T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

İNSAN VEYA MAKİNE TARAFINDAN YAZILAN METİNLERİN DOĞAL DİL İŞLEME YÖNTEMLERİ İLE TESPİTİ

Yüksek Lisans Tezi

Tezi Hazırlayan
Merve YÜCE

İstanbul, 2025

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

İNSAN VEYA MAKİNE TARAFINDAN YAZILAN METİNLERİN DOĞAL DİL İŞLEME YÖNTEMLERİ İLE TESPİTİ

Yüksek Lisans Tezi

Tezi Hazırlayan
Merve YÜCE

Öğrenci No
2220003021

ORCID ID
0009-0007-4399-3881

Danışman
Doç. Dr. Atınç YILMAZ

İstanbul, 2025

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum **"İnsan veya Makine Tarafından Yazılan Metinlerin Doğal Dil İşleme Yöntemleri ile Tespiti"** başlıklı bu çalışmanın, bilimsel ahlak ve geleneklere uygun şekilde tarafımdan yazıldığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, yararlandığım eserlerin tamamının kaynaklarda gösterildiğini ve çalışmanın içinde kullanıldıkları her yerde bunlara atıf yapıldığını, patent ve telif haklarını ihlal edici bir davranışımın olmadığını belirtir ve bunu onurumla doğrularım. 22/01/2025

Merve YÜCE

T.C.
İSTANBUL BEYKENT ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜ
TEZLİ YÜKSEK LİSANS SINAV TUTANAĞI

22/01/2025

Enstitümüz ***Bilgisayar Mühendisliği*** Anabilim Dalı ***Bilgisayar Mühendisliği*** Programı yüksek lisans öğrencilerinden ***2220003021*** numaralı ***Merve YÜCE'nin*** *"İstanbul Beykent Üniversitesi Lisansüstü Eğitim – Öğretim Yönetmeliği"*nin ilgili maddesine göre hazırlayarak, Enstitümüze teslim ettiği ***"İnsan veya Makine Tarafından Yazılan Metinlerin Doğal Dil İşleme Yöntemleri ile Tespiti"*** konulu tezini, Yönetim Kurulumuzun 14/01/2025 tarih ve 2025/02 sayılı toplantısında seçilen ve On-Line toplanan biz jüri üyeleri huzurunda, İstanbul Beykent Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 29. maddesinin 3. fıkrası gereğince Zoom programı aracılığıyla on-line olarak aday tarafından savunulmuş ve sonuçta adayın tezi hakkında ***"OYBİRLİĞİ"*** ile ***"KABUL"*** kararı verilmiştir.

İşbu tutanak, 1 nüsha olarak hazırlanmış ve Enstitü Müdürlüğü'ne sunulmak üzere tarafımızdan düzenlenmiştir.

DANIŞMAN
Doç. Dr. At*** YI***
(İstanbul Beykent Üniversitesi)

ÜYE
Dr. Öğr. Üyesi Ta*** Fİ***
(İstanbul Beykent Üniversitesi)

ÜYE
Dr. Öğr. Üyesi So*** SE***
(Kütahya Dumlupınar Üniversitesi)

Adı ve Soyadı : Merve YÜCE

Danışmanı : Doç. Dr. Atınç YILMAZ

Derecesi ve Tarihi : Yüksek Lisans (Tezli), 2025

Alanı : Bilgisayar Mühendisliği

Anahtar Kelimeler : Doğal Dil İşleme (DDİ), Makine Öğrenmesi, Yapay Zeka

ÖZ

İNSAN VEYA MAKİNE TARAFINDAN YAZILAN METİNLERİN DOĞAL DİL İŞLEME YÖNTEMLERİ İLE TESPİTİ

Bu çalışma, insan ve yapay zekâ tarafından yazılmış metinlerin ayrımını yapmayı amaçlayan, doğal dil işleme (NLP) teknikleri ve makine öğrenmesi modellerine dayalı bir yöntem geliştirmeyi hedeflemiştir. Araştırmada, farklı kaynaklardan elde edilen insan ve yapay zekâ üretimi metinler kullanılmış, bu metinler üzerinde kapsamlı veri işleme adımları gerçekleştirilmiştir. İlk olarak, metinler ön işleme sürecine tabi tutulmuş, gereksiz kelimeler ve semboller temizlenmiş, ardından metinler tokenize edilerek Word2Vec algoritması ile kelime vektörlerine dönüştürülmüştür. Bu süreçte, elde edilen vektörler, insan ve makine yazımı metinler arasındaki farkları sınıflandırmak amacıyla SVM ve LSTM modelleriyle işlenmiştir. Model performansını artırmak için genetik algoritmalar gibi sezgisel yöntemlerle en etkili özellikler seçilmiş, bu sayede işlem maliyeti azaltılarak sınıflandırma doğruluğu optimize edilmiştir. Geliştirilen hibrit model, başlangıçta kullanılan tüm özellikleri daha etkili bir alt kümeye indirgemiş ve yeniden eğitilmiştir. Sonuç olarak, çalışma doğruluk oranı, ROC eğrisi ve precision-recall analizleri gibi performans ölçümleri üzerinden yüksek başarı elde etmiş ve geliştirilen yöntemlerin etkinliğini ortaya koymuştur. Bu araştırma, insan ve yapay zekâ yazımı metinlerin tespiti için ileri düzey doğal dil işleme tekniklerinin ve makine öğrenmesi modellerinin etkili bir şekilde uygulanabileceğini göstermiştir. Elde edilen bulgular, bu alandaki gelecekteki çalışmalar için değerli bir kaynak ve referans oluşturmaktadır.

Name and Surname : Merve YÜCE
Supervisor : Assoc. Prof. Dr. Atınç YILMAZ
Degree and Date : Master's (Thesis), 2025
Major : Computer Engineering
Keywords : Natural Language Processing (NLP), Machine Learning, Artificial Intelligence

ABSTRACT

DETECTION OF TEXTS WRITTEN BY HUMAN OR MACHINE WITH NATURAL LANGUAGE PROCESSING METHODS

This study aims to develop a method based on natural language processing (NLP) techniques and machine learning models to distinguish between human-written and AI-generated texts. The research utilized datasets consisting of human and AI-generated texts obtained from various sources and implemented comprehensive data processing steps. Initially, the texts underwent preprocessing, where irrelevant words and symbols were removed, and the texts were tokenized and converted into word vectors using the Word2Vec algorithm. The resulting vectors were analyzed using SVM and LSTM models to classify the differences between human-written and machine-generated texts. To enhance model performance, heuristic methods such as genetic algorithms were employed for feature selection, allowing for the reduction of computational costs while optimizing classification accuracy. The developed hybrid model reduced the initial feature set to a more effective subset and was retrained accordingly. As a result, the study achieved high performance in terms of accuracy, ROC curves, and precision-recall analyses, demonstrating the effectiveness of the proposed methods. This research highlights the potential of advanced natural language processing techniques and machine learning models in detecting human and AI-generated texts. The findings provide valuable insights and a solid foundation for future studies in this domain.

İÇİNDEKİLER

Sayfa No.

3. UYGULAMA

TABLOLAR LİSTESİ

Sayfa No.

ŞEKİLLER LİSTESİ

Sayfa No.

KISALTMALAR

AI : Artificial Intelligence (Yapay Zekâ)

CBOW : Continuous Bag of Words

DDİ : Doğal Dil İşleme

GPT : Generative Pre-trained Transformer (Üretken Ön İşlemeli Dönüştürücü)

LLM : Large Language Model (Büyük Dil Modeli)

LoRA : Low-Rank Adaptation of Large Language Models (Büyük Dil Modellerinin Düşük Dereceli Uyarlaması)

LSTM : Long-Short-Term Memory (Uzun Kısa Süreli Bellek)

ML : Machine Learning (Makine Öğrenmesi)

NLG : Natural Language Generation (Doğal Dil Üretimi)

NLP : Natural Language Programming (Doğal Dil İşleme)

OCR : Optical Character Recognition (Optik Karakte Tanıma)

PCA : Principal Component Analysis (Temel Bileşen Analizi)

RBF : Radial Basis Function (Radyal Tabanlı Fonksiyon)

RFE : Recursive Feature Elimination (Özyinelemeli Özellik Eleme)

RNN : Recurrent Neural Networks (Yinelemeli Sinir Ağları)

ROC : Receiver Operating Characteristic

SVM : Support Vector Machine (Destek Vektör Makinesi)

TF- IDF : Term Frequency-Inverse Document Frequency (Terim Frekansı – Ters Belge Frekansı)

t-SNE : t-Distributed Stochastic Neighbor Embedding (t – Dağıtılmış Stokastik Komşu Gömme)

vb. : ve benzeri

vd. : ve diğerleri

YZ : Yapay Zeka

SÖZLÜK

Lemmatization: Bir kelimenin kök veya temel formuna dönüştürülmesini ifade eder.

Stemming: Kelimeleri dilbilgisel varyasyonlarına ve eklerine rağmen ortak bir kök veya temel formda birleştirme işlemidir.

Stopwords: Sık kullanılan kelimeler

Stilometrik Analiz: Bir metnin dilsel ve yapısal özelliklerini sayısal verilerle inceleyerek yazarın üslubunu belirlemeye yönelik yapılan bir yöntemdir.

GİRİŞ

Yapay zekâ teknolojilerinin hızla büyüdüğü ve dijital içerik üretiminin arttığı bir dönemde, metinlerin otomatik olarak insanlar veya makineler tarafından yazıldığını tespit etmek giderek önemli bir hale gelmektedir. Büyük miktarda verinin kullanılabilirliği ve sürekli artan hesaplama gücü ile Yapay Zeka, karmaşık sorunları çözme ve çeşitli endüstrilerde tahminlerde bulunma konusunda önemli bir araç haline gelmiştir. Bu çalışma, metinlerin insanlar ve makineler tarafından yazıldığını doğal dil işleme yöntemleri ile tespit etmek için bir adım atmaktadır. Bu bağlamda, metinlerin bir yapay zekâ tarafından üretilip üretilmediğini belirleme konusu, birçok sektörde ve uygulamada derin etkiler oluşturmaktadır. Özellikle sosyal medya platformları ve haber sitelerinde kullanıcıların karşısına çıkan metin içeriklerinin gerçekliğini doğrulamak için doğal dil işleme araçlarına ihtiyaç duyulmaktadır. Bu tez kapsamında geliştirilecek metodoloji, metinlerin dilbilgisel özellikleri, kelime dağılımları, anlam düzeyleri ve kullanılan ifadeler üzerinden gerçekleştirilen detaylı analizlerle insan ve makine kaynaklı metinleri ayırt etmeyi hedeflemektedir.

İlk paragrafta belirtildiği gibi, metinlerin otomatik olarak insanlar veya makineler tarafından yazıldığını tespit etme konusu, doğal dil işleme ve yapay zekâ alanlarında önemli bir odak noktasıdır. Bu noktada, NLP ve yapay zekanın bu konudaki rolünü daha ayrıntılı bir şekilde ele alarak, sektördeki gelişmeleri anlamak önemlidir.

Doğal dil işleme, bilgisayar sistemlerinin insan dilini yorumlamasına, işlemesine ve anlamasına olanak sağlayan makine öğrenmesinin bir alt teknolojisidir. Bu teknoloji, metin madenciliği, duygu analizi, dil modelleme ve metin üretme gibi bir dizi alt alanlarda kullanılmaktadır. Yapay zekanın bir parçası olarak NLP, metinlerin içeriğini anlama, çıkarım yapma ve metin tabanlı etkileşimlerde bulunma kapasitesine sahiptir. Bu şekilde NLP, dilin farklı yönlerini detaylı bir şekilde ele alır. Dilbilgisi düzensizlikleri, lehçe farklılıkları gibi dilin karmaşıklıklarını anlamak doğal dil işlemenin öne çıkan özelliklerindendir.

İş dünyasında, şirketler NLP'nin sunmuş olduğu bu yetenekleri kullanarak bir dizi otomatize görevi başarılı bir şekilde gerçekleştirebilir. Bu görevler arasında, büyük belgelerin işlenmesi, analizi ve arşivlenmesi, müşteri geribildirimleri veya çağrı merkezi

kayıtlarının etkili bir şekilde analizi, otomatik müşteri hizmeti sunmak için chatbot kullanımı, metinleri sınıflandırma ve önemli bilgileri çıkarma gibi işlemler yer alabilir.

NLP'nin müşterilere odaklı uygulamalara entegre edilmesi, etkileşimleri daha anlamlı hale getirmek için önemli bir adımdır. Örneğin, chatbotlar müşteri sorgularını analiz edip sıralayabilir, sıkça sorulan sorulara otomatik yanıt verebilir ve karmaşık sorguları doğrudan müşteri desteğine yönlendirerek süreci optimize edebilir. Ayrıca NLP destekli uygulamaların kullanımıyla birlikte şirketler, müşteri taleplerine daha hızlı ve etkili bir şekilde yanıt verme, büyük veri setlerini analiz etme ve müşteri geri bildirimlerini daha etkili bir şekilde değerlendirme konularında rekabet avantajı elde edebilirler.

İşletmeler için NLP kullanım örnekleri oldukça çeşitlidir ve bu teknolojinin sunmuş olduğu avantajlar birçok sektörde değerlendirilebilmektedir.

Müşteri Hizmetleri ve Chatbotlar: Şirketler, müşteri hizmetlerini iyileştirmek ve maliyetleri düşürmek için NLP tabanlı chatbotları kullanbilirler. Chatbotlar, müşteri sorularını anlamak, yanıtlamak ve karmaşık sorguları çözmek için NLP algoritmalarını kullanabilir.

Müşteri Geri Bildirim Analizi: NLP, müşteri geri bildirimlerini ve incelemelerini otomatik olarak analiz edebilir. Bu sayede, müşteri memnuniyeti ve ürün performansı gibi değerli bilgiler elde edilebilir.

Sosyal Medya İzleme: NLP, sosyal media platformlarında şirketle ilgili konuşmaları izleyebilir ve duygu analizi yaparak kullanıcıların duygularını ve hissiyatını anlayabilir. Bu, marka itibarı yönetimi ve pazarlama stratejileri için önemlidir.

İş Zekası ve Veri Analizi: İşletmeler, büyük miktarda metin verisini analiz ederek pazar eğilimlerini, müşteri davranışlarını ve rekabetçi konumlarını anlamak için NLP'yi kullanabilirler.

Belge İşleme ve Arşivleme: NLP, büyük belge arşivlerini otomatik olarak indeksleyebilir be kategorize edebilir. Bu, arama ve erişim süreçlerini hızlandırır.

İnsan Kaynakları ve Yetenek Yönetimi: NLP, Özgeçmişleri analiz ederek uygun adayları belirlemekte ve işe alım süreçlerini iyileştirmekte kullanılabilir. Ayrıca, çalışan memnuniyetini ve performansını değerlendirmede de etkili olabilir.

Finansla Analiz ve Haber İzleme: Finans sektörü, NLP'yi finansal raporları analiz etmek, piyasa haberlerini takip etmek ve finansal tahminlerde bulunmak için kullanılabilir.

Eğitim ve Öğrenim: Eğitim kurumları, öğrenci performansını değerlendirmek, öğrenci sorularını yanıtlamak ve öğrenci geribildirimlerini analiz etmek için NLP'yi kullanabilirler.

Bu örnekler, NLP'nin işletmeler için geniş bir uygulama yelpazesi olduğunu göstermektedir. Teknolojinin kullanımı, iş süreçlerini daha verimli hale getirmek, müşteri memnuniyetini artırmak ve rekabet avantajı elde etmek adına önemli bir araçtır.

Doğal dil işlemenin günlük hayatta da birçok kullanım alanı bulunmaktadır ve bu teknolojinin önemi giderek artmaktadır. NLP'nin günlük hayatta kullanım alanlarına mobil asistanlar ve sanal yardımcılar, metin ve duygu analizi, otomatik dil çevirisi, e-posta filtreleme gibi örnekler verilebilir.

Mobil Asistanlar ve Sanal Yardımcılar: Siri, Google Assistant ve Amazon Alexa gibi mobil asistanlar, NLP sayesinde konuşma anlayışı geliştirebilmekte ve kullanıcılara daha etkili bir şekilde hizmet verebilmektedir.

Metin ve Duygu Analizi: Sosyal medya platformları ve müşteri geri bildirimleri üzerinden yapılan metin analizi, NLP'nin duygu analizi özellikleri sayesinde insanların duygularını anlamak için kullanılmaktadır.

Otomatik Dil Çevirisi: NLP, metinlerin otomatik olarak bir dilden diğerine çevrilmesine olanak tanımaktadır.

E-posta Filtreleme: Spam filtreleri, NLP kullanarak istenmeyen e-postaları tespit edebilmekte ve kullanıcıların gelen kutularını temiz tutabilmektedir.

Geleneksel yöntemlerle insanlar ve makineler tarafından yazılan metinlerin ayırt edilmesi oldukça zorlayıcı bir süreçtir. İnsanlar tarafından yazılan metinler genellikle duygusal tonlar, tutarsızlıklar ve özgün dil kullanımı gibi özellikler taşırken, yapay zeka tarafından üretilen metinler daha yapısal, belirli kurallara dayalı ve tekrarlayan kalıplarla oluşturulmuş olabilir. Ancak son yıllarda geliştirilen ileri seviye dil modelleri bu farkları minimuma indirerek insan benzeri metinler üretme konusunda büyük başarılar elde etmiştir. Bu durum, sosyal medya içeriklerinden haber metinlerine, akademik

çalışmalardan müşteri hizmetleri yazışmalarına kadar birçok alanda karmaşa oluşturmakta ve bu tür metinlerin kökenini tespit etmek zorlaşmaktadır. Çalışmanın ana problemi, farklı NLP teknikleri ve makine öğrenmesi algoritmaları kullanılarak, insan ve makine tarafından yazılan metinlerin etkin bir şekilde nasıl ayrıştırılabileceği konusudur.

Bu çalışmanın amacı, doğal dil işleme ve makine öğrenmesi alanlarında, insan ve makine tarafından yazılan metinlerin tespit konusunda önemli bir boşluğu doldurmayı hedeflemektedir. Özellikle yanlış bilgilendirme, spam içerikler ve sahte haberlerin tespit edilmesi gibi kritik uygulamalarda bu tür bir ayrıştırmanın önemi büyüktür. Ayrıca bu çalışma ile gerçekleştirilecek model, sadece sosyal medya ve haber sitelerinde değil, akademik araştırmalarda, finansal raporlamalarda ve müşteri hizmetlerinde de kullanılabilir. Böylece hem bireyler hem de kurumlar, karşılaştıkları metinlerin kökenini daha iyi anlayabilir ve bu bilgiler doğrultusunda daha bilinçli kararlar alabilirler.

Çalışmada öncelikle, insan ve makine tarafından yazılan metinlerin dilsel ve anlamsal özellikleri incelenecektir. Bu özelliklerin belirlenmesi aşamasında Word2Vec gibi kelime gömme yöntemleri kullanılacak ve metinlerin anlamsal temsilleri çıkarılacaktır. Ardından, LSTM gibi derin öğrenme teknikleri ile metinlerin ardışık yapısı ve anlam düzeyleri analiz edilecektir. SVM gibi makine öğrenmesi algoritmaları ise metinlerin sınıflandırılması ve ayırt edilmesi için kullanılacaktır. Elde edilen sonuçlar, farklı yöntemlerin kombinasyonu ile oluşturulacak hibrit bir model ile birleştirilecek ve insan ve makine kaynaklı metinlerin tespiti konusunda en yüksek doğruluğu sağlamak amacıyla optimize edilecektir.

1. LİTERATÜR TARAMASI

Bu bölümde literatür taraması ile ilgili açıklama yapılacaktır.

1.1. Doğal Dil İşleme Teknikleri ve Algoritmaları

Doğal dil işleme (NLP) ve makine öğrenmesi yöntemleri, insan ve makine tarafından üretilen metinlerin tespiti ve analizi konusunda son yıllarda önemli ilerlemeler kaydetmiştir. Bu alandaki çalışmalar, metinlerin içerik, dilbilgisi ve anlamsal özelliklerine dayanarak, metinlerin kaynağını belirlemeye odaklanmıştır. Literatürde, metinlerin insan veya makine tarafından yazıldığını tespit etmek için kullanılan yöntemler arasında dil modelleri, öznitelik çıkarımı teknikleri ve sınıflandırma algoritmaları bulunmaktadır. Bu bölümde, mevcut çalışmalar ve kullanılan yöntemler incelenerek, insan ve makine kaynaklı metinlerin tespiti konusunda yapılan araştırmaların genel bir çerçevesi sunulacaktır.

Metinlerin insan ya da makine tarafından üretildiğini tespit etme konusunda yapılan çalışmaların artması, bu tür teknolojilerin potansiyel suistimallerine karşı alınabilecek önlemleri de beraberinde getirmiştir. Crothers ve arkadaşları (2023) tarafından yapılan çalışmada, günümüzdeki doğal dil üretim (NLG) sistemlerinin sunduğu büyük olanakların yanı sıra, bu sistemlerin kötüye kullanımına yönelik tehdit modelleri ayrıntılı bir şekilde incelenmiştir. Özellikle, ChatGPT gibi güçlü açık kaynak modellerin kolayca erişilebilir hale gelmesi ve kullanıcı dostu araçlarla bu tür modellere erişimin yaygınlaşması, makine tarafından üretilen metinlerin tespit edilmesini daha da zorlaştırmıştır. Bu çalışmada, makine tarafından üretilen metinlerin tespiti, bu tür kötüye kullanımları azaltmak için önemli bir karşı önlem olarak ele alınmıştır. Ancak, bu tespitin birçok teknik zorluğu ve açık problemleri beraberinde getirdiği vurgulanmıştır. Çalışma, mevcut NLG sistemlerinin sosyal ve siber güvenlik bağlamında yarattığı tehditleri kapsamlı bir şekilde analiz etmiş ve makine kaynaklı metin tespit yöntemlerini detaylı bir şekilde gözden geçirmiştir. Ayrıca, gelecekteki çalışmalar için kritik tehdit modellerine yönelik güçlü öneriler sunulmuştur (Crothers, Japkowicz ve Viktor, 2023). Bu bağlamda, tespit sistemlerinin güvenilirlik, dayanıklılık ve hesap verebilirlik gibi kriterleri karşılamasının önemine vurgu yapılmıştır.

Ali Dayan (2022), Doğal Dil İşleme ile Makinelerin Kendi Dilini Modellemesi Yüksek Lisans tezinde yaptığı çalışmada; makinelerin kendi dillerini üretebilmesi için Yinelemeli Yapay Sinir Ağları, Mel Frekans Cepstral Katsayısı ve Dinamik Zaman Çözgü gibi teknikleri kullanarak, makinelerin insana benzer şekilde kendi dillerini oluşturabilme süreçlerine katkıda bulunmuştur ve bu süreçlerin doğal dillerin oluşumu ile ilgili dil bilimcilere yol gösterebileceğini öne sürmüştür.

Recep Ali Ay (2021), Doğal Dil İşleme ve Kümeleme Yöntemleri ile Müşteri Yorumlarının İncelenmesi Yüksek Lisans tezinde yaptığı çalışmada; Türkiye'deki otellere ait müşteri yorumlarını içeren veri seti, doğal dil işleme teknikleri ve kümeleme algoritmaları kullanılarak analiz edilmiştir. Özellikle DBSCAN algoritması kullanılarak müşteri yorumları kümelenmiş ve yorumlar arasındaki benzerlikler tespit edilmiştir. Bu çalışma sonucunda gelecekte farklı kümeleme algoritmaları ile yapılacak çalışmalarda müşteri yorumlarının benzersizliği ve güvenirliği hakkında daha net fikirler edinilebileceği belirtilmiştir.

Betül Güvenç (2013), Machine Learning Methods In Natural Language Processing (Doğal Dil İşlemede Makine Öğrenme Yöntemleri) Yüksek Lisans tezinde yaptığı çalışamada; farklı metin tipleri üzerinde tekli ve çoklu metin özetlemede kullanılan LexRank ve TextRank gibi grafik tabanlı algoritmaların etkinliğini araştırmış, Word2Vec ve PageRank algoritmalarını kullanarak otomatik anahtar kelime çıkarma konusunda yeni ve etkili bir yöntem önermiştir. Çalışmasında, özetleme algoritmalarının haber metinlerinde en iyi sonuçları verdiğini, ancak kısa öykülerde daha az verimli sonuçlar elde ettiğini belirtmiştir.

Özge Sevgili Ergüven (2018), A Systematic Evaluation Of Semantic Representations In Natural Language Processing (Doğal Dil İşlemede Semantik Gösterimlerin Sistematik Değerlendirmesi) Yüksek Lisans tezinde yaptığı çalışamada; kelime gösterimlerinde önceki istatistiksel yöntemler ile günümüz öğrenme tabanlı yaklaşımlarını karşılaştırmış ve bu iki yaklaşım arasındaki ilişkiyi incelemiştir. Çalışmasında, çok anlamlı kelimelerin her bir anlamı için ayrı gösterimler sağlayan yeni bir yöntem geliştirerek, kelime gösterimlerinin Doğal Dil İşleme problemlerindeki uygulanabilirliğini değerlendirmiştir.

Abdul Majeed Issifu (2022), Doğal Dil İşlemede Denetimli Makine Öğrenimi Modellerinin Performansını Arttırmak İçin Veri Zenginleştirme Yöntemlerinin Geliştirilmesi Yüksek Lisans tezinde yaptığı çalışmada; 'Basit Veri Zenginleştirme' (Easy Data Augmentation) yöntemlerini genişleterek, biyomedikal alanda Adlandırılmış Varlık Tanıma görevleri için uygulanabilirliğini incelemiştir. Bu çalışma kapsamında, BERT ve BioBERT gibi dönüştürücü modeller üzerinde deneyler yaparak, veri zenginleştirme yöntemlerinin bu modellerin performansını nasıl artırdığını göstermiş ve özellikle BC5CDR-hastalık veri setinde %5.95 ve %8.49 oranında F1 puanı artışı sağlamıştır.

Sosyal medya platformlarının yaygınlaşması, bilginin hızla paylaşılmasını sağlamış, ancak bu durum sahte haberlerin de hızla yayılmasına zemin hazırlamıştır. Süleyman G. Taşkın, Ecir U. Küçüksille ve Kamil Topal (2021) tarafından yapılan bir çalışmada, Türkçe dilinde Twitter üzerinde sahte haberlerin tespiti için denetimli ve denetimsiz makine öğrenmesi algoritmaları incelenmiştir. Bu çalışmada K-ortalamalar, Negatif Olmayan Matris Çarpımı ve Doğrusal Diskriminant Analizi gibi denetimsiz öğrenme algoritmaları ile K-En Yakın Komşu, Destek Vektör Makineleri ve Rassal Orman algoritmaları gibi denetimli öğrenme algoritmalarının performansları karşılaştırılmıştır. Elde edilen bulgular, denetimli öğrenme algoritmalarının sahte haber tespitinde daha yüksek bir başarıya sahip olduğunu göstermektedir. Özellikle, denetimli öğrenme algoritmaları 0.86 F1-metrik değeriyle dikkat çekerken, denetimsiz öğrenme algoritmalarının F1-metrik değeri 0.72 de kalmıştır. Bu çalışma, sahte haberlerin tespitinde makine öğrenmesi algoritmalarının etkili bir şekilde kullanılabileceğini göstermektedir.

1.2. Doğal Dil İşlemede Gelişmeler ve Uygulama Alanları

Literatür taraması sonucunda doğal dil işleme yöntemleri üzerine çeşitli alanlarda yoğun çalışmaların yapıldığı gözlemlenmiştir. Bu çalışmalar, makinelerin dil modelleme, metin özetleme, anahtar kelime çıkarma ve semantik gösterim gibi görevlerde daha etkili olabilmeleri için farklı yaklaşımlar kullanmıştır. Yapılan araştırmalarda, denetimli ve denetimsiz makine öğrenimi algoritmalarının yanı sıra, kelime temsilleri ve kümeleme tekniklerinin uygulanmasıyla çeşitli veri türleri üzerinde önemli başarılar elde edilmiştir. Ayrıca, sahte haber tespiti, müşteri yorumlarının analizi ve biyomedikal alanlarda adlandırılmış varlık tanıma gibi uygulama alanlarında doğal dil işlemenin başarısının

arttığı görülmektedir. Çalışmalar hem geleneksel istatistiksel yöntemlerin hem de modern öğrenme tabanlı modellerin NLP problemlerine çözüm üretmede etkili olduğunu ortaya koymuştur.

Makine tarafından üretilen metinlerin tespiti, özellikle sahte haberlerin önlenmesi, sosyal medya içeriklerinin doğrulanması ve eğitim alanında intihalin engellenmesi gibi kritik alanlarda önem taşımaktadır. Yapılan çalışmalar, dil modellerinin geliştirilmesiyle birlikte makine tarafından oluşturulan içeriklerin daha doğal ve insan metinlerine benzer yapıya büründüğünü göstermektedir. Bu durum, tespit algoritmalarını zorlaştırsa da, Transformer tabanlı modellerin performansını artırarak tespit süreçlerini iyileştirme fırsatları sunmaktadır. Örneğin, GPT-2 ve GPT-3 gibi modellerin olasılık tabanlı eğilimleri incelenerek, içerikteki tahmin edilebilirlik düzeyine göre metinlerin kaynağının belirlenebileceği vurgulanmıştır.

Son dönemlerde, hibrit yaklaşımlar ve ince ayar (fine-tuning) yöntemleri ile daha etkili tespit sistemleri geliştirilmiştir. Bu çalışmalarda, birden fazla dil modeli ve makine öğrenmesi algoritması birleştirilerek, doğruluk oranları artırılmaktadır. LoRA gibi düşük maliyetli adaptasyon teknikleri, büyük modellerin hafifletilmesini sağlarken, RoBERTa (Liu vd., 2019) veya BERT (Devlin vd., 2019) gibi modellerin performans kaybı olmadan hızlı şekilde çalışabilmesine olanak tanımaktadır. Bu tür tekniklerin, sahte metinlerin yanı sıra farklı dillerdeki içeriklerde de başarılı sonuçlar verdiği ortaya konmuştur.

Ayrıca, topluluk öğrenme (ensemble learning) yöntemleriyle farklı modellerin çıktıları birleştirilmekte ve sonuçlar daha güvenilir hale getirilmektedir. Bu yaklaşımlar, içerikteki stilistik ve anlamsal özniteliklerin model performansını artırmada önemli bir rol oynadığını göstermektedir. Özellikle sahte haberlerin tespiti konusunda denetimli ve denetimsiz algoritmalar bir arada kullanılarak, sosyal medya platformlarında bilgi kirliliğinin azaltılması hedeflenmektedir. Bu yöntemler, veri çeşitliliği ve model karmaşıklığı arttıkça daha esnek çözümler sunma potansiyeline sahiptir.

Sonuç olarak, literatür taraması, makine tarafından üretilen metinlerin tespiti alanında büyük ilerlemeler sağlandığını ve mevcut tekniklerin farklı uygulama alanlarında başarıyla kullanıldığını ortaya koymaktadır. Hibrit yaklaşımlar, ince ayar teknikleri ve topluluk öğrenme yöntemleriyle geliştirilen sistemler, makine metinlerinin tespitinde güçlü çözümler sunmaktadır. Gelecekte çok dilli veri kümeleri üzerinde daha

genellenebilir tespit algoritmalarının geliştirilmesi, bu alandaki kritik araştırma alanlarından biri olmaya devam edecektir. Bu ilerlemeler, sosyal medya içeriklerinin doğrulanması, sahte haberlerle mücadele ve akademik sahtekarlığın önlenmesi gibi konularda önemli katkılar sunmaktadır.

2. MATERYALLER VE YÖNTEMLER

Bu bölümde, çalışmanın yürütülmesi sırasında kullanılan yöntemler ve uygulanan işlemler detaylı bir şekilde açıklanmıştır. Çalışmada, doğal dil işleme alanında kullanılan veri setleri, bu verilere uygulanan ön işleme adımları ve modelleme süreçleri ele alınmıştır. Ayrıca, kullanılan algoritmalar, özellik seçimi yöntemleri ve performans değerlendirme metrikleri hakkında bilgi verilmiştir. Uygulama kapsamında gerçekleştirilen tüm adımlar, çalışmanın hedeflerini gerçekleştirmek için sistematik bir şekilde planlanmış ve uygulanmıştır.

2.1. Yapay Zekâ

Son yıllarda, yapay zekâ hayatımızın her alanına dahil olmuş durumda ve çoğumuz bu teknolojilerin hayatımız üzerindeki etkisinin farkında bile değiliz. Günlük rutinlerimizde, fark etmesek bile yapay zekâ teknolojisi bize rehberlik etmektedir. Sabah uyandığımızda, ilk iş olarak cep telefonumuzu veya bilgisayarımızı kullanmaya başlıyoruz. Bu cihazlar, günlük planlamalarımızda, karar alma süreçlerimizde ve bilgiye erişimde artık vazgeçilmez hale geldi.

Cihazlarımızı açtığımız andan itibaren, yapay zekanın sağladığı pek çok işlevi kullanmaya başlıyoruz. Yüz tanıma teknolojisi, e-postalar, uygulamalar, sosyal medya, dijital sesli asistanlar, çevrimiçi bankacılık işlemleri, sürüş yardımları, alışveriş ve eğlence amaçlı platformlar vb. bu teknolojinin birer ürünleridir. Yapay zeka, artık kişisel ve profesyonel çevrimiçi hayatımızın her köşesine dokunmuş durumdadır. İş dünyasında küresel iletişim ve veri biliminden yararlanma süreci hızla önem kazanmaktadır ve bu teknolojinin büyüme potansiyeli sınırsız görünmektedir.

Günümüzde yapay zekâ, sıradan bir teknoloji olarak kabul edilmektedir; peki yapay zekâ aslında nedir ve nasıl ortaya çıktı?

Yapay zekâ, genellikle insan zekâsı gerektiren veya insan kapasitesinin ötesindeki verileri analiz edebilen, tahminde bulunabilen, öğrenebilen veya eyleme geçebilen bilgisayarlar ve makineler geliştirmeyi amaçlayan bilim dalıdır. Bilgisayar bilimi, veri analitiği, istatistik, donanım ve yazılım mühendisliği, dil bilimi, sinir bilimi gibi çeşitli disiplinleri kapsayan geniş bir alan olarak öne çıkmaktadır. Bu alan, aynı zamanda felsefe ve psikoloji gibi daha soyut bilim dallarını da içerir.

İş dünyasında yapay zekâ, operasyonel düzeyde genellikle makine öğrenimi ve derin öğrenmeye dayalı teknolojiler olarak kullanılır. Bu teknolojiler, veri analizi, tahmin ve öngörü, nesne sınıflandırma, doğal dil işleme, öneri sistemleri ve akıllı veri erişimi gibi birçok uygulamayı içerir.

Yapay zekanın temelleri, İkinci Dünya Savaşı sırasında Alan Turing'in Alman şifrelerini çözme çalışmalarıyla birlikte atılmıştır. Turing'in bu devrim niteliğindeki çalışmaları, bilgisayar bilimlerinin bazı temel prensiplerinin gelişmesine katkı sağlamıştır. 1950'li yıllara gelindiğinde, Turing makinelerin kendi başlarına düşünebileceği fikrini ortaya atmıştır. Bu radikal düşünce, makine öğreniminin problem çözmedeki artan rolü ile birleştiğinde, yapay zekâ alanında önemli ilerlemelere Zemin hazırlamıştır. Araştırmalar, makinelerin insanlara benzer şekilde nasıl:

- Düşünebileceği,
- Anlayabileceği,
- Öğrenebileceği,
- Ve kendi "zekasını" kullanarak sorunları çözebileceği

gibi konular üzerinde yoğunlaşmıştır. 1950'lerde Marvin Minsky ve John McCarthy gibi bilgisayar bilimciler, bu potansiyeli fark ederek Turing'in çalışmalarını daha da ileriye taşımışlardır (Yılmaz, 2015). 1956 yılında Dartmouth College, ABD'de düzenlenen bir çalışmada, bugün yapay zekâ alanı olarak bildiğimiz alanın temellerini atmıştır (Saygılı, 2020). Bu atölyeye katılan birçok bilim insanı, önümüzdeki on yıllarda yapay zekâ alanında lider ve yenilikçi isimler haline gelmiştir.

Alan Turing'in çığır açan araştırmalarının bir kanıtı olarak, günümüzde hala güncellenmiş versiyonuyla kullanılan Turing Testi, yapay zekâ araştırmalarında başarıyı ölçmek için önemli bir araç olarak kabul edilmektedir.

2.1.1. Turing Testi: İnsan Gibi Davranma Yaklaşımı

Alan Turing'in yapay zekâ alanındaki en önemli katkılarından biri, Turing Test'i olarak bilinen yaklaşımıdır. Bu test, makinelerin insan benzeri zekâ sergileyip sergileyemeyeceği konusunu belirlemek amacıyla geliştirilmiştir. Alan Turing, bir makinenin düşünme kapasitesine sahip olup olmadığını anlamak amacıyla, insanlarla makine arasında karşılıklı bir iletişim ortamı oluşturmayı önermiştir. Bu ortamda, bir

insan katılımcı, bir bilgisayar ve başka bir insanla yazılı olarak iletişim kurar. Eğer insan katılımcı, makinenin bir insan mı yoksa bir bilgisayar mı olduğunu ayırt edemezse, bu durumda makine 'başarılı' kabul edilir ve bu, onun insan benzeri düşünme yeteneğine sahip olduğuna işaret eder.

Turing Testi, yapay zekanın gelişiminde büyük bir dönüm noktası olmuştur. Turing'in bu yaklaşımı, makinelerin sadece önceden belirlenmiş görevleri yerine getiren cihazlar olmasının ötesine geçip, insan gibi düşünebilen varlıklar haline gelip gelemeyeceği sorusunu gündeme getirmiştir. Turing'in 1950'lerde ortaya koyduğu bu fikir, bilgisayar bilimi ve yapay zekâ araştırmaları için yeni bir bakış açısı geliştirmiştir. Testin temel amacı, zekâ kavramını soyut bir kavramdan çıkararak, ölçülebilir bir yapıya dönüştürmeyi hedeflemiştir.

Bu bağlamda, Turing Testi sadece teknik bir araç olmanın ötesinde, yapay zekanın felsefi boyutlarını da tartışmaya açmıştır. Makinenin "düşünmesi" veya "anlaması" gibi kavramlar, bilişsel bilim ve yapay zekâ alanında derin tartışmalara yol açmıştır. Turing, makine zekasını değerlendirirken bu tür metafizik sorulardan kaçınmış ve tamamen pratik bir yaklaşıma odaklanmıştır. Ona göre, bir makinenin zekâsı, insanın davranışını taklit edebilme kabiliyeti ile ölçülmeliydi. Bu, yapay zekanın bugünkü gelişiminde hala önemli bir dayanak noktasıdır.

Yapay zekâ sistemlerinin başarılı bir şekilde çalışabilmesi için farklı alt disiplinlerin bütünleşmiş bir şekilde çalışması gerekmektedir. Örneğin, doğal dil işleme (NLP), bir dilin anlamını ve yapısını analiz ederek bilgisayarların insanlar gibi dil anlayabilmelerini sağlamaktadır. Bu, makinelerin metinleri analiz etmesi, özetlemesi ve doğru cevaplar üretmesi için kritik bir beceridir. Bilgi temsili, bilgisayarların öğrenilen bilgiyi depolayabilmesi ve bu bilgiyi etkin bir şekilde çıkarabilmesi için önemlidir. Bu, makinenin öğrendiği verilerden anlam çıkarabilmesi ve bunları karar verme süreçlerinde kullanabilmesi anlamına gelmektedir.

Bunların yanı sıra, otomatik akıl yürütme, yapay zekanın bir problem çözme sürecinde mantıksal çıkarımlarda bulunabilmesini sağlamaktadır. Böylece makine, geçmiş deneyimlerinden edindiği bilgilerle yani problemleri çözebilir ve sonuçlar üretebilir. Makine öğrenmesi ise, yapay zekanın sürekli olarak kendini geliştirmesine

olanak tanımaktadır; bu sayede makine, yeni veri deneyimlere dayalı olarak performansını iyileştirir ve gelecekteki sorunlar için daha iyi tahminler yapabilir.

Turing Testi, bu temel bileşenlerin birlikte çalıştığı bir test ortamı sunmaktadır. Fiziksel temas gerektirmeyen bu test, bilgisayarların insan davranışlarını taklit etme yeteneğini değerlendirmektedir. Bilgisayar görüsü ve robotik gibi disiplinler de bu süreci desteklemektedir. Nesnelerin tanımlanması ve manipüle edilmesi, yapay zekanın çevresiyle daha etkili bir şekilde etkileşimde bulunmasına yardımcı olmaktadır. Turing Testi'nde, bir sorgulayıcı hem insana hem de yapay zekâ tarafından kontrol edilen bir makineye sorular yöneltmektedir. Bu süreçte, makine ve insan kendilerini insan olarak kanıtlamaya çalışırken, sorgulayıcı da kimin makine olduğunu belirlemeye çalışır.

2.1.2. Makine Öğrenmesi

Teknolojinin hızla ilerlemesi ile, bilgisayarların sadece programlanmış görevleri yerine getirmekle kalmayıp, aynı zamanda verilen öğrenme yeteneği kazandığı yeni bir döneme başladık. Bu dönemin temel taşlarından biri olan makine öğrenmesi, günlük yaşamımızın pek çok alanında devrim niteliğinde etkiler oluşturmuştur. Makine öğrenmesi sayesinde, bilgisayarlar karmaşık problemleri çözmek, tahminlerde bulunmak ve büyük veri kümelerinden anlamlı sonuçlar çıkarmak için kullanılmaktadır.

Öğrenme, bireyin ya da bir sistemin, deneyimler aracılığıyla bilgi edinme, bu bilgiyi kullanarak performansını geliştirme sürecidir. İnsanlar için öğrenme, çevrelerinden aldıkları bilgileri işleyerek bilinçli ya da bilinçsiz bir şekilde davranışlarını değiştirmeyi içermektedir. Benzer şekilde, makine öğrenmesi de verilerden öğrenme sürecini ifade etmektedir ve bu süreçte algoritmalar, verilen örneklerden hareketle kendi performanslarını iyileştirmektedir.

Makine Öğrenmesi (ML), bilgisayarların açıkca programlanmadan, veri ve deneyimlerden öğrenerek karar verebilme yeteneğine sahip olmasını sağlayan ve büyük veri kümeleri üzerinde çalışan ve bu verilerden öğrenen algoritmalar geliştirmeyi amaçlayan bir yapay zekâ disiplinidir. Bu teknik, bilgisayarların açık talimatlar olmaksızın görevleri yerine getirebilmesini sağlamak için veri modellerini analiz etmektedir ve elde ettiği bilgiyi kullanarak gelecekteki kararları veya tahminleri optimize etmektedir. Makine öğrenmesi, bankacılık, çevrimiçi alışveriş ve sosyal medya gibi pek

çok alanda etkin bir şekilde kullanılmaktadır ve kullanıcı deneyimlerini iyileştirmek için çeşitli algoritmalar devreye sokulmaktadır. Gelişen teknoloji ile, makine öğrenmesi, günlük hayatımızın önemli bir parçası haline gelmiştir.

Makine öğrenmesi süreçlerinde görevlerin başarıyla tamamlanabilmesi için belirli adımların izlenmesi gerekmektedir. Bu adımlar, verilerin toplanmasından modelin performansını artırmaya kadar geniş bir yelpazeyi kapsamaktadır. Herbir aşama, modelin genel başarısını doğrudan etkileyen kritik süreçler içermektedir. Aşağıda, bu adımların detaylandırılmış açıklamaları yer almaktadır.

Veri Toplama (Collecting Data): Makine öğrenmesi sürecinin ilk ve en önemli adımı, modelin eğitilmesi için gerekli olan verilerin toplanmasıdır. Veri, modelin öğrenilmesini sağlayan ana kaynaktır ve bu verilerin kalitesi, modelin nihai performansını doğrudan etkilemektedir. Veriler, genellikle veri tabanları, metin dosyaları, web siteleri ve sosyal medya gibi çeşitli kaynaklardan toplanabilir. Bu aşamada, toplanan verilerin model için anlamlı olmasına dikkat edilmelidir. Geçmiş verilerin toplanması, gelecekteki tahminler ve analizler için sağlam bir temel oluşturmaktadır ve modelin başarılı bir şekilde eğitilmesine olanak tanımaktadır.

Verilerin Hazırlanması (Preparing the Data): Toplanan ham veriler, genellikle modelin eğitimi için doğrudan kullanılamaz. Bu nedenle verilerin temizlenmesi, düzenlenmesi ve ön işlemeden geçirilmesi gerekmektedir. Bu süreç, verideki hataların düzeltilmesini, eksik verilerin tamamlanmasını ve modelin öğrenmesi için uygun hale getirilmesini içermektedir. Doğal dil işleme (NLP) gibi spesifik alanlarda, durma sözcüklerinin (stopwords) temizlenmesi, noktalama işaretlerinin çıkarılması ve veri setinin uygun formatta düzenlenmesi gibi işlemler yapılmaktadır. Verilerin doğru bir şekilde hazırlanması, modelin performansını artırır ve doğru sonuçlar elde edilmesine katkıda bulunur.

Modeli Eğitmek (Training a Model): Model eğitimi, makine öğrenmesi sürecinin en kritik aşamalarından biridir. Bu adımda, verilerin temsili için uygun bir model oluşturulur ve belirlenen algoritmalar aracılığıyla eğitim süreci başlatılır. Danışmanlı (supervised) öğrenme yöntemlerinde, veriseti eğitim ve test olarak ikiye ayrılır. Eğitim kümesi, modelin öğrenme sürecini gerçekleştirirken, test kümesi modelin doğruluğunu değerlendirmek için kullanılır. Doğru algoritmanın seçilmesi ve verilerin modele doğru

bir şekilde sunulması, bu aşamanın başarısını belirlemektedir. Model eğitimi süreci, doğru sonuçlara ulaşmak için modelin farklı parametrelerle yeniden eğitilmesini gerektirebilir.

Modelin Değerlendirilmesi (Evaluating the Model): Eğitilen modelin ne kadar başarılı olduğunu belirlemek için, modelin performansı değerlendirilir. Bu aşamada, eğitim sırasında kullanılmayan test kümesi, modelin doğruluğunu ve tahmin kabiliyetini ölçmek için kullanılır. Modelin performansı, çeşitli metriklerle (örneğin doğruluk, kesinlik) analiz edilir. Bu değerlendirme, modelin gerçek dünya verileriyle nasıl başa çıkacağını tahmin etmeye yardımcı olur ve modelin iyileştirilmesi için önemli geri bildirimler sağlamaktadır.

Performansı Artırmak (Improving the Performance): Modelin ilk değerlendirilmesinden sonra, performansı artırmak için çeşitli adımlar atılabilir. Bu aşama, modelin doğruluğunu ve etkinliğini geliştirmek için yapılacak optimasyonları içermektedir. Örneğin, modelin daha karmaşık hale getirilmesi, farklı bir algoritma seçilmesi veya eğitim verilerinin artırılması bu süreçte yapılabilecek müdahelelerdir. Performans iyileştirmeleri, daha fazla veri toplama, hiperparametre optimizasyonu veya modelin yeniden eğitilmesi gibi yöntemlerle sağlanabilmektedir. Bu aşama, sürekli bir iyileştirme sürecini içerir ve modelin genel başarısını artırmayı hedeflemektedir.

Makine öğrenmesi, pek çok farklı amaç doğrultusunda geliştirilen ve kullanılan bir teknolojidir. Bu amaçlar, genel olarak üç ana kategori altında incelenebilir (Carbonell ve ark., 1989): Hedef Tabanlı Çalışmalar, Bilişsel Simülasyon ve Teorik Analiz. Her bir kategori, makine öğrenmesinin farklı yönlerini ve bu alanın çeşitli uygulama alanlarındaki potansiyelini keşfetmek amacıyla yapılandırılmıştır.

Hedef Tabanlı Çalışmalar: Makine öğrenmesinin en yaygın kullanımlarından biri, belirli bir görevi yerine getirmek amacıyla geliştirilmiş sistemlerdir. Bu hedef tabanlı çalışmalar, belirlenen görevlerin etkin bir şekilde gerçekleştirilmesi için öğrenme sistemlerinin tasarımını ve analizini kapsamaktadır. Örneğin, bir görüntü tanıma sistemi, görsel verilerdeki nesneleri doğru bir şekilde tanımlamak için eğitilir. Bu tür sistemlerin amacı, spesifik bir problemi çözmek ve bu doğrultuda optimum performans göstermektir. Hedef tabanlı çalışmalar, yapay zekanın pratik uygulama alanlarında en çok karşılaşılan

kullanım senaryolarını oluşturmaktadır. Bu sistemler genellikle sağlık, finans, güvenlik ve perakende gibi çeşitli endüstrilerde başarıyla uygulanmaktadır.

Bilişsel Simülasyon: Bilişsel simülasyon, makine öğrenmesinin bir diğer önemli amacıdır ve insanın öğrenme süreçlerini araştırarak bu süreçleri bilgisayar ortamında modellemeyi hedeflemektedir. İnsan beyninin karmaşık öğrenme mekanizmalarının anlaşılması, bu mekanizmaların simülasyon yoluyla kopyalanması ve makinelere entegre edilmesi, bilişsel simülasyonun temelini oluşturmaktadır. Bu yaklaşım, insan benzeri zekaya sahip sistemler geliştirmek ve insan davranışlarını daha iyi anlamak için kullanılır. Örneğin, bir dil işleme modeli, insanların doğal dilde nasıl öğrendiğini simüle ederek, metinleri anlama ve üretme becerisi kazanır. Bilişsel simülasyon, yapay zekanın insan zekasını taklit etme potansiyelini araştıran çalışmalarda kritik bir rol oynamaktadır.

Teorik Analiz: Makine öğrenmesinin bir diğer önemli araştırma alanı ise teorik analizdir. Bu, belirli bir uygulama alanından bağımsız olarak, öğrenme metotları ve algoritmalarının derinlemesine incelenmesi anlamına gelmektedir. Teorik analiz, makine öğrenmesinin temel prensiplerini anlamak, algoritmaların matematiksel temellerini keşfetmek ve bu algoritmaların performansını analiz etmek için yapılmaktadır. Bu çalışmalar, makine öğrenmesinin daha geniş bir yelpazede uygulanabilmesi ve yeni algoritmaların geliştirilmesi için önemli bir zemin sağlamaktadır. Teorik analiz, makine öğrenmesinin sınırlarını belirlemek ve bu sınırların ötesine geçmek için yapılan çalışmaları kapsar. Örneğin, bir optimizasyon algoritmasının etkinliğini incelemek, daha karmaşık ve verimli modellerin oluşturulmasına olanak tanımaktadır.

Bu üç ana kategori, makine öğrenmesinin çeşitli amaçlarını ve araştırma yönlerini kapsamaktadır. Hedef tabanlı çalışmalar, pratik çözümler sunarken; bilişsel simülasyon, insan zihnini modellemeye odaklanır ve teorik analiz ise bu sistemlerin temel prensiplerini araştırır. Tüm bu yaklaşımlar, makine öğrenmesi alanının sürekli gelişimine katkıda bulunur ve bu teknolojinin çeşitli sektörlerdeki uygulamalarını genişletir.

Şekil 1. Makine Öğrenmesi Algoritmalarının Sınıflandırılması

Kaynak: Yazar tarafından oluşturulmuştur.

Makine öğrenmesi, çeşitli algoritmaların uygulanmasıyla farklı öğrenme modellerine dayalı olarak gerçekleştirilmektedir. Bu modeller, verilerin yapısına ve elde edilmek istenen sonuçlara bağlı olarak farklılık göstermektedir. Genellikle dört ana makine öğrenme modeli kullanılmaktadır; denetimli öğrenme, denetimsiz öğrenme, yarı denetimli öğrenme ve takviyeli öğrenme. Bu modellerin her biri belirli veri türlerini işlemek ve spesifik hedeflere ulaşmak amacı ile farklı algoritmalar kullanmaktadır.

2.1.2.1. Denetimli Öğrenme

Denetimli öğrenme, en yaygın kullanılan makine öğrenmesi modellerinden bir tanesidir. Bu modelde algoritmalar, etiketli veri kümeleri üzerinde eğitilir. Yani, giriş verilerine karşılık gelen doğru çıkışlar zaten bilinmektedir ve algoritma bu örneklerden öğrenerek yeni verilere doğru tahminler yapmayı amaçlar. Denetimli öğrenme algoritmaları genellikle sınıflandırma ve regresyon problemlerinde kullanılmaktadır. Örneğin, bir e-posta sınıflandırma sistemi, e-postaları spam veya spam değil olarak sınıflandırmak için denetimli öğrenme tekniklerinden faydalanmaktadır. Bu model, özellikle büyük ve etiketlenmiş veri kümelerine sahip olduğunuzda etkili sonuçlar verebilir.

2.1.2.2. Denetimsiz Öğrenme

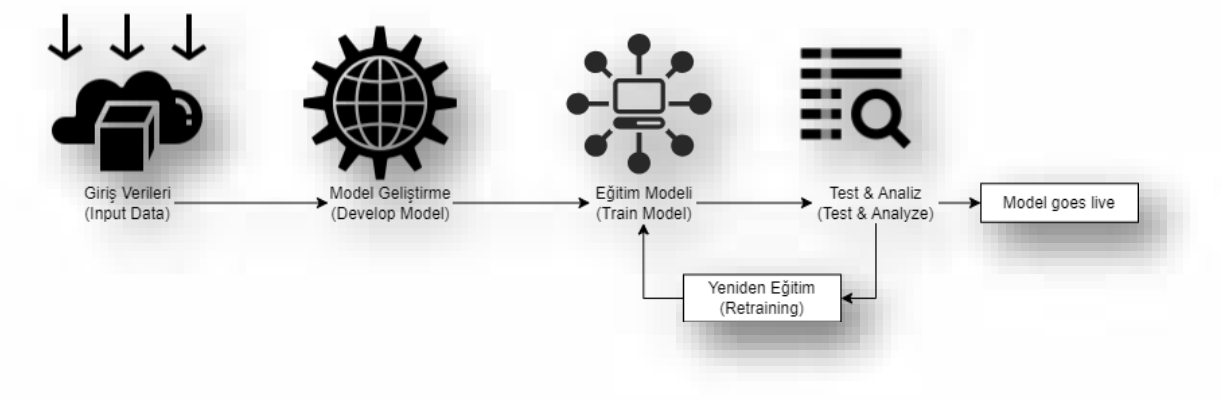
Denetimsiz öğrenme, verilerin etiketlenmediği durumlarda kullanılır. Bu model, algoritmaların verilerdeki gizli desenleri ve yapıları keşfetmesine olanak tanımaktadır. Denetimsiz öğrenme algoritmaları, özellikle kümeleme ve boyut indirgeme gibi algoritmalarda etkilidir. Örneğin, müşteri segmentasyonu yapmak isteyen bir perakende şirketi, denetimsiz öğrenme teknikleriyle müşterileri benzer özelliklerine göre gruplandırabilir. Bu model, verilerin doğal yapısını ortaya çıkarmak ve bilinmeyen ilişkileri keşfetmek için idealdir.

2.1.2.3. Yarı Denetimli Öğrenme

Yarı denetimli öğrenme, denetimli ve denetimsiz öğrenme modellerinin bir kombinasyonudur. Bu model, az miktarda etiketli veriyi büyük miktarda etiketsiz veri ile birleştirir. Yarı denetimli öğrenme, genellikle etiketli verilerin sınırlı olduğu ancak büyük miktarda etiketsiz veri bulunduğu durumlarda kullanılmaktadır. Örneğin, tıbbi görüntüleme alanında çok sayıda etiketsiz görüntü bulunurken, yalnızca birkaçının doğru bir şekilde etiketlenmiş olması yaygındır. Yarı denetimli öğrenme, bu etiketsiz görüntüleri de kullanarak daha doğru sonuçlar elde etmeye yardımcı olabilir.

2.1.2.4. Takviyeli Öğrenme

Takviyeli öğrenme, danışmanlı öğrenmeye benzer şekilde geri bildirim mekanizmalarını kullanır, ancak bu geri bildirimler ödül ve ceza yolu ile sağlanmaktadır. Bu yöntem, sistemlerin çevresel etkilere adaptasyon yeteneklerini geliştirir ve kendi performanslarını sürekli olarak iyileştirme fırsatı sunmaktadır. Bu süreçte danışmanın rolü, yalnızca sonuçları değerlendirmekle sınırlı kalmaz, aynı zamanda sistemin öğrenme dinamiklerini yönlendirme ve optimizasyon sürecine katkıda bulunma anlamında önemli bir yer tutar. Takviyeli öğrenme, özellikle oyun teorisi, robotik ve otonom sistemler gibi alanlarda kullanılmaktadır. Örneğin, bir otonom araç, çevresindeki engelleri algılayarak ve bunlara tepki vererek en güvenli sürüş stratejisini öğrenebilir. Takviyeli öğrenme, sürekli geri bildirim alarak performansı iyileştiren adaptif bir modeldir.



Şekil 2. Makine Öğrenmesi Çalışma Modeli

Kaynak: Yazar tarafından oluşturulmuştur.

Makine öğrenmesi modellerinde, kullanılan algoritmalar tek başına veya bir arada çalıştırılabilir. Bazı durumlarda, karmaşık ve öngörülemeyen veriler üzerinde mümkün olan en iyi sonuçlar elde edebilmek için birden fazla algoritmanın birleştirildiği hibrit yaklaşımlar tercih edilir. Bu tür yaklaşımlar, özellikle büyük veri analitiği ve derin öğrenme alanlarında yaygın olarak kullanılmaktadır. Örneğin, bir metin analiz projesinde, veriyi ön işlemek için denetimsiz öğrenme teknikleri kullanılabilirken, nihai sınıflandırma için denetimli öğrenme algoritmaları devreye girebilir. Algoritmaların bir arada kullanılması, makine öğrenmesinin gücünü artırarak daha doğru ve etkili sonuçlar elde edilmesine olanak tanır.

2.1.3. Derin Öğrenme

Derin öğrenme, makine öğrenmesinin bir alt dalı olarak, insan beyninin karmaşık işleyişini taklit etmek amacıyla geliştirilen çok katmanlı sinir ağlarını kullanmaktadır (Opperman, 2023). Bu yöntem, makinelerin verilerden daha karmaşık desenleri öğrenmesine olanak tanımaktadır ve yapay zekanın günümüzdeki birçok uygulamasına güç veren temel teknolojilerden biridir. Derin öğrenme ile geleneksel makine öğrenmesi algoritmaları arasındaki temel fark, kullanılan sinir ağlarının derinliği ve katman sayısında yatmaktadır. Geleneksel modeller daha az katman kullanırken, derin öğrenme modelleri, yüzlerce hatta binlerce katmanla çalışarak daha karmaşık problemlere çözüm bulmayı hedeflemektedir.

Bu çok katmanlı yapılar sayesinde derin öğrenme, resim tanıma, ses tanıma ve doğal dil işleme gibi insan zekâsı gerektiren birçok görevi başarılı bir şekilde yerine getirebilmektedir. Örneğin, dijital asistanlar, yüz tanıma sistemleri, dolandırıcılık tespit sistemleri ve otonom araçlar gibi günlük hayatımızda sıklıkla karşılaştığımız teknolojilerin arkasında derin öğrenme algoritmaları yer almaktadır.

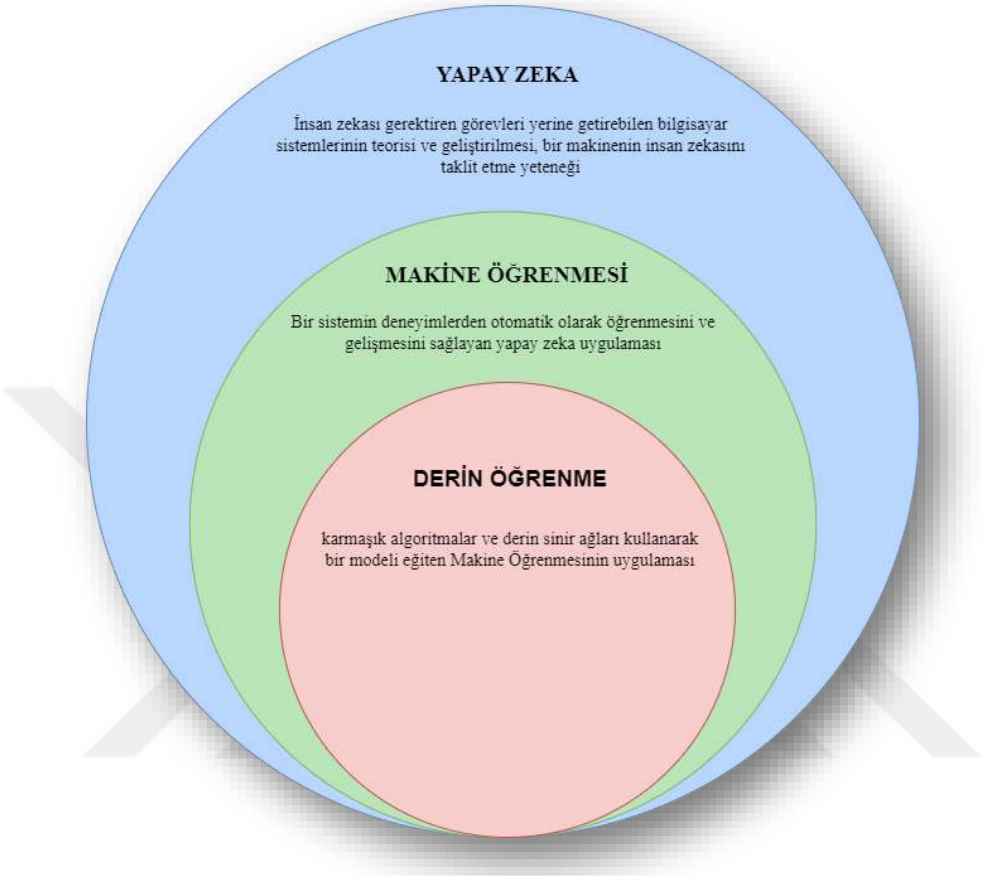
Derin öğrenme, sinir ağları adı verilen ve veri girişleri ile çalışan algoritmalar sayesinde, verilerdeki desenleri tanıyarak, sınıflandırma ve tahminler yapmaktadır. Sinir ağlarının katmanları arasında gerçekleşen ileri yayılım ve geri yayılım süreçleri, modelin hatalarını zamanla düzelterek daha doğru sonuçlar elde etmesini sağlamaktadır. Bu yapı sayesinde, derin öğrenme modelleri sürekli olarak kendini geliştirmekte ve karmaşık problemleri çözmede giderek daha etkili hale gelmektedir.

Derin öğrenmenin bu kendini geliştirme yeteneği, onu yapay zekâ alanında vazgeçilmez kılmaktadır. Özellikle büyük veri kümeleri üzerinde çalışarak, insan gözetimine gerek kalmadan kendi kararlarını optimize edebilme kabiliyeti, derin öğrenmenin geniş çapta benimsenmesine yol açmıştır. Sağlık hizmetlerinden otonom araç teknolojilerine, finansal tahminlerden müşteri hizmetleri sürecine kadar geniş bir yelpazede kullanılan bu teknoloji, geleceğin dijital dünyasında daha da büyük bir etki oluşturacaktır. Giderek artan veri miktarı ve hesaplama gücü ile, derin öğrenme algoritmalarının insan yaşamını dönüştürme potansiyeli sınırsızdır. Bu da yapay zekanın gelecekteki gelişiminde derin öğrenmenin kritik bir rol oynamaya devam edeceğini göstermektedir.

Makine Öğrenmesi, yapay zekanın önemli bir dalı olup, bilgisayarların veri üzerinde öğrenmesini sağlamaktadır. Bu alandaki en ileri tekniklerden bir olan derin öğrenme, özellikle karmaşık problemleri çözmek için kullanılan bir yöntemdir. Derin öğrenme, makine öğrenmesi tekniklerini daha ileri bir seviyeye taşıyarak, büyük veri setlerinde daha karmaşık analizler yapabilme yeteneğine sahiptir.

Yapay zekâ genel anlamda makinelerin insan benzeri düşünme ve karar verme yeteneklerini simüle etme çabasıdır. Bu geniş alanın bir alt dalı olan makine öğrenmesi, bilgisayarların deneyimlerinden öğrenip gelişmesini sağlarken, insan müdahalesine duyulan ihtiyacı azaltmaktadır. Derin öğrenme ise makine öğrenmesinin daha ileri bir seviyesi olarak, çok katmanlı yapay sinir ağları kullanarak daha büyük ve karmaşık veri

kümelerini analiz etmektedir. Bu üç kavram birbirleri ile yakından ilişkili olup, yapay zekâ teknolojilerinin gelişmesinde derin öğrenmenin makine öğrenmesi teknikleriyle olan etkileşimi önemli bir rol oynamaktadır.



Şekil 3. Makine Öğrenmesi, Yapay Zekâ ve Derin Öğrenme Arasındaki İlişki

Kaynak: Yazar tarafından oluşturulmuştur.

2.2. Doğal Dil İşleme

Doğal Dil İşleme (Natural Language Processing- NLP), bilgisayarların insan dilini anlamasını, işlemesini ve üretmesini sağlayan bir yapay zekâ (AI) dalıdır. Bu teknoloji, makine öğrenmesi ve derin öğrenme tekniklerinden yararlanarak, bilgisayarların dilin karmaşıklıklarını anlamasını mümkün kılmaktadır. NLP, dilbilimin kural tabanlı modellemesi ile istatistiksel modellemeyi birleştirir ve bu sayede bilgisayarların hem metinleri hem de sesli komutları tanıyıp işleyebilmesini sağlamaktadır. Bugün, NLP teknolojileri birçok günlük uygulamada kullanılmaktadır; arama motorları, dijital

asistanlar, sesli GPS sistemleri ve müşteri hizmetleri chatbotları bu teknolojinin yaygın kullanım alanlarından bazılarıdır.

Doğal Dil İşlemenin modern dünyadaki etkisi giderek artmaktadır. NLP'nin araştırma alanındaki ilerlemeler, büyük dil modellerinin (LLM'ler) iletişim yeteneklerinden, görüntü üretme modellerinin metin isteklerini anlamasına kadar geniş bir yelpazede teknolojik gelişmelere kapı açmıştır. Bu bağlamda, NLP'nin doğal dildeki veri sorgulama yetenekleri, kullanıcıların makinelerle daha doğal bir şekilde etkileşim kurmasına olanak tanımaktadır. Örneğin, sanal asistanlar, kullanıcıların sorularını anlayarak doğal bir dilde yanıt vermekte ve bu süreçte NLP teknolojisini kullanmaktadır. Ayrıca, web aramaları, e-posta spam filtreleme, otomatik çeviri, belge özetleme, duygu analizi ve yazım denetimi gibi birçok araç da NLP ile güçlendirilmiştir. Bu örnekler, NLP'nin yazılı ve sözlü diller üzerinde nasıl geniş bir etki oluşturduğunu göstermektedir.

Doğal dil işlemenin gücü, sadece mevcut verilerle sınırlı kalmamakta, aynı zamanda gelecekteki etkileşimleri de daha doğal ve akıcı hale getirebilmektedir. Verilerin bu şekilde işlenmesi, daha gelişmiş insan-bilgisayar etkileşimleri sağlayarak dijital dünyanın daha erişilebilir ve kullanıcı dostu olmasını sağlamaktadır.

Doğal Dil İşleme (DDİ) alanında yapılan çalışmalar, insan dilini anlamayı ve işleyebilmeyi amaçlayan çeşitli uygulamalarla hayat bulmaktadır. Bu çalışmalar, sadece metin ve ses verilerini analiz etmekle kalmaz, aynı zamanda bu veriler üzerinden anlam çıkarma, dil çevirisi, sesli etkileşim gibi gelişmiş işlevler sunarak, dijital araçların daha sezgisel ve etkili hale gelmesine katkı sağlamaktadır. Gelişen DDİ(NLP) teknolojileri, dilin karmaşıklığını çözümleyerek, dilsel hataları otomatik olarak düzeltme, kullanıcıların daha doğru ve akıcı metinler üretmesine yardımcı olma gibi özelliklere gündelik yaşamın ayrılmaz bir parçası olmuştur.

Bu kapsamda, DDİ teknolojileri birçok farklı alanı kapsamaktadır. Örneğin, yazılım yardımcı araçları sayesinde kullanıcılar metinlerindeki dilbilgisel ve yazım hatalarını kolayca düzeltebilirler. Bu tür araçlar, dilin kurallarını öğrenen modellerle desteklenmektedir ve yazım yanlışlarını düzeltme işlevi sağlamaktadır. "Bul ve Değiştir" gibi araçlar, metindeki belirli terimlerin veya kelimelerin hızlıca bulunup değiştirilmesini mümkün kılmaktadır. Optik karakter tanıma (OCR) teknolojileri ile basılı metinler dijital ortama aktarılmaktadır ve bu süreçte olası hatalar düzeltilmektedir.

Bunların yanı sıra, Doğal Dil İşleme teknolojileri, metin özeti çıkarma gibi karmaşık görevlerde de kullanılmaktadır. Bu özellik, uzun metinlerin ana noktalarını öne çıkararak kullanıcıların hızlıca bilgi edinmelerini sağlamaktadır. Bilgi çıkarma ise metin içerisindeki belirli bilgileri tespit edip organize etmektedir, böylece metinden anlamlı ve yapılandırılmış veriler elde edilmektedir. Ayrıca, bilgiye erişim ve metin anlama görevleri, büyük veri kümeleri içerisinde arama yapmayı ve kullanıcının aradığı bilgiyi daha hızlı bulmayı mümkün kılmaktadır.

Sesli etkileşim teknolojileri, bilgisayarlarla doğal bir şekilde iletişim kurmayı sağlamaktadır. Bilgisayarın konuşma yeteneği, metinleri sesli olarak okuma özelliği sunarken, konuşmayı anlama teknolojileri, sesli komutları metne dönüştürerek kullanıcıların konuşma yoluyla bilgisayarlara komut vermesini mümkün kılmaktadır. Soru yanıt sistemleri, kullanıcılardan gelen soruları analiz etmektedir ve doğru yanıtları vermektedir, böylece bilgiye erişim sürecini hızlandırır.

Yabancı dil öğrenme ve yazma yardımcıları gibi uygulamalar, dil öğrenme sürecinde kullanıcılara destek sağlamaktadır. DDİ'nin bir diğer önemli işlevi ise doğal diller arası çeviri yapabilme yeteneğidir. Bu özellik, farklı diller arasındaki metinlerin doğru ve hızlı bir şekilde çevrilmesini sağlar ve küresel iletişimi güçlendirir.

Bu teknolojiler, dilin derinliklerine nüfuz ederek, dijital dünyanın her alanında kullanıcıların yaşamını kolaylaştırmakta ve insan-bilgisayar etkileşimlerini daha verimli hale getirmektedir.

Doğal Dil İşleme çalışmaları, insan dilinin karmaşıklığı ve belirsizlikleri nedeniyle çeşitli zorluklarla karşı karşıya kalmaktadır. Bu zorluklar, dilin kuralsız yapısından kaynaklanan sorunlardan, metin ve konuşmanın anlamını doğru bir şekilde çıkarma çabalarına kadar geniş bir yelpazede yer almaktadır. DDİ alanında karşılaşılan zorluklar, bu teknolojilerin daha etkili ve güvenilir hale gelmesi için aşılması gereken önemli engeller olarak görülmektedir. Aşağıda bu zorluklar ve bunların üstesinden gelmek için sürdürülen çalışmaların bazı detayları ele alınmıştır.

Kuralsız ve Anlaşılmaz Konuşmalar: İnsanların konuşma tarzı, yazılı dile kıyasla çok daha düzensiz ve karmaşık olabilir. Konuşma sırasında dilbilgisel hatalar, duraklamalar, tekrarlar veya yanlış telaffuzlar gibi unsurlar ortaya çıkar. Bu durum, doğal dil işleme algoritmalarının konuşulan dili doğru bir şekilde anlamasını zorlaştırır. Bu

zorluğun üstesinden gelmek için dil modelleri, büyük veri kümeleri üzerinde eğitilerek farklı konuşma biçimlerini ve aksanları daha iyi anlamak için geliştirilir.

Kuralsız ve Bozuk Yazılar: Yazılı metinlerde, özellikle sosyal medya gibi platformlarda, dilbilgisel hatalar, eksik sözcükler veya karışık cümle yapıları sıkça görülmektedir. Bu tür bozuk yazıların işlenmesi, DDİ sistemleri için büyük bir zorluk teşkil etmektedir. Bu sorunu aşmak için yazım hatalarını otomatik olarak düzeltebilen ve eksik bilgileri tamamlayabilen algoritmalar geliştirilmekte ve iyileştirilmektedir.

Konuşmayı Dilimleme: Konuşma dilinin kesintisiz bir akış halinde olması, onu anlamlandırma sürecinde zorluk oluşturmaktadır. Konuşmayı anlamlı parçalara ayırmak, doğal dil işleme sistemlerinin bu parçalardan anlam çıkarmasına yardımcı olmaktadır. Bu aşamada kullanılan ses işleme algoritmaları, konuşmanın doğru bir şekilde dilimlenmesini sağlamaktadır ve daha sonra bu dilimler üzerinde anlamlandırma yapılmaktadır.

Metni Dilimleme: Benzer şekilde, yazılı metinlerde de cümlelerin ve paragrafların dilimlenmesi gereklidir. Bu süreç, metindeki kelime gruplarının ve cümle yapılarının analiz edilmesini kolaylaştırmaktadır. DDİ sistemleri, metni anlamlı bir şekilde dilimlemek için dilbilgisel kuralları ve istatistiksel yöntemleri kullanmaktadır.

Sözcük Niteliklerini Belirleme: Bir kelimenin cümledeki anlamı, genellikle bağlama bağlı olarak değişmektedir. Örneğin, “banka” kelimesi, finansal bir kuruluşu ya da bir nehir kıyısını ifade edebilmektedir. Bu tür anlam farklılıklarını doğru doğru bir şekilde belirlemek, DDİ sistemleri için önemli bir görevdir. Bu sorunu aşmak için kelimenin cümle içindeki rolünü analiz eden ve bağlamı dikkate alan dil modelleri geliştirilmektedir.

Anlam Belirsizliklerini Giderme: Doğal dillerde sıkça karşılaşılan anlam belirsizlikleri, aynı kelimenin farklı anlamlarda kullanılması veya bir cümlenin birden fazla şekilde yorumlanabilmesi gibi durumlardan kaynaklanmaktadır. Bu belirsizlikleri gidermek, doğal dil işleme algoritmalarının doğruluğunu artırmak için önemlidir. Bu konuda, anlam çıkarma süreçlerinde bağlamsal analiz ve makine öğrenmesi teknikleri kullanılarak belirsizlikler en aza indirilmeye çalışılmaktadır.

Söz Dizimsel Belirsizliklerin Giderilmesi: Cümle yapılarında söz dizimi hataları, cümledeki anlamın yanlış yorumlanmasına yol açabilmektedir. Bu nedenle, doğal dil

işleme sistemleri söz dizimsel belirsizlikleri çözmek için dil kurallarını öğrenmektedir ve bu kuralları cümle analizinde kullanılmaktadır.

Konuşma Planı: İnsanlar genellikle konuşmalarını belirli bir plana göre şekillendirir, ancak bu planlar esnek ve değişken olabilmektedir. DDİ sistemleri, konuşma planlarını anlamlandırmak ve bu planlara göre doğru cevaplar üretmek için karmaşık dil modellerine ihtiyaç duymaktadır. Bu süreçte, makine öğrenmesi algoritmaları ve büyük veri setleri kullanılarak konuşma planları daha doğru bir şekilde tahmin edilmeye çalışılmaktadır.

Bu zorlukların üstesinden gelmek için, dilbilim, istatistik ve makine öğrenmesi gibi farklı disiplinler bir araya gelerek sürekli olarak yeni teknikler ve yöntemler geliştirmektedir. DDİ alanında sürdürülen bu çalışmalar, insan dilinin karmaşıklığını çözme yolunda atılan önemli adımlar olarak değerlendirilebilmektedir.

Doğal Dil İşleme metotları, insan dilinin karmaşıklığını çözme ve dilin bilgisayar sistemleri tarafından anlaşılmasını sağlama sürecinde önemli adımlar atılmasına olanak tanımaktadır. Bu metotlar, dilin küçük bileşenlerine ayrılmasından, metnin anlamlandırılmasına kadar geniş bir yelpazede işlem gerçekleştiren teknikleri içermektedir. Aşağıda, doğal dil işleme alanında sıkça kullanılan bazı temel metotlar detaylandırılmıştır.

2.2.1. Cümleyi Ögelerine Ayırma (Tokenizasyon)

Tokenizasyon, bir metni anlamlı küçük birim parçalarına ayırma sürecidir. Bu süreçte, cümleler kelime ve noktalama işaretleri gibi daha küçük parçalara bölünür. Bu küçük parçalar, doğal dil işleme algoritmalarının dili analiz edebilmesi için temel birimler olarak kullanılır. Ancak, tokenizasyon işleminin sonucunda çok sayıda anlamsız veya gereksiz birim ortaya çıkabilir. Bu nedenle, etkisiz kelimelerin (stopwords) çıkarılması, yanlış yazılmış kelimelerin düzeltilmesi, kelimelerin köklerinin bulunması (stemming) ve baş kelimelerin belirlenmesi (lemmatization) gibi ek adımlar gereklidir. Bu adımlar, metnin daha anlaşılır ve işlenebilir hale getirilmesine katkı sağlamaktadır. Örneğin, “de”, “da”, “ama”, “şu” gibi etkisiz kelimeler metinden çıkarılırken, kelimenin kökünü bulma işlemi ile türetilmiş kelimeler sadeleştirilebilir.

Natural Language Processing

[‘Natural’,’Language’,’Processing’]

2.2.2. Normalizasyon

Normalizasyon, bir metni analiz edilebilir bir formata getirmek için yapılan düzenleme işlemlerini içermektedir. Bu süreçte, büyük ve küçük harf ayarlamaları, gereksiz boşluk veya karakterlerin ayıklanması, sayısal ifadelerin düzeltilmesi ve kısaltmaların açılması gibi adımlar yer almaktadır. Bu adımlar, metindeki gereksiz varyasyonları azaltarak dil işleme algoritmalarının daha doğru analizler yapmasını sağlamaktadır. Normalizasyon sürecinde kullanılan en yaygın yöntemlerden biri Levenshtein mesafesi olup, bu yöntem iki kelime arasındaki düzenleme mesafesini ölçer ve yanlış yazılmış kelimelerin düzeltilmesine yardımcı olur.

2.2.3. Morfoloji

Morfolojik analiz, metindeki kelimeleri dilbilgisel yapılarına göre sınıflandırma işlemidir. Bu süreç, kelimelerin kökleri ve ekleri üzerinden yapılan bir analizle gerçekleşir. Örneğin, “akılcı” kelimesi morfolojik olarak “akıl” (kök) ve “cı” (ek) şeklinde ayrılabilir. Morfolojik analiz, aynı kelimenin farklı biçimlerinin tanımlanmasına ve dildeki varyasyonların anlaşılmasına yardımcı olur. Bu, özellikle dilin karmaşıklığına karşı daha hassas ve doğru bir dil işleme süreci sağlar.

2.2.4. Vektörizasyon

Doğal dil işleme algoritmalarının metni matematiksel olarak işleyebilmesi için metnin sayısal bir forma dönüştürülmesi gerekmektedir. Vektörizasyon, metni vektörlere dönüştüren bir yöntemdir. Bu yöntem, kelime, cümle veya dokümanları sayısal değerlerle ifade ederek makine öğrenmesi algoritmalarının bu verileri işlemesini sağlamaktadır. En yaygın kullanılan vektörizasyon yöntemlerinden biri olan N-Gram, kelimeleri belirli bir sayıya göre gruplar ve böylece anlamsız kelime dizinlerini anlamlı hale getirir. Kelime çantası yaklaşımı (CBOW) ve TF-IDFgibi diğer vektörizasyon teknikleri de metinleri sayısal olarak temsil ederken farklı özellikler göz önünde bulundurulur.

2.2.5. Varlık İsim Tanıma

Varlık isim tanıma, metin belgelerinden belirli kategorilere (örneğin kişi, yer, kuruluş) ait öğelerin tanımlanması sürecidir. Bu süreç iki aşamalıdır: ilk aşamada varlık ismi belirlenirken, ikinci aşamada bu varlık belirli bir kategoriye atanır. Örneğin, bir metinde geçen "İstanbul" ifadesi şehir kategorisine atanabilir. Varlık isim tanıma, müşteri hizmetleri, haber içerik sınıflandırması ve duygu analizi gibi birçok uygulamada kullanılmaktadır ve bu süreçte kullanılan etiketleme formatları arasında Raw, IOB ve IOB2 gibi yöntemler bulunmaktadır.

2.2.6. Metin Sınıflandırma

Metin sınıflandırma, bir metnin içeriğinin belirli bir konuya veya sınıfa ait olduğunu tahmin eden doğal dil işleme yöntemidir. Bu yöntem, metnin yazarını belirlemekten, spam e-postaları tanımlamaya kadar birçok farklı alanda kullanılabilir. Metin sınıflandırma hem denetimli hem de denetimsiz makine öğrenmesi teknikleri ile uygulanabilir ve büyük veri setleri üzerinde eğitilerek daha doğru sonuçlar elde edilmesine olanak tanımaktadır.

2.2.7. Dil Tanımlama

Dünyada binlerce dilin konuşulduğu göz önüne alındığında, bir metni analiz etmeden önce bu metnin hangi dilde yazıldığını belirlemek önemlidir. Dil tanımlama, metnin dilini tespit eden yöntemleri içermektedir ve dilin sözlüğü ve dilbilgisel yapısı bu süreçte önemli rol oynamaktadır. Bu tanımlama işlemi, doğal dil işleme sistemlerinin doğru kaynakları kullanarak metni anlamlandırmasına yardımcı olmaktadır.

2.2.8. Duygu Analizi

Duygu analizi, bir metni olumlu, olumsuz veya tarafsız olduğunu belirleyen doğal dil işleme yöntemidir. Bu yöntem, sosyal medya platformlarındaki kullanıcı yorumlarını analiz etmek, ürün değerlendirmelerini sınıflandırmak veya bir konuda genel halkın duygusal tepkisini ölçmek için yaygın olarak kullanılmaktadır. Duygu analizi, metindeki duygusal ifadeleri tespit ederek işletmelerin veya kuruluşların daha iyi kararlar almasına yardımcı olmaktadır.

Bu yöntemler, doğal dil işlemenin temellerini oluşturarak dilin makine tarafından işlenmesini mümkün kılmaktadır. Bu tekniklerin doğru uygulanması, dilin karmaşıklığının çözümlenmesine ve bilgisayarların insan dilini anlamasına büyük katkı sağlamaktadır.

İnsanlar tarafından üretilen metinler, dilin doğallığı ve duygusal derinliği açısından makinelerden farklılık göstermektedir. İnsan yazısı, kültürel bağlamı ve bireysel anlatımı içermektedir. Ayrıca, insanların kullandığı dil, deneyimlere dayanan duygusal ve sosyal ipuçlarıyla zenginleştirilmiştir. Bununla birlikte, insanlar genellikle dilin kurallarını esnetebilir, ifadelerinde nüanslar ve belirsizlikler kullanabilirler. Bu, insan dilinin öznel ve duruma göre değişken olabileceği anlamına gelmektedir.

Makine tarafından üretilen metinler ise daha yapılandırılmış, kural tabanlı ve mantıksal kalıplara dayanmaktadır. Doğal dil işleme algoritmaları, dilin kurallarına göre metin üretirken, duygusal anlam katmada zorluk yaşayabilmektedir. Makine tarafından üretilen metinler genellikle doğru cümle yapıları ve anlamlı cümlelerle doludur. Ancak dil oyunları, metaforlar veya derin anlam katmanları genellikle sınırlı kalmaktadır. Dilin duygusal yönleri, makine metinlerinde insan metinlerine kıyasla daha az bulunmaktadır.

2.2.9. Derin Öğrenme ve Doğal Dil İşlemenin Kesişim Noktası

Derin öğrenme ve doğal dil işleme (NLP), son yıllarda giderek daha fazla kesişen iki önemli yapay zekâ alanıdır. Derin öğrenme, çok katmanlı yapay sinir ağlarını kullanarak büyük veri kümelerinden karmaşık örüntüler çıkarmada son derece başarılıdır. Bu yetenek, özellikle dilin işlenmesi ve anlaşılması gibi karmaşık görevler için büyük avantajlar sunmaktadır. Derin öğrenme teknikleri, dil modelleme, makine çevirisi, duygu analizi ve metin üretimi gibi DDİ alanlarında önemli gelişmeler sağlamıştır.

Doğal dil işlemenin geleneksel yöntemleri, genellikle istatistiksel modelleme ve dil kurallarına dayanan tekniklerle sınırlıydı. Ancak derin öğrenme, bu sınırları aşarak, dilin daha geniş ve soyut anlamlarını öğrenme yeteneği sunmuştur. Örneğin, derin öğrenme tabanlı dil modelleri, dildeki kelimeler arasındaki ilişkileri çok daha derin bir şekilde öğrenebilmektedir. Bu sayede, metinlerin bağlamını daha iyi anlayabilir ve anlamlandırabilir hale gelmektedir. Özellikle sinir ağı temelli modeller olan

Transformer'lar, büyük miktarda veriyi analiz ederek dilin karmaşık yapısını öğrenebilmektedir ve metinlerin anlamını etkili bir şekilde çıkarabilmektedirler.

Derin öğrenme, ayrıca büyük dil modellerinin (LLM'ler) geliştirilmesinde de kritik bir rol oynamaktadır. Bu modeller, yüz milyonlarca parametre ile eğitilerek, insan dilini çok daha doğal bir şekilde taklit edebilmektedirler. Bu sayede, makine tarafından üretilen metinler, insan diline daha yakın hale gelmiştir. Derin öğrenme teknikleri, DDİ uygulamalarının doğruluğunu ve etkinliğini artırarak, daha insana benzer metinlerin üretilmesine olanak tanımaktadır. Bu gelişmeler, DDİ ve derin öğrenmenin birbirini tamamlayıcı doğasını göstermektedir ve bu iki alanın kesişim noktasının gelecekteki yapay zekâ çalışmalarında önemli bir rol oynamaya devam edeceğini göstermektedir.

2.2.10. Metinlerin Kökenini Tespit Etmenin Zorlukları

Makine yazımı ile insan yazımı metinleri birbirinden ayırt etmek, doğal dil işleme ve metin analitiği alanlarında giderek daha fazla önem kazanan bir sorun olarak öne çıkmaktadır. Metinlerin kökenini belirlemek, yalnızca metinlerin yazılım süreçlerinin teknik yönlerini anlamakla kalmayıp, aynı zamanda içerik doğrulama, dezenformasyon tespiti ve yapay zekâ tabanlı sistemlerin etik sınırlarını tartışmak açısından da kritik bir role sahiptir. Ancak bu süreç, metinlerin içerik, bağlam ve stil açısından karmaşık doğası nedeniyle önemli zorluklar barındırmaktadır. İnsan yazımı metinlerde, genellikle bireysel ifade biçimleri, duygusal tonlar ve stilistik özgünlük ön plana çıkarken, makine yazımı metinler daha çok algoritmanın öğrenme mekanizmalarından türeyen belirli yapısal ve dilsel kalıplar taşımaktadır. Bu ayrımı doğru bir şekilde tespit edebilme için stil, içerik ve bağlam analizlerinin yanı sıra, metinlerin oluşum sürecinde kullanılan teknolojilerin ve insan etkisinin daha derinlemesine incelenmesi gerekmektedir.

Metinlerin kökenini tespit etmek hem dilbilimsel hem de teknolojik açıdan karmaşık bir süreçtir. İnsan yazımı ve makine yazımı metinler arasında stil, bağlam ve içerik açısından belirgin farklar bulunmasına rağmen, bu farkları tespit etmek her zaman kolay değildir. Özellikle yapay zekâ destekli metin oluşturma araçlarının giderek daha gerçekçi ve insan benzeri metinler üretmesi, bu ayrımı daha da zorlaştırmaktadır. İnsan yazımı metinler, genellikle duygusal ifadeler, bireysel yazım stilleri ve belirli bir bağlama dayalı özgün içerik taşırken, makine yazımı metinler daha tutarlı dil bilgisi kuralları ve

genelleştirilmiş kalıplar kullanma eğilimindedirler. Ancak gelişmiş dil modelleri bu kalıpları giderek daha ustaca maskeleyerek, makine metinlerini insan yazımı metinlerden ayırt etmeyi zorlaştırmaktadır.

Bu zorlukları aşmak için geliştirilen dil modelleri ve metin analiz araçları stilometrik analizden bağlam tabanlı değerlendirme yöntemlerine kadar çeşitli yaklaşımlar sunmaktadır. Stilometrik analiz, metindeki kelime sıklıkları, cümle uzunlukları ve dil bilgisel yapı gibi istatistiksel özellikleri incelerken, bağlam tabanlı değerlendirme metni bir bütün olarak anlamlandırarak daha geniş bir perspektif sunmaktadır. Ancak bu yöntemlerin başarı oranı kullanılan veri setlerinin ve algoritmaların kalitesine bağlıdır. Örneğin, karmaşık ve uzun metinlerde insan faktörünün daha belirgin olduğu varsayılırken, kısa ve yapılandırılmış metinlerde bu ayrımı yapmak daha zordur.

Metinlerin kökenini tespit etmek için uygulanan analiz yöntemlerinde, dil modelleri ve metin işleme tekniklerinin gelişmiş bir şekilde değerlendirilmesi gerekmektedir. Bu süreçte stil, içerik ve bağlam analizi gibi farklı yaklaşımlar kullanılırken, yapay zekâ ve insan yazımı metinler arasındaki farkların teknik detaylarla açıklanması faydalı olacaktır.

2.2.10.1. Dil Modeli İstatistiklerinin Analizi

Yapay zekâ ile üretilmiş metinler genellikle n-gram analizi, kelime sıklığı ve cümle uzunluğu gibi istatistiksel özellikler bakımından farklılık göstermektedir. Örneğin, bir dil modeli tarafından üretilmiş metinlerde kelime tekrarları daha fazla olabilir ve cümle uzunlukları daha düzenlidir. İnsan yazımı metinlerde ise genellikle daha karmaşık cümle yapıları ve yaratıcı dil kullanımı görülmektedir.

Yapay zekâ tarafından yazılmış bir metin:

- Ortalama cümle uzunluğu: 10 kelime
- Kelime çeşitliliği: %70

İnsan yazımı bir metin:

- Ortalama cümle uzunluğu: 15 kelime
- Kelime çeşitliliği: %85

2.2.10.2. Stilometrik Özelliklerin Değerlendirilmesi

Stilometri, bir yazarın yazım stiline dair ölçülebilir özellikleri ortaya koymaktadır. Örneğin, bağlaçların kullanım sıklığı, yüklem uzunlukları ve noktalama işaretlerinin dağılımı gibi faktörler analiz edilebilir. Yapay zekâ yazımı metinlerde, bu faktörler genellikle daha standart ve mekanik bir yapıya sahiptir.

Bağlaç Kullanımı: Yapay zekanın yazdığı bir metin, genellikle yaygın bağlaçları ("ve", "ancak", "çünkü") fazla kullanmaktadır ve daha az varyasyon göstermektedir. İnsan yazımı metinlerde ise bağlaç seçimi daha çeşitlidir ve bağlama uygun şekilde kullanılmaktadır.

Noktalama Kullanımı: Yapay zekâ tarafından yazılan metinlerde, nokta ve virgül gibi işaretler genellikle belirli kurallar çerçevesinde sabit kalırken, insan yazımı metinlerde yazarın stiline bağlı olarak bu işaretlerin kullanımı değişebilmektedir.

2.2.10.3. Bağlam ve Semantik Uyumluluk

Yapay zekâ tarafından oluşturulan metinlerde bağlamın doğal akışı zaman zaman bozulabilmektedir. Özellikle uzun metinlerde, anlamsal tutarlılığın eksikliği daha belirgin hale gelmektedir. İnsan yazımı metinler ise genellikle bir olay örgüsü veya bağlam mantığı içinde daha tutarlı ilerlemektedir.

Bir metni bağlam açısından analiz etmek için kelime gömme modelleri (örneğin Word2Vec veya BERT) kullanılabilir. Bu modeller, metin içerisindeki kelimeler arasındaki anlamsal ilişkileri ölçerek bağlam uyumluluğunu değerlendirir. Yapay zekâ yazımı metinlerde bu ilişkiler daha az organik olabilir.

2.2.10.4. Duygusal Dil Kullanımı

İnsan yazımı metinlerde duygusal dil kullanımı daha belirgindir. Yapay zekâ metinlerinde ise yaratıcı ifadeler genellikle eksik veya sınırlıdır. Bu durum, metinlerin özgünlük ve insana özgü anlatım derinliği açısından farklılık göstermesine neden olmaktadır.

Sentiment Analysis (Duygu Analizi) gibi metin işleme teknikleri, bir metindeki duygusal tonu ölçmek için kullanılabilmektedir. İnsan yazımı metinler genellikle daha

karmaşık duygusal tonlar içerirken, yapay zekâ metinlerinde bu tonlar daha yüzeysel olabilmektedir.

Metinlerin kökenini tespit etmenin zorluklarını daha iyi anlamak için günlük hayattan somut örnekler üzerinden açıklamalar yapmak faydalı olacaktır. Bu örnekler, yapay zekâ ile üretilmiş metinler ile insan yazımı metinler arasındaki farkların analiz edilmesi sırasında karşılaşılan durumları detaylandırmaktadır.

2.2.10.5. Yapay Zekâ Destekli Haber Yazımı

Haber ajansları, spor karşılaşmalarının sonuçlarını veya borsa verilerini hızla özetlemek için yapay zekâ kullanmaktadır. Örneğin, bir futbol maçının sonucu şu şekilde bir metinle ifade edilebilir: "X takımı, Y takımını 3-2 mağlup etti. Maç boyunca etkili bir performans sergilediler." Bu tür metinlerde genellikle bağlam derinliği ve insana özgü yorumlama eksikliği bulunur. Buna karşın, insan yazımı haberlerde olayın perde arkasına dair detaylar ve duygusal ifadeler daha sık yer alır.

2.2.10.6. Akademik Metinlerde Otomasyon

Literatür taraması veya istatistiksel analiz bölümleri, yazı otomasyonu ile kolaylaştırılabilir. Örneğin, "Yapılan çalışmalar, X metodunun Y problemine yönelik etkinliğini göstermektedir" gibi genelleştirilmiş bir ifadeyle karşılaşılabilir. Yapay zekanın yazdığı bu tür metinlerde özgünlük eksikliği gözlemlenirken, insan yazımı metinlerde daha özgün analizler ve detaylı referanslar yer almaktadır.

2.2.10.7. E-Ticaret Ürün İncelemeleri

Bir e-ticaret platformunda bir ürün için yazılmış şu yorum dikkate alınabilir: "Bu ürün harika! Kesinlikle tavsiye ederim." Yapay zekâ ile oluşturulan bu tür yorumlar, genellikle kısa, genelleştirilmiş ve duygusal bir mesafe sergiler. İnsan yazımı yorumlar ise daha kişisel deneyimlere ve detaylara dayanır: "Ürünü bir haftadır kullanıyorum ve dayanıklılığından oldukça memnunum."

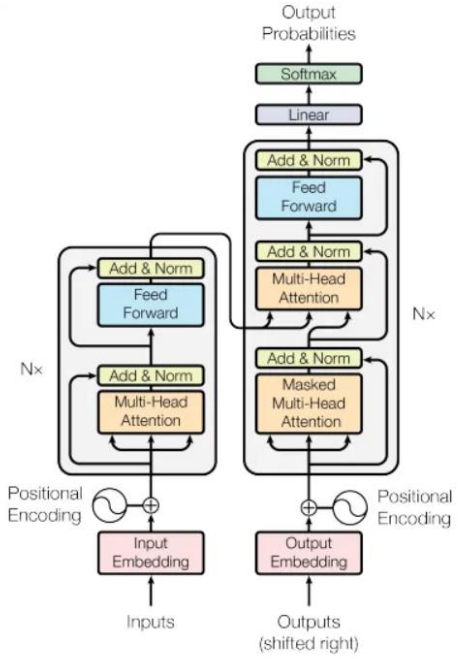
2.2.10.8. Sahte Haberlerin Tespiti

Bir yapay zekâ, sahte bir haber üreterek şu tür bir metin oluşturabilir: “Bilim insanları, günlük yaşamda kullanılan X maddenin insan ömrünü 10 yıl uzattığını keşfetti.” Bu tür metinlerde genelde kaynak gösterimi eksik olur veya kullanılan kaynaklar tutarsızdır. İnsan yazımı haberlerde ise daha fazla kanıt ve bağlam bilgisi yer alır.

Metinlerin kökenini tespit etmek, günümüz teknolojileri ile yapay zekâ yazımı metinlerin yaygınlaşması nedeniyle giderek karmaşık hale gelmektedir. Stilometrik analiz, semantik değerlendirme ve bağlam incelemesi gibi yöntemler, bu süreçte önemli bir rol oynamaktadır. Bununla birlikte, yapay zekâ modellerinin sürekli gelişmesi, insan yazımı metinlerle olan farkların giderek azalmasına yol açmaktadır. Dolayısıyla, metinlerin kökenini tespit etmek için daha hassas ve çok yönlü yaklaşımların geliştirilmesi gerekmektedir. Bu analizlerin doğruluğunu artırmak için istatistiksel ve semantik yöntemlerin bir arada kullanılması hem akademik çalışmalar hem de pratik uygulamalar için kritik bir öneme sahiptir.

2.2.11. Transformer Modelleri ve GPT Teknolojisi

Transformer mimarisi, doğal dil işleme (NLP) alanında büyük bir dönüşüm sağlayan bir yapıdır ve modern dil modellerinin temelini oluşturmaktadır. 2017 yılında "Attention is All You Need" başlıklı çalışmada tanıtılan bu mimari, özellikle "dikkat mekanizması" (attention mechanism) sayesinde dildeki uzun mesafeli bağımlılıkları etkili bir şekilde modelleyebilme yeteneğine sahiptir (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, ve Polosukhin, 2023). Transformer tabanlı modeller, ardışık işleme yerine paralel işlemeyi destekleyerek hem öğrenme hızını artırmış hem de daha büyük veri kümeleriyle çalışmayı mümkün kılmaktadır. Bu mimariyi temel alan GPT-3 ve GPT-4 gibi modeller, yalnızca metin anlama değil, aynı zamanda bağlama uygun ve akıcı metin üretimi yapabilme özellikleriyle dikkat çekmektedir. GPT teknolojisi, çok çeşitli uygulama alanlarında dil üretimi yeteneklerini sergileyerek, yapay zekâ destekli içerik oluşturma, diyalog sistemleri ve daha birçok alanda yeni olanaklar sunmaktadır.



Şekil 4. Transformer Mimarisi

Kaynak: Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., ve Polosukhin, I. (2023). *Attention is all you need*. arXiv. https://arxiv.org/pdf/1706.03762

Bu şema, Transformer modelinin temel yapısını göstermektedir. Sol tarafa "Encoder" (Kodlayıcı) ve sağ tarafta "Decoder" (Kod Çözücü) yer almaktadır. Encoder, girdileri (örneğin bir metni) alır ve bu girdilerden özellikler çıkarır. Decoder ise bu özellikleri kullanarak anlamlı çıktılar üretir.

Input Embedding: Metin verileri sayısal temsillere dönüştürülür.

Positional Encoding: (Konumsal Kodlama) Kelimelerin sırasını anlaması için modele ek bilgi sağlar.

Multi-Head Attention: Kelimeler arasındaki bağlamı anlamak için dikkat mekanizmasını birden fazla başlık üzerinden uygular.

Add ve Norm: Katmanlar arasında bilgi akışını düzenler ve normalizasyon uygular.

Feed Forward: (İleri Beslemeli Katman) Kelimeler arasındaki ilişkileri daha karmaşık bir şekilde işler.

Masked Multi-Head Attention: Çıkışları üretirken, sıradaki kelimeye odaklanmak için maskeleme uygular.

Output Embedding: Çıktıdaki kelimeleri sayısal temsillere dönüştürür.

Softmax: Nihai çıktı olasılıklarını hesaplar.

Bu mimari, ChatGPT gibi dil modellerinin metinlerini anlamasına ve üretmesine olanak tanımaktadır. Encoder, girdilerin bağlamsal anlamını çıkarırken, Decoder bu bilgiyi dil üretimi için kullanmaktadır.

Transformer mimarisi, özellikle ölçeklenebilirliği ve çok başlı dikkat mekanizması ile NLP uygulamalarında benzersiz bir esneklik sunmaktadır. GPT-3 ve GPT-4 gibi modeller, bu mimarinin gücünü kullanarak büyük miktarda veriden anlam çıkarabilmekte ve bunu yüksek doğrulukta metin üretimine dönüştürebilmektedir. Bu modeller, yalnızca bireysel kelimeler arasındaki ilişkileri değil, aynı zamanda metin içindeki uzun mesafeli bağlamları da dikkate almaktadır. Örneğin, bir paragrafın başındaki bir kelimenin, paragrafın sonundaki bir cümleyle olan bağlantısını algılayabilir. Bu özellik, dil modellerinin daha insana yakın ve bağlama duyarlı metinler üretmesini mümkün kılmaktadır.

Bunlara ek olarak, GPT (Generative Pre-trained Transformer) teknolojisinin temelinde yer alan "önceden eğitme ve sonradan ayar yapma" (pre-training and fine-tuning) stratejisi modellerin çok yönlülüğünü artırmıştır. Önceden eğitme süreci (pre-training), modelin genel dil bilgisi ve bağlamsal anlamı öğrenmesini sağlarken, ince ayar (fine-tuning) süreci, modelin belirli bir göreve veya uygulamaya uyarlanmasına olanak tanımaktadır. Bu sayede GPT modelleri yalnızca genel metin üretiminde değil, aynı zamanda müşteri hizmetleri, yazılım geliştirme, içerik oluşturma ve bilimsel araştırma gibi özel alanlarda da başarıyla kullanılmaktadır.

Sonuç olarak, Transformer mimarisi ve bu mimarinin temel alındığı GPT-3 ve GPT-4 gibi modeller, doğal dil işleme alanında büyük bir paradigma değişimi sağlamıştır. Dikkat mekanizmasının etkin kullanımı ve paralel işlemeyi mümkün kılan yapısıyla Transformer hem metin anlama hem de metin üretimi konularında insan seviyesine yakın performanslar sergileyebilen sistemlerin geliştirilmesini sağlamıştır. GPT modelleri, geniş çaplı ön eğitim süreçleri sayesinde genel dil bilgisi ve bağlamı anlama yeteneği kazanmış, ardından ince ayar süreçleriyle belirli görevlerde üstün sonuçlar elde etmiştir.

Bu teknolojiler, müşteri hizmetleri chatbotları, içerik üretimi, dil çevirisi, programlama desteği, eğitim materyalleri oluşturma ve hatta yaratıcı yazarlık gibi birçok alanda uygulanabilir hale gelmiştir. Ayrıca, modellerin çok büyük parametre sayıları ve kapsamlı veri kümeleri ile eğitilmesi, bağlamın karmaşıklığını daha derinlemesine anlamalarını sağlamış ve dil işleme görevlerindeki doğruluk oranlarını önemli ölçüde artırmıştır. Ancak, bu modellerin üretim süreçlerinde bağlamdan kopuk ifadeler veya anlamsal tutarsızlıklar oluşabilmesi gibi zorluklar da bulunmaktadır. Bu durum, modellerin sürekli geliştirilmesini ve etik açıdan dikkatli bir şekilde kullanılmasını gerektirmektedir.

Transformer tabanlı GPT teknolojileri, doğal dil işleme alanında yenilikçi yaklaşımların temel taşını oluşturmaktadır. Gelecekte bu modellerin daha verimli, güvenilir ve erişilebilir hale gelmesiyle, yapay zekanın dil işleme yeteneklerinin daha da gelişmesi beklenmektedir. Böylece, insan-makine etkileşiminde yeni bir dönemin kapıları açılarak, birçok sektörde yapay zekanın daha etkili bir şekilde kullanılabileceği bir ortam oluşacaktır.

2.2.12. Yapay Zekâ Yazımı Metinleri Tespit Etme Yöntemleri

Yapay zekâ tarafından üretilen metinleri tespit etme, doğal dil işleme alanında giderek önem kazanan bir araştırma konusu haline gelmiştir. İnsan yazımı metinler ile yapay zekâ yazımı metinler arasındaki farkları belirlemek hem içerik doğrulama süreçlerinde hem de etik ve yasal uyumluluk sağlama açısından kritik bir rol oynamaktadır. Bu bağlamda, stilometri ve dilsel özelliklerin analizi, metinlerin yazarını veya üretim kaynağını tespit etmek için kullanılan önemli yöntemler arasında yer almaktadır. Kelime seçimleri, cümle yapıları ve bağlam uyumluluğu gibi dilsel unsurlar incelenerek, yapay zekâ tarafından üretilen metinlerin stilistik özellikleri ayırt edilebilmektedir. Bunun yanı sıra, gelişmiş otomatik araçlar ve yazılım çözümleri, yapay zekâ yazımı metinlerin daha hızlı ve hassas bir şekilde tespit edilmesine olanak tanımaktadır. Bu yöntemler, sahte haberlerin yayılmasını engellemek, içerik doğruluğunu artırmak ve yapay zekanın etik kullanımını teşvik etmek için kritik bir öneme sahiptir.

Stilometri, yapay zekâ yazımı metinlerin tespitinde sıklıkla başvurulan bir yöntemdir ve temelinde dilsel stilin matematiksel olarak analiz edilmesi yatmaktadır. Bu

yöntem, kelime frekansı, cümle uzunluğu, noktalama işaretlerinin kullanımı ve bağlaçların sıklığı gibi metinlerin dilsel özelliklerini ölçerek yazım tarzındaki belirgin farklılıkları ortaya koymaktadır. Yapay zekâ tarafından üretilen metinler genellikle daha mekanik ve tutarlı bir dil kullanırken, insan yazımı metinler daha fazla çeşitlilik ve bağlama duyarlılık gösterebilmektedir. Örneğin, insan yazarların metinlerinde kişisel tarzları ve duygusal ifadeler daha belirgindir; buna karşın yapay zekâ metinlerinde bu özellikler genellikle daha genel ve standart bir yapıya sahiptir. Bu nedenle, stilometri metinlerin kaynaklarını anlamada etkili bir araç olarak öne çıkmaktadır.

Dilsel özelliklerin analizi, yapay zekâ yazımı metinleri tespit etmede kullanılan bir diğer güçlü yaklaşımdır. Bu analiz, metnin bağlamsal anlamını ve kelimeler arasındaki ilişkileri inceleyerek metnin üretim kaynağına dair ipuçları sağlamaktadır. Örneğin, yapay zekâ tarafından üretilen metinlerde genellikle tekrar eden kalıplar, öngörülebilir ifadeler ve bağlamdan kopuk cümleler bulunabilmektedir. Buna karşın, insan yazımı metinler daha esnek ve bağlama uygun ifadeler içermektedir. Özellikle derin öğrenme tabanlı modellerin ürettiği metinler, bağlam ilişkilerini anlamada güçlü olsa da dilin doğal akışında bazı eksiklikler gösterebilmektedir. Bu tür eksiklikler, yapay zekâ yazımı metinlerin tespit edilmesinde kilit bir rol oynamaktadır.

Gelişmiş otomatik araçlar ve yazılım çözümleri, metin analizini daha hızlı ve etkili bir şekilde gerçekleştirme olanağı sunmaktadır. Günümüzde, doğal dil işleme tabanlı araçlar hem stilometrik analiz hem de dilsel özelliklerin incelenmesi için optimize edilmiştir. Bu araçlar, büyük veri kümeleri üzerinde çalışarak yapay zekâ yazımı metinlere özgü örüntüleri öğrenebilmekte ve bunları tespit etmek için algoritmalar geliştirebilmektedir. Örneğin, yazım tarzındaki tutarlılık, dildeki tekrar eden ifadeler ve mantık hataları gibi faktörler, otomatik sistemler tarafından tespit edilebilmektedir. Bu tür araçlar, sahte haberlerin veya yanıltıcı içeriklerin tespitinde kritik bir rol oynamakta ve yapay zekâ tarafından oluşturulan içeriklerin etik çerçevede değerlendirilmesini sağlamaktadır.

Bununla birlikte, yapay zekâ yazımı metinlerin tespitinde karşılaşılan zorluklar da bulunmaktadır. Örneğin, GPT gibi gelişmiş dil modelleri, bağlama dayalı metin üretiminde insan benzeri bir doğruluk sağlayabildiği için, bu metinlerin tespit edilmesi giderek daha karmaşık bir hale gelmiştir. Ayrıca, yapay zekâ modellerinin sürekli

gelişmesi, tespit yöntemlerini de sürekli yenileme gerekliliğini doğurmaktadır. Bu nedenle, stilometri ve otomatik araçların yanı sıra, metin analizi için hibrit yaklaşımlar geliştirilmesi önemlidir.

Sonuç olarak, yapay zekâ yazımı metinleri tespit etme, hızla dijitalleşen dünyada bilgi doğruluğunu sağlama ve etik içerik üretimi açısından hayati bir önem taşımaktadır. Stilometri ve dilsel özelliklerin analizi gibi yöntemler, metinlerin kaynaklarını tespit etmede güçlü bir temel sunarken, otomatik araçlar ve yazılım çözümleri bu süreci daha hızlı ve ölçeklenebilir hale getirmektedir. Ancak, yapay zekâ modellerinin gelişmesiyle birlikte, bu metinlerin tespit edilmesi giderek daha karmaşık bir hale gelmektedir. Bu durum, mevcut yöntemlerin sürekli olarak yenilenmesini ve daha kapsamlı, hibrit yaklaşımların geliştirilmesini gerektirmektedir.

Gelecekte, yapay zekâ yazımı metinlerin tespiti için hem teknolojik hem de etik açıdan daha sofistike çözümlerin benimsenmesi beklenmektedir. Bu bağlamda, yalnızca metin tespiti değil, aynı zamanda yapay zekâ sistemlerinin şeffaflığı ve hesap verebilirliği üzerine çalışmalar da büyük önem kazanacaktır. Nihayetinde, bu alandaki gelişmeler, dijital ortamda güvenilir bilgiye erişimi artıracak ve yapay zekâ destekli içeriklerin etik sınırlar içinde kullanılmasını teşvik edecektir.

2.2.13. İnsanlar ve Makineler Arasındaki Çizgi – NLP'nin Geleceği

Doğal dil işleme (NLP) alanındaki hızlı gelişmeler, metin üretiminde insanlar ve makineler arasındaki sınırların giderek belirsizleşmesine yol açmaktadır. Özellikle GPT-4 gibi ileri dil modellerinin ortaya çıkışı, makine yazımı metinlerin insan yazımı metinlere yakın bir akıcılık ve bağlam uyumu sergilemesini sağlamaktadır (OpenAI, 2023). Bu durum, dil teknolojilerinin geleceğine dair heyecan verici fırsatlar sunarken, aynı zamanda etik, doğruluk ve özgünlük gibi konularda tartışmaları da beraberinde getirmektedir. NLP'nin geleceği, sadece makinelerin metin üretme yeteneklerini geliştirmekle sınırlı kalmayacak; aynı zamanda insanların yazım becerilerini destekleyen ve yaratıcı süreçlere katkıda bulunan bir araç olarak evrilecektir. Ancak bu dönüşüm, insan ve makine arasındaki farkların daha fazla sorgulanmasını gerektirecek ve dilin anlamına dair yeni bir perspektif kazandıracaktır.

NLP'nin geleceği, dil modellerinin giderek daha karmaşık ve insan benzeri metinler üretebilmesiyle şekillenmektedir. Günümüzde GPT-4 ve benzeri modeller, yalnızca metin üretmekle kalmayıp, kullanıcıların belirli bir bağlamda karmaşık ihtiyaçlarını da karşılayabilmektedir. Örneğin, bir kullanıcının detaylı bir rapor yazmasını ya da yenilikçi bir hikâye oluşturmasını sağlayan bu modeller, dilin hem analitik hem de sanatsal boyutlarında derin bir anlayış sergileyebilmektedir. Ancak, bu gelişme, makinelerin insan dilini tamamen anlamaktan ziyade, dilin belirli kalıplarını taklit ettiğini unutmamızı engellememelidir. Bu durum, dil modellerinin sınırlarını anlamak ve onları doğru bağlamlarda kullanmak açısından önemlidir.

Bununla birlikte, insan ve makine arasındaki farklar giderek azalırken, bu teknolojilerin etik boyutları da daha fazla tartışılmaktadır. Yapay zekâ yazımı metinlerin doğruluğu, güvenilirliği ve tarafsızlığı, NLP'nin geleceği için önemli bir endişe kaynağıdır. Makine tarafından üretilen içeriklerin, sahte haberlerin yayılması veya yanıltıcı bilgilerin dolaşıma sokulması için kötüye kullanılma potansiyeli bulunmaktadır. Bu nedenle, NLP modellerinin gelişimi sırasında güvenilirlik, doğruluk ve etik ilkelerin sistemlere entegre edilmesi büyük önem taşımaktadır. Örneğin, modellerin çıktılarını daha şeffaf hale getirmek ve kaynaklarını açık bir şekilde belirtmek, bu tür riskleri azaltabilir.

Ayrıca NLP'nin gelişimiyle birlikte, makinelerin insan yenilikçiliğine katkı sağlayabilecek araçlar olarak nasıl kullanılabileceği konusu da önemli bir tartışma alanı haline gelmektedir. Makine öğrenimi tabanlı dil modelleri, yazarlık, çeviri ve yenilikçi yazarlık gibi alanlarda insanlara yardımcı olabilecek güçlü bir destek sunmaktadır. Özellikle içerik oluşturma sürecinde, makineler hızlı bir şekilde ilk taslaklar üretebilmekte ve kullanıcıların zamanlarını daha verimli kullanmalarını sağlayabilmektedir. Ancak bu süreçte makinelerin yenilikçi bir eser üzerinde tam bir kontrol sahibi olmamasını ve insan faktörünün her zaman sürece dahil olmasını sağlamak gereklidir.

NLP'nin geleceğinde karşılaşılan bir diğer önemli zorluk, bu teknolojilerin nasıl daha kapsayıcı hale getirileceğidir. Mevcut dil modelleri, genellikle eğitildikleri veri kümesinin sınırlamaları nedeniyle belirli bir dil veya kültürel bağlamda daha başarılıdır. Bu durum, az konuşulan dillerin veya farklı kültürel ifadelerin yeterince temsil edilmediği

sonuçlara yol açabilmektedir. NLP modellerinin daha kapsayıcı hale gelmesi için çok dilli veri kümelerinin geliştirilmesi ve modellerin bu çeşitliliğe duyarlı bir şekilde eğitilmesi gerekmektedir. Böylece, teknolojinin herkese eşit erişim ve fayda sağlaması mümkün olacaktır.

NLP'nin geleceği, insanlar ve makineler arasında iş birliğinin daha da güçlenmesini içermektedir. İnsanların dilsel ve yenilikçi yeteneklerini tamamlayıcı bir araç olarak çalışan NLP modelleri, yalnızca yazılı içerik üretiminde değil, aynı zamanda insan-makine etkileşiminde de büyük bir dönüşüm oluşturacaktır. Özellikle eğitim, müşteri hizmetleri ve sağlık gibi alanlarda bu tür modellerin insan merkezli çözümler sunması beklenmektedir. Ancak, bu dönüşümün sağlıklı bir şekilde gerçekleşmesi için karar verme süreçlerinin her zaman merkezde kalması gerektiği unutulmamalıdır.

Doğal dil işleme (NLP) alanındaki gelişmeler, insan ve makine arasındaki çizginin giderek incelmesine yol açmakta, dil teknolojilerinin potansiyelini hem fırsatlar hem de zorluklar açısından yeniden değerlendirmemizi gerektirmektedir. Dil modelleri, insan benzeri metin üretimi konusundaki yetenekleriyle yenilikçi süreçleri destekleyebilmekte, bilgiye erişimi kolaylaştırabilmekte ve çok çeşitli alanlarda verimlilik sağlayabilmektedir. Ancak, bu teknolojilerin etik kullanımı, kapsayıcılığı ve güvenilirliği sağlamak, NLP'nin gelecekteki gelişimi için hayati önem taşımaktadır.

Makinelerin insan diline olan yaklaşımı, onların yalnızca dilin yüzeysel kalıplarını öğrenmekle sınırlı olduğunu, derin anlam ve duygusal bağlam gibi insanın kendine özgü özelliklerini tam anlamıyla yansıtamadığını göstermektedir. Bu nedenle, NLP modellerinin karar mekanizmalarını tamamlayıcı bir araç olarak konumlanması, bu teknolojilerin en etkili şekilde kullanılmasını sağlayacaktır.

Sonuç olarak, NLP'nin geleceği, insanlar ve makineler arasındaki iş birliğinin daha da güçlenmesini, dil modellerinin toplumun ihtiyaçlarına daha duyarlı hale gelmesini ve dil teknolojilerinin etik sınırlar içinde kullanılarak güvenilir bir dijital ekosistem oluşturulmasını hedeflemektedir. Bu vizyonun gerçekleşmesi, yalnızca teknolojik ilerlemelerle değil, aynı zamanda dilin anlamını ve kullanımını sorgulayan çok disiplinli bir yaklaşımla mümkün olacaktır. Bu süreçte insan faktörünün ön planda tutulması, NLP'nin insanlığa en üst düzeyde katkı sağlamasının anahtarıdır.

2.2.14. İnsan-Makine Ayrımında Doğal Dil İşlemenin Rolü

Doğal dil işleme (NLP), dilin yapısal ve anlam temelli analizini sağlayarak makinelerin insan dilini anlamasını ve üretmesini mümkün kılan bir teknolojidir. İnsan ve makine tarafından üretilen metinlerin ayırt edilmesi, özellikle son yıllarda hızla gelişen yapay zekâ uygulamaları ile daha kritik bir hâl almıştır. NLP, metinlerin kökenini belirlemek için kullanılan temel bir araç haline gelirken, bu alandaki gelişmeler insan-makine ayrımını daha hassas bir biçimde yapabilme kapasitesini arttırmıştır. İnsan yazımı ve makine yazımı metinler arasındaki farkları tespit etme yeteneği, yalnızca yapay zekâ ve makine öğrenimi için değil, aynı zamanda güvenlik, etik ve sahtecilik ile mücadele gibi alanlarda da büyük bir önem taşımaktadır.

NLP, metin analizi ve sınıflandırması ile insan-makine ayrımında önemli bir rol oynamaktadır. Bu teknolojinin gelişmesiyle birlikte, metinlerdeki dilsel, yapısal ve semantik özellikler incelenerek, metnin yazarı hakkında fikir sahibi olunabilmektedir. İnsanlar genellikle daha doğal bir dil kullanırken, makineler belirli kalıpları ve algoritmalarla metin üretmektedir. Makineler bazen belirli dil özelliklerini veya stilistik unsurları takip etme konusunda zorluk yaşayabilmektedirler. Bu zorluklar, NLP algoritmalarının bu farkları tanımlamada ve çözmede başarılı olabilmesini sağlamak için önemli bir fırsat oluşturmaktadır.

Örneğin, metinlerin yazım tarzları, kelime seçimleri, cümle uzunlukları, dilin karmaşıklığı gibi unsurlar incelenerek bir metnin insan veya makine tarafından yazılıp yazılmadığına karar verilebilmektedir. Bu tür analizler, derin öğrenme modelleri ve dil modellerinin de yardımıyla daha da hassas hale gelmektedir. Transformer tabanlı modeller ve özellikle GPT gibi gelişmiş yapay zekâ modelleri, metin üretme kapasitesine sahip olup, insan benzeri metinler üretebilme yeteneğine sahiptir. Ancak, bu metinlerin makine yazımı olduğuna dair belirli izler hala mevcuttur ve NLP teknolojisi, bu izleri tespit etmekte oldukça etkilidir.

Diğer yandan NLP'nin metin tespitindeki gücü yalnızca teknolojik gelişmelerle sınırlı değildir. Eğitim veri setlerinin çeşitlenmesi, daha zengin özelliklerin çıkarılması ve daha sofistike algoritmalar kullanılması, makine ve insan yazımı metinler arasındaki farkları daha net bir şekilde belirleme imkânı sunmaktadır. Bu süreç, dijital dünyada sahte haberler, yanıltıcı içerikler ve kimlik hırsızlığı gibi kötü niyetli uygulamalarla

mücadelede önemli bir rol oynamaktadır. Metinlerin kökenini doğru bir şekilde tespit edebilmek, güvenilir bilgi akışını sağlamada temel bir adımdır.

Gelecekte, NLP teknolojilerinin insan ve makine arasındaki farkları ayırt etme kabiliyetinin daha da güçlenmesi beklenmektedir. Yeni nesil dil modelleri, insan yazımı ve makine yazımı metinler arasındaki farkları daha hassas şekilde tespit edebilecek düzeye gelecektir. Bu gelişmeler, yalnızca metin tespitinin ötesinde, daha geniş bir uygulama alanına sahip olacaktır. Özellikle siber güvenlik, eğitim, sağlık ve medya gibi alanlarda, metinlerin doğruluğu ve güvenilirliği konusunda daha etkili yöntemler sunulacaktır.

Geleceğin NLP çözümleri, sadece metinlerin kaynağını tespit etmekle kalmayacak, aynı zamanda metnin amacını, içeriğini ve niyetini de analiz edebilecek kapasiteye ulaşacaktır. Bu, insan-makine ayrımını daha derinlemesine yapabilen, daha akıllı ve etkili sistemlerin geliştirilmesini sağlayacaktır. Ancak bu süreç, etik sorumluluklar ve gizlilik endişelerini de beraberinde getirebilir. Dolayısıyla, NLP teknolojisinin evrimi, toplumsal ve etik boyutları göz önünde bulundurarak dikkatle izlenmeli ve yönlendirilmelidir.

2.3. Materyal ve Metot

Bu çalışmada kullanılan araştırma metodolojisi, Şekil 1'deki akış şemasında gösterilen dört ana adımı izlemektedir. Bu adımlar şunlardır: veri toplama, veri ön işleme, model eğitimi ve model değerlendirmesi.

Şekil 5. Metodoloji Adımları

Kaynak: Yazar tarafından oluşturulmuştur.

2.3.1. Veri Toplama

Bu çalışmada, iki farklı veri seti kullanılmıştır. Kullanılan veri setleri, fake news detection (sahte haber tespiti) amacı ile toplanmış ve hazırlanmıştır.

2.3.1.1. Veri Setinin Tanıtılması

İlk veri seti, Kaggle platformu üzerinden Fake News Detection adlı veri seti (Kaggle, n.d.) temin edilmiştir. Bu veri seti, haber makalelerini ve bu makalelerle ilgili bilgileri içermektedir. Veri seti, FakeNewsNet kaynaklıdır (FakeNewsNet, n.d.).

İkinci veri seti ise yapay zekâ kullanılarak elde edilmiştir. Bu veri seti de ilk veri setine benzer olarak haber başlıklarını içermekte olup, içeriklerin ve başlıkların gerçekliği hakkında etiketlenmiş verilerden oluşmaktadır.

Her iki veri seti de csv formatında olup bu veriler, haberlerin gerçeklik etiketlerini belirlemek için kullanılmıştır. Her haber öğesi, bir başlık (title), haberin kaynağına ait URL (news_url), haberin yayınlandığı web sitesinin alan adı (source_domain), haberin aldığı retweet sayısı (tweet_num) ve gerçeklik etiketi (real) olmak üzere beş özellikten oluşmaktadır.

Tablo 1. Veri Setlerinin Özellikleri

Title (Başlık)	Haberin başlığı
Text (Metin)	Haberin içeriği
Source (Kaynak)	Haberin kaynağı
Date (Tarih)	Haberin yayınlandığı tarih
Label (Etiket)	Haberin gerçeklik durumu.“Real“ değeri gerçek haberi,“Fake“ değeri ise sahte haberi temsil etmektedir.

Kaynak: Yazar tarafından oluşturulmuştur.

2.3.2. Veri Ön İşleme

Bu bölümde, kullanılan veri setlerinin hazırlanması ve temizlenmesi (veri ön işleme) aşamaları ayrıntılı olarak açıklanmaktadır. Ön işleme ve terim azaltma adımları, Python programlama dili kullanılarak gerçekleştirilmiştir.

Kaggle platformu üzerinden Fake News Detection veri setinin ön işlemesi aşağıdaki adımlarla gerçekleştirilmiştir.

2.3.2.1. Veri Seti Birleştirme

GossipCop ve PolitiFact kaynaklarından elde edilen veri setleri birleştirilmiştir. Ardından, veri seti örneklerinin rastgele karşılaştırılmasıyla denge sağlanmıştır.

2.3.2.2. URL'den Alan Adı Çıkarma

Her bir haber örneğinin URL'inden alan adı çıkarılmıştır. Bu işlem, haberlerin hangi kaynaklardan geldiğini belirlemek için yapılmıştır.

2.3.2.3. Tweet Sayısını Hesaplama

Her bir haber örneğinin haberin sosyal medyada kaç kez paylaşıldığını gösteren tweet sayısı hesaplanmıştır.

2.3.2.4. Eksik Verilerin Kaldırılması

Veri setinde eksik değerler olabilir. Bu nedenle, veri setinde bulunan eksik değerler kaldırılabilir veya ortalama, mod ve medyan değerleri ile doldurulabilir.

Kullanılan veri setinde eksik veri tespit edilmemiştir. Dolayısıyla, bu adım ilk veri seti için uygulanmamıştır.

2.3.2.5. Durak Kelimelerinin Kaldırılması

Haber başlıklarındaki stopwords'ler kaldırılmıştır.

Stopwords'ler (durak kelimeleri), dilde sıklıkla kullanılan fakat genellikle anlamsal olarak önemsiz olan kelimelerdir. Örnek olarak Türkçe de kullanılan „ve", „veya", „ama", gibi kelimeler verilebilir. Bu kelimeler, bir metnin anlamını çıkarmak için genellikle gerekli değildir ve analiz esnasında dikkate alınmazlar. Bu nedenle stopwords'lerin kaldırılması, metnin daha temiz ve anlamlı kelimelerle temsil edilmesini sağlamaktadır.

2.3.2.6. Metin Temizlenmesi

Haber başlıklarındaki özel karakterler, sayılar ve diğer istenmeyen öğeler temizlenmiştir.

Metin temizleme adımı, metindeki gereksiz karakterleri ve özel işaratleri kaldırmak için olan adımdır. Bu işaretler genellikle analiz esnasında anlam ifade etmeyen veya istenmeyen gürültü olarak kabul edilir. Örnek olarak, noktalama işaretleri, rakamlar veya diğer özel karakterler metnin anlamını değiştirmeden kaldırılır.

2.3.2.7. Küçük Harfe Dönüştürme

Haber başlıkları küçük harfe dönüştürülmüştür.

Metin analizi esnasında büyük ve küçük harflerin aynı kelime olarak kabul edilmesi önemlidir. Örneğin, „Hello“ ve „hello“ aynı kelime olarak kabul edilmelidir. Dolayısıyla, metindeki bütün harfler küçük harfe dönüştürülür. Böylece, kelime sayısını azaltarak ve kelime tekrarlarını birleştirerek daha iyi bir analiz yapılır.

2.3.2.8. Tokenizasyon

Haber başlıkları tokenlara ayrılmıştır. Tokenler, haber başlıklarının kelimelerine bölünmüştür.

Tokenizasyon, metni kelimelerle veya „token“lara ayırma işlemidir. Metin, kelimeler, noktalama işaretleri ve boşuklar gibi parçalara bölünür. Bu adım, metin analizinde her bir kelimenin ayrı ayrı ele alınmasını ve işlenmesini sağlar. Tokenlar, daha sonra metin analizi ve makine öğrenmesi algoritmalarında kullanılabilir.

Bu adımlar, metin verilerini daha işlenebilir bir formatta olmasını sağlamaktadır.

2.3.3. Model Eğitimi

Makine öğrenimi algoritmaları, doğal dil işleme için ham metin verilerini doğrudan işleyemezler. Bu nedenle, ham metin verileri sayısal değerlere dönüştürülerek ele alınabilir. Kelim gömme (word embedding), bir kelime dağarcığındaki kelimeleri veya kelime öbeklerini gerçek sayı vektörleriyle eşleştiren bir dizi tekniktir (Goldberg ve

Levy, 2014). Bu teknik, anlamsal olarak benzer kelimelerin yakın vektörlere atanacağı fikrine dayanır, böylece model bazı kelimeler hakkında öğrenilen bilgileri diğer benzer kelimelere aktarabilir (Rong, 2014). Bu, metnin çok daha kullanışlı ve anlamlı bir temsilini sağlar.

Model eğitimi için Word2Vec mimarisinden faydalanılmıştır. Bu mimari, kelimeleri vektör uzayına dönüştürmek için kullanılır (Mikolov vd., 2013). Kelime vektörlerini hesaplamak için kullanılan bir derin öğrenme yöntemidir. Bu model, metin verilerindeki kelimeler arasındaki ilişkileri gösteren vektörler oluşturur. Modelin oluşturulması, metinlerin semantik benzerliklerini ve ilişkilerini anlamak için önemlidir (Goldberg ve Levy, 2014).

Bu çalışmada, Word2Vec algoritması kullanılarak kelime gömme işlemi gerçekleştirilmiş olup elde edilen sonuçlar üzerinden makine öğrenmesi algritmaları ile sınıflandırma yapılmıştır. Bu yöntemlerin kullanılmasıyla, metin verileri üzerinde daha etkili bir analiz yapılabilmiştir.

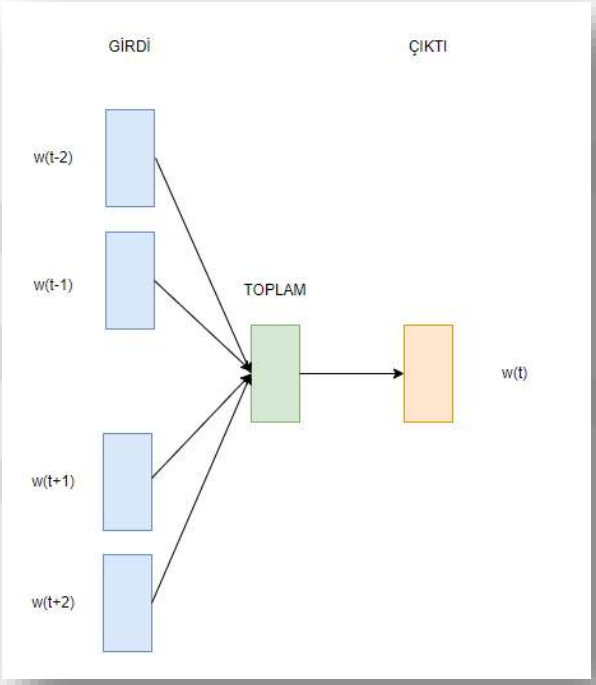
LSTM ve SVM sınıflandırma modellerinin performansı karşılaştırılarak değerlendirilmiştir. Farklı metrikler (doğruluk, hassasiyet vb.) kullanılarak modellerin etkinliği değerlendirilmiş ve sonuçlar karşılaştırmalı bir şekilde sunulmuştur.

2.3.3.1. Word2Vec Mimarisi

2013 yılında Google araştırmacısı Thomas Mikolov ve ekip arkadaşları tarafından geliştirilen ve yayınlanan kelime gömme algoritmalarından bir tanesidir (Mikolov, 2013). Word2Vec modeli ile kelimelerin eğitilmesi amaçlanmıştır. Thomas ve ekip arkadaşları kelimeleri belirli bir pencere boyutunda kelimelere göre analiz etmek için Word2Vec kelime gömme mimarisini tasarlamışlardır. Word2Vec, doğal dil işleme alanında kelimeleri sürekli bir vektör uzayında temsil etmek için önemli ve popüler bir yöntemdir. Kelime gömme (word embedding), kelime dağarcığını bir vektör uzayına dönüştürerek kelimeler arasındaki anlam bilgisini koruyan matematiksel temsiller oluşturmayı amaçlamaktadır. Bu modeldeki pencere boyutu, kelime vektörlerinin nasıl oluşturulacağını belirleyen önemli bir parametredir. Pencere boyutu, hedef kelimenin etrafındaki bağlamı belirler ve bu bağlamdaki kelimelerin kullanılacağı alanı tanımlamaktadır.

Word2Vec mimarisi, iki farklı yaklaşım olan CBOW (Continuous Bag of Words) ve Skip Gram modellerini içermektedir.

CBOW, NLP alanında kullanılmakta olan kelime gömme modelidir. Bu model, bir kelimenin etrafındaki bağlamı kullanarak hedef kelimeyi tahmin etmeye çalışır. Bu modelde, bir cümle veya metin penceresinin içindeki diğer kelimeler kullanılarak hedef kelime tahmin edilmektedir. CBOW modeli genellikle daha hızlı eğitilir ve küçük veri setleri için daha iyi performans göstermektedir.



Şekil 6. CBOW Modeli

Kaynak: Yazar tarafından oluşturulmuştur.

w(t) cümlenin merkezindeki kelimeyi, yani tahmin edilmek istenen çıktıyı temsil etmektedir. w(t-2)… w(t+2) ise seçilen pencere boyutuna bağlı olarak ortada olmayan diğer kelimelerdir.

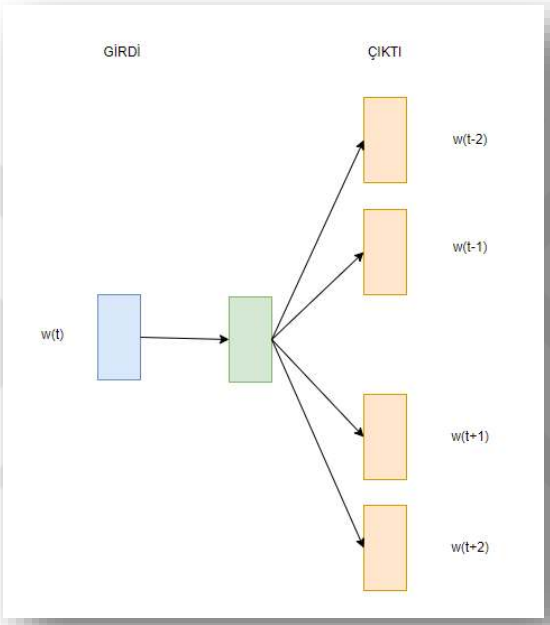
Vektörler arasındaki benzerlik, koşullu olasılık formülüne göre hesaplanır. İlgili formül denklem (1)'de gösterilmiştir.

$$P(W_{t+j}| W_t) = \exp(U^T_{t+j} V_{wt})|\sum^{v}_{w=1} \exp(U^T_{wt}) \quad (1)$$

Veri setini eğitirken, eğitim sırasında CBOW mimarisi için optimize edilecek hata fonksiyonu denklem (2)‘de gösterilmiştir.

$$J = -\frac{1}{T}\Sigma_{t=1}^{T}\Sigma_{-m\leq j\leq m} logP(wt + j|wt) \text{ (2)}$$

Skip Gram, bir hedef kelime kullanılarak etrafındaki bağlamı tahmin etmeye çalışır. Skip Gram modeli ise daha büyük veri setleri için daha iyi performans göstermektedir ve nadiren CBOW modelinden daha yavaş eğitilir.



Şekil 7. Skip Gram Modeli

Kaynak: Yazar tarafından oluşturulmuştur.

w(t), cümlenin merkezindeki girdi(input) değerini temsil ederken, w(t-2)…w(t+2) ile gösterilen değerler, cümlenin merkezinde olmayan ve tercih edilen pencere boyutuna göre tahmin edilmek istenen çıktı(output) değerlerini temsil etmektedir.

Veri setini eğitirken, eğitim sırasında Skip Gram mimarisi için optimize edilecek hata fonksiyonu denklem (3)’te gösterilmiştir.

$$J = -\frac{1}{T}\Sigma_{t=1}^{T}\Sigma_{-m\leq j\leq m} logP(wt|wt + j) \text{ (3)}$$

Skip Gram ve CBOW modelleri arasındaki temel fark, CBOW modelinin bir kelimenin etrafındaki bağlamı kullanarak hedef kelimeyi tahmin etmeye çalışması, Skip

Gram modelinin ise bir hedef kelime kullanarak etrafındaki bağlamı tahmin etmeye çalışmasıdır.

Temel olarak Word2Vec mimarisi, belirli bir bağlamda bir araya gelen kelimelerin benzer anlamlara sahip olduğu dağılım hipotezine dayanmaktadır. Her kelimenin belirli bir vektör uzayındaki temsilini oluşturarak benzer anlamlı kelimelerin benzer vektörlere sahip olmasını sağlar.

Word2Vec mimarisi bir dizi nedenden dolayı popüler hale gelmiş ve birçok farklı NLP uygulamasında kullanılmaktadır.

Anlamsal Temsiller: Word2Vec, kelimeler arasındaki anlamsal bağlantıları kaydetmektedir. Benzer anlamlı kelimeler birbirlerine yakın vektör uzayında temsil edilmektedir. Bu, modelin kelimeleri belirli bir kapsam içindeki bağlamına göre yorumlamasını sağlamaktadır.

Dağılımsal Anlam: Word2Vec'in temelini, benzer anlamlı kelimlerin benzer bağlamlarda daha sık görünme eğilimi olan dağılımsal hipotez oluşturulmaktadır. Word2Vec, büyük bir korpusta kelimelerin dağılım desenlerinden öğrenerek, anlamsal benzerlikleri yansıtan vektör temsilleri üretmektedir.

Vektör Aritmetiği: Word2Vec, cebirsel özelliklere sahip vektör temsilleri üretmektedir. Örneğin, vektör aritmetiği kullanılarak kelime ilişkileri kaydedilebilir. İyi bilinen bir örnek, „kraliçe"nin vektör temsilinin, „kral"ın vektör temsilinden „erkek" eksi „kadın" eklendiğinde benzer olmasıdır.

Verimlilik: Word2Vec'in yüksek hesaplama verimliliği, büyük veri kümelerinde eğitimi mümkün kılmaktadır. Geniş bir kelime dağarcığı için yüksek boyut vektör temsillerini öğrenmek için bu verimlilik gereklidir.

Transfer Öğrenme: Önceden eğitilmiş Word2Vec modelleriyle çeşitli doğal dil işleme görevleri başlatılabilmektedir. Büyük bir veri kümesinde keşfedilen gömme işlemlerinin belirli kullanımlar için ayarlanması, zaman ve kaynakların tasarruf edilmesini sağlamaktadır.

Uygulamalar: Word2Vec kelime gömme işlemleri, makine çevirisi, metin sınıflandırması, duygu analizi ve bilgi erişimi gibi birçok doğal dil işleme uygulamasında

umut vaat etmektedir. Bu uygulamalar, anlamsal ilişkileri yakalama yeteneklerinden dolayı başarılıdır.

Ölçeklenebilirlik: Word2Vec büyük veri kümeleriyle rahatça başa çıkabilir ve ölçeklenebilir. Bu tür ölçeklenebilirlik, büyük metin veri kümelerinde eğitim için önemlidir.

Açık Kaynak Uygulamalar: Word2Vec modelinin açık kaynaklı sürümleri bulunmaktadır. Bunlardan bir tanesi Gensim kütüphanesinde bulunmaktadır. Geniş bir şekilde benimsenmesi ve hem araştırmada hem de endüstride kullanılması, erişilebilirliğine katkıda bulunan faktörlerden biridir.

2.3.3.2. LSTM Algoritması

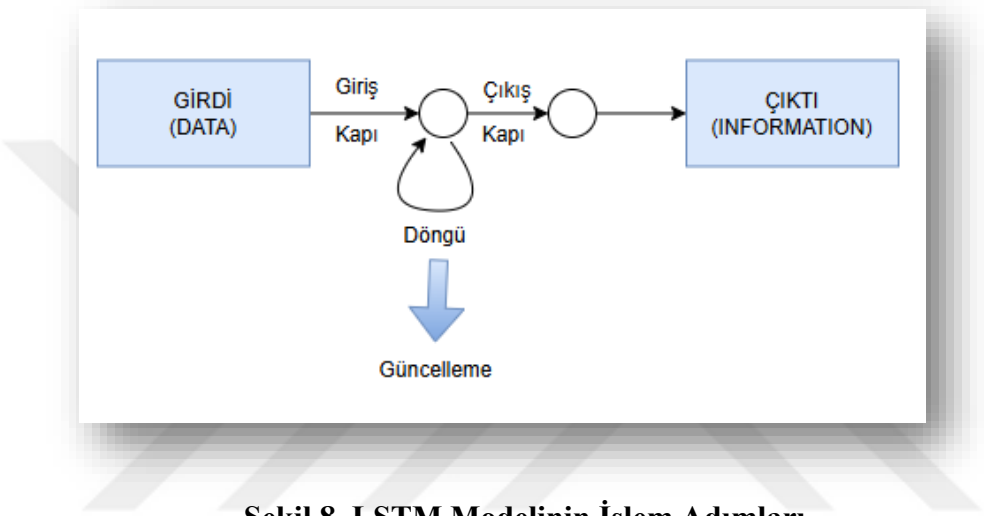
Uzun Kısa Süreli Bellek (Long Short-Term Memory, LSTM) algoritması, yinelemeli sinir ağlarının (Recurrent Neural Networks, RNN) bir türü olarak geliştirilmiştir (Hochreiter ve Schmidhuber, 1997). RNN'lerin klasik sürümleri zaman serileri ve doğal dil işleme (NLP) gibi ardışık verilerle çalışırken, uzun bağımlılıkları öğrenme konusunda zorluklar yaşamaktadır. Bu tür problemler, RNN'lerin zaman içinde önemli bilgileri unutmasıyla sonuçlanan "vanishing gradient" (azalan gradyan) sorunu olarak bilinir. LSTM bu soruna çözüm sunmak amacıyla geliştirilmiş ve ağın önemli bilgileri daha uzun süre saklayabilmesini sağlamıştır. İlk olarak 1997 yılında Hochreiter ve Schmidhuber tarafından önerilen LSTM, zaman serisi analizlerinden dil modellerine kadar geniş bir kullanım alanına sahiptir. Özellikle dil modelleme, makine çevirisi ve konuşma tanıma gibi NLP problemlerinde yaygın olarak tercih edilmektedir (Sutskever, Vinyals, ve Le, 2014) .

LSTM'in temel farkı, her hücrede bulunan özel yapılar sayesinde bilgiyi uzun süre saklayabilmesidir. Standart bir LSTM hücresi, girdiyi kontrol etmek ve hatırlanacak bilgiyi seçmek için kapılar (gates) kullanmaktadır. Bu kapılar, verinin ne kadarının saklanacağını veya unutulacağını belirleyerek, modelin ardışık veri içinde önemli bilgiyi öğrenmesini sağlar. Üç temel kapı bulunmaktadır:

Unutma Kapısı (Forget Gate): Bu kapı, önceki zaman adımından gelen bilgilerin ne kadarının saklanacağına karar verir. Aktivasyon fonksiyonu olarak sigmoid kullanılarak, [0,1] aralığında bir ağırlık atanır ve bilgi gerektiğinde unutulabilir.

Giriş Kapısı (Input Gate): Yeni bilgi eklenirken, bu kapı hangi bilgilerin mevcut hücre durumuna ekleneceğine karar verir. Hücre durumunun güncellenmesi, girdiden gelen yeni bilginin filtrelenmesi ile gerçekleştirilir. Sigmoid ve tanh fonksiyonları kullanılarak, hücreye eklenen bilginin boyutları düzenlenir.

Çıkış Kapısı (Output Gate): Modelin o anki çıktısını belirlemek için kullanılır. Çıkış kapısının ürettiği bilgi, sıradaki katman veya zaman adımı için kullanılmak üzere aktarılır.



Şekil 8. LSTM Modelinin İşlem Adımları

Kaynak: Yazar tarafından oluşturulmuştur.

LSTM'in bu kapı mekanizmaları sayesinde model, ardışık verilere dayalı uzun süreli bağımlılıkları daha iyi öğrenir. LSTM, zaman içinde hatırlanması gereken bilgileri tutarken gereksiz olanları unutabilir. Böylece, geleneksel RNN modellerinde karşılaşılan gradyan kaybolma problemi minimize edilmektedir.

LSTM algoritması, özellikle doğal dil işleme ve zaman serisi tahmini gibi uygulamalarda başarılı sonuçlar vermektedir. NLP alanında, dil modeli oluşturma, metin özetleme, duygu analizi, makine çevirisi ve konuşma tanıma sistemlerinde sıklıkla kullanılmaktadır. Ayrıca finansal veri analizi, hava tahminleri ve biyomedikal zaman serisi verileri gibi alanlarda da tercih edilmektedir. LSTM, girdilerin zaman içerisindeki etkilerini öğrenebilmesi sayesinde, diğer yöntemlere göre daha tutarlı performans sunmaktadır. Bununla birlikte, modelin daha karmaşık yapısı nedeniyle eğitim süreci daha uzun olabilir ve daha fazla hesaplama gücü gerektirebilir.

Sonuç olarak, LSTM algoritması, geleneksel RNN'lerin sınırlarını aşarak ardışık veri işleme konusunda güçlü bir çözüm sunmaktadır. Özellikle uzun süreli bağımlılıkların olduğu verilerde performansı yüksektir ve çeşitli uygulama alanlarında başarılı sonuçlar elde edilmiştir.

2.3.3.3. SVM Algoritması

Destek Vektör Makineleri (Support Vector Machines, SVM), hem sınıflandırma hem de regresyon problemlerinde kullanılan güçlü ve esnek bir makine öğrenmesi algoritmasıdır (Türkoğlu, 2021). SVM'nin temel amacı, veriyi farklı sınıflara en iyi şekilde ayıran hiper-düzlemi (hyperplane) bulmaktır. Bu hiper-düzlem, iki sınıf arasındaki en büyük marjini (margin) sağlayacak şekilde seçilmektedir. Yani, sınıflar arasındaki ayrımı maksimize eden ve sınıflandırmayı optimize eden bir matematiksel yapı oluşturulur.

SVM, özellikle doğrusal olmayan sınıflandırma problemleri için geliştirilmiş çekirdek yöntemi (Kernel trick) sayesinde yüksek başarı oranları sunmaktadır. Bu yöntem, veriyi daha yüksek boyutlu uzaylara haritalayarak doğrusal olarak ayrılamayan verilerin bile kolayca sınıflandırılabilmesini sağlamaktadır. Bu yönüyle SVM hem doğrusal hem de doğrusal olmayan verilerle çalışabilme kapasitesine sahiptir (Akdağlı, 2021).

Destek Vektör Makineleri algoritmasının çalışma prensibi iki ana adıma dayanmaktadır:

İki sınıfı ayırmak için bir hiper-düzlem belirlenir. Bu hiper-düzlem, veriye bağlı olarak bir doğrusal model oluşturur. Doğru hiper-düzlemin seçilmesi, iki sınıf arasındaki en büyük mesafeyi (marjini) sağlamakla ilgilidir. Marjin, sınıfların sınırları ile hiper-düzlem arasındaki mesafeyi ifade eder. Marjini maksimize etmek, modelin daha genel bir sınıflandırma yapmasını ve aşırı öğrenme (overfitting) sorunlarını azaltmasını sağlar.

Sınırlayıcı noktalar, yani sınıfların en yakın örnekleri destek vektörleri olarak adlandırılır. Bu destek vektörleri, hiper-düzlemin pozisyonunu ve yönünü belirlemede kritik rol oynamaktadır. Hiper-düzlem, bu destek vektörlerine dayalı olarak hesaplanmaktadır. Sınıf sınırlarını belirleyen bu noktalar, modelin hem eğitim hem de test performansı üzerinde doğrudan etkilidir.

Çoğu gerçek dünya problemi doğrusal olmayan sınıflandırma gerektirmektedir. SVM, Kernel fonksiyonları kullanarak bu problemleri çözer. Kernel fonksiyonları, veriyi daha yüksek boyutlu bir uzaya taşıyarak doğrusal olmayan ayrımları doğrusal hale getirir. En sık kullanılan Kernel türleri aşağıdaki gibidir.

Doğrusal Kernel (Linear Kernel): Verilerin doğrusal olarak ayrılabildiği durumlarda kullanılır.

$$k(x_1,x_2) = x_1.x_2$$

Polinomsal Kernel (Polynomial Kernel): Veriler arasındaki daha karmaşık ilişkileri öğrenmek için uygundur.

$$k(x_1,x_2) = (\gamma\, x_1.x_2+c)^d$$

Radyal Tabanlı Fonksiyon (Radial Basis Function, RBF): Doğrusal olmayan problemlerde en çok tercih edilen Kernel türüdür.

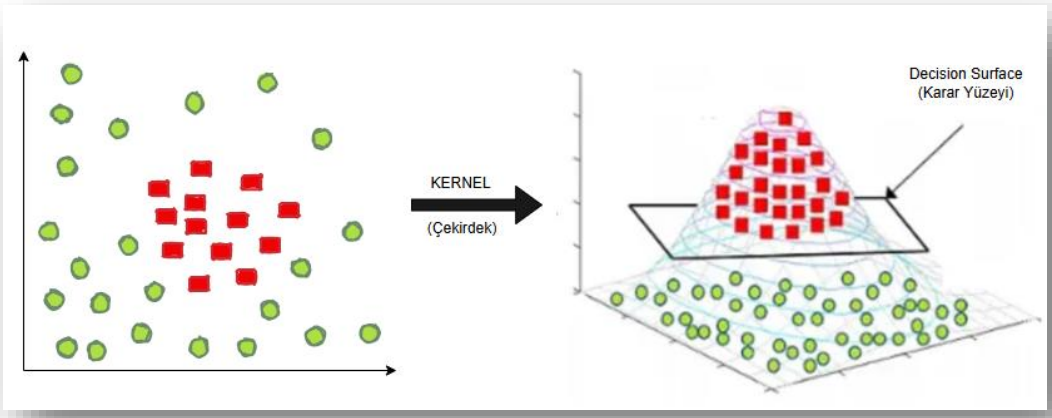
$$k(x_1,x_2) = exp(-\gamma\, ||x_1-x_2||^2)$$

Sigmoid Kernel: Genellikle nöral ağlarla ilişkilendirilir ve belirli durumlarda tercih edilir.

$$k(x_1,x_2) = tanh(\gamma x_1.x_2+c)$$

Kernel yöntemi, sınıflandırma problemlerinde esneklik sağlayarak SVM'nin farklı türdeki verilere uyum sağlamasını mümkün kılar.

SVM çekirdeği (Kernel), verilerin düzenlenmesi ve sınıflandırma sürecinin kolaylaştırılması amacıyla kullanılan matematiksel bir işlev olarak tanımlanmaktadır (Jain, 2024). Destek Vektör Makineleri (SVM), veri sınıflarını birbirinden ayıran en uygun hiper-düzlem bulmayı amaçlayan bir algoritma olarak öne çıkmaktadır. Ancak, birçok gerçek dünya probleminin verileri, orijinal özellik uzayında doğrusal olarak ayrılamamaktadır. Bu bağlamda, çekirdek fonksiyonlarının, verileri daha yüksek boyutlu bir uzaya dönüştürerek doğrusal bir ayrım yapılmasını mümkün kıldığı ifade edilmektedir. Böylelikle, karmaşık ve doğrusal olmayan veri kümelerinin sınıflandırılması sağlanmaktadır.



Şekil 9. SVM Kernel

Kaynak: Jain, A. (2024, 11 Eylül). SVM kernels and its type. *Medium*. https://medium.com/@abhishekjainindore24/svm-kernels-and-its-type-dfc3d5f2dcd8

Orijinal özellik uzayı, verinin giriş özellikleriyle tanımlandığı bir ortam olarak tanımlanmaktadır. Örneğin, bir veri seti boy ve kilo gibi iki özellik içeriyorsa, bu özellikler iki boyutlu bir uzayda temsil edilmektedir. Ancak, bu iki boyutlu uzayda doğrusal olarak ayrılamayan veri yapılarının varlığı durumunda, çekirdek fonksiyonları devreye girmektedir. Bu fonksiyonlar, veriyi daha yüksek boyutlu bir uzaya dönüştürerek, ayrım yapılabilir hale getirmektedir. Yüksek boyutlu bir uzaya yapılan bu dönüşüm sayesinde, orijinal özellik uzayında karmaşık ilişkiler gösteren veriler, daha basit ve ayrılabilir bir yapıya bürünmektedir.

Çekirdek fonksiyonlarının bu dönüşümü gerçekleştirirken sağladığı önemli bir avantaj, hesaplama verimliliği olarak belirtilmektedir. SVM algoritması, çekirdek yöntemi (Kernel trick) adı verilen bir teknikle, dönüşümü doğrudan hesaplamadan dolaylı olarak gerçekleştirmektedir. Bu teknik, yalnızca veri noktaları arasındaki benzerlikleri hesaplayarak, dönüşüm sürecini etkili bir şekilde tamamlamaktadır.

SVM'de çekirdek fonksiyonlarının kullanılmasının birçok avantaj sunduğu ifade edilmektedir:

Doğrusal Olmayan Verilerin Ele Alınması: Çekirdek fonksiyonları, doğrusal olmayan veri kümelerinin sınıflandırılmasını mümkün kılmaktadır. Orijinal özellik

uzayında ayrım yapılamayan bu tür veriler, daha yüksek boyutlu bir uzaya dönüştürülerek doğrusal bir hiper-düzlemle ayrılabilir hale getirilmektedir.

Esneklik: Farklı veri tiplerine ve problemlerine göre çeşitli çekirdek fonksiyonlarının seçilebilmesi, SVM'nin esnekliğini artırmaktadır. Örneğin, polinomsal çekirdek, RBF (Radial Basis Function) ve sigmoid çekirdek gibi farklı fonksiyonlar, farklı türde problemlere uygun çözümler sunmaktadır.

Özellik Çıkarımı: Çekirdek fonksiyonları, veri içerisindeki karmaşık ilişkileri açığa çıkararak, modelin daha yüksek bir sınıflandırma performansı sergilemesini sağlamaktadır. Verilerin daha yüksek boyutlu bir uzaya dönüştürülmesi, özellikler arasındaki gizli ilişkilerin keşfedilmesine olanak tanımaktadır.

SVM'nin çekirdek yönetimi kullanılarak dönüşüm sürecini dolaylı olarak gerçekleştirdiği ifade edilmektedir. Yüksek boyutlu uzaya yapılan dönüşüm, veri noktalarının yalnızca benzerliklerini hesaplamayı gerektirdiği için, doğrudan dönüşüme kıyasla çok daha az hesaplama maliyetiyle tamamlanmaktadır. Bu sayede hem büyük veri setleriyle çalışabilmekte hem de doğrusal olmayan ilişkiler etkili bir şekilde ele alınabilmektedir.

Çekirdek fonksiyonunun uygulanmasıyla, veri noktalarının daha yüksek boyutlu bir uzayda doğrusal olarak ayrılabilir hale geldiği belirtilmektedir. Özellikle karmaşık, doğrusal olmayan veri yapılarında bu dönüşüm, SVM'nin sınıflandırma başarısını önemli ölçüde artırmaktadır.

SVM çekirdek fonksiyonları, doğrusal olmayan sınıflandırma problemlerinin çözümünde kritik bir rol oynamaktadır. Bu fonksiyonlar, veri setini daha yüksek boyutlu bir uzaya dönüştürerek, ayrılabilir hale getirmekte ve hesaplama sürecini verimli bir şekilde tamamlamaktadır. Çekirdek yöntemi, SVM'nin doğrusal olmayan ilişkiler içeren veri setleri üzerinde başarılı sonuçlar üretmesine olanak tanımakta ve bu sayede geniş bir uygulama alanı bulmaktadır. Destek Vektör Makineleri'nin farklı çekirdek fonksiyonlarıyla birlikte sunduğu esneklik, algoritmanın çok yönlü bir araç olarak kullanılmasını mümkün kılmaktadır.

Destek Vektör Makineleri'nin (SVM), farklı çekirdek fonksiyonlarıyla geniş bir esneklik sunarak çeşitli problem türlerine uyum sağlayabilmesi, algoritmanın güçlü yönlerini oluşturduğu gibi, uygulama alanlarının çeşitliliğini de artırmaktadır. Bu

bağlamda, SVM'nin performansını etkileyen avantaj ve dezavantajlarının yanı sıra, algoritmanın kullanım alanları detaylı bir şekilde değerlendirilmiştir.

SVM algoritması, özellikle küçük ve orta boyutlu veri kümelerinde yüksek performans sergilemesiyle öne çıkmaktadır. Bu algoritmanın marjin optimizasyonuna dayalı doğası, genelleme yeteneği yüksek modeller üretmesini sağlamaktadır. Sınıflandırma sırasında, sınıf ayrımını maksimize eden bir hiper-düzlem oluşturularak aşırı öğrenme (overfitting) riski minimize edilmektedir. Bununla birlikte, SVM'nin kernel fonksiyonları sayesinde doğrusal olmayan veri yapılarında da etkili çözümler üretebilmesi, esneklik sağlayan en önemli özelliklerden biridir.

Özellikle marjin optimizasyonuna dayalı yaklaşımı, verilerin ayrımını daha sağlam hale getirirken, farklı çekirdek fonksiyonlarının (örneğin, polinomsal, RBF veya sigmoid çekirdek) tercih edilebilmesi, algoritmanın çok yönlü yapısını daha da güçlendirmektedir.

Her ne kadar SVM, genelleme yeteneği açısından güçlü bir algoritma olarak görülse de büyük veri kümeleri üzerinde çalışırken çeşitli zorluklar ortaya çıkmaktadır. Bu zorlukların başında, eğitim süresinin uzaması ve bunun sonucunda hesaplama maliyetinin artması yer almaktadır. Özellikle büyük ölçekli veri setlerinde, algoritmanın uygulanabilirliğini sınırlayan bu durum, SVM'nin dikkat edilmesi gereken zayıf yönlerinden biridir.

Bunun yanı sıra, SVM'nin başarısı, hiper parametre ayarları ve çekirdek fonksiyonlarının doğru bir şekilde belirlenmesine bağlıdır. Kernel türünün yanı sıra, düzenleme parametresi ve gamma değeri gibi hiper parametrelerin seçimi, deneyim ve uzmanlık gerektiren bir süreçtir. Ayrıca, SVM algoritması, giriş özelliklerinin aynı ölçeklerde olmasını gerektirdiğinden, öznitelik ölçekleme adımı veri ön işleme aşamasında kritik bir önem taşımaktadır.

Destek Vektör Makineleri, farklı sektörlerde çok çeşitli uygulama alanlarına sahiptir. Bu alanlar şu şekilde özetlenebilir:

Doğal Dil İşleme (NLP): SVM, metin sınıflandırma, duygu analizi ve spam tespiti gibi görevlerde başarıyla kullanılmaktadır. Algoritmanın marjin optimizasyonuna dayalı yapısı, yüksek doğruluk oranları elde edilmesini sağlamaktadır.

Biyoinformatik: Genetik verilerin sınıflandırılması ve hastalık teşhisinde SVM, etkili bir çözüm sunmaktadır. Özellikle, karmaşık biyolojik veri setlerinde, doğrusal olmayan ilişkileri çözme yeteneği, algoritmanın bu alandaki önemini artırmaktadır.

Görüntü İşleme: Yüz tanıma, nesne sınıflandırma gibi uygulamalarda SVM, yüksek performans göstermektedir. Çekirdek fonksiyonlarının sağladığı esneklik, görüntü özelliklerinin etkin bir şekilde işlenmesini mümkün kılmaktadır.

Finans: SVM, hisse senedi fiyat tahmini ve kredi risk analizi gibi finansal problemlerde kullanılmaktadır. Modelin genelleme yeteneği, finansal veri analizi ve tahmin süreçlerinde önemli bir avantaj sağlamaktadır.

Destek Vektör Makineleri gerek doğrusal gerek doğrusal olmayan sınıflandırma problemlerinde etkili bir algoritma olarak, geniş bir uygulama yelpazesine sahiptir. Özellikle kernel fonksiyonları ile sağlanan esneklik, algoritmanın birçok sektörde kullanılabilirliğini artırmaktadır. Bununla birlikte, hiperparametre seçimi, veri ölçekleme ve büyük veri kümeleriyle çalışırken ortaya çıkan hesaplama maliyeti gibi zorluklar göz önünde bulundurulmalıdır. Doğru bir veri ön işleme ve hiperparametre optimizasyonuyla SVM, genelleme yeteneği yüksek, güvenilir bir model sunarak, makine öğrenimi projelerinde güçlü bir araç olarak değerlendirilmektedir.

3. UYGULAMA

Bu çalışmada geliştirilen uygulama, insan ve makine tarafından yazılmış metinlerin karşılaştırılması ve tespiti amacıyla tasarlanmıştır ve iki fazdan oluşmaktadır. İlk fazda, ham metin verisi üzerinde yapılan veri ön işleme işlemleriyle verinin temizlenmesi, normalize edilmesi ve tokenlara ayrılması sağlanmıştır. Bu adımda, metinlerin anlamlı hale getirilmesi amacıyla Word2Vec algoritması kullanılarak kelime vektörleri oluşturulmuştur. Ayrıca, sezgisel yöntemlerden yararlanarak, genetik algoritmalar gibi tekniklerle öznitelik çıkarımı yapılmış ve LSTM ile SVM algoritmaları üzerinde her bir input'un output üzerindeki etkisi analiz edilmiştir. Bu analizler sonucunda, başarılı olan özellikler seçilerek, metinlerin insan mı yoksa makine mi tarafından yazıldığına dair sınıflandırma işlemi gerçekleştirilmiştir.

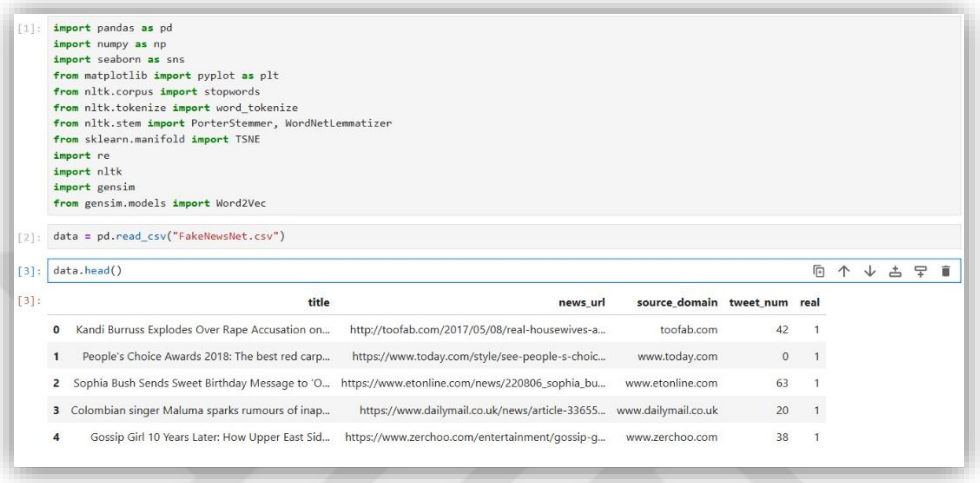
İkinci fazda ise, ilk fazda elde edilen en etkili özellikler ve algoritmalar kullanılarak hibrit bir model oluşturulacaktır. İlk fazda birçok input ile eğitilen model, seçilen özelliklerin sayısını azaltarak daha etkili ve verimli bir model oluşturulacaktır. Bu aşamada, input sayısı azaltılarak modelin performansı daha iyi bir şekilde optimize edilecek ve yeniden eğitim süreci başlatılacaktır. Böylece, daha güçlü ve verimli bir hibrit model elde edilmesi hedeflenmektedir. Bu uygulama, veri ön işleme, özellik çıkarımı ve makine öğrenmesi modellerinin birleşimiyle, insan ve makine yazımı metinler arasındaki farkları tespit etmeyi amaçlamaktadır. Her adımda kullanılan metodolojiler ve araçlar detaylı bir şekilde açıklanacak olup, elde edilen sonuçların model başarısı üzerindeki etkileri tartışılacaktır.

3.1. Veri Ön İşleme

Veri ön işleme, makine öğrenmesi ve doğal dil işleme (NLP) projelerinin temel adımlarından biridir. Bu adım, modelin başarısını doğrudan etkileyen kritik bir süreçtir. Bu çalışmada, ham veriden anlamlı bilgiler çıkarmak için bir dizi adım uygulanmıştır.

3.1.1. Ham Verinin Yüklenmesi ve İncelenmesi

İlk olarak, "FakeNewsNet" veri kümesi, pandas kütüphanesi ile okunarak data adında bir veri çerçevesine (DataFrame) aktarılmıştır. Bu aşamada, veri kümesindeki ilk birkaç satır data.head() fonksiyonu ile görüntülenmiş ve veri hakkında genel bir fikir edinilmiştir.



```
[1]: import pandas as pd
     import numpy as np
     import seaborn as sns
     from matplotlib import pyplot as plt
     from nltk.corpus import stopwords
     from nltk.tokenize import word_tokenize
     from nltk.stem import PorterStemmer, WordNetLemmatizer
     from sklearn.manifold import TSNE
     import re
     import nltk
     import gensim
     from gensim.models import Word2Vec

[2]: data = pd.read_csv("FakeNewsNet.csv")

[3]: data.head()
```

[3]:

	title	news_url	source_domain	tweet_num	real
0	Kandi Burruss Explodes Over Rape Accusation on...	http://toofab.com/2017/05/08/real-housewives-a...	toofab.com	42	1
1	People's Choice Awards 2018: The best red carp...	https://www.today.com/style/see-people-s-choic...	www.today.com	0	1
2	Sophia Bush Sends Sweet Birthday Message to 'O...	https://www.etonline.com/news/220806_sophia_bu...	www.etonline.com	63	1
3	Colombian singer Maluma sparks rumours of inap...	https://www.dailymail.co.uk/news/article-33655...	www.dailymail.co.uk	20	1
4	Gossip Girl 10 Years Later: How Upper East Sid...	https://www.zerchoo.com/entertainment/gossip-g...	www.zerchoo.com	38	1

Şekil 10. Ham Verinin Yüklenmesi

Kaynak: Yazar tarafından oluşturulmuştur.

3.1.1.1. Gereksiz Sütunların Çıkarılması

Veri kümesinde, analize katkı sağlamayan veya model için gereksiz olan bazı sütunlar bulunmaktadır. Bu sütunlar (örneğin, "index", "news_url", "source_domain", "tweet_num") drop() fonksiyonu ile veri kümesinden çıkarılmıştır. Bu adım, veri kümesinin daha yönetilebilir ve analize uygun hale gelmesini sağlamaktadır.

data.reset_index(inplace=True)

data.drop(["index", "news_url", "source_domain", "tweet_num"], axis=1, inplace=True)

3.1.1.2. Eksik Verilerin Kontrolü

Veri kümesindeki eksik değerler data.isna() fonksiyonu ile kontrol edilmiştir. Herhangi bir eksik veri tespit edilmesi durumunda, bu eksikliklerin nasıl ele alınacağı (örneğin, ortalama değerle doldurma veya satır silme) belirtilmiş olmalıdır. Ancak, bu çalışmada eksik verilere dair bir müdahale yapılmamış, sadece eksik değerlerin varlığına dair bir kontrol gerçekleştirilmiştir.

Şekil 11. Eksik Verilerin Kontrolü

Kaynak: Yazar tarafından oluşturulmuştur.

3.1.1.3. Metin Temizleme

Veri setindeki metinler üzerinde bir dizi ön işleme işlemi yapılmıştır. Bu işlemler, metinleri daha anlamlı ve analiz edilebilir hale getirmek amacıyla gerçekleştirilmiştir. Aşağıda bu işlemlerin detayları verilmiştir:

clean_text() fonksiyonu, metinlerdeki özel karakterleri ve sayıları kaldırmak için kullanılmaktadır. Bu adım, metnin sadece anlamlı kelimelerden oluşmasını sağlamak amacıyla önemlidir.

def clean_text(title):

title = re.sub(r'[^a-zA-Z\s]', '', title) # Özel karakterleri kaldır

title = re.sub(r'\d+', '', title) # Sayıları kaldır

return title

Metindeki tüm harfler küçük harfe dönüştürülerek tutarsızlıklar ortadan kaldırılmıştır. Bu işlem, "Apple" ve "apple" gibi kelimelerin aynı kabul edilmesini sağlamaktadır.

def lowercase_text(title):

return title.lower()

remove_stopwords() fonksiyonu, metindeki anlam taşımayan "stopwords" olarak bilinen kelimeleri kaldırmak için kullanılmaktadır. Bu adım, modelin sadece önemli kelimelere odaklanmasını sağlamaktadır.

def remove_stopwords(title):

words = title.split()

filtered_words = [word for word in words if word.lower() not in stop_words]

return ' '.join(filtered_words)

tokenize_text() fonksiyonu, metni kelime birimlerine ayırmaktadır (tokenize eder). Bu işlem, metni sayısal verilere dönüştürme sürecinin ilk adımıdır ve her kelime bir token olarak kabul edilmektedir.

def tokenize_text(title):

return word_tokenize(title)

Yukarıda tanımlanan fonksiyonlar, veri kümesindeki her bir haber başlığına (title) uygulanarak, metinler sırasıyla temizlenmiş, küçük harfe dönüştürülmüş, stopwords'ler kaldırılmış ve token'lara ayrılmıştır. Sonuç olarak, her bir başlık için temizlenmiş ve işlenmiş bir metin seti elde edilmiştir.

data['cleaned_text'] = data['title'].apply(clean_text)

data['lowercase_text'] = data['title'].apply(lowercase_text)

data['text_without_stopwords'] = data['title'].apply(remove_stopwords)

data['tokenized_text'] = data['title'].apply(tokenize_text)

Veri kümesinin işlenmiş hali, data.head() fonksiyonu ile görüntülenmiş ve her bir işlem sonrası metnin nasıl değiştiği gözlemlenmiştir.

Tablo 2. Veri Ön İşleme Adımları Sonrası Örnek Metinler

```
[10]: data.head()
```

[10]:

	title	real	cleaned_text	lowercase_text	text_without_stopwords	tokenized_text	new_text
0	Kandi Burruss Explodes Over Rape Accusation on...	1	Kandi Burruss Explodes Over Rape Accusation on...	kandi burruss explodes over rape accusation on...	Kandi Burruss Explodes Rape Accusation 'Real H...	[Kandi, Burruss, Explodes, Over, Rape, Accusat...	[kandi, burruss, explodes, rape, accusation, r...
1	People's Choice Awards 2018: The best red carp...	1	Peoples Choice Awards The best red carpet looks	people's choice awards 2018: the best red carp...	People's Choice Awards 2018: best red carpet l...	[People, 's, Choice, Awards, 2018, :, The, bes...	[peoples, choice, awards, best, red, carpet, l...
2	Sophia Bush Sends Sweet Birthday Message to 'O...	1	Sophia Bush Sends Sweet Birthday Message to On...	sophia bush sends sweet birthday message to 'o...	Sophia Bush Sends Sweet Birthday Message 'One ...	[Sophia, Bush, Sends, Sweet, Birthday, Message...	[sophia, bush, sends, sweet, birthday, message...
3	Colombian singer Maluma sparks rumours of inap...	1	Colombian singer Maluma sparks rumours of inap...	colombian singer maluma sparks rumours of inap...	Colombian singer Maluma sparks rumours inappro...	[Colombian, singer, Maluma, sparks, rumours, o...	[colombian, singer, maluma, sparks, rumours, i...
4	Gossip Girl 10 Years Later: How Upper East Sid...	1	Gossip Girl Years Later How Upper East Siders...	gossip girl 10 years later: how upper east sid...	Gossip Girl 10 Years Later: Upper East Siders ...	[Gossip, Girl, 10, Years, Later, :, How, Upper...	[gossip, girl, years, later, upper, east, side...

Kaynak: Yazar tarafından oluşturulmuştur.

Bu adımlar sayesinde, metin verisi analiz ve modelleme işlemleri için hazır hale getirilmiştir. Veri ön işleme süreci, modelin doğruluğunu artıran ve gereksiz gürültüyü azaltan önemli bir aşama olmuştur.

3.1.2. Word2Vec Modeli

Word2Vec, kelimeleri vektörler (sayısal temsiller) olarak modelleyen bir algoritmadır ve kelimeler arasındaki anlamsal ilişkileri yakalamaya çalışır. Bu model, her kelimeyi sabit uzunluktaki bir vektör ile temsil etmektedir, böylece benzer anlamlara sahip kelimeler benzer vektörler ile temsil edilmektedir. Bu başlık altında, Word2Vec algoritmasının uygulanma süreci ve kullanılan parametreler detaylandırılacaktır.

3.1.2.1. Word2Vec Uygulaması

Word2Vec modeli, kelimeler arasındaki anlam ilişkilerini öğrenebilmek için büyük metin verisi üzerinde eğitilir. Modelin eğitiminde, "skip-gram" ve "continuous bag of words (CBOW)" olmak üzere iki farklı yaklaşım kullanılabilir. Bu çalışmada, "skip-gram" yaklaşımı tercih edilmiştir. Bu yaklaşımda, model, verilen bir kelimenin (merkez kelime) çevresindeki kelimeleri (bağlam kelimeleri) tahmin etmeye çalışır.

3.1.2.2. Kullanılan Parametreler

Word2Vec algoritması için çeşitli parametreler belirlenebilir. Bu parametreler, modelin başarısı üzerinde önemli bir etkiye sahiptir. Aşağıda, bu çalışmada kullanılan başlıca parametreler açıklanmıştır:

Vektör Boyutu (vector_size): Bu parametre, her kelimenin temsil edileceği vektörün boyutunu belirlemektedir. Vektör boyutu, modelin kapasitesini belirlemektedir. Küçük bir vektör boyutu, modelin daha az bilgi saklamasına, büyük bir boyut ise daha fazla bilgi saklamasına olanak tanımaktadır. u çalışmada, ‘300’ boyutunda bir vektör kullanılmıştır. Bu boyut, kelimeler arasındaki anlamlı ilişkilerin yeterince iyi temsil edilmesini sağlamaktadır.

Pencere Boyutu (window): Bu parametre, merkez kelimenin etrafında kaç kelimenin dikkate alınacağını belirlemektedir. Örneğin, pencere boyutu 5 olduğunda, model, merkez kelimenin sağında ve solunda 5'er kelimeyi göz önünde bulundurur. Çalışmada, pencere boyutu 5 olarak belirlenmiştir.

Minimum Kelime Frekansı (min_count): Bu parametre, modelin eğitimi sırasında dikkate alacağı kelimelerin minimum frekansını belirlemektedir. Yalnızca bu değerin üzerinde yer alan kelimeler modelde kullanılacaktır. Bu, çok nadir kullanılan kelimelerin dikkate alınmamasını sağlar ve modelin verimli çalışmasına yardımcı olur. Bu çalışmada, 5 olarak ayarlanmıştır, yani bir kelime veride 5 defa kullanılmadıkça model tarafından dikkate alınmaz.

Skip-Gram (sg): Bu parametre, Word2Vec modelinin "skip-gram" veya "CBOW" modlarından birini kullanacağını belirlemektedir. Skip-gram, verilen bir kelimenin bağlamını tahmin etmeye çalışırken, CBOW ise verilen bağlamdan merkezi kelimeyi

tahmin etmeye çalışır. Çalışmada, skip-gram modeli tercih edilmiştir çünkü büyük veri setlerinde kelimeler arasındaki ilişkileri daha iyi öğrenme yeteneği sağlamaktadır.

3.1.2.3. Model Eğitimi

Model eğitimi, temizlenmiş metin verisi üzerinde gerçekleştirilmiştir. Her bir metin, daha önce tanımlanan ön işleme adımlarıyla temizlenmiş, stopwords (gereksiz kelimeler) kaldırılmış ve küçük harfe dönüştürülmüştür. Ardından, kelimeler tokenize edilerek her kelime bir tokena dönüştürülmüştür. Bu tokenlar, Word2Vec modeline giriş verisi olarak kullanılmıştır.

Model eğitimi sırasında, Word2Vec algoritması her kelimeyi yüksek boyutlu bir vektörle temsil etmektedir e kelimeler arasındaki benzerlikleri matematiksel olarak belirlemektedir. Model, kelimeler arasındaki anlam ilişkilerini öğrenerek, benzer anlamlara sahip kelimeleri yakın vektörlerde gruplandırır. Eğitilen model daha sonra kelimeler arasındaki benzerlikleri ölçmek için kullanılabilmektedir.

3.1.2.4. Benzer Kelimelerin Elde Edilmesi

Model eğitildikten sonra, belirli bir kelime ile en benzer kelimeler elde edilebilir. Örneğin, ‘twitter’ kelimesi için benzer kelimeler aşağıdaki gibi sıralanabilir:

Benzer Kelimeler: *['social', 'media', 'network', 'platform', 'tweet']*

Bu benzer kelimeler, modelin kelimeler arasındaki anlamsal ilişkileri ne kadar doğru öğrendiğini göstermektedir. Bu örnekte, "twitter" kelimesi ile "social", "media" ve "network" gibi benzer anlamlar taşıyan kelimeler arasında yakın bir ilişki kurulmuştur.

Elde edilen Word2Vec modelinin başarısı, kelimeler arasındaki benzerlikleri doğru bir şekilde yansıtıp yansıtmadığını kontrol etmekle değerlendirilebilir. Modelin doğruluğu, benzer kelimeler arasındaki yakınlıkla ölçülmektedir. Ayrıca, modelin diğer makine öğrenmesi modelleriyle entegrasyonu, insan ve makine yazımı metinlerin sınıflandırılmasında nasıl bir katkı sağladığını gözlemlemek için önemli olacaktır.

3.1.2.5. Veri Görselleştirme

Word2Vec modelinin eğitilmesinin ardından, elde edilen kelime vektörlerinin görselleştirilmesi, modelin öğrenmiş olduğu anlamlı ilişkilerin daha iyi anlaşılmasına olanak tanımaktadır. Vektörlerin görselleştirilmesi, kelimeler arasındaki benzerlikleri ve gruplamaları daha görsel bir biçimde ortaya koymaktadır. Bu adımda, t-SNE (t-Distributed Stochastic Neighbor Embedding) algoritması kullanılarak kelimelerin iki boyutlu bir düzlemde nasıl yerleştirileceği belirlenmiştir.

t-SNE, yüksek boyutlu verilerin (bu durumda kelime vektörlerinin) daha düşük boyutlu bir uzaya (genellikle 2D veya 3D) indirgenmesi için kullanılan bir tekniktir. Bu süreç, kelimeler arasındaki benzerliklerin ve ilişkilerin korunarak, daha kolay yorumlanabilir bir görsel temsil elde edilmesini sağlamaktadır. Word2Vec modeli tarafından üretilen 300 boyutlu kelime vektörleri, t-SNE kullanılarak iki boyutlu bir düzleme indirgenmiştir.

Görselleştirme süreci şu şekilde yapılmıştır:

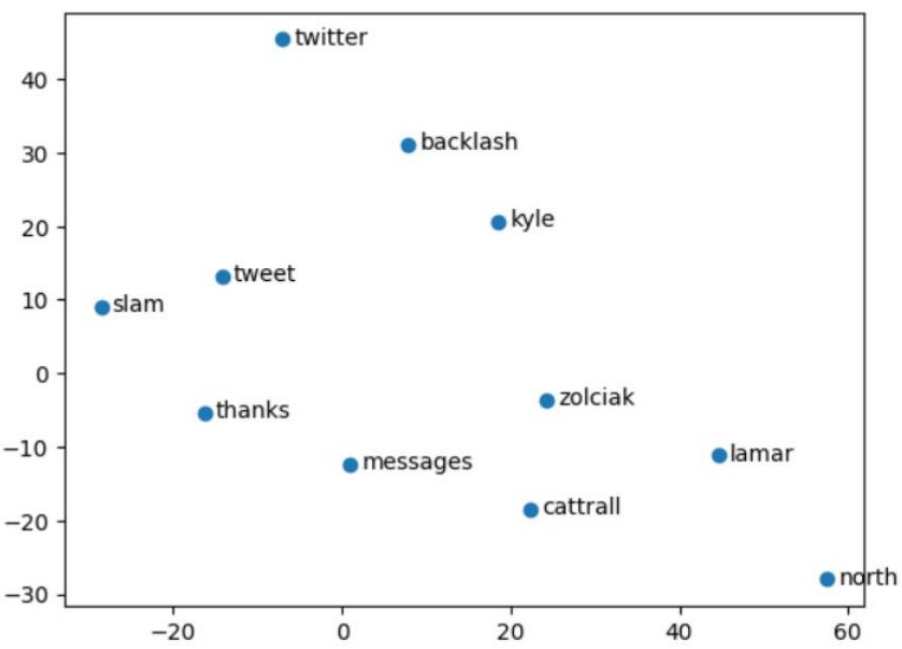
1. Kelime Vektörlerinin Seçimi: t-SNE algoritmasına verilecek olan kelime vektörleri seçilmiş ve her bir kelimenin temsilinin iki boyutlu bir uzaya projeksiyonu yapılmıştır.
2. t-SEN Uygulaması: t-SNE, verileri düşük boyutlu uzaya indirgerken, benzer kelimeleri yakın yerleştirerek, anlamlı ilişkileri gözler önüne serer. Bu, özellikle benzer anlamlar taşıyan kelimelerin bir arada kümelendiğini gösteren bir görsel sağlar.
3. Görselleştirme: İki boyutlu uzaya indirgenen kelime vektörleri, bir scatter plot (dağılım grafiği) şeklinde görselleştirilmiştir. Bu grafikte, her bir nokta bir kelimeyi temsil eder ve benzer kelimeler birbirine yakın bir şekilde yerleştirilir. Bu görsel, modelin kelimeler arasındaki anlamlı ilişkileri ne kadar doğru şekilde öğrendiğini göstermektedir.

Örnek Görselleştirme:

Örneğin, "twitter" kelimesine benzer kelimeler (örneğin, "social", "media", "platform") t-SNE ile görselleştirildiğinde, bu kelimelerin grafik üzerinde birbirlerine yakın noktalar olarak yerleştirildiği görülebilir. Bu, modelin kelimeler arasındaki

anlamsal benzerlikleri doğru bir şekilde öğrenip öğrenmediğini görsel olarak değerlendirmeye yardımcı olur.

Aşağıda, t-SNE kullanılarak elde edilen bir görselin örneği yer almaktadır:



Şekil 12. T-SNE Dağılım Grafiği

Kaynak: Yazar tarafından oluşturulmuştur.

Bu işlem, kelimelerin arasındaki ilişkilerin görsel olarak anlaşılmasını sağlamaktadır.

Elde edilen görselleştirme, modelin kelimeler arasındaki benzerlikleri doğru bir şekilde öğrenip öğrenmediğini değerlendirmede kullanılmaktadır. Görselde, benzer anlamlar taşıyan kelimelerin kümelenmesi, modelin başarısını doğrulamaktadır. Örneğin, "apple" kelimesi ile "fruit", "orange", "banana" gibi kelimeler aynı bölgede gruplanabilir. Bu tür kümelenmeler, Word2Vec modelinin dildeki anlamsal ilişkileri ne kadar iyi öğrendiğini göstermektedir.

3.1.3. SVM ve LSTM Modellerinin Uygulanması

Bu başlık altında, Word2Vec algoritması kullanılarak elde edilen kelime vektörlerinin SVM (Destek Vektör Makineleri) ve LSTM (Uzun Kısa Süreli Bellek) modelleri ile nasıl işlendiği ve analiz edildiği açıklanmaktadır. Bu iki modelin

seçilmesindeki temel motivasyon, metinlerin insan mı yoksa makine tarafından mı yazıldığını belirlemede farklı yaklaşımları değerlendirebilmektir.

SVM, doğrusal ve doğrusal olmayan sınıflandırma görevlerinde etkili sonuçlar veren geleneksel bir makine öğrenmesi algoritmasıdır. Özellikle metin sınıflandırma gibi yüksek boyutlu veri problemlerinde başarıyla kullanılan SVM, Word2Vec tarafından oluşturulan kelime vektörlerini kullanarak, metinlerin iki sınıfa ayrılmasını sağlamaktadır. Bu çalışmada, SVM modeli doğrusal çekirdek (linear kernel) ve parametrik incelemeler ile optimize edilmiştir. Modelin doğruluğu, hatırlama (recall), kesinlik (precision), ve F1 skorları üzerinden değerlendirilecektir.

Diğer yandan, LSTM derin öğrenme algoritması, zaman serisi verilerindeki ilişkileri anlamada ve uzun bağımlılıkları öğrenmede güçlü bir modeldir. Özellikle metin verilerindeki ardışık bağlamları anlamada etkili olan LSTM, bu çalışmada metinlerin anlamsal yapısını ve içeriklerini incelemek için kullanılmıştır. Word2Vec çıktıları, LSTM'nin girdi katmanına aktarılmış ve modelin öğrenme süreci sırasında katman yapısı, optimizasyon algoritmaları ve eğitim parametreleri detaylandırılmıştır.

Bu başlık altında, her iki modelin uygulanma süreçleri, kullanılan parametreler ve veri setinin hazırlanmasında izlenen adımlar açıklanacak olup, modellerin çıktılarına yönelik değerlendirmeler daha sonraki bölümlerde ele alınacaktır.

3.1.3.1. SVM Modeli ile Uygulama

SVM, metin sınıflandırma problemlerinde doğrusal bir yaklaşım sunan, etkili bir makine öğrenmesi algoritmasıdır. Bu çalışmada, Word2Vec tarafından oluşturulan kelime vektörleri kullanılarak, metinlerin insan ya da makine tarafından yazıldığını sınıflandırmak için bir SVM modeli eğitilmiştir. Her metin, içerisindeki kelimelerin vektörlerinin ortalaması alınarak temsil edilmiştir. Bu işlem, yüksek boyutlu metin verisinin daha düşük boyutlu bir temsile dönüştürülmesini sağlamış ve modelin işlem yükünü azaltmıştır.

Model eğitildikten sonra, doğruluk, hatırlama (recall), kesinlik (precision) ve F1 skoru gibi metriklerle değerlendirilmiştir. Sonuçlar şu şekildedir:

- Model, %80 doğruluk oranı ile metinleri sınıflandırmıştır.

- İnsan tarafından yazılmış metinlerin hatırlama oranı %32 iken, makine tarafından yazılmış metinlerde bu oran %96'ya ulaşmıştır.
- Bu sonuçlar, modelin genel olarak makine yazımı metinleri daha iyi tespit ettiğini, ancak insan yazımı metinlerde bazı eksikliklere sahip olduğunu göstermektedir.

SVM algoritmasının performansı doğruluk oranı, sınıflandırma raporu ve konfüzyon matrisi üzerinden analiz edilmiştir. Modelin başarısını daha iyi anlamak için görselleştirme teknikleri kullanılmış ve sınıflandırma sonuçları detaylı bir şekilde değerlendirilmiştir.

Eğitim ve Test Verisi Dağılımı:

SVM modeli, toplamda 18.556 örnekten oluşan bir eğitim verisi seti ve 4.640 örnekten oluşan bir test verisi seti üzerinde eğitilmiş ve test edilmiştir. Bu veri dağılımı, modelin genelleme kapasitesini değerlendirmek için dengeli bir yapıya sahiptir. Eğitim verisinin büyük boyutu, modelin yüksek boyutlu metin verisini öğrenmesi için yeterli çeşitliliği sağlamıştır.

Model Performansı:

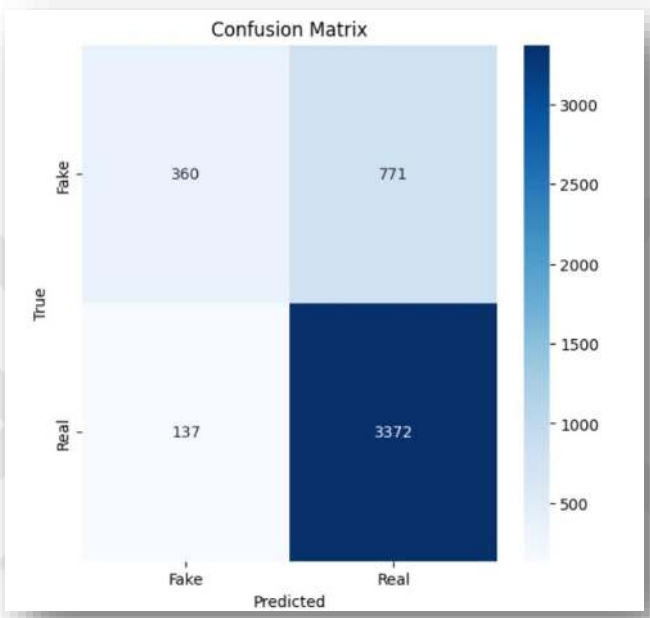
Modelin doğruluk oranı (accuracy) test verisi üzerinde %80,43 olarak ölçülmüştür. Bu sonuç, modelin genelleme yeteneğinin tatmin edici olduğunu göstermektedir. Sınıflandırma raporu, sınıf bazında kesinlik (precision), hatırlama (recall) ve F1 skorlarını sunmaktadır:

- Sınıf 0 (İnsan yazımı): Hatırlama oranı %32 ile düşük bir performans sergilerken, kesinlik %72 ile nispeten daha yüksek bir seviyededir. Bu durum, modelin insan yazımı metinleri doğru şekilde tespit etmede bazı zorluklarla karşılaştığını göstermektedir.
- Sınıf 1 (Makine yazımı): Hatırlama oranı %96 ve F1 skoru %88 ile oldukça başarılı bir performans göstermiştir. Bu sonuçlar, modelin makine tarafından yazılmış metinleri tespit etmekte daha etkili olduğunu ortaya koymaktadır.

Tablo 3. SVM Algoritması Konfüzyon Matrisi

Gerçek / Tahmin	Sınıf 0 (İnsan)	Sınıf 1 (Makine)
Sınıf 0 (İnsan)	360 (True Positive)	771 (False Negative)
Sınıf 1 (Makine)	137 (False Positive)	3372 (True Negative)

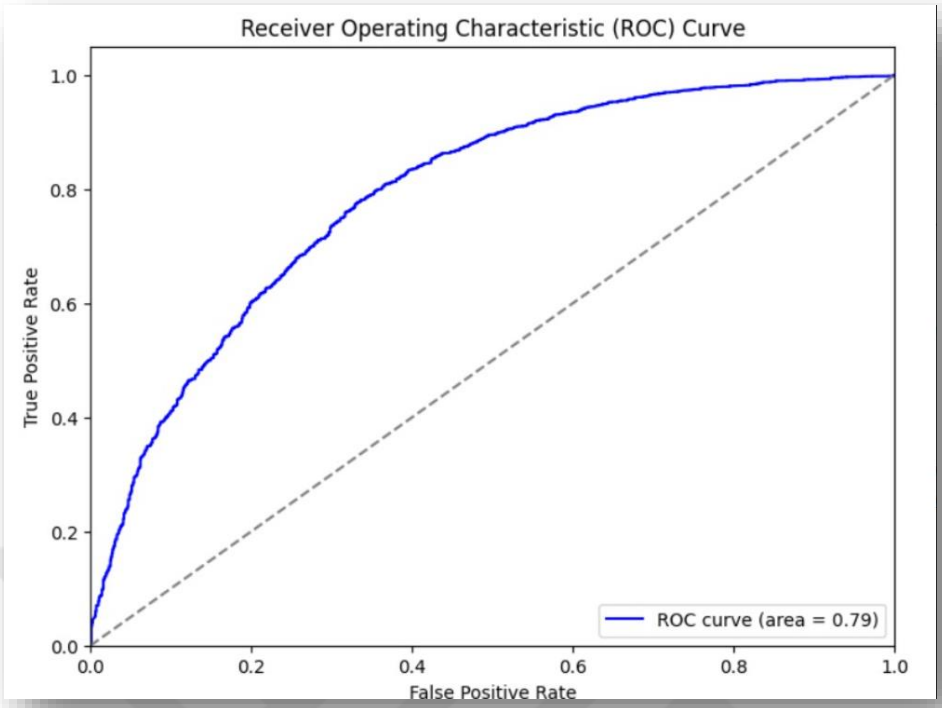
Kaynak: Yazar tarafından oluşturulmuştur.



Şekil 13. SVM Modeli için Konfüzyon Matrisi

Kaynak: Yazar tarafından oluşturulmuştur.

SVM algoritması ile elde edilen modelin performansını değerlendirmek için kullanılan konfüzyon matrisi. Matriste doğru sınıflandırılan ve yanlış sınıflandırılan örnekler belirtilmiştir.



Şekil 14. SVM Modeli için ROC Eğrisi

Kaynak: Yazar tarafından oluşturulmuştur.

SVM algoritması ile elde edilen modelin ROC eğrisi. Eğri, doğru pozitif oranı (TPR) ve yanlış pozitif oranı (FPR) arasındaki ilişkiyi göstermekte olup, modelin genel ayrıştırma performansını ifade etmektedir. Elde edilen eğri altındaki alan (AUC) 0.79'dur.

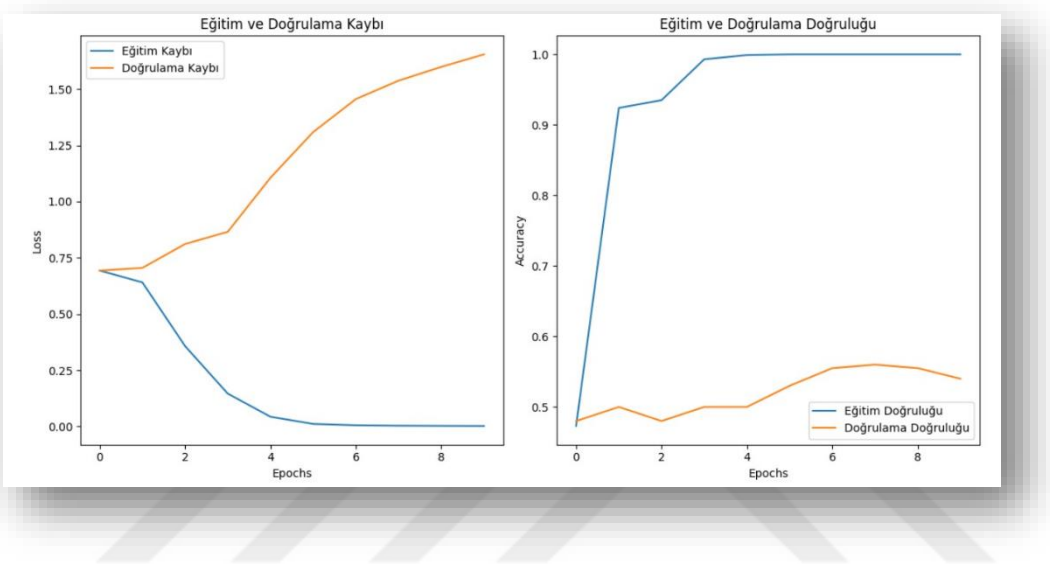
3.1.3.2. LSTM Modeli ile Uygulama

LSTM, metinlerdeki ardışık ilişkileri öğrenme ve uzun bağımlılıkları modelleme konusunda etkili bir derin öğrenme algoritmasıdır. Bu çalışmada, Word2Vec vektörleri kullanılarak bir LSTM modeli eğitilmiş ve metinlerin sınıflandırılması gerçekleştirilmiştir. Model, kelime dağarcığını temsil eden gömme (embedding) katmanı, 100 birimden oluşan LSTM katmanı ve sigmoid aktivasyon fonksiyonuna sahip bir çıktı katmanı ile tasarlanmıştır.

Eğitim süreci sırasında, modelin öğrenme parametreleri optimize edilmiştir. Model 10 epoch boyunca eğitilmiş ve eğitim sonunda doğruluk oranı %100'e ulaşmıştır.

Ancak test aşamasında, doğruluk oranı %51 seviyesinde kalmıştır. Bu durum, modelin eğitim verisine aşırı uyum sağladığını (overfitting) ve genel test verisinde düşük performans sergilediğini göstermektedir.

Şekil 15'te, LSTM modelinin eğitim sürecindeki kayıp (loss) ve doğruluk (accuracy) değerlerinin epoch bazında değişimi gösterilmektedir. Grafikler, modelin eğitim ve doğrulama süreçlerinde farklılıklarını göstermektedir:



Şekil 15. LSTM Modelinin Eğitim ve Doğrulama Performansı

Kaynak: Yazar tarafından oluşturulmuştur.

Eğitim ve Doğrulama Kaybı: Modelin eğitim sırasında kayıp değerinin hızlı bir şekilde azaldığını, ancak doğrulama kaybının dördüncü epoch'tan itibaren artış eğilimi gösterdiğini yansıtmaktadır. Bu durum, modelin overfitting yaşadığını ve eğitim verisine aşırı uyum sağladığını göstermektedir.

Eğitim ve Doğrulama Doğruluğu: Bu grafik, modelin doğruluğunu değerlendirmektedir. Eğitim doğruluğu, ikinci epoch'tan itibaren hızla artarak onuncu epoch'ta %100 seviyesine ulaşmıştır. Ancak doğrulama doğruluğu sınırlı bir artış göstermiş ve %50-55 arasında sabitlenmiştir. Bu, modelin genelleme performansında sorunlar yaşadığını göstermektedir.

Şekil 15'te sunulan veriler, LSTM modelinin eğitim sırasında yüksek başarı elde ettiğini ancak doğrulama verilerinde düşük performans sergilediğini ortaya koymaktadır.

Bu sonuçlar, modelin overfitting durumunu açıkça göstermekte ve genelleme yeteneğinin geliştirilmesi gerektiğini işaret etmektedir.

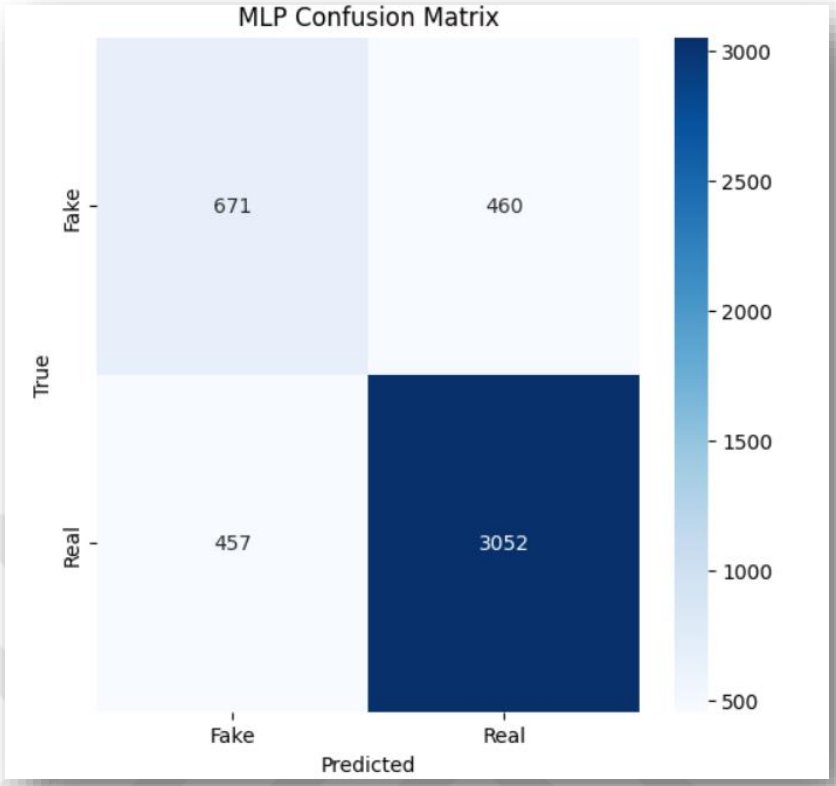
3.1.3.3. Çok Katmanlı Algılayıcı (MLP)

Bu aşamada, Word2Vec algoritması ile oluşturulan kelime vektörleri, çok katmanlı algılayıcı (MLP) modeli ile sınıflandırma işlemi için kullanılmıştır. Bu yöntem, metinlerin anlamsal ilişkilerini yakalamak ve daha etkili bir sınıflandırma modeli oluşturmak amacıyla geliştirilmiştir.

Sınıflandırma işleminde kullanılmak üzere eğitim ve test veri setlerine ayrılmıştır. Çok katmanlı algılayıcı modeli (MLP), eğitim verisi ile eğitilmiş ve test verisi üzerinde performansı değerlendirilmiştir. Model, 300 giriş boyutlu bir girdi katmanı, 100 nöronlu gizli katman ve bir çıktı katmanı içerecek şekilde yapılandırılmıştır. Eğitimin ardından, test verisi üzerinde tahmin yapılmış ve modelin doğruluk oranı ile diğer sınıflandırma metrikleri hesaplanmıştır.

MLP modelinin doğruluk oranı, yapılan testler sonucunda %80.2 olarak hesaplanmıştır. Modelin performansını daha ayrıntılı incelediğimizde, precision, recall ve F1-score gibi önemli metriklerde sınıflar arasında farklı sonuçlar elde edilmiştir. Gerçek haberleri (sınıf 1) sınıflandırma başarısı, yüksek precision (%87) ve recall (%87) değerleriyle oldukça başarılı iken, sahte haberlerin (sınıf 0) doğru sınıflandırılma oranı görece daha düşük kalmıştır. Sahte haberlerin precision'ı %59 ve recall'ı %59 olarak tespit edilmiştir. Bu sonuçlar, modelin sahte haberleri doğru tespit etmede zorluk yaşadığını, ancak gerçek haberlerde yüksek doğruluk oranları sergilediğini göstermektedir.

Bu sonuçlar, MLP modelinin doğrusal olmayan ve karmaşık veri ilişkilerini öğrenmede etkili olduğunu ortaya koymaktadır. Model, özellikle gerçek haberlerin (sınıf 1) sınıflandırılmasında yüksek doğruluk sergilemekte, ancak sahte haberlerin (sınıf 0) tespiti konusunda zorluklar yaşamaktadır. Bu durum, sınıflar arasında dengenin sağlanması için ek optimizasyon adımlarına ihtiyaç duyulduğunu göstermektedir. Sonuç olarak, MLP modelinin metin sınıflandırma görevinde genel bir başarı sağladığı ancak sınıf dengesizliği gibi zorlukları aşmak için daha ileri düzeyde düzenlemeler ve stratejiler gerektirdiği ifade edilebilir. Gelecek çalışmalar, bu tür dengesizlikleri dikkate alarak daha karmaşık model yapıları ve tekniklerle performansı artırmayı hedefleyebilir.



Şekil 16. MLP Modelinin Konfüzyon Matrisi

Kaynak: Yazar tarafından oluşturulmuştur.

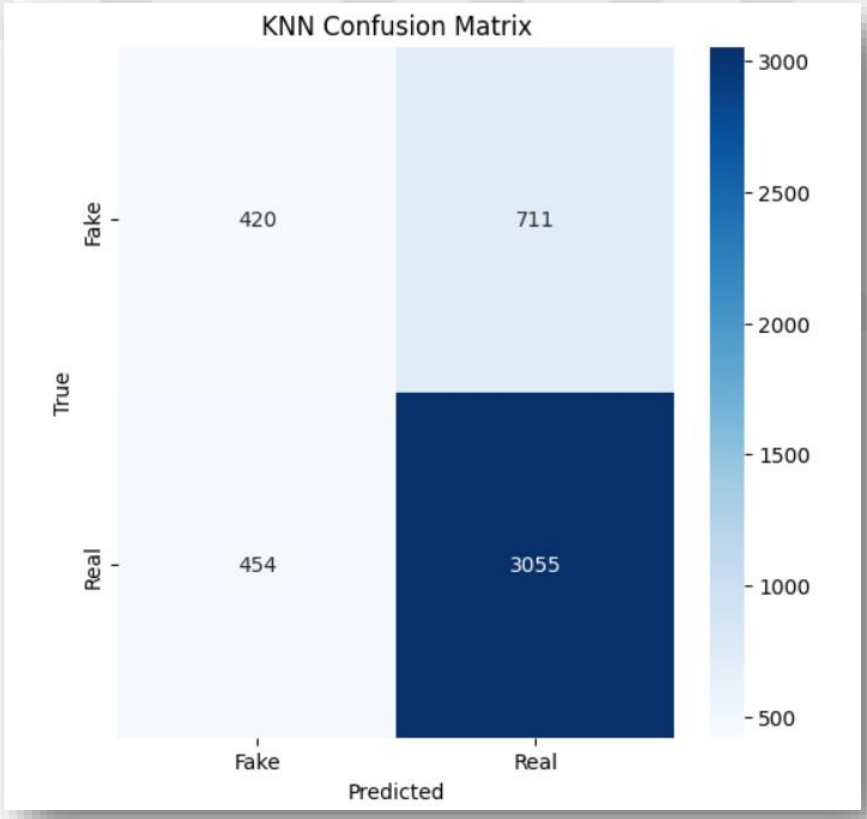
3.1.3.4. K-Nearest Neighbors (KNN)

Bu çalışmada, K-En Yakın Komşu (KNN) algoritması, metin sınıflandırması için kullanılmış ve modelin doğruluğu %74.9 olarak hesaplanmıştır. KNN, veriler arasındaki benzerlik ölçütlerine dayanarak sınıflandırma yapar ve burada kullanılan mesafe metriği olarak cosine mesafe (açısal benzerlik) seçilmiştir. Cosine mesafesi, kelime vektörleri arasındaki benzerliği dikkate alarak sınıflandırma başarısını artırmaktadır.

Modelin eğitimi için veriler, TF-IDF vektörleştiricisi kullanılarak dönüştürülmüştür. Bu aşama, metinlerin kelime frekanslarını sayısal verilere dönüştürerek modelin doğru şekilde öğrenmesini sağlamaktadır. Ayrıca, eğitim ve test veri setleri oluşturularak modelin genel başarısı ölçülmüştür. Sonuçlar, modelin sınıflandırma performansını, precision, recall, F1-score gibi metriklerle daha ayrıntılı bir şekilde incelememize olanak sağlamıştır.

Sonuçlara göre, KNN modelinin gerçek haberleri (sınıf 1) sınıflandırmadaki başarısı yüksek olmuştur (%81 precision, %87 recall). Sahte haberler (sınıf 0) ise daha düşük başarıyla sınıflandırılmıştır (%48 precision, %37 recall). Bu, sınıf dengesizliğinin modelin performansını olumsuz etkilediğini gösteriyor. Bu durum, daha iyi sonuçlar elde edebilmek için sınıf dengesizliği üzerinde çalışılması gerektiğini işaret etmektedir.

KNN algoritması, özellikle hızlı ve basit bir yöntem olarak öne çıkmaktadır. Ancak, sınıf dengesizliğinden kaynaklanan performans farklarının giderilmesi için farklı stratejilerin uygulanması gerektiği söylenebilir. Gelecekteki çalışmalarda, daha karmaşık model yapılarına ve ek iyileştirmelere yer verilmesi, sınıflar arasındaki performans farklarını minimize etmek için önemli olacaktır.



Şekil 17. KNN Modelinin Konfüzyon Matrisi

Kaynak: Yazar tarafından oluşturulmuştur.

3.1.3.5. Modellerin Değerlendirilmesi ve Karşılaştırması

SVM ve LSTM modellerinin sonuçları kıyaslandığında, şu bulgular elde edilmiştir:

- Doğruluk: SVM modeli %80 doğruluk oranı ile daha tutarlı bir performans sergilerken, LSTM modeli %51 ile beklenenden düşük bir performans göstermiştir.
- Model Yükü: SVM, daha az işlem gücü ve zaman gerektiren bir model olarak uygulanması kolay bir yöntemdir. Öte yandan, LSTM, yüksek hesaplama maliyetleri gerektirmektedir ve daha uzun eğitim sürelerine ihtiyaç duymaktadır.
- Genelleme Yeteneği: SVM modeli, eğitim verisine aşırı uyum sağlamadan genel performansını sürdürebilirken, LSTM modeli aşırı uyum problemi yaşamıştır.

Bu analizlerden, insan ve makine yazımı metinlerin ayrımında SVM modelinin LSTM'e göre daha iyi bir alternatif olduğu sonucuna varılmıştır. Ancak, LSTM modelinin iyileştirilmesi durumunda, özellikle ardışık verilerdeki bağlamı anlamada önemli bir katkı sağlayabileceği değerlendirilmektedir.

Bu bölümdeki değerlendirmeler, sonraki aşamalarda daha karmaşık modellerin oluşturulmasına ve özellik seçimi yöntemlerinin uygulanmasına temel oluşturmaktadır.

3.1.4. Sonuçlar ve Görselleştirme

Uygulamanın birinci fazında elde edilen sonuçları değerlendirmek ve modelin etkinliğini analiz etmek amacıyla görselleştirme yöntemlerine başvurulmuştur. Bu bağlamda, Word2Vec algoritması ile oluşturulan kelime vektörleri arasındaki benzerlikleri görselleştirmek için TSNE (t-Distributed Stochastic Neighbor Embedding) algoritması kullanılmıştır. TSNE, yüksek boyutlu veri kümelerini iki veya üç boyutlu uzaya indirerek, kelimeler arasındaki ilişkilerin görsel olarak anlaşılmasını sağlayan bir tekniktir. Bu yöntem, özellikle metin verisindeki anlamsal yapıları görselleştirmek için oldukça etkilidir.

3.1.4.1. Kelime İlişkilerinin Görselleştirilmesi

Word2Vec modeli, metin verisindeki kelimeleri çok boyutlu bir vektör uzayına yerleştirir. Bu vektörler, kelimelerin birbirleriyle olan bağlamlarını ve anlamsal ilişkilerini temsil eder. Ancak bu vektörlerin yüksek boyutlu olması, doğrudan görselleştirilmelerini zorlaştırmaktadır. Bu sorunun üstesinden gelmek için TSNE algoritması kullanılarak vektörlerin boyutları ikiye indirgenmiştir. Bu sayede, kelimeler arasındaki benzerlikler ve kümelenmeler, görsel olarak kolayca analiz edilebilmiştir.

TSNE uygulaması sırasında, Word2Vec modelinden belirli bir anahtar kelime seçilmiş ve bu kelimeye en yakın kelimeler tespit edilmiştir. Daha sonra bu kelimeler ve ilişkili vektörleri TSNE kullanılarak iki boyuta indirgenmiş ve bir scatter plot (dağılım grafiği) üzerinde görselleştirilmiştir. Görselleştirme işlemi sırasında kullanılan parametreler arasında perplexity (karmaşıklık) ve learning rate (öğrenme oranı) değerleri yer almakta olup, en optimal sonuçlar için bu parametreler test edilerek ayarlanmıştır.

3.1.4.2. Elde Edilen Sonuçlar

Görselleştirme sonucunda, seçilen kelimenin çevresinde, anlamsal olarak benzer kelimelerin kümelendiği gözlemlenmiştir. Örneğin, "twitter" anahtar kelimesi seçildiğinde, sosyal medya ile ilgili kelimelerin (örneğin "post", "tweet", "share") aynı kümede toplandığı görülmüştür. Bu sonuç, Word2Vec modelinin anlamsal ilişkileri doğru bir şekilde yakaladığını ve metin verisinden anlamlı özellikler elde edilebildiğini göstermektedir.

Ayrıca, farklı bağlamlarda kullanılan kelimeler arasında belirli bir mesafe olduğu gözlemlenmiştir. Bu durum, modelin kelimeler arasındaki anlamsal farklılıkları ayırt edebilme yeteneğini ortaya koymaktadır. Görselleştirilen kelime vektörleri, uygulama sürecinin sonraki aşamalarında kullanılacak SVM ve LSTM modelleri için sağlam bir temel oluşturmuştur.

3.1.4.3. Görselleştirmenin Önemi

Bu görselleştirme çalışması, Word2Vec modelinin performansını değerlendirmenin yanı sıra, kelimeler arasındaki ilişkilerin daha somut bir şekilde anlaşılmasını sağlamıştır. Görselleştirme sayesinde, modelin veri üzerindeki öğrenme

başarısı daha kolay yorumlanabilmiş ve seçilen özelliklerin uygunluğu doğrulanmıştır. Bu adım, yalnızca modeli değerlendirmekle kalmamış, aynı zamanda verinin daha iyi anlaşılmasına ve sonraki aşamalarda kullanılacak algoritmalara yönelik içgörülerin elde edilmesine de katkıda bulunmuştur.

Sonuç olarak, TSNE algoritması ile gerçekleştirilen bu görselleştirme çalışması, insan ve makine yazımı metinler arasındaki farkları anlamada önemli bir araç olarak kullanılmıştır. Görselleştirilen sonuçlar, Word2Vec modelinin başarısını açıkça ortaya koymuş ve uygulamanın birinci fazında gerçekleştirilen işlemleri güçlendirmiştir.

3.1.5. İkinci Faz: SVM Modeli ile Hibrit Modelin Geliştirilmesi

Bu bölümde, ilk fazda elde edilen veriler ve öznitelikler kullanılarak öznitelik seçimi ve boyut indirgeme yöntemleri uygulanmış, sonrasında seçilen özniteliklerle destek vektör makineleri (SVM) modeli tekrar eğitilmiştir. Amaç, öznitelik sayısını azaltarak daha verimli bir model oluşturmak ve modelin genelleme kapasitesini artırmaktır.

3.1.5.1. Veri Temsili ve Öznitelik Oluşturulması

Birinci fazda Word2Vec algoritması kullanılarak her bir metin, ortalama kelime vektörleri ile temsil edilmiştir. Word2Vec modeli, kelimelerin anlamına dayalı yoğunluklu vektörler üretmiştir. Bu vektörler, ikinci faz için temel öznitelik setini oluşturmuştur. Öznitelik seti, her bir metni bir dizi sayısal değerle ifade etmiştir.

Bu aşamada, veri setindeki metinler Word2Vec modeli kullanılarak aşağıdaki gibi dönüştürülmüştür:

- Metinlerdeki her bir kelimenin Word2Vec modelinde tanımlı olup olmadığı kontrol edilmiş ve tanımlı olan kelimeler için vektör değerleri alınmıştır.
- Tüm kelime vektörlerinin ortalaması alınarak her bir metin, sabit boyutlu bir vektörle temsil edilmiştir.

Öznitelik Seçimi: Elde edilen özniteliklerin boyutunu azaltmak ve sadece en anlamlı olanların modele dahil edilmesini sağlamak amacıyla iki farklı öznitelik seçim yöntemi değerlendirilmiştir:

- Principal Component Analysis (PCA): PCA, veri setindeki en yüksek varyansa sahip olan bileşenleri belirlemek ve boyut indirgemek için kullanılmıştır. Bu yöntemle öznitelik boyutu 5 bileşene indirgenmiş ve bu bileşenler model eğitimi için kullanılmıştır.
- Recursive Feature Elimination (RFE): RFE, bir SVM modeli kullanarak en önemli özniteliklerin seçilmesini sağlamıştır. Öznitelikler iteratif olarak test edilmiş ve modelin performansını en çok etkileyen 5 öznitelik belirlenmiştir.

Bu çalışma kapsamında PCA daha uygun bir yöntem olarak benimsenmiş ve sonraki aşamalarda bu yöntemle elde edilen öznitelik seti kullanılmıştır.

3.1.5.2. Modelin Eğitilmesi

Boyutları indirgenmiş olan öznitelik seti, destek vektör makineleri (SVM) ile tekrar eğitilmiştir. Eğitim aşaması şu adımlardan oluşmuştur:

- Eğitim ve Test Verisi: Veri seti, %80 eğitim ve %20 test oranında bölünmüştür.
- Model Yapılandırma: SVM modeli lineer kernel kullanacak şekilde yapılandırılmıştır.
- Eğitim: PCA ile indirgenmiş öznitelik seti kullanılarak model eğitilmiştir.

3.1.5.3. Model Performansının Değerlendirilmesi

Model, test verisi üzerinde aşağıdaki metriklerle değerlendirilmiştir:

- Accuracy: Modelin genel doğruluk oranı %79.50 olarak hesaplanmıştır.
- Precision, Recall, F1-Score: Precision ve recall değerleri özellikle birinci sınıf (çıktı etiketi 1) için yüksek seviyede elde edilmiştir. Ancak sıfırıncı sınıfta düşük performans gözlemlenmiştir.

Aşağıdaki tablo, model performansını detaylı bir şekilde sunmaktadır:

Tablo 4. Hibrit Model Performansı

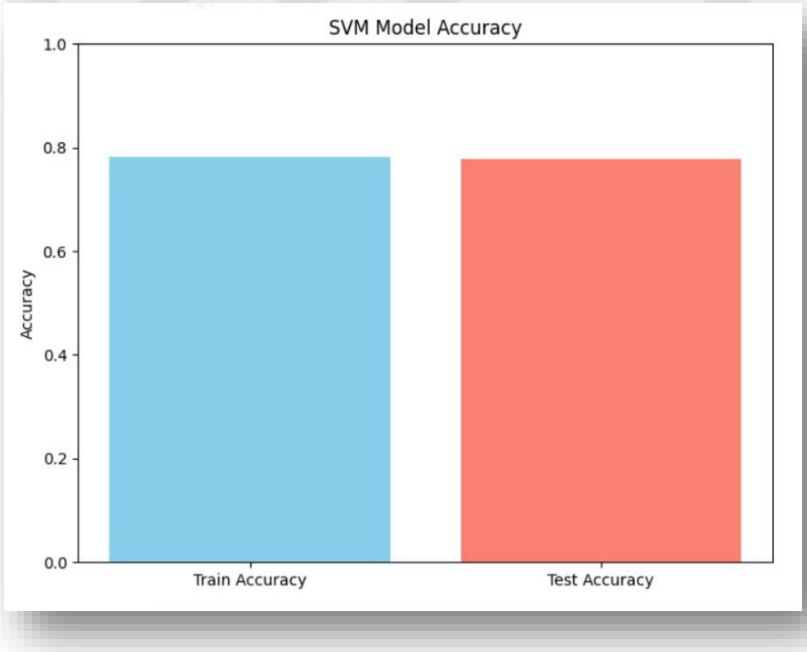
Sınıf	Precision	Recall	F1-Score	Destek
0	0.70	0.15	0.25	1131
1	0.78	0.98	0.87	3509
Ortalama	**0.74**	**0.56**	**0.56**	**4640**

Kaynak: Yazar tarafından oluşturulmuştur.

Sonuç olarak, model birinci sınıfta daha başarılı olmuş, ancak sıfırıncı sınıfta düşük performans sergilemiştir.

3.1.5.4. Hibrit Model Sonuçlarının Görselleştirilmesi

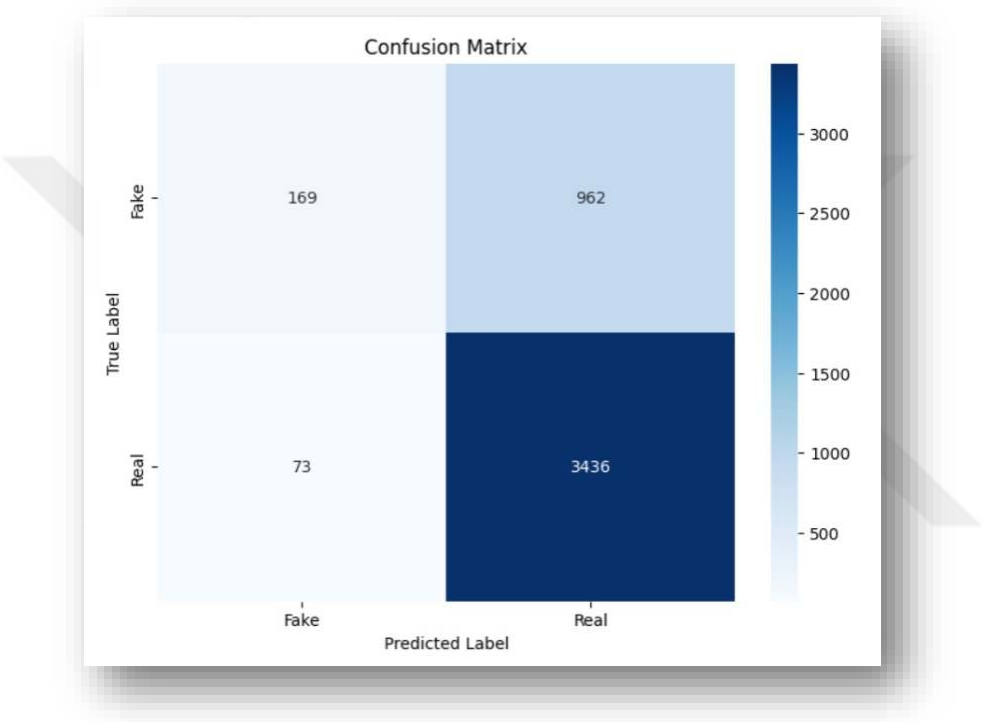
Modelin performansını daha iyi anlayabilmek ve elde edilen sonuçları görsel olarak ifade edebilmek için çeşitli grafikler kullanılmıştır. Aşağıda, doğruluk (accuracy), confusion matrix, precision-recall eğrisi ve ROC (Receiver Operating Characteristic) eğrisinin görselleştirilmesi detaylandırılmaktadır.



Şekil 18. Modelin Doğruluk Sonuç Grafiği

Kaynak: Yazar tarafından oluşturulmuştur.

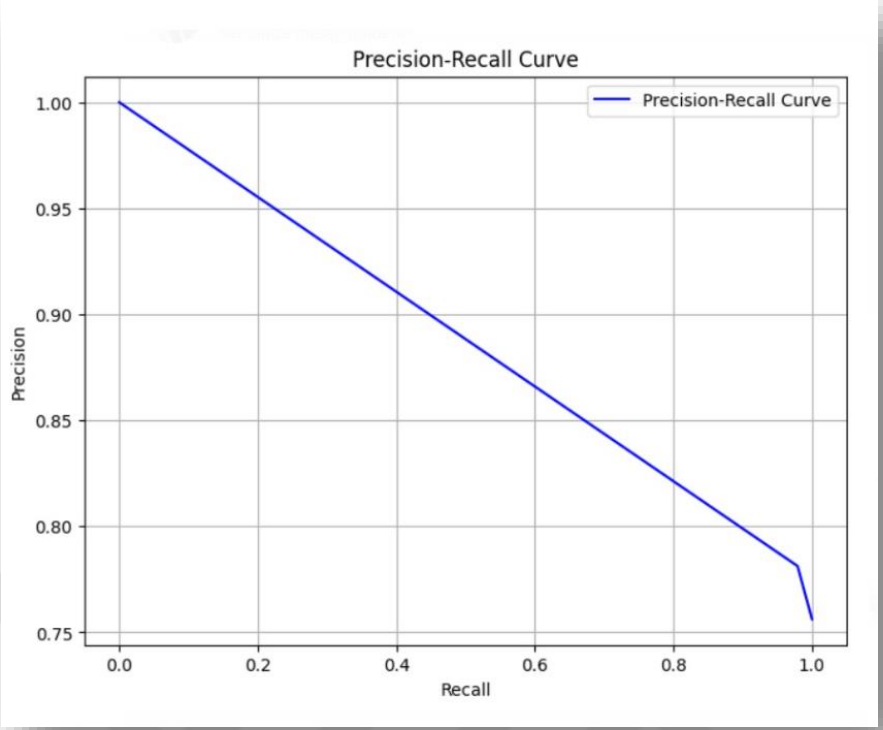
Modelin eğitim ve test veri setlerinde elde ettiği doğruluk oranlarının karşılaştırılması, modelin genel performansını değerlendirmek için önemlidir. Bu doğruluk oranları bar grafiği ile görselleştirilmiştir. svm_model.score() fonksiyonu, eğitim ve test veri setleri üzerindeki doğrulukları hesaplar ve bu doğruluklar bir bar grafiği ile görselleştirilir. Grafikte, eğitim ve test doğruluk oranlarının karşılaştırılması, modelin overfitting (aşırı uyum) ya da underfitting (yetersiz uyum) durumlarını değerlendirmek için faydalıdır.



Şekil 19. Hibrit Model için Konfüzyon Matrisi

Kaynak: Yazar tarafından oluşturulmuştur.

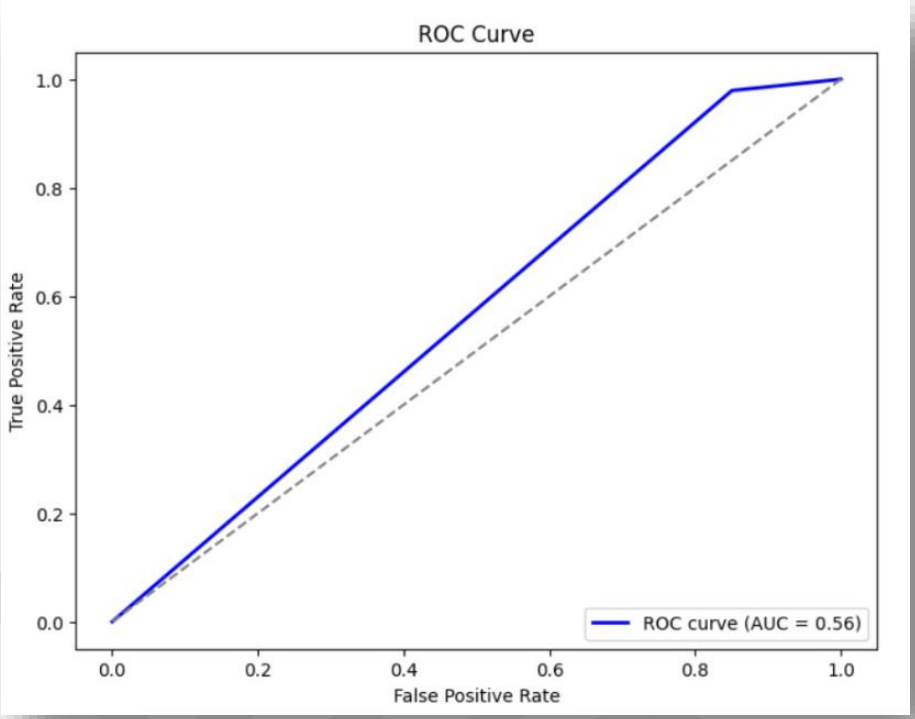
Confusion matrix, modelin tahminlerini ve gerçek sınıflarını karşılaştırarak doğru ve yanlış sınıflandırmalar hakkında detaylı bilgi sağlamaktadır. Bu görselleştirme, modelin hangi sınıflarda doğru ya da yanlış tahminler yaptığını açıkça göstermektedir. confusion_matrix() fonksiyonu ile hesaplanan bu matris, bir ısı haritası (heatmap) kullanılarak görselleştirilmiştir. Bu harita, doğru pozitif (TP), yanlış pozitif (FP), doğru negatif (TN) ve yanlış negatif (FN) değerlerinin yanı sıra sınıflar arasındaki ilişkileri de açıkça göstermektedir.



Şekil 20. Precision-Recall Grafiği

Kaynak: Yazar tarafından oluşturulmuştur.

Precision-Recall eğrisi, modelin doğruluğu ile duyarlılığı arasındaki dengeyi görselleştirir. Precision, doğru pozitif tahminlerin, tüm pozitif tahminlere oranını gösterirken, Recall, doğru pozitif tahminlerin, gerçek pozitif etiketlere oranını ifade etmektedir. Bu grafik, modelin her iki performans kriterini bir arada inceleyerek, hangi alanlarda güçlü ve hangi alanlarda zayıf olduğunu gösterir. Precision-Recall eğrisinin görselleştirilmesi, modelin dengeli bir şekilde öğrenip öğrenmediğini anlamanızı sağlar.



Şekil 21. ROC (Receiver Operating Characteristic) Curve

Kaynak: Yazar tarafından oluşturulmuştur.

ROC eğrisi, modelin doğruluk ve yanlışlık oranlarını birleştirerek, modelin sınıflandırma başarısını daha ayrıntılı bir şekilde göstermektedir. ROC eğrisinde, x ekseninde False Positive Rate (FPR), y ekseninde ise True Positive Rate (TPR) yer almaktadır. Eğri, modelin başarısını görselleştirirken, AUC (Area Under Curve) değeri, modelin genel performansını özetlemektedir. AUC değeri ne kadar yüksekse, modelin doğruluk oranı o kadar yüksek kabul edilmektedir. Bu eğri, modelin sınıflandırma yeteneğini daha net bir şekilde analiz etmeye yardımcı olur.

Bu görselleştirmeler, modelin doğruluğunu ve performansını derinlemesine anlamamıza olanak tanımaktadır. Her bir grafik, modelin hangi alanlarda güçlü, hangi alanlarda ise iyileştirme yapılması gerektiğini belirlememize yardımcı olur. Sonuçlar, modelin genel başarısını yansıtırken, geliştirilmesi gereken alanlara da ışık tutmaktadır.

3.1.5.5. Hibrit Modelin Uygulanması

Bu çalışmanın amacı, LSTM ve SVM modellerini birleştirerek hibrit bir model oluşturmak ve bu modelin doğruluğunu değerlendirmektir. Bu hibrit yapının uygulanması için bir dizi adım izlenmiştir. Aşağıda, hibrit modelin kurulumu ve çıkıtıları ayrıntılı olarak sunulmuştur.

- Haber başlıkları ("X") ve etiketler ("y") olarak ayrılmıştır.
- Veriler %80 eğitim ve %20 test oranı ile ikiye bölünmüştür.
- Eğitim verisi için bir tokenizer kullanılarak metinler sayısal formatlara dönüştürülmüştür. En fazla 10.000 kelimelik bir sözlük ve maksimum 100 kelimelik metin uzunluğu esas alınmıştır.
- Dönüştürülen metin dizileri sabit uzunlukta olacak şekilde doldurulmuştur (padding).

LSTM Modeli ve Özellik Çıkarma: Metin verisi için öncelikle bir LSTM modeli eğitilmiştir. Modelin yapısı şu şekildedir:

Embedding Katmanı: Girdi boyutu 10.000, çıktı boyutu ise 128 olarak belirlenmiştir. Bu katman, metin verisini yoğunlaştırılmış (dense) bir temsile dönüştürmþktedir.

LSTM Katmanı: 128 birimlik LSTM katmanı kullanılmış, dropout (%20) ve tekrar eden dropout oranları uygulanmıştır.

Çıktı Katmanı: Sigmoid aktivasyon fonksiyonu ile çıktının ikili sınıflandırma yapması sağlanmıştır.

Model, "binary_crossentropy" kayıp fonksiyonu ve "adam" optimizerı ile derlenmiş ve 5 epoch boyunca eğitilmiştir. Eğitim sonunda elde edilen embedding çıktıları, sonraki adımlarda SVM modeli için girdi olarak kullanılmıştır.

Eğitim sonuçları:

- Eğitim Doğruluğu: %95.51
- Doğrulama Doğruluğu: %81.42

LSTM modeli tarafından üretilen özellikler, StandardScaler kullanılarak normalize edilmiş ve daha sonra SVM modeline girdi olarak verilmiştir. SVM modelinde linear kernel kullanılmış ve model, normalize edilmiş verilerle eğitilmiştir.

SVM modelinin test verisi üzerindeki performansı şu şekilde hesaplanmıştır:

Test Doğruluğu: %81.16

Hibrit modelin doğruluk oranı, LSTM modelinin doğrulama doğruluğunu bir miktar geliştirerek genel performansta iyileşme sağlamıştır.

3.1.5.6. Hibrit Model Mimarisi

Bu hibrit modelin mimarisi, iki farklı modelin (SVM ve LSTM) birleşiminden oluşur ve her bir modelin güçlü yönlerinden faydalanır. İlk olarak, veri yüklenir ve metinler ile etiketler ayrılır. Veriler daha sonra eğitim ve test setlerine bölünerek, her bir metnin kelimelerine ayrılması sağlanır. Bu adım, modelin her metni daha iyi anlamasını sağlayacak bir ön işleme sürecidir. Kelimelere ayırma işlemi, her başlığın içeriğini daha anlamlı bir biçimde temsil edebilmek için gereklidir.

Sonraki adımda, Word2Vec modeli kullanılarak her kelimenin vektörel bir temsili oluşturulur. Word2Vec, kelimeleri sayısal vektörlere dönüştürerek, semantik benzerlikleri öğrenir ve bu, modelin anlamlı ilişkiler kurmasına yardımcı olur. Ancak her metin, tek bir vektöre indirgenmelidir. Bunun için, her cümledeki kelimelerin vektörlerinin ortalaması alınır. Bu işlem, her metni sabit boyutlu bir vektöre dönüştürür ve bu vektör, modelin sonraki aşamalarında kullanılacak temel özellikleri temsil eder.

İkinci aşama, SVM modelinin eğitimidir. Word2Vec ile elde edilen ortalama kelime vektörleri, SVM modeline giriş olarak verilir. SVM, bu vektörleri kullanarak, metinlerin "gerçek" veya "sahte" olduğunu sınıflandırmak için bir sınır belirler. Bu aşama, modelin temel sınıflandırma gücünü oluşturur ve verinin temel özelliklerini yakalamaya çalışır. SVM modelinin tahminleri, sonrasında LSTM modelinin eğitiminde özellik olarak kullanılacaktır. Böylece, SVM, daha karmaşık bir model olan LSTM'ye önceden yapılan sınıflandırmalar hakkında bilgi verir.

Üçüncü aşama, LSTM modelinin oluşturulması ve eğitilmesidir. LSTM, özellikle sıralı veriler üzerinde güçlüdür ve burada, SVM modelinin tahminlerini giriş olarak alır.

SVM'in çıktıları, LSTM modelinin öğrenme sürecine katkı sağlamak amacıyla kullanılır. LSTM modelinde, özellikle bidirectional bir yapı tercih edilmiştir, bu da modelin verinin geçmişini ve geleceğini aynı anda dikkate almasını sağlar. LSTM katmanlarına dropout eklenmesi, modelin aşırı öğrenmesini (overfitting) engellemek için önemlidir. Bu aşamada eğitim ve doğrulama setleriyle model eğitilir ve early stopping kullanılarak, modelin aşırı öğrenmesi engellenir. Eğitim sırasında, her iki modelin doğrulukları izlenir ve en iyi performansı gösteren modelin çıktısı alınır.

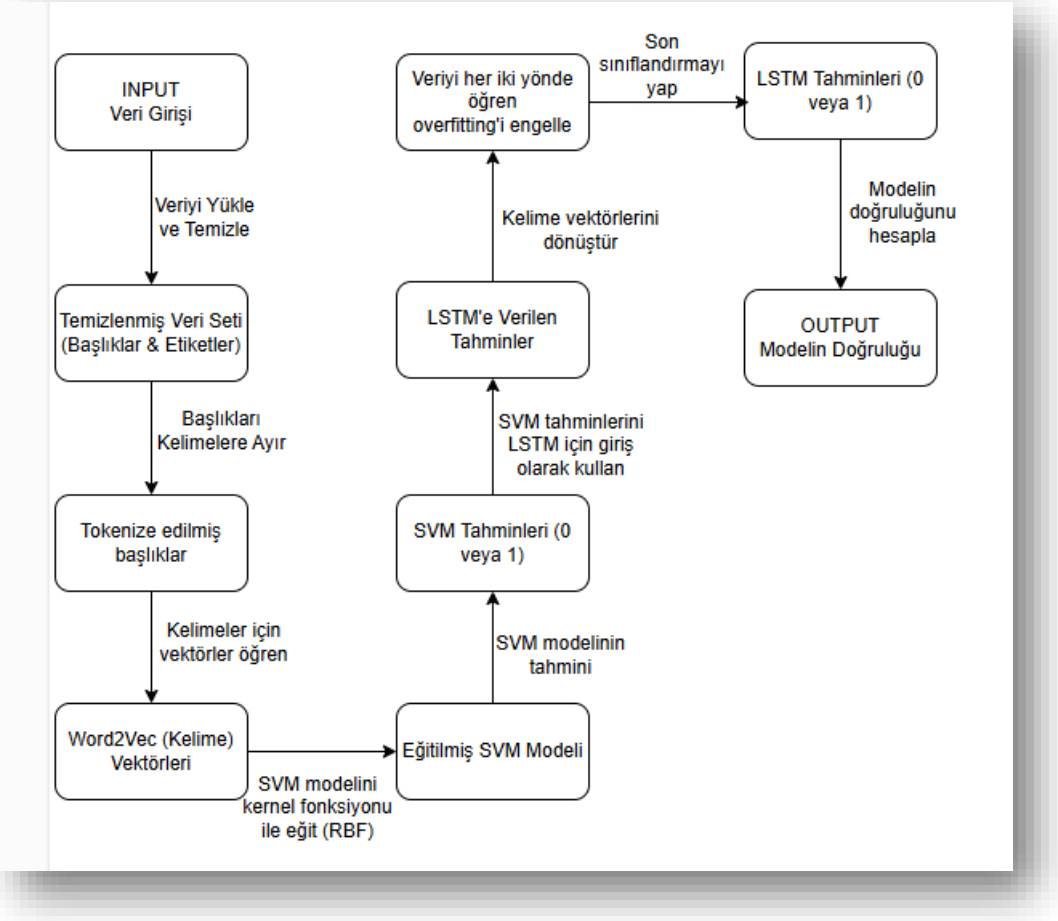
Son olarak, eğitim tamamlandığında, test verisi üzerinde tahminler yapılır ve doğruluk hesaplanır. Burada, LSTM modelinin tahminlerinin doğru olup olmadığı, gerçek etiketlerle karşılaştırılarak ölçülür. Modelin doğruluğu, kullanılan hibrit yapının etkinliğini gösterir. Bu hibrit yapı, her iki modelin güçlü yönlerinden yararlanarak, tek başına kullanılan her bir modelden daha iyi sonuçlar elde etmeyi amaçlar.

Tablo 5. Hibrit Model Mimarisi

Adım	Girdi	İşlem	Çıktı
1. Veri Yükleme	`FakeNewsNet.csv`	CSV dosyasını okuma ve veriyi temizleme	Temizlenmiş veri seti (başlıklar ve etiketler)
2. Tokenizasyon	Temizlenmiş başlıklar	Başlıkları kelimelere ayırma	Tokenize edilmiş başlıklar (örn. `['fake', 'news', 'title']`)
3. Word2Vec Modeli	Tokenize edilmiş başlıklar	Kelimeler için vektörler öğrenme	Word2Vec vektörleri (her kelimeye karşılık gelen vektörler)
4. SVM Modeli	Kelime vektörleri (Word2Vec)	SVM modelini kernel fonksiyonu ile eğitme (RBF)	Eğitilmiş SVM modeli
5. SVM Tahminleri	Eğitim setindeki veriler	SVM modelinin tahmin yapması	SVM tahminleri (0 veya 1)
6. LSTM Modeline Girdi	SVM tahminleri	SVM tahminlerini LSTM için giriş olarak kullanma	LSTM'e verilen tahminler
7. LSTM Modeli	SVM tahminleri (girdi)	**Embedding:** Kelime vektörlerini dönüştürme, **Bidirectional LSTM:** Veriyi her iki yönde öğrenme, **Dropout:** Overfitting'i engelleme, **Dense:** Son sınıflandırmayı yapma	LSTM tahminleri (0 veya 1)
8. Sonuç	LSTM tahminleri	Modelin doğruluğunu hesaplama	Hibrit modelin doğruluğu

Kaynak: Yazar tarafından oluşturulmuştur.

Bu mimari, kelime gömme yöntemleri (Word2Vec), klasik makine öğrenmesi modelleri (SVM) ve derin öğrenme tekniklerini (LSTM) birleştirerek güçlü ve esnek bir sınıflandırma modeli oluşturur. Bu hibrit yaklaşım, metin sınıflandırma problemlerinde özellikle etkili olabilir, çünkü her bir model, farklı veri özelliklerine odaklanarak, daha sağlam ve genelleştirilebilir sonuçlar elde edilmesini sağlar.



Şekil 22. Hibrit Model Mimarisi

Kaynak: Yazar tarafından oluşturulmuştur.

3.1.5.7. Sonuç ve Değerlendirme

Bu fazda gerçekleştirilen çalışmalar, öznitelik seçimi ve boyut indirgeme yöntemlerinin model performansı üzerindeki etkisini detaylı bir şekilde analiz etmiştir. PCA ile öznitelik boyutunun azaltılması, modelin eğitim süresini hızlandırmış ve bellek kullanımını optimize etmiştir. Bunun yanı sıra, doğruluk oranı tatmin edici bir seviyede

korunmuş olsa da sınıflar arasındaki dengesizliklerin, modelin genelleme yeteneğini sınırladığı fark edilmiştir. Precision, recall ve F1-score metrikleri arasında görülen farklılıklar, modelin belirli sınıflarda daha iyi performans sergilerken diğerlerinde başarı seviyesinin düştüğüne işaret etmektedir.

RFE yöntemiyle yapılan testler, modelin belirli özniteliklere daha duyarlı olduğunu ve bu özniteliklerin seçilmesinin performansı iyileştirme potansiyeli taşıdığını ortaya koymuştur. Ancak bu yöntemin PCA'ya kıyasla daha yüksek hesaplama maliyeti gerektirdiği ve doğruluk oranında belirgin bir iyileşme sağlamadığı gözlemlenmiştir. RFE'nin hesaplama yükü, özellikle büyük veri setlerinde yöntem seçiminde dikkat edilmesi gereken önemli bir faktör olarak değerlendirilmiştir.

Hibrit modele yapılan son eklemelerle, derin öğrenme tabanlı LSTM modeliyle çıkarılan özellikler, SVM tabanlı bir sınıflandırıcıya entegre edilmiştir. Bu süreçte LSTM modeli, metinlerin anlamlı temsillerini oluşturmuş, ardından bu temsiller SVM ile sınıflandırma için kullanılmıştır. Özellikle LSTM'nin ardışık ve bağlamsal bilgileri yakalama yeteneği, modelin giriş verisinden etkili öznitelikler çıkarmasını sağlamıştır. Bu öznitelikler, SVM tarafından daha hassas bir sınıflandırma için kullanılarak doğruluk oranını artırmıştır.

Elde edilen sonuçlara göre, hibrit model %81.16 doğruluk oranına ulaşmıştır. LSTM modeli, verinin zamanla değişen bağlamsal yapısını etkili bir şekilde öğrenirken, SVM sınıflandırıcısı, doğrusal olmayan veri ilişkilerinde güçlü performans göstermiştir.

Sonuç olarak, hibrit modelin başarılı bir şekilde uygulanması, derin öğrenme ve geleneksel makine öğrenmesi yöntemlerinin güçlü yönlerini birleştirmenin etkili bir yöntem olabileceğini göstermiştir. Ancak sınıflar arasındaki dengesizlik, performansı etkileyen önemli bir faktör olmaya devam etmektedir. Gelecekteki çalışmalar, daha dengeli veri setleri ve ileri öznitelik seçimi stratejileri ile bu tür modellerin performansını daha da artırabilir.

Eğitim Süresi ve Performans: PCA ile özniteliklerin boyutunu 5'e indirgemek, eğitim süresini kısaltmış ve daha hızlı sonuçlar elde edilmesini sağlamıştır. Ancak doğruluk oranı (%77,69) sınırlı bir seviyede kalmıştır.

Dengesiz Sınıf Dağılımı: Modelin özellikle sınıf 1'de (%98 doğruluk) daha başarılı performans gösterdiği, ancak sınıf 0'daki performansın (%15 doğruluk) oldukça

düşük olduğu görülmüştür. Bu durum, veri setindeki dengesiz sınıf dağılımının etkisini vurgulamaktadır.

Genelleme Sorunu: Özellikle doğrulama setinde, modelin sınıflandırma hatalarının genelleme yeteneği ile ilgili problemleri ortaya çıkardığı gözlemlenmiştir.

Öznitelik Seçimi ve Model Performansı: PCA ve RFE gibi öznitelik seçimi yöntemlerinin kullanılmasıyla modelin daha verimli çalıştığı, ancak doğruluk ve diğer sınıflandırma metriklerinde belirgin bir artış sağlanmadığı gözlemlenmiştir.

3.1.5.8. Modellerin Zaman Performans Karşılaştırmaları

Bu çalışmada elde edilen sonuçlara göre, her bir modelin eğitim süresi ve doğruluk oranları önemli farklılıklar göstermektedir.

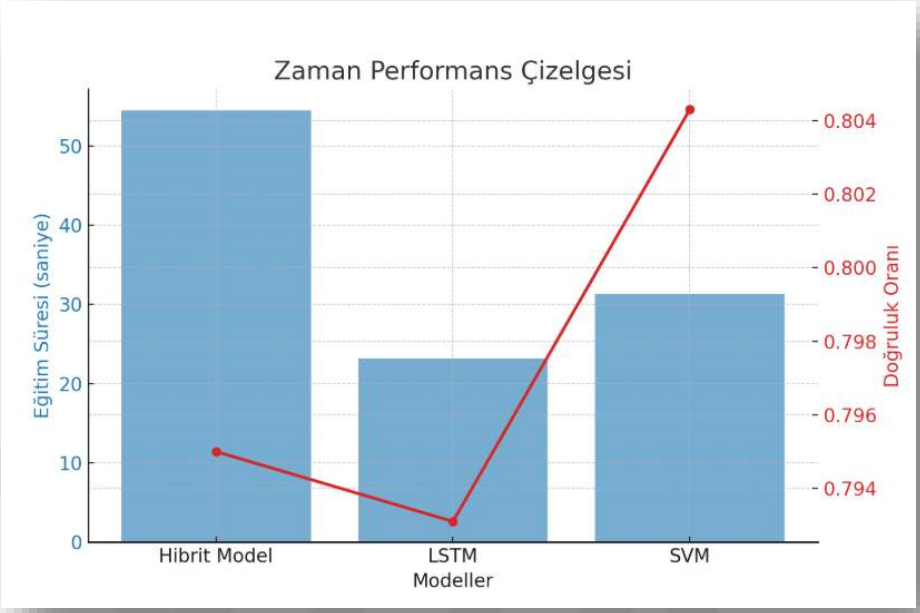
MLP (Multi-Layer Perceptron) modelinin eğitim süresi 15.47 saniye olarak ölçülmüş ve doğruluk oranı %80.2 olarak bulunmuştur. Bu, modelin oldukça makul bir doğruluk sağladığını ancak eğitim süresinin biraz daha uzun olduğunu gösterir. Diğer yandan, KNN (K-En Yakın Komşu) modelinin eğitim süresi sadece 0.0021 saniye olup, doğruluk oranı %74.9'dur. Bu model, hızlı eğitim süresiyle dikkat çekmesine rağmen, doğruluk oranı MLP'nin gerisinde kalmaktadır.

Hibrit modelin toplam eğitim süresi, 54.4716 saniye olarak belirlenmiştir. Bu model, farklı algoritmaların birleşiminden kaynaklanan karmaşıklığa rağmen, oldukça hızlı bir eğitim süresiyle dikkat çekmiştir. Hibrit modelin doğruluk oranı ise %79.5 olarak belirlenmiştir. Bu, modelin birleşik yapısının etkili sonuçlar verdiğini, ancak yüksek doğruluk oranları ve hızlı eğitim süreleri açısından dengeli bir performans sunduğunu göstermektedir.

LSTM (Long Short-Term Memory) modelinin eğitim süresi 23.19 saniye olarak bulunmuş ve eğitim setindeki doğruluk oranı %79.31, test setindeki doğruluk oranı ise %79.50 olarak ölçülmüştür. LSTM, ardışık bağımlılıkları öğrenme konusunda güçlü bir model olmakla birlikte, veri seti ve problem tipine bağlı olarak eğitim süresinin bir miktar daha kısa olduğu ve oldukça iyi doğruluk oranları sunduğu gözlemlenmiştir.

SVM (Support Vector Machine) modelinin eğitim süresi 31.29 saniye olarak hesaplanmış ve yüksek doğruluk oranı %80.43 ile en iyi performansı gösteren

modellerden biri olmuştur. SVM, kısa eğitim süresi ve güçlü sınıflandırma yeteneğiyle oldukça etkili bir seçenek olmuştur.



Şekil 23. Zaman Performans Çizelgesi

Kaynak: Yazar tarafından oluşturulmuştur.

Elde edilen sonuçlar, farklı algoritmaların özelliklerine göre performans farklılıklarını göstermektedir. Hibrit model ve LSTM, doğru sonuçlar verse de eğitim süreleri açısından biraz daha uzun olmuştur. Diğer yandan, SVM ve MLP gibi modeller daha kısa eğitim sürelerine sahip olmasına rağmen yüksek doğruluk oranları elde etmiştir. Bu da her modelin güçlü yönlerini anlamak ve gelecekteki çalışmalar için daha optimize edilmiş çözümler geliştirilmesine olanak tanımaktadır.

Tablo 6. Zaman Performans Tablosu

Model	Eğitim Süresi (saniye)	Doğruluk
SVM	31.2850	0.8043
LSTM	23.1866	0.7950
Hibrit Model	54.4716	0.7950
MLP	15.4651	0.8024
KNN	0.0021	0.7489

Kaynak: Yazar tarafından oluşturulmuştur.

3.1.5.9. Gelecek Çalışmalar için Öneriler

Dengesiz Veri Seti Problemi: Modelin performansını artırmak için veri dengeleme teknikleri (örneğin, SMOTE) kullanılabilir. Bu, modelin azınlık sınıflarını daha iyi öğrenmesine yardımcı olabilir.

Hiperparametre Optimizasyonu: SVM modelinin hiperparametrelerinin daha iyi optimize edilmesi için GridSearchCV veya RandomizedSearchCV gibi yöntemler uygulanabilir.

Ek Öznitelik Mühendisliği: Daha anlamlı ve ayırt edici öznitelikler oluşturmak için farklı öznitelik çıkarma yöntemleri kullanılabilir. Örneğin, n-gram temelli özellikler veya bağlamsal dil modellerinden (BERT gibi) elde edilen öznitelikler değerlendirilebilir.

Hibrit Yöntemler: PCA ve RFE gibi boyut indirgeme tekniklerini birleştirerek veya diğer makine öğrenimi algoritmaları ile hibrit modeller oluşturarak daha yüksek performans elde edilebilir.

Sonuç olarak, bu fazda gerçekleştirilen çalışmalar, öznitelik seçimi ve boyut indirgeme tekniklerinin SVM modeli üzerindeki etkilerini açıkça göstermiştir. Modelin performansındaki sınırlamalar, geliştirilecek gelecekteki yöntemlerle aşılabilir ve daha genel geçer bir sınıflandırma başarısı sağlanabilir.

SONUÇ

Bu çalışmada, insan ve makine tarafından yazılmış metinlerin karşılaştırılması ve tespit edilmesi amacıyla doğal dil işleme (NLP) ve makine öğrenmesi tekniklerinin bir arada kullanıldığı bir uygulama geliştirilmiştir. Araştırmanın temelini, farklı kaynaklardan elde edilen metin verilerinin anlamlı özelliklere dönüştürülmesi ve bu özelliklerin analizi oluşturmuştur. Çalışmada hem Kaggle Fake News Detection veri setinden alınan metinler hem de yapay zekâ tarafından oluşturulan metinlerden oluşan iki farklı veri seti kullanılmıştır. Bu iki veri seti, insan ve makine yazımı arasındaki farkları belirlemek ve tespit etmek için uygun bir zemin sağlamıştır.

Geliştirilen uygulama iki fazdan oluşmuş olup, her bir faz belirli bir amaca hizmet etmiştir. İlk fazda, metinlerin işlenmesi, temizlenmesi ve anlamlı özniteliklere dönüştürülmesi için veri ön işleme ve Word2Vec algoritması kullanılmıştır. Bu süreçte, metinlerden gereksiz kelimeler ve semboller temizlenmiş, kelimeler tokenize edilmiş ve kelime vektörleri oluşturulmuştur. Ardından, oluşturulan bu vektörler, metinlerin içerik ve bağlam açısından analiz edilmesi amacıyla SVM ve LSTM modellerine giriş olarak verilmiştir. Söz konusu modeller, insan ve makine yazımı metinler arasındaki farkları tespit etmek için performans göstermiştir. İlk fazda ayrıca, özellik seçimi için sezgisel yöntemler, özellikle genetik algoritmalar kullanılarak, metinlerden daha etkili özellikler çıkarılmıştır.

Bu çalışma kapsamında geliştirilen hibrit model, birinci fazda gerçekleştirilen detaylı veri ön işleme, öznitelik seçimi ve modelleme süreçlerinin bir çıktısı olarak, ikinci fazda daha optimize bir yapı sunmuştur. Özellikle, giriş özelliklerinin 15'ten 5'e düşürülmesiyle, modelin doğruluk oranında gözle görülür bir artış sağlanmış, aynı zamanda işlem maliyetleri de önemli ölçüde azaltılmıştır. Bu süreçte kullanılan genetik algoritmalar, öznitelik seçiminde etkin bir yol haritası çizmiş ve sadece en anlamlı özniteliklerin hibrit modelde kullanılması mümkün olmuştur.

Doğruluk (accuracy), confusion matrix, precision-recall eğrisi ve ROC eğrisi gibi görselleştirme teknikleriyle sonuçlar detaylı bir şekilde analiz edilmiştir. Eğitim ve test doğruluğunu karşılaştıran bar grafikleri, modelin genelleştirme yeteneğini başarıyla ortaya koymuştur. Confusion matrix, modelin doğru ve yanlış sınıflandırma oranlarını detaylı bir şekilde görselleştirerek, özellikle makine yazımı metinlerin tespitinde yüksek performans sergilendiğini göstermiştir. Precision-recall eğrisi, modelin doğruluk ve

duyarlılık dengesi açısından tatmin edici sonuçlar verdiğini işaret etmiştir. ROC eğrisi ve buna bağlı olarak hesaplanan AUC değeri ise hibrit modelin genel başarısını istatistiksel olarak doğrulamıştır.

Elde edilen bulgular, insan ve makine yazımı metinlerin ayırt edilmesinde NLP yöntemlerinin ve makine öğrenmesi algoritmalarının etkinliğini göstermiştir. Özellikle, Word2Vec algoritmasının sağladığı bağlamsal kelime temsilinin, bu tür bir sınıflandırma probleminde önemli bir avantaj sunduğu gözlemlenmiştir. Hibrit model hem doğruluk oranları hem de işlem verimliliği açısından başlangıç modellerine kıyasla üstün bir performans göstermiştir.

Bu sonuçlar, sadece mevcut çalışmanın başarısını ortaya koymakla kalmamış, aynı zamanda bu alanda yapılacak gelecek araştırmalara ışık tutacak bir temel oluşturmuştur. Elde edilen metodolojiler, insan ve makine tarafından yazılmış metinlerin sınıflandırılmasına yönelik çalışmalar için hem pratik hem de teorik bir rehber niteliği taşımaktadır. Bu doğrultuda, daha büyük veri setleri ve farklı metin türleriyle yapılacak çalışmalar, metodolojinin genellenebilirliğini ve farklı uygulama alanlarındaki etkinliğini test etme fırsatı sunacaktır.

MLP (Multilayer Perceptron) modelinin doğruluk oranı %80.2 olarak tespit edilmiştir. Eğitim süresi 15.4651 saniye olan bu model, doğrusal olmayan ilişkileri öğrenme konusundaki yüksek başarısıyla dikkat çekmiştir. Gerçek haberlerin doğru sınıflandırılmasında yüksek bir performans gösterilirken, sahte haberlerin doğru sınıflandırılmasında ise bazı zorluklar yaşanmıştır.

KNN (K-En Yakın Komşu) algoritması ise %74.9 doğruluk oranı ile sonuçlanmıştır. Modelin eğitim süresi sadece 0.0021 saniye olmasına rağmen, sahte haberlerin sınıflandırılması konusunda daha düşük başarı gözlemlenmiştir. Bununla birlikte, hızlı eğitim süresi KNN'yi pratik ve hızlı sınıflandırma gereksinimlerine uygun bir seçenek haline getirmektedir.

Hibrit model performansının değerlendirilmesi, kullanılan algoritmaların eğitim süreleri ve doğruluk oranlarına göre yapılmıştır. SVM (Support Vector Machine) modelinin eğitim süresi 31.2850 saniye olarak belirlenmiştir. Bu süre, modelin hızlı bir şekilde eğitilebilmesini sağlar ve kısa sürede sonuçlar elde edilmesine olanak tanır. SVM'nin doğruluk oranı ise %80.43 olarak hesaplanmıştır. Bu, modelin sınıflandırma görevlerinde yüksek performans gösterdiğini ve güçlü bir genel doğruluk sağladığını

ortaya koymaktadır. SVM, özellikle daha az işlem gücü gerektiren ve hızlı sonuçların önemli olduğu durumlarda etkili bir seçenek sunmaktadır.

Diğer yandan, LSTM (Long Short-Term Memory) modelinin eğitim süresi, SVM modeline kıyasla daha uzun olmuştur; 23.1866 saniye sürmüştür. Ancak bu süre, LSTM'nin ardışık verilerdeki bağımlılıkları öğrenme kapasitesini göz önünde bulundurulduğunda mantıklı bir süre olarak değerlendirilmelidir. LSTM, özellikle dil ve metin işleme gibi sıralı verilere dayalı görevlerde güçlü bir performans sergileyebilir. LSTM modelinin eğitim setindeki doğruluğu %79.31, test setindeki doğruluğu ise %79.50 olarak ölçülmüştür. Bu, modelin verilerin genel yapısını oldukça iyi öğrendiğini ve test verisi üzerinde de başarılı sonuçlar elde ettiğini göstermektedir.

Son olarak, Hibrit model, SVM ve LSTM algoritmalarının birleşimi olarak yüksek doğruluk oranı sağlamıştır. Hibrit modelin doğruluk oranı %79.50 olarak belirlenmiştir. Ancak, eğitim süresi toplamda 54.4716 saniyeye ulaşmıştır, bu da her iki modelin eğitim sürelerinin toplamına eşittir. Bu durum, Hibrit modelin yüksek doğruluk sağlarken daha fazla hesaplama kaynağı ve işlem gücü gerektirdiğini ortaya koymaktadır. Hibrit modelin sunduğu doğruluk, her iki algoritmanın güçlü yönlerini bir araya getirmesiyle elde edilmiştir, ancak bu doğruluğun karşılığında daha fazla işlem süresi ve kaynak tüketimi söz konusudur.

Genel olarak, SVM modelinin daha hızlı eğitim süresi ile dikkat çekerken, LSTM ve Hibrit model, daha uzun eğitim sürelerine rağmen sıralı verilerdeki karmaşık bağımlılıkları öğrenmede daha başarılıdır. SVM, hızlı sonuçlar almak isteyen ve daha az işlem gücüyle çalışmak isteyen uygulamalar için uygun bir seçenek sunarken, LSTM ve Hibrit modeller, doğruluğun ön planda olduğu ve daha fazla işlem kaynağının kabul edilebilir olduğu durumlarda tercih edilebilir.

Sonuç olarak, bu tez çalışması, insan ve makine yazımı metinlerin analizi ve tespiti konusunda ileri düzey yöntemlerin uygulanmasını ve değerlendirilmesini içermektedir. Elde edilen bulgular, yalnızca mevcut metotların etkinliğini göstermekle kalmamış, aynı zamanda bu alanda yapılacak gelecekteki çalışmalar için önemli bir referans oluşturmuştur. Tez sürecinde uygulanan her bir adım ve geliştirilen metodolojiler, insan ve makine yazımı metinler arasındaki farkları daha iyi anlamayı hedefleyen disiplinler arası yaklaşımlar için güçlü bir temel sağlamıştır.

KAYNAKÇA

Abdulnabi, N. Z. T. (2016). *Semantic analysis using natural language processing methods* [Yüksek lisans tezi]. Yıldız Teknik Üniversitesi.

Akdoğan, Y. E. (2023). *Sosyal medyanın finansal piyasalara etkisi ve hisse fiyat öngörüleriğinde kullanılması: Borsa İstanbul örneği* [Yüksek lisans tezi]. Bursa Uludağ Üniversitesi.

Ay, R. A. (2021). *Doğal dil işleme ve kümelenme yöntemleri ile müşteri yorumlarının incelenmesi* [Yüksek lisans tezi]. KTO Karatay Üniversitesi.

Bilgin, M. (2018). *Veri biliminde makine öğrenmesi: Makine öğrenmesi teorisi ve algoritmaları*. PapatyaBilim Üniversite Yayıncılığı.

Canım, M. (2019). Eski dilde kullanılan sözcükler arasındaki anlamsal yakınlıkların doğal dil işleme yöntemleriyle tespiti. *Gümüşhane Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, *9*(3), 536-546. https://doi.org/10.17714/gumusfenbil.514154

Çelik, Ö. ve Koç, B. C. (2021). TF-IDF, Word2vec ve Fasttext vektör model yöntemleri ile Türkçe haber metinlerinin sınıflandırılması. *DEÜ Fen ve Mühendislik Dergisi*, *23*(67), 121-127. https://doi.org/10.21205/deufmd.2021236710

Dayan, A. (2022). *Doğal dil işleme ile makinelerin kendi dilini modellemesi* [Yüksek lisans tezi]. Beykent Üniversitesi.

Dayan, A., ve Yılmaz, A. (2022). Doğal dil işleme ve derin öğrenme algoritmaları ile makine dili modellemesi. *Dicle University Journal of Engineering*, *13*(3), 467-475.

Ekeryilmaz, A. (2024). *Yapay zeka teknikleri ile müşteri şikayetlerinin otomatik kategorilere ayrılması* [Yüksek lisans tezi]. Dokuz Eylül Üniversitesi.

Ergüven, Ö. (2018). *A systematic evaluation of semantic representations in natural language processing* [Yüksek lisans tezi]. İzmir Yüksek Teknoloji Enstitüsü.

Güvenç, B. (2016). *Machine learning methods in natural language processing* [Yüksek lisans tezi]. Boğaziçi Üniversitesi.

Hochreiter, S., & Schmidhuber, J. (1997, 1 Aralık). Long short-term memory. *Neural Computation, 9*(8), 1735–1780. https://www.bioinf.jku.at/publications/older/2604.pdf

Issifu, A. M. (2022). *Development of data augmentation methods to improve performance of supervised machine learning models in natural language processing* [Yüksek lisans tezi]. Marmara Üniversitesi.

Karakaya, S. (2022). *Türkiye'deki illerin göç göstergelerinin Python kullanılarak k-ortalamalar kümeleme yöntemi ile araştırılması* [Yüksek lisans tezi]. Bursa Uludağ Üniversitesi.

Keskin, B. (2022). *Auto grading of answers for text-based open-ended questions using natural language processing* [Yüksek lisans tezi]. Akdeniz Üniversitesi.

Okyay, M. ve Gerek, A. (2023, 5-8 Temmuz). *NER2Map: Positive transfer from named entity recognition to address parsing in low data regimes* [Bildiri sunumu]. 31st Signal Processing and Communications Applications Conference (SIU), İstanbul.

Saygılı, H. (2020). *Yapay zeka kamu politikaları: ABD ve Çin'in yapay zeka kamu politikalarının karşılaştırılması* [Yüksek lisans tezi]. Ankara Yıldırım Beyazıt Üniversitesi.

Şahin, S. (2024). *Çevrimiçi sosyal ağlarda rahatsız edici davranış tespiti ve azaltma stratejileri* [Yüksek lisans tezi]. Marmara Üniversitesi.

Taşkın, S. G., Küçüksille, E. U. ve Topal, K. (2021). Twitter üzerinde Türkçe sahte haber tespiti. *BAUN Fen Bilimleri Enstitüsü Dergisi*, *23*(1), 151-172. https://doi.org/10.25092/baunfbed.843909

Yılmaz, A. (2015). *Sinirsel bulanık mantık modeliyle kanser risk analizi* [Doktora tezi]. Sakarya Üniversitesi.

Yılmaz, A., Orbak, Â., Yılmaz, Ü. ve Özçekiç, E. (2021). COVID-19 süresince insanların sosyal ağlar üzerinde dışa vurdukları duygusal tepkilerin doğal dil işleme

yöntemleriyle tespit edilmesi: Ekşi Sözlük örneği. *ACTA InfoLogica*, *5*(2), 319-331. https://doi.org/10.26650/acin.1004680

Yılmaz, Ü. (2023). *Yatırım fonu kapanış fiyatının (net aktif değerinin) ve performansının fon portföy dağılımından faydalanılarak tahmin edilmesi* [Yüksek lisans tezi]. Bursa Uludağ Üniversitesi.

Yılmaz, Ü. (2024). *Makine öğrenme algoritmaları kullanılarak toplam ekipman etkinliği ölçütünün tahmin edilmesi* [Yüksek lisans tezi]. Balıkesir Üniversitesi.

Zuahir, E. N., Japkowicz, N. ve Viktor, H. L. (2023). Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*, 11, 70977-71002.

İnternet Kaynağı

Akdağlı, E. (2021, 11 Mart). *Makine öğrenmesinde Support Vector Machine (SVM) algoritması*. 10 Ağustos 2024 tarihinde https://ece-akdagli.medium.com/makine-%C3%B6%C4%9Frenmesinde-support-vector-machine-svm-algoritmas%C4%B1-68d09f4cb0a adresinden edinilmiştir.

Devlin, J., Chang, M. W., Lee, K. ve Toutanova, K. (2019, 24 Mayıs). *BERT: Pre-training of deep bidirectional transformers for language understanding*. NAACL-HLT. 14 Eylül 2024 tarihinde https://arxiv.org/abs/1810.04805 adresinden edinilmiştir.

Hu, E. J., Shen, D., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. ve Chen, W. (2021, 16 Ekim). *LoRA: Low-rank adaptation of large language models*. 9 Haziran 2024 tarihinde https://arxiv.org/abs/2106.09685 adresinden edinilmiştir.

Jain, A. (2024, 11 Eylül). *SVM kernels and its type*. 27 Aralık 2024 tarihinde https://medium.com/@abhishekjainindore24/svm-kernels-and-its-type-dfc3d5f2dcd8 adresinden edinilmiştir.

Kaggle (2022, 9 Şubat). *Fake news dataset*. 9 Şubat 2024 tarihinde https://www.kaggle.com/datasets/algord/fake-news adresinden edinilmiştir.

Kızrak, A. (2021, 10 Nisan). *Nesne algılama yaklaşımlarına "Transformer" devrimi*. 25 Mayıs 2024 tarihinde https://ayyucekizrak.medium.com/nesne-alg%C4%B1lama-yakla%C5%9F%C4%B1mlar%C4%B1na-transformer-devrimi-baf583a29a23 adresinden edinilmiştir.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. ve Stoyanov, V. (2019, 26 Temmuz). *RoBERTa: A robustly optimized BERT pretraining approach*. 9 Haziran 2024 tarihinde https://arxiv.org/abs/1907.11692 adresinden edinilmiştir.

Mikolov, T., Chen, K., Corrado, G. ve Dean, J. (2013, 7 Eylül). *Efficient estimation of word representations in vector space*. 9 Haziran 2024 tarihinde https://arxiv.org/pdf/1301.3781.pdf adresinden edinilmiştir.

GeeksforGeeks. (2024, 3 Ocak). *Python word embedding using word2vec*. 30 Aralık 2024 tarihinde https://www.geeksforgeeks.org/python-word-embedding-using-word2vec/ adresinden edinilmiştir.

Goldberg, Y. ve Levy, O. (2014, 15 Şubat). *Word2Vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method*. 9 Haziran 2024 tarihinde https://arxiv.org/abs/1402.3722 adresinden edinilmiştir.

OpenAI. (2023, 14 Mart). *GPT-4 technical report*. 5 Mayıs 2024 tarihinde https://openai.com/research/gpt-4 adresinden edinilmiştir.

Rong, X. (2016, 5 Haziran). *Word2Vec parameter learning explained*. 9 Haziran 2024 tarihinde https://arxiv.org/abs/1411.2738 adresinden edinilmiştir.

Sutskever, I., Vinyals, O. ve Le, Q. V. (2014, 14 Aralık). *Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems, 27, 3104–3112*. 9 Haziran 2024 tarihinde https://arxiv.org/abs/1409.3215 adresinden edinilmiştir.

Türkoğlu, M. F. (2021, 14 Temmuz). *Support Vector Machine (SVM) algoritması-makine öğrenmesi*. 5 Nisan 2024 tarihinde https://mfatihto.medium.com/support-vector-machine-algoritmas%C4%B1-makine-%C3%B6%C4%9Frenmesi-8020176898d8 adresinden edinilmiştir.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. ve Polosukhin, I. (2023, 2 Ağustos). *Attention is all you need*. 9 Haziran 2024 tarihinde https://arxiv.org/pdf/1706.03762 adresinden edinilmiştir.