DOĞAL DİL İŞLEME KULLANILARAK ANA AKIM MEDYADA TÜRKÇE SAHTE HABER TESPİTİ

İsa KULAKSIZ

Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY

Yüksek Lisans Tezi

Bilgisayar Mühendisliği Eğitimi Ana Bilim Dalı

2025

T.C.
ATATÜRK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ EĞİTİMİ ANA BİLİM DALI

DOĞAL DİL İŞLEME KULLANILARAK ANA AKIM MEDYADA TÜRKÇE SAHTE HABER TESPİTİ

(Turkish Fake News Detection on Mainstream Media Using Natural Language Processing)

YÜKSEK LİSANS TEZİ

İsa KULAKSIZ

Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY

Erzurum

Ocak, 2025

FEN BİLİMLERİ ENSTİTÜSÜ
Graduate School of Natural and Applied Sciences

T.C.
ATATÜRK ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü Müdürlüğü
TEZ KABUL VE ONAY TUTANAĞI

DOĞAL DİL İŞLEME KULLANILARAK ANA AKIM MEDYADA TÜRKÇE SAHTE HABER TESPİTİ

Dr. Öğr. Üyesi Ahmet COŞKUNÇAY danışmanlığında, İsa KULAKSIZ tarafından hazırlanan bu çalışma, 15/01/2025 tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Bilim Dalı'nda yüksek lisans tezi olarak oybirliği (3/3) ile kabul edilmiştir.

Jüri Başkanı:	Doç. Dr. Mahir KAYA *Tokat Gaziosmanpaşa Üniversitesi*	Aslı Islak İmzalıdır
Danışman:	Dr. Öğr. Üyesi Ahmet COŞKUNÇAY *Atatürk Üniversitesi*	Aslı Islak İmzalıdır
Jüri Üyesi:	Dr. Öğr. Üyesi Gülşah TÜMÜKLÜ ÖZYER *Atatürk Üniversitesi*	Aslı Islak İmzalıdır

Enstitü Yönetim
Kurulunun …./…/…. tarih
ve …… sayılı kararı.

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili maddelerinde belirtilen şartları yerine getirdiğini onaylarım.

Prof. Dr. Alper NUHOĞLU
Enstitü Müdürü

Aslı Islak İmzalıdır

FEN BİLİMLERİ ENSTİTÜSÜ
Graduate School of Natural and Applied Sciences

T.C.
ATATÜRK ÜNİVERSİTESİ REKTÖRLÜĞÜ
FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜ

ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU

Yüksek Lisans Tezi olarak Dr. Öğr. Üyesi Ahmet COŞKUNÇAY danışmanlığında sunulan "DOĞAL DİL İŞLEME KULLANILARAK ANA AKIM MEDYADA TÜRKÇE SAHTE HABER TESPİTİ" başlıklı çalışmanın tarafımızdan bilimsel etik ilkelere uyularak yazıldığını, yararlanılan eserlerin kaynakçada gösterildiğini, Fen Bilimleri Enstitüsü tarafından belirlenmiş olan Turnitin Programı benzerlik oranlarının aşılmadığını ve aşağıdaki oranlarda olduğunu beyan ederiz.

Tez Bölümleri	Tezin Benzerlik Oranı (%)	Maksimum Oran (%)
Giriş	*8*	**30**
Kuramsal Temeller	*2*	**30**
Materyal ve Metot	*6*	**35**
Araştırma Bulguları ve Tartışma	*1*	**20**
Sonuçlar ve Öneriler	*0*	**20**
Tezin Geneli	*7*	**25**

Not: *Yedi kelimeye kadar benzerlikler ile Başlık, Kaynakça, İçindekiler, Teşekkür, Dizin ve Ekler kısımları tarama dışı bırakılabilir. Yukarıdaki azami benzerlik oranları yanında tek bir kaynaktan olan benzerlik oranlarının* ***%5'den*** *büyük olmaması gerekir.*

Sunulan bilgilerin doğru olduğunu, aksi halde doğacak hukuki sorumlulukları kabul ettiğimizi beyan ederiz.

Tez Yazarı (Öğrenci)	**Tez Danışmanı**
İsa KULAKSIZ	Dr. Öğr. Üyesi Ahmet COŞKUNÇAY
15.1.2025	**15.1.2025**
İmza: Aslı Islak İmzalıdır	İmza: Aslı Islak İmzalıdır

* Tez ile ilgili YÖKTEZ'de yayınlamasına ilişkin bir engelleme var ise aşağıdaki alanı doldurunuz.

☐ Tezle ilgili patent başvurusu yapılması / patent alma sürecinin devam etmesi sebebiyle Enstitü Yönetim Kurulunun …./…/…. tarih ve ……. sayılı kararı ile teze erişim 2 (iki) yıl süreyle engellenmiştir.

☐ Enstitü Yönetim Kurulunun …./…/…. tarih ve ……. sayılı kararı ile teze erişim 6 (altı) ay süreyle engellenmiştir.

Not: Bu form, Tezin son şekline uygun olarak bilgisayar ortamında doldurulmalı, çıktısı imzalanıp Tezin sonuna eklenmelidir.

TEŞEKKÜR

Bu araştırma sürecinin başlangıcından bitimine kadar destek ve yardımlarını esirgemeyen değerli danışmanım Dr. Öğr.Üyesi Ahmet COŞKUNÇAY'a bana ayırdığı değerli zamanı ve yönlendirmeleri için minnettarım. Ayrıca, bana her zaman destek olan kıymetli ailemede teşekkür ederim.

İsa KULAKSIZ

ÖZ

YÜKSEK LİSANS TEZİ

DOĞAL DİL İŞLEME KULLANILARAK ANA AKIM MEDYADA TÜRKÇE SAHTE HABER TESPİTİ

İsa KULAKSIZ

Ocak 2025, 69 Sayfa

Amaç: Bu araştırma, sahte haberlerin ana akım medyada tespiti için bir dilbilimsel model geliştirmeyi hedeflemektedir. Sınıflandırma algoritmaları üzerinde çalışılmış ve kelime temsil yöntemlerinin, sınıflandırma sonuçları üzerindeki etkisi incelenmiştir. Özellikle, bu çalışma uzman tabanlı manuel kontrol yöntemlerine olan ihtiyacı azaltarak bir haberin gerçeklik durumunu belirlemede kolaylık sağlamıştır.

Yöntem: Bu çalışmada, çeşitli haber sitelerinden toplanan haberler sahte (0) ve gerçek (1) olarak etiketlenmiştir. En başarılı modeli belirlemek için makine öğrenimi, LSTM ve BERTurk algoritmalarıyla eğitimler gerçekleştirilmiştir. Ayrıca, n-gram ve kelime çantası yöntemleri kullanılarak özellik çıkarımı yapılmıştır.

Bulgular: Bulgular, BERTurk modelinin %98,21 doğrulukla diğer sınıflandırma algoritmalarına göre daha iyi performans gösterdiğini ve makine öğrenimi algoritmalarında ise TF-IDF'nin CountVectorizer ve Word2Vec'e göre daha etkili bir kelime temsil yöntemi olduğunu ortaya koymuştur.

Sonuçlar: Bu araştırma, ana akım medyadan toplanan haberlerin gerçekliğini ayırt etmek için bir sınıflandırma yapmıştır. Sonuç olarak, kullanılan algoritmaların ve yöntemlerin bu konuda etkili olduğu görülmüştür.

Anahtar Kelimeler: Sahte haber tespiti, Makine öğrenimi, Transformer, Derin öğrenme, Sınıflandırma

ABSTRACT

MASTER'S THESIS

TURKISH FAKE NEWS DETECTION USING ON MAINSTREAM MEDIA USING NATURAL LANGUAGE PROCESSING

İsa KULAKSIZ

January 2025, 69 Pages

Purpose: The aim of this research is to develop a linguistic model for the detection of fake news in mainstream media. Classification algorithms have been investigated and the impact of word representation methods on classification results has been investigated. In particular, this study has facilitated determining the accuracy of news by reducing the need for expert-based manual control methods.

Method: In this study, news articles collected from various news sites were labeled as fake (0) and real (1). To determine the most successful model, training was conducted using machine learning, LSTM and BERTurk algorithms. In addition, feature extraction was performed using n-gram and bag-of-words methods.

Findings: The findings revealed that the BERTurk model performs better than other classification algorithms, with 98.21% accuracy, and that TF-IDF is more effective word representation method than CountVectorizer and Word2Vec in machine learning algorithms.

Conclusions: This research has classified news collected from mainstream media to distinguish of their accuracy. As a result, it has been observed that the algorithms and methods used are effective in this regard.

Keywords: Fake news detection, Machine learning, Transformer, Deep Learning, Classification

İÇİNDEKİLER

KABUL VE ONAY TUTANAĞI i
ETİK VE BİLDİRİM SAYFASI ii
TEŞEKKÜR iii
ÖZ iv
ABSTRACT v
İÇİNDEKİLER vi
TABLOLAR DİZİNİ viii
ŞEKİLLER DİZİNİ ix
KISALTMALAR VE SİMGELER DİZİNİ ix
BİRİNCİ BÖLÜM 1
Giriş 1
Araştırmanın Kapsamı ve Amacı 2
Araştırmanın Sınırlılıkları 2
Organizasyon 3
İKİNCİ BÖLÜM 4
Kuramsal Çerçeve ve İlgili Araştırmalar 4
Literatür Çalışması 4
Yapay Zeka 8
Doğal Dil İşleme 8
Makine Öğrenimi 9
Derin Öğrenme 10
Haberleşme Protokolleri 11
SOAP 11
REST 12
Sahte Haber Tanımı 12
ÜÇÜNCÜ BÖLÜM 16
Yöntem 16
Araştırma Yöntemi 16
Veri Toplama 17

Veri Ön İşleme 19
Özellik Çıkarımı 23
Sınıflandırma Algoritmaları 28
Değerlendirme Metrikleri 39
DÖRDÜNCÜ BÖLÜM 43
Bulgular 43
BEŞİNCİ BÖLÜM 50
Tartışma ve Sonuç 50
Öneriler 51
KAYNAKÇA 52
ÖZ GEÇMİŞ 56

TABLOLAR DİZİNİ

Tablo 1. Sahte haber tespit araştırmalarının özeti 6
Tablo 2. Yanıltıcı Bilgilere ait Matrix 14
Tablo 3. Word2Vec model parametreleri ve açıklamaları 26
Tablo 4. SVM için hiperparametre optimizasyonu 30
Tablo 5. k-NN için hiperparametre optimizasyonu 31
Tablo 6. Lojistik Regresyon için hiperparametre optimizasyonu 32
Tablo 7. Karar Ağacı için hiperparametre optimizasyonu 33
Tablo 8. Rastgele Orman için hiperparametre optimizasyonu 34
Tablo 9. BERT modelinin eğitiminde kullanılan parametreler ve açıklamaları 38
Tablo 10. Karmaşıklık matrisi temsili gösterimi 39
Tablo 11. BERTurk modeline ait değerlendirme metrikleri 44
Tablo 12. Modellerin doğruluk performansı 44
Tablo 13. Modelin TF-IDF yöntemi ile elde edilen doğruluk metrikleri 45
Tablo 14. Modelin CountVectorizer yöntemi ile doğruluk metrikleri 46
Tablo 15. Varsayılan parametreler ile elde edilen CountVectorizer karmaşıklık matrisi 49

ŞEKİLLER DİZİNİ

Şekil 1. TRT ve www.teyit.org platformunun paylaştığı haber örneği 1
Şekil 2. İmitasyon oyunu 8
Şekil 3. Dilbilimine ait temel analiz aşamaları 9
Şekil 4. Makine öğrenimine ait akış diyagramı 10
Şekil 5. Derin öğrenme akış diyagramı 11
Şekil 6. TRT gündem RSS servisinden dönen SOAP formatta bilgiler 12
Şekil 7. TRT gündem için REST formatında bilgiler 12
Şekil 8. Sahte haberler: Tanımlamadan teyite 13
Şekil 9. Sahte haberlerin dilbilimsel analiz süreci 13
Şekil 10. Model için akış diyagramı 16
Şekil 11. Veri setini toplanmasında kullanılan sözde kod 17
Şekil 12. Veri setinden örnek bir kesit 17
Şekil 13. Veri seti hakkında bilgiler 18
Şekil 14. Kosinüs benzerliği oranları 19
Şekil 15. Veri ön işleme aşamaları 20
Şekil 16. Veri ön işleme için kullanılan kod bloğu 20
Şekil 17. Kök indirgeme yöntemi için ilgili kod bloğu 22
Şekil 18. Ön işleme sonrası güncel veri seti 22
Şekil 19. Veri setinde en sık kullanılan kelimeler 23
Şekil 20. N-gram için python kodu 25
Şekil 21. En sık geçen kelimelerin triagram gösterimi 25
Şekil 22. Skip-gram modelinin yapısı 26
Şekil 23. Skip-gram modelinin örnek üzerinde çalışma prensibi 27
Şekil 24. BOW yöntemi için kod bloğu 27
Şekil 25. En sık geçen kelimelerin BOW gösterimi 28
Şekil 26. Naive Bayes temsili gösterimi 29
Şekil 27. SVM temsili gösterimi 29
Şekil 28. k-NN Temsili Gösterimi 31
Şekil 29. Karar ağacı temsili gösterimi 33
Şekil 30. LSTM temsili gösterimi 35
Şekil 31. LSTM modelinin mimari kodu 36

Şekil 32. Transformer model mimarisi 37
Şekil 33. BERT model mimarisi 38
Şekil 34. Doğrulama süreci 41
Şekil 35. k-katlı çapraz doğrulama temsili gösterimi 42
Şekil 36. BERTurk modeline ait karmaşıklık matrisi 44
Şekil 37. TF-IDF yöntemi ile elde edilen değerlendirme metrikleri 46
Şekil 38. CountVectorizer yöntemi ile elde edilen değerlendirme metrikleri 47
Şekil 39. LSTM için eğitim ve doğrulama grafiği 47
Şekil 40. LSTM için ROC eğrisi 48

KISALTMALAR VE SİMGELER DİZİNİ

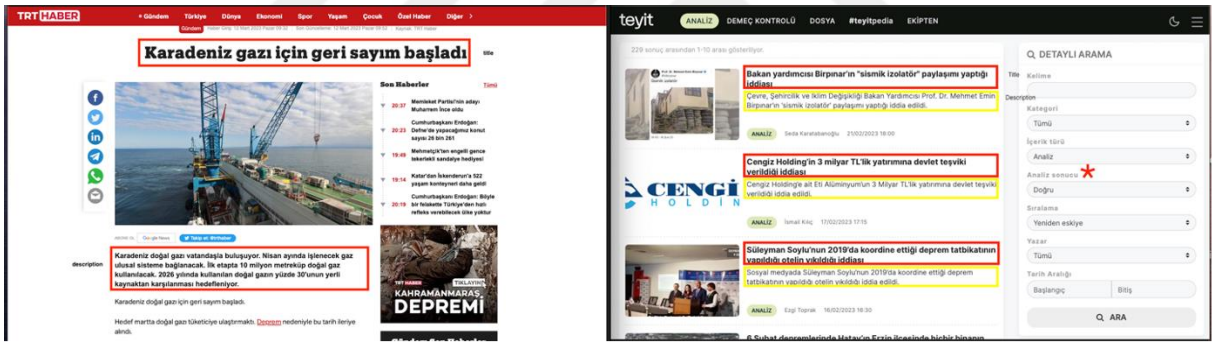
Kısaltmalar

SVM	Destek Vektör Makinesi
k-NN	k En Yakın Komşu
LSTM	Uzun Kısa Süreli Bellek
KDS	Karar Destek Sistemi
DDİ	Doğal Dil İşleme
API	Uygulama Programlama Arayüzü
RSS	Zengin Site Özeti
SOAP	Basit Nesne Erişim Protokolü
REST	Temsili Durum Aktarımı
YSA	Yapay Sinir Ağları
TRT	Türkiye Radyo Televizyon Kurumu
BOW	Bag of Words
ANN	Yapay Sinir Ağları
NER	Adlandırılmış Varlık Tanıma

BİRİNCİ BÖLÜM

Giriş

Son yıllarda internetin toplum tarafından yaygın bir şekilde kullanılmasıyla birlikte haber kaynaklarıda çeşitlilik göstermiştir. Web 1.0'ın (1995'ten itibaren) ilk aşamasında, içeriği temelde tek bir fikri aktarmayı amaçlayan monologlardan oluşurken Web 2.0 (2000'den itibaren), Facebook ve Twitter gibi sosyal medya platformlarında (Choudhury, 2014; Lazer vd., 2018) geniş ölçekli dinamik içerik oluşturma yetkisi vererek sıradan kullanıcıların haber paylaşımını kolaylaştırmıştır. Bu artış ile haberlerin doğruluğunun araştırılmadan kasti veya bilmeden kamuoyuna paylaşılması sahte haberlerin toplumda ön görülemez sonuçlara yol açabileceğini göstermektedir. 2016 yılında Birleşik Krallık'ın Avrupa Birliği'nden çekilmesini içeren haberler ve ABD başkanlık seçimlerinin sonuçları, sahte haberlerin toplumdaki yaygınlığını önemli ölçüde artırmıştır (Kucharski, 2016).



Şekil 1. TRT ve www.teyit.org platformunun paylaştığı haber örneği

Hint-Avrupa dillerinde sahte haberlerin tespiti üzerine pek çok çalışma yapılırken, Türkçenin analizi kendine özgü morfolojik yapısı nedeniyle önemli bir zorluk teşkil etmektedir. Facebook veya Twitter/X gibi sosyal medya platformları üzerinde paylaşılan haberlerin eğitim seti olarak toplanabilmesi için API hizmetlerinden faydalanılırken ana akım medya üzerinde API veya RSS hizmetlerinin sınırlılığı görülmektedir. Bu çalışma, Türkçe'nin kendine özgü morfolojik yapısının beraberinde getirdiği zorlukları ele alarak, gerçek dünyadan oluşturulan veri setini kullanarak ana akım medyadaki sahte haberlerin tespiti için literatüre önemli bir katkı sunmakta ve BERTurk transformer modelinin performansına dikkat çekmektedir. Bu araştırmada kullanılan veri seti Şekil 1'de görüldüğü gibi ana akım medya kaynaklarından biri olan TRT Gündem ve uzman odaklı manuel kontrol yöntemlerinden biri

olan Teyit.org'dan toplamıştır. Teyit.org, haberin sahte mi yoksa gerçek mi olduğunu ve saygın bir haber kaynağının bunu yayınladığını doğruluyor. Bununla birlikte, bu platform Uluslararası Gerçek Kontrol Ağı'nın (IFCN) bir parçasıdır (Ünver, t.y.).

Bu araştırmada klasik makine öğrenimi (SVM, Lojistik Regresyon, Rastgele Orman, k-NN, Karar Ağacı ve Naive Bayes), LSTM ve BERTurk algoritmalarıyla ana akım medya üzerinde sahte haberlerin tespiti için bir dilbilimsel bir model önermektedir. Bunlara ek olarak TF-IDF, CountVectorizer ve Word2Vec gibi farklı kelime temsil yöntemlerinin sınıflandırma sonuçları üzerindeki etkisi de araştırılmaktadır. Sonuçlar, Transformer tabanlı BERTurk modelinin %98,21 ile diğer makine öğrenimi algoritmaları ve LSTM'den daha iyi performans gösterdiğini ve makine öğrenimi algoritmaları arasında TF-IDF yönteminin daha etkili olduğunu ortaya koymaktadır. Ayrıca, hiperparametre optimizasyonuyla yapılan parametre seçiminin modelin doğruluk metriğinde daha iyi skor elde ettiği kanıtlanmıştır.

Araştırmanın Kapsamı ve Amacı

Sahte haberlerin ayırt edilebilmesi oldukça titiz ve zaman alıcı bir süreçtir. Türkçe dilde yapılan çalışmalarda genelde kullanılan veri seti sosyal medya üzerinden toplanırken ana akım medya üzerindeki çalışmalar çok sınırlı kalmıştır. Bu alanda yapılan araştırmalarda kullanılan modeller ile birlikte uzman odaklı doğrulama yöntemlerine ihtiyaç duymadan ilgili haberlerin gerçekliğinin ayırt edilebilmesi için bir dilbilimsel model önermektedir.

Araştırmanın Sınırlılıkları

Türkçe, köken olarak "Ural-Altay" dil ailesine mensup ve sondan eklemeli bir dildir. Bundan dolayı, Türkçe dilinde doğal dil işleme yöntemleri için kelimelerin köklerine ayrılması önemli zorlukları ortaya çıkarmaktadır (Çöltekin, 2014; YAMANAN, 2016).

Çekoslovakya-lı-laş-tır-ama-dık-lar-ımız-dan-mı-sınız?

- -Çekoslovakya: kök (ülke adı)
- -lı: İlgili olma eki (Çekoslovakyalı: Çekoslovakyadan olan)
- -laş: Dönüşme eki (Çekoslovakyalılaş: Çekoslovakyalı gibi olma durumu)
- -tır: Ettirgenlik eki (Çekoslovakyalılaştır: Çekoslovakyalılaştırma eylemi)
- -ama: Olumsuzluk eki
- -dık: Sıfat-fiil eki
- -lar: Çoğul eki
- -ımız: 1.çoğul iyelik şahıs eki
- -dan: Ayrılma hal eki

- -mı: Soru eki
- •-sınız: 2.çoğul şahıs eki

Ayrıca, ana akım medyadan haberlerin toplandığı ilgili RSS servisinde, kullanıcılara yalnızca günlük haberler SOAP formatla sunulmuştur. Bu durum, hem haber kaynağından verisetini CSV formatında toplayabilmek hem de sayfalandırma için bir betik diline ihtiyaç duyulmasına yol açmıştır.

Organizasyon

Bu araştırmanın devamındaki akış düzeni aşağıdaki gibi planlanmıştır.

- Bölüm 2'de kuramsal çerçeve başlığı altında literatür çalışmalarına değinilmekte, yapay zeka başlığı altında ise doğal dil işleme, makine öğrenimi ve derin öğrenme ve haberleşme protokollerinden bahsedilmektedir.
- Bölüm 3'de materyal ve yöntem başlığı altında, kullanılan veri setine ait bilgiler ve ön işleme aşamaları verilmiştir. Ayrıca, araştırmada kullanılan sınıflandırma algoritmalarından bahsedilmiştir.
- Bölüm 4'de araştırma bulguları başlığı altında, araştırmadan elde edilen sınıflandırma sonuçları verilmiştir.
- Bölüm 5'de tartışma ve sonuç başlığı altında, araştırmanın literatürdeki çalışmalar ile karşılaştırılması ve çıkan bulguların yorumlanması yapılmıştır.

İKİNCİ BÖLÜM

Kuramsal Çerçeve ve İlgili Araştırmalar

Bu bölüm, araştırmanın teorik olarak altyapısını sunmayı hedeflemektedir. İlk olarak, kuramsal çerçeve başlığı altında literatürde yer alan çalışmalara değinilmiştir. Ardından, yapay zeka ve alt dalları detaylandırılmış ve haberleşme protokollerine ilişkin temel bilgiler aktarılmıştır.

Literatür Çalışması

Doğal Dil İşleme (DDİ) alanında yapılan araştırmalar insanlar ve makineler arasındaki iletişimi kolaylaştırmayı amaçlayarak çeviri, sınıflandırma, özetleme ve sohbet robotlarının geliştirilmesine odaklanmaktadır. Sahte haber tespiti alanındaki çalışmalar, geleneksel ve sosyal medyadaki gerçek ve sahte bilgileri tanımlamak için makine öğrenimi ve derin öğreneme algoritmalarının kullanımını analiz eden ve karşılaştıran çalışmaları kapsamaktadır.

2016 ABD başkanlık seçim kampanyası sırasında gazetecilerin BuzzFeed'deki etiketli haberlerin sonuçlarına dayanarak geliştirdikleri model üzerinde XGB %86 en iyi sınıflandırma sonucunu vermiştir. Buna karşılık (Kaliyar vd., 2021a, 2021b) %92,30 YSA tabanlı model, %98,90 doğruluğa ulaşan CNN ve Bert modelleri içeren bir EchoFakeD modeli önermişlerdir. Uzman tabanlı manuel kontrol yöntemleri, Politifact ve FactCheck.org gibi haber sitelerinden toplanan halka açık LIAR veri setinde XGB ve makine öğrenimi algoritmalarının (Khanam vd., 2021)kullanıldığı çalışmaların yanı sıra Bi-LSTM derin öğrenme algoritması da kullanılmıştır (Aslam vd., 2021). Bi-LSTM modeli %89 ile en yüksek doğruluğu sağlarken onu %75 ile XGBoost ve %73 ile Random Forest takip etmiştir. Twitter'da paylaşılan İngilizce haberlerden sahte haberlerin tespitine yönelik bir çalışmada (Monti vd., 2019), Rastgele Orman %87 ve Geometrik Derin Öğrenme %73 doğruluk oranlarını elde ederken (Meyers vd., 2020), CNN modeli %92,7 gibi önemli ölçüde daha yüksek bir doğruluk elde etmiştir. Bir diğer derin öğrenme yöntemi LSTM ile yapılan çalışmada %82 oranında doğruluk sağlanmıştır (Ajao vd., 2018). Bu sonuçlar, derin öğrenme yöntemlerinin daha yüksek doğruluk sağladığını göstermektedir. (Ahmad vd., 2020) Kaggle platformundan alınan DS2, DS3 ve ISOT veri setlerinin birleştirilmesiyle oluşturulan DS4 üzerinde makine öğrenmesi algoritmalarının etkileri üzerine bir çalışma sunulmuştur. Sonuçlar incelendiğinde, Rastgele Orman ve Perez-LSVM ISOT veri setinde %99, Karar

Ağacı ve XGBoost DS2'de %94, Perez-LSVM DS3'te %96 ve Rastgele Orman DS4'te %91 oranda doğruluk elde etmeyi başarmıştır. (Mertoğlu & Genç, 2020) dijital kütüphanelerde yer alacak haberlerin sınıflandırılmasını otomatikleştirerek sahte haberleri tespit etmek için yenilikçi bir yöntem önermiştir. GDELT, Teyit.org ve MVN ile oluşturulan TRFN veri setinde model, sınıflandırma algoritmaları (k-En Yakın Komşu, Karar Ağaçları, Gaussian Naive Bayes, Rastgele Orman, Destek Vektör Makineleri, ExtraTrees Sınıflandırıcı, Lojistik Regresyon) üzerinde eğitilmiştir. ExtraTrees algoritmasının %96,81 doğrulukla en iyi performansı sağladığı belirtilmiştir. (Taskin vd., 2022) Teyit.org ve Twitter API'yi sayesinde toplanan veri setinde makine ve derin öğrenme algoritmalarının sahte haber tespiti üzerindeki performansını karşılaştırmıştır. Sosyal medyayı temel alan bu çalışmada SVM %90 ile en yüksek doğruluğa ulaşmıştır. (Bozuyla & ÖZÇİFT, 2022) tarafından yapılan araştırmada Twitter API'si ve çeşitli Türk uzman odaklı manuel doğrulama yöntemleriyle (ör, Teyit, Malumatfurus ve Dogrulugune) toplanan verileri analiz eden, salgın sırasındaki COVID-19 sahte haberlerine tespitine ilişkin bir model sunulmuştur. Doğruluğu artırmak için makine, derin öğrenme algoritmaları ve transformer tabanlı algoritmalar üzerinde çalışılmıştır. Bu araştırmada, BerTURK modeli sahte COVID-19 haberlerini %98,5 doğrulukla tespit etmiştir. (Güler & GÜNDÜZ, 2023) BuzzFeed, ISOT ve SOSYalan sosyal medyadan toplanan veri setleri üzerinde CNN ve RNN-LSTM derin öğrenme algoritmalarının Word2vec kelime temsili yöntemleriyle karşılaştırması incelenmiştir. Sonuçlar RNN-LSTM algoritmasının %93,41 ve CNN %85,66 oranda doğruluğa ulaştığını gösterirken ayrıca bu çalışmada, Türkçe tabanlı modelin %87,14 ile %92,48 arasında bir doğruluk değeri elde ettiği, İngilizce tabanlı modelin ise %85,15 ile %99,9 arasında bir doğruluk elde ettiğini göstermiştir. (Koru & Uluyol, 2024) Twitter ve Kaggle gibi çeşitli kaynaklardan (BuzzFeedNews, LIAR, GossipCop, ISOT, Twitter15 ve Twitter16) toplanan veri seti üzerinde yapılan bir çalışmada, ön işleme aşamaları sonrasında BERT ve BERTurk+CNN transformer modelleri kullanılarak %90 ile %94 arasında sınıflandırma oranı elde edilmiştir. Bunların yanısıra, yapılan bir tez çalışmasında ise farklı haber kaynaklarından toplanan ve Teyit.org tarafından doğrulanan veri seti üzerinde, XGB algoritması %97 doğruluk oranı elde ederek boosting algoritmalarının daha iyi performans vermiştir (YILDIRIM, 2022). Tablo 1, yukarıdaki araştırmaların sahte haber tespiti üzerine kullanılan yöntemleri ve elde ettikleri sonuçları özetlemektedir.

Tablo 1. Sahte haber tespit araştırmalarının özeti

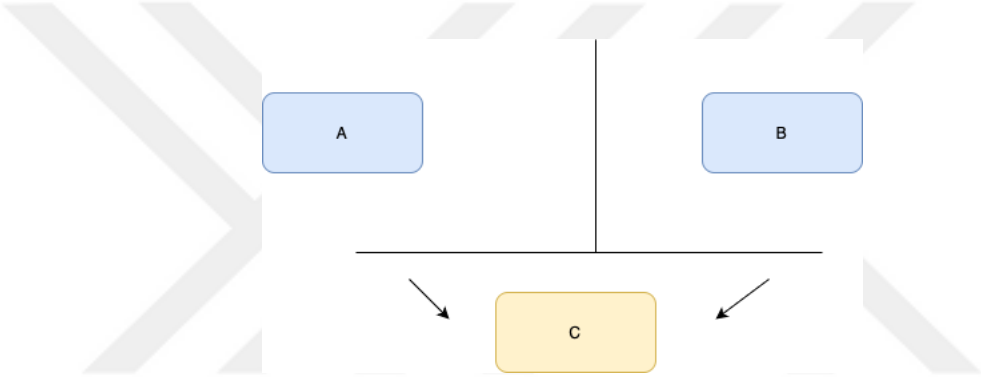
Referans	Veri Seti	Veri Sayısı	Dil	Yöntem (Algoritma)	Özet
Kaliyar vd., 2021b, 2021a	BuzzFeed	20800	İngilizce	YSA, CNN, BERT	EchoFakeD modeli %98,90 doğruluk sağlamıştır.
Khanam vd., 2021 **Aslam vd., 2021**	LIAR	12800	İngilizce	XGB, ML, Bi-LSTM	Bi-LSTM modeli %89 ile en yüksek doğruluk sağlamıştır.
Monti vd., 2019	Twitter / X	1084	İngilizce	Rastgele Orman, Geometrik Derin Öğrenme	Rastgele Orman %87, Geometrik Derin Öğrenme %73 doğruluk oranı elde etmiştir.
Meyers vd., 2020	Twitter / X	belirtilmemiş	İngilizce	CNN	Twitter'da paylaşılan haberlerde %92.7 doğruluk elde edilmiştir.
Ajao vd., 2018	Twitter / X	5800	İngilizce	LSTM	LSTM ile %82 doğruluk sağlanmıştır.

Tablo 1. (Devamı)

Ahmad vd., 2020	DS2, DS3, DS4 ve ISOT	DS2 182929 DS3 4009, ISOT 17903	İngilizce	Rastgele Orman, Perez-LSVM, Karar Ağacı, XGB	Rastgele Orman ve Perez-LSVM %99 ile en iyi performansı vermiştir.
Mertoğlu & Genç, 2020	GDELT, Teyit.org ve MVN	Toplam 85183	Türkçe	ML	ExtraTrees %96.81 ile en iyi performansı vermiştir.
Taskin vd., 2022	Teyit.org ve Twitter / X	18021	Türkçe	ML, DL	SVM %90 en iyi performansı vermiştir.
Koru & Uluyol, 2024	BuzzFeedNews, LIAR, GossipCop, ISOT, Twitter15 ve Twitter16	Belirtilmemiş	İngilizce	BERT, BERTurk+CNN	BERT tabanlı modeller de %90 ile %94 arasında doğruluk oranı elde edilmiştir.
YILDIRIM, 2022	Teyit.org ve çeşitli haber siteleri.	3007	Türkçe	Boosting	XGB %97 ile en iyi performansı vermiştir.

Yapay Zeka

İkinci Dünya Savaşı sırasında, İngiliz bilim insanı Alan Turing tarafından yayımlanan bir makalede "makineler düşünebilir mi?" sorusuna yönelik yöntemler ve deneyler açıklanmıştır (Turing, 1950). Bu makalede Turing Testi veya İmitasyon Oyunu olarak bilinen kavram, insan zekasının makineler aracılığıyla nasıl taklit edileceği konusunu ele almıştır. Bu deneyde üç katılımcı yer alır: bir erkek (A), bir kadın (B) ve bir hakem (C). A ve B hakemin göremediği bir odada bulunmaktadır. Hakem için oyunun amacı A ve B'yi birbirinden ayırt edip edemediğini belirlemektir. Alan Turing, bilgisayar programının insanı taklit edebildiği zaman o programa ait bir zekanın olabileceğini öne sürmüştür. Şekil 2'de İmitasyon oyununa ait görsel yer almaktadır.



Şekil 2. İmitasyon oyunu

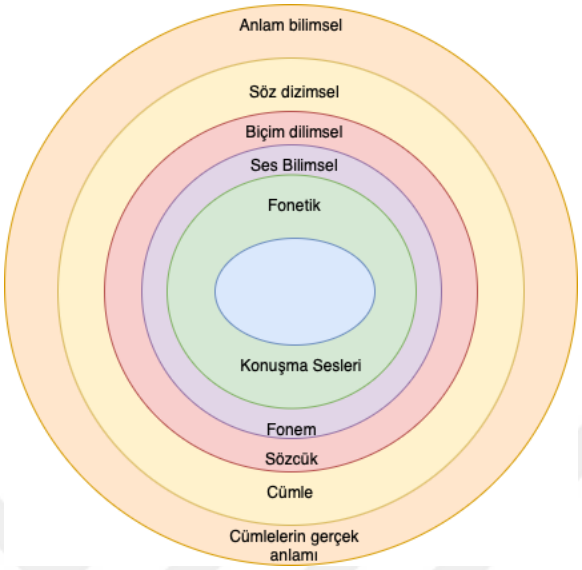
Bunun yanısıra, Cahit Arf'ın Atatürk Üniversitesi'nde yaptığı benzer bir çalışmada, makinelerin anlaşılabilmesi için sağduyunun, şeytani bir zekadan daha önemli olduğu vurgulanmıştır (Arf, 1959).

Günümüzde Yapay Zeka'nın birçok alt dalı bulunmakta ve bunlar insan yaşamını kolaylaştırmaktadır. Bu alt dallardan bazıları Görüntü İşleme, Doğal Dil İşleme ve Karar Destek Sistemleri'dir.

Doğal Dil İşleme

Doğal Dil İşleme, metin tabanlı yapılan çalışmalarda dilbilimsel özelliklerinin anlaşılabilmesine ve yeniden üretilebilmesine olanak sağlar. Bunların insan yaşamına getireceği kolaylıklar arasında; metin sınıflandırma, bilgi çıkarımı ve duygu analizi gibi birçok önemli alt başlık yer almaktadır (Küçük & Arıcı, 2018).

Dil, zaman içinde sürekli gelişen dinamik ve karmaşık bir yapıya sahiptir. Bundan dolayı, dile eklenen yeni kelimelerin yapısal özelliklerinin değişmesi, dilin sürekli evirilen bir yapıya sahip olduğunu gösterir(YILDIRIM, 2022). Dilbilim, (DDİ) çalışmalarında sıklıkla kullanılan çeşitli analiz seviyelerine sahiptir. Bu analiz seviyeleri Şekil 3'de gösterilmiştir.

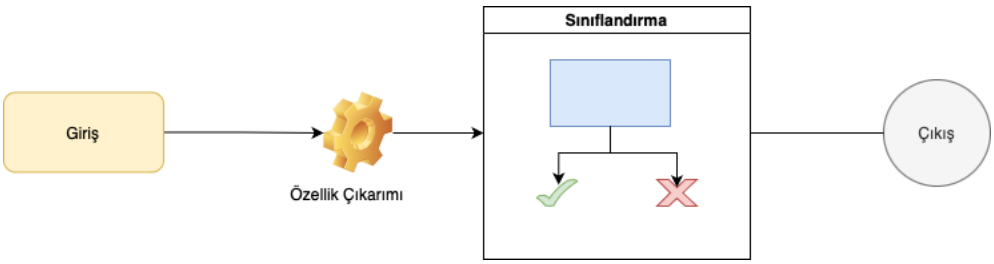


Şekil 3. Dilbilimine ait temel analiz aşamaları (YILDIRIM, 2022)

DDİ, duygu analizi, dil analizi, kelime türü bulma (POS, konuşmanın parçası) gibi ara analiz seviyelerini de içermektedir.

Makine Öğrenimi

Makine öğrenimi, bilgisayarların veri setlerini kullanarak belirli bir kritere göre başarımlarını zamanla artıracak biçimde programlanmasıdır. Bu programlama, istatistik bilimini kullanarak model oluşturma üzerinde kuruludur. Makine öğreniminin temeli, gözlemlenmiş bir örneklemden anlamlı çıkarımlar elde etmektir. Zamanla beklenen durum değişebilir veya farklı durumlar için özelleştirilme ihtiyacı söz konusu olabilir. Bu durumda, her olası koşul için ayrı bir model geliştirmekten ziyade genel olarak uyum sağlayabilen sistemler üretilmektedir (Alpaydın, 2011). Günümüzde birçok uygulama alanı vardır: Sağlık, perakende, finans teknojileri ve sahtekarlık tespitinde kullanılır. Şekil 4'de Makine öğreniminde kullanılan akış gösterilmiştir.



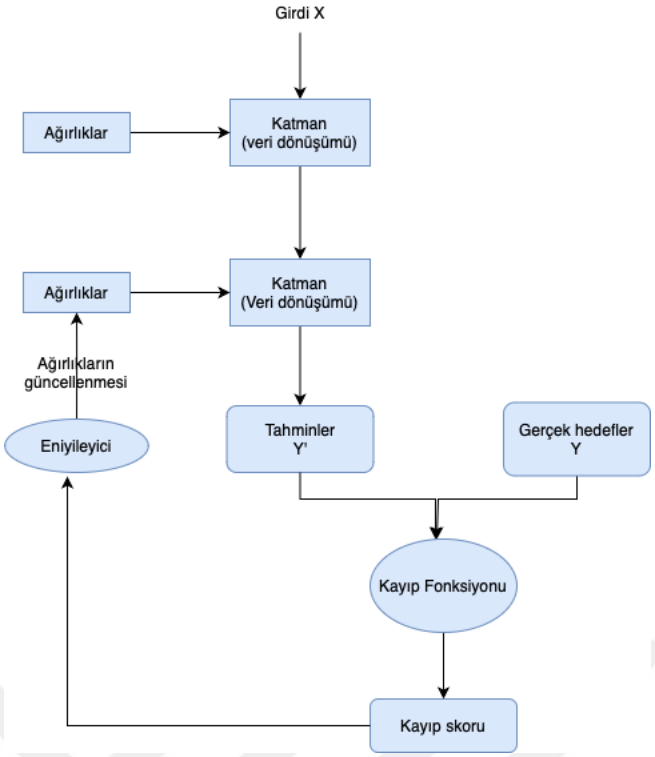
Şekil 4. Makine öğrenimine ait akış diyagramı

Makine öğrenimi kategorileri aşağıda belirtilmiştir (Chollet, 2019):

- Denetimli Öğrenme: Bir veri seti üzerinde girdi ile uzman tarafından verilen doğru çıktı arasında bilinen ilişkiyi eşleştirmeyi kapsar.
- Denetimsiz Öğrenme: Bir uzman yoktur ve elimizde yalnızca veriler vardır; amaç verileri sıkılaştırmak, verilerdeki gürültüyü azaltmaktır.
- Yarı Denetimli Öğrenme: Modelin eğitimi için etiketli olmayan verilerinin kullanıldığı bir tekniktir.
- Pekiştirmeli Öğrenme: Bir modelin çevresiyle etkileşime girerek deneyim yoluyla öğrenme sürecidir.

Derin Öğrenme

Derin öğrenme, birbirini takip eden ara katmanlardan oluşan hesaplama modellerinin, soyutlama düzeyine sahip veri gösterimlerini öğrenmesini sağlayan çok aşamalı bir yöntemdir (LeCun vd., 2015). Bu katmanlı gösterim sayesinde "sinir ağı" olarak adlandırılan modeller birbirini takip eden katmanlar aracılığıyla öğrenir. Her katman, veriyi bir önceki katmandan alır ve bu veriyi dönüştürerek işler. Bu işlem, en son katmana kadar devam eder. En son katman ise, öğrenilen bilgiler sayesinde bir tahminde bulunur. Ayrıca derin öğrenmeyi, makine öğreniminden ayıran en önemli özellik, öznitelik çıkarımının otomatikleştirilmiş olması sayesinde sorunların çözümüne kolaylık sağlamasıdır. Şekil 5'de Derin öğrenmede kullanılan akış diyagramına yer verilmiştir.



Şekil 5. Derin öğrenme akış diyagramı

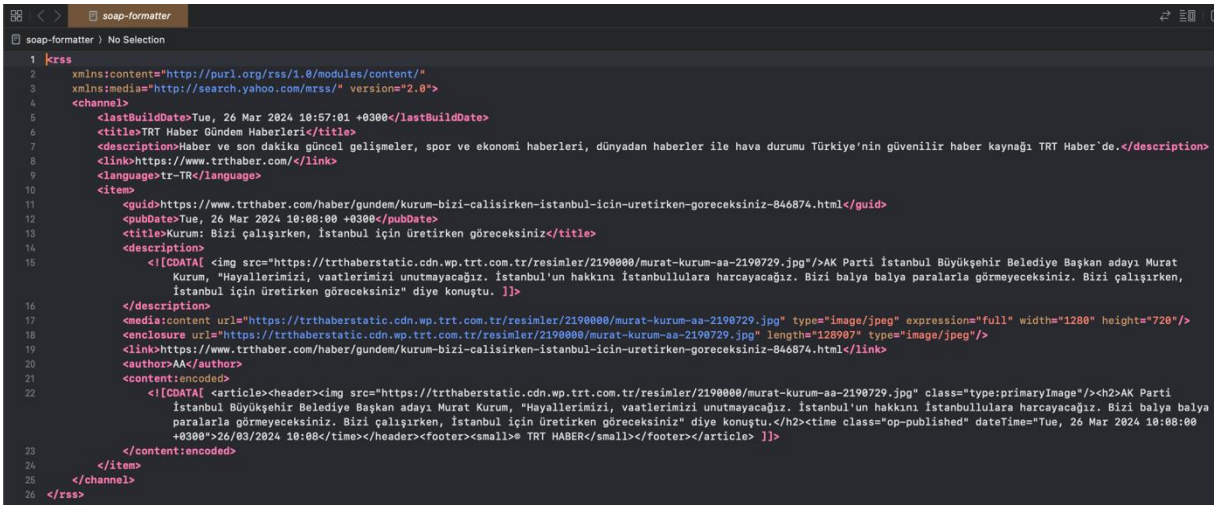
Kullanım alanları:

- Görüntü İşleme: Nesne tanıma, yüz tanıma ve görselleri sınıflandırma
- Doğal Dil İşleme: Çeviri, sınıflandırma, özetleme ve duygu analizi
- Konuşma ve Ses Tanıma: Sesli asistanlar
- Genetik: DNA veya RNA dizilimi analizi
- Finans Teknolojileri: Sahtekarlık tespiti, kredi-risk değerlendirmesi

Haberleşme Protokolleri

SOAP

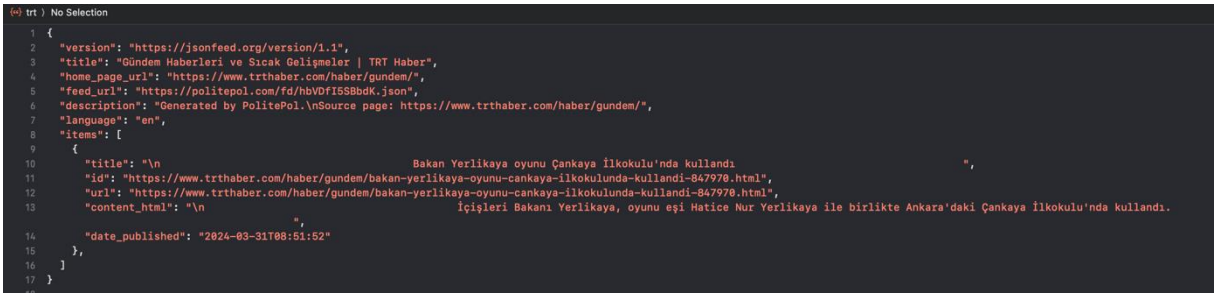
SOAP (Simple Object Access Protocol), web servislerinin haberleşmesini sağlayan ve istemci/sunucu mantığına dayalı bir protokoldür (Mumbaikar & Padiya, 2013). XML tabanlı bir mesajlaşma sistemi kullanır. Özellikle bankacılık alanında, IBM ana sistem makineleri üzerinde web servisleri arasındaki iletişimde sıklıkla kullanılır. Şekil 6'da TRT Gündem'den dönen haber başlık ve açıklama/özet SOAP formatta verilmiştir.



Şekil 6. TRT gündem RSS servisinden dönen SOAP formatta bilgiler

REST

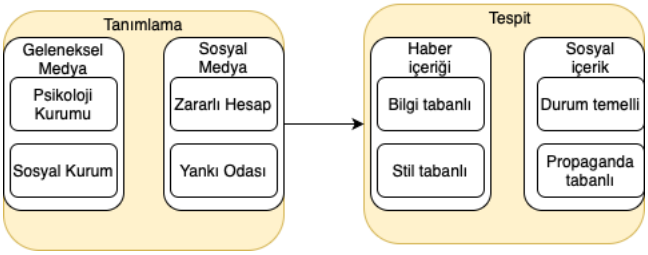
REST (Representational State Transfer), Roy Fielding tarafından geliştirilmiş XML, JSON, HTML gibi farklı veri formatlarını destekleyen istemci/sunucu modeline dayalıdır. Bu modelde, kaynaklara erişim için standart HTTP metodları (GET, POST, PUT, DELETE) kullanılmaktadır. Ayrıca, REST web servislerinin SOAP'a göre daha hızlı çalıştığı kabul edilmektedir (Arragokula & Ratnam, 2016; Mumbaikar & Padiya, 2013). Şekil 7'de TRT Gündem'den dönen haber başlık ve açıklama/özet REST formata dönüştürülmüş halidir.



Şekil 7. TRT gündem için REST formatında bilgiler

Sahte Haber Tanımı

Sahte haber, toplumu kasıtlı olarak manipüle etmek için yazılmış olan haber içerikleridir (Allcott & Gentzkow, 2017). Buradan bir sahte haber tanımının iki temel özelliğinin belirlenebileceği söylenebilir: özgünlük ve niyet. Bazı haberler kullanıcıyı yanıltmaya yönelik bilgiler barındırmasının yanısıra bazıları ise genellikle eğlence odaklı (Zaytung vb.) bilgiler içermektedir.



Şekil 8. Sahte haberler: Tanımlamadan teyite

Bu tanıma göre aşağıdaki maddeler sahte haber değildir(Shu vd., 2017):

- Eğlence Odaklı Haberler: Kullanıcıyı yanıltma amacı taşımayan, gerçek dışı olduğu kolayca belirlenebilen, eğlence amaçlı haberlerdir.
- Söylentiler: Haber niteliği taşımayan, gerçekliği teyit edilememiş bilgiler.
- Komplo Teorileri: Gerçek veya yanlış olduğu kesin olarak kanıtlanamayan spekülasyona dayalı bilgiler.
- İstem Dışı: Kasti olmayan yanlış bilgiler.
- Şaka/Dolandırıcılık İçerikli Haberler: Eğlence veya dolandırıcılık amacıyla üretilen gerçek dışı haberler.

Şekil 9. Sahte haberlerin dilbilimsel analiz süreci

Şekil 9'da geleneksel medyada yer alan sahte haberlerin analiz sürecine ait görsel yer almaktadır.

Gazete veya ilgili haber kaynağına ait web sayfalarında yer alan bilgilerin kendisi "geleneksel medya" olarak ifade edilmektedir. Geleneksel sahte haberler, genelde okuyucuları bireysel zayıflıklarından faydalanır. Bunlar özellikle bazı ideolojik gruplar tarafından yanlış bilgilerinin gerçek bilgi olarak sunulması yoluyla toplumda kargaşaya sebep olmaktadır.

Sahte ve yanıltıcı haberleri çeşitli türlerde sınıflandıran ve tanımlayan kurumlar bulunmaktadır. Bununla birlikte, sahte haberlerden bahsedilirken yanıltıcı bilgiler ile ilgili kavramlar Tablo 2'de sınıflandırılmıştır.

Tablo 2. Yanıltıcı Bilgilere ait Matrix (Thota vd., 2018)

	Hiciv ve Parodi	**Yanlış Bağlantı**	**Yanıltıcı İçerik**	**Yanlış İçerik**	**Sahte İçerik**	**Değiştirilmiş İçerik**	**Uydurma İçerik**
Kötü Gazetecilik		X	X	X			
Parodi	X				X		X
Kışkırtma					X	X	X
Tutku				X			
Partizanlık			X	X			
Kar		X			X		X
Siyasi Etki			X	X		X	X
Propaganda			X	X	X	X	X

- Yanıltıcı bilgi türleri aşağıda listelenmiştir:
- Hiciv: Kullanıcıyı yanıltma niteliği taşır fakat zarar verme niteliği taşımaz.
- Yanlış Bağlantı: Gerçek içeriği yanlış başlık ve altyazılar ile paylaşmak.
- Yanıltıcı İçerik: İçeriği belirli bir konuya oturtmak için altyazılı açıklamalar kullanmak.
- Yanlış İçerik: Gerçek içerik, yanlış içerik bilgisinde paylaşılır.
- Değiştirilmiş İçerik: Ham bilgiler ve resimler değiştirilmiştir.
- İçerik Sahtekarlığı: Bir haber kuruluşunu taklit ederek okuyucuyu manipüle ederek gerçek gibi görünen içerik.

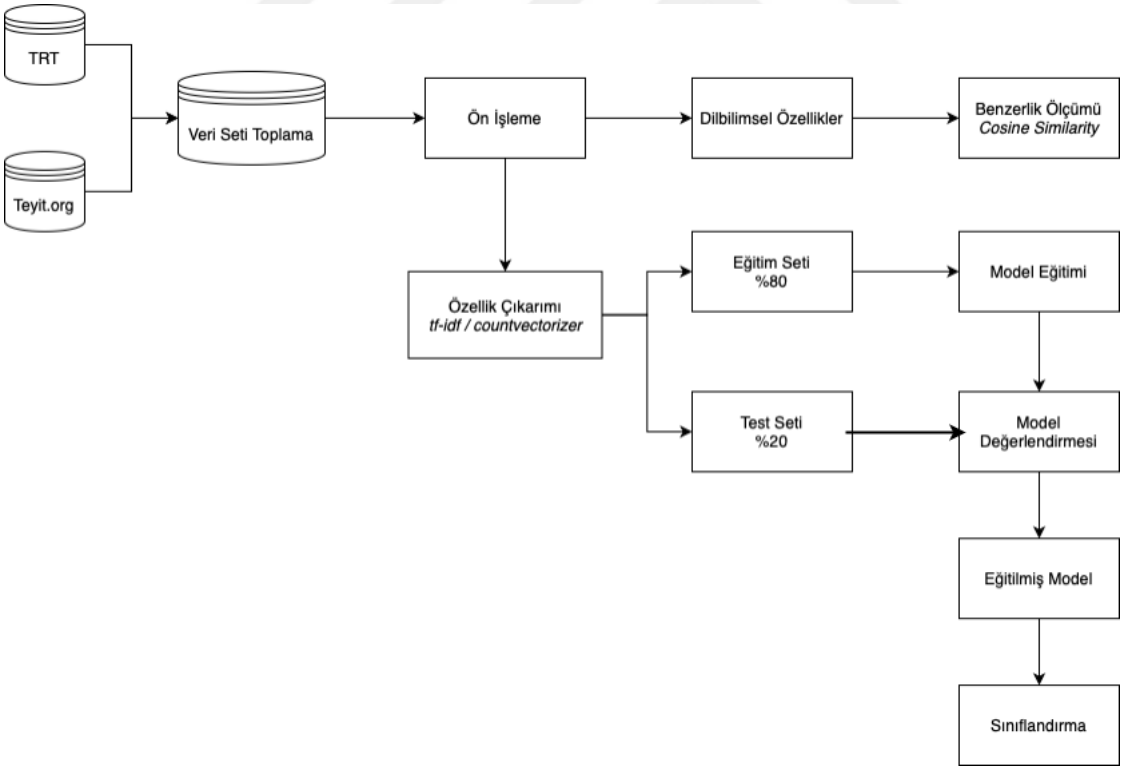
- Uydurma İçerik: Genellikle toplumu manipüle etmek amacıyla tasarlanmış yanlış ve yanıltıcı bilgiler.

ÜÇÜNCÜ BÖLÜM

Yöntem

Araştırma Yöntemi

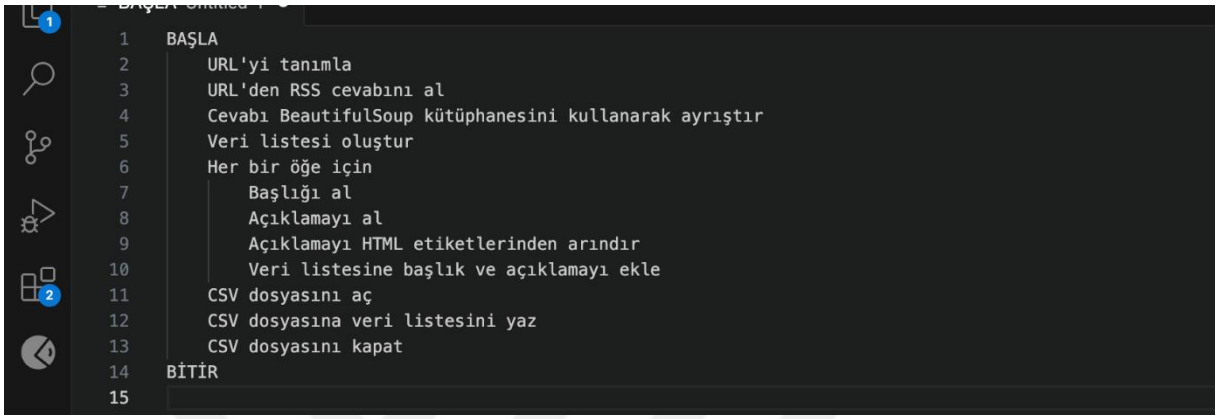
Sahte haberlerin gün geçtikçe artmasıyla birlikte bu tür haberlerin olumsuz etkilerinden korunmak için herhangi bir uzmana ihtiyaç duymadan sınıflandırma algoritmalarıyla tespit etme gerekliliği kaçınılmaz hale gelmiştir. Bu model de veri seti %80 eğitim ve %20 test olarak ayrılmıştır. Ardından klasik makine öğrenimi, LSTM ve BERTurk algoritmalarıyla sınıflandırma sonuçları incelenmiştir. Şekil 10'da Türkçe sahte haber tespit edilmesi için gerçekleştirilen işlemleri anlatan bir akış diyagramını aittir. İlk olarak, veri seti çeşitli kaynaklardan toplanarak birleştirilmiş ve ön işleme aşamaları tamamlanmıştır. Ardından, veri eğitim ve test setlerine ayrılarak, klasik makine öğrenmesi ve derin öğrenme algoritmaları ile sınıflandırma işlemi gerçekleştirilmiştir.



Şekil 10. Model için akış diyagramı

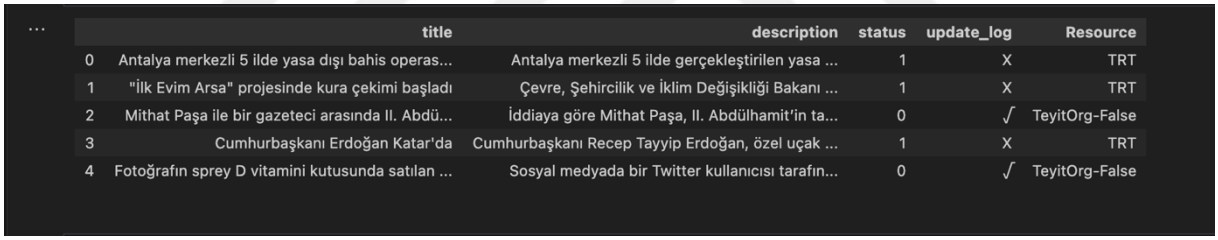
Veri Toplama

Bu çalışmada kullanılan veri seti, TRT Gündem'in RSS servisinden ve Teyit.org uzman tabanlı manuel kontrol platformundan toplanmıştır. İki farklı kaynaktan elde edilen veri setinin toplanması için ilgili SOAP servislerde kullanılan veri kazıma yöntemine ait sözde kod, Şekil 11'de yer almaktadır. Toplam 5325 adet haber içeren veri seti kullanılmıştır. Veri setinin ilk 5 satırının özetine ait görsel Şekil 12'de sunulmuştur.



```
BAŞLA
    URL'yi tanımla
    URL'den RSS cevabını al
    Cevabı BeautifulSoup kütüphanesini kullanarak ayrıştır
    Veri listesi oluştur
    Her bir öğe için
        Başlığı al
        Açıklamayı al
        Açıklamayı HTML etiketlerinden arındır
        Veri listesine başlık ve açıklamayı ekle
    CSV dosyasını aç
    CSV dosyasına veri listesini yaz
    CSV dosyasını kapat
BİTİR
```

Şekil 11. Veri setini toplanmasında kullanılan sözde kod

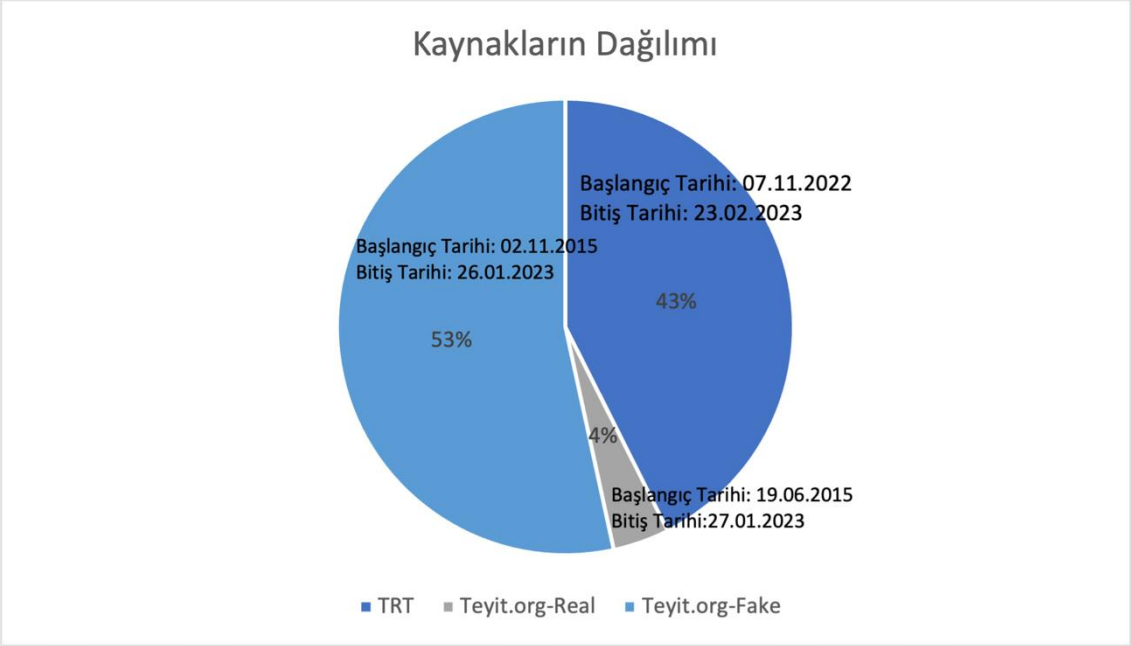


	title	description	status	update_log	Resource
0	Antalya merkezli 5 ilde yasa dışı bahis operas...	Antalya merkezli 5 ilde gerçekleştirilen yasa ...	1	X	TRT
1	"İlk Evim Arsa" projesinde kura çekimi başladı	Çevre, Şehircilik ve İklim Değişikliği Bakanı ...	1	X	TRT
2	Mithat Paşa ile bir gazeteci arasında II. Abdü...	İddiaya göre Mithat Paşa, II. Abdülhamit'in ta...	0	√	TeyitOrg-False
3	Cumhurbaşkanı Erdoğan Katar'da	Cumhurbaşkanı Recep Tayyip Erdoğan, özel uçak ...	1	X	TRT
4	Fotoğrafın sprey D vitamini kutusunda satılan ...	Sosyal medyada bir Twitter kullanıcısı tarafın...	0	√	TeyitOrg-False

Şekil 12. Veri setinden örnek bir kesit

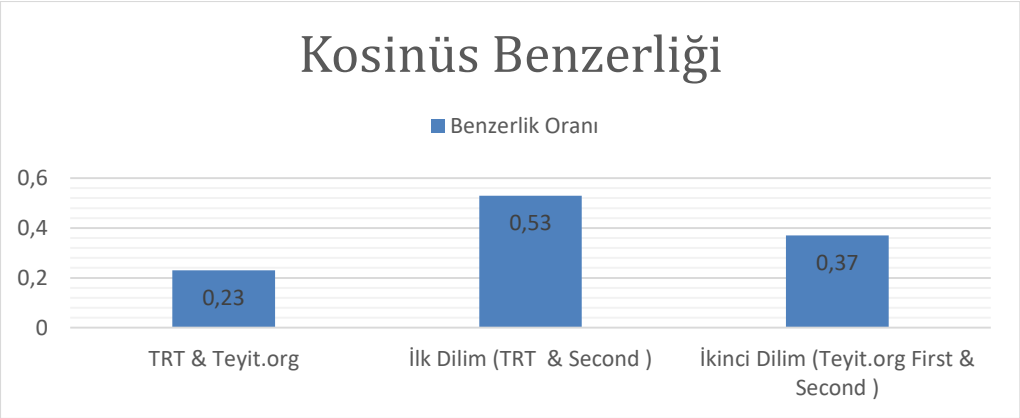
Ayrıca, veri setinde 5 adet sütun bulunmaktadır ve ilgili sütunların özellikleri aşağıda listelenmiştir. Şekil 13'de veri setindeki kaynakların dağılımı ve hangi tarih aralığında toplandığı görselleştirilmiştir.

1. title: Haberin Başlığı
2. description: Haberin Özeti
3. status: Sınıflandırma Sonucu - Sahte (0), Gerçek (1)
4. update_log: İlgili habere veri dönüşümü işlemi yapıldı mı? Evet (✓) Hayır (✗)
5. resource: Haberin Kaynağı



Şekil 13. Veri seti hakkında bilgiler

Çalışmada farklı veri kaynaklarından toplanan haberlerin birleştirilmesinin gerekliliğini açıklamak için yapısal benzerliklerinin incelenmesi önemlidir. Belgeler arasındaki benzerliklerin mesafeye dayalı ölçümü için Kosinüs Benzerliği veya Levenshtein Mesafe metrikleri kullanılmaktadır (Siahaan vd., 2018). Kosinüs Benzerliği, iki metin belgesi arasındaki yapısal benzerliği ölçmek için vektör modellemesini temel alır. 0 ile 1 aralığında değer alabilmektedir; 0 belgelerin ilişkisiz olduğunu, 1 ise aynı olduğunu belirtir (Falah & Suryawan, 2022; Rahutomo vd., 2012). Kosinüs metriği ile TRT ve Teyit.org'un kendileri arasındaki ve karşılıklı benzerlikleri hesaplanmıştır. Bu yöntemde, kelime haznesindeki benzersiz kelime sayısı ve metinlerdeki kelime frekansları dikkate alınırken, kelimelerin anlamı ve sıralaması göz ardı edilir. TRT'nin kendi arasındaki benzerliği 0.53, Teyit.org'un 0.37 olarak hesaplanmıştır. TRT ve Teyit.og haber kaynakları birbirleriyle karşılaştırıldığında ise 0.23 olarak benzerliği hesaplanmıştır. Bu sonuç, TRT ve Teyit.org arasındaki haber makalelerinin birbirine olan benzerliğinin düşük olmadığını göstermektedir. TRT, kendi içindeki benzerliği yüksek olsa da, Teyit.org'un kendi içindeki benzerlik oranı daha düşüktür. Bu farklılık, Teyit.org platformunda ana akım ve sosyal medya içeriklerinin de yer alıyor olmasıdır. Şekil 14'de benzerlik oranlarına ait görsel sunulmuştur.



Şekil 14. Kosinüs benzerliği oranları

TRT ve Teyit arasındaki en benzer ve en benzemez örnekler ile bunların benzerlik oranları aşağıda yer almaktadır. Bu yöntem yapılmadan önce parti isimleri ve kişi isimleri filtrelenmiştir.

En benzer haber başlık örneği:

- TRT: “Mehmetçik Hatay'da 15 bin ekmek, 10 bin kişilik yemek dağıtıyor”
- TeyitOrg: "Avustralya’da 10 bin vahşi devenin itlaf edileceği iddiası"
- Benzerlik Oranı: 0.36

İlk haber, yardım faaliyetlerine odaklanırken, ikinci haber Avustralya’da gerçekleşen hayvan itlafını ele almaktadır. Her iki kaynak da içerik olarak farklı olsada tutar bilgisi gibi bazı ortak terimler içermektedir. Bu sebeple kelime tabanlı bir vektörizasyon yöntemiyle yüksek benzerlik göstermiştir.

En benzemez haber başlık örneği:

- TRT: "Antalya merkezli 5 ilde yasa dışı bahis operasyonu: 89 gözaltı."
- TeyitOrg: "Alman bira firmasının piyasaya sürdüğü bira şişesinde 'La İlahe İllallah' yazdığı iddiası"
- Benzerlik Oranı: 0.0

TRT haberi yasa dışı bahis operasyonlarına odaklanırken, Teyit tamamen farklı bir konuyu, içecek firmasını ele almaktadır. Ortak kelimelerin bulunmaması nedeniyle benzerlik sıfır olarak hesaplanmıştır.

Veri Ön İşleme

Veri ön işleme, makine öğrenimi modellerinde eğitimden önce gerçekleştirilen bir dizi işlemi içerir. Ön işleme aşamaları için Şekil 15'deki adımlar takip edilmiştir.

Şekil 15. Veri ön işleme aşamaları

Teyit.org'dan toplanan sahte haberlerin miktarı, gerçek haberlerin miktarına kıyasla oldukça yüksektir (211/2845). Bu durum, veri setinde dengeli örneklerle temsil edilmediği için aşırı uyum(overfitting) problemini ortaya çıkardığı gözlemlenmiştir. Aşırı uyum, modelin test seti üzerinde genelleme yapmadan eğitim setini ezberlemesidir, bu da sınıflandırma sonucunu olumsuz etkileyebilir. Aşırı uyumu azaltmak için kullanılan yöntemler arasında stopwords, veri indirgeme ve veri dönüştürme vb. işlemi yapılmıştır. Son durumda, veri setinde %53,4 sahte haber %46,6 gerçek haber yer almaktadır. Veri setinde sık kullanılan kelimeleri (edatlar, bağlaçlar vb.) ezberlemesini önlemek için stopwords'leri kaldırmak gerekmektedir. Şekil 16'da ön işleme aşamasında cümlede noktalama veya regexlerin kaldırılması gibi işlemleri içeren kod bloğu gösterilmiştir.

```
title_desc = df[['title', 'description']]

def clean_dataset(data):
    temp = data.lower()
    temp = re.sub("rt @[A-Za-z0-9_]+","", temp)
    temp = re.sub("@[A-Za-z0-9_]+","", temp)
    temp = re.sub("#[A-Za-z0-9_]+","", temp)
    temp = re.sub(r'http\S+', '', temp)
    temp = re.sub('[()!?|:]', ' ', temp)
    temp = re.sub('\[.*?\]',' ', temp)
    temp = temp.split()
    temp = [w for w in temp if not w in stopwords]
    temp = " ".join(word for word in temp)
    return temp

df['title'] = title_desc.apply(lambda row : clean_dataset(row['title']), axis = 1)
df['description'] = title_desc.apply(lambda row : clean_dataset(row['description']), axis = 1)
```

Şekil 16. Veri ön işleme için kullanılan kod bloğu

Makine öğrenimi modellerinde gürültü azaltma, yanlış veya eksik verilerden kaynaklanan tutarsızlıkları ve hataları giderme işlemidir. Bu tür hatalar, modelin sınıflandırma sonucu üzerinde olumsuz etkiye yol açabilir. Teyit.org haber kaynağından toplanan veri setleri arasında karşılaşılan zorluklardan biri, cümlenin sonlarında "iddia edildi" ifadesinin sıklıkla kullanılmasıdır. Bu ifadeyi içeren haberlerin birçoğunun sahte haber olması ihtimali, bir ön yargıya veya haber kaynağının eğitim setinde ezberlenmesine neden olabilir. Bu

nedenle, bu ifadenin geçtiği yerler kendisinden önceki cümlenin anlamını koruyacak şekilde değiştirilmiştir.

Şekil 15'de son olarak görülen ön işleme aşamasıysa kök indirgeme yöntemidir. Kök indirgeme, NLP (Doğal Dil İşleme) de bir kelimenin kökünü belirleyebilmek için kullanılan morfolojik bir tekniktir. Bu yöntem, benzer anlamlara sahip kelimelerin aynı kesme noktalarıyla boyutunun küçültülmesine olanak sağlar. Çoğu açık kaynaklı NLP kütüphanesi Hint-Avrupa metinleriyle dilleriyle çalışırken, Türkçe gibi sondan eklemeli dillerde eklerin aşırı kullanımı nedeniyle NLP araştırmaları oldukça zorlaşır (Akın & Akın, 2017). Bu çalışmada, Türkçe metinler üzerinde çalışan açık kaynak kütüphanelerinden biri olan Zemberek'den faydalanılarak kök indirgeme (Stemming) işlemi gerçekleştirilmiştir. Kök indirgeme, kelimeden ekleri keserek o kelimenin en yalın kökünün bulunmasıdır. Lematizasyon ise kelimenin morfolojik yapısını dikkate alarak kelimenin sözlükteki esas formunu bulur.

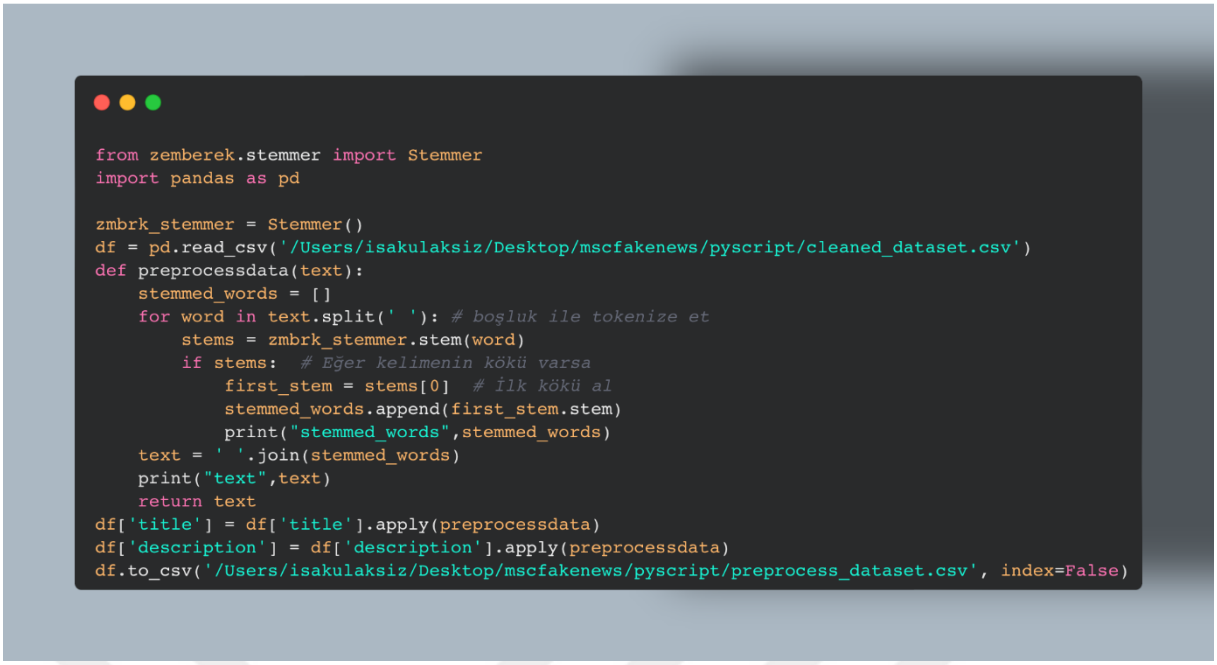
Zemberek kütüphanesini kullanarak "gidebilirseniz" kelimesinin kök indirgeme ve lematizasyon sonuçları aşağıda yer almaktadır.

- Kök indirgeme (stemming): gid, gidebil
- Lematizasyon (lemmatization): git, gidebil

Bunlara ek olarak, Zemberek Kütüphanesinde aşağıdaki gereksinimler ve daha fazlası için de çözümler bulunmuştur.

- Normalization (TurkishSentenceNormalizer): Girdinin hatalı karakterlerini düzeltmek ve eksik harfleri tamamlamak için kullanılır.
- Morphology (TurkishMorphology): Türkçe morfolojik analiz, anlam ayrımı ve kelime üretimi için kullanılır.
- Tokenization (TurkishTokenizer): Girdileri kelimelere ayıran bir ayraçtır.
- NER (Named Entity Recognition): İsimlendirilmiş varlık tespiti yapar.
- Classification (FastTextClassifier): FastText projesinin Java'ya uyarlanmış biçimidir.
- Language Identification (LanguageIdentifier): Girdinin yazı dilini tespit eder.
- Language Modelling (BaseLanguageModel): Dil modeli karşılaştırmasını sağlar.

Şekil 17'de veri setinde indirgeme işlemi için ilgili geliştirme yer alırken Şekil 18'de, eğitim için kullanılan güncel veri setine ait örnek sunulmuştur.



```python
from zemberek.stemmer import Stemmer
import pandas as pd

zmbrk_stemmer = Stemmer()
df = pd.read_csv('/Users/isakulaksiz/Desktop/mscfakenews/pyscript/cleaned_dataset.csv')
def preprocessdata(text):
    stemmed_words = []
    for word in text.split(' '): # boşluk ile tokenize et
        stems = zmbrk_stemmer.stem(word)
        if stems:  # Eğer kelimenin kökü varsa
            first_stem = stems[0]  # İlk kökü al
            stemmed_words.append(first_stem.stem)
            print("stemmed_words",stemmed_words)
    text = ' '.join(stemmed_words)
    print("text",text)
    return text
df['title'] = df['title'].apply(preprocessdata)
df['description'] = df['description'].apply(preprocessdata)
df.to_csv('/Users/isakulaksiz/Desktop/mscfakenews/pyscript/preprocess_dataset.csv', index=False)
```

Şekil 17. Kök indirgeme yöntemi için ilgili kod bloğu

	title	description	status	Resource
0	antalya merkez 5 il yas dış bahis operasyon 89...	antalya merkez 5 il gerçek yas dış bahis opera...	1	TRT
1	ev proje kur çek başla	şehir değişik bakan murat milyon altyapı hazır...	1	TRT
2	mithat paşa bir gazete ara abdülhamit hakkında...	göre mithat abdülhamit taht indir sonra bir ga...	0	TeyitOrg-False
3	cumhurbaşkanı erdoğan katar	cumhurbaşkanı recep tayyip özel uçak kat emir ...	1	TRT
4	fotoğraf sprey d vitamin kutu sat el dezenfekt...	sosyal medya bir twitter kullan tarafından new...	0	TeyitOrg-False

Şekil 18. Ön işleme sonrası güncel veri seti

Haber başlık ve özet kısmının adım adım kök indirgeme işleminde nasıl çalıştığına dair bir örnek log dosyasından alınarak aşağıda sunulmuştur:

Girdi: TBMM Genel Kurulu'nda, 2023 Yılı Merkezi Yönetim Bütçe Kanun Teklifi kabul edildi.

- stemmed_words[‘tbmm’]
- stemmed_words[‘tbmm’, ‘genel’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘yıl’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘merkezi’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘merkezi’, ‘yönet’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘merkezi’, ‘yönet’, ‘bütçe’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘merkezi’, ‘yönet’, ‘bütçe’, ‘kanu’]

- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘merkezi’, ‘yönet’, ‘bütçe’, ‘kanu’, ‘teklif’]
- stemmed_words[‘tbmm’, ‘genel’, ‘2023’, ‘merkezi’, ‘yönet’, ‘bütçe’, ‘kanu’, ‘teklif’,’kabul’]

Çıktı: tbmm genel 2023 merkezi yönet bütçe kanu teklif kabul

Şekil 19. Veri setinde en sık kullanılan kelimeler

Şekil 15’de, veri dönüştürme işleminde “iddia edildi” ifadesinin kendinden önceki kelimeyle değiştirilmesi sonrasında, veri setinde ön işleme öncesi ve sonrasında en sık geçen kelimeler Şekil 19’da kelime bulutu yöntemi ile gösterilmiştir. Bireylerin onurunu ve çıkar çatışmasını önlemek amacıyla özel isimler (“Atatürk”, “Erdoğan”, ”Çavuşoğlu”, “Bozdağ”) kelime bulutu’ndan kaldırılmıştır.

Özellik Çıkarımı

Büyük ölçekli veri kümeleriyle çalışırken, metin gruplaması önemli bir zorluk haline gelir. Modelin performansını etkileyen önemsiz ve ilişkisiz kelimelerin varlığı, bu zorluğu daha da artırır. Bundan dolayı, bir belgedeki kelimelerin kullanım sıklığı ve önemini

belirlemek için TF-IDF (Term Frequency-Inverted Document Frequency), CountVectorizer ve Word2Vec gibi yöntemler kullanılarak metinler sayısal vektörlere dönüştürülür (Kaur vd., 2020). Bu yöntemlerin yanısıra, kelime çantası ve n-gram öznitelik çıkarımı yöntemleri aşağıda açıklanmıştır.

TF-IDF (Term Frequency-Inverted Document Frequency)

TF-IDF, bir metindeki her kelimenin önemini ölçmek için kullanılan bir ölçüttür. Bu yöntemde, bir kelimenin önemi, kullanım sıklığı arttıkça artar. TF-IDF yöntemi, terim frekansı (TF) ve ters belge frekansı (IDF) olmak üzere iki ölçümden oluşur. Bir t terimi için TF-IDF'nin hesaplanması için kullanılan formül aşağıda verilmiştir.

$$\text{tf} - \text{idf} = tf(t,d) * idf(t)$$

N, belgedeki kelime sayısıdır ve df(t), t'deki belge sıklığıdır.

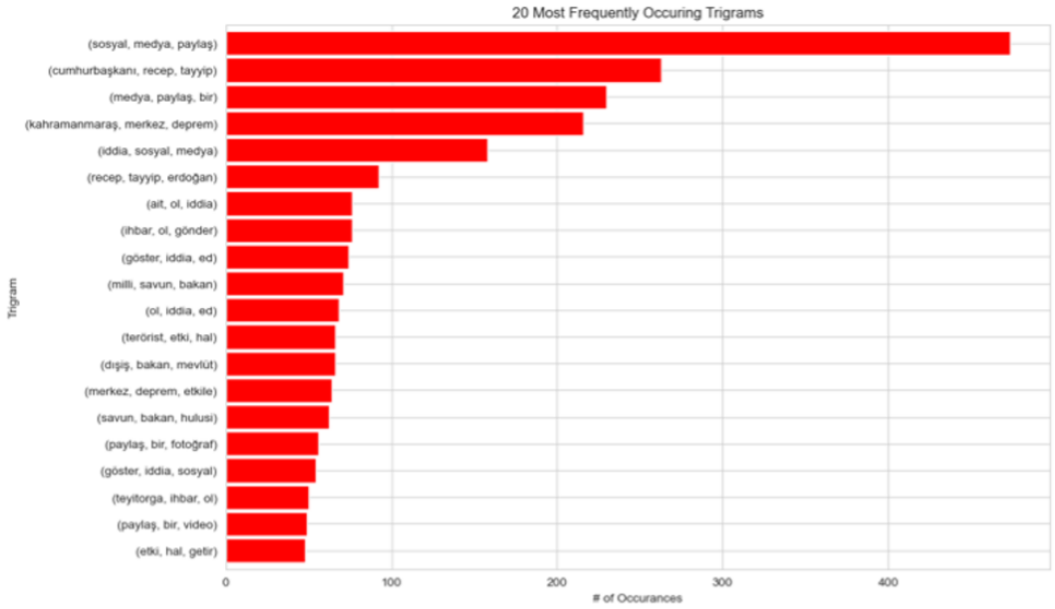
$$idf(t) = \log \frac{\text{n}}{\text{df(t)}} + 1$$

CountVectorizer

CountVectorizer, bir belgedeki kelimelerin sıklığını esas alır. Özellikle, NLP alanında yapılan çalışmalarda kelime öbeklerini parçalamada kullanılmaktadır. CountVectorizer, kelime öbeklerinin seçimini iyileştirmek için unigramlar (min_d f = 1), bigramlar (min_d f = 2) ve trigramlar (min_d f = 3) gibi birçok parametre kullanılır (Kaur vd., 2020). Şekil 20'de n-gramlar kelimeler kullanılarak oluşturulmuş kod bloğu yer almaktadır. Şekil 21'de ise en sık geçen 20 kelime bu yöntemle görselleştirilmiştir.

```
nltk.download('wordnet')
nltk.download('omw-1.4')

def word_lemmatizer(text):
  wnl = nltk.stem.WordNetLemmatizer()
  words = re.sub(r'[^\w\s]', '', text).split()
  return [wnl.lemmatize(word) for word in words if word not in stopwords]

words = word_lemmatizer(' '.join(df['title'].tolist() + df['description'].tolist()))

trigrams_series = (pd.Series(nltk.ngrams(words, 3)).value_counts())[:20]
trigrams_series.sort_values().plot.barh(color='red', width=.9, figsize=(12, 8))
plt.title('20 Most Frequently Occuring Trigrams')
plt.ylabel('Trigram')
plt.xlabel('# of Occurances')
plt.show()
```

Şekil 20. N-gram için python kodu

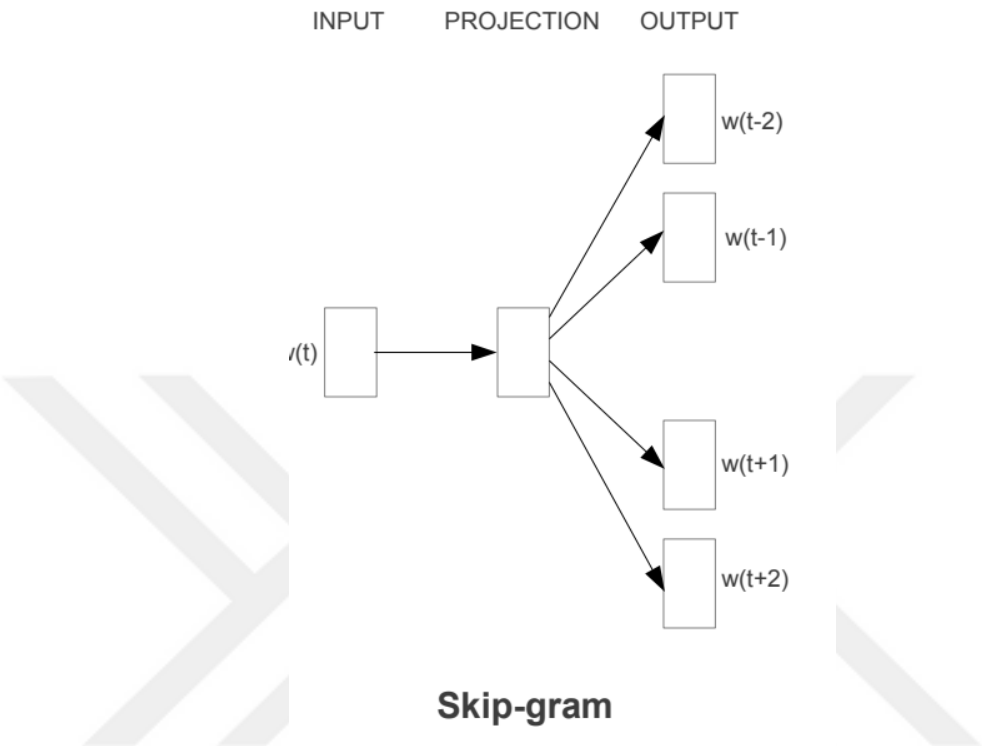


Şekil 21. En sık geçen kelimelerin triagram gösterimi

Word2Vec

(Mikolov vd., 2013) tarafından 2013 yılında geliştirilen Word2Vec, büyük veri kümelerinde, kelimeler arasındaki semantik ve sözdizimsel benzerlikleri yakalamak amacıyla kullanılan tahmine dayalı bir kelime temsil yöntemidir. Her kelimeyi vektör uzayında temsil eder, benzer anlamlara sahip kelimelerin birbirine yakın konumlandırılmasını sağlar. CBOW ve Skip-gram gibi farklı öğrenme algoritmalarına sahip iki temel yöntemi bulunmaktadır. Bu

araştırmada, Wor2Vec kelime temsil yönteminde Skip-gram algoritması tercil edilmiştir. Skip-gram, bir cümlenin merkezindeki kelimeyi girdi olarak alır ve bu kelimenin çevresindeki kelimeleri tahmin etmeye çalışır (Aydoğan & Karcı, 2019). Bu durum, Şekil 22'deki görselde sunulmuştur.



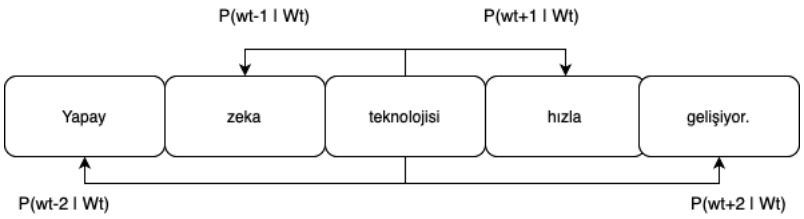
Şekil 22. Skip-gram modelinin yapısı (Mikolov vd., 2013)

Model eğitiminde, kullanılan parametreler ve açıklamaları Tablo 3'de verilmiştir.

Tablo 3. Word2Vec model parametreleri ve açıklamaları

Parametre Adı	**Açıklama**	**Parametre**
sentences	Eğitim verisi	X_train_tokenized
vector_size	Kelimelerin boyutu	300
window	Pencere büyüklüğü	5
min_count	Minimum frekans	1
workers	Eğitim esnasında kullanılan paralel işçi sayısı	1
epochs	Eğitim Süresi	10

Şekil 23'de, bir cümle Skip-gram algoritmasının adımları görselleştirilmiştir. Pencere boyutu 1 olarak alındığında, "teknolojisi" girdisinin çıktıları "zeka" ve "hızla"dır.



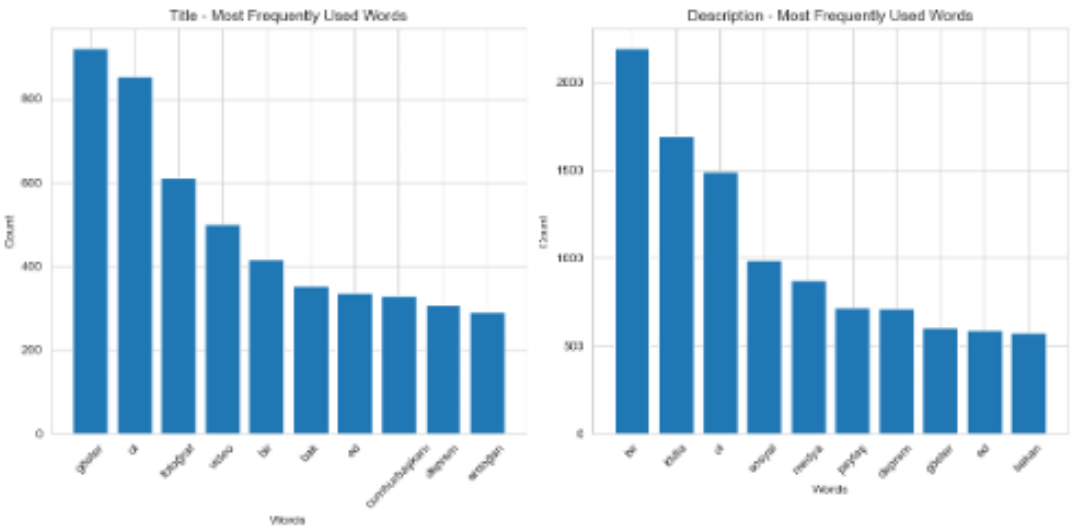
Şekil 23. Skip-gram modelinin örnek üzerinde çalışma prensibi

Kelime Çantası(Bag of Words)

BOW, temel itibariyle Terim Sıklığı'na (TF) benzer, çünkü metin içeriğinden sözcük sıklıklarına göre sözcük grupları (n-gram) çıkarılabilir. Başka bir deyişle, Terim Sıklığı'na (TF) benzemektedir çünkü metin içeriğinden sözcük sıklıklarına göre sözcük grupları (n-gram) çıkarılabilmektedir (Ahmed vd., 2017). Şekil 24'de kelime çantası yöntemi için kullanılan python kodu Şekil 25'de ise bu yöntem sonucunda en sık geçen kelimeler görselleştirilmiştir.

```
title_x = [word for word, count in title_word_counts.most_common(10)]
title_y = [count for word, count in title_word_counts.most_common(10)]

desc_x = [word for word, count in desc_word_counts.most_common(10)]
desc_y = [count for word, count in desc_word_counts.most_common(10)]
```

Şekil 24. BOW yöntemi için kod bloğu



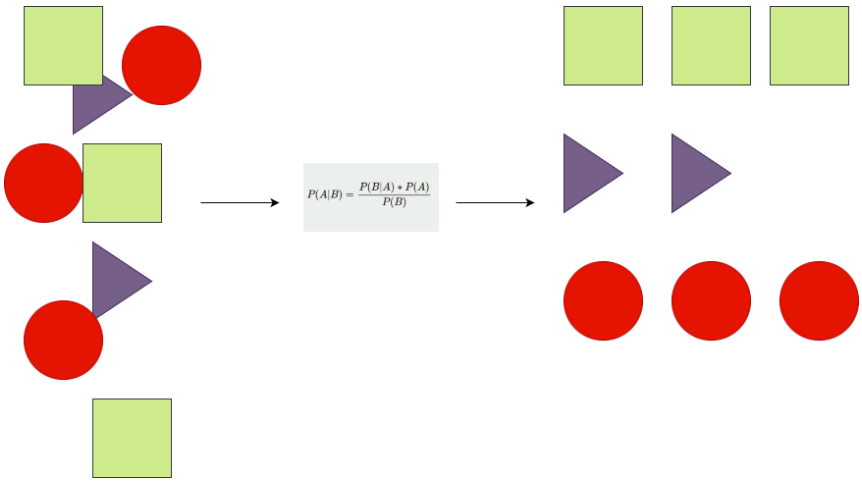
Şekil 25. En sık geçen kelimelerin BOW gösterimi

Sınıflandırma Algoritmaları

Aşağıda, SVM, Lojistik Regresyon, Rastgele Orman, k-NN, Karar Ağacı, Naive Bayes, LSTM ve BERTurk algoritmalarının açıklamaları ve sınıflandırma performansını etkileyen hiperparametre optimizasyonunda kullanılan parametrelere yer verilmiştir. Hiperparametre optimizasyonu eğitim aşamasında yapılmaktadır. Makine öğrenimi algoritmalarında kullanılan hiperparametreler için uzman görüşleri alınarak ilgili değerler denenmiştir. Bu işlem sonrasında seçilen en iyi hiperparametreler ile model test verisinde değerlendirilir ve doğruluğu hesaplanmaktadır.

Naive Bayes

Naive Bayes, veri setinin daha sınırlı olduğu ve parametre tahmininin olduğu durumlarda etkilidir. Bu bağlamda, olasılıklar bir kez hesaplanır ve hafızaya kaydedilir ardından bu modelin olasılıkları tek tek dikkate alınarak çıkarılır (Alpaydın, 2011; Atasoy, t.y.). Şekil 26'da Naive Bayes algoritmasına ait bir görsele yer verilmiştir.

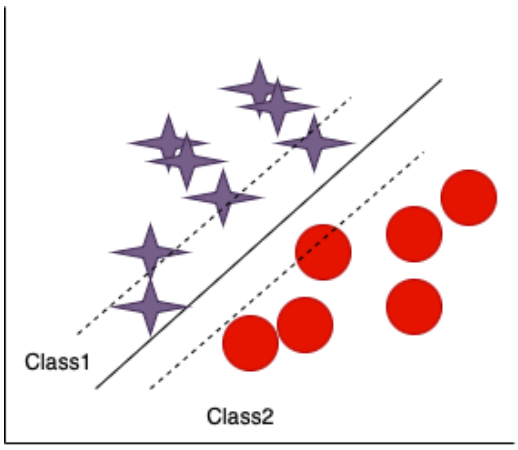


Şekil 26. Naive Bayes temsili gösterimi

Şekil 26'da bulunan formülde, P(A|B), herhangi bir örneği gözlemlemeden önce A parametre değerlerinin olasılıklarını sağlar. P(B|A) ise, dağılım A parametre değerlerini aldığında B örneğini elde etme olasılığını ifade eder (Alpaydın, 2011).

Destek Vektör Makinesi (SVM)

SVM, doğrusal ve doğrusal ayrılamayan büyük ölçekli veri setleri üzerinde sınıflandırma gerçekleştirir. SVM'de veri seti iki boyutlu bir vektör uzayında noktalar olarak temsil edilir. Bernhard Scholkopf'a göre, SVM'nin diğer makine öğrenimi algoritmalarından ayırt edici en önemli üstünlüğü, hesaplamalı öğrenme teorisinin ilkelerini kullanarak teorik inceleme kapasitesinde yatmaktadır ve aynı zamanda gerçek dünyadaki problemlerin çözümünde etkili olduğunu göstermektedir (Hearst vd., 1998; Kaur vd., 2020). Şekil 27'de SVM algoritmasına ait bir görsele yer verilmiştir.



Şekil 27. SVM temsili gösterimi (Hearst vd., 1998)

$$J(\theta)^n = {}^1\!/_2 \sum_{j=1}^{n} \theta_j^2$$

$$\theta^T x^i \geq 1, y^i = 1,$$

$$\theta^T x^i \leq -1, y^i = 0,$$

SVM ile birlikte bir modelin maliyet fonksiyonu yukarıdaki formülde (Ahmad vd., 2020) doğrusal bir çekirdek kullanılarak hesaplanır.

Tablo 4'de SVM algoritmasına ait parametre ayarlamalarına yer verilmiştir.

Tablo 4. SVM için hiperparametre optimizasyonu

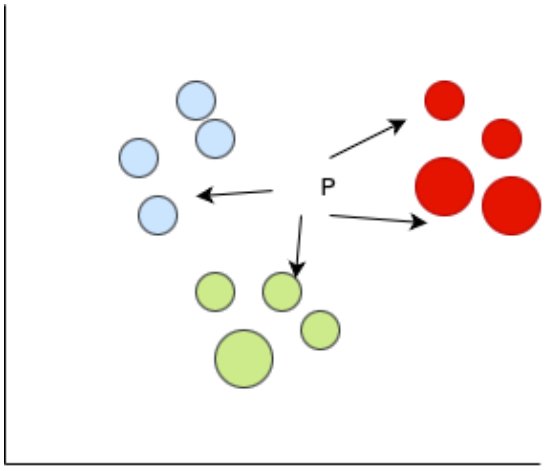
Model	**Giriş Parametreleri**	**En İyi Parametreler**
SVM	'C': [0.1, 1, 10], 'kernel': ['linear', 'rbf']	'C': 1, 'kernel': 'rbf'}

Parametre açıklamaları:

- 'C': SVM modelinin aşırı uyum(overfitting) durumunu önlemesi için kullanılan bir parametredir.
- 'kernel' : SVM modelinin fonksiyon tipidir.(linear, kernel,poly)

k-En Yakın Komşu

KNN, temel olarak tahmin edilecek değerin oy çokluğuyla belirlenmesine dayanır. K noktası, belirli bir noktaya en yakın komşuların sayısını temsil eder (Ahmad vd., 2020). Belirlenen noktanın, tüm gözlem noktalarına olan uzaklıkları hesaplandıktan sonra büyükten küçüğe doğru sıralanıp aralarından en küçük k tanesi seçilir. Ardından seçilen gözlem noktasının hangi sınıf değerinde tekrarlandıysa o sınıfa atanması sağlanır. Şekil 28'de algoritmasına ait bir görsele yer verilmiştir.



Şekil 28. k-NN Temsili Gösterimi

Mesafe ölçümleri için aşağıdaki metrikler kullanılmaktadır.

$$\text{Öklid: } \sqrt{\sum_{i=1}^{k} (x_i - y_i)}$$

$$Manhattan: \sum_{i=1}^{k} |x_i - y_i|$$

$$\text{Minkowski: } \left(\sum_{i=1}^{k} (|x_i - y_i|)^q\right)^{1/q}$$

Tablo 5'de k-NN algoritmasına ait parametre ayarlamalarına yer verilmiştir.

Tablo 5. k-NN için hiperparametre optimizasyonu

Model	**Giriş Parametreleri**	**En İyi Parametreler**
k-NN	'n _neighbors': [3, 5, 7], 'weights':['uniform','distance'], 'algorithm': ['auto', 'ball_tree', 'kd_tree']	'algorithm':'auto', 'n_neighbors':7,'weights':'distance'}

Parametre açıklamaları:

- 'n-neighbors': En yakın komşuların sayısıdır.
- 'algorithm' : k-NN algoritmasında kullanılan komşuluk arama algoritmasıdır. (auto, ball_tree, kd_tree, brute)

- ‘weights’ : Bu parametre komşulara atanan ağırlıklıları belirtir.(distance, uniform)

Lojistik Regresyon

Lojistik Regresyon, temeli ikili çıktıya (doğru/yanlış vb.) sahip bir sonucun olasılığına dayanmaktadır (LaValley, 2008). İstatistiksel tahmine dönüştürmek için “sigmoid” fonksiyonunu kullanır. Optimal olasılık tahmini sağlamak için maaliyet fonksiyonunu en aza indirmek gerekmektedir.

$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

Maliyet fonksiyonu aşağıdaki formül ile hesaplanır.

$$Cost(p(x), y) = \begin{cases} log\big(p(x)\big), & y = 1, \\ -log\big(1 - p(x)\big), & y = 0, \end{cases}$$

Tablo 6’da Lojistik Regresyon algoritmasına ait parametre ayarlamalarına yer verilmiştir.

Tablo 6. Lojistik Regresyon için hiperparametre optimizasyonu

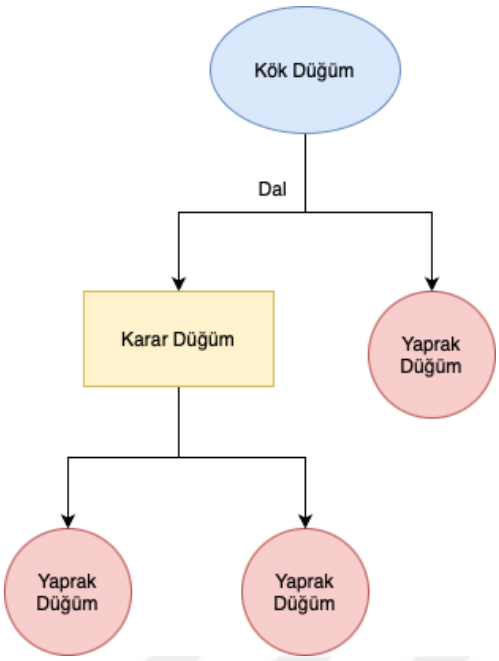
Model	**Giriş Parametreleri**	**En İyi Parametreler**
Lojistik Regresyon	'penalty': ['11', '12'], 'C': [0.1, 1, 10]	{'C': 1, 'penalty': 'l2'}

Parametre açıklamaları:

- ‘C’: Bu değer C(Ceza) parametresi modelin karmaşıklığı ile hata oranı arasındaki dengeyi kontrol etmeye yardımcı olur.
- ‘penalty’: Lojistik Regresyon modelinin ceza türünün L2 olduğunu gösterir.

Karar Ağacı

Karar Ağaçları, temel itibariyle akış şemalarına benzemektedir. Bu yapı içerisinde dallar, yapraklar ve düğümler bulunur. Her düğüm, bir özelliği temsil ederken son düğüme “yaprak” ve en üstteki düğüme ise “kök” denir (Quinlan, 1996). ID3 ve C4.5 algoritmalarıyla entropiye dayalı karar ağacı oluşturma yöntemleri mevcuttur. Şekil 29’da Karar Ağacı algoritmasına ait bir görsele yer verilmiştir.



Şekil 29. Karar ağacı temsili gösterimi

Kullanım alanları (Özkan, 2020):

- Bankaya kredi almak için başvuran müşterilerin kredi-risk durumunun “iyi” ve “kötü” olarak sınıflandırılması.
- Sosyal medyada paylaşılan gönderiler üzerinde duygu durumunun “pozitif” ve “negatif” olarak sınıflandırılması.
- Tıp alanında hastaların genleri üzerinde yapılan çalışmalarda hangi tip kanser türü olduğunun belirlenmesi.

Tablo 7’de Karar Ağacı algoritmasına ait parametre ayarlamalarına yer verilmiştir.

Tablo 7. Karar Ağacı için hiperparametre optimizasyonu

Model	**Giriş Parametreleri**	**En İyi Parametreler**
Karar Ağacı	'criterion': ['gini', 'entropy'], 'max_depth': [None, 10, 20], 'min_samples_split':[12,5,10], 'min_samples_leaf': [1, 2, 41	{'criterion':'gini', 'max_depth':20, 'min_samples_leaf':4, 'min_samples_split': 10}

Parametre açıklamaları:

- ‘criterion’ : Karar ağacının bölünme noktasının seçimindeki metriktir.(gini, entropy)

- ‘max_depth’ : Karar ağacının maksimum derinliğini tanımlar.
- ‘min_samples_leaf’ : Karar ağacı üzerindeki bir yaprağın sahip olduğu minimum örnektir.
- ‘min_samples_split’ : Bir düğümün(node) bölünmeden önce kaç örneğe sahip olması gerektiğini tanımlar.

Rastgele Orman

Rastgele Ormanlar, yüksek doğruluk elde etmek için regresyon ağaçlarını kullanarak aykırı değerleri kaldırır ve aşırı uyumu(overfitting) engeller (Y. Liu vd., 2012).

Kullanım alanları (Rigatti, 2017):

- Sağlık alanında yapılan çeşitli biyolojik çalışmalar
- Veri bilimi projeleri

Tablo 8’de Rastgele Orman algoritmasına ait parametre ayarlamalarına yer verilmiştir.

Tablo 8. Rastgele Orman için hiperparametre optimizasyonu

Model	**Giriş Parametreleri**	**En İyi Parametreler**
Rastgele Orman	_estimators': [50, 100, 200], 'max_depth': [None, 10, 201, 'min_samples_split':[2,5, 10], 'min _samples_leaf': [1, 2,4]	{'max_depth':None, 'min_samples_leaf':1, 'min_samples_split':2, 'n_estimators': 100}

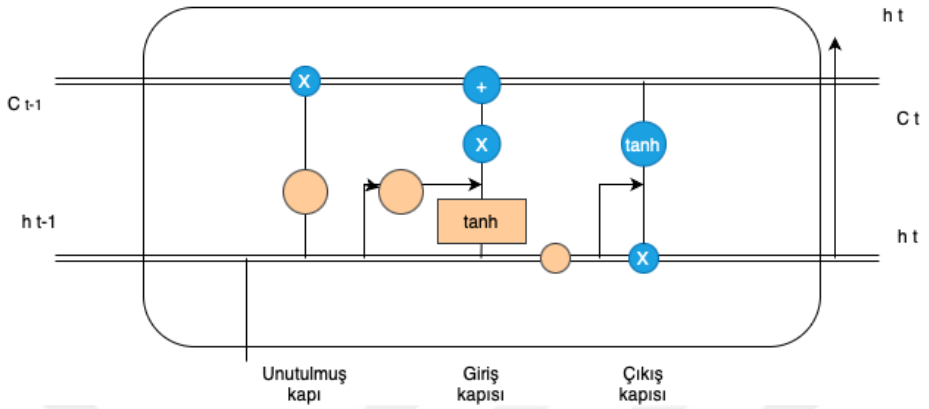
Parametre açıklamaları:

- ‘max_depth’ : Her bir karar ağacının maksimum derinliğini ifade eder. None, değeri herhangi bir sınırlılık olmadığını ifade eder.
- ‘min_samples_leaf’ : Bir yaprağın sahip olduğu minimum örnektir.
- ‘n-estimators’: Model için kullanılacak karar ağacının sayısını belirtir.

Uzun Kısa Süreli Bellek (LSTM)

Tekrarlayan Sinir Ağları'nın (RNN) önemli bir dezavantajı, belirli bir 't' anına kadar tüm önceki adımlardan gelen girdilerle ilgili bilgiyi tutabilmesidir. Ancak, bu yöntemle modelin uzun sürelerde öğrenilmesi zorlaşır. Bu sebeple, Uzun Kısa Süreli Bellek (LSTM)

algoritması, Hochreiter ve Schmidhuber tarafından 1997'de gradyan sönümleme problemi üzerine yapılan çalışmaların zirvesini oluşturmuştur (Chollet, 2019; Hochreiter & Schmidhuber, 1997). Şekil 30'da LSTM algoritmasına ait bir görsele yer verilirken ANN gradyan bozulması probleminin çözümü için unutma kapıları kullanıldığı görülmektedir.



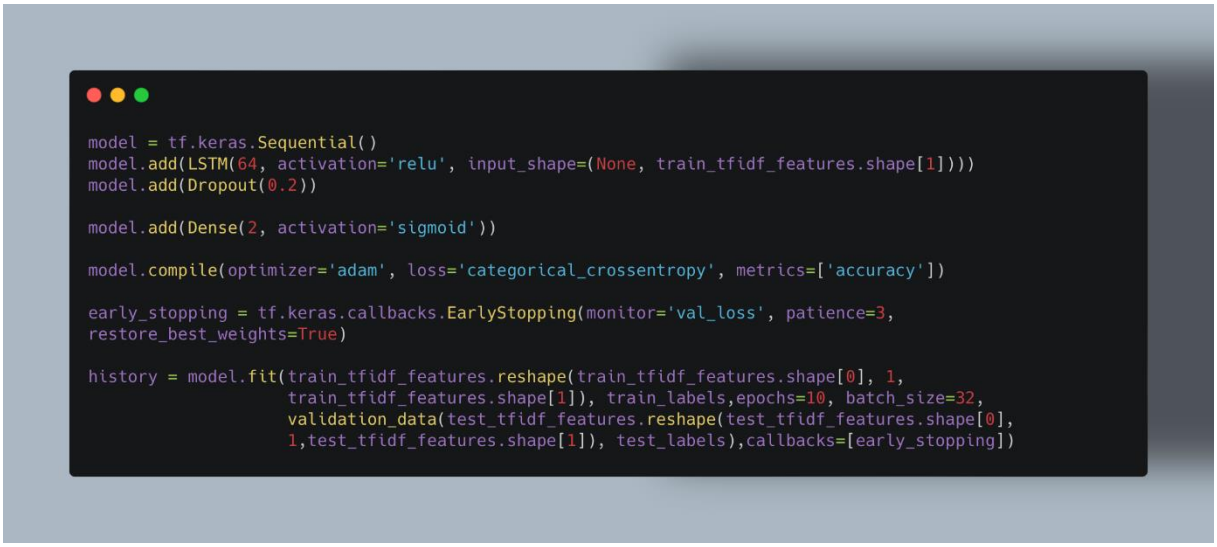
Şekil 30. LSTM temsili gösterimi (Chollet, 2019; Hochreiter & Schmidhuber, 1997)

Şekil 30'daki LSTM modelinde, TF-IDF ve CountVectorizer yöntemleriyle sayısal vektörlere dönüştürüldükten sonra 64 nöronlu bir giriş katmanıyla beslenir. Modelin aşırı öğrenme durumunu önlemek için dropout ve erken durdurma yöntemleri uygulanmıştır. Çıkış katmanında ise sigmoid aktivasyonu kullanılmıştır. Son olarak model, adam optimizasyonu ve ikili çapraz entropi kaybı ile derlenir ve eğitilir.

Parametre açıklamaları:

- 'optimizer' : Bu parametre modelin ağırlıklarının güncellenme yöntemini ifade eder.
- 'loss': Modelin hatalı tahminleri için maliyet fonksiyonunu tanımlar.
- 'epochs': Modelin eğitim setini kaç kez gezineceğini tanımlar.
- 'batch_size': Bu değer, eğitim aşamasının her bir adımında modele beslenen örnek sayısıdır.
- 'validation_data': Modele eğitim esnasında bir doğrulama veriseti sağlayarak performansını değerlendirmek için kullanılır.
- 'verbose': Her bir epoch sonunda ilerleme durumunu yazdırmak için kullanılır.

Şekil 31'de Sınıflandırma işlemi için LSTM modeline ait kod bloğu yer almaktadır.

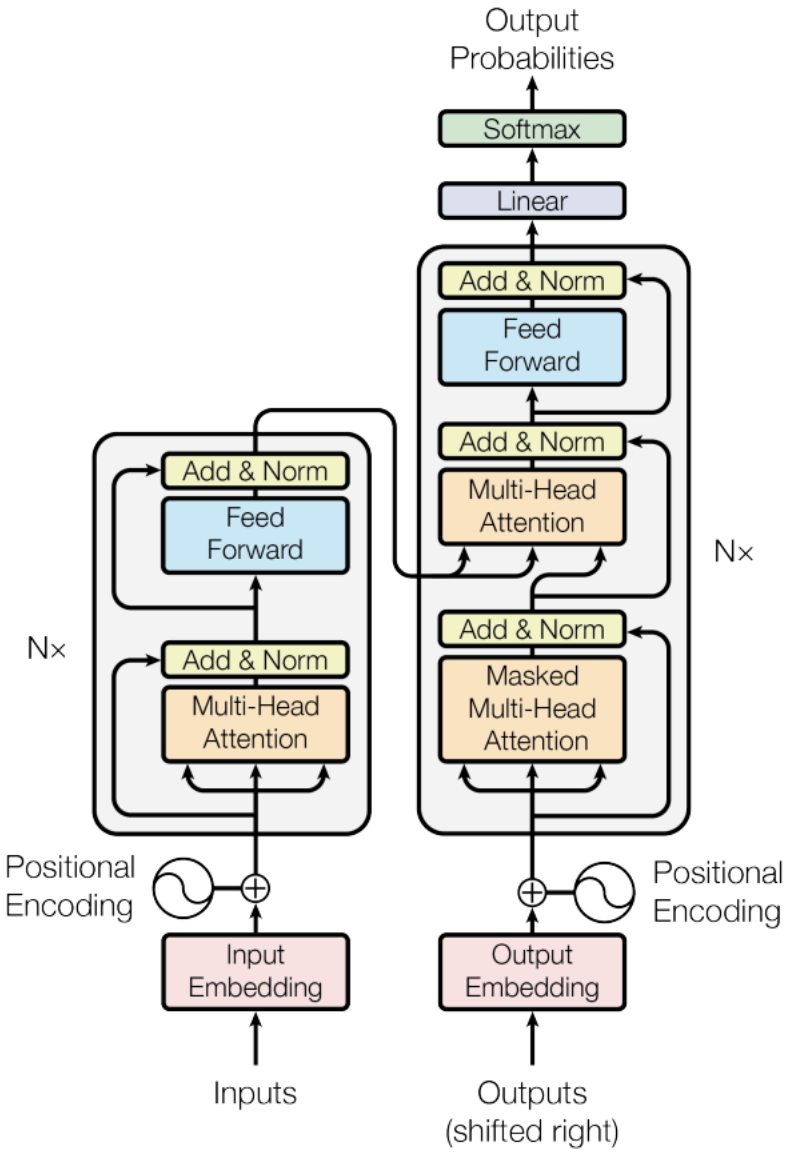


```
model = tf.keras.Sequential()
model.add(LSTM(64, activation='relu', input_shape=(None, train_tfidf_features.shape[1])))
model.add(Dropout(0.2))

model.add(Dense(2, activation='sigmoid'))

model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

early_stopping = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=3,
restore_best_weights=True)

history = model.fit(train_tfidf_features.reshape(train_tfidf_features.shape[0], 1,
                    train_tfidf_features.shape[1]), train_labels,epochs=10, batch_size=32,
                    validation_data(test_tfidf_features.reshape(test_tfidf_features.shape[0],
                    1,test_tfidf_features.shape[1]), test_labels),callbacks=[early_stopping])
```

Şekil 31. LSTM modelinin mimari kodu

BERTurk

Google, 2017 yılında yayımlanan (Waswani vd., 2017) "Attention is All You Need" adlı makalesinde, özellikle doğal dil işleme (NLP) alanında önemli bir yenilik olan Transformer modelini tanıtmıştır. Bu model, sıralı verilerle çalışan geleneksel RNN modellerinin aksine, tamamen dikkat (attention) mekanizmalarına dayanan bir model önermişlerdir. Transformer modelinin temelinde iki ana bileşen bulunur: dikkat (self-attention) ve kodlayıcı-çözücü (encoder-decoder) yapıları. Dikkat mekanizması, her bir kelimenin diğer kelimelerle olan bağlamını dikkate alırken, kodlayıcı (encoder) giriş verisini gizli bir temsile dönüştürür ve çözücü (Decoder) bu temsili kullanarak doğru çıktıyı üretir. Sınıflandırma ve NER gibi işlerde yalnızca kodlayıcı (encoder) içeren modeller (BERT ve ROBERTA) daha etkili olurken özetleme veya sohbet robotu gibi işlerde ise çözücü (decoder) içeren modeller (BART, GPT ve T5) tercih edilmektedir.

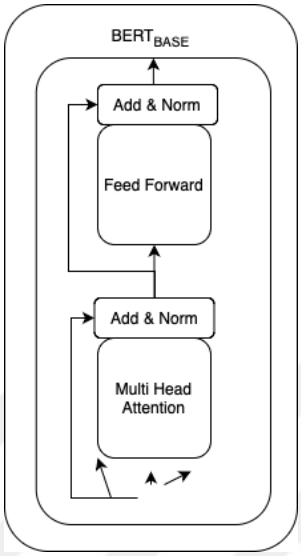


Şekil 32. Transformer model mimarisi (Waswani vd., 2017)

2018 yılında Google tarafından geliştirilen BERT (Bidirectional Encoder Representations from Transformers), sınıflandırma, duygu analizi ve NER işlerinde adeta isviçre çakısı gibi kullanılır. Bu model, kelimelerin bağlama dayalı anlamlarını öğrenmek için çift yönlü bir dikkat mekanizması kullanır. BERT modeli, Wikipedia ve Google'ın BooksCorpus devasa veri seti kullanılarak eğitilmiştir. Bu büyük veri setini işlemek için Google, 64 TPU kullanarak modeli dört gün boyunca eğitmiştir (Devlin vd., 2019). Maskeli Dil Modeli (MLM), bir cümledeki bazı kelimeleri maskeler ve BERT'i bu kelimeleri tahmin etmeye zorlar. Bu yaklaşım, maskelenen kelimenin hem solundaki hem de sağındaki kelimeleri dikkate alır, modelin metni çift yönlü bir şekilde anlamasını sağlar ve bu sayede bağlama dayalı çok güçlü bir öğrenme süreci gerçekleştirilir. Sonraki Cümle Tahmini (Next

Sentence Prediction, NSP) ise modelin iki cümlenin ardışık olup olmadığını takip ederek cümleler arasındaki ilişkileri öğrenmesine yardımcı olur.

Şekil 33'de BERTbase modelinin mimarisi görselleştirilmiştir. Bu model, 12 Transformer katmanı, 768 gizli boyutu, 12 dikkat başlığı ve 110 milyon parametre ile 4 TPU kullanılarak 4 gün boyunca eğitilmiştir.



Şekil 33. BERT model mimarisi

Tablo 9. BERT modelinin eğitiminde kullanılan parametreler ve açıklamaları

Parametre	**Değer**	**Açıklama**
MAX_LEN	300	Girdinin maksimum uzunluğu
TRAIN_BATCH_SIZE	4	Eğitim esnasında her bir adımda işlenecek örnek sayısı
VALID_BATCH_SIZE	4	Doğrulama esnasında her bir adımda işlenecek örnek sayısı
EPOCH	1	Modelin, tüm eğitim veri seti üzerinde geçiş sayısı
LEARNING_RATE	1e-05	Modelin, ağırlıklarını güncellerken kullanılan öğrenme oranı (epsilon)

Tablo 9'da BERTurk modelinin eğitimi için kullanılan parametre ve açıklamalara yer verilmiştir.

Değerlendirme Metrikleri

Modelin sınıflandırma performansını tespit edebilmek için öncelikle eğitim seti üzerinde tahminlerin yapılması gerekmektedir. Sınıflandırma işlemlerinde gerçek veri ile tahmin değerini kıyaslamak amacıyla karışıklık matrisinden (confusion matrix) faydalanılır. Matrisin satırları gerçek değerleri, sütunları ise tahmin edilen sınıf etiketlerini içermektedir. Tablo 10'da karışıklık matrisinin her bir elemanı aşağıdaki tabloda gösterilmektedir.

Bu tabloya göre:

- TP (True Positive): Bir sınıflandırma modelinin gerçekte pozitif olan bir örneği (sahte haber) doğru bir şekilde pozitif olarak tahmin ettiği durumları temsil eder. Tablo 10'da 100 haber gerçekten sahte iken, model doğru bir şekilde bunları sahte olarak tahmin etmiştir.
- FN (False Negative): Bir sınıflandırma modelinin pozitif olan bir örneği negatif olarak tahmin ettiği durumları temsil eder. Tablo 10'da 10 haber gerçekten sahte iken, model bu haberleri gerçek haber olarak yanlış tahmin etmiştir.
- FP (False Positive): Bir sınıflandırma modelinin negatif olan bir örneği pozitif olarak tahmin ettiği durumları temsil eder. Tablo 10'da 20 haber gerçekten gerçek haber iken, model bu haberleri sahte haber olarak yanlış tahmin etmiştir.
- TN (True Negative): Bir sınıflandırma modelinin doğru bir şekilde negatif sınıfı (örneğin, gerçek haber) tahmin ettiği durumları temsil eder. Tablo 10'da 150 haber gerçekten gerçek haber iken, model doğru bir şekilde bunları gerçek olarak tahmin etmiştir.

Tablo 10. Karmaşıklık matrisi temsili gösterimi

	TAHMİN	
	SAHTE	GERÇEK
GERÇEK	TP (100)	FP (20)
	FN (10)	TN (150)

Bunun yanısıra, bir modelin performansını gösteren çeşitli metrikler mevcuttur.

Doğruluk (Accuracy): Sınıflandırma sonucunun doğruluk değeri, gerçek değerlerin tahmin edilen değerlerle hangi oranda örtüştüğünü tanımlar. Doğruluk değerinin hesaplanma formülü aşağıda verilmiştir.

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

$$Hata\ Oranı = 1 - Doğruluk$$

Kesinlik (Precision): Karışıklık matrisinde pozitif olarak tahmin edilen örneklerin gerçekten pozitif olma oranını ölçer. Kesinlik değerinin hesaplanma formülü aşağıda verilmiştir.

$$Kesinlik = \frac{TP}{\mathrm{TP} + \mathrm{FP}}$$

Duyarlılık (Recall): Modelin doğru olarak tespit ettiği gerçek pozitif örneklerin oranını ifade eder. Kesinlik değerinin hesaplanma formülü aşağıda verilmiştir.

$$Duyarlılık = \frac{TP}{\mathrm{TP} + \mathrm{FN}}$$

F Ölçütü (F): Duyarlılık ve kesinlik ölçümlerinin harmonnik ortalamasını F ölçütü olarak tanımlanır. F Ölçütü değerinin hesaplanma formülü aşağıda verilmiştir.

$$F\ \text{Ölçütü} = (2)\frac{(Duyarlılık)(Kesinlik)}{\mathrm{Duyarlılık} + \mathrm{Kesinlik}}$$

Aşırı Öğrenme Durumu

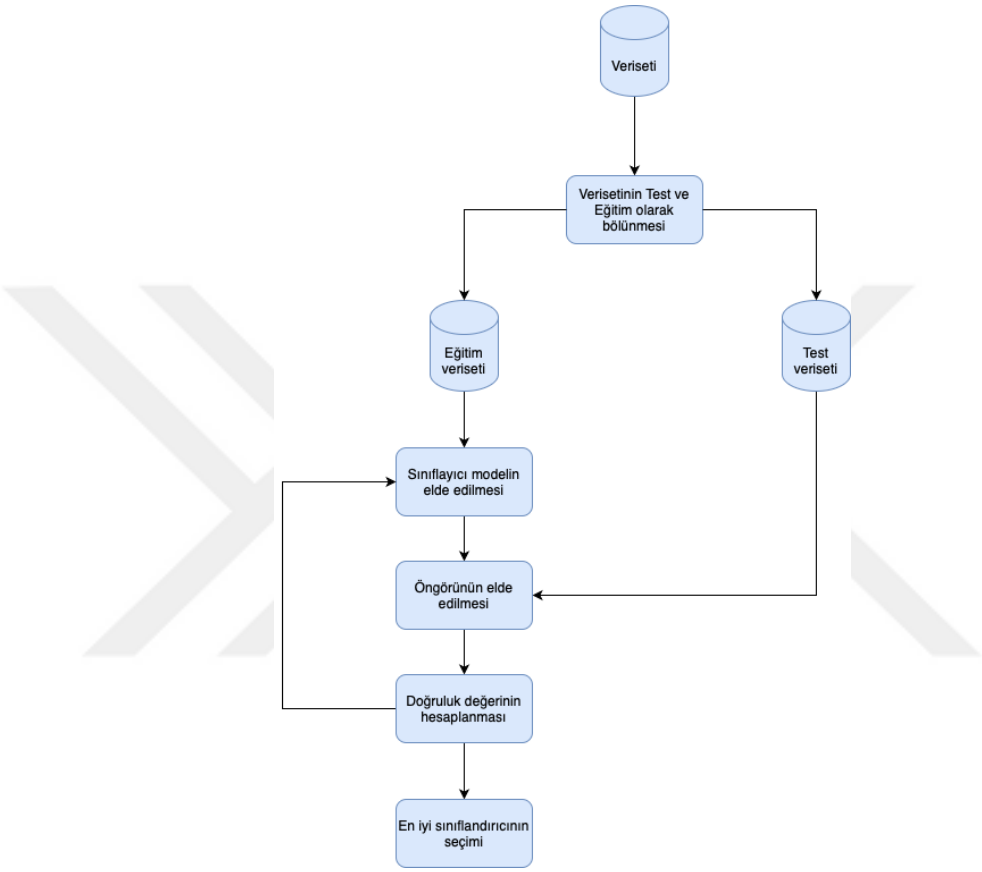
Sınıflandırma sürecinde, eğitim setinde kullanılan modelin doğruluk metriğinin yüksek olması amaçlanır. Ancak, eğitim setinde elde edilen doğruluk metriğinin test setinden elde edilen doğruluk oranından ciddi bir şekilde düşük olması modelin ezberlediğini gösterebilir. Bu durumun nedeni, eğitim veri setinin sınırlı olması ve sınıflandırma algoritmalarının karmaşıklığıdır.

Aşırı öğrenmeyi engellemek için alınabilecek adımlar şunlardır (Ying, 2019):

- Erken durdurma (early-stopping): Eğitim sürecinde, modelin mimarisini aşırı öğrenmeyi durdurmak için düzenlemek.
- Ağ indirgeme (network reduction): Eğitim sürecinde, veri setindeki gürültüleri çıkararak modelin karmaşıklığını azaltmak.

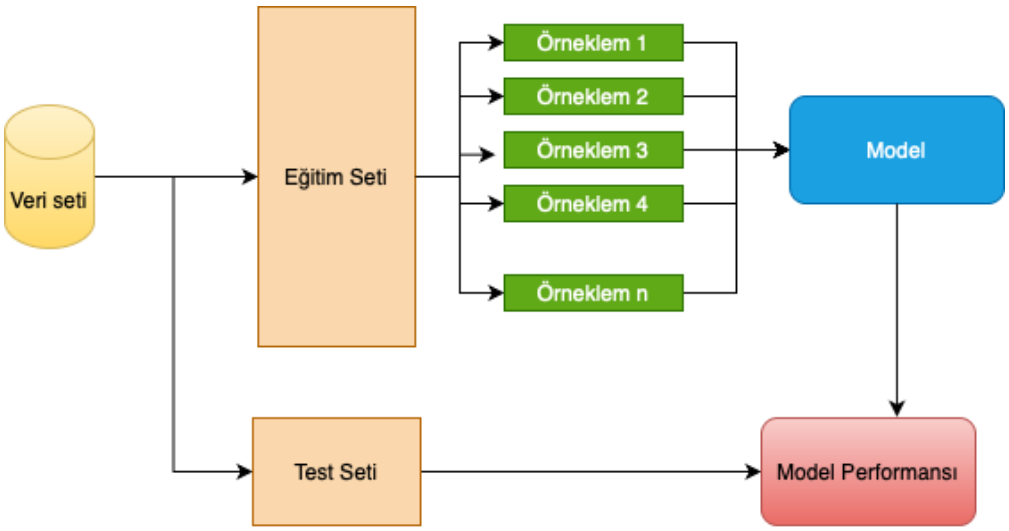
- Veri genişletme (data-expansion): Veri setinin farklı kaynaklardan beslenerek çeşitlendirilmesi.
- Düzenleme (regularization): Modelin performansını artırmak için daha kullanışlı özelliklerin çıkarımı yoluyla aşırı uyumu azaltmak.

Modelin doğrulama metriğinin hesaplanması süreci bir dizi işlemleri takip eder. Şekil 34''de doğrulama sürecine ait akış görselleştirilmiştir.



Şekil 34. Doğrulama süreci

Sınıflandırma sürecinde modelin gerçek doğruluğunu ölçmek için kullanılan bir diğer yöntem k-Katlı Çapraz Doğrulama'dır. Bu yöntemde veri seti k adet alt parçaya bölündükten sonra parçalardan biri test için geri kalan k-1 adeti eğitim için kullanılır. Bu süreç, k adet tekrarın sonunda modelin performansını hesaplamak için doğruluk değerlerinin ortalaması alınarak sonuçlanır (Anguita vd., 2012). Şekil 35'de k-Katlı Çapraz Doğrulama yöntemine ait örnek görsele yer verilmiştir.



Şekil 35. k-katlı çapraz doğrulama temsili gösterimi

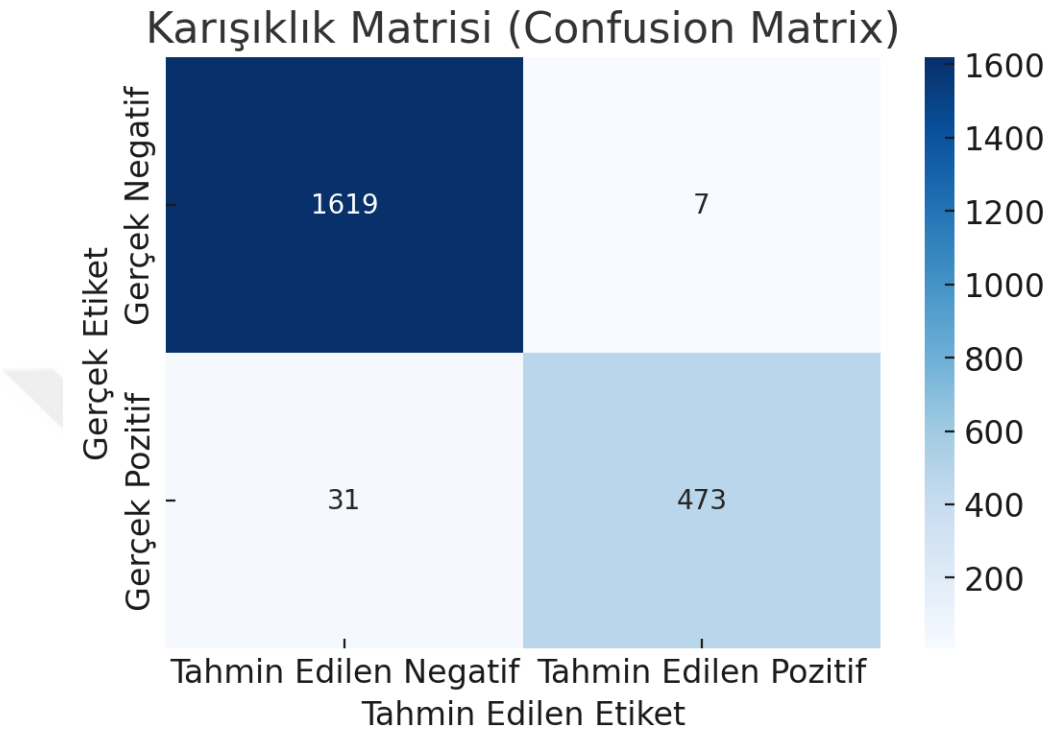
DÖRDÜNCÜ BÖLÜM

Bulgular

Son yıllarda teknolojinin hızlı gelişimi ile birlikte sosyal medya platformları toplumun büyük bir kesimi tarafından yaygın olarak kullanılmaktadır. Bu durum, kullanıcıların anonim ve doğrulanabilirliği kanıtlanmamış bir çok içeriğe maruz kalmasına sebep olmuştur. Bu tür paylaşımlar yalnızca toplumda kargaşaya neden olmakla kalmayıp aynı zamanda ekonomi, sağlık ve turizm gibi pek çok alanı da olumsuz etkilemektedir. Bunlar arasında başlıca örnek COVID-19 döneminde yayımlanan haber makaleleri ve ABD seçim kampanyalarıdır. Bir uzmanın manuel olarak herhangi bir haberin sahte olup olmadığını belirlemesi dikkatli ve ciddi emek gerektiren bir iştir. Bu çalışma, uzman tabanlı manuel kontrol yöntemlerine olan ihtiyacı azaltarak sınıflandırma algoritmaları aracılığıyla haberin doğruluğunu sağlamayı hedeflemiştir. Veri seti %80 eğitim ve %20 test olarak bölünmüş ve aşırı uyumu önlemek için birçok farklı yöntem uygulanmıştır. Model eğitim sürecinde TF-IDF, CountVectorizer ve Word2Vec ile sayısal vektörlere dönüşüm sağlanmış ve elde edilen değerlendirme metrikleri Tablo 12, 13 ve 14'de karşılaştırılmıştır. Yalnızca Teyit.org platformundan alınan eşit sayıda gerçek ve sahte haber üzerinde k-NN algoritması Tablo 5'de kullanılan en iyi parametreler ile eğitim yapılmış doğruluk değeri %62,35 elde edilmiştir. Bu sonuç, Teyit.org platformundaki haberler üzerinde sınırlı başarı sağladığını göstermektedir. Bunun yanısıra, hiperparametre optimizasyonu ile algoritmaların alabileceği en iyi parametreler seçilerek sınıflandırma performansının arttırılması sağlanmıştır. Tablo 11'de BERTurk modelinin performansı ve Tablo 12'de hiperparametre ayarlamasıyla elde edilen doğruluk değerleri karşılaştırıldığında transformer tabanlı kodlayıcı (encoder) içeren BERTurk modelinin %98,21 ile en iyi performansı verdiğini göstermiştir. Ayrıca, bulgular TF-IDF'nin CountVectorizer ve Word2Vec'e göre sınıflandırma sonuçları üzerinde daha iyi performans verdiğini ortaya çıkarmıştır. Tablo 12'de, diğer sınıflandırma algoritmalarına göre daha düşük performans gösteren Naive Bayes olmuştur. Tablo 13 ve Tablo 14'de modelin TF-IDF ve CountVectorizer yöntemleri kullanılarak elde edilen değerlendirme metrikleri (kesinlik, duyarlılık ve f1) sonuçları karşılaştırılmıştır.

Tablo 11. BERTurk modeline ait değerlendirme metrikleri

Model	Doğruluk (%)	Kesinlilik (%)	Duyarlılık (%)	F1 (%)
BERTurk	98.21	98.54	93.84	96.13



Şekil 36. BERTurk modeline ait karmaşıklık matrisi

Tablo 12. Modellerin doğruluk performansı

Model	Hiperparametre Optimizasyonu (%)			Doğruluk (%)	
	TF-IDF	CountVectorizer	Word2Vec	TF-IDF	CountVectorizer
SVM	**93.12**	92.86	88.84	92.40	91.65
Lojistik Regresyon	92.72	92.75	89.12	91.74	91.46
Rastgele Orman	92.61	92.72	89.12	92.12	91.56
k-NN	87.86	84.23	88.93	84.62	82.93
Karar Ağacı	87.74	88.15	85.27	85.74	86.30
Naive Bayes	74.39	75.33	87.43	74.39	75.33
LSTM	92.21	92.40	-	92.21	**92.40**

Makine öğrenimi algoritmaları arasında SVM modeli, TF-IDF ile en yüksek doğruluğu sağlayan algoritmadır. Doğruluk, TF-IDF için %92,40 dan hiperparametre ayarlaması sayesinde %93,12'ye yükselmiştir. Lojistik Regresyon modeli, TF-IDF ve

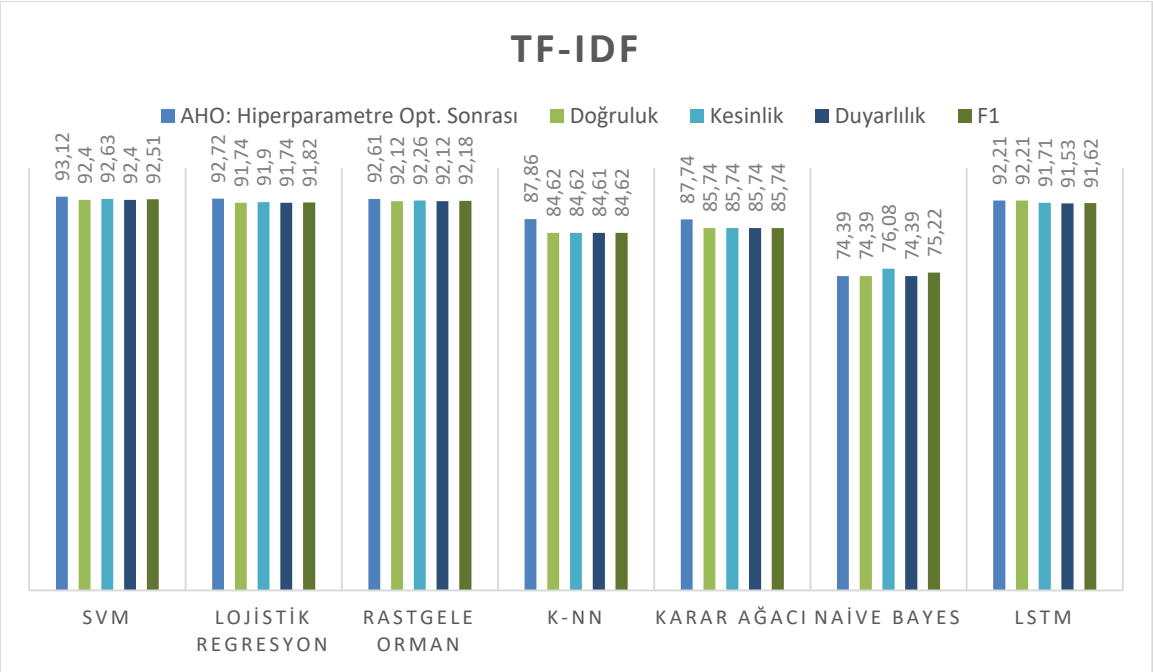
CountVectorizer arasında neredeyse eşit performans göstermiştir. Bu durum, modelin iki farklı kelime temsil yöntemiyle de tutarlı performans gösterdiğini ispatlamıştır. Rastgele Orman modelinin sonuçlarına bakıldığında Lojistik Regresyon modeline benzer oranda sınıflandırma işlemini yapabilmiştir. k-NN modeli, sınıflandırma işleminde hiperparametre ayarlaması ile nispeten küçük bir artış elde etmesine rağmen bu modelin doğruluk oranı daha düşük olarak ölçülmüştür. Naive Bayes modeli, TF-IDF ve Counvectorizer yöntemlerinin her ikisiyle de diğer modellere kıyasla en düşük performansı göstermiştir. Son olarak, derin öğrenme tabanlı LSTM modeli, %92,21 doğruluk oranına ulaştığı görülmektedir. Hiperparametre olmadan klasik makine öğrenimi algoritmaları ile yarışabilir fakat hiperparametre sonunda SVM modelinin gerisinde kalmıştır.

Bu araştırmadan elde edilen bulgular, BERTurk modelinin %98,21 doğruluk oranına sahip olduğunu ve Tablo 1 incelendiğinde (Bozuyla & ÖZÇİFT, 2022) tarafından geliştirilen bir başka transformers tabanlı BERTurk modelinin %98,5 dışında, diğer birçok modelden daha yüksek bir performans sergilediğini göstermektedir.

Tablo 13'de makine öğrenimi ve LSTM algoritmalarına ait eğitim sürecinde tf-idf yöntemi ile sayısal vektörlere dönüşümü sağlanmış ve elde edilen değerlendirme metrikleri verilmiştir. Şekil 37'de bu yöntemden elde edilen sonuçlar görselleştirilmiştir.

Tablo 13. Modelin TF-IDF yöntemi ile elde edilen doğruluk metrikleri

Model	**Kesinlik (%)**	**Duyarlılık (%)**	**F1 (%)**
SVM	92.63	92.40	92.51
Lojistik Regresyon	91.90	91.74	91.82
Rastgele Orman	92.26	92.12	92.18
k-NN	84.62	84.61	84.62
Karar Ağacı	85.74	85.74	85.74
Naive Bayes	76.08	74.39	75.22
LSTM	91.71	91.53	91.62

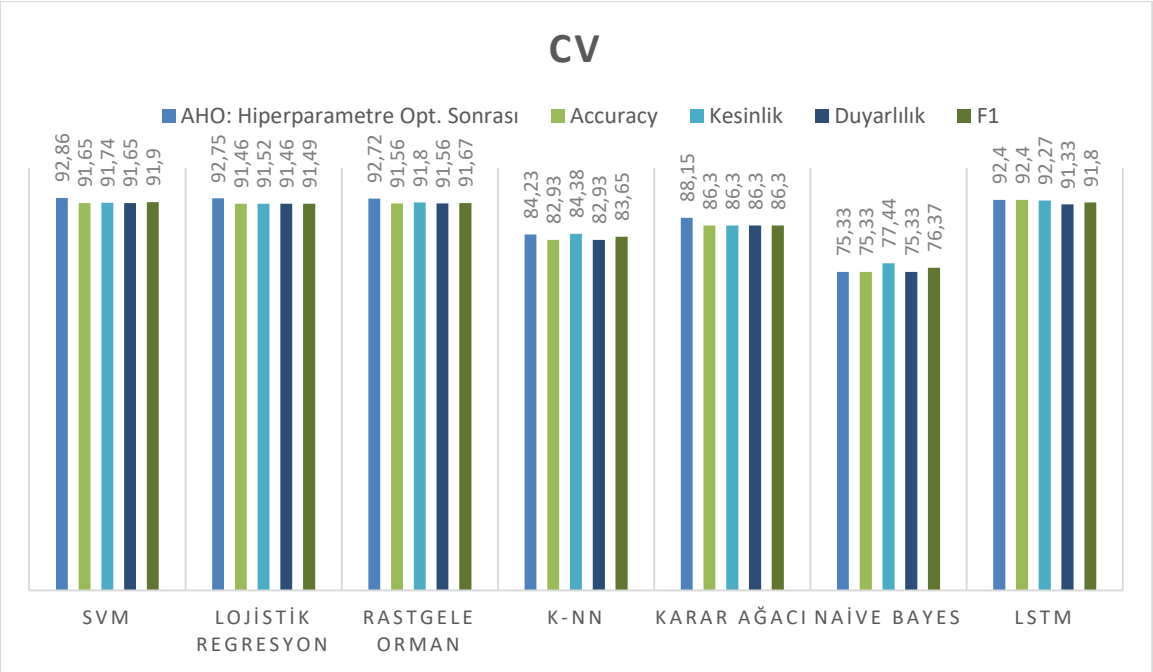


Şekil 37. TF-IDF yöntemi ile elde edilen değerlendirme metrikleri

Tablo 14'de makine öğrenimi ve LSTM algoritmalarına ait eğitim sürecinde countvectorizer yöntemi ile sayısal vektörlere dönüşümü sağlanmış ve elde edilen değerlendirme metrikleri verilmiştir. Şekil 38'de bu yöntemden elde edilen sonuçlar görselleştirilmiştir.

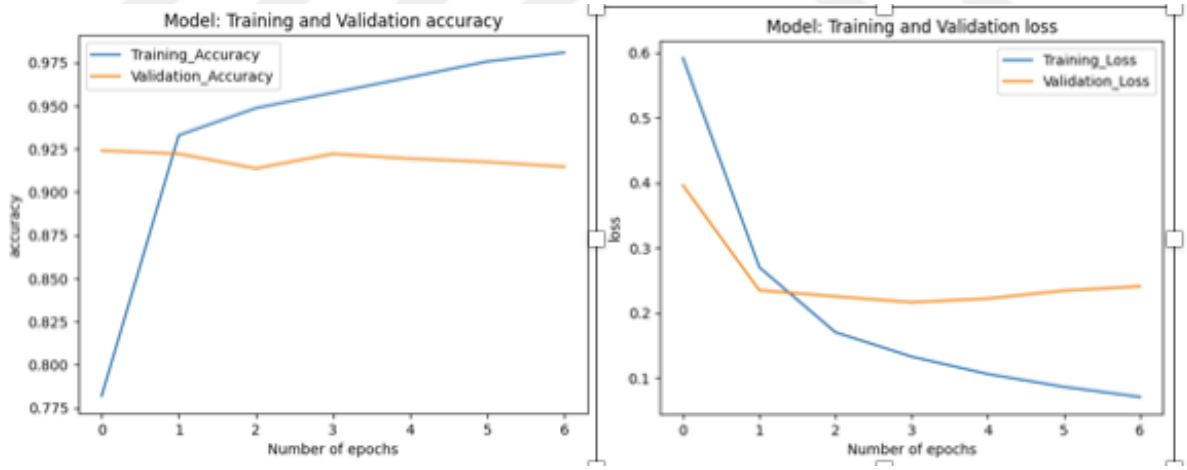
Tablo 14. Modelin CountVectorizer yöntemi ile doğruluk metrikleri

Model	Kesinlik (%)	Duyarlılık (%)	F1 (%)
SVM	91.74	91.65	91.90
Lojistik Regresyon	91.52	91.46	91.49
Rastgele Orman	91.80	91.56	91.67
k-NN	84.38	82.93	83.65
Karar Ağacı	86.30	86.30	86.30
Naive Bayes	77.44	75.33	76.37
LSTM	92.27	91.33	91.80



Şekil 38. CountVectorizer yöntemi ile elde edilen değerlendirme metrikleri

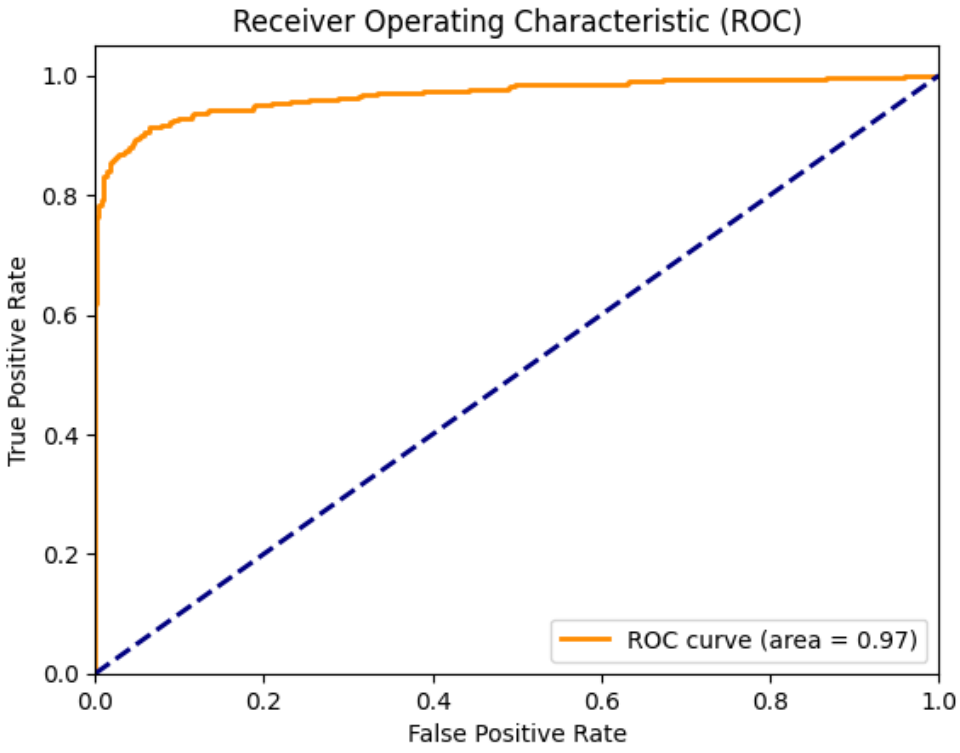
Hangi modelin sınıflandırmada daha iyi performans gösterdiğini daha detaylı inceleyebilmek için Tablo 15'deki makine öğrenimi algoritmalarına ait karmaşıklık matrisi verilmiştir.



Şekil 39. LSTM için eğitim ve doğrulama grafiği

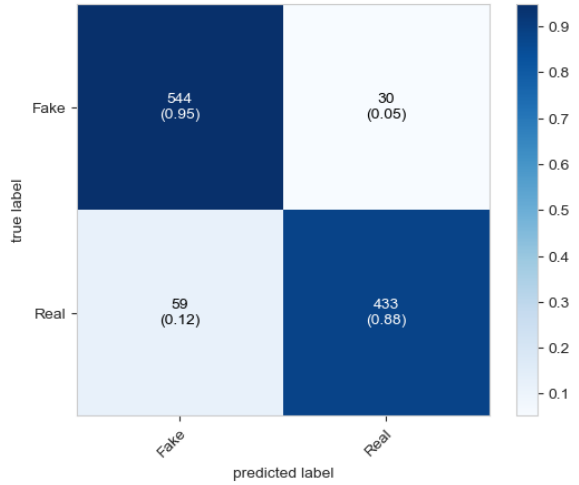
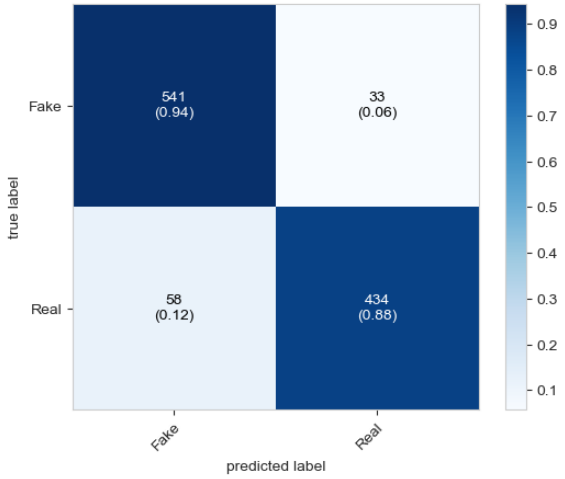
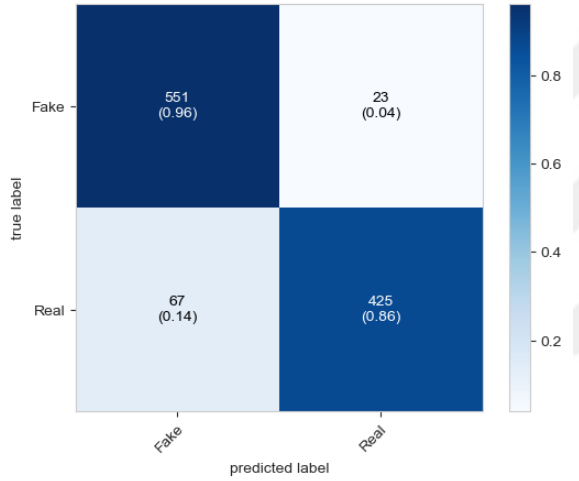
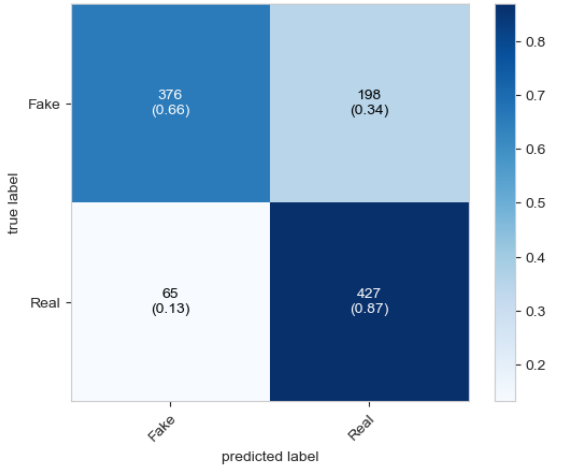
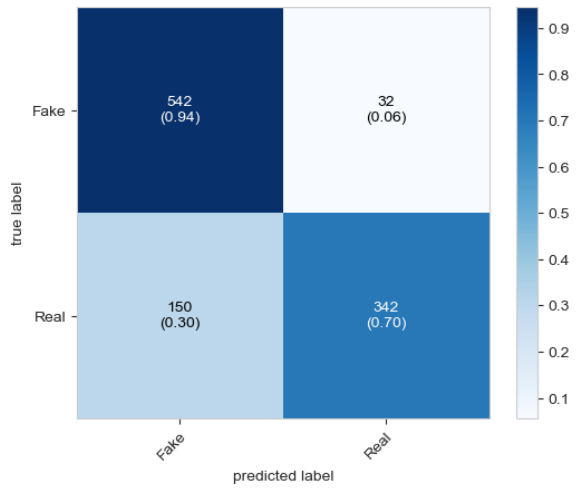
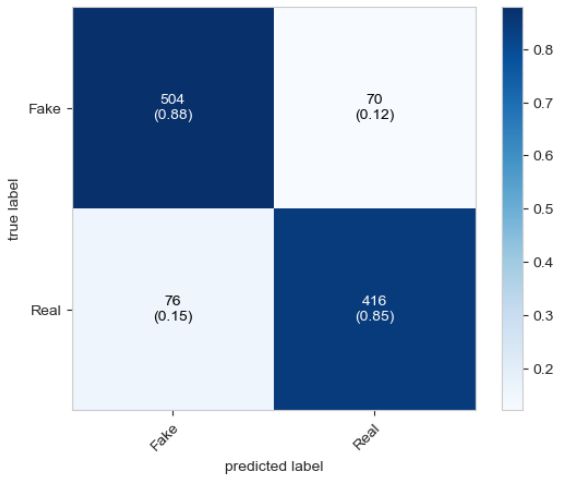
Şekil 39'da LSTM modelinin eğitim ve doğruluk sonuçları ile kayıp eğrisine ait görsel yer almaktadır. Eğitim ve doğrulama kaybı arasındaki fark, modelin eğitim verilerine uyum sağladığını gösterir. Eğitim ve doğrulama doğruluk eğrilerinin grafik üzerinde ilerleyen adımlarda bir noktada kesişmesi, modelin doğru şekilde genelleme yapıp yapmadığını gösterir. Eğitim ve doğrulama doğruluğu arasındaki farkın minimal olması, modelin aşırı

öğrenme (overfitting) problemini önlediğini gösterir. Ayrıca, Derin Öğrenme algoritmalarında modelin performansını değerlendirmek için ROC eğrileri, gerçek pozitif oranın (TPR) yanlış pozitif orana (FPR) karşı grafiğini çizerek ikili sınıflandırıcıların performansını ölçer (Kılınç, 2021). ROC eğrisinin altındaki alan (AUC) 0 ila 1 arasında değişen bir değere sahip olabilir. Şekil 40'da, bu çalışmada LSTM modeli, gerçek ve sahte haberleri ayırt etmede dikkate değer bir başarı elde ederek 0,97'lik bir ROC eğrisi göstermiştir.



Şekil 40. LSTM için ROC eğrisi

Tablo 15. Varsayılan parametreler ile elde edilen CountVectorizer karmaşıklık matrisi

	
SVM	Lojistik Regresyon
	
Rastgele Orman	Naive Bayes
	
k-NN	Karar Ağacı

BEŞİNCİ BÖLÜM

Tartışma ve Sonuç

Tezin hedefi, ana akım medya da yer alan haber makalelerinin doğrulanabilirliğinin teyitinde bir uzmanın çeşitli analiz yöntemlerine olan ihtiyacı azaltması amaçlanmaktadır. Bu çalışmada, gerçek dünyadan toplanan sınırlı veri seti üzerinde haberlerin doğruluğunu belirlemenin mümkün olabileceğini göstermiştir. Literatürde yapılan diğer çalışmalardan nitel farkı, ana akım medya kaynaklarından yeni bir veri seti oluşturulmuş olmasıdır. Bu veri seti, doğrulama süreçlerinde yüksek doğruluk oranlarına ulaşarak litaratüre önemli bir katkı sunmaktadır. Şekil 10'daki akış diyagramında, TRT ve Teyit haber kaynaklarından veri setleri birleştirilmiştir. Bu süreçte, kosinüs benzerliği metriği kullanılarak Teyit.org platformundaki haber makaleleri ile TRT haber makalelerinin benzerlik oranları ölçülmüştür. Benzerlik oranının 0.23 olarak hesaplanması, iki kaynağın birbiriyle uyumsuz olmadığını ortaya koymaktadır. Türkçe stopwords'ler kaldırılmış ve Zemberek kütüphanesi kullanılarak haber başlıkları ve özetleri üzerinde kök indirgeme işlemi yapılmıştır. Veri seti, test için %20 oranında ayrılmıştır. Bu işlemlerden sonra, modelin klasik makine öğrenimi, LSTM ve BERTurk algoritmaları ile eğitime hazır hale getirilmiştir. Elde edilen bulgular, modelin sınıflandırma işleminde başarılı sonuçları ortaya koyduğunu göstermiştir. Ayrıca, sınıflandırma algoritmalarında en iyi hiperparametreler seçilerek modellerin performansını daha iyi hale getirilmeye çalışılmıştır.

Bu çalışmada, BERTurk modeli ile %98,21 doğruluk oranı elde edilmiştir. Sonuçlar, literatürdeki benzer çalışmalarla karşılaştırıldığında kayda değer bir başarı olarak değerlendirilebilir. Örneğin, (Mertoğlu & Genç, 2020) çalışmasında ExtraTrees algoritmasıyla %96,81 doğruluk oranı elde ederken, (Taskin vd., 2022) SVM algoritmasıyla %90 doğruluk elde etmişlerdir. Ayrıca, (Bozuyla & ÖZÇİFT, 2022) çalışmasında BERTurk modeliyle %98,5 doğruluk oranına ulaşılmış, (YILDIRIM, 2022) ise XGBoost algoritmasıyla %97 doğruluk raporlamıştır. Bu sonuçlar, transformer tabanlı modellerin sınıflandırma problemlerinde daha yüksek performans gösterdiğini ortaya koymaktadır. Bununla birlikte, haberlerin teyitinde daha yüksek doğruluk oranlarına ulaşmak için gelecekte yalnızca kodlayıcı (encoder) içeren transformer tabanlı modellerin (DistilBERT, ROBERTA ve ModernBERT) karşılaştırılması üzerinde çalışılabilir.

Öneriler

Sonuç olarak, toplumda dilbilimsel analiz yöntemleriyle haberlerin gerçekliğinin analizinin ne kadar değerli olabileceğii vurgulanmaktadır. Gelecek çalışmalarda, araştırmada kullanılan veri setinin hem nitel hem de nicel anlamda genişletilmesi sağlanabilir. Bunun yanı sıra, haberlerin gerçek, sahte, yanıltıcı veya daha fazla alt kategorilere ayrılmasıyla birlikte daha detaylı ve kapsamlı sınıflandırma işlemlerine olanak tanınabilir. Ayrıca, yapay zeka hakkında sınırlı bilgiye sahip sıradan kullanıcılar, CBOT veya Dataiku gibi araçlarla bir akış diyagramı oluşturarak herhangi bir algoritmayı sınıflandırma işlemlerinde kullanabilir. Ayrıca, AutoML teknolojisi sayesinde, kendi alanlarına özel olarak uyarlanmış modeller geliştirerek bu süreci otomatikleştirme imkânına sahip olabilmektedir. Geliştirilen modeller, devlet veya kuruluşlarda dezenformasyonu önlemek için kullanılabilecektir.

KAYNAKÇA

Akın A & Akın M. (2007). Zemberek, an open source NLP framework for Turkic languages. *Structure, 10*(2007): 1-5.

Ahmad, I., Yousaf, M., Yousaf, S., & Ahmad, M. O. (2020). Fake News Detection Using Machine Learning Ensemble Methods. *Complexity*, *2020*, e8885861. https://doi.org/10.1155/2020/8885861

Ahmed, H., Traore, I., & Saad, S. (2017). Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. Içinde I. Traore, I. Woungang, & A. Awad (Ed.), *Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments* (ss. 127-138). Springer International Publishing. https://doi.org/10.1007/978-3-319-69155-8_9

Ajao, O., Bhowmik, D., & Zargari, S. (2018). Fake News Identification on Twitter with Hybrid CNN and RNN Models. *Proceedings of the 9th International Conference on Social Media and Society*, 226-230. https://doi.org/10.1145/3217804.3217917

Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of economic perspectives*, *31*(2), 211-236. https://www.aeaweb.org/articles?id=10.1257%2Fjep.31.2.211&fbclid=IwAR04My3

Alpaydın, E. (2011). *Yapay öğrenme*. Boğaziçi Üniversitesi Yayınları.

Anguita, D., Ghelardoni, L., Ghio, A., Oneto, L., & Ridella, S. (2012). The 'K' in K-fold Cross Validation. *Computational Intelligence*.

Arf, C. (1959). Makine düşünebilir mi ve nasıl düşünebilir. *Atatürk Üniversitesi-Üniversite Çalışmalarını Muhite Yayma ve Halk Eğitimi Yayınları Konferanslar Serisi*, *1*, 91-103.

Arragokula, S., & Ratnam, M. Y. (2016). Architectural Styles and the Design of Network-based Software Architectures. *International Journal of Ethics in Engineering & Management Education*. https://www.academia.edu/download/43012070/IJEEE-31-40.pdf

Aslam, N., Ullah Khan, I., Alotaibi, F. S., Aldaej, L. A., & Aldubaikil, A. K. (2021). Fake detect: A deep learning ensemble model for fake news detection. *complexity*, *2021*, 1-8. https://www.hindawi.com/journals/complexity/2021/5557784/

Atasoy, Y. (t.y.). *Veri Madenciliği Yöntemleri Ile Ankilozan Spondilit Hastaliğinda Radyografik Progresyona Etkili Faktörlerin Analizi* (Tez No. 394520) [Yüksek lisans tezi, İstanbul Üniversitesi-İstanbul]. Yükseköğretim Kurulu Ulusal Tez Merkezi.

Aydoğan, M., & Karcı, A. (2019). Kelime Temsil Yöntemleri ile Kelime Benzerliklerinin İncelenmesi. *Çukurova Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 34*(2), 181-196. https://doi.org/10.21605/cukurovaummfd.609119

Bozuyla, M., & ÖZÇİFT, A. (2022). Developing a fake news identification model with advanced deep languagetransformers for Turkish COVID-19 misinformation data. *Turkish Journal of Electrical Engineering and Computer Sciences*, *30*(3), 908-926. https://journals.tubitak.gov.tr/elektrik/vol30/iss3/27/

Chollet, F. (2019). Python ile Derin Öğrenme. *Baskı ed. Buzdağı Yayınevi, Ankara*.

Choudhury, N. (2014). World wide web and its journey from web 1.0 to web 4.0. *International Journal of Computer Science and Information Technologies*, *5*(6), 8096-8100. https://www.academia.edu/download/58974182/History_of_the_web20190420-62701-1kyua80.pdf

Çöltekin, Ç. (2014). A set of open source tools for Turkish natural language processing. İçinde N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, & S. Piperidis (Ed.), *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)* (ss. 1079-1086). European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2014/pdf/437_Paper.pdf

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (arXiv:1810.04805). arXiv. https://doi.org/10.48550/arXiv.1810.04805

Falah, Z. F., & Suryawan, F. (2022). Recommendation System to Propose Final Project Supervisors using Cosine Similarity Matrix. *Khazanah Informatika : Jurnal Ilmu Komputer Dan Informatika*, *8*(2), Article 2. https://doi.org/10.23917/khif.v8i2.16235

Güler, G., & GÜNDÜZ, S. (2023). Deep Learning Based Fake News Detection on Social Media. *International Journal of Information Security Science*, *12*(2), 1-21. https://dergipark.org.tr/en/pub/ijiss/article/1231423

Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines. *IEEE Intelligent Systems and their Applications*, *13*(4), 18-28. IEEE Intelligent Systems and their Applications. https://doi.org/10.1109/5254.708428

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Kaliyar, R. K., Goswami, A., & Narang, P. (2021a). EchoFakeD: Improving fake news detection in social media with an efficient deep neural network. *Neural Computing and Applications*, *33*(14), 8597-8613. https://doi.org/10.1007/s00521-020-05611-1

Kaliyar, R. K., Goswami, A., & Narang, P. (2021b). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia Tools and Applications*, *80*(8), 11765-11788. https://doi.org/10.1007/s11042-020-10183-2

Kaur, S., Kumar, P., & Kumaraguru, P. (2020). Automating fake news detection system using multi-level voting model. *Soft Computing*, *24*(12), 9049-9069. https://doi.org/10.1007/s00500-019-04436-y

Khanam, Z., Alwasel, B. N., Sirafi, H., & Rashid, M. (2021). Fake news detection using machine learning approaches. *IOP conference series: materials science and engineering*, *1099*(1), 012040. https://iopscience.iop.org/article/10.1088/1757-899X/1099/1/012040/meta

Kılınç, H. K. (2021). *Sahte haberlerin derin öğrenme ve makine öğrenmesi ile tespiti* (Tez No. 663975)[Yüksek lisans tezi, Hasan Kalyoncu Üniversitesi-Gaziantep]. Yükseköğretim Kurulu Ulusal Tez Merkezi.

Koru, G. K., & Uluyol, Ç. (2024). Detection of Turkish Fake News from Tweets with BERT Models. *IEEE Access*. https://ieeexplore.ieee.org/abstract/document/10399808/

Kucharski, A. (2016). Study epidemiology of fake news. *Nature*, *540*(7634), 525-525. https://www.nature.com/articles/540525a

Küçük, D., & Arici, N. (2018). Doğal Dil İşlemede Derin Öğrenme Uygulamalari Üzerine Bir Literatür Çalişmasi. *Uluslararası Yönetim Bilişim Sistemleri ve Bilgisayar Bilimleri Dergisi*, *2*(2), 76-86. https://dergipark.org.tr/en/pub/uybisbbd/issue/41787/443574

LaValley, M. P. (2008). Logistic Regression. *Circulation*, *117*(18), 2395-2399. https://doi.org/10.1161/CIRCULATIONAHA.106.682658

Lazer, D. M. J., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild, D., Schudson, M., Sloman, S. A., Sunstein, C. R., Thorson, E. A., Watts, D. J., & Zittrain, J. L. (2018). The science of fake news. *Science*, *359*(6380), 1094-1096. https://doi.org/10.1126/science.aao2998

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436-444. https://www.nature.com/articles/nature14539

Liu, Y., Wang, Y., & Zhang, J. (2012). New Machine Learning Algorithm: Random Forest. Içinde B. Liu, M. Ma, & J. Chang (Ed.), *Information Computing and Applications* (ss. 246-252). Springer. https://doi.org/10.1007/978-3-642-34062-8_32

Mertoğlu, U., & Genç, B. (2020). Automated fake news detection in the age of digital libraries. *Information Technology and Libraries*, *39*(4). https://ejournals.bc.edu/index.php/ital/article/view/12483

Meyers, M., Weiss, G., & Spanakis, G. (2020). Fake News Detection on Twitter Using Propagation Structures. Içinde M. van Duijn, M. Preuss, V. Spaiser, F. Takes, & S. Verberne (Ed.), *Disinformation in Open Online Media* (ss. 138-158). Springer International Publishing. https://doi.org/10.1007/978-3-030-61841-4_10

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space* (arXiv:1301.3781). arXiv. https://doi.org/10.48550/arXiv.1301.3781

Monti, F., Frasca, F., Eynard, D., Mannion, D., & Bronstein, M. M. (2019). *Fake News Detection on Social Media using Geometric Deep Learning* (arXiv:1902.06673). arXiv. http://arxiv.org/abs/1902.06673

Mumbaikar, S., & Padiya, P. (2013). Web services based on soap and rest principles. *International Journal of Scientific and Research Publications*, *3*(5), 1-4. https://www.academia.edu/download/88306479/ijsrp-p17115.pdf

Özkan, Y. (2020). *Veri madenciliği yöntemleri*. Papatya Yayıncılık Eğitim.

Quinlan, J. R. (1996). Learning decision tree classifiers. *ACM Computing Surveys*, *28*(1), 71-72. https://doi.org/10.1145/234313.234346

Rahutomo, F., Kitasuka, T., & Aritsugi, M. (2012). *Semantic Cosine Similarity*.

Rigatti, S. J. (2017). Random Forest. *Journal of Insurance Medicine*, *47*(1), 31-39. https://doi.org/10.17849/insm-47-01-31-39.1

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, *19*(1), 22-36.

Siahaan, A. P. U., Aryza, S., Hariyanto, E., Rusiadi, Lubis, A. H., Ikhwan, A., & Kan, P. L. E. (2018). Combination of levenshtein distance and rabin-karp to improve the accuracy of document equivalence level. *International Journal of Engineering & Technology*, *7*(2.27), Article 2.27. https://doi.org/10.14419/ijet.v7i2.27.12084

Taskin, S. G., Kucuksille, E. U., & Topal, K. (2022). Detection of Turkish Fake News in Twitter with Machine Learning Algorithms. *Arabian Journal for Science and Engineering*, *47*(2), 2359-2379. https://doi.org/10.1007/s13369-021-06223-0

Thota, A., Tilak, P., Ahluwalia, S., & Lohia, N. (2018). Fake news detection: A deep learning approach. *SMU Data Science Review*, *1*(3), 10. https://scholar.smu.edu/datasciencereview/vol1/iss3/10/

Turing, A. (1950). Machinery and intelligence. *Mind: A Quarterly Review of Psychology and Philosophy*, *59*(236), 433-460.

https://www.robertspahr.com/teaching/hnm/turing_computing_machinery_and_intelligence.pdf

Ünver, A. (t.y.). *Emerging technologies and automated fact-checking: tools, techniques and algorithms*.

Waswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *NIPS*. https://www.aiotlab.org/teaching/dl_app/slides/6_3_attention_n_bert.pdf

Yamanan, E. (2016). Türkçenin Güncel Söz Varliği. *Millî Eğitim Dergisi*, *45*(210), 85-91. https://dergipark.org.tr/en/pub/milliegitim/issue/36140/406013

Yildirim, E. (2022). *Hizlandirilmiş Makine Öğrenmesi Algoritmalari Ile Türkçe Sahte Haber Tespiti* (Tez No. 755464) [Yüksek lisans tezi, Karabük Üniversitesi-Karabük]. Yükseköğretim Kurulu Ulusal Tez Merkezi.

Ying, X. (2019). An overview of overfitting and its solutions. *Journal of physics: Conference series*, *1168*, 022022. https://iopscience.iop.org/article/10.1088/1742-6596/1168/2/022022/meta

ÖZ GEÇMİŞ

Kişisel Bilgiler		
Adı Soyadı	**:**	İsa KULAKSIZ
Doğum tarihi	**:**	
Doğum Yeri	**:**	
Uyruğu	**:**	
Adres	**:**	
Tel	**:**	
E-mail	**:**	
Eğitim		
Lise	**:**	
Lisans	**:**	
Yüksek lisans	**:**	
Doktora	**:**	
Yabancı Dil Bilgisi		
	:	
Üye Olunan Mesleki Kuruluşlar		
Tezden Üretilmiş Yayınlar		