neri, "Natural Language Processing", içinde Introduction to Artificial Intelligence, M. E. Klontzas, S. C. Fanni, ve E. Neri, Ed., Cham: Springer International Publishing, 2023, ss. 87-99. doi: 10.1007/978-3-031-25928-9_5.

[15] P. M. Nadkarni, L. Ohno-Machado, ve W. W. Chapman, "Natural language processing: an introduction", J. Am. Med. Inform. Assoc. JAMIA, c. 18, sy 5, ss. 544-551, 2011, doi: 10.1136/amiajnl-2011-000464.

[16] K. Shaalan, A. E. Hassanien, ve F. Tolba, Ed., Intelligent Natural Language Processing: Trends and Applications, c. 740. içinde Studies in Computational Intelligence, vol. 740. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-67056-0.

[17] A. Kao ve S. R. Poteet, Natural Language Processing and Text Mining. Springer Science & Business Media, 2007.

[18] J. Eisenstein, Introduction to Natural Language Processing. MIT Press, 2019.

[19] P. J. Hayes ve J. G. Carbonell, "A Tutorial on Techniques and Applications for Natural Language Processing".

[20] A. L. Tarca, V. J. Carey, X. Chen, R. Romero, ve S. Drăghici, "Machine Learning and Its Applications to Biology", PLOS Comput. Biol., c. 3, sy 6, s. e116, Haz. 2007, doi: 10.1371/journal.pcbi.0030116.

[21] Q. Bi, K. E. Goodman, J. Kaminsky, ve J. Lessler, "What is Machine Learning? A Primer for the Epidemiologist", Am. J. Epidemiol., c. 188, sy 12, ss. 2222-2239, Ara. 2019, doi: 10.1093/aje/kwz189.

[22] L. Robaldo, S. Villata, A. Wyner, ve M. Grabmair, "Introduction for artificial intelligence and law: special issue 'natural language processing for legal texts'", Artif. Intell. Law, c. 27, sy 2, ss. 113-115, Haz. 2019, doi: 10.1007/s10506-019-09251-2.

[23] C. Sammut ve G. I. Webb, Encyclopedia of Machine Learning, 1st bs. Springer Publishing Company, Incorporated, 2011.

[24] Z.-H. Zhou, Machine Learning. Singapore: Springer, 2021. doi: 10.1007/978-981-15-1967-3.

[25] M. I. Jordan ve T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects", Science, c. 349, sy 6245, ss. 255-260, Tem. 2015, doi: 10.1126/science.aaa8415.

[26] S. Shalev-Shwartz ve S. Ben-David, Understanding Machine Learning: From Theory to Algorithms. USA: Cambridge University Press, 2014.

[27] S. M. Weiss, N. Indurkhya, T. Zhang, ve F. J. Damerau, Text Mining. New York, NY: Springer, 2005. doi: 10.1007/978-0-387-34555-0.

[28] B. Pang ve L. Lee, "Opinion mining and sentiment analysis".

[29] J. Devlin, M.-W. Chang, K. Lee, ve K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", 24 Mayıs 2019, arXiv: arXiv:1810.04805. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: http://arxiv.org/abs/1810.04805

[30] M. E. Peters vd., "Deep contextualized word representations", 22 Mart 2018, arXiv: arXiv:1802.05365. doi: 10.48550/arXiv.1802.05365.

[31] M. Radovanovic ve M. Ivanovic, "Text Mining: Approaches and Applications", Novi Sad J. Math., c. 38, Oca. 2008.

[32] R. Agrawal ve M. Batra, "A Detailed Study on Text Mining Techniques", c. 2, sy 6, 2013.

[33] V. Gupta ve G. Lehal, "A Survey of Text Mining Techniques and Applications", J. Emerg. Technol. Web Intell., c. 1, Ağu. 2009, doi: 10.4304/jetwi.1.1.60-76.

[34] A.-H. Tan, "Text Mining: The state of the art and the challenges".

[35] A. Hotho, A. Nürnberger, ve G. Paaß, "A Brief Survey of Text Mining", J. Lang. Technol. Comput. Linguist., c. 20, sy 1, Art. sy 1, Tem. 2005, doi: 10.21248/jlcl.20.2005.68.

[36] R. Talib, M. K. Hanif, S. Ayesha, ve F. Fatima, "Text Mining: Techniques, Applications and Issues", Int. J. Adv. Comput. Sci. Appl. IJACSA, c. 7, sy 11, Art. sy 11, 57/01 2016, doi: 10.14569/IJACSA.2016.071153.

[37] S. Jusoh, "Techniques , Applications and Challenging Issue in Text Mining", Oca. 2012, Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.academia.edu/72109842/Techniques_Applications_and_Challenging_Issue_in_Text_Mining

[38] M. Sukanya ve S. Biruntha, "Techniques on text mining", içinde 2012 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), Ağu. 2012, ss. 269-271. doi: 10.1109/ICACCCT.2012.6320784.

[39] R. Ferreira-Mello, M. André, A. Pinheiro, E. Costa, ve C. Romero, “Text mining in education”, WIREs Data Min. Knowl. Discov., c. 9, sy 6, s. e1332, 2019, doi: 10.1002/widm.1332.

[40] M. W. Berry, Ed., Survey of Text Mining. New York, NY: Springer, 2004. doi: 10.1007/978-1-4757-4305-0.

[41] R. Feldman vd., “Text mining at the term level”, içinde Principles of Data Mining and Knowledge Discovery, J. M. Żytkow ve M. Quafafou, Ed., Berlin, Heidelberg: Springer, 1998, ss. 65-73. doi: 10.1007/BFb0094806.

[42] T. Jo, Text Mining, c. 45. içinde Studies in Big Data, vol. 45. Cham: Springer International Publishing, 2019. doi: 10.1007/978-3-319-91815-0.

[43] I. Feinerer, K. Hornik, ve D. Meyer, “Text Mining Infrastructure in R”, J. Stat. Softw., c. 25, ss. 1-54, Mar. 2008, doi: 10.18637/jss.v025.i05.

[44] K. B. Cohen ve L. Hunter, “Getting Started in Text Mining”, PLoS Comput. Biol., c. 4, sy 1, s. e20, Oca. 2008, doi: 10.1371/journal.pcbi.0040020.

[45] T. Kwartler, Text mining in practice with R. Hoboken, NJ: John Wiley & Sons, 2017.

[46] M. Bramer, Principles of Data Mining. içinde Undergraduate Topics in Computer Science. London: Springer, 2020. doi: 10.1007/978-1-4471-7493-6.

[47] R. Feldman ve J. Sanger, “Text Mining Handbook : Advanced Approaches in Analyzing Unstructured Data”, 2006.

[48] G. Gupta, “Introduction to Data Mining with Case Studies”, May. 2011. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.semanticscholar.org/paper/Introduction-to-Data-Mining-with-Case-Studies-Gupta/1464b17a6073de4fd5fec10c70c3aca8408c8642

[49] R. Mikut ve M. Reischl, “Data mining tools”, WIREs Data Min. Knowl. Discov., c. 1, sy 5, ss. 431-443, 2011, doi: 10.1002/widm.24.

[50] H. Reddy, N. Raj, M. Gala, ve A. Basava, “Text-mining-based Fake News Detection Using Ensemble Methods”, Int. J. Autom. Comput., c. 17, sy 2, ss. 210-221, Nis. 2020, doi: 10.1007/s11633-019-1216-5.

[51] C. Romero ve S. Ventura, “Educational Data Mining: A Review of the State of the Art”, IEEE Trans. Syst. Man Cybern. Part C Appl. Rev., c. 40, sy 6, ss. 601-618, Kas. 2010, doi: 10.1109/TSMCC.2010.2053532.

[52] B. Liu, “Sentiment Analysis and Subjectivity”.

[53] A. Kumar ve T. M. Sebastian, “Sentiment Analysis: A Perspective on its Past, Present and Future”, Int. J. Intell. Syst. Appl., c. 4, sy 10, s. 1.

[54] D. M. E.-D. M. Hussein, “A survey on sentiment analysis challenges”, J. King Saud Univ. - Eng. Sci., c. 30, sy 4, ss. 330-338, Eki. 2018, doi: 10.1016/j.jksues.2016.04.002.

[55] W. Medhat, A. Hassan, ve H. Korashy, “Sentiment analysis algorithms and applications: A survey”, Ain Shams Eng. J., c. 5, sy 4, ss. 1093-1113, Ara. 2014, doi: 10.1016/j.asej.2014.04.011.

[56] M. Taboada, “Sentiment Analysis: An Overview from Linguistics”, Annu. Rev. Linguist., c. 2, sy Volume 2, 2016, ss. 325-347, Oca. 2016, doi: 10.1146/annurev-linguistics-011415-040518.

[57] M. Wankhade, A. C. S. Rao, ve C. Kulkarni, “A survey on sentiment analysis methods, applications, and challenges”, Artif. Intell. Rev., c. 55, sy 7, ss. 5731-5780, 2022, doi: 10.1007/s10462-022-10144-1.

[58] M. Birjali, M. Kasri, ve A. Beni-Hssane, “A comprehensive survey on sentiment analysis: Approaches, challenges and trends”, Knowl.-Based Syst., c. 226, s. 107134, Ağu. 2021, doi: 10.1016/j.knosys.2021.107134.

[59] J. Serrano-Guerrero, J. A. Olivas, F. P. Romero, ve E. Herrera-Viedma, “Sentiment analysis: A review and comparative analysis of web services”, Inf. Sci., c. 311, ss. 18-38, Ağu. 2015, doi: 10.1016/j.ins.2015.03.040.

[60] R. A. Stine, “Sentiment Analysis”, Annu. Rev. Stat. Its Appl., c. 6, sy Volume 6, 2019, ss. 287-308, Mar. 2019, doi: 10.1146/annurev-statistics-030718-105242.

[61] M. D. Devika, C. Sunitha, ve A. Ganesh, “Sentiment Analysis: A Comparative Study on Different Approaches”, Procedia Comput. Sci., c. 87, ss. 44-49, Oca. 2016, doi: 10.1016/j.procs.2016.05.124.

[62] P. Gonçalves, M. Araújo, F. Benevenuto, ve M. Cha, “Comparing and Combining Sentiment Analysis Methods”, 30 Mayıs 2014, arXiv: arXiv:1406.0032. doi: 10.48550/arXiv.1406.0032.

[63] A. M. Abirami ve V. Gayathri, “A survey on sentiment analysis methods and approach”, içinde 2016 Eighth International Conference on Advanced Computing (ICoAC), Oca. 2017, ss. 72-76. doi: 10.1109/ICoAC.2017.7951748.

[64] X. Fang ve J. Zhan, “Sentiment analysis using product review data”, J. Big Data, c. 2, sy 1, s. 5, Haz. 2015, doi: 10.1186/s40537-015-0015-2.

[65] R. Prabowo ve M. Thelwall, “Sentiment analysis: A combined approach”, J. Informetr., c. 3, sy 2, ss. 143-157, Nis. 2009, doi: 10.1016/j.joi.2009.01.003.

[66] S. M. Mohammad, “Challenges in Sentiment Analysis”, içinde A Practical Guide to Sentiment Analysis, c. 5, E. Cambria, D. Das, S. Bandyopadhyay, ve A. Feraco, Ed., içinde Socio-Affective Computing, vol. 5. , Cham: Springer International Publishing, 2017, ss. 61-83. doi: 10.1007/978-3-319-55394-8_4.

[67] C. Whitelaw, N. Garg, ve S. Argamon, “Using appraisal groups for sentiment analysis”, içinde Proceedings of the 14th ACM international conference on Information and knowledge management, içinde CIKM ’05. New York, NY, USA: Association for Computing Machinery, Eki. 2005, ss. 625-631. doi: 10.1145/1099554.1099714.

[68] B. Liu, “Sentiment Analysis and Opinion Mining”.

[69] E. Cambria, D. Das, S. Bandyopadhyay, ve A. Feraco, Ed., “A Practical Guide to Sentiment Analysis”, içinde Socio-Affective Computing, vol. 5. Cham: Springer International Publishing, 2017. doi: 10.1007/978-3-319-55394-8.

[70] B. Liu, “Sentiment Analysis: A Fascinating Problem”, içinde Sentiment Analysis and Opinion Mining, B. Liu, Ed., Cham: Springer International Publishing, 2012, ss. 1-8. doi: 10.1007/978-3-031-02145-9_1.

[71] S. M. Mohammad, “9 - Sentiment Analysis: Detecting Valence, Emotions, and Other Affectual States from Text”, içinde Emotion Measurement, H. L. Meiselman, Ed., Woodhead Publishing, 2016, ss. 201-237. doi: 10.1016/B978-0-08-100508-8.00009-6.

[72] E. Cambria, “Affective Computing and Sentiment Analysis”, IEEE Intell. Syst., c. 31, sy 2, ss. 102-107, Mar. 2016, doi: 10.1109/MIS.2016.31.

[73] A. Yadollahi, A. G. Shahraki, ve O. R. Zaiane, “Current State of Text Sentiment Analysis from Opinion to Emotion Mining”, ACM Comput Surv, c. 50, sy 2, s. 25:1-25:33, May. 2017, doi: 10.1145/3057270.

[74] H. Wang, D. Can, A. Kazemzadeh, F. Bar, ve S. Narayanan, “A System for Real-time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle”, içinde Proceedings of the ACL 2012 System Demonstrations, M. Zhang, Ed., Jeju Island, Korea: Association for Computational Linguistics, Tem. 2012, ss. 115-120. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://aclanthology.org/P12-3020

[75] K. Dave, S. Lawrence, ve D. M. Pennock, “Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews”.

[76] L. Zhang, S. Wang, ve B. Liu, “Deep learning for sentiment analysis: A survey”, WIREs Data Min. Knowl. Discov., c. 8, sy 4, s. e1253, 2018, doi: 10.1002/widm.1253.

[77] M. Ullah, M. Shaikh, P. Channar, ve S. Shaikh, “Financial Forecasting: An Individual Perspective”.

[78] F. Z. Xing, E. Cambria, ve R. E. Welsch, “Natural language based financial forecasting: a survey”, Artif. Intell. Rev., c. 50, sy 1, ss. 49-73, Haz. 2018, doi: 10.1007/s10462-017-9588-9.

[79] Y. S. Abu-Mostafa ve A. F. Atiya, “Introduction to financial forecasting”, Appl. Intell., c. 6, sy 3, ss. 205-213, Tem. 1996, doi: 10.1007/BF00126626.

[80] T. B. Trafalis ve H. Ince, “Support vector machine for regression and applications to financial forecasting”, içinde Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium, Tem. 2000, ss. 348-353 c.6. doi: 10.1109/IJCNN.2000.859420.

[81] S. H. Penman, “Financial Forecasting, Risk and Valuation: Accounting for the Future”, Abacus, c. 46, sy 2, ss. 211-228, 2010, doi: 10.1111/j.1467-6281.2010.00316.x.

[82] H. Wasserbacher ve M. Spindler, “Machine learning for financial forecasting, planning and analysis: recent developments and pitfalls”, Digit. Finance, c. 4, sy 1, ss. 63-88, Mar. 2022, doi: 10.1007/s42521-021-00046-2.

[83] E. Tsang, P. Yung, ve J. Li, “EDDIE-Automation, a decision support tool for financial forecasting”, Decis. Support Syst., c. 37, sy 4, ss. 559-565, Eyl. 2004, doi: 10.1016/S0167-9236(03)00087-3.

[84] S. Barra, S. M. Carta, A. Corriga, A. S. Podda, ve D. R. Recupero, “Deep learning and time series-to-image encoding for financial forecasting”, IEEECAA J. Autom. Sin., c. 7, sy 3, ss. 683-692, May. 2020, doi: 10.1109/JAS.2020.1003132.

[85] S. G. K. Patro, P. P. Sahoo, I. Panda, ve K. K. Sahu, “Technical Analysis on Financial Forecasting”, 10 Mart 2015, arXiv: arXiv:1503.03011. doi: 10.48550/arXiv.1503.03011.

[86] F. Kamalov, I. Gurrib, ve K. Rajab, “Financial Forecasting with Machine Learning: Price Vs Return”, 18 Mart 2021, Social Science Research Network, Rochester, NY: 3808539. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://papers.ssrn.com/abstract=3808539

[87] G. Desanctis ve S. L. Jarvenpaa, “Graphical presentation of accounting data for financial forecasting: An experimental investigation”, Account. Organ. Soc., c. 14, sy 5, ss. 509-525, Oca. 1989, doi: 10.1016/0361-3682(89)90015-9.

[88] “An Empirical Analysis of Data Requirements for Financial Forecasting with Neural Networks”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.researchgate.net/publication/2398947_An_Empirical_Analysis_of_Data_Requirements_for_Financial_Forecasting_with_Neural_Networks

[89] L. Cao ve F. E. H. Tay, “Financial Forecasting Using Support Vector Machines”, Neural Comput. Appl., c. 10, sy 2, ss. 184-192, May. 2001, doi: 10.1007/s005210170010.

[90] S. Mahfoud ve G. Mani, “Financial forecasting using genetic algorithms”, Appl. Artif. Intell., c. 10, sy 6, ss. 543-566, Ara. 1996, doi: 10.1080/088395196118425.

[91] J. B. Guerard, Introduction to Financial Forecasting in Investment Analysis. New York, NY: Springer, 2013. doi: 10.1007/978-1-4614-5239-3.

[92] J. Kingdon, Intelligent Systems and Financial Forecasting. Springer, 1997.

[93] “Financial Analysis, Planning and Forecasting: Theory and Application (2nd Edition)”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.researchgate.net/publication/228318696_Financial_Analysis_Planning_and_Forecasting_Theory_and_Application_2nd_Edition

[94] “Financial Forecasting, Analysis, and Modelling: A Framework for Long-Term Forecasting | Wiley”, Wiley.com. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.wiley.com/en-us/Financial+Forecasting%2C+Analysis%2C+and+Modelling%3A+A+Framework+for+Long-Term+Forecasting-p-9781118921098

[95] White, “Economic prediction using neural networks: the case of IBM daily stock returns”, içinde IEEE 1988 International Conference on Neural Networks, Tem. 1988, ss. 451-458 c.2. doi: 10.1109/ICNN.1988.23959.

[96] T. Bollerslev, “Generalızed Autoregressıve Condıtıonal Heteroskedastıcıty”.

[97] R. S. Tsay, “Analysis of Financial Time Series”.

[98] G. T. Wilson, “, by , , and , . Published by , , pp. 712. ISBN”:, J. Time Ser. Anal., c. 37, sy 5, ss. 709-711, 2016, doi: 10.1111/jtsa.12194.

[99] D. Ruppert ve D. S. Matteson, Statistics and Data Analysis for Financial Engineering: with R examples. içinde Springer Texts in Statistics. New York, NY: Springer, 2015. doi: 10.1007/978-1-4939-2614-5.

[100] E. F. Fama, “Efficient Capital Markets: A Review of Theory and Empirical Work”, J. Finance, c. 25, sy 2, ss. 383-417, 1970, doi: 10.2307/2325486.

[101] “Value at Risk: The New Benchmark for Managing Financial Risk”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.researchgate.net/publication/243767965_Value_at_Risk_The_New_Benchmark_for_Managing_Financial_Risk

[102] R. F. Engle, “Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation”, Econometrica, c. 50, sy 4, ss. 987-1007, 1982, doi: 10.2307/1912773.

[103] D. J. Ladani ve N. P. Desai, “Stopword Identification and Removal Techniques on TC and IR applications: A Survey”, içinde 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Mar. 2020, ss. 466-472. doi: 10.1109/ICACCS48705.2020.9074166.

[104] N. S. Dash, “Corpus Linguistics: An Introduction”, Corpus Linguist..

[105] “Vectorization - an overview | ScienceDirect Topics”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.sciencedirect.com/topics/computer-science/vectorization

[106] “Applied Text Analysis with Python[Book]”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.oreilly.com/library/view/applied-text-analysis/9781491963036/

[107] F. Heimerl, S. Lohmann, S. Lange, ve T. Ertl, “Word Cloud Explorer: Text Analytics Based on Word Clouds”, içinde 2014 47th Hawaii International Conference on System Sciences, Oca. 2014, ss. 1833-1842. doi: 10.1109/HICSS.2014.231.

[108] K. N. Murthy ve P. Scholar, “Word Cloud In Python”, c. 24, sy 01, 2020.

[109] K. Welbers, W. Van Atteveldt, ve K. Benoit, “Text Analysis in R”, Commun. Methods Meas., c. 11, sy 4, ss. 245-265, Eki. 2017, doi: 10.1080/19312458.2017.1387238.

[110] G. Salton ve C. Buckley, “Term-weighting approaches in automatic text retrieval”, Inf. Process. Manag., c. 24, sy 5, ss. 513-523, Oca. 1988, doi: 10.1016/0306-4573(88)90021-0.

[111] C. H. F. 2022Time to read: 10 minutes, “Implementing logistic regression from scratch in Python”, IBM Developer. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://developer.ibm.com/articles/implementing-logistic-regression-from-scratch-in-python/

[112] “Support Vector Machine(Destek Vektör Makinesi) | by Ali Öksüz | Medium”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://alioksuz.medium.com/support-vector-machine-destek-vekt%C3%B6r-makinesi-24ac75fc65bf

[113] “Python ile Sınıflandırma Analizleri – Rastgele Orman (Random Forest) Algoritması – Miraç ÖZTÜRK”. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://miracozturk.com/python-ile-siniflandirma-analizleri-rastgele-orman-random-forest-algoritmasi/

[114] "Structure of the XGBoost model", ResearchGate. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.researchgate.net/figure/Structure-of-the-XGBoost-model_fig3_366528085

[115] "Fast Gradient Boosting with CatBoost - Fritz ai". Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://fritz.ai/fast-gradient-boosting-with-catboost/

[116] M. Waheed, H. Afzal, ve K. Mehmood, "NT-FDS—A Noise Tolerant Fall Detection System Using Deep Learning on Wearable Devices", Sensors, c. 21, sy 6, Art. sy 6, Oca. 2021, doi: 10.3390/s21062006.

[117] "8.4. Yinelemeli Sinir Ağları — Derin Öğrenmeye Dalış 0.17.5 documentation". Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://tr.d2l.ai/chapter_recurrent-neural-networks/rnn.html

[118] "Long Short-Term Memory | Neural Computation | MIT Press". Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://direct.mit.edu/neco/article-abstract/9/8/1735/6109/Long-Short-Term-Memory?redirectedFrom=fulltext

[119] "LSTM Ağlarını Anlamak by cbarkinozer | Medium". Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://cbarkinozer.medium.com/lstm-a%C4%9Flar%C4%B1n%C4%B1-anlamak-edb5ee62ea11

[120] "9.2. Uzun Ömürlü Kısa-Dönem Belleği (LSTM) — Derin Öğrenmeye Dalış 0.17.5 documentation". Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://tr.d2l.ai/chapter_recurrent-modern/lstm.html

[121] B. C. Mateus, M. Mendes, J. T. Farinha, R. Assis, ve A. M. Cardoso, "Comparing LSTM and GRU Models to Predict the Condition of a Pulp Paper Press", Energies, c. 14, sy 21, Art. sy 21, Oca. 2021, doi: 10.3390/en14216958.

[122] Anishnama, "Understanding Gated Recurrent Unit (GRU) in Deep Learning", Medium. Erişim: 09 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://medium.com/@anishnama20/understanding-gated-recurrent-unit-gru-in-deep-learning-2e54923f3e2

[123] Y. Bengio, A. Courville, ve P. Vincent, "Representation Learning: A Review and New Perspectives", 23 Nisan 2014, arXiv: arXiv:1206.5538. doi: 10.48550/arXiv.1206.5538.

[124] S. J. Pan ve Q. Yang, "A Survey on Transfer Learning", IEEE Trans. Knowl. Data Eng., c. 22, sy 10, ss. 1345-1359, Eki. 2010, doi: 10.1109/TKDE.2009.191.

[125] "14.8. Dönüştürücülerden Çift Yönlü Kodlayıcı Temsiller (BERT) — Derin Öğrenmeye Dalış 0.17.5 documentation". Erişim: 10 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://tr.d2l.ai/chapter_natural-language-processing-pretraining/bert.html

[126] "15.7. Doğal Dil Çıkarımı: BERT İnce Ayarı — Derin Öğrenmeye Dalış 0.17.5 documentation". Erişim: 10 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://tr.d2l.ai/chapter_natural-language-processing-applications/natural-language-inference-bert.html

[127] F. Özger, "Makine Öğrenmesi Algoritmalarının Hibrit Yaklaşımı İle Ağ Anomalisi Tespiti".

[128] N. I. DeliBalta, "Zaman Serileri Tahmininde Melez Bir Yaklaşım", 2023.

[129] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, ve R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", J. Mach. Learn. Res., c. 15, sy 56, ss. 1929-1958, 2014.

[130] A. Y. Ng, "Feature selection, L1 vs. L2 regularization, and rotational invariance", içinde Proceedings of the twenty-first international conference on Machine learning, içinde ICML '04. New York, NY, USA: Association for Computing Machinery, Tem. 2004, s. 78. doi: 10.1145/1015330.1015435.

[131] J. Wei ve K. Zou, "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks", 25 Ağustos 2019, arXiv: arXiv:1901.11196. doi: 10.48550/arXiv.1901.11196.

[132] R. Caruana, S. Lawrence, ve C. Giles, “Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping”, içinde Advances in Neural Information Processing Systems, MIT Press, 2000. Erişim: 10 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://proceedings.neurips.cc/paper_files/paper/2000/hash/059fdcd96baeb75112f09fa1dcc740cc-Abstract.html

[133] S. Ioffe ve C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”, 02 Mart 2015, arXiv: arXiv:1502.03167. doi: 10.48550/arXiv.1502.03167.

[134] “Deep Learning - Ian Goodfellow, Yoshua Bengio, Aaron Courville - Google Kitaplar”. Erişim: 10 Kasım 2024. [Çevrimiçi]. Erişim adresi: https://www.deeplearningbook.org/

[135] R. Kohavi, “A study of cross-validation and bootstrap for accuracy estimation and model selection”, içinde Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2, içinde IJCAI’95. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., Ağu. 1995, ss. 1137-1143.

ÖZGEÇMİŞ

Ahmet Tunahan KORKMAZ

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

AKADEMİK FAALİYETLER

Bildiriler:

1. A.T. Korkmaz, M. Ulaş, M. Karabatak ve Y. Santur, “Data Mining in Finance: Concepts, Trend, and Applications”, *3rd ICAENS*, Tem. 2022, ss. 1-15. doi: 10.362290499.