KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

TÜRKİYE'DEKİ HAVAYOLU FİRMALARIYLA İLGİLİ SOSYAL MEDYA YORUMLARININ MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE SINIFLANDIRILMASI

HATİCE ELİF EKİM

KOCAELİ 2021

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

TÜRKİYE'DEKİ HAVAYOLU FİRMALARIYLA İLGİLİ SOSYAL MEDYA YORUMLARININ MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE SINIFLANDIRILMASI

HATİCE ELİF EKİM

Dr.Öğr.Üyesi Alpaslan Burak İNNER **Danışman, Kocaeli Üniv.**	.....
Prof.Dr. Kerem KÜÇÜK **Jüri Üyesi, Kocaeli Üniv.**	.....
Dr.Öğr.Üyesi Ersin Kaya **Jüri Üyesi, Konya Teknik Üniv.**	.....

Tezin Savunulduğu Tarih: 28.04.2021

ÖNSÖZ VE TEŞEKKÜR

Bu tez çalışması, havayolu firmalarının sosyal medya departmanlarının iş yükünü ve hata payını azaltmak sosyal medyadan gelen kullanıcı yorumlarının hızlı bir şekilde ilgili departmana makine öğrenmesi algoritmalarıyla iletilmesini sağlamak amacıyla gerçekleştirilmiştir.

Tez çalışmasında, desteğini esirgemeyen, kıymetli tecrübeleri ile tez çalışmalarıma yön veren değerli danışmanım Dr. Öğr. Üyesi Alpaslan Burak İNNER'e ve diğer jüri üyesi hocalarıma teşekkür eder, saygılarımı sunarım.

Hayatım boyunca bana güç veren, sıkıntılarımı, mutluluklarımı benimle birlikte yaşayan, özellikle bu zorlu süreçte sabırla destekleyerek yoluma devam etmemi sağlayan anneme teşekkürlerimi sunarım.

Nisan – 2021 Hatice Elif EKİM

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

TABLOLAR DİZİNİ

SİMGELER VE KISALTMALAR DİZİNİ

b	:	Sabit değer
xi	:	Destek vektör değeri
yi	:	Destek vektörü sınıf etiketi
w	:	Hiper düzlemin ağırlık vektörü

Kısaltmalar

CNN	:	Convolutional Neural Network (Evrişimli Sinir Ağları)
IDF	:	Inverse Document Frequency (Ters Doküman Frekansı)
LSTM	:	Long Short-Term Memory (Uzun Kısa Süreli Bellek)
LR	:	Logistic Regression (Lojistik Regresyon)
MNB	:	Multinomial Naive Bayes (Multinomial Naive Bayes)
ME	:	Maximum Entropy (Maksimum Entropi)
RF	:	Random Forest (Rastgele Orman)
SVM	:	Support Vector Machine (Destek Vektör Makineleri)
TF	:	Term Frequency (Terim Frekansı)
ANOVA	:	Analysis of Variance (Varyans Analizi)

TÜRKİYE'DEKİ HAVAYOLU FİRMALARIYLA İLGİLİ SOSYAL MEDYA YORUMLARININ MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE SINIFLANDIRILMASI

ÖZET

Son yıllarda teknolojinin gelişmesi ve internet kullanım oranının artmasıyla mikroblog adı verilen sosyal medya platformları, milyarlarca insanın çeşitli konularda görüşlerini, fikirlerini, duygularını ve şikayetlerini paylaştığı platformlar haline gelmiştir. Sosyal medya kullanımının dünya genelinde yaygınlaşması ile şikayet, öneri, talep bildirimleri sadece web siteleri, telefon veya e-mail aracılığıyla değil, sosyal medya özellikle Twitter aracılığıyla da paylaşılmaktadır. Firmaların, Twitter aracılığı ile gerçekleştirdikleri hızlı geri dönüşler ve alınan aksiyonlar müşteri memnuniyeti açısından büyük önem arz etmektedir. Özellikle müşteriyle direkt iletişim halinde olması gereken hizmet sektörü, memnuniyeti artırmak ve daha iyi hizmet sunmak amacıyla müşterileriyle Twitter üzerinden iletişime geçmektedir.

Bu tez çalışmasında, Türk havacılık sektöründe yer alan firmaların, misafirleriyle Twitter üzerinden gerçekleştirdikleri iletişimi hızlandırmak ve konunun şirket içinde ilgili departmana ulaşmasını kolaylaştırmak, harcanan insan eforunu ve hata payını en aza indirmek amacıyla metin madenciliği çalışması gerçekleştirilmiştir. Türk havayolu firmaları hakkında yapılan 14.406 adet Türkçe Twitter paylaşımı üzerinde, doğal dil işleme ve metin madenciliği çalışmaları gerçekleştirilerek, havayolu firmalarının web sitelerinde yer alan sık sorulan sorular başlıklarından belirlenen konu başlıklarına göre ilgili tweet'in hangi başlık altında yer aldığı, Naive Bayes, Destek Vektör Makineleri, Lojistik Regresyon, Karar Ağaçları, Rastgele Orman geleneksel makine öğrenmesi algoritmaları, Adaboost, XGBoost, Bagging topluluk öğrenme yöntemleri ile CNN ve LSTM derin öğrenme algoritmaları kullanılarak sınıflandırılmıştır.

Sınıflandırma başarılarını ölçmek için geleneksel yöntemler ve topluluk yöntemlerinde farklı öznitelik seçim yöntemleri ve öznitelik sayıları parametre olarak belirlenirken derin öğrenme yöntemlerinde farklı aktivasyon fonksiyonları ile deneyler gerçekleştirilmiştir.

Çalışma sonucunda, geleneksel yöntemlerden SVM, topluluk öğrenmesi yöntemlerinden XGBoost ve derin öğrenme yöntemleri ile yaklaşık %77 doğruluk oranı elde edilmiştir.

Anahtar Kelimeler: Çok Sınıflı Metin Sınıflandırma, Havayolu, Makine Öğrenmesi, Metin Madenciliği, Twitter.

CLASSIFICATION OF SOCIAL MEDIA COMMENTS ABOUT AIRLINE COMPANIES IN TURKEY BY MACHINE LEARNING METHODS

ABSTRACT

The development of technology in recent years and increase internet usage ratio called the microblog Twitter, Facebook-like social media platforms, billions of views on various issues of people, ideas, has become a platform shared their feelings and complaints. With the widespread use of social media around the world, complaints, suggestions, and request notifications are shared not only through websites, phone, or e-mail, but also via social media, especially Twitter. Fast feedbacks and actions taken by companies via Twitter are of great importance for customer satisfaction. In particular, the service sector, which should be in direct contact with the customer, communicates with its customers via Twitter in order to increase satisfaction, and provide better service.

In this thesis, a text mining study was carried out in order to accelerate the communication of companies in the Turkish aviation industry with their guests via Twitter, to facilitate the issue to reach the relevant department within the company, and to minimize the human effort and margin of error. Natural language processing and text mining studies were carried out on 14,406 Turkish Twitter posts about Turkish airline companies, according to the topics determined from the frequently asked questions titles on the websites of the airline companies, under which heading the relevant tweet is located, Naive Bayes, Support Vector The machines are classified using Logistic Regression, Decision Trees, Random Forest traditional machine learning algorithms, Adaboost, XGBoost, Bagging ensemble learning methods and CNN and LSTM deep learning algorithms.

In order to measure the classification success, different feature selection methods and feature numbers were determined as parameters in traditional methods and ensemble methods, while experiments were carried out with different activation functions in deep learning methods.

As a result of the study, approximately 77% accuracy rate was obtained with SVM from traditional methods, XGBoost from ensemble learning methods and deep learning methods.

Keywords: Multi-Class Text Classification, Airline, Machine Learning, Text Mining, Twitter.

GİRİŞ

Günümüzde internet kullanımının yaygınlaşması ile tüm dünyada milyarlarca insan düşüncelerini, şikayetlerini ve yorumlarını sosyal medya platformları aracılığı ile ifade etmektedir. Farklı sosyal medya kanalları üzerinden her yeni yıl geçmiş yıllara göre çok daha hızlı, büyük miktarda ve farklı tiplerde veri üretilmektedir. Sosyal medya kanalları aracılığı ile video, görsel, metin, ses vb… farklı tiplerde içerikler üretilmesine rağmen içeriklerin büyük bir kısmını, insanların düşüncelerini yazıya dökmesiyle oluşan metinler oluşturmaktadır. İnsanların düşüncelerini yazıya dökmek için kullandıkları sosyal medya platformalarından en yaygın kullanıma sahip olanlardan birisi de Twitter'dır.

Firmalar, kişilerin sosyal medya etkileşimlerinin artmasını fırsat bilerek müşteri memnuniyetini sosyal medya kanallarından bilhassa da kullanıcı etkileşimi hızlı olan Twitter aracılığı ile sağlamayı hedeflemektedir. Özellikle hizmet sektöründe yer alan firmalar için Twitter kullanıcılarının paylaşımlarına verilen hızlı cevapların ve alınan hızlı aksiyonların müşteri memnuniyetini olumlu yönde etkilediği ve bu olumlu etkinin kişiden kişiye yayıldığı görülmektedir.

Firmalar, Twitter'dan gelen kullanıcı yorumlarını, mesajlarını veya paylaşımlarını şirket içi ilgili departmana ileterek müşterilerine daha doğru ve hızlı çözümler sunmaya çalışmaktadır. Fakat Twitter üzerinden yapılan kullanıcı paylaşımlarının incelenmesi ve ilgili departmana iletilmesi için arka tarafta birçok personel çalışmakta bu da hem iş gücü hem de zaman kaybına dolayısı ile şirket için maddi bir bedele mal olmaktadır. Hatta Twitter paylaşımlarının yanlış departmana iletilmesi ve doğru departmana geri atama süreçleri harcanan eforu artırmaktadır.

Firma, müşteri arasındaki iletişimi hızlı bir şekilde gerçekleştirmek, aradaki insan eforunu ve hata payını en aza indirmek makine öğrenmesi teknikleri ile mümkündür.

Bu tez çalışmasında, aradaki insan eforunu en aza indirmek, süreci hızlandırmak ve hata payını azaltmak amacıyla Türk havacılık sektöründe faaliyet gösteren firmalar ile ilgili Twitter paylaşımları üzerinden çok sınıflı metin sınıflandırma çalışması

gerçekleştirilmiştir. Türk hava yolu firmalarının web sitelerinde yer alan sık sorulan sorular konu başlıkları analiz edilmiş ve ortak konular belirlenerek “Kampanyalar”, “Bagaj/ Eşya/ Diğer Taşıyabileceklerim”, “Bilet İşlemlerim”,“Sefer İptalleri”, “Sefer Değişiklik ve Gecikmeleri”, “İade İşlemleri”, “Uçuş ve Uçak İçi”, “Seyahat Planlama” olmak üzere her bir konu farklı bir departman olacak şekile 10 tane ana başlık çıkarılmıştır. Bu başlıklar dışında kalan konular için de “Diğer İşlemler” başlığı belirlenmiştir. Türk havayolu firmaları ile ilgili Türkçe Twitter paylaşımlarından oluşan veri seti belirlenen başlıklar ile etiketlenmiştir. Fakat Twitter verileri kısaltmalar, hatalı yazılmış kelimeler ve günlük konuşma dilinde tercih edilmeyen sosyal medyaya özgü sözcüklerden oluştuğu için bu veriler üzerinde herhangi bir işlem yapmadan sınıflandırmaya çalışmak oldukça zordur. Bu nedenle veriler üzerinde çalışmaya başlamadan önce yapay zeka ve dilbilimin alt kategorisi olan doğal dil işleme yöntemleri ile filtrelenmesi, işlenmesi ve temizlenmesi gerekmektedir. Bu bağlamda ilk aşama ham verilerin temizlenmesi ve işlenebilir hale getirilmesi olmuştur. Daha sonra etiketli veriler üzerinde MNB, Doğrusal SVM, MNB, LR, RF ve DT geleneksel makine öğrenmesi algoritmaları, Adaboost, XGBoost, Bagging topluluk öğrenmesi yöntemleri, CNN ve LSTM derin öğrenme yöntemleri ile çok sınıflı metin sınıflandırma çalışması yapılmıştır.

Bölüm 1’de problem tanımı, sınıflandırma problemi, Makine Öğrenmesi algoritmaları, öznitelik seçimi ve performans metrikleri ile ilgili genel bilgi verilmektedir. Bölüm 2’de çok sınıflı metin sınıflandırma ve havayolu firmaları ile ilgili Twitter yorumları üzerinde gerçekleştirilen sınıflandırma çalışmaları ile ilgili daha önce gerçekleştirilmiş çalışmaların bulunduğu literatür taraması yer almaktadır. Bölüm 3’de çalışmada kullanılan veri setleri ve uygulanan yöntemler detaylı olarak açıklanmıştır. Bölüm 4’de çalışmanın deney sonuçları verilmiştir. Bölüm 5’te elde edilen sonuçlar değerlendirilmiş ve gelecek çalışmalar için öneriler sunulmuştur.

1. GENEL BİLGİLER

1.1. Problem Tanımı

Birçok kurumsal firma, sosyal medya paylaşımlarını yönetmek, sosyal medya müşteri problemlerini hızlı bir şekilde çözmek, çözülemeyen konuları ilgili birime iletmek, müşteri memnuniyetini sağlamak vb. görevleri yerine getirmek için sosyal medya ekibi çalıştırmaktadır. Özellikle hizmet sektöründe yer alan, insanla birebir etkileşim içinde olan firmalar için sosyal medya kanalları ile müşteri etkileşimi büyük önem arz etmektedir. Özellikle kurumsal firmalar da sosyal medya hesaplanın iyi yönetilmesi, müşteri iletişiminin en iyi ve kolay şekilde sağlanması konusuna önem vermekte ve bu bağlamda sosyal medya departmanlarına büyük yatırımlar yapmaktadır. Sosyal medya departmanı vasıtasıyla manuel yürütülen sistem, firmalar için zaman, efor ve para kaybına sebep olmaktadır.

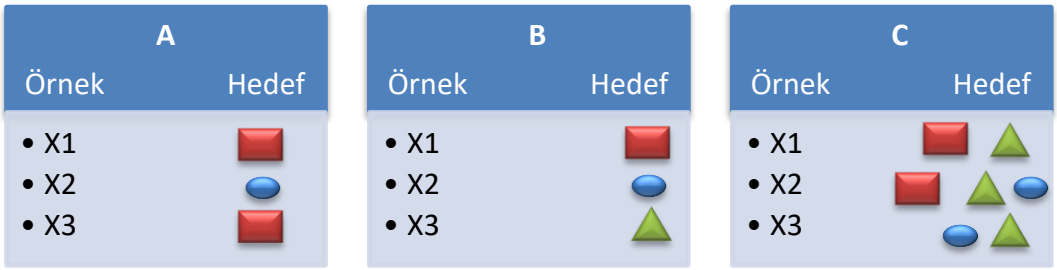
Sosyal medya paylaşımlarının insan gücüne gerek kalmadan ilgili departmana ulaşması makine öğrenmesi teknikleri ile mümkündür. Günümüzde makine öğrenmesi teknikleri hem hızlı olması hem de yüksek doğruluk oranları elde edilmesi sebebiyle birçok alanda sınıflandırma amacıyla kullanılmaktadır.

Bu tez çalışmasında, geleneksel makine öğrenmesi ve derin öğrenme teknikleriyle Türk havacılık sektöründe faaliyet gösteren firmalar ile ilgili Türkçe Twitter paylaşımlarının insan gücüne gerek kalmadan sınıflandırılarak firma içinde ilgili departmana hızlı ve doğru bir şekilde ulaşmasını sağlamak amaçlanmıştır.

1.2. Sınıflandırma Problemi

Geçmişten günümüze, tıp (hastalık teşhisi), biyoloji (canlı türleri sınıflandırması), kimya (ilaç etkilerinin belirlenmesi), sosyal medya (paylaşımların olumlu/olumsuz sınıflandırılması), üretim (kusurlu ürün tespiti) gibi farklı alanlarda, probleme veya kullanılacak olan veriye göre sınıflandırma probleminin ele alındığı birçok çalışma gerçekleştirilmiştir. Gerçekleştirilen çalışmalar sonucunda sınıflandırma probleminin çözümü makine öğrenimi için önemli çalışma alanlarından biri haline gelmiştir [1].

Sınıflandırma süreci, Şekil 1.1'de gösterildiği gibi ikili sınıflandırma (binary / binominal classification), çok sınıflı sınıflandırma (multi-class / multinomial classification) ve çoklu etiketli sınıflandırma (multi label classification) olmak üzere üç grupta ele alınabilir [2].



Şekil 1. 1. A) İkili sınıflandırma B) çok sınıflı sınıflandırma C) çoklu etiketli sınıflandırma

1.2.1. İkili sınıflandırma (binary / binominal classification)

İkili sınıflandırma, oldukça sık karşılaşılan gözetimli sınıflandırma yöntemidir. Y sınıf etiketleri listesinde yalnız iki değer bulunur. Veri seti elemanları, bir sınıflandırma kuralı ile belirlenen iki sınıftan en uygun olanına atanır. İkili sınıflandırmada her bir örneğe tek etiket atanmaktadır.

1.2.2. Çok sınıflı sınıflandırma (multi-class / multinomial classification)

Çok sınıflı (Multi-class) sınıflandırma, etiket listesi ikiden daha fazla değere sahip olan gözetimli sınıflandırma yöntemidir. Her bir örnek Y sınıf etiketleri listesinden birbirinden bağımsız yalnız bir sınıfa dahil olabilir. Çok sınıflı sınıflandırmada her bir örneğe tek etiket atanmaktadır.

1.2.3. Çoklu etiketli sınıflandırma (multi label classification)

Çoklu etiketli sınıflandırmada, etiket listesi çok sınıflı sınıflandırmada olduğu gibi ikiden fazla değere sahip olabilir fakat her bir örnek Y sınıf etiketleri listesinde bir ya da birden çok sınıf etiketine atanabilir. Çoklu etiketli sınıflandırmada her bir örneğe birden çok etiket atanmaktadır.

1.3. Makine Öğrenmesi

Son yıllarda, sosyal medya kullanımının artmasıyla saatte terabaytlar seviyesine ulaşan veri miktarı metin sınıflandırma çalışmalarına verilen önemi ve artırmıştır. Metin sınıflandırma yöntemleri olarak kullanılan makine öğrenmesi teknikleri, günümüzde tercih edilen popüler alanlardan biri haline gelmiştir.

Makine öğrenmesi terimi ilk kez "Some Studies in Machine Learning Using the Game of Checkers" adlı makalede Arthur Lee Samuel (1959) tarafından kullanılmıştır. Samuel, makine öğrenmesi terimini, tam olarak programlama yapmaksızın bilgisayarlara kendi kendine öğrenme yeteneği kazandıran çalışma alanı olarak tanımlamıştır [3]. Samuel'in tanımlaması üzerine, makine öğrenmesi, araştırmacılar ve önde gelen kuruluşlar tarafından da farklı cümlelerle ifade edilmiştir. Stanford üniversitesi 2019 yılında makine öğrenmesi tanımını şu şekilde yapmıştır; bilgisayarları, açıkça programlamadan harekete geçirme bilimi.

Makine öğrenmesi, uydu görüntülerinin haritalanması [4], sesten anlam çıkarma, metin sınıflandırma, otonom robotlar, sürücüsüz araçlar, yüz tanıma, e-posta filtreleme, optik karakter tanıma, ürün önerme ve hastalıkların belirlenmesi gibi birçok konularında önemli hesaplama alanlarından biri olmuştur.

İnternetteki anlık veri artışı üzerine araştırmacılar da farklı platformlardan elde edilen veriler üzerinde gerçekleştirdikleri makine öğrenimi çalışmalarına ağırlık vermişlerdir.

El Rahman ve ark. belirledikleri markalar (McDonalds ve KFC) ile ilgili Twitter verilerini toplayarak denetimli makine öğrenmesi yöntemleri ile duygu analizi çalışması gerçekleştirmişlerdir [5].

Çiftçi ve Apaydın, Hepsiburada ve Beyazperde.com'dan elde ettikleri kullanıcı yorumlarından oluşan veri kümesi üzerinde makine öğrenmesi algoritmalarının başarı oranlarını karşılaştırmışlardır [6].

Makine öğrenmesi modelleri; denetimli öğrenme (supervised learning), denetimsiz öğrenme (unsupervised learning) ve yarı denetimli öğrenme (semi-supervised learning) olarak üç grupta incelenmektedir. Ayrıca hem denetimli hem de denetimsiz

öğrenme algoritmalarının kullanılabildiği derin öğrenme (deep learning) algoritmaları son yıllarda popülerliği artan bir makine öğrenmesi sınıfıdır. Hangi öğrenme modelinin uygulanacağı problem tipine göre belirlenmektedir.

Denetimli makine öğrenmesi algoritmaları, sınıflandırma (classification) ve regresyon (regression), denetimsiz makine öğrenmesi algoritmaları, kümeleme (clustering) ve boyut azaltma (dimensionality reduction) başlıkları altında incelenmektedir.

Denetimli öğrenmede, bir grup girdi değerine karşılık gelen, hedef değerleri verilerek oluşturulan modelin, girdi-hedef arası ilişkiyi öğrenerek, hedef değerlere en yakın çıktıların üretilmesi amaçlanmaktadır [7]. Sistem, önceden verilen eğitim setleri ve bunların eğitim kümesindekilerle benzerlikleri karşılaştırarak tahminde bulunur [8]. Denetimli öğrenmede amaç, en düşük hata payı ile en doğru tahmini yapabilmektir. Bu nedenle denetimli öğrenme algoritmaları araştırmacılar tarafından en çok tercih edilen algoritmalar arasında yer almaktadır. Fakat denetimli öğrenme yöntemlerinin uygulanmasında başarı oranını etkileyen en önemli sorun yeterli miktarda etiketli veriye sahip olmamaktır.

Denetimli öğrenmede, modeli eğitmek ve sınıflandırıcı performansını ölçmek için etiketli veri setlerine ihtiyaç vardır. Eğitim setini test için kullanmak performans değerlendirmesi için iyi bir yöntem olmayacağı için etiketli veri seti belirli oranlarda eğitim ve test olmak üzere iki gruba ayrılır.

Bu öğrenme modelinde, test için kullanılan veriler eğitim verileri ile karıştırılmaz. Test verilerinin gerçek sınıfları bilindiği için sınıflandırıcı tahminleri ve gerçek etiketler kıyaslanarak model başarısı tespit edilir [9].

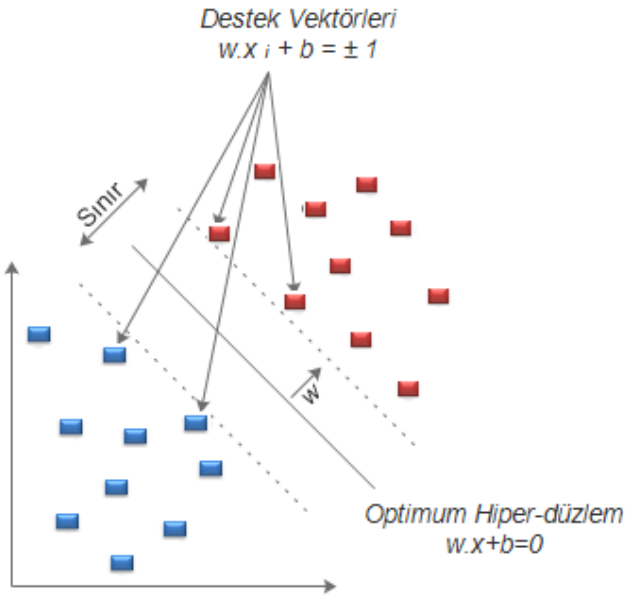
Veri setinden bilgi çıkarmak için kullanılan farklı yaklaşımlara sahip sınıflandırıcılar mevcuttur. Bununla birlikte sınıflandırıcı performansı bir veri setinden diğerine değişebilmektedir. Bu nedenle veri seti üzerinde en başarılı sınıflandırıcı performansını tespit etmek için birden fazla sınıflandırıcı ile veri setini test etmek önemlidir.

1.3.1. Destek vektör makineleri (support vector machine)

Destek Vektör Makineleri (SVM), istatistiksel öğrenme teorisini temel alan, basit ve etkili denetimli öğrenme algoritmalarından biridir.

SVM ilk olarak, Vapnik ve Cortes (1995) tarafından, sınıflandırma ve örüntü problemlerine çözüm için geliştirilmiştir [10]. SVM, doğrusal (linear) olarak ayrılan ve iki sınıflı problemleri çözmek için geliştirilmiştir fakat daha sonra doğrusal olmayan (non-linear) ve ikiden fazla sınıf etiketinin bulunduğu problemlerin çözümü için de kullanılmaya başlamıştır.

SVM'de, hiper düzlem adı verilen sınır çizgisi ile ayrılan farklı sınıflara ait destek vektörleri arasındaki uzaklığı maksimize etmek amaçlanmaktadır. Hiper düzleme en yakın sınıflara ait örnekler destek vektörleri olarak isimlendirilir. Destek vektörleri, $w.\ xi + b = \pm 1$ formülü ile ifade edilir. Destek vektörleri, hiper düzleme paralel bir düzlem üzerinde bulunur ve dahil olduğu sınıfın sınırını belirler [11]. Şekil 1.2'de optimum hiper düzlem ve destek vektör makinelerinin düzlem üzerindeki durumları gösterilmiştir. Kesikli çizgilerle gösterilen, üzerinde destek vektörlerinin yer aldığı düzleme sınır düzlemleri denir. Sınır düzlemlerinin tam ortasından geçerek her iki düzleme eşit uzaklıkta bulunan düzleme ise hiper düzlem adı verilmektedir.



Şekil 1. 2. Optimum hiper düzlem ve sınıflara ait destek vektörleri

Pang ve ark., makine öğrenmesini ilk kez duygu sınıflandırmak amacıyla kullanmışlardır. Çalışmalarında, NB, ME ve SVM sınıflandırıcılarını kullanarak film yorumlarını olumlu/olumsuz sınıflandırmışlardır. Öznitelik seçimi için unigram, unigram-bigram ve unigram-POS metodlarını kullanmışlardır. En iyi performansı SVM ile elde etmişlerdir [12].

1.3.2. Naive Bayes

Naive Bayes (NB), basitliği ve hesaplama açısından verimli olması nedeni ile yaygın olarak tercih edilen sınıflandırma yöntemidir. Naive Bayes modeli, şartlı bağımsızlık ilkesine dayanmaktadır [13].

Naive Bayes modeli, Bayes teoremi ile belirli bir özellik kümesinin belirli bir etikete ait olma olasılığını tahmin etmeye çalışır. Bayes teoremi, farklı parametreler ile bir olayın gerçekleşme ihtimalinin değişebileceğini ifade eder.

Naive Bayes modeli, bir sınıfa ait her özelliği diğer özelliklerden bağımsız olduğunu farz etmektedir. Bu sebeple her bir özellik, sonuç olasılığını bağımsız olarak etkiler. Bu varsayıma bağlı olarak Naive Bayes'in, özellikle girdi boyutunun fazla olduğu durumlar için uygun olduğu söylenebilir [14].

Naive Bayes algoritmasının metin belgeleri için özelleştirilmiş hali Multinomial Naive Bayes algoritmasıdır. Naive Bayes belirli kelimelerin doküman içinde bulunup bulunmadığını belirlerken, Multinomial Naive Bayes, doküman içerisinde bulunan kelimerin tekrar etme sayılarından frekans hesabı yaparak olasılık kümesi oluşturmaktadır.

Naive Bayes kullanımı örneklendirilecek olursa, bir x örneğinin, n boyutlu bir uzayda Di sınıfına ait olma olasılığı Denklem (1.1)'deki gibi hesaplanmaktadır.

$$P(Di|x)=\frac{P(Di)*P(x|Di)}{P(x)} \tag{1.1}$$

Denklem (1.2)'de gösterildiği gibi daha açık bir şekilde ifade edilecek olursa;

$$P(Di|x)=\frac{P(Di)P(f1, f2,\ldots fn|Di)}{P(x)P(f1, f2,\ldots fn)} \tag{1.2}$$

x'in özelliklerinin istatistik olarak bağımsız dağıldığını ifade eden Naive hipotezi dikkate alındığında Denklem (1.3)'deki gibi yeniden düzenlenebilir.

$$P(Di|x) = \frac{P(Di)P(f1, f2,\ldots fn|Di)}{P(f1, f2,\ldots fn)} \quad (1.3)$$

Denklem (1.4)'de gösterildiği gibi sınıf etiketi olarak, olasılığı en yüksek sınıf belirlenir.

$$Sınıf = \text{argmax}\ (P(Di|x)) \quad (1.4)$$

Denklem (1.5)'de gösterildiği gibi payda sabit olduğu için önemsenmez ve denklem sadeleştirilir.

$$P(Di)P(di|x) \propto P(Di)P(Di)P(f1|Di\ ,\ f2|Di\ldots fn|Di) \quad (1.5)$$

Onan çalışmasında, makine öğrenmesi algoritmalarından, NB, SVM ve LR ile Türkçe Twitter mesajlarında duygu sınıflandırma işlemi gerçekleştirmiştir. Metin temsili için 1-gram, 2-gram ve 3-gram öz nitelik bulma yöntemlerinden yararlanılmıştır. Çalışma sonucunda, 1-gram ve 2-gram öznitelik setlerinin birleştirilmesiyle oluşturulan öznitelik seti ve NB sınıflandırma algoritması ile en yüksek başarı oranı %77,78 elde edilmiştir [15].

1.3.3. Lojistik regresyon (logistic regression)

Lojistik Regresyon (LR) modeli, iki veya çok sınıflı sınıflandırma işlemleri için kullanılan çok değişkenli istatistik yöntemlerinden biridir. LR'de, az sayıda değişkeni kullanarak, bağımlı ve bağımsız değişkenler arası ilişkiyi tanımlayabilen bir model kurmak amaçlanmaktadır [16]. LR modeli, matematiksel olarak Denklem (1.6) ile ifade edilmiştir [18]. Burada, x kelime vektörü, y ikili sınıflandırmada o sınıfa ait olup, olmama durumu ve θ bağımsız değişkenlerin (kelime vektörleri) regresyon çarpanlarını oluşturan parametre vektörünü ifade etmektedir. LR, kategorik bağımlı değişken ile bir veya daha fazla bağımsız değişken arasındaki ilişkiyi, lojistik fonksiyon aracılığı ile olasılıkları tahmin ederek ölçmektedir.

$$P(y|x) = \frac{1}{1+\exp(-y\theta^T x)} \quad (1.6)$$

1.3.4. Karar ağacı (decision tree)

Karar ağacı (DT) yapısı, her düğümün bir özniteliği, her bağlantının bir kuralı ve her yaprağın bir sonucu temsil ettiği bir ağaç yapısıdır.

DT yapısında en üstte kök düğümü yer almaktadır. DT, özyinelemeli olarak bölümlere ayrılır ve öznitelik değerlerinin temelindeki bölümlemeyi öğrenir. Böylece ağaç yapısı karar verme işlemini gerçekleştirmiş olur. Karar ağaçları, kolaylıkla anlaşılabilmekte ve yorumlanabilmektedir.

Literatürde yaygın bilinen DT algoritmaları, ID3, C4.5, C5.0 ve CART olmak üzere dört çeşittir.

Tek değişkenli karar ağaçlarından ID3 algoritması, her bir düğüm için hedeflerin en büyük bilgi kazancını sağlayabilen bilgi kazancı yaklaşımını kullanmaktadır. ID3 algoritmasında, ağaçlar maksimum boyuta gelene kadar büyütüldükten sonra ağacın görünmeyen verilere genelleme yeteneğini geliştirmek için budama işlemi uygulanır.

C4.5 algoritması, ID3 algoritmasının eksikliklerini ve yetersizliklerini gidermek amacıyla 1993 yılında Quinlan tarafından geliştirilmiştir [19]. C4.5 algoritması, bölünme bilgisi ile bilgi kazancından faydalanarak hesaplanan kazanç oranı yaklaşımını kullanmaktadır.

C5.0 algoritması, C4.5 algoritmasının daha az bellek kullanan geliştirilmiş bir versiyonudur.

İstatistiksel bir yaklaşım olan CART algoritmasında, her karar düğümünden en büyük bilgi kazancını sağlayan özellik ve eşiği kullanarak ikili ağaçlar oluşturulur.

1.3.5. Rastgele orman (random forest)

Rastgele Orman (RF), Leo Breiman tarafından karar ağaçları temel alınarak geliştirilmiştir [20]. RF, n tane karar ağacının rastgele bir araya gelmesiyle oluşturulur. Sınıflandırılmak istenen dokümanın hangi sınıfa ait olduğunun tespiti için ağaç kümesinde yer alan her bir ağaca giriş vektörü verilir. Verilen giriş vektörüne göre her bir ağaç çıkış üretir. Tüm ağaç kümesi arasında oylama yapılır ve

en çok oyu alan sınıflandırma sonucu olarak seçilir. Her bir ağaç, eğitim setinde yer alan örneklerin rastgele bir şekilde yenisiyle değiştirilmesi ile oluşturulur.

RF algoritması, başarılı sonuçlar verdiği için regresyon ve sınıflandırma problemlerinde kullanılmaktadır. RF algoritmasının, özellikle sınıflandırma problemlerinde daha doğru sonuçlar ürettiği, aşırı öğrenme ve gürültülere daha dayanıklı olduğu görülmüştür. Ayrıca, yükseltme (boosting) [21] ve torbalamaya (bagging) [22] kıyasla daha hızlı çalışmaktadır [20].

1.3.6. Adaptive boosting (adaboost)

Adaptive Boosting (Adaboost), Yoav Freund ve Robert Schapire tarafından formüle edilmiş bir meta-algoritmadır. Adaboost algoritması, performansını artırmak için topluluk yapısı içindeki zayıf sınıflandırıcıları kullanan yinelemeli bir topluluk sınıflandırıcısıdır. Adaboost algoritmasında, topluluğun sınıflandırıcıları teker teker eklenir, burada sonraki her sınıflandırıcı, önceki topluluk üyelerinin doğru şekilde sınıflandırmada başarısız olduğu veriler kullanılarak eğitilir. Yani son eğitim tahminine dayalı olarak mevcut öğrenme modelini eğitmek için eğitim setini seçer [23]. Bu sınıflandırıcının bir dezavantajı, gürültü noktalarına ve aykırı değerlere karşı çok hassas olmasıdır. Sınıflandırıcıya beslenen eğitim verilerinin yüksek kalitede olması gerekmektedir [24].

1.3.7. Extreme gradient boosting (xgboost)

Extreme Gradient Boosting (XGBoost) ölçeklenebilir, taşınabilir ve hesaplamalı olarak derlenmiş bir gradyan ağacı güçlendirme algoritması paketidir. Gradyan ağacı algoritması optimizasyon problemini iki temel adımda (önce adımın yönünü, ardından adım boyutunu belirler) çözmeye çalışırken, XGBoost adım boyutunu ve yönünü tek seferde bulur. XGBoost algoritması, işlem süresinin düşürülmesi ve bellek kaynaklarının en verimli şekilde kullanılması için tasarlanmıştır [25]. Düzenlileştirme (regularizasyon) parametrelerinin ayarlanabilmesi sayesinde modelin aşırı eğitilmesi (overfitting) önlenmekte ve modeldeki ağaçların karmaşıklığı kontrol edilerek başarı oranı yükseltilebilmektedir [26].

1.3.8. Bagging algoritması

Bagging algoritması, Breiman tarafından geliştirilen topluluk öğrenmesi yöntemidir [22]. Eğitim setinin çeşitli örneklemleri üzerinde, eğitilmiş temel öğrenme algoritmalarının birleştirilmesiyle sınıflandırıcı topluluğu oluşturulması prensibine dayanır. Veri setinden oluşan eğitim setlerini çeşitlendirmek amacıyla genellikle basit rastgele yerine koyarak örnekleme yöntemi uygulanmaktadır. Elde edilen eğitim setleriyle eğitilen sınıflandırma yöntemlerinin çıktıları, çoğunluk oylaması vasıtasıyla birleştirilmektedir [27].

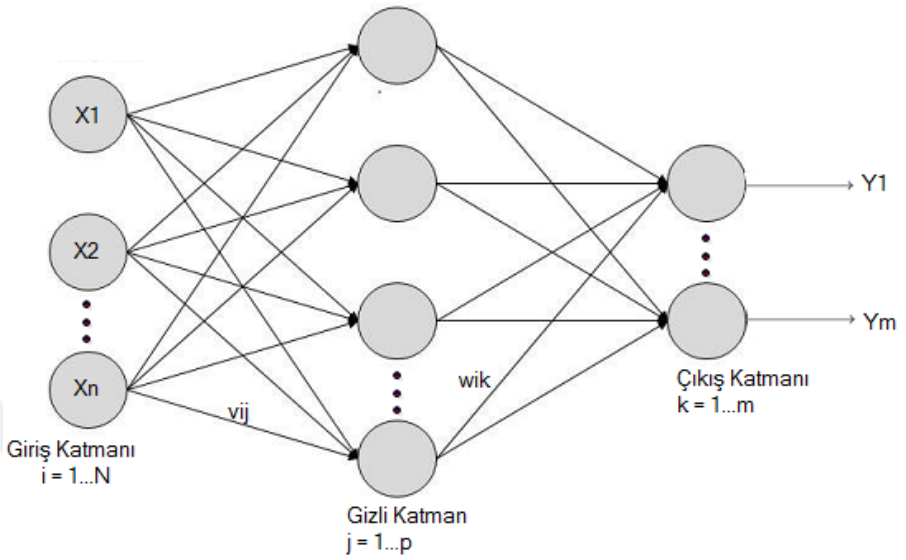
1.4. Derin Öğrenme

Makine öğrenmesi algoritmalarından hem denetimli hem de denetimsiz öğrenme algoritmalarının kullanılabildiği derin öğrenme (deep learning) algoritmaları son yıllarda popülerliği artan bir makine öğrenimi sınıfıdır. Aynı anda birçok işlemi yapan, çok katmanlı sinir ağı yapısıyla son yıllarda yapılan çalışmaların arttığı makine öğrenmesi alanıdır. Derin öğrenmenin insan beynini örnek alan, çok katmanlı ve doğrusal olmayan yapısı ile karmaşık problemlerin çözülebilmesi amaçlanmıştır. Öğrenme süreci, sonucun başarı oranı belli bir seviyeye ulaşana kadar tekrarlanır. Derin öğrenmenin diğer makine öğrenimi yöntemlerinden farkı, birden fazla doğrusal olmayan işlem katmanı ile çok yüksek miktarda veri işleme kapasitesine sahip olması ve hesaplama gücü yüksek donanımlara gereksinim duymasıdır [28]. Derin öğrenmenin en önemli özelliği, çok katmanlı mimarisi sayesinde büyük miktarda veriyi işleyebilmesidir. Derin öğrenmede gereken eğitim süresi yüksek olabilir fakat verileri test etmek daha az zaman alır [29]. Derin öğrenme, bilgisayarla görme, konuşma tanıma ve doğal dil işleme gibi birçok alanda başarı sağlamıştır [30]. Son yıllarda veri miktarının artması ve işlem gücü yüksek donanım araçlarının gelişmesiyle, derin öğrenme yöntemlerinin duygu analizinde kullanıldığı çalışmalarda artış görülmektedir.

1.4.1. Yapay sinir ağı (artificial neural network)

Yapay Sinir Ağları (ANN), insan sinir sistemi dikkate alınarak tasarlanmış ve bu yapıda öğrenmeyi hedefleyen güçlü bir makine öğrenmesi yöntemidir. Şekil 1.3'de gösterildiği gibi birden fazla nöron birleşerek katmanları, katmanlar birleşerek ANN

modelini meydana getirmektedir. Bir ANN modeli, giriş ve çıkış katmanları olmak üzere en az iki katmandan oluşmaktadır. Giriş katmanında eğitim için kullanılacak öznitelik sayısı kadar, çıkış katmanında sınıf sayısı kadar nöron bulunmalıdır.



Şekil 1. 3. En sık tercih edilen YSA modeli olan çok katmanlı algılayıcı sinir ağı (MLP); giriş katmanı, gizli katman ve çıkış katmanı

Giriş katmanı, verilerin okunduğu katmandır. Her bir özellik bir nöron tarafından temsil edildiği için özellik sayısı kadar nöron içermektedir. Bütün girişlerin belirli ağırlık değerleri vardır. Bu girişler ara katmandaki nöronlara bu ağırlık değerleriyle bağlanmıştır. Ağırlıklar ilgili özelliğin önem derecesini ifade etmektedir.

Gizli (ara) katman, giriş katmanı ile çıkış katmanı arasında yer alan verilerin işlendiği katmandır. Verilerin işlendikten sonra çıkış katmanına ulaşması ağırlık değerleri kullanılarak yapılmaktadır. Gizli katman sayısı ve bir gizli katmanda yer alacak nöron sayısı tam olarak belli olmamakla birlikte, eğitimin kalitesine önemli etkileri olan iki unsurdur [31].

Çıkış katmanı, verilerin dahil olduğu sınıfların belirlendiği katmandır. Çıkış katmanı, oluşturulan modele göre tek bir nörondan oluşabileceği gibi sınıf çeşidi kadar nörondan da oluşabilmektedir. Hesaplanan ya da beklenen değerler arası fark, “hata fonksiyonu” ile hesaplanmaktadır. Bulunan hata değeri, gerçek sonuca ne kadar yakın ya da uzak olunduğunu göstermektedir. Hesap edilen değere göre

“optimizasyon fonksiyonu” ile ağırlıklar güncellenerek öğrenme işlemi gerçekleştirilir.

Yapay sinir ağlarında öğrenme işlemi ağırlıkların güncellenmesi ile başlamaktadır ve hatanın optimize edilmesi ile sürdürülmektedir. Beklenen sonuçları veren ağırlıklara ulaşıldığında, eğitimin tamamlanmış olduğu söylenebilir.

1.4.2. Konvolüsyonel sinir ağları (convolutional neural networks)

Konvolüsyonel sinir ağları (CNN), hayvanların görme merkezinden yola çıkılarak geliştirilmiş olan çok katmanlı algılayıcı çeşitlerinden biridir. Burada gerçekleştirilen işlem, bir nöron hücresinin kendi bölgesinde yer alan uyaranlara verdiği yanıttır. Konvolüsyonel sinir ağları, bir ya da daha fazla konvolüsyonel katman, altörnekleme katmanı ve ardından standart çok katmanlı sinir ağı gibi bir ya da birden çok bağlı katmandan oluşmaktadır.

Yann LeChun, 1988 yılında ilk konvolüsyonel ağ olan LeNet isimli mimariyi ortaya atmıştır. Bu mimarinin iyileştirmeleri 1998 yılına kadar sürmüştür. Bu ağ sisteminde, alt katmanlar, maksimum havuzlama ve konvolüsyon katmanlarından, alt katmanlardan sonra gelen üst katmanlar ise tamamen bağlı geleneksel MLP’den oluşmaktadır.

1.4.3. Tekrarlayan sinir ağları (recurrent neural network)

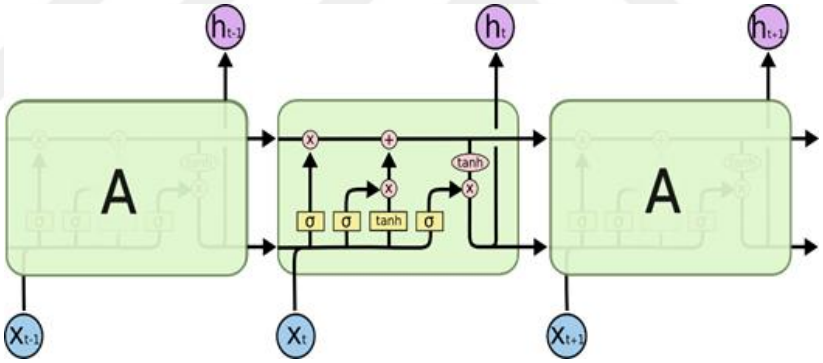
Jeff Elman tarafından tasarlanan ilk basit tekrarlayan ağ tasarım simülasyonunda isim ve fiil kategorileri düzgün bir şekilde ayrılmıştır. Bununla birlikte isimler, canlı/cansız ve insan/hayvan olarak hayvanlar ise avcı/yırtıcı gibi alt başlıklar altında gruplandırılmıştır.

Tekrarlayan sinir ağı, sıralı bilgileri kullanmak için ilgili bağlantıların yönlü döngü oluşturduğu bir modeldir. Geleneksel sinir ağında giriş ve çıkışların bağımsız olduğu düşünülmektedir fakat bu durum doğal dil işleme için uygun görünmemektedir. Örneğin, bir cümle içinde bir kelimeden sonra gelebilecek olan kelimenin tahmini, hangi kelimenin o kelimeden önce geldiğini bilmekle mümkündür. Bu yapının tekrarlanan olarak isimlendirilmesi, ilgili cümledeki her kelime için aynı görevi çıktılara göre gerçekleştirmesidir.

1.4.4. Uzun kısa süreli hafıza ağları (long short term memory)

Hochreiter ve Schmidhuber tarafından 1997 yılında tanıtılan Uzun Kısa Süreli Hafıza Ağları, uzun süreli bağımlılıkları öğrenebilen özel bir RNN türüdür [32]. RNN ve LSTM modellerinin farklılaşması saklı durum değerlerinin hesaplanması sırasında ortaya çıkmaktadır [33]. LSTM'ler uzun süreli bağımlılık problemini saklı durum hesaplamalarıyla önlemek için tasarlanmıştır. Şekil 1.4'de gösterildiği gibi özel bir yapıya sahip olan blokların sıralı bir şekilde kendini tekrar etmesi ile oluşmaktadır. Şekil 1.4'de gösterildiği şekilde LSTM'de RNN'den farklı olarak bir yerine dört tane sinir ağı katmanı vardır.

LSTM'deki hafızalar, hücre (cell) adı verilmektedir. Hücreler, hafızada hangi bilgilerin tutulacağı, hangi bilgilerin sileceği konusunda karar vericidir. Bir hücre anlık durum çıktısını oluşturmak için bir önceki durum çıktısı, güncel hafıza bilgileri ve güncel girdi bilgilerini akıllı bir şekilde birleştirilir.



Şekil 1. 4. LSTM Ağ Yapısı

LSTM ağlarında, hücre durumları her bir zaman aralığında, kapı veya geçit (gate) olarak adlandırılan yapılar aracılığıyla belirlenen uygun değerler ile değişmektedir. Kapı, isteğe bağlı olarak bilgi geçişine izin veren veya bilgi geçişini engelleyen bir bileşendir. LSTM dört adet sinir ağı katmanı aracılığı ile girdi (input), unutma (forget) ve çıktı (output) kapıları, hücre durumunda ağ hafızasını oluşturmaktadır.

LSTM'de ilk kararı girdi kapısı katmanı (input gate layer) vermektedir. Şekil 2'de gösterilen girdi kapısı katmanı hücre durumundan hangi bilgilerin atılacağı kararını vermek için kullanılmaktadır ve sonucunda 0-1 aralığında bir değer döner. Geri

dönen değer, 0 ise bilginin silinmesi gerektiğini, 1 ise bilginin saklanması gerektiğini ifade etmektedir.

İkinci adım unutma kapısı katmanı (forget gate layer) ve tanh katmanı iki parçalı bilgi saklama aşamasını göstermektedir. Bu katmanda, hücre durumunda hangi verilerin saklanacağı belirlenmekte ve yeni eklenen aday bilgilerin yer aldığı vektörler hesaplanmaktadır. Bu iki bilgiyle hücrenin yeni durumu, unutulan ve güncellenen bilgilerle birlikte oluşturulmaktadır.

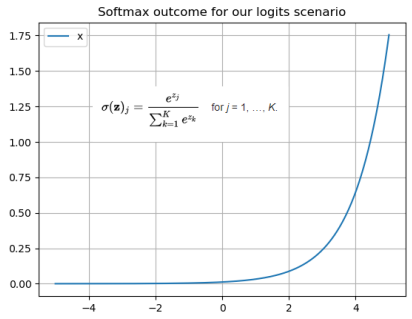
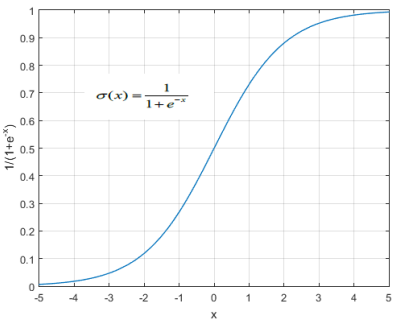
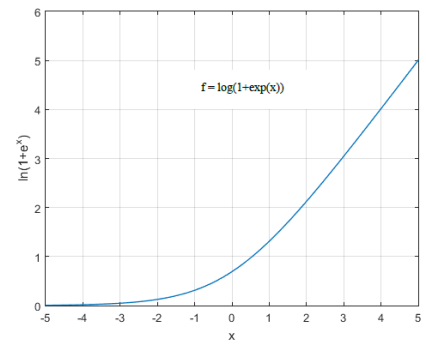
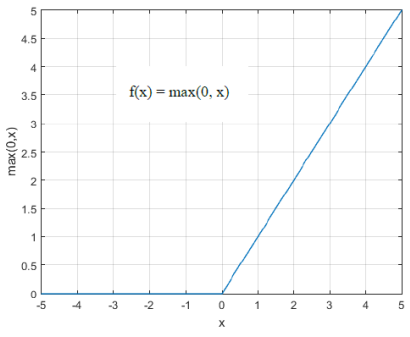
Son olarak çıktı kapısı katmanı (output gate layer), neyin çıktı olarak gönderileceğine karar verilen katmandır.

Pragya ve ark., ISEAR veri seti üzerinde MNB, SVM, RF, CNN ve LSTM öğrenme yöntemleri ile yedi duygu sınıfı için duygu analizi çalışması gerçekleştirmişlerdir. MNB ve DVM sınıflandırıcıları ile yapılan deneyde TF-IDF ve BOW vektörizasyon işlemlerinin sınıflandırma sonuçlarına etkileri karşılaştırılmıştır. Ek olarak, LSTM, RF ve CNN yöntemleri ile yapılan deneylerde Glove2Vec ve Word2Vec vektörizasyon yöntemlerinin sınıflandırma sonuçlarına etkileri gözlemlenmiştir. Çalışma sonucunda, %64 başarı oranı ile LSTM yönteminin diğer sınıflandırıcılara göre daha başarılı olduğunu görülmüştür [34].

1.5. Aktivasyon Fonksiyonları

Aktivasyon fonksiyonları, bir katmanda yer alan nöronların çıktı değerlerini sonraki katmanlara iletmek için kullanılan fonksiyonlardır. Çıktı değerinin, sonraki katmanlara iletilip iletilmeyeceğine belirlenen eşik değer ile karar verilmektedir. Nöronun aktiflik durumuna karar verebilmesi için aktivasyon fonksiyonlarına ihtiyaç duyulmaktadır. Aktivasyon fonksiyonları ile bir nöronun çıktı değeri kontrol edebilecek ve dış bağlantıların nöronu aktif olarak görüp görmeyeceğine karar verilebilecektir. Yapay sinir ağı modelleri çoğunlukla doğrusal olmayan sınıflandırmalarda kullanıldığı için aktivasyon fonksiyonları da genellikle doğrusal olmayan bir fonksiyonlardan seçilmektedir [35]. Tablo 1.1'de çalışmada kullanılan aktivasyon fonksiyonları ve özellikleri verilmiştir [36].

Tablo 1. 1. Aktivasyon fonksiyonları

Aktivasyon Fonksiyonu	Grafik	Açıklama
Softmax		Birden fazla sigmoid fonksiyonun kombinasyonu olarak tanımlanır. Girdinin belirli sınıfa ait olma olasılığını ölçmek için (0,1) aralığında değer üretmektedir.
Sigmoid		Sigmoid fonksiyonu, tanım kümesindeki elemanların her biri için (0,1) aralığında bir değer üretmektedir.
Softplus		Softplus fonksiyonu, dönüşümü optimize edilecek bir fonksiyonun parametreleri üzerinde pozitif değerleri sınırlandıran aktivasyon fonksiyonudur.
ReLu		Relu fonksiyonu, negatif değerleri 0'a dönüştürürken, pozitif değerleri olduğu gibi tutmaktadır; bu da fonksiyonun daha hızlı çalışmasını sebep olmaktadır.

1.6. Öznitelik Seçimi

Öznitelik seçimi, veri kümesi içinden, sınıflandırma başarısını etkileyen, alakasız niteliklerin silinmesi ya da önemli niteliklerin seçilmesi işlemidir. Veri setinde yer alan her bir verinin sahip olduğu çeşitli nitelikler (özellikler) vardır. Her bir nitelik, sınıflandırma algoritmasının başarısında aynı etkiye sahip olmayabilir. Hatta bazı nitelikler gereksiz ve sınıflandırma başarısını, performansını olumsuz olarak etkileyen nitelikler olabilmektedir.

Son yıllarda sınıflandırma başarısını ve performansını arttırmak için öznitelik seçim konusun araştırmacılar için önemli bir çalışma alanı olmuştur. Öznitelik seçimi, biyoinformatik, metin madenciliği, görüntü analizi gibi alanlarda yaygın bir şekilde uygulanmaktadır.

1.6.1. Ki kare istatistiği (chi square)

Ki-kare istatistiği, bir özellik ve sınıf arası bağımsızlığın derecesini ölçen bir istatistiktir. Eğer ilgili c sınıfındaki bir f niteliğinin skoru düşükse bu nitelik daha az bilgi içerdiği için yok sayılabilir [37]. Bir f niteliğinin ki-kare değeri Denklem (1.7) ve Denklem (1.8)'e göre hesaplanmaktadır.

$$\chi^2(f,c)=\frac{N\,(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{1.7}$$

$$\chi^2(f)=\sum_{i=1}^{C} P(c_i)\chi^2(f,c_i) \tag{1.8}$$

1.6.2. Bilgi kazancı (information gain)

Bilgi Kazancı öznitelik seçim yönteminde, sınıflar hakkında en çok bilgi veren kelimelerin yani sınıfları ayırma özelliği en yüksek olan kelimelerin seçilmesine olanak sağlanmaktadır. Bilgi Kazancı yönteminde, sıfır olan (IG=0) terimler elenerek öznitelik olarak kullanılmamaktadır.

Bilgi kazancı ölçümü için belirsizliğin ve beklenmeyen durumların ortaya çıkma olasılığını ölçen entropi kullanılmaktadır [38]. Entropi değeri yüksek verilerde daha

fazla bilgi bulunmaktadır. Sınıflandırma işleminde, öznitelikler ile ne kadar bilgi kazanılabileceğini gösterilmektedir.

Bilgi kazancı algoritmasında her bir özniteliğin bilgi kazancı değeri hesaplanarak öznitelik seçimi yapılmaktadır. Yapılan işlemler sonucunda elde edilen kazanç verileri daha önce belirlenmiş olan eşik değerden düşük çıkarsa, bu öznitelikler sınıflandırma sürecinde kullanılmaya uygun değildir [39].

I bilgi kaynağı olmak üzere, I kaynağının entropisi Denklem (1.9)'da gösterildiği gibi tanımlanmaktadır. Bu kaynağın $\{w_1, w_2, w_3, \ldots, w_N\}$ olmak üzere N mesaj oluşturabildiği varsayılırsa, tüm mesajlar birbirinden bağımsız olarak oluşturulmaktadır ve w_i mesajlarının oluşturulma olasılıkları p_i'dir. A niteliğinin I veri kümesindeki bilgi kazancı Denklem (1.10)'da, Bilgi Kazancı Denklem (1.11)'de tanımlanmaktadır.

$$H(I)=\sum_{i=1}^{N} -p_i \log p_i \tag{1.9}$$

$$P(v)= |I(v)|/ |I| \tag{1.10}$$

$$Gain(I,A)=Entropy(I)- \sum P(v)\ Entropy(I(v)) \tag{1.11}$$

1.6.3. Varyans analizi (analysis of variance)

Varyans analizi (ANOVA), öznitelik seçimi için kullanılan yöntemlerden biridir [40]. ANOVA, her bir özelliğin etiket üzerindeki etkisini hesapladıktan sonra etiket üzerinde en fazla etkiye sahip olan herhangi bir özelliğin en bilgilendirici özellik olduğunu kabul eder. ANOVA yönteminde hesaplanan değer, her özelliğin etiketiyle birlikte kovaryansıdır. Bu yöntem ile bir özellik değiştiğinde, etikette ne kadar değişiklik olduğunu bilmemize olanak tanır.

1.7. Sınıflandırma Performans Metrikleri

Sınıflandırma işleminde gerçekleştirilen deney sonuçlarının değerlendirilmesi için, farklı ölçüm yöntemleri kullanılmaktadır.

Genellikle tek etiketli sınıflandırma çalışmalarına uygulanan değerlendirme ölçütleri bir verinin sınıflandırılmasıyla ilgili sadece doğru veya yanlış iki olası durum değerlendirilmektedir.

Değerlendirme ölçütü, sınıflandırıcı performansını ölçen ölçüm aracıdır. Farklı ölçüm metrikleri, sınıflandırıcıların farklı özelliklerini değerlendirmektedir [41].

En etkili algoritmanın tespiti için bir çok kriter kullanılmaktadır. Bu kriterler; doğru pozitif (DP) oranı, doğru negatif (DN) oranı, yanlış pozitif (YP) oranı, yanlış negatif (YN) oranı, hassasiyet (precision), geri çağırma (recall), f-ölçütü (f-measure), kappa (κ) istatistiği, alıcı işlem karakteristiği eğrisi (ROC: Receiver Operating Characteristic), ortalama mutlak hata (MAE: Mean Absolute Error), kök ortalama kare hata (RMSE: Root Mean Square Error), Matthews korelasyon katsayısı (MCC: Matthews Correlation Coefficient) şeklindedir [42].

1.7.1. Karmaşıklık matrisi (confussion matrix)

Karmaşıklık matrisi, tahminlerin doğruluğu hakkında anlaşılması kolay bilgiler sağlayan bir ölçüm aracıdır. Özellikle sınıflandırma algoritmaları ile çalışılırken araştırmacılar tarafından sıklıkla tercih edilmektedir. Karmaşıklık Matrisi, Tablo 1.2'de gösterilmiştir.

Değerlendirme ölçme yöntemlerinin tümü, S sınıfına göre doğru pozitif (TP), doğtu negatif (TN), yanlış pozitif (FP) ve yanlış negatif (FN) sayıları yani gerçek test değerleri ve tahmin edilen test değerleri sayıları kullanılmaktadır.

Tablo 1. 2. Karmaşıklık matrisi

Gerçek Sınıflar / Tahmin Edilen Sınıflar	S_1	$-S_1$
S_1	Doğru Pozitif (TP)	Yanlış Negatif (FN)
$-S_1$	Yanlış Pozitif (TP)	Doğru Negatif (TN)

1.7.2. Doğruluk (accuracy)

Doğruluk (Accuracy), tüm doğru tahminlerin toplam sayısının veri setinin toplam sayısına bölünmesi ile elde edilen, en çok tercih edilen sınıflandırma ölçütüdür. Doğruluk değeri, 1,0 (en iyi) ve 0,0 (en kötü) arasında değişmektedir. Doğruluk değeri, Denklem (1.12)'de gösterildiği gibi hesaplanmaktadır [43].

Model performansını değerlendirirken ana performans ölçütü olarak doğruluk kullanmanın dezavantajı, sınıflar arasında büyük oranda dengesizlik varsa ölçüt yüksek sonuçlar vermemektedir.

$$Acc=\frac{TP+TN}{TP+FP+TN+FN} \tag{1.12}$$

1.7.3. Hassasiyet (precision)

Hassasiyet (Precision), doğru pozitif tahmin sayısının, pozitif tahminler toplamına bölünmesiyle hesaplanan ölçüdür. Hassasiyet değeri, Denklem (1.13)'de gösterildiği gibi hesaplanmaktadır [43].

$$p=\frac{TP}{TP+FP} \tag{1.13}$$

1.7.4. Geri çağırma (recall)

Geri çağırma (Recall), doğru pozitif tahmin sayısının, pozitif örneklerin toplam sayısına bölünmesiyle hesaplanan ölçüdür. Geri çağırma değeri, Denklem (1.14)'te gösterildiği gibi hesaplanmaktadır [43].

$$r=\frac{TP}{TP+TN} \tag{1.14}$$

1.7.5. F-ölçütü (f-measure)

F-Ölçütü (F-Measure), geri çağırma ve hassasiyet ölçüm değerlerinin harmonik ortalaması alınarak hesaplanmaktadır. F-Ölçütü değeri, Denklem (1.15)'te gösterildiği gibi hesaplanmaktadır [43].

$$\text{F-measure}=\frac{2\text{RecallPrecision}}{\text{Recall+Precision}} \tag{1.15}$$

1.7.6. Kappa (κ) istatistiği

κ istatistiği, sınıflandırma algoritmasının performans başarısı ile ilgilenmektedir. Kategorik değişkenler için yapılan analizlerin anlaşılabilmesi için kullanılan istatistiki bir veridir. κ istatistiği, ki-kare tablosunu temel alan bir değerdir [44]. p0 ve pe 'nin iki kategorik değişken arasındaki gözlenen ve beklenen değerlerini gösteren κ istatistiği Denklem (1.16) ile hesaplanır [42].

$$\kappa=\frac{p_0 - p_e}{1 - p_e} \tag{1.16}$$

p_0 ve p_e değerleri sırası ile Denklem (1.17) ve Denklem (1.18) ile hesaplanır.

$$p_0=\frac{TP+TN}{TP+TN+FP+FN} \tag{1.17}$$

$$p_e=\frac{(TP+FP)x(TP+FN)+(FN+TN)x(FP+TN)}{TP+TN+YP+YN} \tag{1.18}$$

2. LİTERATÜR ÇALIŞMALARI

Bu tez çalışması kapsamında Türkçe Twitter paylaşımları üzerinde çok sınıflı metin sınıflandırma çalışması gerçekleştirmek amaçlanmaktadır.

Günümüz sınıflandırma çalışmalarının çoğu tek etiketli sınıflandırmaya odaklanarak birden fazla sınıf etiketinin bulunabilirliğini göz ardı etmiştir.

2.1. Çok Sınıflı Metin Sınıflandırma Çalışmaları

Bu bölümde, literatürde yapılmış çok sınıflı metin sınıflandırma çalışmaları yer almaktadır.

Rabbimov ve Kobilov, Özbek haber metinleri üzerinde çok sınıflı sınıflandırma çalışması gerçekleştirmişlerdir. Çok sınıflı metin sınıflandırma çalışmasında SVM, DTC, RF, LR, MNB olmak üzere 6 farklı makine öğrenme algoritması kullanmışlardır. Öznitelik çıkarma yöntemi olarak TF-IDF algoritmasını kelime ve karakter düzeyinde n-gram'lar ile birlikte kullanmışlardır. En yüksek doğruluk oranı olan %86.88 değerini SVM ile elde etmişlerdir [45].

Osmanoğlu ve ark., Anadolu Üniversitesi eCampus sisteminden toplanan geri bildirimler ile materyalin kişiler üzerindeki etkisini tespit etmek amacıyla analiz çalışması gerçekleştirmişlerdir. Bu bağlamda, veri seti üzerinde makine öğrenmesi teknikleriyle, pozitif, negatif veya nötr olarak sınıflandırma gerçekleştirerek negatif geri dönüşlü makalelerin geliştirilmesi için fikir vermeyi amaçlamışlardır. Çalışmada, denetimli makine öğrenmesi yöntemlerinden, DT, MLP, XGB, SVM, MLR, gaussian NB ve KNN algoritmalarını kullanmışlardır. Eğitim için, 6059 adet etiketli veri kullanılmıştır. %77,5 başarı oranıyla LR algoritması en başarılı sınıflandırma algoritması olmuştur [46].

Jabreel ve Moreno, Twitter paylaşımları üzerinde çoklu duygu sınıflandırma sınıflandırması problemini ele alan çok etiketli duygu analizi konusuna yeni bir yaklaşım önermişlerdir. Gerçekleştirilen çalışmada ilk olarak, çok etiketli

sınıflandırma problemini, ikili sınıflandırma problemine dönüştürmek için yeni bir yöntem önermişlerdir. Daha sonra, ikili sınıflandırma problemine dönüşen sorunu çözmek için yeni bir derin öğrenme yöntemi olan BNet'i önermişlerdir. Çalışma neticesinde 0.59 doğruluk oranı elde etmişlerdir [47].

Gürcan tarafından yapılan çalışmada, haber metinleri üzerinde denetimli öğrenme modelleri ile önceden belirlenen beş sınıfa (ekonomi, politika, spor, sağlık ve teknoloji) göre çok sınıflı metin sınıflandırma gerçekleştirmiştir. Gerçekleştirilen çalışmada, MNB, BNB, SVM, KNN ve DT algoritmalarının sınıflandırma performansları, farklı parametreler ile test edilmiştir. Çalışma sonucunda yaklaşık %90 sınıflandırma başarısı ile MNB en başarılı sonucu vermiştir [48].

Chen, Huang ve ark., CNN ve LSTM modellerine dayanan derin duygu temsil modeli önermişlerdir. Önerilen model, metnin kısmi özelliklerini yakalamak için iki CNN katmanı kullanmaktadır. Ardından özellikler bağlamsal bilgileri yakalayabilen LSTM ile beslenmektedir. Son olarak, geliştirilmiş derin öğrenme modelini çok sınıflı duyarlılık sınıflandırmasına uygulamışlardır. Çalışma sonucunda, çok sınıflı duyarlılık analizi için tasarlanan modelin, % 78,42'lik bir doğruluk elde ederek, mevcut SVM, CNN, LSTM yöntemlerinden daha iyi olduğu gözlemlemişlerdir [49].

Franko ve Parlak, İspanyolca dokümanlar üzerinde, 10 farklı kategoride, NB ve ME makine öğrenmesi teknikleriyle çok sınıflı metin sınıflandırma çalışması gerçekleştirmişlerdir. Çalışma sonucunda, doğruluk, kesinlik ve geri çağırma değerlerinde en başarılı sonuçları ME yöntemi ile elde etmişlerdir [50].

Quispe, Ocsa ve Coronado, haber verilerinin yer aldığı RCV1 veri seti üzerinde kelime temsil yöntemi olarak gizli anlamsal indeksleme (latent semantic indexing), özellik çıkarma yöntemi olarak CNN ve çok katmanlı algılayıcı (multi-layer perceptron) kullanarak çok etiketli ve çok sınıflı metin sınıflandırma çalışması gerçekleştirmişlerdir. Çalışma sonucunda, veri setindeki metin uzunluğu arttıkça modelin son teknoloji sınıflandırma tekniklerinden daha iyi performans gösterdiği, metinlerin boyutu düşük olduğunda önerilen modelin kesinliğinin zayıf olduğu görülmüştür [51].

2.2. Havayolu Firmaları ile İlgili Twitter Yorumları Üzerinde Gerçekleştirilen Sınıflandırma Çalışmaları

Bu bölümde, literatürde gerçekleştirilmiş olan havayolu firmaları ile ilgili Twitter yorumları üzerinde gerçekleştirilmiş olan sınıflandırma çalışmaları yer almaktadır.

Bilgin ve Şentürk, Amerikadaki 6 büyük havayolu firması ile ilgili 1 haftalık Türkçe ve İngilizce Twitter paylaşımları üzerinde yarı denetimli ve denetimli öğrenme yöntemlerini karşılaştırılarak duygu analizi çalışması gerçekleştirmişlerdir. Çalışma sonucunda, hassasiye metriği cinsinden, Türkçe veri seti üzerinde, yarı denetimli yöntem ile % 44.88, İngilizce veri seti üzerinde yarı denetimli yöntem ile % 62.06 başarı elde etmişlerdir. Yarı denetimli öğrenme yönteminin hem Türkçe hem de İngilizce veri kümesi üzerinde danışmanlı öğrenmeye kıyasla daha başarılı sonuçlar elde ettiği gözlenmiştir [52].

Yılmaz çalışmasında, ISEAR ve Twitter'dan elde edilen havayolu firmaları ile ilgili İngilizce paylaşımlar üzerinde, makine öğrenmesi yöntemleri ile beş duygu sınıfı için sınıflandırma gerçekleştirmiştir. Çalışma kapsamında, temel sınıflandırıcılar olarak MNB, SVM, DT sınıflandırıcıları ve Bagging, Boosting, Voting topluluk öğrenmesi yöntemleri kullanılmıştır. Havayolu firmaları ile ilgili Twitter yorumları üzerinde gerçekleştirilen sınıflandırma çalışmalarında, temel makine öğrenmesi yöntemlerinden SVM, % 62 ile k katlı çapraz doğrulama ortalaması en yüksek sınıflandırma yöntemi olurken, MNB, Doğrusal SVM ve RF sınıflandırıcıları ile oluşturulan Voting sınıflandırıcısı ile k katlı çapraz doğrulama sonucunda % 64 başarı oranı elde edilmiştir. Çalışma sonunda, topluluk öğrenmesi yöntemlerinin, temel sınıflandırıcıların başarı oranlarını artırdığı görülmüştür [53].

Rane ve Kumar çalışmalarında, 6 büyük ABD havayolu firması ile ilgili paylaşılan Twitter yorumlarından oluşan veri kümesi üzerinde, DT, RF, SVM, KNN, LR, Gauss NB ve AdaBoost algoritmaları ile pozitif, nötr ve negatif olmak üzere üçlü duygu analizi yapmışlardır. Çalışmada, öznitelik çıkarım metodu olarak Word2vec kütüphanesinin genişletilmiş bir versiyonu olan Doc2vec tercih edilmiştir. Model eğitiminde sınırlı sayıda tweet kullanılması çalışma için dezavantaj oluşturmuştur. Çalışma sonucunda, AdaBoost (% 84.5) ve RF (% 86.5) algoritmalarının yüksek performans gösterdiği gözlemlenmiştir [54].

Engüllü çalışmasında, “Sentiment Strength Twitter”, “Stanford Twitter Sentiment” ve “IMDB Movie Reviews” veri setleri üzerinde makine öğrenmesi tabalı, MNB, SVM, RF ve LR sınıflandırıcıları ve sözlük tabanlı, Valence Aware Dictionary and Sentiment Reasoner (VADER) duygu analizi aracı ve kendi oluşturduğu Domain Based Lexicon yaklaşımı ile Twitter duygu analizi çalışması gerçekleştirmiştir. Veri setlerinden havayolu veri seti üzerinde, ortalama en yüksek doğruluk oranı SVM (% 91) sınıflandırma yöntemi ile elde edilmiştir [55].

Kocak ve ark. çalışmalarında, Twitter kullanıcılarının hava taşımacılığı ile ilgili yorumlarını derleyerek bir duyarlılık analizi çalışması gerçekleştirmişlerdir. 8672 adet kullanıcı yorumu pozitif, negatif ve nötr olmak üzere üç sınıf etiketi ile ayrıştırılmıştır. Çalışmada etiketler bir etiket bulutu içinde toplanarak sonuçlar Makine Öğrenmesi Yöntemi ve SMO sınıflandırmasında standart ve normalize kernel polinomları ile analiz edilmiştir. Çalışma sonucunda, standart kernel polinomu başarı oranı % 58 ve normalize kernel polinomu başarı oranı % 55 olarak bulunmuştur. Sınıflandırma verilerinin dengesiz dağılmış olması ve kelimelerin tamamının analizde kullanılmış olmasının çalışma başarısının düşük çıkmasındaki etkenler olduğu sonucuna varılmıştır [56].

3. MATERYAL VE UYGULANAN YÖNTEMLER

Çalışma kapsamında kullanılan materyal ve uygulanan yöntemler bu bölümde açıklanmıştır.

Bu çalışmada gerçekleştirilen uygulamalar, nesne yönelimli ve üst düzey bir programlama dili olan Python ile Google Colab üzerinde gerçekleştirilmiştir. Bu çalışmada, Python programlama dilinin V.3.6.9 versiyonu kullanılmıştır. Çalışmada veri ön işleme adımları için Python programlama dilinin NLTK kütüphanesi, makine öğrenmesi geleneksel sınıflandırma algoritmaları için Scikit-learn, Pandas ve Numpy kütüphaneleri, derin öğrenme algoritmaları için Keras ve Tensorflow kütüphaneleri kullanılmıştır.

İlk aşamada, yapılan çalışmada kullanılan verilerin toplama ve etiketleme süreçleri hakkında bilgiler verilmiştir. İkinci aşamada, veri setlerinde yer alan metinlere, istenmeyen karakterlerin ve durak köklerin temizlenmesi, yazım denetimi, kök bulma gibi metin ön işleme adımları uygulanmıştır. Üçüncü aşamada, veri seti %80 eğitim %20 test olacak şekilde ayrılmıştır. Dördüncü aşamada, veri setinde yer alan kelimelerin kelime vektörü temsili modeli açıklanmıştır. Beşinci aşamada, TF-IDF yöntemi ile oluşturulan özellik vektörleri arasında en faydalı özellikler seçilmiştir. Son aşamada toplanan veriler ve eğitilen kelime modeli kullanılarak eğitilmiş makine öğrenmesi yaklaşımlarından olan SVM, MNB, LR, DT, RF, Adaboost, XGBoost, Bagging ve derin öğrenme yaklaşımlarından LSTM ve CNN sınıflandırıcıları ile sınıflandırma gerçekleştirilmiş ve sınıflandırma performansları karşılaştırılmıştır.

Çalışmada gerçekleştirilen, veri setinin oluşturulması, veri ön işleme, veri setinin eğitim ve test olarak ayrılması, özellik çıkarımı, terim ağırlıklandırma, özellik seçimi, sınıflandırma, değerlendirme / karşılaştırma, çok sınıflı sınıflandırma aşamaları Şekil 3.1'de tablo olarak gösterilmiştir.

Şekil 3. 1. Çok sınıflı sınıflandırma aşamaları

3.1. Veri Setinin Hazırlanması

Bu çalışma kapsamında, Türk havayolu firmaları ile ilgili belirli tarih aralığında paylaşılan Türkçe Twitter verileri farklı yöntemler ile toplanmıştır.

Verilerin toplanması iki farklı yöntem ile gerçekleştirilmiştir. Bu yöntemler Twitter API ve Twitter Archive Google Sheets (TAGS) 'dir. Toplamda 14.406 adet kullanılabilir veri elde edilmiştir.

3.1.1. Twitter api aracılığı ile veri toplama

Bu çalışma kapsamında, Twitter verilerinin büyük bir kısmı Twitter API aracılığı ile elde edilmiştir. Twitter API'ye erişim sağlamak için Python kütüphanesi olan Tweepy kütüphanesi kullanılmıştır.

İlk olarak, Twitter geliştirici hesabı için API'nin hangi amaçla kullanılacağının açıklandığı bir başvuru oluşturulmuştur. Başvuruya istinaden, Twitter ekibinin talep ettiği proje ile ilgili detaylı soruların yer aldığı email cevaplandıktan sonra Twitter ekibi tarafından erişim maili iletilmiştir. Twitter geliştirici sitesinden, Twitter API'ye erişebilmek için oluşturulan Twitter uygulaması ile consumer key, consumer secret, access token ve access token secret bilgileri elde edilmiştir. Elde edilen key ve token bilgileri, Tweepy kütüphanesinden Twitter API'sine erişim sağlamak için kimlik doğrulama işleminde kullanılmıştır. Kimlik doğrulama işleminden sonra Twitter'dan veri çekebilme yetkisi elde edilmiştir.

Twitter veri seti, 2019 yılının ikinci yarısından 2020'nin ilk yarısına Türkiye'deki havayolu firmalarının sosyal medya hesaplarının kullanıcı adları etiketlenerek paylaşılmış tweetler baz alınarak oluşturulmuştur. Kullanıcı adları; "pegasusdestek", "ucurbenipegasus", "AJ_Destek", "anadolujet", "OnurAir", "SunExpress", "TK_TR" şeklindedir.

3.1.2. Twitter archive google sheets (tags) aracılığı ile veri toplama

Bu çalışma kapsamında, Twitter API veri çekme limitlerinden dolayı 2020'nin son 6 ay Twitter verileri "Twitter Archive Google Sheets (TAGS)" ile birden çok Google Drive Docs makrosunun zamanlanması ile elde edilmiştir.

Martin Hawksey tarafından oluşturulmuş olan "Twitter Archive Google Sheets (TAGS)" kişisel Twitter API bilgilerine erişim gerektirmeyen sadece kimlik doğrulama ile herkese açık hesapların Twitter verilerini Google Drive tablolarda arşivleyebilen bir projedir. Çalışma kapsamında yeni bir Google ve Twitter hesabı oluşturularak "TAGS v6.1" sürümü indirilmiştir. TAGS v6.1, Google oturumu açıldıktan ve Drive izinleri ve Twitter izinleri verildikten sonra kullanıma hazır olacaktır.

Google Drive üzerinde her bir havayolu firması için ayrı doküman oluşturulmuştur. Türk havayolu firmalarının sosyal medya hesaplarının kullanıcı adları ve belirli etiketler baz alınarak günlük Twitter veri çekme işlemi zamanlanmıştır. Kullanıcı adları; "pegasusdestek", "ucurbenipegasus", "AJ_Destek", "anadolujet", "OnurAir", "SunExpress", "TK_TR" şeklindedir. Etiketler; "#anadolujet","#ucmayankalmasın", "#pegasushavayolları", "#flypgs", "#sunexpress", "#TürkHavaYolları", "#thy", "#thydishat", "#thyucuslari", "#thyichat" olarak belirlenmiştir.

3.2. Veri Setinin Etiketlenmesi

Türk havayolu firmalarının web sitelerinde yer alan sık sorulan sorular toplanarak ortak başlıklardan sınıflandırma başlıkları belirlenmiştir ve toplanan Twitter verileri bu doğrultuda etiketlenmiştir. Havayolu sektörü ile ilgisi olmayan tweetler silinmiştir, belirlenen kategorilere uymayan paylaşımlar "Diğer İşlemler" başlığı altında değerlendirilmiştir.

Çalışma kapsamında 11 adet etiket başlığı belirlenmiştir. Belirlenen başlıklar; "Kampanyalar", "Diğer İşlemler", "Bagaj/ Eşya/ Diğer Taşıyabileceklerim", "Bilet İşlemleri", "Sefer İptalleri", "Sefer Değişiklik ve Gecikmeleri", "İade İşlemleri", "Uçuş ve Uçak İçi", "Seyahat Planlama" şeklindedir. Etiketli veri dağılımı Tablo 3.1'de gösterilmiştir.

Tablo 3. 1. Twitter veri setindeki her bir sınıfta yer alan tweet sayısı

Etiket	**Tweet Sayısı**
Kampanyalar	1048
İade İşlemleri	1606
Diğer İşlemler	760
Sefer İptalleri	1375
Bilet İşlemleri	1627
Seyahat Planlama	1082
Uçuş ve Uçak İçi	1469
Kontuar ve Check-in	1014
Bilet Satış ve Destek Kanalları	1620
Sefer Değişiklik ve Gecikmeleri	1285
Bagaj/ Eşya/ Diğer Taşıyabileceklerim	1520
Toplam	14406

Verilerin yer aldığı doküman "Etiket" ve "Tweet Sayısı" sütunlarından oluşmaktadır. Veri setindeki sınıflara ait tweet örnekleri Tablo 3.2'de gösterilmiştir.

Tablo 3. 2. Veri setindeki sınıflara ait tweet örnekleri

Mesaj	**Etiket**
@flymepegasus ücretsiz uçak bileti kampanyası yaptınız, şifreyi de verdiniz ama şifreyi girince "Beklenmedik bir hata oluştu." bilgisi veriyor. Ya kampanyayı yapmayın ya da seviyenizi bu kadar düşürmeyin. Yönetim kurulunuz endüstri meslek çıkışlı sanırım.	Kampanyalar
@SunExpress pandemi nedeniyle iptal olan uçuşlara ilişkin ücret iadeleri ne zaman başlayacak? Sitenizi takip ediyorum fakat detaylı bir bilgi yok.	İade İşlemleri

Tablo 3. 3.(Devam) Veri setindeki sınıflara ait tweet örnekleri

Mesaj	Etiket
#sunexpress parayı düşündüğün kadar canımızı da düşünseydin keşke. Bir uçağı yan koltukları doldurmadan uçuramıyordunuz değil mi? Aldığınız tüm güvenlik önlemleri uçakta hiçe sayılıyor yazıklar olsun. @SunExpress	Diğer İşlemler
@anadolujet Yalancılar iptal edilen uçuşların parasını verin önce.	Sefer İptalleri
@anadolujet @AJ_Destek Batman İstanbul 18 Eylül uçuşuna bilet aldım, ödeme yapıldı fakat SMS yahut Email alamadım. Yardımcı olur musunuz?	Bilet İşlemlerim
@anadolujet neden Ankara-Adana uçuşlarını aktarmalı yaptın acaba	Seyahat Planlama
Uluslararası ucuslarda bebekli ailelere (0-2 Yas) daha rahat edecekleri bir yer veriyormusunuz? Evetbis bu yeri nasil isteyeceğiz? @pegasusdestek	Uçuş ve Uçak İçi
@flymepegasus saygisiz,ukala kontuar gorevlileri, neredeyse her ucusta rötar,ucusa yarim saat kala cocuklu aile olmamiza ragmen alinmayan ve ucaga bizim goturmemizi istenen bagaj! Daha neler neler ! Kesinlikle birdaha aslaaa	Kontuar ve Check-in
@ucurbenipegasus BKM Express ile 487,88 TL ödeme yapmış olmama rağmen siteniz geçersiz ödeme hatası vererek bilet satin alma işlemini iptal etti. Acilen bu hatanın düzeltilmesi için benimle irtibata geçilmesini istiyorum.	Bilet Satış ve Destek Kanalları
@ucurbenipegasus yine şaşırtmadın bizi #pegasus 18.04.2019 PC3308 Izmir - Istanbul ucusumuz, operasyonel nedenlerle 22:20 yerine 23:00'da gecikmeli olarak yapilacaktir. B002	Sefer Değişiklik ve Gecikmeleri
@anadolujet @AJ_Destek @THY_Teknik Uçuş sonrası kırılan yırtılan valizleri değiştirme işlemini çalışanlarınızın kendi insiyatifine mi bırakıyorsunuz? Bagaj tesliminde hasar gören valizimi çalışanlarınız değiştirmiyor!	Bagaj/ Eşya/ Diğer Taşıyabileceklerim

3.3. Veri Ön İşleme (Data Pre-Processing)

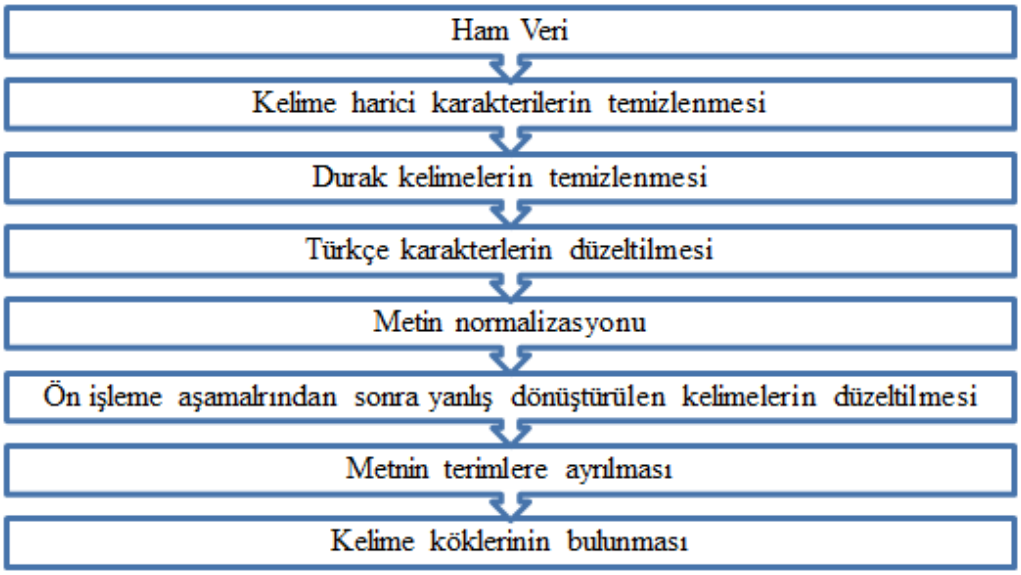
Sosyal medya verileri, kısaltmalar, hatalı yazılmış kelimeler ve günlük konuşma dilinde tercih edilmeyen sosyal medyaya özgü sözcüklerden oluşmaktadır.

Bu sebeple sosyal medya verileri üzerinde çalışmak oldukça zordur. Sosyal medya metninin sayısallaştırılmasını kolaylaştırmak için veriler üzerinde çalışmaya başlamadan önce yapay zeka ve dilbilimin alt kategorisi olan doğal dil işleme yöntemleri ile filtrelenmesi ve işlenmesi gerekmektedir.

Literatürde farklı metinler üzerinde yapılan çalışmalarda tercih edilen ön işleme metotları; metin normalizasyonu (normalization), Türkçe karakterlerin düzeltilmesi, (deasciifier), durak kelimelerin temizlenmesi (stop words), retweet ve tekrar eden mesajların elenmesi, kök alma (stemming), dizgi parçalama, kelime bölütleyici (tokenization), morfolojik çözümleyici (morphological analyzer), morfolojik belirsizlik giderici (morphological disambiguator), bağlılık ayrıştırıcısı (dependency parser) şeklindedir. [47-48].

Ön işleme çalışması kapsamında ilk olarak tüm veriler küçük harfe dönüştürülmüş ve gereksiz boşluklar temizlenmiştir.

Bu çalışmada kapsamında, elde edilen Twitter verileri üzerinde gerçekleştirilen veri ön işleme adımları sırasıyla Şekil 3.2'de gösterilmiştir.



Şekil 3. 2. Veri ön işleme aşamaları

3.3.1. Noktalama işareti, rakam, sembol, emoji, url bilgilerinin temizlenmesi

Çalışma kapsamında gerçekleştirilen çok sınıflı veri sınıflandırma işleminde duygu sınıflandırma yapılmadığı için nokta, virgül, ünlem gibi noktalama işaretleri anlam ifade etmemektedir. Bu nedenle verilerde kullanılan bütün noktalama işaretleri temizlenmiştir.

Verilerde kullanılan sayılar, kullanıcı adlarını belirtmek için kullanılan @ sembolü, etiketlemek için kullanılan # sembolü, emojiler ve URL bilgiler de veriler üzerinde çalışmayı zorlaştırdığı için temizlenmiştir. Temizlenmiş veriler adım adım Tablo 3.3'de gösterilmiştir. Temizleme işleminden sonra külliyatta 318.834 adet kelime bulunmaktadır.

Tablo 3. 4. Noktalama işareti, rakam, sembol, emoji, url bilgileri temizlenmiş veri

Ön İşleme Adımı	Tweet
Ham veri	@flypgs 28 gun once biletimi iptal ettim. 7 is gunu içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi. Bugün 22. iş günü. Bankayla görüşün denildi banka bize glen bişey yok DİYOR!!!!!#şikayetvar https://t.co/6K3KdJ9m28
Tüm harflerin küçük harfe dönüştürülmesi	@flypgs 28 gun once biletimi iptal ettim. 7 is gunu içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi. bugün 22. iş günü. bankayla görüşün denildi banka bize glen bişey yok diyor!!!!! #şikayetvar https://t.co/6K3KdJ9m28
Noktalama işaretlerinin, rakamların ve sembollerin temizlenmesi	@flypgs gun once biletimi iptal ettim is gunu içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi bugün iş günü bankayla görüşün denildi banka bize glen bişey yok diyor #şikayetvar https://t.co/6K3KdJ9m28
@ ve # ile başlayan ifadelerin temizlenmesi	gun once biletimi iptal ettim is gunu içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi bugün iş günü bankayla görüşün denildi banka bize glen bişey yok diyor https://t.co/6K3KdJ9m28
Emojilerin temizlenmesi	gun once biletimi iptal ettim is gunu içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi bugün iş günü bankayla görüşün denildi banka bize glen bişey yok diyor https://t.co/6K3KdJ9m28
URL'lerin temizlenmesi	gun once biletimi iptal ettim iş günü içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi bugün is gunu bankayla görüşün denildi banka bize glen bişey yok diyor

3.3.2. Durak kelimelerin çıkarılması

Durak kelimeler, anlam ifade etmeyen, sınıflandırma işlemine katkısı olmayan, veri seti boyutunu gereksiz yere artıran, metinde sıkça geçen bağlaç, edat gibi kelimelerdir. Bu nedenle çalışma kapsamında durak kelimeler temizlenmiştir.

Durak kelime temizleme için, Python nltk kütüphanesi kullanılmıştır fakat kütüphanede var olan durak kelimeler çalışma için yetersiz kaldığı için "selam, merhaba, bile, beri, artık, kadar vb." sözlükler eklenerek liste zenginleştirilmiştir. Çalışmada Ek–A'da yer alan kelimeleri içeren bir durak kelime listesi kullanılarak durak kelime filtrelemesi uygulanmıştır. Durak kelime çıkarma işleminden sonra veriler Tablo 3.4'te gösterilmiştir.

Durak kelime çıkarma işleminden sonra külliyatta 188.540 adet kelime bulunmaktadır.

Tablo 3. 5. Durak kelimeler çıkarıldıktan sonra veri

Ön İşleme Adımı	**Tweet**
Noktalama işareti, rakam, sembol, emoji, URL bilgileri ve tekrar eden harflerden arınmış veri	gun once biletimi iptal ettim is gunu içnde işlem yaptığınız karta iade olcak denildi ama hala iade edilmedi bugün iş günü bankayla görüşün denildi banka bize geln bişey yok diyor
Durak kelimelerin çıkarılması	gun once biletimi iptal ettim is gunu içnde işlem yaptığınız karta iade olcak denildi iade edilmedi bugün iş günü bankayla görüşün denildi banka bize geln bişey diyor

3.3.3. Türkçe karakter düzeltici (deasciifier)

Durak kelimeler temizlendikten sonra Türkçe karakterleri düzeltmek amacıyla Türkçe doğal dil işleme için çevrimiçi araçlar sunan İTU NLP Web servisinin "Deasciifier" aracı kullanılmıştır [57]. Türkçe karakter düzeltme işleminden sonra veriler Tablo 3.5'te gösterilmiştir.

Tablo 3. 6. Türkçe karakterler düzeltildikten sonra veri

Ön İşleme Adımı	Tweet
Durak kelimelerin çıkarılması	gun once biletimi iptal ettim is gunu içnde işlem yaptığınız karta iade olcak denildi iade edilmedi bugün iş günü bankayla görüşün denildi banka bize geln bişey diyor
Türkçe karakterlerin düzeltilmesi	**gün önce** biletimi iptal ettim **iş günü** içnde işlem yaptığınız karta iade olcak denildi iade edilmedi bugün iş günü bankayla görüşün denildi banka bize geln bişey diyor

3.3.4. Metin normalizasyonu (normalizer)

Türkçe karakter düzeltme aşamasından sonra tweetler üzerinde metin normalizasyonu işlemi gerçekleştirilmiştir. Normalizasyon işlemi için İTÜ NLP kütüphanesinden faydalanılmıştır [57]. Metin normaizasyonu işleminden sonra veriler Tablo 3.6'da gösterilmiştir.

Normalizasyon işleminden sonra bazı kelimeler büyük harf ile başlamıştır. Kök bulma işleminden önce bütün kelimeler tekrar küçük harfe dönüştürülmüştür.

Tablo 3. 7. Metin normalizasyonu yapıldıktan sonra veri

Ön İşleme Adımı	Tweet
Türkçe karakterlerin düzeltilmesi	gün önce biletimi iptal ettim iş günü içnde işlem yaptığınız karta iade olcak denildi iade edilmedi bugün iş günü bankayla görüşün denildi banka bize geln bişey diyor
Metin normalizasyonu	Gün önce biletimi iptal ettim iş günü **içinde** işlem yaptığınız karta idi olacak denildi idi edilmedi bugün iş günü bankayla görüşün denildi banka bize **gelen bir şey** diyor

3.3.5. Metin ön işlemeden sonra yanlış dönüştürülen kelimelerin düzeltilmesi

Normalizasyon aşamasından sonra bazı kelimelerin yanlış kelimelere dönüştürülmesi sebebiyle belirlenen kelimeler üzerinde düzeltme işlemi uygulanmıştır.

Yanlış dönüştürülen kelimeler pnr -> pınar, iade ->idi vb. şekildedir. Bu kelimeler orijinal hallerine dönüştürülmüştür.

3.3.6. Kelimelere ayırma (tokenization)

Tüm metin temizleme işlemleri tamamlandıktan sonra kelime köklerinin bulunmasına hazırlık için metinler kelimelerine ayrılmıştır.

Kelimelere ayırma işlemi için Zemberek doğal dil işleme kütüphanesindeki fonksiyonlardan yararlanılmıştır. Zemberek, Türk dilleri ve Türk dilleri arasından özellikle Türkçe için geliştirilen, platform bağımsız, açık kaynak ve genel amaçlı bir doğal dil işleme kütüphanesidir [58].

3.3.7. Kelime köklerinin bulunması (stemming)

Bir kelimeye gelen çoğul eki, fiil çekim ekleri gibi eklerin kelime köklerinden ayrılarak kelimenin en yalın eksiz hale dönüştürülmesi işlemidir.

Kelimelerin doküman içerisindeki kelime sıklıklarına bakılırken, aynı köke sahip farklı ek almış kelimelerin aynı kelime olarak kabul edilmesi için kelime köklerinin bulunması önemlidir [59].

Kelimelere ayırma işlemi gerçekleştirildikten sonra kök bulma yöntemi olarak Zemberek doğal dil işleme kütüphanesindeki fonksiyonlardan yararlanılmıştır. Zemberek, kelime analizi ve kök bulma işlemleri için kök ve ek sözlüğü kullanan sözlük tabanlı bir kök bulma yöntemi sunmaktadır [58]. Kök bulma işleminden sonra veriler Tablo 3.7'de gösterilmiştir.

Kök bulma işlemi öncesinde ve sonrasında metinlerde ayırıcı özelliği olmadığı gerekçesiyle tek karakter kalan harfler filtrelenmiştir.

Tablo 3. 8. Kelime kökleri bulunduktan sonra veri

Ön İşleme Adımı	**Tweet**
Normalizasyon uygulanmış veri	Gün önce biletimi iptal ettim iş günü içinde işlem yaptığınız karta iade olacak denildi iade edilmedi bugün iş günü bankayla görüşün denildi banka bize gelen bir şey diyor
Kelime kökleri bulunmuş veri	gün önce bilet iptal et iş gün iç işlem yap kart iade ol de iade et bugün iş gün banka gör de banka biz gel bir şey de

3.4. Öznitelik Çıkarımı (Feature Extraction)

Özellik çıkarımı, sınıflandırma işlemi için veri setinde ayırt edici özelliği yüksek, veri setini en iyi temsil eden özelliklerin seçilmesidir. Özellik çıkarımı ile mevcut özellik uzayının boyutu düşürülerek zaman ve performans kazancı sağlanmaktadır [60].

Literatürde, Bag of Words (BoW), Word2vec, Doc2vec, kelime seviyesinde N-gram ve karakter düzeyinde N-gram gibi farklı özellik çıkarım yöntemleri kullanılmaktadır.

Bu çalışmada, veri kümesini temsil etmek için tercih edilen kelime düzeyinde N-gram modelinde, 1-Gram (Unigram) ve 2-Gram (Bigram) birlikte kullanılmıştır. N-gramlar, kelimeleri toplu halde incelemeye izin verdikleri için daha fazla bilgi kazancı sağlayabilmektedirler. N-gram modelinde, bir sonraki kelimenin görülme ihtimali önceki n-1 kelimeye dayandığını varsayılmaktadır. Burada belirtilen n değişkeni, tekrar derecesini, gram ise bu tekrarın dizilim içindeki ağırlığını ifade etmektedir. Literatürde n-gram modeline, n = 1 için "unigram", n = 2 için "bigram" ve n = 3 için "trigram" ifadeleri kullanılmaktadır. Cümlelerin kelimelerine ayrılmasına örnek olarak, "bugün değiştirdiğim bilet için iki kere ücret almışsınız" cümlesi için n-gramlar Tablo 3.8'de gösterildiği gibi elde edilebilir.

Tablo 3. 9. Kelime seviye n-gram kullanımları

Unigram	"bugün", "değiştirdiğim", "bilet", "için", "iki", "kere", "ücret", "almışsınız"
Bigram	"bugün_değiştirdiğim", "bilet_için", "iki_kere", "ücret_almışsınız"
Trigram	"bugün_değiştirdiğim_bilet", "için_iki_kere", "ücret_almışsınız"

Çalışmada kullanılan örnek Unigram ve Bigram'lar Tablo 3.9'da gösterilmiştir.

Tablo 3. 10. Çalışmada kullanılan unigram ve bigram örnekleri

Bagaj/ Eşya/ Diğer Taşıyabileceklerim	
Unigram	Bigram
bagaj	bebek araba
valiz	teslim et

Tablo 3. 11.(Devam) Çalışmada kullanılan unigram ve bigram örnekleri

Bilet Satış ve Destek Kanalları	
Unigram ulaş telefon	Bigram yaş altı seyahat danışman
Bilet İşlemleri	
Unigram bilet flex	Bigram bilet pnr bilet al
Kampanyalar	
Unigram kampanya indir	Bigram bedava bilet yurtdışı bekle
Kontuar ve Check-in	
Unigram check checkin	Bigram check in checkin yap
Sefer Değişiklik ve Gecikmeleri	
Unigram rötar saat	Bigram rötar yap uçak saat
Sefer İptalleri	
Unigram iptal otel	Bigram uçuş iptal iptal ol
Seyahat Planlama	
Unigram sefer uç	Bigram uç zaman uç yok
Uçuş ve Uçak İçi	
Unigram koltuk mesafe	Bigram sosyal mesafe al koltuk
İade İşlemleri	
Unigram iade para	Bigram iş günü bilet ücret
Diğer	
Unigram ol teşekkür	Bigram giriş yap sabiha gökçen

3.5. Terim Ağırlıklandırma (Term Weighting)

En yaygın bilgiye erişme yöntemlerinden biri Terim Sıklığı (Term Frequency – TF) ve Ters Doküman Sıklığı'dır (Inverse Document Frequency) [61]. TF-IDF, bir veri setinde yer alan cümle içerisindeki herhangi bir terimin önemini yansıtmayı sağlayan özellik vektörizasyon yöntemidir. TF-IDF özellik vektörizasyon yöntemi ile bir kelimenin veri seti içerisindeki değeri ölçülmektedir. TF-IDF, bir anahtar kelimenin tüm veri seti içerisinde geçiş sıklığının kıyaslanmasına imkan sağlar. TF-IDF

yönteminde, cümlelerde bulunan kelimelerin ne sıklıkla kullanıldığı, diğer dokümanlarda geçme sıklıkları birlikte hesaplanarak, kategoriler için en önemli kelimeler belirlenmektedir.

Terim Sıklığı, bir kelimenin dokümanda bulunma sıklığını yani kelimenin dokümanda kaç kez geçtiğini ifade etmektedir. Bir terim, uzun belgelerde kısa belgelere kıyasla çok daha fazla sayıda görülebilmektedir. Bu bağlamda, normalleştirmeyi sağlayabilmek amacıyla genellikle terim frekansı belgedeki toplam terim sayısına bölünür. Terim sıklığının eşitliği Denklem (3.1)'de gösterilmiştir [62].

$$TF(t)=\frac{\text{t terimin belgede geçme sayısı}}{\text{dokümandaki toplam terim sayısı}} \tag{3.1}$$

Ters Doküman Sıklığı, dokümanda yer alan herhangi bir kelimenin dokümandaki önemini ölçmektedir. Ters Doküman Sıklığı Denklem (3.2)'de gösterilmiştir [62]. Ters Doküman Sıklığı hesaplanırken her terimin önem derecesi eşit olarak alınır. Bununla birlikte bazı terimlerin dokümandaki önemi düşük olmasına rağmen çok daha fazla görülebildiği bilinmektedir. Her dilde bu tarz kelimeler mevcuttur ve tek başlarına bir anlam ifade etmezler. Türkçe için "ve", "ama" ve "neden" gibi durak kelimeler örnek verilebilir. Bu nedenle metin ön işleme adımında bu kelimeler temizlenmektedir.

$$IDF(t)=\log_2\left(\frac{\text{Toplam doküman sayısı}}{\text{t terimini içeren doküman sayısı}}\right) \tag{3.2}$$

TF-IDF yöntemi, düşük hesaplama maliyeti ve kolay uygulanabilirliği açısından metin madenciliğinde ve sınıflandırma problemlerinde tercih edilen yöntemlerden biridir.

Bu çalışmada, veri seti üzerinde özellik vektörlerinin oluşturulurken, veri setinde çok sık geçen terimlerin etkisinin azaltılması için terim frekansının ters doküman frekansı ile ağırlıklandırılmış şekli TF-IDF vektörizasyonu kullanılmıştır. N-gramlar ile elde edilen kelimeler (Unigram ve Bigram) sıralı olarak elde edildikten sonra veri setindeki n-gram kelimelerinin ağırlıkları TF-IDF ile hesaplanarak özellik olarak kullanılmıştır.

Veri seti üzerinde TF-IDF ve Unigram – Bigram'ın birlikte kullanılması sonucu 16.773 adet özellik elde edilmiştir.

Bu çalışmada, Scikit-learn kütüphanesinin özellik çıkarım yöntemlerinden TF-IDF yöntemini gerçekleştiren "TfidfVectorizer" fonksiyonu kullanılmıştır (Bk. Tablo 3.10).

TfidfVectorizer içerisinde kullanılan parametre değerleri şu şekildedir;

- Sublinear_df: Frekans için logaritmik form kullanabilmek amacıyla sublinear_df parametresi true olarak ayarlanmıştır.
- Min-df: Özellik vektörü oluşturulurken terimlerin dokümanda geçme sıklığı belirtilen eşik değerden küçük terimleri göz ardı etmektedir. Min_df parametresi 3 olarak ayarlanmıştır.
- Ngram_range: (1,2) ayarlanmıştır.

Tablo 3. 12. Terim ağırlıklandırma yöntemi ve scikit-learn fonksiyonu

Terim Ağırlıklandırma Yöntemi	Scikit-learn Fonksiyonu
TF-IDF	from sklearn.feature_extraction.text import TfidfVectorizer

3.6. Özellik Seçimi (Feature Selection)

Özellik seçimi, veriye ait özelliklerden, veri kümesinin veya sınıfının değerlerini belirleyen özelliklerin belirlenmesi işlemidir [63]. Özellik seçimi, arama uzayını küçülterek, sınıflama işleminin başarısını artırmaktadır.

Bu çalışmada, özellik seçimi için Ki kare (Chi-square) istatistiği, Bilgi Kazancı (Information Gain) ve Varyans Analizi (Analysis of Variance) yöntemleri kullanılmıştır.

Ki kare, bir özellik ve sınıf arası bağımsızlığın derecesini ölçen bir istatistiktir. Düşük hesaplama maliyeti ve kolay uygulanabilir olmasından dolayı özellik çıkarımı için tercih edilmiştir.

Bilgi kazancı hesaplanırken alt bölümlere bölünmeden önce entropi bulunur, daha sonra tüm alt bölümlerin entropisi bulunarak iki değer arasındaki farkın büyük olduğu değişken en iyi kriter olarak seçilir. Sık tercih edilen özellik çıkarma yöntemlerinden biri olan bilgi kazancı yöntemi bu çalışmada tercih edilmiştir.

Varyans analizi, 3 veya daha fazla grup arasında, belirli bir değişkene bağlı olarak farklılık olup olmadığını belirlemek amacıyla kullanılmaktadır. Yani kategorik bir değişken ile sayısal bir değişken arasındaki ilişkiyi ölçmek için kullanılmaktadır. Çok sınıflı sınıflandırma problemindeki başarısı nedeniyle bu çalışmada tercih edilmiştir [64].

Bu çalışmada, özellik seçim yöntemlerini uygulayabilmek için Scikit-learn kütüphanesinden Tablo 3.11'de gösterilen fonksiyonlar kullanılmıştır. Özellik ve sınıf arasındaki ilişki ölçüldükten sonra en iyi k tane özelliği seçmek için "SelectKBest" fonksiyonu kullanılmıştır. Gerçekleştirilen çalışmada, seçilen öznitelik sayısını ifade eden k parametrik değeri "500, 1000, 2000 ve 4000" olarak belirlenerek denemeler yapılmıştır.

Tablo 3. 13. Özellik seçim yöntemi ve scikit-learn fonksiyonu

Özellik Seçim Algoritması	**Scikit-learn Kütüphanesi Fonksiyonu**
Chi2	sklearn.feature_selection.chi2
IG	sklearn.feature_selection.mutual_info_classif
ANOVA	sklearn.feature_selection.f_classif

3.7. Veri Setinin Eğitim ve Test Verisi Olarak Ayrılması

Makine öğrenmesi yöntemlerinde sınıflandırma modellerini yorumlayabilmek için veri seti eğitim ve test verisi olmak üzere iki gruba ayrılmaktadır. Veri seti ayırma işlemi için literatürde birçok yöntem mevcuttur. Bu çalışmada 14.406 adet tweet içeren veri seti %80 eğitim %20 test olarak ayrılmıştır.

3.8. Sınıflandırma

Bu tez çalışması kapsamında, Twitter'dan elde edilen yorumların, 11 başlık olarak belirlenen sınıflandırma başlıklarından hangisine ait olduğunu tespit etmek için çok sınıflı sınıflandırma çalışması gerçekleştirilmiştir. Önceki bölümlerde, metinlerin

temizlenmesi, sayısal vektörlere dönüştürülmesi ve özellik çıkarımı işlemleri detayları ile anlatılmaktadır. Bu aşamalarda, metinler sınıflandırma aşaması için kullanıma uygun hale getirilmiştir.

Bu çalışmada, birden fazla sınıf etiketi kullanıldığı için çok sınıflı sınıflandırmayı destekleyen sınıflandırma modelleri tercih edilmiştir. Sınıflandırma aşamasında, geleneksel makine öğrenmesi algoritmalarından SVM, MNB, LR, DT, RF, Adaboost, XGBoost, Bagging sınıflandırma algoritmaları, derin öğrenme algoritmalarından LSTM ve CNN algoritmaları kullanılarak bir değerlendirme yapılmıştır.

Çalışmada kullanılan sınıflandırma algoritmaları ve Sckitlearn kütüphanesi fonksiyonları Tablo 3.12'de gösterilmiştir.

Tablo 3. 14. Çalışmada kullanılan sınıflandırıcılar ve scikit-learn fonksiyonları

Sınıflandırma Algoritması	**Scikit-learn Kütüphanesi Sınıflandırıcısı**
SVM	from sklearn.svm import LinearSVC
MNB	from sklearn.naive_bayes import MultinomialNB
LR	from sklearn.linear_model import LogisticRegression
DT	from sklearn.tree import DecisionTreeClassifier
RF	from sklearn.ensemble import RandomForestClassifier
AdaBoost	from sklearn.ensemble import AdaBoostClassifier
XGBoost	import xgboost.XGBClassifier
Bagging	from sklearn.ensemble import BaggingClassifier
LSTM	from keras.models import Sequential from keras.layers import Dense, Embedding, LSTM, SpatialDropout1D

3.8.1. Doğrusal destek vektör makinesi

Çalışmada ilk olarak, geleneksel makine öğrenmesi yöntemlerinden biri olan Doğrusal Destek Vektör Makinesi ile bir değerlendirme yapılmıştır. SVM sınıflandırıcısının çalışma prensibi Bölüm 1.3.1'de detaylı olarak açıklanmıştır.

Çalışma kapsamında, Tweetler'in belirlenen etiketlere göre çok sınıflı olarak sınıflandırılması amaçlanmaktadır, bu bağlamda Doğrusal SVM sınıflandırıcısı, çok sınıflı sınıflandırma desteği, ölçeklenebilir olması ve literatürde birçok çalışmada başarılı sonuçlar vermiş olması nedeniyle tercih edilmiştir. Çok sınıflı sınıflandırma (OneVsRestClassifier) desteği şeması ile ele alınmaktadır.

Bu çalışmada, doğrusal SVM sınıflandırıcı için scikit-learn kütüphanesinde yer alan "LinearSVC" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

SVM fonksiyonu içerisinde kullanılan performansı etkileyen parametre değerleri şu şekildedir;

- C: Düzenlilik parametresi 0.5 ayarlanmıştır.
- Max_iter: 1000
- Class_weight: Her sınıf için C parametresinin çarpanları {0:1.0, 1:1.0} olarak belirlenmiştir.

3.8.2. Multinomial naive bayes

Multinomal Naive Bayes sınıflandırıcı, çalışma prensini NB sınıflandırıcısına benzemektedir fakat özelliklerin ayrık olarak dağıtıldığını varsaydığı için NB sınıflandırıcısından ayrılmaktadır. MNB sınıflandırıcı da çok sınıflı veri kümesi üzerinde sınıflandırma işlemi gerçekleştirebilmektedir. Literatürde, birçok çalışmada yüksek sonuçlar elde edilmesi sebebiyle MNB algoritması çalışmada kullanılmıştır.

Çalışma kapsamında, tweetler'in belirlenen etiketlere göre çok sınıflı olarak sınıflandırılması amaçlanmaktadır, bu bağlamda genellikle farklı özelliklere sahip veriler üzerinde yüksek başarı oranları elde edilen MNB sınıflandırıcısı kullanılmıştır.

Bu çalışmada, MNB sınıflandırıcı için scikit-learn kütüphanesinde yer alan "MultinomialNB" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

MNB içerisinde fonksiyonun ön tanımlı parametre değerleri kullanılmıştır.

3.8.3. Karar ağaçları

Karar ağaçları (DT), kök düğümünden itibaren özyinelemeli olarak bölümlere ayrılır ve özellik değerlerinin temelindeki bölümlemeyi öğrenir. Bu şekilde ağaç karar verme işlemini gerçekleştirir. Kolay anlaşılır ve yorumlanabilir olması sebebiyle karar ağaları algoritması çalışmada kullanılmıştır.

Bu çalışmada, scikit-learn kütüphanesinin CART algoritmasının en güncel versiyonu ile geliştirilen "DecisionTreeClassifier" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

DT içerisinde fonksiyonun ön tanımlı parametre değerleri kullanılmıştır.

3.8.4. Rastgele orman

Rastgele Orman (RF) sınıflandırıcı, aşırı öğrenmeyi engellemek ve tahmin başarısını arttırmak için veri setinin farklı alt kümeleri üzerinde uygulanan birçok karar ağacı sınıflandırıcısının birleşimi ile oluşan bir sınıflandırıcıdır. RF sınıflandırıcı, kolektif öğrenmesi yöntemlerinden biridir.

Bu çalışmada, RF sınıflandırıcı için scikit-learn kütüphanesinde yer alan "RandomForestClassifier" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

RF fonksiyonu içerisinde kullanılan performansı etkileyen parametre değerleri şu şekildedir;

- N_estimators: Temel sınıflandırıcı sayısı (karar ağacı sayısı) 1000 olarak ayarlanmıştır.
- Criterion: Bilgi kazancının kullanılabilmesi için "entropy" olarak ayarlanmıştır.

3.8.5. Lojistik regresyon

Lojistik Regresyon (LR) sınıflandırıcı, regresyondan ziyade çoğunlukla ikili sınıflandırma için kullanılan doğrusal bir modeldir. LR, olasılıkları tahmin ederek bazı bağımsız değişkenler ile bağımlı bir değişken arasındaki ilişkiyi ölçer.

Bu çalışmada, LR sınıflandırıcı için scikit-learn kütüphanesinde yer alan "LogisticRegression" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

LR fonksiyonu içerisinde kullanılan performansı etkileyen parametre değerleri şu şekildedir;

- Multi_class: Çok sınıflı metin sınıflandırma çalışması gerçekleştirildiği için multi_class değeri "multinomial" olarak ayarlanmıştır. Bu parametrede her bir sınıfın tahmin edilen olasılığını bulmak için softmax fonksiyonu kullanılmaktadır.
- Solver: "lbfgs" olarak ayarlanmıştır.

3.8.6. Adaptive boosting

AdaBoost sınıflandırıcı, veri seti üzerinde ilk olarak temel sınıflandırıcıyı uygular daha sonra hatalı sınıflandırılan veri setindeki örneklerin ağırlıklarını değiştirir ve yeniden ağırlıklandırılmış veri seti üzerinde temel sınıflandırıcıyı tekrar uygular. Kolektif öğrenme Boosting yöntemlerinden en popüler yöntem olduğu için çalışmada tercih edilmiştir.

Bu çalışmada, AdaBoost sınıflandırıcı için scikit-learn kütüphanesinde yer alan "AdaBoostClassifier" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

Adaboost fonksiyonu içerisinde kullanılan performansı etkileyen parametre değerleri şu şekildedir;

- N_estimators: Temel sınıflandırıcı sayısı (karar ağacı sayısı) 700 olarak ayarlanmıştır.
- Learning_rate: Öğrenme hızı 0.1olarak ayarlanmıştır.
- Base_estimator: Default "None" olarak ayarlanmıştır. Bu şekilde temel sınıflandırıcı olarak DT kullanılmış olur.

3.8.7. Extreme gradient boosting

XGBoost sınıflandırıcı, diğer kolektif öğrenme modellerine kıyasla daha hızlıdır ve çoğu zaman diğer kolektif öğrenme yöntemlerine göre daha başarılı sonuçlar üretmektedir. Optimize edilecek birçok parametre barındırması modelin çalışmada tercih edilme nedenlerinden biridir.

Bu çalışmada, XGBoost sınıflandırıcı için xgboost kütüphanesinde yer alan "XGBClassifier" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

XGBoost fonksiyonu içerisinde kullanılan performansı etkileyen parametre değerleri şu şekildedir;

- Max_depth: Karar ağaçlarının maksimum derinliği yani her bir karar ağacında maksimum sayıda kullanılacak özellik sayısı 5 olarak ayarlanmıştır.
- Learning_rate: Öğrenme hızı 0.5 olarak ayarlanmıştır.
- N_estimators: Temel sınıflandırıcı sayısı (karar ağacı sayısı) 500 olarak ayarlanmıştır.
- Objective: Çok sınıflı sınıflandırma probleminde öğrenme için seçilen amaç fonksiyonu 'multi:softmax' olarak seçilmiştir.

3.8.8. Bootstrap aggregation

Bootstrap Aggregation (Bagging) sınıflandırıcı ile mevcut eğitim setininalt kümelerinden yeni eğitim setleri türetilerek yeni ağaçlar oluşturulur. Oluşturulan ağaçlar ile temel öğreticinin yeniden eğitilmesi sağlanır. Bagging sınıflandırıcıda amaç türetilen yeni veri setleri ile farklılıkları oluşturmak ve bu sayede toplam sınıflandırma başarısını artırmaktır. Bagging sınıflandırıcılar, modeldeki yüksek varyansı düşürmeye yardımcı olur. Kolektif öğrenme yöntemleri arasında popüler bir yöntem olduğu için çalışmada tercih edilmiştir.

Bu çalışmada, Bagging sınıflandırıcı için scikit-learn kütüphanesinde yer alan "BaggingClassifier" fonksiyonu kullanılmıştır (Bk. Tablo 3.12).

BaggingClassifier fonksiyonu içerisinde kullanılan performansı etkileyen parametre değerleri şu şekildedir;

- N_estimators: Temel sınıflandırıcı sayısı (karar ağacı sayısı) 500 olarak ayarlanmıştır.
- Base_estimator: Default "None" olarak ayarlanmıştır. Bu şekilde temel sınıflandırıcı olarak DT kullanılmış olur.

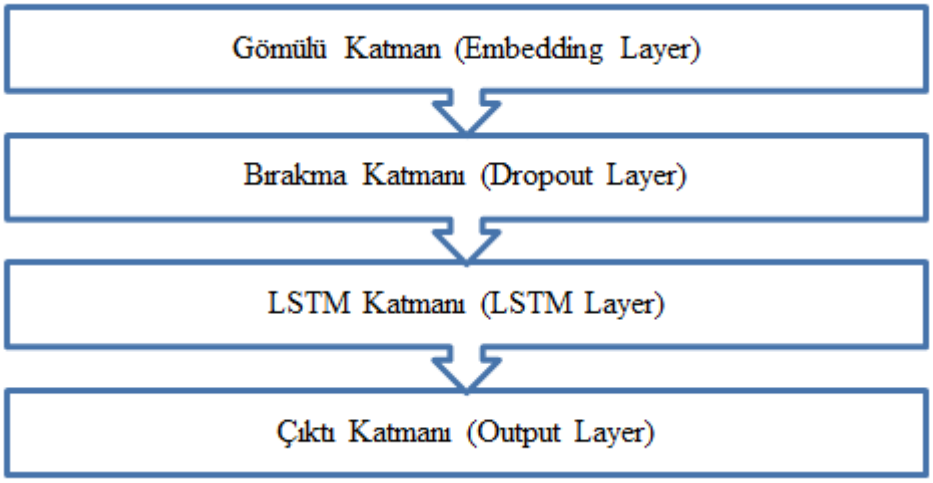
3.8.9. Uzun kısa süreli hafıza ağları

Çalışmada kullanılan bir diğer sınıflandırma yöntemi ise geleneksel makine öğrenmesi yöntemlerinden farklı olarak derin öğrenme yöntemleri arasında yer alan LSTM'dir. LSTM sınıflandırıcı modeli oluşturulurken geleneksel makine öğrenmesi yöntemlerinde kullanılan veri seti ile çalışmalar gerçekleştirilmiştir.

Kullanılan veri setinde toplam 14406 adet etiketli tweet mevcuttur. % 80'lik bir kısmı eğitim için kullanılırken, yaklaşık %20'lik kısmı test için kullanılmıştır. LSTM uygulaması, Python dilinde, Tensorflow arka planında Keras derin öğrenme kütüphanesi kullanarak gerçekleştirilmiştir.

Çalışmada, en sık kullanılan kelime sayısı 4.000 olarak ve her tweet'deki maksimum kelime sayısı 20 olarak sınırlandırılmıştır. Maksimum kelime sınırından büyük olan yorumların devamı göz ardı edilmiştir. 11.136 eşsiz öznitelik elde edilmiştir.

Birçok LSTM konfigürasyonu ile gerçekleştirilen deneyler neticesinde üç katmanlı LSTM ile en iyi sonuca ulaşılmıştır. İlk katman gömülü katmandır. Bu katmanda, her kelime 50 uzunluk vektörü ile temsil edilmektedir. Modelin aşırı öğrenmesini (overfitting) engellemek için SpatialDropout1D 0.2 olarak belirlenmiştir. Sonraki katman, 100 saklı bellek birimine sahip olan LSTM katmanıdır. En son katman çıktı katmanı, bu katmanda aktivasyon fonksiyonu olarak softmax, sigmoid ve softplus kullanılarak denemeler yapılmıştır. Çıktı katmanında parametre öğrenimi için diğer yöntemlere kıyasla daha verimli olan adam algoritması, kayıp fonksiyonu olarak, çok sınıflı sınıflandırma problemine uygun olan categorical_crossentropy kullanılmıştır. LSTM model özeti Şekil 3.3'te gösterilmiştir.



Şekil 3. 3. LSTM model özeti

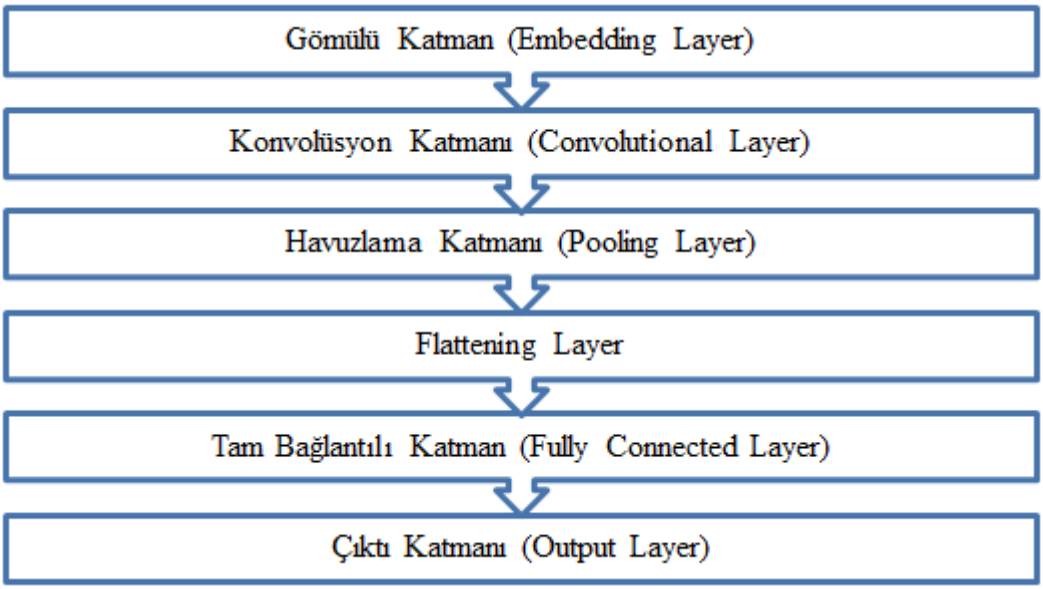
3.8.10. Konvolüsyonel sinir ağları

CNN algoritması, görüntü işleme için yaygın olarak tercih edilen derin öğrenme algoritmalarından biridir.

Kullanılan veri setinde toplam 14406 adet etiketli tweet mevcuttur. % 80'lik bir kısmı eğitim için kullanılırken, yaklaşık %20'lik kısmı test için kullanılmıştır. CNN uygulaması, Python dilinde, Tensorflow arka planında Keras derin öğrenme kütüphanesi kullanarak gerçekleştirilmiştir. İlk aşamada, TF-IDF yöntemi ile metinler sayısallaştırılmıştır. Çalışmada, maksimum özellik sayısı 20.000 olarak alınmıştır, çok sınıflı sınıflandırma için aktivasyon fonksiyonu olarak softmax ve parametre öğrenimi için Nadam algoritması kullanılmıştır. Öğrenme adımı 0.001 olarak alınmıştır.

Modelde, 1 konvolüsyon katmanı, 1 havuzlama katmanı, 3 gizli katman 1 tane de çıkış katmanı kullanılmıştır. Her bir katman toplam 64 düğüme sahiptir. Gizli katmanların her birinde, işlem yükü sigmoid ve tanjant hiperbolik fonksiyonlarına göre daha az olan "ReLU" aktivasyon fonksiyonu kullanılmıştır. Çıkış katmanı aktivasyon fonksiyonu olarak "Sigmoid, Softmax ve Softplus" ile ayrı ayrı deneyler yapılmıştır. Kayıp fonksiyonu olarak "categorical_crossentropy" kullanılmıştır.

CNN model özeti Şekil 3.4'te gösterilmiştir.



Şekil 3. 4. CNN model özeti

4. DENEYSEL ÇALIŞMALAR VE TARTIŞMA

Bu tez çalışmasında, Twitter veri seti üzerinde, Türk havayolu firmalarının web sitelerinde yer alan sık sorulan sorular başlıklarından belirlenen "Bagaj/ Eşya/ Diğer Taşıyabileceklerim", "Bilet İşlemleri", "Sefer Değişiklik ve Gecikmeleri", "Bilet Satış ve Destek Kanalları", "Seyahat Planlama", "Uçuş ve Uçak İçi", "Sefer İptalleri", "Kampanyalar", "Kontuar ve Check-in", "İade İşlemleri" ve "Diğer İşlemler" sınıflarına göre çok sınıflı sınıflandırma çalışması yapılmıştır.

Çalışmalarda veri seti %80 eğitim %20 test verileri olarak ayrılmıştır. Eğitim verileri test için kullanılmamıştır.

Eğitim ve test olarak ayrılan veriler üzerinde sınıflandırma algoritmalarının farklı parametrelere göre başarı değerleri karşılaştırılmıştır. Karşılaştırma parametreleri şunlardır; Özellik seçim yöntemi olarak belirlenen 3 farklı yöntem (Ki Kare, Bilgi Kazancı, ANOVA) ile geleneksel ve topluluk öğrenmesi yöntemleri 500, 1000, 2000, 3000 ve 4000 öznitelik ile test edilmiştir.

Bu çalışmada deneyler ilk olarak geleneksel makine öğrenmesi yöntemleri olan Doğrusal SVM, MNB, LR, DT, RF, Adaboost, XGBoost, Baaging daha sonra derin öğrenme yöntemlerinden LSTM ve CNN kullanılarak yapılmıştır.

4.1. Deneysel Sonuçlar

Twitter veri seti Tablo 3.1'de görüldüğü gibi 1520 Bagaj/ Eşya/ Diğer Taşıyabileceklerim, 1627 Bilet İşlemleri, 1285 Sefer Değişiklik ve Gecikmeleri, 1620 Bilet Satış ve Destek Kanalları, 1082 Seyahat Planlama, 1469 Uçuş ve Uçak İçi, 1375 Sefer İptalleri, 1048 Kampanyalar, 1014 Kontuar ve Check-in, 1606 İade İşlemleri, 760 Diğer İşlemler sınıflandırma etiketlerinden oluşacak şekilde toplamda 14406 örnek içermektedir.

Oluşturulan Twitter veri seti üzerinde üç farklı öznitelik ve beş farklı öznitelik sayısı ile gerçekleştirilen çalışmalar;

- Ki kare öznitelik seçim yöntemi ve Doğrusal SVM ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.1'de,
- Bilgi Kazancı seçim yöntemi ve Doğrusal SVM ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.2'de,
- ANOVA seçim yöntemi ve Doğrusal SVM ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.3'te,
- Ki kare öznitelik seçim yöntemi ve MNB ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.4'te,
- Bilgi Kazancı seçim yöntemi ve MNB ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.5'te
- ANOVA seçim yöntemi ve MNB ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.6'da,
- Ki kare öznitelik seçim yöntemi ve LR ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.7'de,
- Bilgi Kazancı seçim yöntemi ve LR ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.8'de
- ANOVA seçim yöntemi ve LR ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.9'da,
- Ki kare öznitelik seçim yöntemi ve DT ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.10'da,
- Bilgi Kazancı seçim yöntemi ve DT ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.11'de
- ANOVA seçim yöntemi ve DT ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.12'de,
- Ki kare öznitelik seçim yöntemi ve RF ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.13'de,
- Bilgi Kazancı seçim yöntemi ve RF ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.14'te
- ANOVA seçim yöntemi ve RF ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.15'de,
- Ki kare öznitelik seçim yöntemi ve Adaboosting ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.16'da,

- Bilgi Kazancı seçim yöntemi ve Adaboosting ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.17'de
- ANOVA seçim yöntemi ve Adaboosting ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.18'de,
- Ki kare öznitelik seçim yöntemi ve XGBoost ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.19'da,
- Bilgi Kazancı seçim yöntemi ve XGBoost ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.20'de
- ANOVA seçim yöntemi ve XGBoost ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.21'de,
- Ki kare öznitelik seçim yöntemi ve Bagging ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.22'de,
- Bilgi Kazancı seçim yöntemi ve Bagging ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.23'de
- ANOVA seçim yöntemi ve Bagging ile gerçekleştirilen çalışmanın başarı değerleri Tablo 4.24'te,
- LSTM sınıflandırıcısının başarı değerleri Tablo 4.25'te,
- CNN sınıflandırıcısının başarı değerleri Tablo 4.26'da

gösterilmiştir.

Tablo 4. 1. Twitter veri seti için ki kare öznitelik seçim yöntemi ile doğrusal SVM sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.73	0.70	0.27	0.75	0.73	0.74	0.07
1000	0.75	0.72	0.25	0.76	0.74	0.75	0.12
2000	0.76	0.73	0.24	0.76	0.76	0.76	0.19
3000	0.76	0.73	0.24	0.77	0.76	0.76	0.27
4000	**0.77**	**0.74**	**0.24**	**0.77**	**0.77**	**0.77**	**0.45**

Tablo 4. 2. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile doğrusal SVM sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.69	0.65	0.31	0.70	0.68	0.69	0.24
1000	0.71	0.67	0.29	0.72	0.70	0.70	0.26

Tablo 4. 2.(Devam) Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile doğrusal SVM sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
2000	0.73	0.70	0.27	0.74	0.72	0.73	0.38
3000	0.74	0.71	0.26	0.75	0.74	0.74	0.53
4000	**0.75**	**0.72**	**0.25**	**0.76**	**0.75**	**0.75**	**0.56**

Tablo 4. 3.Twitter veri seti için ANOVA öznitelik seçim yöntemi ile doğrusal SVM sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.73	0.70	0.27	0.75	0.73	0.74	0.11
1000	0.75	0.72	0.25	0.76	0.75	0.75	0.15
2000	0.76	0.73	0.24	0.76	0.76	0.76	0.18
3000	0.76	0.74	0.24	0.77	0.76	0.76	0.25
4000	**0.77**	**0.74**	**0.23**	**0.77**	**0.77**	**0.77**	**0.42**

Tablo 4. 4. Twitter veri seti için ki kare öznitelik seçim yöntemi ile MNB sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.69	0.65	0.31	0.74	0.67	0.69	0.04
1000	0.70	0.66	0.30	0.74	0.69	0.70	0.09
2000	0.71	0.68	0.29	0.75	0.70	0.71	0.18
3000	**0.72**	**0.68**	**0.28**	**0.76**	**0.71**	**0.72**	**0.29**
4000	0.72	0.68	0.28	0.76	0.71	0.72	0.38

Tablo 4. 5. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile MNB sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.64	0.59	0.36	0.71	0.62	0.64	0.05
1000	0.65	0.61	0.35	0.73	0.64	0.65	0.11
2000	0.66	0.62	0.34	0.72	0.64	0.66	0.25
3000	**0.67**	**0.63**	**0.33**	**0.73**	**0.65**	**0.67**	**0.38**
4000	0.67	0.63	0.33	0.73	0.65	0.67	0.54

Tablo 4. 6. Twitter veri seti için ANOVA öznitelik seçim yöntemi ile MNB sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.69	0.65	0.31	0.73	0.67	0.69	0.06
1000	0.70	0.67	0.30	0.74	0.69	0.70	0.11
2000	0.71	0.67	0.29	0.75	0.69	0.71	0.19
3000	**0.72**	**0.68**	**0.28**	**0.76**	**0.71**	**0.72**	**0.28**
4000	0.72	0.68	0.28	0.76	0.71	0.72	0.36

Tablo 4. 7. Twitter veri seti için ki kare öznitelik seçim yöntemi ile LR sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.72	0.69	0.28	0.74	0.71	0.72	7.79
1000	0.74	0.70	0.26	0.76	0.73	0.74	30.4
2000	0.74	0.71	0.26	0.76	0.74	0.75	45.2
3000	0.74	0.71	0.26	0.76	0.74	0.74	63.4
4000	**0.75**	**0.72**	**0.25**	**0.76**	**0.74**	**0.75**	**78.7**

Tablo 4. 8. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile LR sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.67	0.63	0.33	0.70	0.66	0.67	9.87
1000	0.69	0.66	0.31	0.72	0.68	0.69	27.40
2000	0.72	0.69	0.28	0.74	0.72	0.72	47.19
3000	0.72	0.68	0.28	0.74	0.71	0.72	64.56
4000	**0.73**	**0.69**	**0.27**	**0.75**	**0.72**	**0.73**	**77.95**

Tablo 4. 9. Twitter veri seti için ANOVA öznitelik seçim yöntemi ile LR sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.72	0.69	0.28	0.74	0.72	0.72	9.48
1000	0.74	0.70	0.26	0.75	0.73	0.74	29.38
2000	0.74	0.71	0.26	0.76	0.74	0.75	48.34
3000	0.75	0.72	0.25	0.76	0.74	0.75	58.88
4000	**0.75**	**0.72**	**0.25**	**0.77**	**0.75**	**0.75**	**79.50**

Tablo 4. 10. Twitter veri seti için ki kare öznitelik seçim yöntemi ve DT sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.65	0.61	0.35	0.66	0.65	0.65	1.00
1000	0.66	0.63	0.34	0.68	0.66	0.67	1.75
2000	0.67	0.63	0.33	0.67	0.66	0.67	2.99
3000	**0.68**	**0.64**	**0.32**	**0.68**	**0.67**	**0.68**	**4.27**
4000	0.67	0.64	0.33	0.68	0.67	0.68	5.40

Tablo 4. 11. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ve DT sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.64	0.60	0.36	0.64	0.64	0.64	1.03
1000	0.65	0.61	0.35	0.66	0.65	0.65	1.46
2000	0.66	0.62	0.34	0.66	0.66	0.66	2.37
3000	0.66	0.62	0.34	0.66	0.66	0.66	3.27
4000	**0.67**	**0.63**	**0.33**	**0.67**	**0.67**	**0.67**	**3.96**

Tablo 4. 12. Twitter veri seti için ANOVA öznitelik seçim yöntemi ve DT sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.66	0.62	0.34	0.67	0.66	0.66	1.05
1000	**0.67**	**0.63**	**0.33**	**0.67**	**0.67**	**0.67**	**1.92**
2000	0.66	0.63	0.34	0.67	0.66	0.66	3.32
3000	0.67	0.63	0.33	0.68	0.67	0.67	4.56
4000	0.67	0.63	0.33	0.67	0.67	0.67	5.35

Tablo 4. 13. Twitter veri seti için ki kare öznitelik seçim yöntemi ile RF sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.75	0.72	0.25	0.76	0.75	0.75	69.75
1000	0.76	0.73	0.24	0.77	0.76	0.76	103.23
2000	0.76	0.73	0.24	0.77	0.76	0.76	162.60
3000	0.76	0.74	0.24	0.78	0.76	0.77	227.70
4000	**0.77**	**0.74**	**0.23**	**0.78**	**0.76**	**0.77**	**266.40**

Tablo 4. 14. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile RF sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.61	0.56	0.39	0.71	0.59	0.61	3.19
1000	**0.62**	**0.58**	**0.38**	**0.69**	**0.61**	**0.63**	**2.19**
2000	0.61	0.57	0.39	0.73	0.59	0.62	2.71
3000	0.61	0.56	0.39	0.72	0.58	0.60	3.35
4000	0.60	0.55	0.40	0.72	0.57	0.59	3.25

Tablo 4. 15. Twitter veri seti için ANOVA öznitelik seçim yöntemi ile RF sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.75	0.73	0.25	0.76	0.76	0.76	71.22
1000	0.76	0.73	0.24	0.77	0.76	0.76	105.7
2000	0.76	0.73	0.24	0.77	0.76	0.76	164.9
3000	0.76	0.74	0.24	0.77	0.76	0.76	226.2
4000	**0.76**	**0.74**	**0.24**	**0.78**	**0.76**	**0.77**	**350.3**

Tablo 4. 16. Twitter veri seti için ki kare öznitelik seçim yöntemi ile oluşturulan AdaBoosting sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	**0.70**	**0.66**	**0.30**	**0.73**	**0.69**	**0.70**	**123.76**
1000	0.70	0.66	0.30	0.74	0.69	0.71	246.30
2000	0.61	0.57	0.39	0.67	0.61	0.62	799.42
3000	0.64	0.60	0.36	0.67	0.64	0.64	1132.72
4000	0.66	0.62	0.34	0.70	0.65	0.66	1481.9

Tablo 4. 17. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile oluşturulan AdaBoosting sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	**0.65**	**0.62**	**0.35**	**0.69**	**0.63**	**0.64**	**135.36**
1000	0.65	0.62	0.35	0.70	0.63	0.65	236.12
2000	0.57	0.54	0.43	0.59	0.57	0.57	452.35
3000	0.55	0.51	0.45	0.57	0.55	0.56	648.41
4000	0.55	0.51	0.45	0.57	0.55	0.55	896.67

Tablo 4. 18. Twitter veri seti için ANOVA öznitelik seçim yöntemi ile oluşturulan AdaBoosting sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	**0.70**	**0.67**	**0.30**	**0.74**	**0.70**	**0.71**	**131.7**
1000	0.70	0.66	0.30	0.73	0.69	0.70	253.0
2000	0.60	0.56	0.40	0.66	0.60	0.61	813.8
3000	0.65	0.61	0.35	0.67	0.65	0.65	1175.1
4000	0.65	0.61	0.35	0.67	0.64	0.65	1634.1

Tablo 4. 19. Twitter veri seti için ki kare öznitelik seçim yöntemi ile oluşturulan XGBoost sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.75	0.72	0.25	0.77	0.75	0.75	164.8
1000	0.76	0.73	0.24	0.77	0.76	0.76	323.0
2000	**0.76**	**0.74**	**0.24**	**0.78**	**0.76**	**0.77**	**642.4**
3000	0.76	0.73	0.24	0.77	0.76	0.77	940.3
4000	0.76	0.74	0.24	0.78	0.76	0.77	1248

Tablo 4. 20. Twitter veri seti için bilgi kazancı öznitelik seçim yöntemi ile oluşturulan XGBoost sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.71	0.68	0.29	0.73	0.70	0.71	161.0
1000	0.71	0.68	0.29	0.73	0.71	0.72	320.4
2000	0.74	0.71	0.26	0.75	0.74	0.74	630.5
3000	0.74	0.72	0.26	0.76	0.74	0.75	941.4
4000	**0.75**	**0.72**	**0.25**	**0.76**	**0.75**	**0.75**	**1242.5**

Tablo 4. 21. Twitter veri seti için ANOVA öznitelik seçim yöntemi ile oluşturulan XGBoost sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.75	0.72	0.25	0.76	0.75	0.75	161.71
1000	0.76	0.73	0.24	0.77	0.76	0.76	319.25
2000	0.76	0.74	0.24	0.78	0.76	0.77	637.22
3000	**0.77**	**0.74**	**0.23**	**0.78**	**0.77**	**0.77**	**929.75**
4000	0.77	0.74	0.23	0.78	0.76	0.77	1242.6

Tablo 4. 22. Twitter veri seti için ki kare öznitelik seçim yöntemi ile oluşturulan Bagging sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.71	0.68	0.29	0.72	0.71	0.71	294.5
1000	**0.73**	**0.70**	**0.27**	**0.74**	**0.73**	**0.73**	**547.0**
2000	0.73	0.69	0.27	0.74	0.72	0.73	947.2
3000	0.73	0.70	0.27	0.75	0.73	0.73	1332.1
4000	0.73	0.70	0.27	0.74	0.73	0.73	1653.2

Tablo 4. 23. Twitter veri seti için bilgi kazancı öznitelik seçim ile oluşturulan Bagging sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.69	0.66	0.31	0.70	0.69	0.69	245.4
1000	**0.71**	**0.67**	**0.29**	**0.72**	**0.71**	**0.71**	**594.3**
2000	0.71	0.68	0.29	0.72	0.71	0.71	965.4
3000	0.71	0.67	0.29	0.73	0.72	0.71	1445.8
4000	0.71	0.67	0.29	0.72	0.71	0.71	1946.2

Tablo 4. 24. Twitter veri seti için ANOVA öznitelik seçim yöntemi ile oluşturulan Bagging sınıflandırıcısının başarı değerleri

Öznitelik sayısı	Doğruluk	Kappa istatistiği	Kayıp	Hassasiyet	Geri çağırma	F-ölçütü	Model eğitim süresi (sn)
500	0.72	0.69	0.28	0.73	0.72	0.73	69.02
1000	**0.73**	**0.70**	**0.27**	**0.74**	**0.73**	**0.73**	**126.4**
2000	0.73	0.70	0.27	0.74	0.73	0.73	225.5
3000	0.73	0.70	0.27	0.74	0.73	0.73	282.4
4000	0.73	0.70	0.27	0.74	0.73	0.73	343.5

Tablo 4. 25. Twitter veri seti için LSTM sınıflandırıcısının başarı değerleri

Aktivasyon fonksiyonu	Eğitim doğruluk (%)	Eğitim kayıp	Test doğruluk (%)	Test kayıp	Kappa istatistiği	Hassasiyet	Geri çağırma	F-ölçütü
Sigmoid	86.27	44.11	**75.97**	0.88	0.73	0.78	0.76	0.76
Softmax	84.46	52.97	**74.30**	0.93	0.68	0.71	0.70	0.70
Softplus	84.35	52.07	**72.50**	0.98	0.69	0.70	0.70	0.70

Tablo 4. 26. Twitter veri seti için CNN sınıflandırıcısının başarı değerleri

Aktivasyon fonksiyonu	Eğitim doğruluk (%)	Eğitim kayıp	Test doğruluk (%)	Test kayıp	Kappa istatistiği	Hassasiyet	Geri çağırma	F-ölçütü
Sigmoid	91.66	0.05	**75.94**	0.12	0.72	0.76	0.75	0.75
Softmax	91.27	0.18	**74.72**	0.13	0.71	0.7	0.74	0.74
Softplus	87.36	0.22	**72.04**	0.17	0.68	0.71	0.68	0.68

Twitter veri seti üzerinde yapılan deney sonuçlarına göre tüm değerlendirme kriterleri göz önüne alındığında en başarılı sınıflandırma sonuçlarından biri, Ki kare öznitelik seçim yöntemi ile Doğrusal SVM sınıflandırıcısının seçilen en iyi 4000 öznitelik ile çalıştırılması ile elde edilmiştir. Doğrusal SVM sınıflandırıcının hata matrisi Tablo 4.27'de gösterilmiştir.

Hata matrisi başlıkları şu şekilde numaralandırılmıştır. Diğer İşlemler = 0, Sefer Değişikliği ve Gecikmeleri = 1, Bilet Satış ve Destek Kanalları = 2, Bilet İşlemleri = 3, İade İşlemleri = 4, Kontur ve Check-in = 5, Bagaj/Eşya ve Diğer Taşıyabileceklerim = 6, Seyahat Planlama = 7, Uçuş ve Uçak içi = 8, Kampanyalar = 9, Sefer İptalleri =10.

Hata matrisi ile SVM sınıflandırıcısının her bir sınıfta yer alan örnekler için sınıflandırma başarısı analiz edilebilmektedir. Hata matrisi incelendiğinde en yüksek sınıflandırma başarısı "İade İşlemleri" sınıfını ifade eden metinlerin sınıflandırılması ile elde edildiği görülmektedir. En düşük sınıflandırma başarısı ise "Diğer" sınıfını oluşturan metinlerde gözlemlenmiştir. Hiçbir sınıfa dahil olmayan metinlerin "Diğer" sınıfı altında toplanmış olması bu başlığın sınıflandırma başarısının düşmesine sebep olmuştur.

Tablo 4. 27. Twitter veri setinin ki kare öznitelik seçim yöntemi ve doğrusal SVM sınıflandırıcı ile oluşturulan hata matrisi

Gerçek Sınıflar/Tahmin Edilen Sınıflar	0	1	2	3	4	5	6	7	8	9	10
0	**56**	6	21	12	2	10	10	12	13	1	4
1	6	**203**	7	5	3	6	2	3	9	1	12
2	9	3	**189**	31	17	6	5	5	9	10	16
3	12	5	41	**209**	26	0	5	7	1	6	18
4	2	1	22	15	**298**	0	2	1	1	2	7
5	10	4	8	4	1	**138**	18	0	4	0	5

Tablo 4. 28.(Devam) Twitter veri setinin Ki kare öznitelik seçim yöntemi ve Doğrusal SVM sınıflandırıcı ile oluşturulan hata matrisi

Gerçek Sınıflar/Tahmin Edilen Sınıflar	0	1	2	3	4	5	6	7	8	9	10
6	1	3	8	5	1	4	**267**	3	5	4	2
7	2	7	10	8	2	0	2	**180**	5	5	4
8	7	12	4	5	2	7	0	1	**268**	2	2
9	3	0	6	15	1	1	0	8	0	**166**	1
10	5	14	14	13	9	4	1	6	2	2	**200**

Yorumlar

Gerçekleştirilen çalışmada, Ki kare, Bilgi Kazancı ve Varyans Analizi (ANOVA) olmak üzere 3 farklı öznitelik seçim yöntemi SVM, MNB, DT, RF ve LR olmak üzere 5 farklı geleneksel makine öğrenimi ve Adaboost, XGBoost, Bagging olmak üzere 3 faklı topluluk öğrenmesi yöntemi üzerinde farklı sayıda öznitelikler seçerek deneyler yapılmıştır.

Gerçekleştirilen deney sonuçlarına göre;

- Ki Kare ve Varyans Analizi öznitelik seçim yöntemleri ile Bilgi Kazancı öznitelik yöntemine göre daha yüksek başarı oranları elde edilmiştir.
- Neredeyse tüm modellerde aynı miktar öznitelik ile model eğitim süresi Ki Kare ve Varyans Analizinde daha düşük olurken Bilgi Kazancında da bu süre daha uzun olmuştur.
- SVM ve LR'de öznitelik sayısının artması model başarısını olumlu olarak etkilerken MNB de 3000 yerine 4000 öznitelik kullanılmasının başarı oranına bir etkisi olmamıştır.
- Seçilen öznitelik seçim yöntemi sınıflandırma başarısı üzerinde doğrudan etki sahibidir. Tablo 4. 13., Tablo 4. 14., ve Tablo 4. 15 incelendiğinde Ki Kare ve ANOVA öznitelik seçim yöntemleri model başarısını artırırken Bilgi Kazancı başarıyı görünür bir şekilde düşürmüştür.
- Öznitelik miktarının artması bütün algoritmalar için doğruluk oranını artırmadığı Tablo 4. 16., Tablo 4. 17. ve Tablo 4. 18.'de Adaboost algoritmasının sonuçlarında gösterilmiştir. Adaboost algoritması 500 öznitelik ile elde ettiği başarı oranını öznitelik miktarının artırılması ile elde edememiştir.

- Topluluk öğrenmesi yöntemlerinden XGBoost ile Adaboost ve Bagging'e göre daha yüksek başarı oranları elde edilmiştir. XGBoost algoriması ile daha fazla parametreye müdahale edebilmek model başarısını artırıcı rol oynamıştır.
- Ki Kare ve ANOVA öznitelik seçim yöntemleri ile 500 adet özniteliğin seçilmesi sonucu en yüksek (%77) doğruluk oranı XGBoost ve Random Forest ile elde edilmiştir. Düşük öznitelik ile yüksek başarı elde edilmiştir.
- Aynı miktar öznitelik kullanarak gerçekleştirilen çalışmalarda Ki Kare öznitelik seçim yöntemi ile model eğitimi daha kısa sürede gerçekleştirilmiştir.
- Bütün deneysel çalışmalar gösteriyor ki öznitelik miktarı arttıkça model eğitim süresi de paralel olarak artmaktadır.
- Tablo 4.25 ve Tablo 4.26'dan görüleceği gibi derin öğrenme yöntemlerinde kullanılan çıkış aktivasyon fonksiyonu model başarı değerlerini etkilemektedir.
- Hem LSTM hem de CNN algoritmasında en yüksek test doğruluk oranı çıkışta kullanılan Sigmoid aktivasyon fonksiyonu ile elde edilmiştir.
- Tablo 4.27'de Twitter veri setinin ki kare öznitelik seçim yöntemi ve Doğrusal SVM sınıflandırıcı ile oluşturulan hata matrisi gösterilmiştir. Burada 0 diğer kategorisini temsil etmektedir. Hem verilerin anlamsız oluşu hem de veri miktarının diğer kategorilere kıyasla az olması başarının düşük olmasına sebep olmuştur. En yüksek değer olan Bilet Satış ve Destek Kanalları kategorisi 4 numara ile temsil edilmiştir. Veri miktarının yeterli olması ve verilerin düzgün olması başarı oranına pozitif etki sağlamıştır.

5. SONUÇLAR VE ÖNERİLER

Sosyal ağlarda yapılan paylaşım sayısının artması metin sınıflandırma problemine verilen önemi artırarak yapılan çalışmaların sayısında önemli bir artış sağlamıştır. Bu tez çalışmasında, Türk havayolu firmaları hakkında yapılan Türkçe Twitter paylaşımları üzerinde, doğal dil işleme ve metin madenciliği çalışmaları gerçekleştirilerek, havayolu firmalarının web sitelerinde yer alan sık sorulan sorular başlıklarından belirlenen konu başlıklarına göre çok sınıflı sınıflandırma çalışması yapılmıştır. Çalışmada geleneksel makine öğrenmesi ve derin öğrenme teknikleri kullanılmıştır.

Twitter paylaşımlarından oluşan veri seti, yazım hataları, kısaltmalar, günlük konuşma dilinde yer almayan sosyal medyaya özgü ifadeler ve emojilerin yer aldığı kirli metinlerden oluşmaktadır. Bu paylaşımlar üzerinde yapılacak duygu analizi çalışmalarında, bu konu göz önüne alınarak uygun doğal dil işleme yöntemleri belirlenmiştir. Verinin mümkün olduğunca temizlenmesi makine öğrenmesi yöntemlerinde elde edilen başarıyı artırmıştır. Sosyal medya metinlerinin yazı diline dönüştürülmesinde kullanılan ITU NLP araçları ile düzenlenen metinler çalışma başarısının artmasında etkili olmuştur.

Makine öğrenimi yaklaşımlarında kullanılan Ki Kare, Bilgi Kazancı ve Varyans Analizi öznitelik seçim yöntemleri ile gerçekleştirilen deneysel çalışmalar ışığında öznitelik seçiminin model başarısı üzerinde önemli bir etkiye sahip olduğu söylenebilir. Yanlış öznitelik seçim yönteminin model başarısını düşürdüğü görülmüştür.

Öznitelik sayısının fazla olmasının her zaman model başarısını artırmadığı, topluluk öğrenme yöntemi olan Adaboosting algoritmasında öznitelik miktarının artmasının model başarısını düşürdüğü gözlemlenmiştir.

N-gram kullanırken, N değerinin seçiminin model başarısı üzerinde önemli olduğu N sayısının artması, kontrol sayısını ve buna bağlı olarak işlem gücünü artırdığı için

hesaplama süresinin uzadığı, N sayısı az seçildiğinde de model başarısının düşük olduğu görülmüştür.

Farklı miktarlarda veriler üzerinde yapılan deneylerde, dengeli bir veri setinde veri miktarı arttıkça başarı oranının da paralel olarak arttığı görülmüştür.

Tercih edilen denetimli öğrenme algoritmalarından, SVM, NB, LR, DT ve RF'nin sosyal medya paylaşımları üzerinden yapılan veri analizinde SVM ve RF'nin Twitter veri kümesi üzerinde uygun ön işleme, öznitelik seçme yöntemleri ve öznitelik sayılarıyla başarı oranının % 77'ye ulaştığı görülmüştür.

Son yıllarda veri miktarının artması ve donanımın gelişmesi ile denetimli, denetimsiz ve yarı denetimli yöntemleri bir arada kullanabilen derin öğrenme algoritmalarının da sınıflandırma problemlerinde, veri miktarı çok olduğu durumlarda yüksek başarı sağladığı görülmüştür. Bu nedenle bu çalışmada CNN ve LSTM derin öğrenme yöntemleri ile de deneyler yapılmıştır. Bu çalışmalarda etiketli veri miktarı derin öğrenme yöntemleri için az olsa da % 76'lara ulaşan başarı oranları elde edilmiştir. Gerçekleştirilen çalışmalarla derin öğrenme algoritmaları için kullanılan çıkış fonksiyonlarının da başarı değerleri üzerinde etkili olduğu görülmüştür.

Gerçekleştirilen çalışma ve incelenen çalışmalar ışığında, diğer dillerde yapılan çalışmalara kıyasla Türkçe metinler üzerinde yapılan doğal dil işleme çalışmalarının yetersiz olduğu, hem Türkçe hem de diğer diller için metin sınıflandırma çalışmalarının gelişime açık olduğu, özellikle derin öğrenme alanında yapılan metin sınıflandırma problemi önemli araştırma konuları haline geldiği, Türkçe metinler üzerinde derin öğrenme çalışmalarına ihtiyaç duyulduğu söylenebilir.

Gelecek çalışmalarda, topluluk öğrenmesi yöntemleri ve geleneksel makine öğrenmesi algoritmaları birlikte kullanılarak algoritma başarı oranları yükseltilebilir. Bir çok parametrik değer içeren XGBoost algoritmasının parametre değerleri iyileştirilerek daha yüksek başarı oranları elde edilebilir.

Derin öğrenme yöntemlerinde kelime gömme yöntemleri ile model başarıları iyileştirilebilir.

KAYNAKLAR

[1] Ayhan S., Erdoğmuş Ş., Destek Vektör Makineleriyle Sınıflandırma Problemlerinin Çözümü İçin Çekirdek Fonksiyonu Seçimi, *Eskişehir Osmangazi Üniversitesi İİBF Dergisi*, 2014, **9**(1), 175-198.

[2] Karagoz G. N., Yazici A., Dokeroglu T., Cosar A, A new framework of multi-objective evolutionary algorithms for feature selection and multi-label classification of video data, *International Journal of Machine Learning and Cybernetics*, 2021, **12**(1), 53-71.

[3] Samuel A. L., Some Studies in Machine Learning Using the Game of Checkers, 1959, *IBM Journal of Research and Development*, **3**(3), 210–229.

[4] Koc-san D., Kentsel Alanların WorldView-2 uydu görüntülerinden makine öğrenme algoritmaları kullanılarak tematik haritalanması, *Jeodezi ve Jeoinformasyon Dergisi*, 2013, (107) , 71-80.

[5] El Rahman S. A., Alotaibi F. A., Alshehri W. A., Sentiment Analysis of Twitter Data, *In 2019 International Conference on Computer and Information Sciences (ICCIS)*, 2019, 1–4.

[6] Çiftçi B., Apaydın M. S., A Deep Learning Approach to Sentiment Analysis in Turkish, *In 2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, 2018, 1–5.

[7] Atalay M., Çelik E., Büyük Veri Analizinde Yapay Zekâ Ve Makine Öğrenmesi Uygulamaları, *Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 2017, 9, 155–172.

[8] Witten I.H., Frank E., Hall M.A., Pal C.J, *Data Mining: Practical Machine Learning Tools and Techniques*, 578, 1, 2005.

[9] Gomes H. M., Bifet A., Read J., Barddal J. P., Enembreck F., Pfharinger B., Abdessalem T., Adaptive random forests for evolving data stream classification. *Machine Learning*, 2017, **106**(9-10), 1469-1495.

[10] Cortes C., Vapnik V., Support-vector networks, *Machine Learning*, 1995, **20**(3), 273-297.

[11] Burges C.J., A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, 1998, **2**(2), 121-167.

[12] Pang B., Lee L., Vaithyanathan S., Thumbs up? Sentiment Classification using Machine Learning Techniques, 2002, arXiv preprint cs/0205070.

[13] Eyheralendy S., Lewis D., Madigan D., On the Naive Bayes Model for Text Categorization, *In International workshop on artificial intelligence and statistics*, 2003, 93-100.

[14] McCallum A., Nigam K., A comparison of event models for naive bayes text classification, *In AAAI-98 workshop on learning for text categorization*, 1998, **752**(1), 41-48.

[15] Onan A., Twitter Mesajları Üzerinde Makine Öğrenmesi Yöntemlerine Dayalı Duygu Analizi, *Yönetim Bilişim Sistemleri Dergisi*, 2017, 3, 1–14.

[16] Bircan H., Lojistik Regresyon Analizi: Tıp Verileri Üzerine Bir Uygulama, *Kocaeli Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 2004, **2**, 185-208.

[17] Zhu, X., Text categorization with logistic regression, *University of Wisconsin-Madison*, 2007, 1-3.

[18] Hilbe J., *Logistic Regression Models*, 5th ed., Chapman & Hall/CRC, 2011.

[19] Quinlan J. R., *C4.5: programs for machine learning*, Morgan Kaufmann, California, 2014.

[20] Breiman L., Random Forests,Machine learning, *Kluwer Academic Publishers*, 2001, **45**(1), 5-32.

[21] Freund Y., Schapire R.E., Experiments with a new boosting algorithm. In: Machine Learning, *Proceedings of the Thirteenth International Conference*, 1996, 148–156.

[22] Breiman L., Bagging predictors, *Machine Learning*, 1996, **26**(2), 123-140.

[23] Onan A., Korukoğlu S., Bulut H., Ensemble of keyword extraction methods and classifiers in text classification, *Expert Systems with Applications*, 2016, 57, 232-247.

[24] Rane A., Kumar A., Sentiment Classification System of Twitter Data for US Airline Service Analysis, Sentiment classification system of twitter data for US airline service analysis. *In 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, 2018, 1, 769-773.

[25] Brownlee J., A Gentle Introduction to XGBoost for Applied Machine Learning, Machine Learning Mastery, https://machinelearningmastery.com/ gentle-introduction-xgboost-applied-machine-learning/ (Ziyaret Tarihi: 02 Şubat 2021).

[26] Polat K. K., Bayazıt, N. G., & Yıldız, O. T. Türkçe Duruş Tespit Analizi, *Avrupa Bilim ve Teknoloji Dergisi*, 2021, **23**, 99-107.

[27] Onan A., A clustering based classifier ensemble approach to corporate bankruptcy prediction, *Alphanumeric Journal*, 2018, **6**(2), 365-376.

[28] Şeker A., Diri B., Balık H. H., Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme, *Gazi Mühendislik Bilimleri Dergisi*, 2017, **3**, 47–64.

[29] Chakraborty K., Bag R., Bhattacharyya S., Relook into Sentiment Analysis performed on Indian Languages using Deep Learning, *In 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, 2018, 208–213.

[30] Chen Y., Zhang Z., Research on text sentiment analysis based on CNNs and SVM, *In 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2018, 2731–2734.

[31] Yüksek A. G., Arslan H., ve Kaynar O. Comparison of the effects dimensionalty methods in the training of neuro-fuzzy (ANFIS) classifications, *In 2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, 2017, 1-9.

[32] Hochreiter S., Schmidhuber J., Long Short Term Memory, *Neural Computation*, 1997, **9**(8), 1735–1780.

[33] Ayata D., Saraçlar M., Özgür A. Political opinion/sentiment prediction via long short term memory recurrent neural networks on Twitter, *In 2017 25th Signal Processing and Communications Applications Conference (SIU)*, 2017, 1-4.

[34] Pragya A., Piyush G., Harsh G., Navnith R., Emotion analysis using Word Embedding and Neural Network, Github, https://github.com/Harsh24893/EmotionRecognition/blob/master/rep (Ziyaret Tarihi: 08 Kasım 2020).

[35] Gazel S. E. R., Bati, C. T., Derin Sinir Ağları ile En iyi Modelin Belirlenmesi: Mantar Verileri Üzerine Keras Uygulaması, *Yüzüncü Yıl Üniversitesi Tarım Bilimleri Dergisi*, 2019, **29**(3), 406-417.

[36] Nwankpa C., Ijomah W., Gachagan A., Marshall S. Activation functions: Comparison of trends in practice and research for deep learning, *arXiv preprint arXiv:1811.03378*, 2018.

[37] Nicholls C., Song F., Comparison of feature selection methods for sentiment analysis, *In: Proceedings of the 23rd Canadian conference on Advances in Artificial Intelligence (AI'10)*, 2010, 286-289.

[38] Peng H., Long F., Ding C., Feature selection based on mutual information: criteria of max dependency, max relevance, and min-redundancy, *IEEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, **27**(8), 1226-1238.

[39] Gündüz H., Borsa İstanbul (BIST) 100 Endeksi Yönünün Ekonomi Haberleri İle Tahmin Edilmesi, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2013, 335700.

[40] Guyon I., Elisseeff A., An introduction to variable and feature selection, *Journal of machine learning research*, 2003, 1157–1182.

[41] Hossin M., Sulaiman M. N., A review on evaluation metrics for data classification evaluations, *International Journal of Data Mining & Knowledge Management Process*, 2015, **5**(2), 1.

[42] Filiz E., Makine Öğrenmesi Yöntemleri ve Eğitim Verisi Üzerine Bir Uygulama: Uluslararası Matematik ve Fen Eğilimleri Araştırması 2015 Türkiye Örneği, Doktora Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2019, 598315.

[43] Sokolova M., Lapalme G., A systematic analysis of performance measures for classification tasks, *Information processing & management*, 2009, **45**(4), 427-437.

[44] Donner A., Klar N., The statistical analysis of kappa statistics in multiple samples, *Journal of clinical epidemiology*, 1996, **49**(9), 1053-1058.

[45] Rabbimov I. M., Kobilov S. S., Multi-Class Text Classification of Uzbek News Articles using Machine Learning, *In Journal of Physics: Conference Series*, 2020, **1546**(1), 012097.

[46] Osmanoğlu U. Ö., Atak O. N., Çağlar K., Kayhan H., Can, T., Sentiment Analysis for Distance Education Course Materials: A Machine Learning Approach, *Journal of Educational Technology and Online Learning*, 2020, **3**(1), 31–48.

[47] Jabreel M., Moreno A., A Deep Learning-Based Approach for Multi- Label Emotion Classification in Tweets, *Applied Sciences*, 2019, **9**(6), 1123.

[48] Gürcan F., Multi-Class Classification of Turkish Texts with Machine Learning Algorithms, *In 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 2018, 1-5.

[49] Chen B., Huang Q., Chen Y., Cheng L., Chen R., Deep neural networks for multi-class sentiment classification, *IEEE 4th International Conference on Data Science and Systems*, 2018, 854-859.

[50] Franko S., Parlak I. B., A comparative approach for multiclass text analysis. *In 2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, 2018, 1-6.

[51] Quispe O., Ocsa A., Coronado R., Latent semantic indexing and convolutional neural network for multi-label and multi-class text classification, *In 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, 2017, 1-6.

[52] Bilgin M., Şentürk İ. F., Danışmanlı ve yarı danışmanlı öğrenme kullanarak doküman vektörleri tabanlı tweetlerin duygu analizi, *Balıkesir Üniversitesi Fen Bilim. Enstitüsü Dergisi*, 2019, **21**, 822–839.

[53] Yılmaz S., Topluluk Öğrenme Yöntemini Kullanarak Twitter Verisi Üzerinde Duygu Algılama ve Tanımlama, Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir, 2019, 579190.

[54] Rane A., Kumar A., Sentiment Classification System of Twitter Data for US Airline Service Analysis, *In 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, 2018, **1**, 769–773.

[55] Engüllü B., Twitter Sentiment Analysis, Yüksek Lisans Tezi, Bahçeşehir Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2018, 527527.

[56] Kocak B. B., Polat I., Kocak C. B., Determination of Twitter Users Sentiment Polarity Toward Airline Market in Turkey: a Case of Opinion Mining, *PressAcademia Procedia*, 2016, **2**(1), 684-691.

[57] Eryigit G., ITU Turkish NLP Web Service, *In Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, 2014, 1-4.

[58] Akın A. A., Akın M. D., Zemberek, an Open Source Nlp Framework for Turkic Languages, *Structure*, 2007, **10**(2007), 1-5.

[59] Göker H., Tekedere H., Fatih Projesine Yönelik Görüşlerin Metin Madenciliği Yöntemleri İle Otomatik Değerlendirilmesi, *Bilişim Teknolojileri Dergisi*, 2017, **10**(3), 291-299.

[60] Çoban O., Özyer B., Özyer G. T., Sentiment analysis for Turkish Twitter feeds, *In 2015 23nd Signal Processing and Communications Applications Conference (SIU)*, 2015, 2388-2391.

[61] Alupoaie S., Cunningham P., Using tf-idf as an edge weighting scheme in user-object bipartite networks, *arXiv preprint arXiv:1308.6118*, 2013.

[62] Çalış K., Gazdağı O., Yıldız, O., Reklam İçerikli Epostaların Metin Madenciliği Yöntemleri ile Otomatik Tespiti, *Bilişim Teknolojileri Dergisi*, 2013, **6**(1), 1-7.

[63] Kaynar O., Yıldız M., Görmez Y., Albayrak, A., Makine Öğrenmesi Yöntemleri ile Duygu Analizi, *In International Artificial Intelligence and Data Processing Symposium (IDAP'16)*, 2016, 234–241.

[64] Kocakafa T., Özellik Oluşumu ve Özellik Seçimi(Feature Selection)-1, Veri Bilimi Okulu, https://www.veribilimiokulu.com/ozellik-olusumu-ve-ozellik-secimifeature-selection-1/ (Ziyaret tarihi:15 Mart 2021).

EKLER

Ek-A Durak Kelimeler

acaba	ama	aslında	az	bazı	belki	biri	tk_tr
birşey	biz	bu	çok	çünkü	da	de	anadolu
defa	diye	eğer	en	gibi	hem	hep	anadolujet
her	hiç	için	ile	ise	kez	ki	nereye
mı	mu	mü	nasıl	ne	neden	nerde	pegasusdestek
kim	niçin	niye	o	sanki	şey	siz	flymepegasus
tüm	ve	veya	ya	yani	bile	beri	onurair
size	artık	kadar	via	sonra	merhaba	pegasus	thy_teknik
olan	zaten	thy	evet	böyle	şöyle	rağmen	aj_destek
birkaç	hepsi	nerede	şu	artık	bir	hala	sunexpress
daha	jet	selam	tüm	niçin	birkaç	birkez	atlasglobal

KİŞİSEL YAYIN VE ESERLER

[1] **Ekim H. E**., İnner A. B., Duygu Analizi ve Fikir Madenciliği Uygulamaları Üzerine Literatür Taraması, *Kahramanmaraş Sütçü İmam Üniversitesi Mühendislik Bilimleri Dergisi*, 2021, **24**(2), 93-114.

ÖZGEÇMİŞ

Hatice Elif Ekim ilk, orta, lise ve üniversite eğitimini Konya'da tamamladı. 2012 yılında Karatay Cemil Keleşoğlu Anadolu Lisesi'nden mezun oldu. 2017 yılında Selçuk Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği bölümünden mezun oldu. Aynı yıl Kocaeli Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği bölümünde yüksek lisans eğitimine başladı. 2018 yılından beri İstanbul'da özel bir firmada yazılım uzmanı olarak çalışmaktadır.