

ACK_ARA_20011701_20011901

by Muhammet Kayra Bulut

Submission date: 06-Jan-2023 10:23PM (UTC+0300)

Submission ID: 1989313896

File name: ACK_ARA_20011701_20011901.pdf (8.07M)

Word count: 8442

Character count: 55188

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BİR DUYGU ANALİZ YÖNTEMİ OLARAK METİNDEN
EMOJİ TAHMİNİ

20011701 – Muhammet Ali ŞEN
20011901 – Muhammet Kayra BULUT

BİLGİSAYAR PROJESİ

Danışman
Doç. Dr. Ali Can Karaca

Aralık, 2022

© Bu projenin bütün hakları Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü'ne aittir.

TEŞEKKÜR

In this study, Assoc. Prof. Dr. Ali Can Karaca supported us with his consultancy as a guide and helped us in terms of technique and motivation. Assoc. Prof. Dr. Ali Can Karaca increasing our motivation to work. We would like to thank our Yıldız technical computer engineering and especially Ali Can Karaca.

1 Muhammet Ali ŞEN

Muhammet Kayra BULUT

İÇİNDEKİLER

KISALTMA LİSTESİ	vi
ŞEKİL LİSTESİ	vii
TABLO LİSTESİ	ix
ÖZET	x
ABSTRACT	xi
1 Giriş	1
1.1 Doğal Dil İşleme	1
1.2 Doğal Dil İşleme Uygulamaları	3
1.3 Projenin Amacı	5
1.4 Veri Seti	8
1.5 Aynı Veri Setiyle Yapılmış Çalışmalar	9
2 Ön İnceleme	10
2.1 Projeye Olan İhtiyaç	10
2.2 Proje Kapsamı	10
2.3 Projenin Gereksinimleri	11
3 Fizibilite	12
3.1 Teknik Fizibilite	12
3.1.1 Yazılım Fizibilitesi	12
3.1.2 Donanım Fizibilitesi	12
3.2 İş Gücü ve Zaman Fizibilitesi	13
3.3 Ekonomik Fizibilite	13
3.4 Yasal Fizibilite	14
4 Sistem Analizi	15
4.1 Gereksinimler	15
4.2 Hedefler	16
4.3 Performans Metrikleri	17

5 Sistem Tasarımı	18
5.1 Yazılım Tasarımı	18
5.1.1 LSTM	19
5.1.2 Naive Bayes	20
5.1.3 KNN	20
5.1.4 Destek Vektör Makineleri	21
6 Uygulama	22
6.1 Modellerin Özellikleri	22
6.2 Modellerin Eğitilmesi İçin Kullanılan Yöntem	24
6.3 Modellerin Eğitilmesi	24
7 Deneysel Sonuçlar	25
7.1 Naive Bayes	25
7.1.1 Naive Bayes Nedir	25
7.2 Destek Vektör Makineleri	25
7.2.1 SVM Nedir	25
7.3 Veri Seti	26
7.3.1 Veri Setinin Boyutunu Değiştirme	26
7.3.2 Veri Setinde Oran Değiştirme	27
7.4 Sonuç	27
8 Performans Analizi	28
8.1 TF-IDF ile Counter Vectorizer Arasındaki Farklar	28
8.2 Makine Öğrenmesi	29
8.2.1 Normal Veri Kümesi Sonuçları	29
8.2.2 ROS Sonuçları	41
8.2.3 Çarpraz Doğrulama Skorları	60
8.2.4 SMOTE	61
8.3 Derin Öğrenme	73
8.3.1 Normal Veri Kümesi Sonuçları	73
8.3.2 ROS Sonuçları	75
9 Sonuç	78
9.1 ROS Sonuçları	78
9.1.1 CV	78
9.1.2 TF-IDF	79
9.2 Genel Doğruluk Oranları	81
9.3 Vektörize Yöntemleri Kiyaslama	82
Referanslar	83

KISALTMA LİSTESİ

AI	Yapay Zeka (Artificial Intelligence)
CL	Bilgisayarlı Dilbilim (Computational Linguistics)
CPU	Merkez İşlem Ünitesi (Central Processing Unit)
DDİ	Doğal Dil İşleme
DL	Derin Öğrenme (Deep Learning)
GPU	Graif İşlem Ünitesi (Graphics Processing Unit)
LSTM	Uzun Kusa Süreli Bellek (Long Short-Term Memory)
ML	Makine Öğrenmesi (Machine Learling)
NLG	Doğal Dil Üretme (Natural Language Generation)
NLP	Doğal Dil İşleme (Natural Language Processing)
NLU	Doğal Dil Anlama (Natural Language Understanding)
ROS	Rastgele Aşırı Örnekleme (Random Over Sampling)
SVM	Destek Vektör Makineleri (Support Vector Machine)
TPU	Tensor İşlem Üniteleri (Tensor Processing Units)

ŞEKİL LİSTESİ

Sekil 1.1 DDİ Aşamaları	2
Sekil 1.2 Temizlenmemiş Veri	8
Sekil 1.3 Temizlenmiş Veri	8
Sekil 2.1 Google Colab ve Python	11
Sekil 3.1 Proje İş / Zaman Çizelgesi	13
Sekil 4.1 Örnek Emoji Oranları	15
Sekil 4.2 Sekil 4.1'in Emoji Karşılıkları	16
Sekil 5.1 Metin Sınıflandırma Akış Şeması	18
Sekil 5.2 Bayes Teroemi	20
Sekil 6.1 Naive Bayes ile elde edilen başarım	23
Sekil 6.2 SVM ile elde edilen başarım	23
Sekil 8.1 Multinomial Naive Bayes Normal Sınıflandırma Raporu	29
Sekil 8.2 SVM Normal Sınıflandırma Raporu	31
Sekil 8.3 SVM TF-IDF Normal Sınıflandırma Raporu	33
Sekil 8.4 KNN Normal Sınıflandırma Raporu	35
Sekil 8.5 KNN-TFIDF Normal Sınıflandırma Raporu	37
Sekil 8.6 Multinomial Naive Bayes-TFIDF Normal Sınıflandırma Raporu	39
Sekil 8.7 ROS Veri Seti Durumu	41
Sekil 8.8 Multinomial Naive Bayes ROS Sınıflandırma Raporu	42
Sekil 8.9 Multinomial Naive Bayes-TFIDF ROS Sınıflandırma Raporu	44
Sekil 8.10 SVM ROS Sınıflandırma Raporu	46
Sekil 8.11 SVM TF-IDF ROS Sınıflandırma Raporu	48
Sekil 8.12 SVM POLY ROS Sınıflandırma Raporu	50
Sekil 8.13 SVM RBF ROS Sınıflandırma Raporu	52
Sekil 8.14 SVM SIGMOID ROS Sınıflandırma Raporu	54
Sekil 8.15 KNN ROS Sınıflandırma Raporu	56
Sekil 8.16 KNN-TFIDF ROS Sınıflandırma Raporu	58
Sekil 8.17 Multinomial Naive Bayes Çarpraz Doğrulama Skoru	60
Sekil 8.18 SVM Çarpraz Doğrulama Skoru	60
Sekil 8.19 KNN Çarpraz Doğrulama Skoru	60
Sekil 8.20 Multinomial Naive Bayes SMOTE Sınıflandırma Raporu	61

Sekil 8.21 SVM SMOTE Sınıflandırma Raporu	63
1 Şekil 8.22 KNN SMOTE Normal Sınıflandırma Raporu	65
Sekil 8.23 Multinomial Naive Bayes-TFIDF SMOTE Sınıflandırma Raporu	67
1 1 Sekil 8.24 KNN TF-IDF SMOTE Sınıflandırma Raporu	69
1 1 Sekil 8.25 SVM TF-IDF SMOTE Sınıflandırma Raporu	71
1 1 Sekil 8.26 LSTM Normal Veri Kümesi Sınıflandırma Raporu	73
1 1 Sekil 8.27 Normal Emoji-Veri Sayıları	75
1 1 Sekil 8.28 ROS Uygulanan Emoji-Veri Sayıları	76
1 1 Şekil 8.29 LSTM ROS Sınıflandırma Raporu	76

TABLO LİSTESİ

Tablo 1.1 Dijital Mecralarda Dakikalık Üretilen İçerik Sayısı	6
Tablo 3.1 Ekonomik Fizibilite Tablosu	14
Tablo 7.1 Yöntem Kiyaslama tablosu	26
Tablo 7.2 Yöntem Kiyaslama tablosu	27
Tablo 8.1 Multinomial Naive Bayes Normal Doğruluk tablosu	30
Tablo 8.2 SVM Normal Doğruluk tablosu	32
Tablo 8.3 SVM TF-IDF Normal Doğruluk tablosu	34
Tablo 8.4 KNN Normal Doğruluk tablosu	36
Tablo 8.5 KNN-TFIDF Normal Doğruluk tablosu	38
Tablo 8.6 Multinomial Naive Bayes-TFIDF Normal Doğruluk tablosu	40
Tablo 8.7 Multinomial Naive Bayes ROS Doğruluk tablosu	43
Tablo 8.8 Multinomial Naive Bayes-TFIDF ROS Doğruluk tablosu	45
Tablo 8.9 SVM ROS Doğruluk tablosu	47
Tablo 8.10 SVM TF-IDF ROS Doğruluk tablosu	49
Tablo 8.11 SVM POLY ROS Doğruluk tablosu	51
Tablo 8.12 SVM RBF ROS Doğruluk tablosu	53
Tablo 8.13 SVM SIGMOID ROS Doğruluk tablosu	55
Tablo 8.14 KNN ROS Doğruluk tablosu	57
Tablo 8.15 KNN-TFIDF ROS Doğruluk tablosu	59
Tablo 8.16 Multinomial Naive Bayes SMOTE Doğruluk tablosu	62
Tablo 8.17 SVM SMOTE Doğruluk tablosu	64
Tablo 8.18 KNN SMOTE Doğruluk tablosu	66
Tablo 8.19 Multinomial Naive Bayes-TFIDF SMOTE Doğruluk tablosu	68
Tablo 8.20 KNN TF-IDF SMOTE Doğruluk tablosu	70
Tablo 8.21 SVM TF-IDF SMOTE Doğruluk tablosu	72
Tablo 8.22 LSTM Normal Veri Kümesi Doğruluk tablosu	74
Tablo 8.23 LSTM ROS Doğruluk tablosu	77
Tablo 9.1 CV f-1 Score Sonuçları	78
Tablo 9.2 TF-IDF f-1 Score Sonuçları	80
Tablo 9.3 Accuracy Sonuçları	81
Tablo 9.4 f-1 Skor Sonuçları	82

ÖZET

Bir Duygu Analiz Yöntemi Olarak Metinden Emoji Tahmini

Muhammet Ali ŞEN

Muhammet Kayra BULUT

Bilgisayar Mühendisliği Bölümü

Bilgisayar Projesi

Danışman: Doç. Dr. Ali Can Karaca

Günümüzde doğal dil işlemenin önemi bir hayli artmaktadır. Makineye insan dilini anlatabilmek ve buna bağlı olarak insanın anlayabileceğii şekilde makinenin çıktı üretmesini sağlayabilmek bilgisayar bilimleri alanında trendler arasındadır. Artık yapay öğrenme metodlarıyla NLP sistemleri sayesinde makineler, kelimelerin anımlarının yanında cümle anlamını hatta paragraf bağlamını anlayabilecek hale geleceklerdir.

Anlatılmak istenen mesajın sadece %7 si metinler yoluyla aktarılmaktadır. İletilmek istenen mesaj en çok görsel iletişim olan jest ve mimiklerle alıcıya ulaştırılmaktadır. Bu nedenle metin mesajlarında iletimin güçlendirilmesi için emojiler kullanılmaktadır. Emojiler sayesinde metinlere duygusal kazanımı sağlanmakta ve iletilem mesajın doğruluğu artmaktadır.

Bu çalışmamız sayesinde emoji kullanılmayarak duygusu henüz tanımlanmamış metinlerin duygularını tahmin edebilecek bir sistemin, bir kaç farklı yapay öğrenme yöntemiyle karşılaşturmalarak tasarılanarak çıktılarının analiz edilmesi hedeflenmiştir.

Anahtar Kelimeler: uzun kısa süreli bellek (LSTM), Doğal Dil İşleme (NLP), Naive Bayes, makine öğrenmesi, destek vektör makineleri (SVM), derin öğrenme, emoji tahmini, yapay sinir ağları eğitimi, Keras, TensorFlow, duygusal analizi

ABSTRACT

Emoji from Text as an Emotion Prediction Method

Muhammet Ali ŞEN

Muhammet Kayra BULUT

Department of Computer Engineering

Computer Project

Advisor: Assoc. Prof. Dr. Ali Can Karaca

Today, the importance of natural language processing is increasing considerably. It is among the trends in the field of computer science to be able to explain the human language to the machine and, accordingly, to enable the machine to produce output in a way that can be understood by the human. Now, thanks to artificial learning methods and NLP systems, machines will be able to understand the meaning of sentences as well as the meaning of words and even the context of paragraphs.

Only 7% of the message to be transferred is conveyed through texts. The message to be conveyed is delivered to the receiver with gestures and facial expressions, which are mostly visual communication. For this reason, emojis are used to strengthen the transmission in text messages. With emojis, the texts gain emotion and the accuracy of the message is increased.

With this study, it is aimed to analyze the outputs of a system that can predict the emotions of texts that have not yet been defined by using emoji, by designing them in comparison with several different artificial learning methods.

Keywords:

long short-term memory (LSTM), natural language processing (NLP), Naive Bayes, machine learning, support vector machine (SVM), deep learning, emoji prediction, neural network training, Keras, TensorFlow, sentiment analysis

1

Giriş

Bu bölümde, bir duygusal analiz yöntemi olarak metinden emoji tahmini projesinin hedefleri ve kapsamı hakkında bilgiler verilecektir.

1.1 Doğal Dil İşleme

Doğal dil işleme (NLP) yapay zekanın (AI) bir koludur ve bilgisayarların, insan dilinin yapısını kavramasını ve insan dilini anlayarak çıktılar verebilmesini sağlar [1]. Doğal dil işleme, insan dilini metin ve ses olarak sorgulayabilir. Pek çok insan farkında bile olmadan doğal dil işleme sistemleriyle etkileşime geçmiştir. Örneğin, Apple-Siri, Microsoft-Cortana, Google-Assistan gibi sanal asistanların yanında chatbot sistemleri, otomatik telesekreter sistemlerinin ardından temel teknoloji doğal dil işlemedir. Bu sistemler ve asistanlara sorular sorulduğunda hem kullanıcının talebini anlayıp hem de doğal bir dille yanıt vermesini sağlayan doğal dil teknolojileridir. Doğal dil işleme hem konuşma hem de yazılı metin için geçerlidir ve tüm dillerde uygulanabilir. Doğal dil işleme destekli araçlara; web arama, istenmeyen e-posta filtreleri, otomatik metin veya konuşma çevirisi, belge özetteme, duygusal analizi ve dil bilgisi/yazım denetimi gibi araçlar örnek olarak verilebilir. Mesela, kimi e-posta programları mesaj içeriğine göre uygun yanıt tahminlerinde bulunabilir. Bu araçlar, mesajınızı okumak, anlamlandırmak ve yanıtlamak için doğal dil teknolojilerinden yararlanır.

Genel anlamda doğal dil işleme ile benzer anlamda kullanılan birkaç tane terim daha vardır.

Bunlar:

- Doğal dil oluşturma (NLG) üretme anlamına gelir.
- Doğal dil anlayışı (NLU) bilgisayarları kullanarak insan dilini anlamaya demektir.

NLG, bir durumla alakalı sözlü açıklama özelliğine sahiptir. Buna aynı zamanda "grafik

dil bilgisi" olarak bilinen bir kavram vasıtasıyla anlamlı bilgileri özütleşterek metne dönüştüren "dil çıktıları" da denir [2].

Uygulamada NLU, doğal dil işleme manasında kullanılır. Bilgisayarların, doğal dillerin yapısını ve manasını anlayarak, üreticilerin ve tüketicilerin doğal konuşma yöntemleriyle, bilgisayarlarla iletişime girmesine imkân veren anlayıştır. Bilgisayarlı dilbilimi (CL) insan dillerinin sayısal/dijital özelliklerini inceleyen bilimsel bir alandır. Doğal dil işlemeyse insan dilini anlayan, aynı dille insanlara dönüt üreten, bileşen olarak bilişim sistemlerini kullanan oluşumlar üretmekle ilgilenen bir mühendislik koludur [2].

Doğal dil işleme aşamaları olarak biçimsel analiz (morphology), sözcüksel analiz (syntax), anlam analizi (semantics), söylem analizi (discourse), pragmatik analiz (pragmatics) olarak sınıflandırılmıştır [2]. Bunlar haricinde ses hece analizi olarak fonoloji de doğal dil işlemesi olarak可以说. Biz bu çalışmamızda anlam analizi olarak metinden duygusal çıkarımı yapmaktadır.



Şekil 1.1 DDİ Aşamaları

Doğal dil işlemeyle alakalı uğraşlar 20.yy ortalarında dijital bilgisayarların bulunmasından az bir zaman geçmesinin ardından başlamıştır ve doğal dil işleme hem dilbiliminden hem de yapay zekadan (AI) yararlanır. Ancak geçtiğimiz on yılda ortaya çıkan büyük ilerlemeler sayesinde, yapay zekanın önemli bir kolu olan ve geçmiş bilgilerle eğitilen ve sentez yapan sistemler geliştirilmiştir. Bu gelişmelerin altında yatan en önemli faktörlerden biri de makine öğrenimidir. Büyük veri kümelerinden çok karmaşık örnekleri öğrenebilen makine öğrenmesinin (ML) alt dallarından biri olan DL sayesinde, doğal dil işleme teknolojileri çok gelişmiştir.

1.2 Doğal Dil İşleme Uygulamaları

Sürekli işleri otomatikleştirme:

ChatBot olarak bilinen sistemler doğal dil işlemeyle beraber kullanıldığından bugünden de gerçek müşteri temsilcilerinin gerçekleştirdiği birçok rutin misyonu tamamlayarak görevlilerin daha önemli ve zor görevlerde istihdam edilmesini sağlayabilir. Mesela, chatbot sistemleri ve sanal asistanlar birçok farklı kullanıcı isteğini anlamlandırabilir, bu istekleri uygun şekilde veri işleme sistemlerine gönderir.

Arama optimizasyonu:

Doğal dil işleme ortama göre kelime manalarını netleştirerek (mesela, "değişken" kelimesi bilişim ve IT bağlamlarda farklı anlamlarda olur), yakın anlamlı cümle ve kelimeleri ilişkilendirerek (örneğin, "gül" aramasında içinde "çicek" geçen dökümanları göstererek) ve biçimbilimsel varyasyonları dikkate alarak SSS ve belge için anahtar kelime aramalarını daha iyi hale getirebilir. Etkili doğal dil işleme destekli akademik arama algoritmik sistemleri avukatlar, doktorlar ve diğer alanında uzman kişiler için ilgili en son araştırmalara erişimi gayet yeterli seviyede optimize edebilir.

Arama motoru iyileştirmesi:

Doğal dil işleme arama sonuçlarında daha üst sıralarda görünmeniz için çok elverişli bir araçtır. Arama motorları sonuçlarını düzenlemek amacıyla doğal dil işleme yöntemlerinden faydalanan. Aynı zamanda bu yöntemleri nasıl daha efektif şekilde kullanacağını bilmek arama sonucundan üst sıralarda çıkmak açısından önem arz etmektedir. Bu vesileyle istenilen sonuçlar daha yüksek düzeyde fark edilirlikle ulaşır.

Büyük veri analizi: Belge sınıflandırma ve içerik modelleme gibi doğal dil işleme teknikleri işletme içi raporlar, haber sunumları veya bilimsel makaleler gibi büyük belge koleksiyonlarında içerik karmaşıklığını analiz etme misyonunu kolaylaştırır.

Sosyal ağ analizleri:

Doğal dil işleme kullanıcı incelemelerini ve sosyal medya içeriklerini yorumlayarak ve anlamlandırarak büyük veriler konusunda çok iyi bir anlayışa kavuştamanızı sağlayabilir. Duygu analizi, sosyal medyada yer alan içeriklerdeki başta olumlu/olumsuz içerikleri dahası diğer duyguları tespit ederek kullanıcıların isteklerinin doğrultusunda anlık doğru ölçüm gerçekleştirir. Bu, zamanla çok iyi edinimler sağlayabilir.

Pazar tahminleri:

Hedef pazarnızın dilini analiz etmek üzere doğal dil işleme'den faydalananak hedef kitlenizin taleplerini daha doğru tespit eder ve aynı zamanda hedef kitlenizle nasıl daha iyi etkileşim kurabileceğinizi anlamış olursunuz. İlgi doğrultulu duygusal analizi, sosyal mecralardaki belli başlı ilgi alanları veya ürünlerle alakalı duygu guyu (örneğin, "kulaklık muhteşem ama mikrofonu çizirtili") tespit ederek ürün modellemesi ve pazarlama için temel veriler sunar.

İçeriği yönetme:

Kurumunuz çok fazla müşteri yorumu veya mail gibi doğrudan metinsel mesajlar alıyorsa doğal dil işleme, kelimelerle birlikte metinsel mesajların da duygusunu ve hedefini de değerlendirerek içeriği düzenlemenize imkan verir.

Bunlar dışında makine çevirisi, spam filtreleme, metin özetleme, soru yanıtlama, duygusal analizi, dil tanımlama, sözcük anlam belirsizliğini giderme, intihal tespiti, sohbet robottu ve sanal asistanlarda doğal dil işlemenin uygulama alanları arasındadır [2].

1.3 Projenin Amacı

Gündelik hayatı dijital teknolojilerin yaygınlaşmasıyla birlikte dijital dönüşümler hız kazanmaktadır. Artık birçok işletme stratejilerini dijital platformlara taşımakta ve dijital platformlar aracılığıyla yapılan geri bildirimlerin önemi şirketler için çok önemli hale gelmektedir. Aynı zamanda şirketlerin dijital mecralara önem vermesiyle, dijital mecraların kullanıcı sayısı beklenmedik ölçüde artmaktadır ve kullanıcıların dijital mecraları tercih oranını artırmaya da, şirketlerin dijital mecraları diğer şirketler de tercih etmeye ve bu yönde geliştirmeler yapmaya başlamıştır. Bu şekilde beslemeli bir döngüye sahip olan dijital mecralarda kullanıcı sayısının artmasıyla ve web 2.0 teknolojilerinin güncel hayatı bir hayli dahil olmasına birlikte kullanıcılar artık internet ortamında çok rahat içerik üretebilir hale gelmiştir. Bu içerikleri her kullanıcı doğal dili ile üretmektedir. İçeriklerin yanı sıra kullanıcılar güncel hayatı kullandığı, gözlemlediği her türlü olguya internet ortamına taşımakta ve yorumlamaktadır. Bu yorumlara göre, şirketler satış ve pazarlama stratejileri geliştirmekte ve kendilerini sürekli dinamik tutmaktadır. Bu değişime ayak uydurmayan şirketler tedricen pazar paylarını kaybetmektedir. Kullanıcıların içerik üretebileceği bir platform olan Twitter'da bile dakikada yaklaşık 200.000 [3] tweet atılmaktadır. Bunu saat veya gün bazlı düşündüğümüzde bu kadar çok verinin insanlar tarafından işlenmesi, anlaşılmaması ve yorumlanması imkansız hale gelmektedir. Diğer platformlar için istatistiksel bilgiler tablo 1.1'de gösterilmiştir [3]. Bu kadar yüksek hacimli verilerin sadece metin içeren kısımları da çok yekün tutmaktadır. Doğal dil işleme yöntemleriyle bu verilerin anlaşılmaması kolaylaşmıştır. Bunun dışında bilişim sistemlerinin insan dilinde konuşulan veya yazılın içerikleri en yakın anlamıyla birebir yakın şekilde anaması ve sonuç üretmesi çok önemlidir. Bilişim sistemlerinin bunu daha doğru anaması ve dönüt vermesi için makine öğrenimi ve derin öğrenme gibi yöntemlerden faydalankmaktadır. İletişimde verilmek istenen mesajın sadece %7'si kelimelerden oluşmaktadır [4]. Bunun haricinde beden dili, ses tonları ve mimikler iletilmek istenen mesajın çoğunu oluşturmaktadır. Kisacası internet ortamında üretilen metinsel içeriklerde anlam tam anlamıyla alıcıya iletilememektedir. Metin halinde üretilen içeriklerin duygudan yoksun olması nedeniyle emojiler geliştirilmiştir. Bu sayede içerik üreticileri iletişim istediği mesajın anlamını daha anlaşırlı hale getirebilmek amacıyla emojileri sıklıkla kullanmaktadır.

Bir ürünün marka analizi, kişilerin veya grupların sosyal politika tercihi veya borsa hareketlerinin ölçülmesi gibi alanlar duyu analizi çalışmalarıyla netliğe kavuşmaktadır.

Tablo 1.1 Dijital Mecralarda Dakikalık Üretilen İçerik Sayısı

Platform	İçerik Tipi	İçerik Sayısı/Dakika
Youtube	Video/Görüntü	500 saat
Instagram	Hikaye/Görüntü	695 bin
Tinder	Tinder Kaydılması/Görüntü	2 milyon
Mail	E-posta/Metin-Ses-Görüntü	197.6 milyon
WhatsApp/Messenger	Mesaj/Metin-Ses-Görüntü	69 milyon
LinkedIn	Bağlantı	9.132
Netflix	Dizi-Film/Görüntü	28 bin kişi
TikTok	İndirme/Görüntü	5 bin
Twitch	İzlenme/Görüntü	2 milyon
Twitter	Tweet/Metin-Görüntü-Ses	200 bin

Bu çalışmamızda bir anlam analizi yöntemi olarak metinlerden duygusal çıkarımı yapılması kısacası duygusal analizi yapılması istenmiştir. Duygu analizinde amaç yazarın hedefe karşı aldığı tavırın tespiti [5]. Duygu analizi doğal dil işlemede genellikle bir sınıflandırma yöntemi olarak kullanılır. Bir metne ait duygunun tespitini çoğunlukla olumlu olumsuz veya nötr olark sınıflandırma yapan çalışmalar mevcuttur. Ayrıca bazı çalışmalarda ekstra duygusal (mutluluk, üzüntü, korku, şaşkınlık vb.) analizleri de yapılmıştır [6].

Bizim çalışmamızda da sadece olumlu/olumsuz bir sınıflandırma haricinde birçok duygunun sınıflandırması yapılması istenmiştir. Bunun için bir duygusal aktarım yöntemi olan emojilerden yararlanılmıştır. Buradaki amaç metne ait duygunun tespiti için emojiinin kullanılmasıdır. Öncelikle emoji ile duygularınılmış metinler açık kaynak ortamında temin edilmiştir [7]. Bu metinlerde sadece bir emoji ikonu olmasına kısacası yalnızca bir duygusal içermesine dikkat edilmiştir.

Elde edilen verilerde düzenli ifade (regular expression) yöntemleriyle bir dizi temizleme işlemleri yapılmıştır. Örnek olarak '#' (hashtag) '@' (mention) gibi ifadeler temizlenmeye çalışılmıştır. Ayrıca anlam içermeyen karakterler de veri setinden çıkarılmaya çalışılmıştır ve tüm veri seti küçük harf formatına alınmıştır. Veri setindeki dugu aktarımını sağlayan emojiler ise etiket (Label) olarak veri setindeki metinlerin yanına ekstra sütun açılarak işlenmiştir. Kisacası her cümlegenin bir yanındaki sütunda o cümleye ait emoji bir etiket olarak kullanılmıştır.

Bu veri setleriyle model eğitilmeye çalışılmıştır. Veri setimiz öncelikle test ve eğitim dataası olarak iki parçaya ayrılmıştır. Bunun için genel kabul görmüş oranlar baz alınmıştır (80 eğitim - 20 test). Ayrıca eğitim verisi üzerinden doğrulama verisi (validation data) bölüntülenmiştir. Ayrıca bu şekilde eğitim aşamasında çalışmanın nasıl ilerlediği tespit edilmeye çalışılmış aşırı öğrenme, ezberleme (overfit) veya yanlış öğrenme (underfit) gibi durumların olup olmadığı tespit edilmeye çalışılmıştır. Eğitim verisi eğitime toplu şekilde gönderilmemiş belirli parçalara ayrılarak (batch size) gönderilmiş bu sayede hem eğitim aşamaları incelenmiş hem de modelimiz daha hızlı eğitilmeye çalışılmıştır. Her eğitim döngüsünde (epoch) eğitim verisinin doğruluk oranı (train accuracy) ile doğrulama verisinin doğruluk oranları (val accuracy) karşılaştırılmış ve eğitim döngüsü (epoch) gerekliyse erken sonlandırılmıştır.

Öncelikle makine öğrenmesi algoritmalarında Naive Bayes ve Destek Vektör Makineleri algoritmaları yardımıyla eğitim verisi üzerinde sınıflandırma yapılmaya çalışılmıştır. Bu şekilde eğitilmiş modelmizi test verimiz üzerinde çalıştırarak doğruluk oranlarımız hesaplanmıştır. Kisacası eğitilen modelimiz test verisi üzerinde denenerek yüzde kaç oranında doğru eğitildiği hesaplanmıştır. Akabinde aynı veri setimiz bu sefer bir derin öğrenme yöntemi olan Uzun Kısa Süreli Bellek (LSTM) algoritmasıyla eğitilmiştir. Bu aşamada eğitimde yukarıda bahsedildiği gibi modelimizin iyi eğitilip eğitilmediği doğrulama veri setiyle her eğitim döngüsünde kontrol edilmiştir.

1.4 Veri Seti

Veri setinin temizlenmemiş hali Şekil 1.2'de verilmiştir.

Şekil 1.2 Temizlenmemiş Veri

```
Can't stop drinkin' about you @ Saint & Second,7  
"@OneRepublic: Ought. Ummmmm favorite season,0  
"#mickeymouse #88thbirthday #vivaorlando #waltdisneyworld Happy B-Day Mickey!!!!!! You!""I...",0  
seasons greetings @ Farmers Branch Historical Park,17  
myfavvvv @ Great Lakes Mall,13  
"we're baaaaaaack @ Fayetteville, Arkansas",6  
/// W I D E /// : @user #RVCKOR #bmtroubleyou #m3 #f80 #Conduktco @ The BMW Store,10  
"All you can say at this point is ""Ahhh"" lol #CubanCoffee #ElLibre #cafecito / Day 3: S.E U.S.",2  
Billllllliard #copper #prohibition #moonshine #comingsoon @ Moonshine Pipe Company,1  
"I loooove making new friends justice_cailey @ Gulf Shores, Alabama",9  
Ooooweeeeeeeeeeee!!!!!! #Kobe #MambaOut #KStateMBB #NikeBasketball @ Bramlage Coliseum,4  
BOOM BOOM BOOM BOOMThe coolest party in DMV on a Tuesday.,4  
B R U N E T T E #Repost @user Blow Dry: #tinatobar (haircut by me as.,1  
Andddd they're off! #SUPERHEROK_____** :@Partnr4StrngFam#BestOfGainesville...,18
```

Veri setinin temizlenmiş hali Şekil 1.3'de verilmiştir.

Şekil 1.3 Temizlenmiş Veri

TEMİZLENMİŞ	TEMİZLENMEMİŞ
stop drinkin saint amp second,7	Can't stop drinkin' about you @ Saint & Second,7
ought ummmmm favorite season,0	"@OneRepublic: Ought. Ummmmm favorite season,0
happy b day mickey,0	#mickeymouse #88thbirthday #vivaorlando #waltdisneyworld Happy B-Day Mickey!!!!!! You!""I...",0
farmers branch historical park,17	seasons greetings @ Farmers Branch Historical Park,17
myfavvvv great lakes mall,13	myfavvvv @ Great Lakes Mall,13
baaaaaaaack fayetteville arkansas,6	we're baaaaaaack @ Fayetteville, Arkansas,6
w e bmw store,10	/// W I D E /// : @user #RVCKOR #bmtroubleyou #m3 #f80 #Conduktco @ The BMW Store,10
say point ahhh lol day e u,2	All you can say at this point is "Ahhh" lol #CubanCoffee #ElLibre #cafecito / Day 3: S.E U.S.,2
billllllliard moonshine pipe company,1	Billllllliard #copper #prohibition #moonshine #comingsoon @ Moonshine Pipe Company,1
loooove making new friends justicecailey gulf shores alabama,9	I loooove making new friends justice_cailey @ Gulf Shores, Alabama,9
ooooweeeeeeee bramlage coliseum,4	Ooooweeeeeeee!!!!!! #Kobe #MambaOut #KStateMBB #NikeBasketball @ Bramlage Coliseum,4
boom boom boom boomthe coolest party dmv tuesday,4	BOOM BOOM BOOM BOOMThe coolest party in DMV on a Tuesday.,4
b r u n e e blow dry haircut,1	B R U N E T T E #Repost @user Blow Dry: #tinatobar (haircut by me as.,1
andddd,18	Andddd they're off! #SUPERHEROK_____** :@Partnr4StrngFam#BestOfGainesville...,18

Veri setine bakıldığında, aslında veriler gerçek dünya verileri olduğu için elde edilen sonuçlar pek de sağlıklı değil. Örneğin veri kümesinde "baaaaack" şeklinde bir kelime var ama "back" ile "baaaaack" kelimelerinin vektör değerleri ve tokenization sonuçları farklı değerlerde. Ama kelimeler aslında bağlamları içinde benzer anlamda kullanılmış. Benzer şey "loooooooveeee" ve "love" için de söylenebilir. Başka örneklerdeye "ough" ya da "ummmmm" gibi anlamsız kelimeler mevcut. Bu kelimelerin pek bir anlamı olmadığı için yine sonuçlara etkisi negatif olacaktır. Temizleme işlemi sonrasında Şekil 1.3'de görüldüğü gibi bazı cümlelerden geriye çok az kelime kalmıştır. Buysa anlam çıkarma işini yine zorlaştıran bir durumdur. Bunlardan dolayı, veri seti gerçek dünyanın verilerindenoluştuğu için, bu verileri ML ve DL yöntemleriyle yorumlamak oldukça meşakkatli olacaktır.

1.5 Aynı Veri Setiyle Yapılmış Çalışmalar

Proje başarımlarımızı aynı veri setini kullanan yedi tane projenin başarımlarıyla kıyasladık. Sonuçları kıyasladığımızda benzer sonuçlar alındığını gördük. Bir DL yöntemi olan LSTM'in başarısının, bir ML yöntemi olan SVM'nin başarısından daha düşük olması şaşırtıcı olsa da, tüm projelerdeki sonuçlar benzerdi. Hatta "SVM's Perform Better than RNN's at Emoji Prediction" isimli bir projeye bile rastladık.

2 Ön İnceleme

Bu bölümde, projenin ön incelemesi yapılarak projenin gidişatına yön verecek kararlar tartışılmıştır.

2.1 Projeye Olan İhtiyaç

Dijital ortamlarda üretilen metinsel ifadelerin duyguyu pek yansıtmasının nedeniyle emojiler kullanılmakta ve metne duyu verilmek istenmektedir. Ancak bu şekilde kullanılmamış sadece metinlerde ibaret yazıların olumlu/olumsuz başta olmak üzere hangi duyu içerdiğini hesaplamak oldukça zordur. Bu duyu etiketleme işlemi için genelde işletmeler kullanıcı yorumlarını okuyup tasnifleyen çalışanlar istihdam etmektedir. Duyu etiketleme işlemlerini yapay zekâ yöntemleriyle makineye öğretmek ve öğrenilen, modellenen makine üzerinden çıkarımlarda bulunma ihtiyacı doğmuştur.

2.2 Proje Kapsamı

Projemizde yapılması amaçlanan sistemden beklenenler şunlardır:

- Sistemin önce kelimeleri vektörize etmesi, sonra cümle içerisindeki dizilim sıralamasını da kullanarak cümleyivektörize etmesi, ayrıca birden fazla cümle varsa cümleleri vektörize etmesi ve sonuçta öğrenmesi.
- Sistemin çeşitli makine ve derin öğrenme algoritmalarıyla sınıflandırılmış/kümelenmiş veriler benzerlik tespit etmesi.
- Sınıflandırılmamış veya etiketlenmemiş metinlere emoji tahmininde bulunması.
- En iyi çözümün bulunabilmesi için halihazırda literatürde önerilen makine öğrenmesi ve derin öğrenmesi yöntemlerinin aynı koşullar altında karşılaştırılması.

2.3 Projenin Gereksinimleri

Projemizde istenilen sonuçların tasarlanması için daha önce etiketlenmiş (emoji kullanılmış metinsel veriler) data setlerine ihtiyaç duyulmaktadır. Bu nedenle twitter platformu üzerinden toplanmış verilere gereksinim duyulmuştur. Ayrıca bu verilerin çeşitli makine öğrenimi ve derin öğrenme yöntemleriyle modellenmesi için de araçlara, kütüphanelere (Pytorch, Tensorflow, NumPy, Pandas) ihtiyaç vardır. Bu tip kütüphanelerin kullanılabilirliğine imkan sağlayan geliştirme ortamlarına (Anaconda Navigator, Jupyter Notebook, Colab) ihtiyaç duyulmuştur.



Şekil 2.1 Google Colab ve Python

3

Fizibilite

3.1 Teknik Fizibilite

Bu bölümde projenin uygulanabilirliği ile ilgili fizibilite çalışmaları hakkında bilgiler verilmiştir.

3.1.1 Yazılım Fizibilitesi

Belirlenen algoritma doğrultusunda kullanılacak programlar da belirlendi. Yazılım araçlarının açık kaynak kodları, ürünün uygulanmasını kolaylaştırdı. Model eğitim sürecinde sistem gereksinimlerinin karşılandığı görüldü.

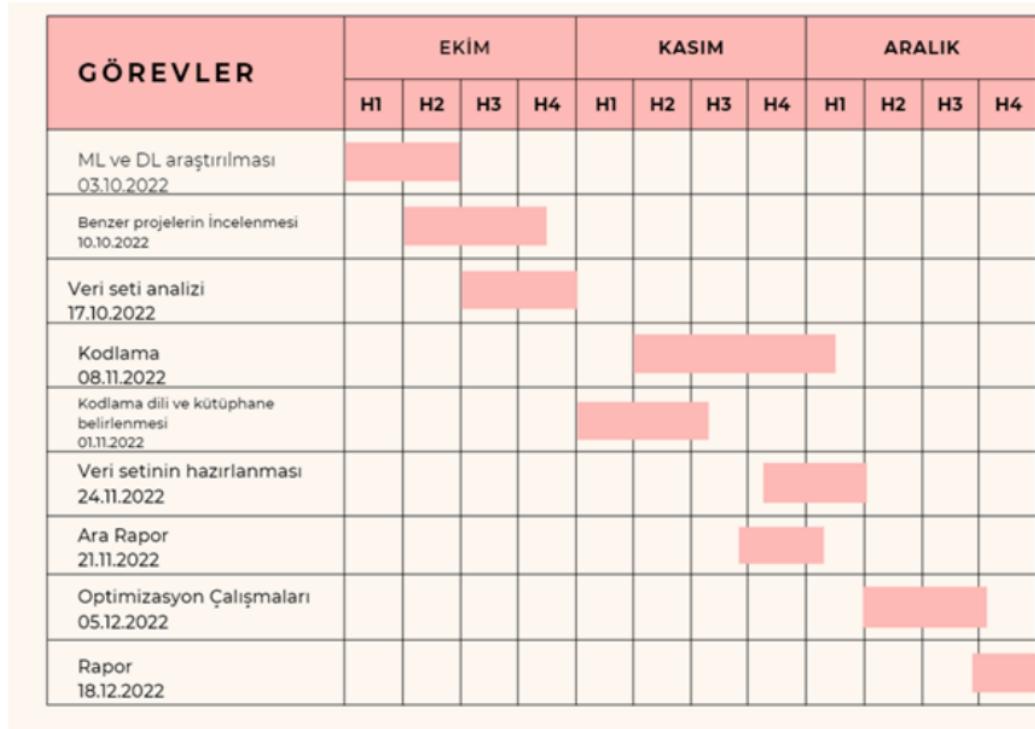
Proje Windows 10 üzerinde geliştirilecektir. Python (3.8) ve kütüphaneleriyle yazılmacaktır. Ayrıca bazı işlemler için Jupyter Notebook kullanılacaktır. Ayrıca NumPy ve Pandas gibi Python kütüphaneleri ile TensorFlow, Keras gibi AI işlemlerinde kullanılan kütüphaneler ücretsiz olarak temin edilebilmektedir.

3.1.2 Donanım Fizibilitesi

Model eğitimi çok fazla işlem gücü gerektiren bir durum olduğundan dolayı, işlem hacmi fevkalade GPU'ların kullanılması elzemdir. Tabii ki daha düşük işlem gücü hacmine sahip donanımların kullanılması da pekâlâ mümkündür, lakin görece düşük işlem gücü, eğitim için beklenen süreyi de devasa boyutlara çıkarabilmektedir. Bundan dolayı yüksek hacimli işlem yapabilen GPU ve TPU kullanılması büyük avantaj sağlayacaktır. Projede ise içinde iyi işlem hacmine sahip bir CPU ve GPU bulunduran bir bilgisayarın yanı sıra, TPU ve GPU desteği sağlayan çevrimiçi hizmetler de kullanılmıştır. Bu projede birden fazla insan emek sarfedeceğinden dolayı, yapılan işlemleri ve yazılan kodları hızlı ve efektif olarak tutabilmek için bulut sistemi kullanılmıştır. Bu sisteme kodlardaki ilerlemeler de eşzamanlı olarak tutulabilmektedir.

3.2 İş Gücü ve Zaman Fizibilitesi

2 kişi 3 ay sürede gerçekleşmiştir. Şekil 3.1'de görevlerin tamamlanması için gereken ve harcanan zaman gösterilmiştir.



Şekil 3.1 Proje İş / Zaman Çizelgesi

3.3 Ekonomik Fizibilite

Geliştirme ortamı olarak kullanılan Google Colab ve Jupyter Notebook'un ücretsiz sürümlerini kullandığımız için, geliştirme ortamı ücreti olarak bir ücret ödenmemektedir.

Derin öğrenme ve makine öğrenmesi için kullandığımız kütüphaneler de ücretsiz olduğu için, derin öğrenme ve makine öğrenmesi için kullandığımız kütüphanelere bir ücret ödenmemektedir.

Projede kullandığımız veri seti de açık kaynaklı ve ücretsizdir.

Kişisel bilgisayarlarımızın donanımı da (Acer Nitro-5) projeyi gerçekleştirmek açısından gayet yeteri seviyededir.

Detaylı maliyet tablosu Tablo 3.1'de gösterilmiştir.

Tablo 3.1 Ekonomik Fizibilite Tablosu

Araç	Adet	Fiyat	Maliyet
Apple MacBook Pro 2011	1	2000 TL	2000 TL
Google Colab	2	0 TL	0 TL
Jupyter Notebook	2	0 TL	0 TL
Acer Nitro AN515- 44	1	9500 TL	9500 TL
Toplam Maliyet			11500 TL

3.4 Yasal Fizibilte

Projede kullanılan veri seti CodaLab [7] üzerinden alınmıştır. Açık kaynaklı olduğu için, kullanımı herhangi bir yasal sorun teşkil etmemektedir.

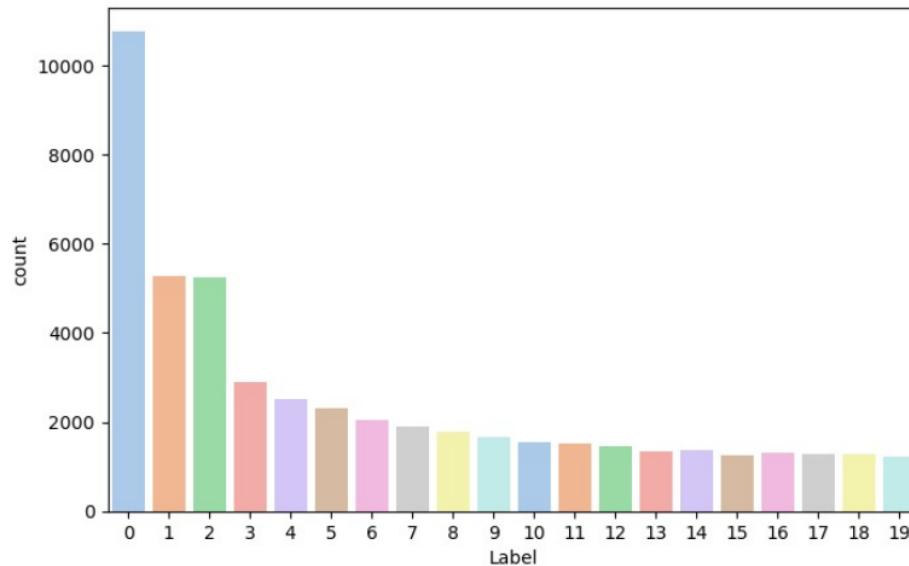
Proje herhangi bir şekilde ve durumda yasa ve yönetmelikleri ihlal edecek bir veri tutmamaktadır. Aynı zamanda tüm kullanılan kütüphane ve frameworkler açık kaynaklı ve ücretsizdir. Bundan dolayı herhangi bir patent gibi koruyucu hak da kullanılmamıştır. Projede kullanılan veri seti de yine açık kaynaklı veri setidir.

4 Sistem Analizi

Bu bölümde projenin hedefleri detaylandırılmış, gereksinim analizi yapılmış ve performans metrikleri belirlenmiştir.

4.1 Gereksinimler

Metinlerin içindeki kelimelerin, vektörel karşılıkları kullanılarak, metinlerin kelime kelime ayırtırılarak, önce kelimelerin ayrı ayrı duygusal analizi çıkarımlarıyla ve çeşitli eğitim metodlarıyla emoji'lerle ilişkilerinin bulunması. Sonrasında da kelimelerin oluşturduğu metinlerin bütüncül şekilde ele alınarak metin bazında alaka saptama. Şekil 4.1'de örnek emoji oranları gösterilmiştir. Şekil 4.2'de Şekil 4.1'in emoji karşılıkları gösterilmiştir [7].



Şekil 4.1 Örnek Emoji Oranları

0	❤️	Red heart
1	😍	Smiling face with heart eyes
2	😂	Face with tears of joy
3	💕	Two hearts
4	🔥	Fire
5	😊	Smiling face with smiling eyes
6	😎	Smiling face with sunglasses
7	✨	Sparkles
8	💙	Blue heart
9	😘	Face blowing a kiss
10	📸	Camera
11	🇺🇸	United States
12	☀️	Sun
13	💜	Purple heart
14	😉	Winking face
15	💯	Hundred points
16	🤗	Beaming face with smiling eyes
17	🎄	Christmas tree
18	📸	Camera with flash
19	😜	Winking face with tongue

Şekil 4.2 Şekil 4.1'in Emoji Karşılıkları

4.2 Hedefler

Çalışmamızda birincil hedef eğitilen modelimiz yardımıyla daha önce sınıflandırılmış yeni gelen metinlere en uygun emojiyi bulmaktadır. Bu sayede herhangi bir emoji kullanılmamış dolayısıyla duygusal sınıflandırılması yapılmamış metinlerin anlamlarını en doğru şekilde tespit edebilmek ve doğru duyguyu yakalayabilmektir. Bu çalışmanın yapılabilmesi için literatürde kabul gören bir kaç makine öğrenmesi ve derin öğrenme yöntemlerinden yararlanılmıştır.

Yapılan çalışma sonucunda en doğru sonuçlara hangi algoritma veya yöntem ile ulaşıldığı karşılatırılmak istenmiştir.

4.3 Performans Metrikleri

Eğitilen modellerden hedeflenen ise metinlerin emoji karşılıklarını en doğruya yakın şekilde gösterebilmesidir. Eğitilen modelimiz test veri seti üzerinde çalıştırılarak, test veri setindeki cümlelere uygun emojileri tahmin etmesi beklenmiştir. Yapılan tahmin sonucu çıkan emoji etiketler ile gerçekte test veri setinde bulunan doğru emoji etiketlerinin eşleşme oranları karşılaştırılarak performansı ölçülmeye çalışılmıştır. Yüzdesel olarak doğru veya yanlış tahminler hesaplanmış ayrıca yanlış tahminlerde yapılan süre gelen bir hata veya düzenli olarak yanlış bir tahmin varsa, bu yanlış tahminin neden olduğu ortaya çıkarılması hedeflenmiştir. Bu sayede en doğru sonuçlara ulaşmak istenmiştir.

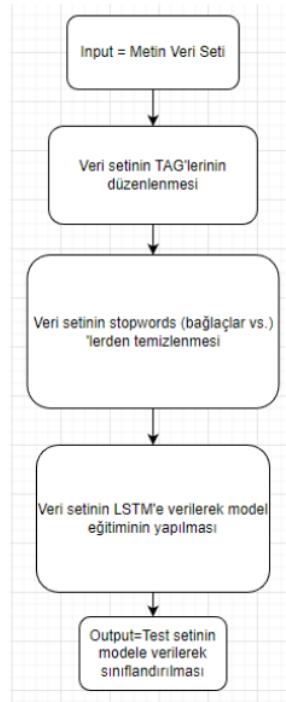
5

Sistem Tasarımı

Bu bölümde sistemi oluşturan bileşenlerden teker teker bahsedilmiştir.

5.1 Yazılım Tasarımı

Modellerin eğitimi sırasında Şekil 5.1'deki akış diyagramı örneğinde de bir örneği bulunan, LSTM, SVM, KNN, Multinomial Naive Bayes kullanılmıştır.



Şekil 5.1 Metin Sınıflandırma Akış Şeması

5.1.1 LSTM

LSTM derin öğrenme alanında kullanılan yapay bir yinelemeli sinir ağı (RNN) mimarisidir. LSTM geri beslemeli çalışır. Anlık verinin yanında veri dizilerini de işleyebilmektedir. Sıradan bir LSTM ünitesi giriş, çıkış ve unut kapısından oluşmaktadır [8].

LSTM ağları, zaman serisi verilerini kullanarak sınıflandırma işleme ve tahmin etme için çok uygundur. Çünkü zaman serisindeki ehemmiyetli durumlar arasında belli süreli gecikmeler olur. LSTM'ler, geleneksel RNN'leri eğitirken karşılaşabilecek unutulan ve yok olan gradyan sorunlarını çözmek için üretilmiştir

Vanishing Gradient Problemiye aktivasyon fonksiyonları vesilesiyle inputumuzu yalnızca belli bir kesit zamanında kullanılır duruma getirebiliriz. Bu kesit zamanı ekseriyetle eksi bir ve bir ya da sıfır ve bir aralığıdır. Küçük bir kesit zamanına indirgediğimiz için inputumuzdaki majör bir farklılık aktivasyon fonksiyonunda gereği kadar majör bir farklılığa sebep olmayabilir. Bu sebeple türevi de minör olur. Ve türevi çok minörse, o seviye gereği miktarda öğrenemez

LSTM bu problemi dört adımda çözüyor [9], bu adımlar:

- **Unutma Kapısı (Forget Gate):**

Hangi bilginin unutulmayacağı veya silineceğine karar verir. Lojiji çok da kafa karıştırıcı değildir. En basit çarpma kuralı olan bir sayı sıfır ile çarpılırsa sonucu sıfır olarak gözlemleriz. Forget Gate'de de benzer mantıkla bir durum gerçekleşiyor. Unutmak istediğimiz tüm girdilerin ağırlığına sıfır değeri veriyoruz.

Evetki hidden layerdan (gizli katmandan) elde edilen bilgiler ve anlık bilgiler Sigmoid Fonksiyonu isimli bir fonksiyona girer. Sıfıra ne kadar yakınsarsa o kadar unutulması gerekiyor, Bire ne kadar yakınsa bilginin o kadar unutulmayacağı anlamına gelir.

- **Girdi Kapısı (Input Gate):**

Cell State'i güncellemek için uygundur. İlk olarak unutma kapısında olduğu üzere Sigmoid fonksiyonu kullanılır, hangi bilginin unutulmayacağına karar verilir. Ardından ağı düzgün hale getirmek amacıyla Tanh fonksiyonu ile beraber eksi bir ile bir arasına indirgenir ve elde edilen iki sonuç çarpılır.

- **Hücre Durumu (Cell State):**

Cell State'in hücrenin kapsadığı en önemli görevi bilgiyi hareket ettirmektir. Hareket ettirilmesi icap eden verileri alır ve hücre sonuna, hücre sonundan da

da farklı hücrelere hareket ettirir. Yani ağ üstündeki veri trafiğini Cell State vesilesiyle sağlarız. öncelikle Forget Gate'den (Unutma Kapısı) gelen sonuç ile ondan önceki katmanın sonucu çarpılır. Ardından Input Gate'den elde edilen değer ile toplanır.

- Çıktı Kapısı(Output Gate):

Bir sonraki seviyeye gidecek değer seçilir. Bu değer, tahminleme amacıyla kullanılır. İlk olarak evvelki değer ile anlık input Sigmoid fonksiyonunda işlenir. Cell State'den elde edilen değer Tanh fonksiyonunda işlendikten sonra iki değer çarpılır ve bir sonraki seviyeye "hemen önceki değer" olarak gönderilir. Cell State ilerler.

5.1.2 Naive Bayes

İsmini matematikçi Thomas BAYES'den alan Naive Bayes algoritması istatistiksel olarak tahmine dayalı bir sınıflandırma algoritmasıdır. Kompleks makine öğrenmesi yöntemleriyle kıyaslandığında kolay öğrenilebilmesi ve uygulanabilmesi yönüyle tercih edilir [10]. Naive Bayes algoritmasının temeli olan Bayes Teoreminin matematiksel karşılığı şekil 5.4 de mevcuttur.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Şekil 5.2 Bayes Teroemi

Naive Bayes makine öğrenmesi algoritması daha çok spam filtreleri, metin analizi vb. alanlarda kullanılmaktadır. .

Naive Bayes algoritması öncelikle kullanım kolaylığı nedeniyle tercih sebebi olmaktadır. İkinci olarak ise diğer sınıflandırma yapan diğer makine öğrenmesi algoritmalarına karşılık eğitim verisinin yalnızca bir kez taranması yeterlidir. Ayrıca empty (kayıp - boş) veriler de olasılık hesaplarına katılmayarak ele alınabilmektedir. Basit ilişkilerin olduğu durumlarda genellikle iyi sonuç çikaran bir yöntemdir [11].

5.1.3 KNN

K-nearest neighbors (K-en yakın komşular) algoritması komşu yakınlıklarına göre karar veren bir sınıflandırma algoritmasıdır. KNN algoritması girdi olarak alınan bir girdi noktasının etiketini, girdi olarak verilen girdi noktasına en yakın olan diğer

veri noktalarının etiketlerine bakarak seçer. Mesela, veri noktalarının bulunduğu yerin (x, y) ile gösterildiğini varsayırsak, girdi olarak alınan girdi noktası (a, b) için etiketi tespit edilirken, (a, b) noktasının en yakınında bulunan k tane veri noktasının etiketleri dikkate alınır bununla beraber en çok tekrar eden etiket girdi olarak alınan girdi noktasına etiket olarak atanır. K değeri büyük oranda insan tarafından göz ile belirlenir ve algoritmanın performansını etkileyen önemli metriklerden biridir.

KNN algoritmasının özellikleri şunlardır:

- KNN diğer algoritmala göre çok daha kolay anlaşılmış ve uygulaması basittir.
- KNN algoritması daha önceden hazırlanmış veri setleri için de çalışır bundan dolayı veri ön işleme ihtiyacı duymaz.
- KNN algoritmasını için en büyük dezavantaj, veri seti ne kadar büyürse işlem hızının da o ölçüde yavaşlamasıdır. Bundan dolayı, veri setleri büyündükçe algoritma çalışma hızı önemli ölçüde yavaşlamaktadır.

5.1.4 Destek Vektör Makineleri

SVM, destek vektör makinesi (support vector machine) algoritması, girdi olarak alınan bir veri seti üzerinde iki sınıf arasındaki ayrimı optimum düzeye yapan hiperdüzlemi bulmayı hedefler. Girdi olarak alınan bir veri noktalarının düzlemlerdeki yerlerinin (x, y) ile ifade edildiğini varsayırsak, hiperdüzlem verileri iki sınıfa optimum şekilde ayırabilen düzlemdir.

SVM algoritmasının özellikleri şunlardır:

- SVM algoritması çoğunlukla yüksek performansla çalışır, SVM'den alınan sonuçlara gayet başarılı olabilmektedir.
- SVM algoritmasıyla veriyi kullanmadan önce veriyi bir dizi ön işleminden geçirmek gereklidir. Bu işlemler veri setinin performansını etkileyebilir.
- SVM algoritması, veri noktalarının özelliklerini daha iyi anlamlandırmak amacıyla kernel fonksiyonlarını kullanır. Bununla beraber çok daha kafa karıştırıcı veri setleri üzerinde daha iyi sonuçlar verebilir.

6

Uygulama

Bu bölümde modelin özellikleri ve projenin aşamalarıyla alakalı bilgi verilmektedir.

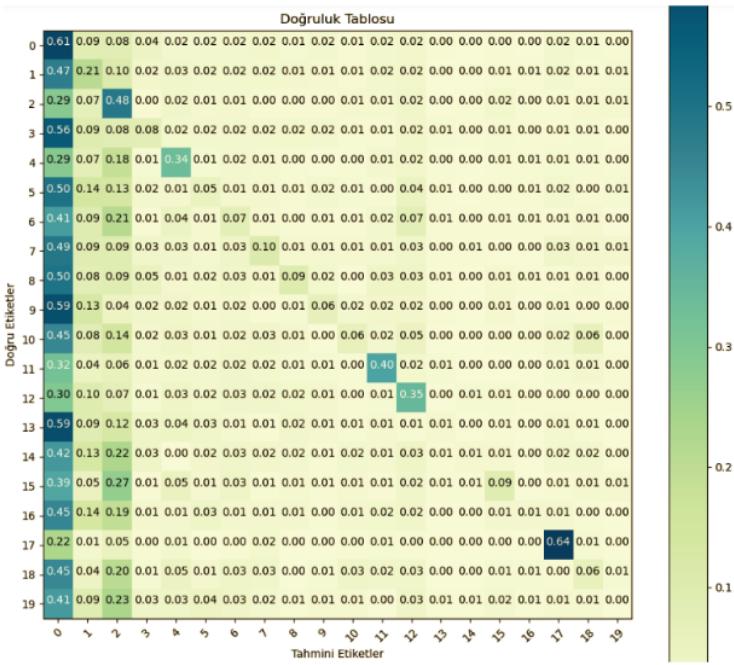
6.1 Modellerin Özellikleri

Modeller toplamda yaklaşık 50.000 emoji içeren metinle eğitilmiştir. Veri setinde mutlu surat, kalp ,kalpli surat ,alev de dahil olmak üzere toplam 20 emoji bulunmaktadır. Bu emojilerden, optimum sonuç alabilmek için belli bir kısmı göz ardı edilmiştir.

Modellerin erken geliştirme esnasında, deneysel olarak elde edilen, Şekil 6.1 ve 6.2'de ortaya çıkan sonuçlarda görülmektedir. SVM ile elde edilen başarıım Şekil 6.1'de görülmektedir. Naive Bayes ile elde edilen başarısma Şekil 6.2'de görülmektedir.



Şekil 6.1 Naive Bayes ile elde edilen başarım



Şekil 6.2 SVM ile elde edilen başarım

6.2 Modellerin Eğitilmesi İçin Kullanılan Yöntem

Eğitim sürecinde 50.000 adet veri setinin bir kısmı test (%20-%30) bir kısmı da eğitim verisi olarak böülümlere ayrılmıştır. Aynı veri kümesi SVM, Naive Bayes, LSTM yöntemleriyle kullanılmıştır.

Bu yöntemler elde ettikleri doğruluk oranlarına göre kıyaslanmıştır.

Her makine öğrenmesi ve derin öğrenme yöntemi farklı sayıda sınıflara göre farklı oranda başarıyı elde etmektedir. Aynı zamanda yine her makine öğrenmesi ve derin öğrenme yöntemi farklı sayıda eğitim veri setine göre farklı oranda başarıyı elde etmektedir. Bundan dolayı her yöntemin optimum çalışabileceği ortak sınıf sayısı bulunmuştur.

6.3 Modellerin Eğitilmesi

Eğitim aşaması çok fazla işlem gücü gerektirdiğinden, kişisel bilgisayarların gücü yerinde google colab'ın sunduğu GPU kullanılmıştır.

Eğitim veri setimizimin bir kısmı doğrulama veri seti olarak ayrılmıştır. Bu sayede eğitim aşamasında eğitim veri setimizin doğru eğitilip eğitilmemiği doğrulama veri setimizle sürekli olarak test edilmiştir. Eğitim sırasında aşırı öğrenme-ezberleme (overfit) olduğu görüldüğünden, eğitim tur sayısı (epoch) değeri belli bir sayıyla tutularak, veri setindeki sınıf sayısı düşürülerek veya doğrulama veri setimizin (val data) rastgele alınması gibi yöntemlerle sorun aşılmasına çalışılmıştır.

7

Deneysel Sonuçlar

Bu bölümde projeye alakalı alınan deneysel sonuçlar gösterilmiştir.

7.1 Naive Bayes

7.1.1 Naive Bayes Nedir

Naive Bayes, bir sınıflandırma algoritmasıdır ve ismi ünlü matematik bilimci Thomas Bayes'den gelmektedir. Aynı zamanda bir makine öğrenmesi algoritmasıdır. Bu algoritma, alınan girdilerin sınıfını (kategorisini) tespit etmek amacıyla olasılığı temel alan hesaplama işlemleri yapar. Naive Bayes algoritması, sisteme girdi olarak verilen verileri inceler sonrasında sisteme girdi olarak verilen verileri kullanarak bir sınıflandırma yapar. 5 Bu sınıflandırmanın temel amacı, verilerin yer aldığı sınıfı tespit etmektedir. Naive Bayes eğiticili sınıflandırma algoritmalarından biridir.

7.2 Destek Vektör Makineleri

3

7.2.1 SVM Nedir

SVM, bir makine öğrenmesi algoritmasıdır. SVM algoritması, verilen verileri kullanarak sınıflandırma ve benzeri birçok tahminleme sıkıntılara çare olabilir. 4 SVM algoritması, dışarıdan alınan girdilerin iki sınıfını ayırmak amacıyla bir hiper düzleme oluşturmaya çalışır. Oluşturmayı çalıştığı hiper düzleme, iki sınıf arasındaki oluşturulabilecek ve ayırcı özelliği en yüksek olan düzlemdir. İsmi de aslında tam olarak bu özelliğinden gelmektedir. SVM, dışarıdan girdi olarak alınan verilerin karmaşıklık oraniyla orantılı şekilde sonuç vermektedir. Genellikle diğer makine öğrenmesi algoritmalarından daha iyi performans gösterir.

7.3 Veri Seti

Veri seti [7] toplamda 70.000 tweetten oluşmaktadır ve yirmi sınıfından oluşmaktadır. Naive Bayes ve SVM yöntemlerini deneySEL olarak kıyaslayabilmek amacıyla bu veri setindeki sınıfları kirparak kullandık ve bu veri setini farklı oranlarda kullanarak, yöntemlerin veri seti büyüklüğüne ve eğitim, test oranlarına göre başarım oranlarını kıyasladık. Kirptığımız haliyle sınıf sayısı dört olmuş oldu.

7.3.1 Veri Setinin Boyutunu Değiştirme

Test ve eğitim veri oranı %20'ye eşitken oluşan sonuçlar Tablo 7.1'de gösterilmiştir.

Tablo 7.1 Yöntem Kiyaslama tablosu

Veri Seti	Test Kümesi Uzunluğu	Eğitim Kümesi Uzunluğu	Test Kümesi/ (Test Kümesi + Eğitim Kümesi)	NAIVE BAYES Başarı Oranı(%)	SUPPORT VECTOR MACHINE Başarı Oranı(%)
1. Veri Seti	14000	56000	%20	43,64	44,31
2. Veri Seti	11000	44000	%20	42,13	43,49
3. Veri Seti	9000	36000	%20	43,14	43,21
4. Veri Seti	5000	20000	%20	41,82	41,78
5. Veri Seti	3000	12000	%20	39,67	41,07
6. Veri Seti	2000	8000	%20	40,4	40,1

7.3.2 Veri Setinde Oran Değiştirme

Tablo 7.13'de görüldüğü üzere, en yüksek başarım en büyük veri setinde sağlanmıştır. Bundan dolayı bu veri setindeki test ve eğitim verilerinin dağılım oranını değiştirerek test ettim. Test ve eğitim verisi sayıları toplam 70.000'ken oluşan sonuçlar Tablo 7.22'de gösterilmiştir.

Tablo 7.2 Yöntem Kiyaslama tablosu

Veri Seti	Test Kümesi Uzunluğu	Eğitim Kümesi Uzunluğu	Test Kümesi/ (Test Kümesi + Eğitim Kümesi)	NAIVE BAYES Başarı Oranı(%)	SUPPORT VECTOR MACHINE Başarı Oranı(%)
1. Veri Seti	21000	49000	%30	43,22	44,04
2. Veri Seti	24500	45500	%35	43	44,08
3. Veri Seti	28000	42000	%40	42,98	44,31
4. Veri Seti	35000	35000	%50	42,94	44,10

7.4 Sonuç

Sonuç olarak Tablo 7.2'de görüldüğü gibi test ve eğitim kümesi oranları ne kadar değişse de, sonuca olan etkisi her iki yöntem için de ciddiye alınır düzeyde değil. Tablo 7.1'e baktığımızda da, belli seviyede eğitim ve test verim olduğu zaman başarım ciddiye alınır biçimde değişmiyor, ama test ve eğitim kümescini çok küçülttüğümüz zaman, gözle görülür bir azalma söz konusu.

8

Performans Analizi

Bu bölümde aynı veri setini [7] çeşitli DL ve ML yöntemlerinde kullanılarak elde edilen sonuçlar gösterilmiş ve yorumlanmıştır. Vektörize yöntemi olarak aksi belirtilmediği sürece Counter Vectorizer yöntemi kullanıldı. Counter Vectorizer yöntemiyle aldığımız sonuçları kıyaslamak amacıyla aynı zamanda TF-IDF vektörize yöntemini de kullandık.

8.1 TF-IDF ile Counter Vectorizer Arasındaki Farklar

TF-IDF (Term Frequency-Inverse Document Frequency) ve Counter Vectorizer yöntemleri, metinleri sayı vektörlerine dönüştürmek için kullanılan döküman tabanlı özellik çıkarım yöntemleridir. İki yöntemin özellik çıkarım stratejisi biraz farklıdır.

TF-IDF, bir döküman içindeki herhangi bir kelimeyi diğer dökümanlarda arayarak, kelimenin döküman içindeki önemini tespit eder. Tespit edilmek istenen kelimenin önemi, kelimenin dökümanda geçme sıklığı ile ters orantılı olarak hesaplanır. Kelimenin, daha fazla geçtiği dökümanlarda ağırlığını diğer kelimelere göre azaltır. Böylece, dökümanlar arasındaki benzerlik durumu Count Vectorizer yöntemine kıyasla daha doğru ölçümlenebilir.

Counter Vectorizer ise, sadece döküman içindeki kelime sıklıklarına bakarak bir sayı vektörü oluşturur. Diğer dökümanlarla kıyaslama işine girişmez. Bundan dolayı, Counter Vectorizer yöntemi, dökümanlar arasındaki benzerlikleri ölçmek için pek de uygun değildir, ama kelime sıklıklarını ölçümlemek için tercih edilebilir.

Özetle, TF-IDF kelimeler için sadece sıklığı değil, aynı zamanda ayırtıcı özelliğini ölçerken, Counter Vectorizer kelime için yalnızca sıklığı ölçer. Bundan dolayı, iki yöntem de farklı hedefler doğrultusunda tercih edilebilir.

8.2 Makine Öğrenmesi

ML yöntemlerinin sonuçları yer almaktadır.

8.2.1 Normal Veri Kümesi Sonuçları

8.2.1.1 Multinomial Naive Bayes

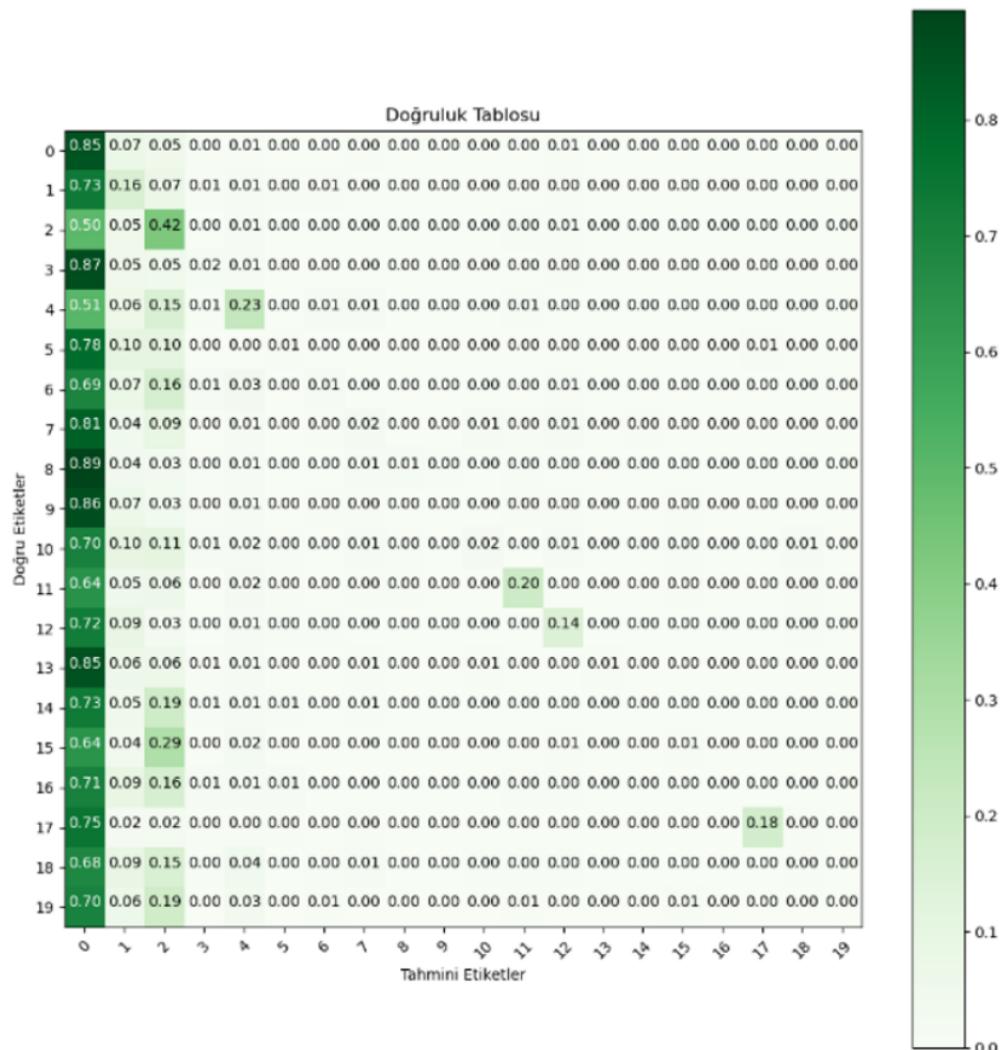
Multinomial Naive Bayes yöntemi için sınıflandırma raporu Şekil 8.1'de gösterilmiştir.

Şekil 8.1 Multinomial Naive Bayes Normal Sınıflandırma Raporu

print(classification_report(y_test, y_predict_test))				
	precision	recall	f1-score	support
0	0.25	0.85	0.39	1632
1	0.22	0.16	0.18	776
2	0.36	0.42	0.39	791
3	0.22	0.02	0.03	465
4	0.49	0.23	0.32	351
5	0.12	0.01	0.01	364
6	0.15	0.01	0.02	301
7	0.17	0.02	0.03	273
8	0.40	0.01	0.03	275
9	0.00	0.00	0.00	244
10	0.50	0.02	0.04	239
11	0.69	0.20	0.31	208
12	0.49	0.14	0.22	220
13	0.50	0.01	0.01	192
14	0.00	0.00	0.00	196
15	0.17	0.01	0.01	168
16	0.00	0.00	0.00	207
17	0.79	0.18	0.30	201
18	0.11	0.00	0.01	221
19	0.00	0.00	0.00	174
accuracy			0.27	7498
macro avg	0.28	0.11	0.11	7498
weighted avg	0.28	0.27	0.19	7498

Multinomial Naive Bayes yöntemi için doğrulama tablosu Tablo 8.1'de gösterilmiştir.

Tablo 8.1 Multinomial Naive Bayes Normal Doğruluk tablosu



8.2.1.2 SVM

SVM yöntemi için sınıflandırma raporu Şekil 8.2'de gösterilmiştir.

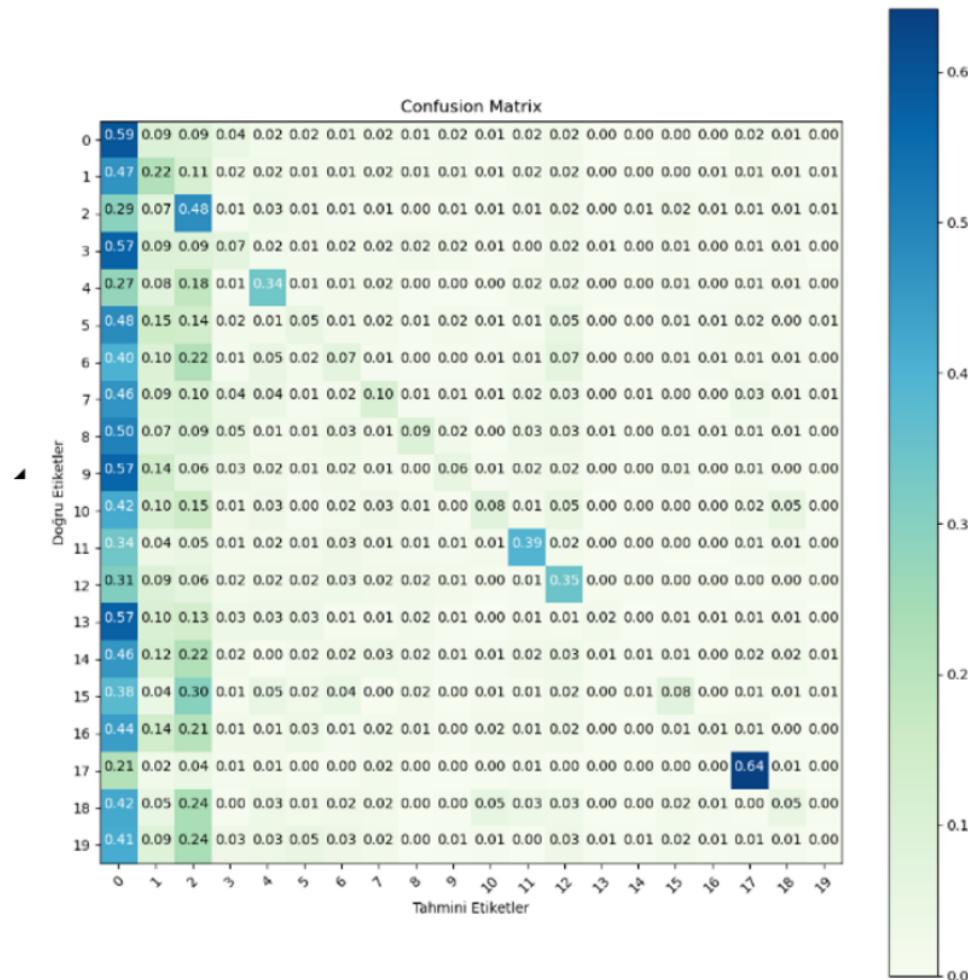
Şekil 8.2 SVM Normal Sınıflandırma Raporu

```
5]: print(classification_report(y_test, y_predict_test))
```

	precision	recall	f1-score	support
0	0.25	0.88	0.39	1632
1	0.29	0.09	0.14	776
2	0.36	0.40	0.38	791
3	0.50	0.01	0.02	465
4	0.54	0.25	0.34	351
5	0.00	0.00	0.00	364
6	0.24	0.01	0.03	301
7	0.45	0.08	0.13	273
8	0.40	0.03	0.05	275
9	0.00	0.00	0.00	244
10	0.50	0.02	0.04	239
11	0.60	0.35	0.44	208
12	0.38	0.33	0.35	220
13	0.50	0.01	0.01	192
14	0.00	0.00	0.00	196
15	0.67	0.01	0.02	168
16	0.00	0.00	0.00	207
17	0.62	0.59	0.60	201
18	0.15	0.01	0.02	221
19	0.00	0.00	0.00	174
accuracy			0.29	7498
macro avg	0.32	0.15	0.15	7498
weighted avg	0.31	0.29	0.21	7498

SVM yöntemi için doğrulama tablosu Tablo 8.2'de gösterilmiştir.

Tablo 8.2 SVM Normal Doğruluk tablosu



8.2.1.3 SVM TF-IDF

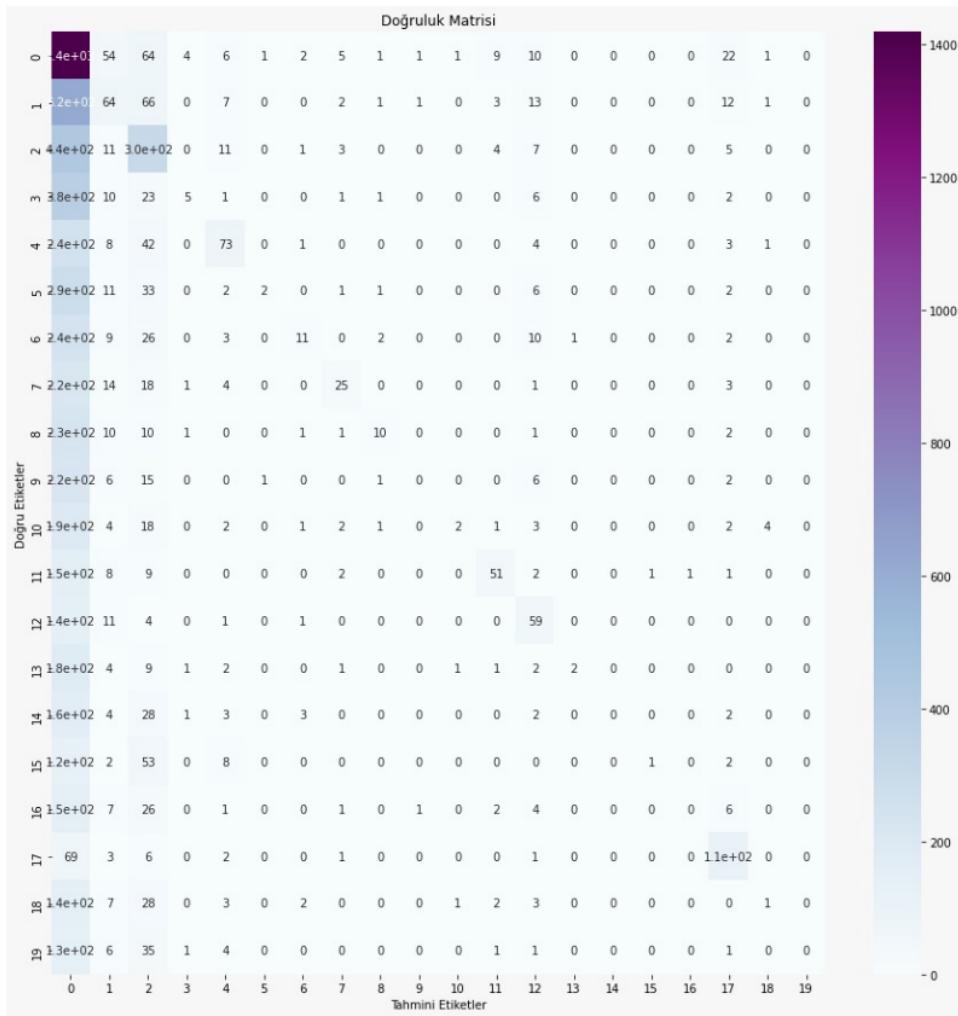
SVM yöntemiyle beraber TFIDF vektörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.3'de gösterilmiştir.

Şekil 8.3 SVM TF-IDF Normal Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.25	0.89	0.39	1600
1	0.25	0.08	0.12	788
2	0.37	0.39	0.38	784
3	0.36	0.01	0.02	432
4	0.55	0.19	0.29	376
5	0.50	0.01	0.01	346
6	0.48	0.04	0.07	306
7	0.56	0.09	0.15	282
8	0.56	0.04	0.07	268
9	0.00	0.00	0.00	249
10	0.40	0.01	0.02	231
11	0.69	0.23	0.34	225
12	0.42	0.27	0.33	218
13	0.67	0.01	0.02	202
14	0.00	0.00	0.00	206
15	0.50	0.01	0.01	187
16	0.00	0.00	0.00	196
17	0.61	0.57	0.59	190
18	0.12	0.01	0.01	192
19	0.00	0.00	0.00	182
accuracy			0.29	7460
macro avg	0.36	0.14	0.14	7460
weighted avg	0.35	0.29	0.20	7460

SVM yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.3'de gösterilmiştir.

Tablo 8.3 SVM TF-IDF Normal Doğruluk tablosu



8.2.1.4 KNN

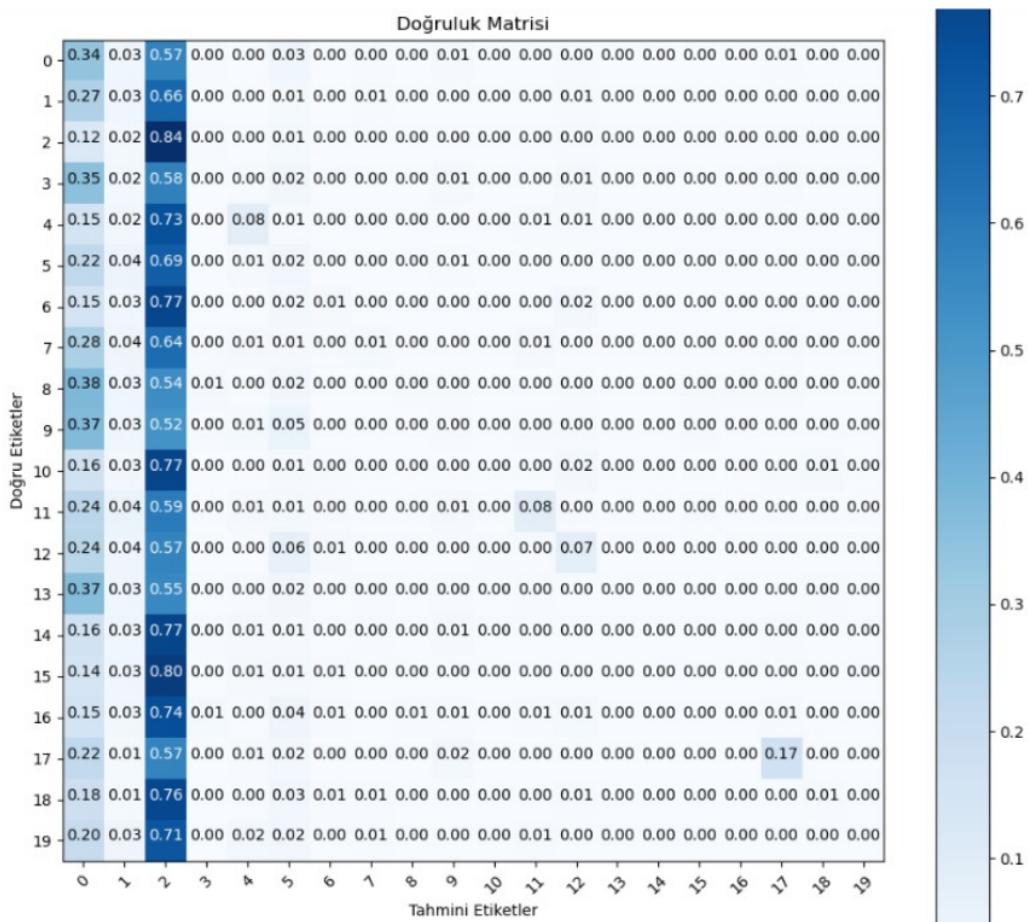
KNN yöntemi için sınıflandırma raporu Şekil 8.4'de gösterilmiştir.

Şekil 8.4 KNN Normal Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.29	0.34	0.31	1614
1	0.12	0.03	0.05	792
2	0.13	0.84	0.23	785
3	0.00	0.00	0.00	433
4	0.53	0.08	0.14	377
5	0.05	0.02	0.03	347
6	0.18	0.01	0.02	307
7	0.14	0.01	0.02	284
8	0.11	0.00	0.01	269
9	0.03	0.00	0.01	251
10	0.00	0.00	0.00	232
11	0.56	0.08	0.14	229
12	0.31	0.07	0.12	219
13	0.00	0.00	0.00	202
14	0.00	0.00	0.00	207
15	0.00	0.00	0.00	187
16	0.00	0.00	0.00	196
17	0.63	0.17	0.26	192
18	0.14	0.01	0.01	193
19	0.00	0.00	0.00	182
accuracy			0.18	7498
macro avg	0.16	0.08	0.07	7498
weighted avg	0.18	0.18	0.12	7498

KNN yöntemi için doğrulama tablosu Tablo 8.4'de gösterilmiştir.

Tablo 8.4 KNN Normal Doğruluk tablosu



8.2.1.5 KNN-TFIDF

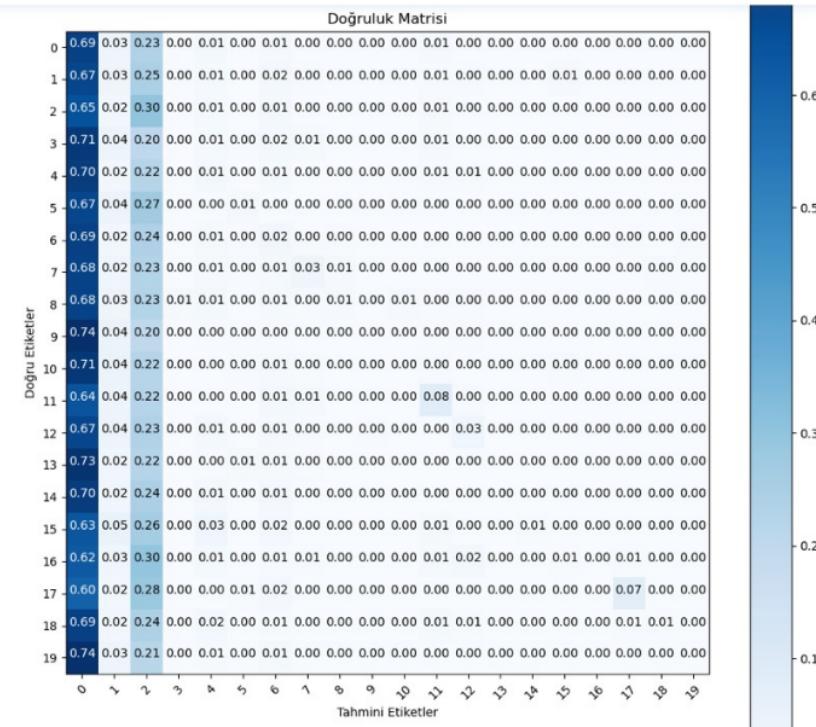
KNN yöntemiyle beraber TFIDF vektörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.5'de gösterilmiştir.

Şekil 8.5 KNN-TFIDF Normal Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.22	0.69	0.33	1600
1	0.11	0.03	0.05	788
2	0.13	0.30	0.18	784
3	0.00	0.00	0.00	432
4	0.07	0.01	0.02	376
5	0.15	0.01	0.02	346
6	0.09	0.02	0.04	306
7	0.29	0.03	0.06	282
8	0.22	0.01	0.01	268
9	0.00	0.00	0.00	249
10	0.00	0.00	0.00	231
11	0.33	0.08	0.13	225
12	0.30	0.03	0.05	218
13	0.00	0.00	0.00	202
14	0.00	0.00	0.00	206
15	0.00	0.00	0.00	187
16	0.00	0.00	0.00	196
17	0.61	0.07	0.13	190
18	0.20	0.01	0.01	192
19	0.00	0.00	0.00	182
accuracy			0.19	7460
macro avg	0.14	0.06	0.05	7460
weighted avg	0.14	0.19	0.11	7460

KNN yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.5'de gösterilmiştir.

Tablo 8.5 KNN-TFIDF Normal Doğruluk tablosu



8.2.1.6 Multinomial Naive Bayes-TFIDF

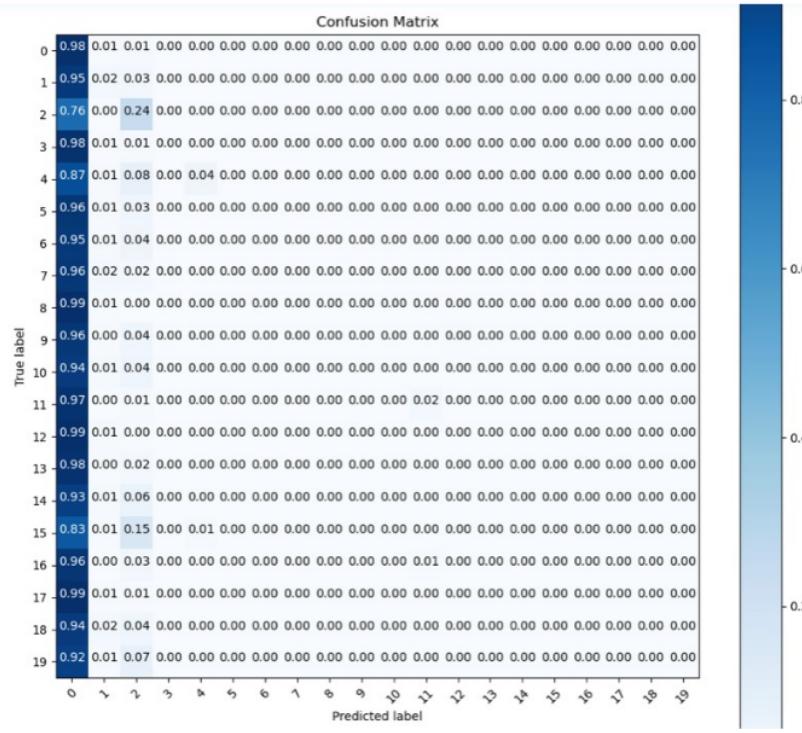
Multinomial Naive Bayes yöntemiyle beraber TFIDF vekötörize yöntemi kullanılmışlığı durumda sınıflandırma raporu Şekil 8.6'da gösterilmiştir.

Şekil 8.6 Multinomial Naive Bayes-TFIDF Normal Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.22	0.98	0.37	1600
1	0.25	0.02	0.04	788
2	0.49	0.24	0.32	784
3	0.00	0.00	0.00	432
4	0.76	0.04	0.08	376
5	0.00	0.00	0.00	346
6	0.00	0.00	0.00	306
7	0.00	0.00	0.00	282
8	0.00	0.00	0.00	268
9	0.00	0.00	0.00	249
10	0.00	0.00	0.00	231
11	0.80	0.02	0.03	225
12	0.00	0.00	0.00	218
13	0.00	0.00	0.00	202
14	0.00	0.00	0.00	206
15	0.00	0.00	0.00	187
16	0.00	0.00	0.00	196
17	0.00	0.00	0.00	190
18	0.00	0.00	0.00	192
19	0.00	0.00	0.00	182
accuracy			0.24	7460
macro avg	0.13	0.06	0.04	7460
weighted avg	0.19	0.24	0.12	7460

Multinomial Naive Bayes yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.6'da gösterilmiştir.

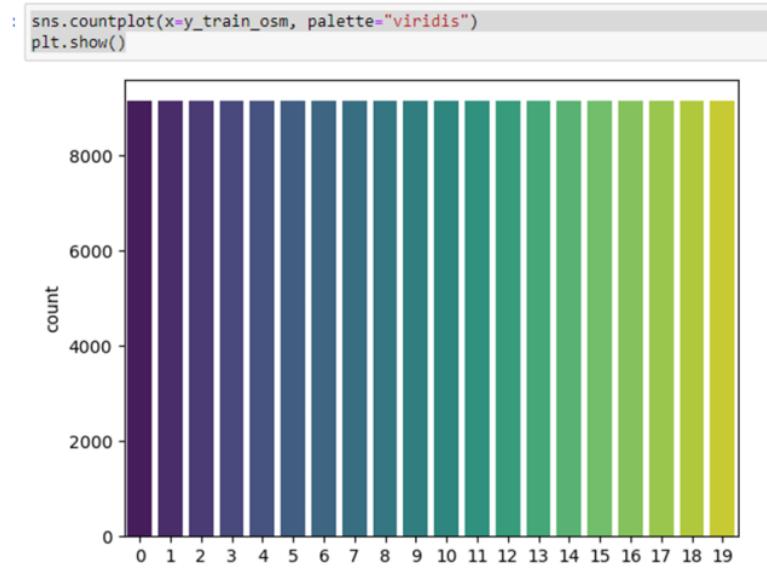
Tablo 8.6 Multinomial Naive Bayes-TFIDF Normal Doğruluk tablosu



8.2.2 ROS Sonuçları

ROS Sonrası veri setinin durumu Şekil 8.7'de gösterilmiştir

Şekil 8.7 ROS Veri Seti Durumu



8.2.2.1 Multinomial Naive Bayes

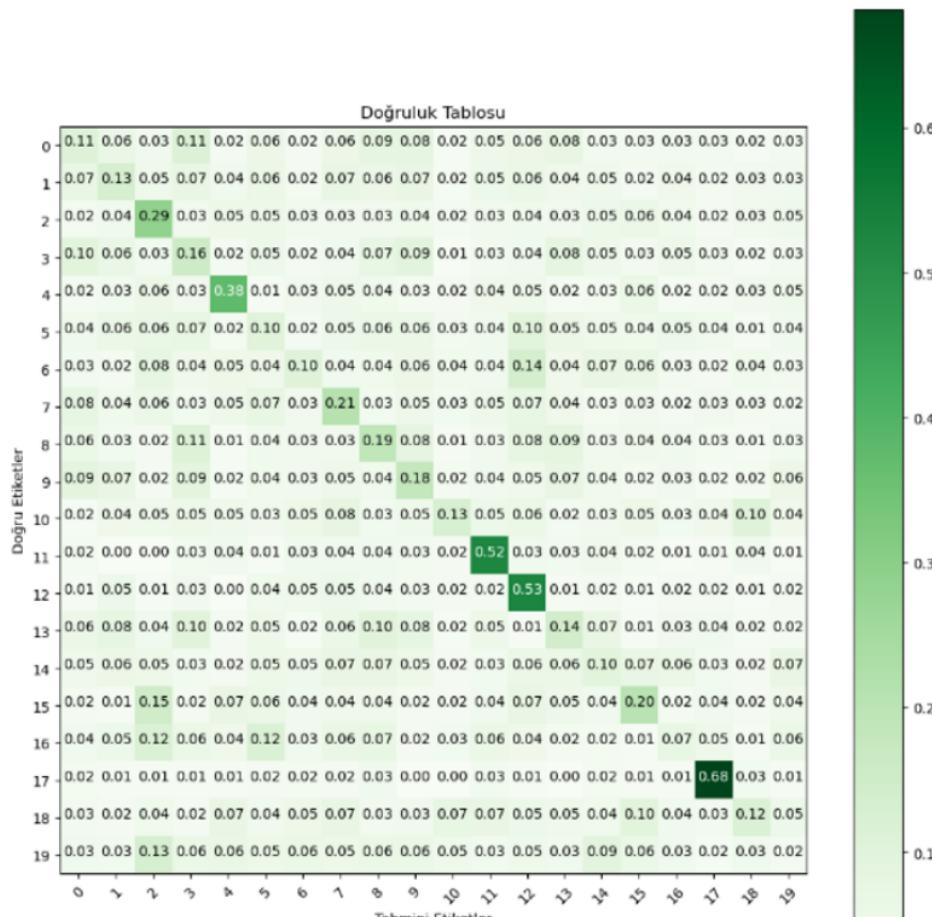
Multinomial Naive Bayes yöntemi için sınıflandırma raporu Şekil 8.8'de gösterilmiştir.

Şekil 8.8 Multinomial Naive Bayes ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.39	0.11	0.17	1632
1	0.24	0.13	0.17	776
2	0.42	0.29	0.34	791
3	0.14	0.16	0.15	465
4	0.36	0.38	0.37	351
5	0.09	0.10	0.10	364
6	0.12	0.10	0.11	301
7	0.13	0.21	0.16	273
8	0.11	0.19	0.14	275
9	0.10	0.18	0.13	244
10	0.16	0.13	0.14	239
11	0.26	0.52	0.35	208
12	0.22	0.53	0.31	220
13	0.07	0.14	0.09	192
14	0.06	0.10	0.07	196
15	0.11	0.20	0.14	168
16	0.06	0.07	0.06	207
17	0.41	0.68	0.51	201
18	0.12	0.12	0.12	221
19	0.02	0.02	0.02	174
accuracy			0.19	7498
macro avg	0.18	0.22	0.18	7498
weighted avg	0.24	0.19	0.19	7498

Multinomial Naive Bayes yöntemi için doğrulama tablosu Tablo 8.7'de gösterilmiştir.

Tablo 8.7 Multinomial Naive Bayes ROS Doğruluk tablosu



8.2.2.2 Multinomial Naive Bayes TF-IDF

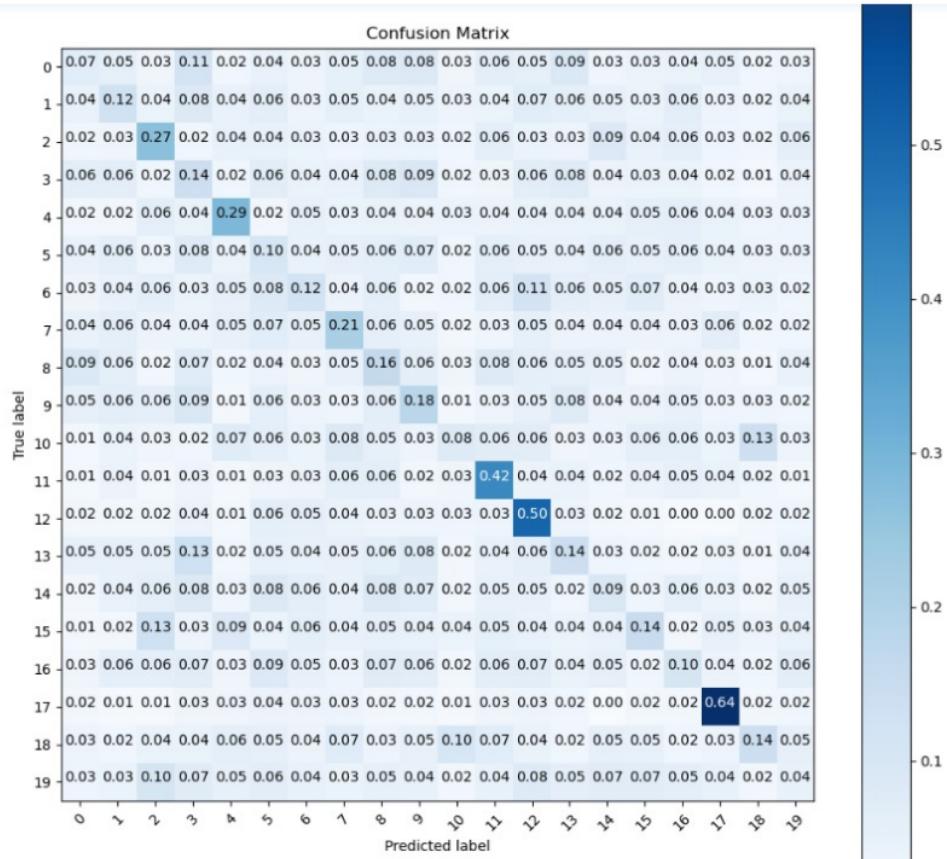
Multinomial Naive Bayes yöntemiyle beraber TFIDF vekötörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.9'da gösterilmiştir.

Şekil 8.9 Multinomial Naive Bayes-TFIDF ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.35	0.07	0.11	1600
1	0.26	0.12	0.17	788
2	0.45	0.27	0.33	784
3	0.11	0.14	0.12	432
4	0.31	0.29	0.30	376
5	0.09	0.10	0.09	346
6	0.12	0.12	0.12	306
7	0.15	0.21	0.18	282
8	0.10	0.16	0.12	268
9	0.10	0.18	0.13	249
10	0.09	0.08	0.08	231
11	0.20	0.42	0.27	225
12	0.22	0.50	0.30	218
13	0.07	0.14	0.09	202
14	0.06	0.09	0.07	206
15	0.09	0.14	0.11	187
16	0.06	0.10	0.08	196
17	0.32	0.64	0.42	190
18	0.12	0.14	0.13	192
19	0.03	0.04	0.04	182
accuracy			0.17	7460
macro avg	0.16	0.20	0.16	7460
weighted avg	0.23	0.17	0.17	7460

Multinomial Naive Bayes yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığında durumda doğrulama tablosu Tablo 8.8'de gösterilmiştir.

Tablo 8.8 Multinomial Naive Bayes-TFIDF ROS Doğruluk tablosu



8.2.2.3 SVM

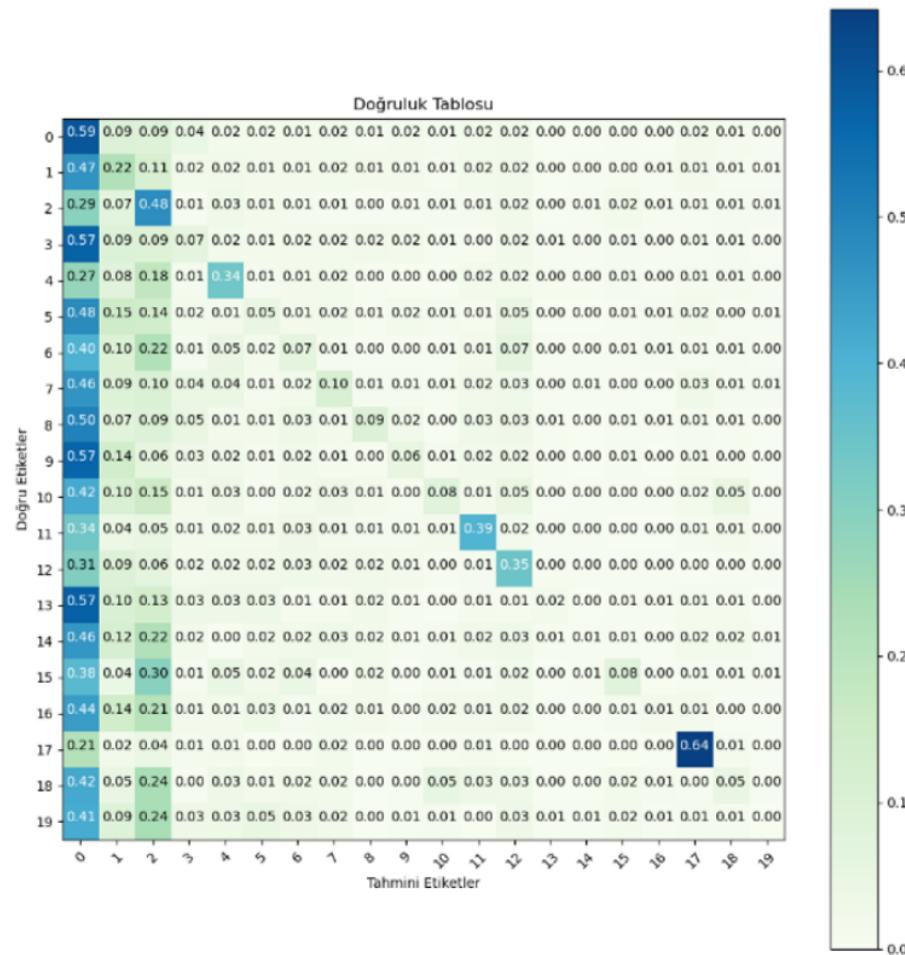
SVM yöntemi için sınıflandırma raporu Şekil 8.10'da gösterilmiştir.

Şekil 8.10 SVM ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.28	0.59	0.38	1632
1	0.22	0.22	0.22	776
2	0.31	0.48	0.38	791
3	0.16	0.07	0.09	465
4	0.41	0.34	0.37	351
5	0.13	0.05	0.07	364
6	0.15	0.07	0.09	301
7	0.20	0.10	0.13	273
8	0.26	0.09	0.13	275
9	0.15	0.06	0.08	244
10	0.21	0.08	0.11	239
11	0.43	0.39	0.41	208
12	0.29	0.35	0.32	220
13	0.14	0.02	0.03	192
14	0.05	0.01	0.01	196
15	0.20	0.08	0.11	168
16	0.10	0.01	0.03	207
17	0.59	0.64	0.62	201
18	0.13	0.05	0.07	221
19	0.00	0.00	0.00	174
accuracy			0.28	7498
macro avg	0.22	0.18	0.18	7498
weighted avg	0.24	0.28	0.24	7498

SVM yöntemi için doğrulama tablosu Tablo 8.9'da gösterilmiştir.

Tablo 8.9 SVM ROS Doğruluk tablosu



8.2.2.4 SVM TF-IDF

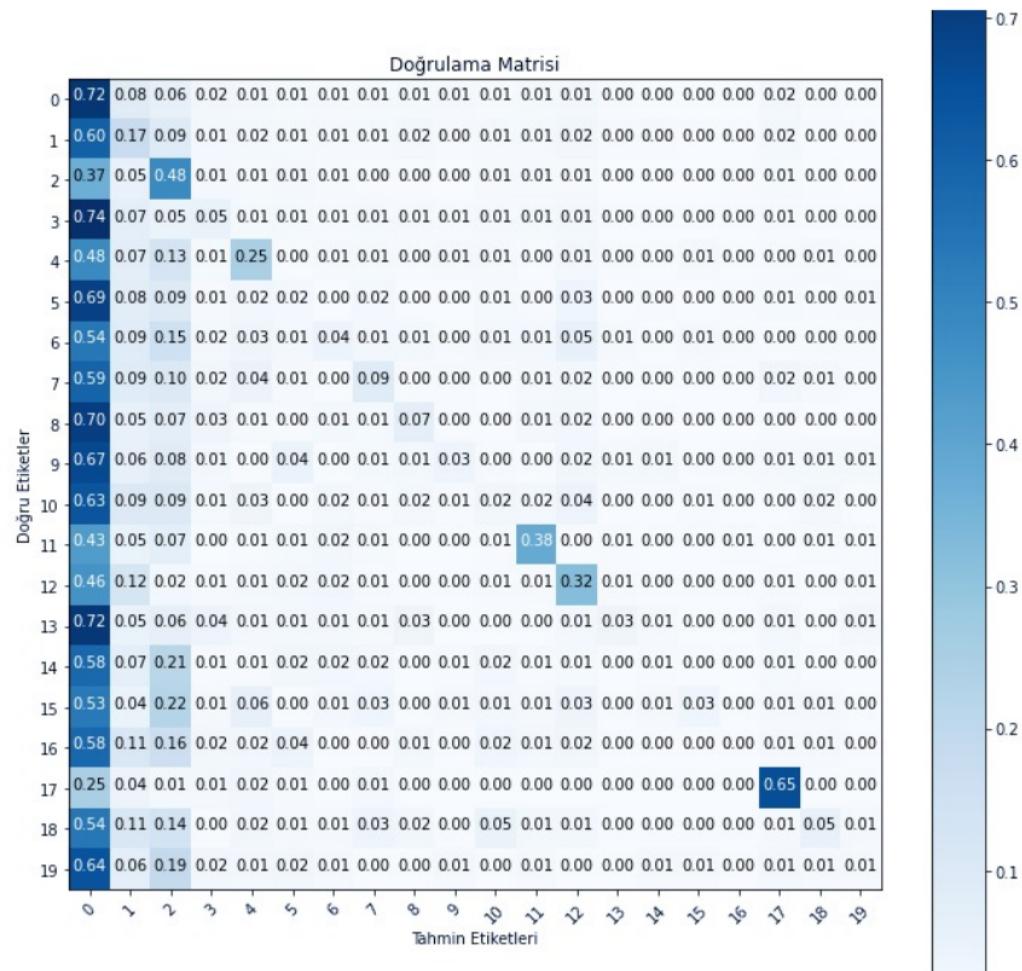
SVM yöntemiyle beraber TFIDF vekötörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.11'de gösterilmiştir.

Şekil 8.11 SVM TF-IDF ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0.0	0.26	0.72	0.38	1348
1.0	0.21	0.17	0.19	655
2.0	0.38	0.48	0.43	646
3.0	0.17	0.05	0.08	367
4.0	0.44	0.25	0.32	315
5.0	0.07	0.02	0.03	287
6.0	0.17	0.04	0.06	253
7.0	0.25	0.09	0.13	234
8.0	0.22	0.07	0.10	225
9.0	0.19	0.03	0.06	208
10.0	0.08	0.02	0.03	191
11.0	0.60	0.38	0.46	187
12.0	0.37	0.32	0.35	180
13.0	0.25	0.03	0.05	170
14.0	0.07	0.01	0.01	174
15.0	0.29	0.03	0.05	158
16.0	0.00	0.00	0.00	164
17.0	0.63	0.65	0.64	161
18.0	0.25	0.05	0.09	155
19.0	0.08	0.01	0.01	155
accuracy			0.29	6233
macro avg	0.25	0.17	0.17	6233
weighted avg	0.26	0.29	0.23	6233

SVM yöntemiyle beraber TFIDF vekörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.10'da gösterilmiştir.

Tablo 8.10 SVM TF-IDF ROS Doğruluk tablosu



8.2.2.5 SVM POLY

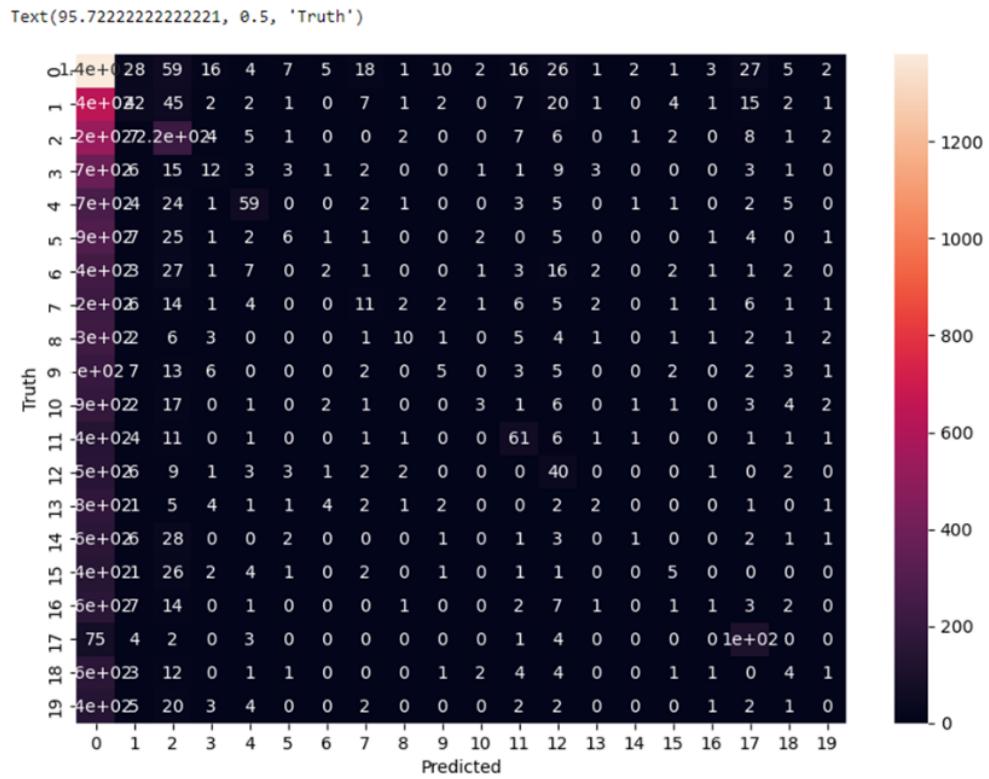
SVM POLY yöntemi için sınıflandırma raporu Şekil 8.12'de gösterilmiştir.

Şekil 8.12 SVM POLY ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.24	0.86	0.37	1614
1	0.28	0.05	0.09	792
2	0.37	0.27	0.31	785
3	0.21	0.03	0.05	433
4	0.56	0.16	0.24	377
5	0.23	0.02	0.03	347
6	0.12	0.01	0.01	307
7	0.20	0.04	0.06	284
8	0.45	0.04	0.07	269
9	0.20	0.02	0.04	251
10	0.25	0.01	0.02	232
11	0.49	0.27	0.35	229
12	0.23	0.18	0.20	219
13	0.14	0.01	0.02	202
14	0.14	0.00	0.01	207
15	0.23	0.03	0.05	187
16	0.08	0.01	0.01	196
17	0.56	0.54	0.55	192
18	0.11	0.02	0.03	193
19	0.00	0.00	0.00	182
accuracy			0.26	7498
macro avg	0.25	0.13	0.13	7498
weighted avg	0.27	0.26	0.18	7498

SVM POLY yöntemi için doğrulama tablosu Tablo 8.11'de gösterilmiştir.

Tablo 8.11 SVM POLY ROS Doğruluk tablosu



8.2.2.6 SVM RBF

SVM RBF yöntemi için sınıflandırma raporu Şekil 8.13'de gösterilmiştir.

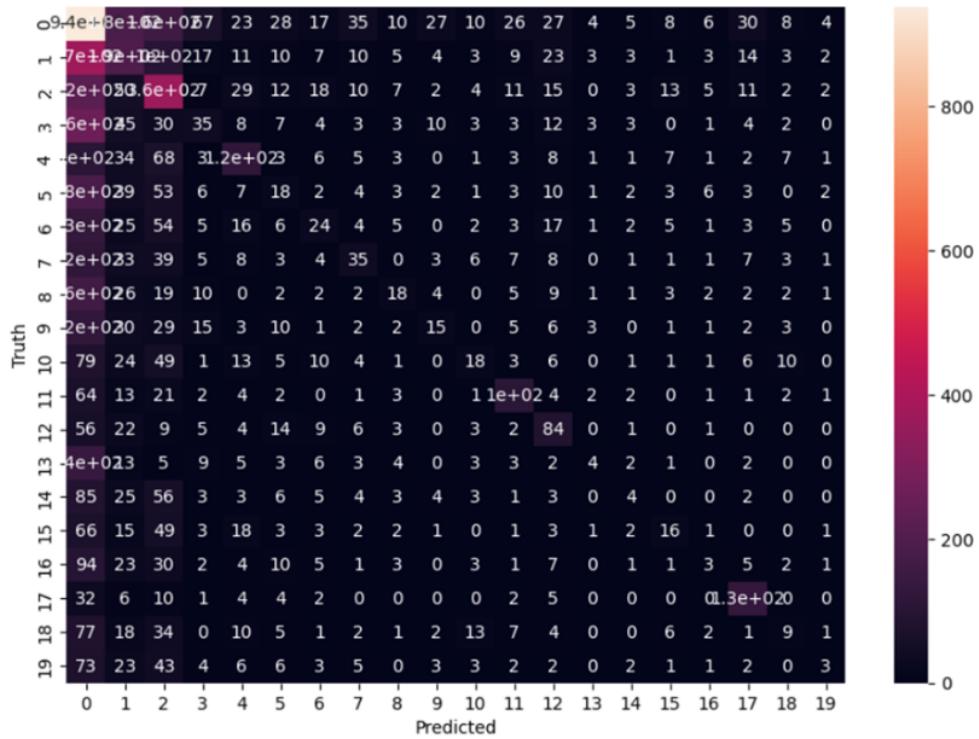
Şekil 8.13 SVM RBF ROS Sınıflandırma Raporu

print(classification_report(y_test, y_predict_test))				
	precision	recall	f1-score	support
0	0.28	0.58	0.38	1614
1	0.23	0.24	0.24	792
2	0.30	0.46	0.36	785
3	0.17	0.08	0.11	433
4	0.40	0.32	0.35	377
5	0.11	0.05	0.07	347
6	0.19	0.08	0.11	307
7	0.25	0.12	0.17	284
8	0.24	0.07	0.10	269
9	0.19	0.06	0.09	251
10	0.23	0.08	0.12	232
11	0.52	0.46	0.49	229
12	0.33	0.38	0.35	219
13	0.17	0.02	0.04	202
14	0.11	0.02	0.03	207
15	0.24	0.09	0.13	187
16	0.08	0.02	0.03	196
17	0.57	0.66	0.61	192
18	0.16	0.05	0.07	193
19	0.15	0.02	0.03	182
accuracy			0.28	7498
macro avg	0.25	0.19	0.19	7498
weighted avg	0.25	0.28	0.24	7498

SVM RBF yöntemi için doğrulama tablosu Tablo 8.12'de gösterilmiştir.

Tablo 8.12 SVM RBF ROS Doğruluk tablosu

Text(95.7222222222221, 0.5, 'Truth')



8.2.2.7 SVM SIGMOID

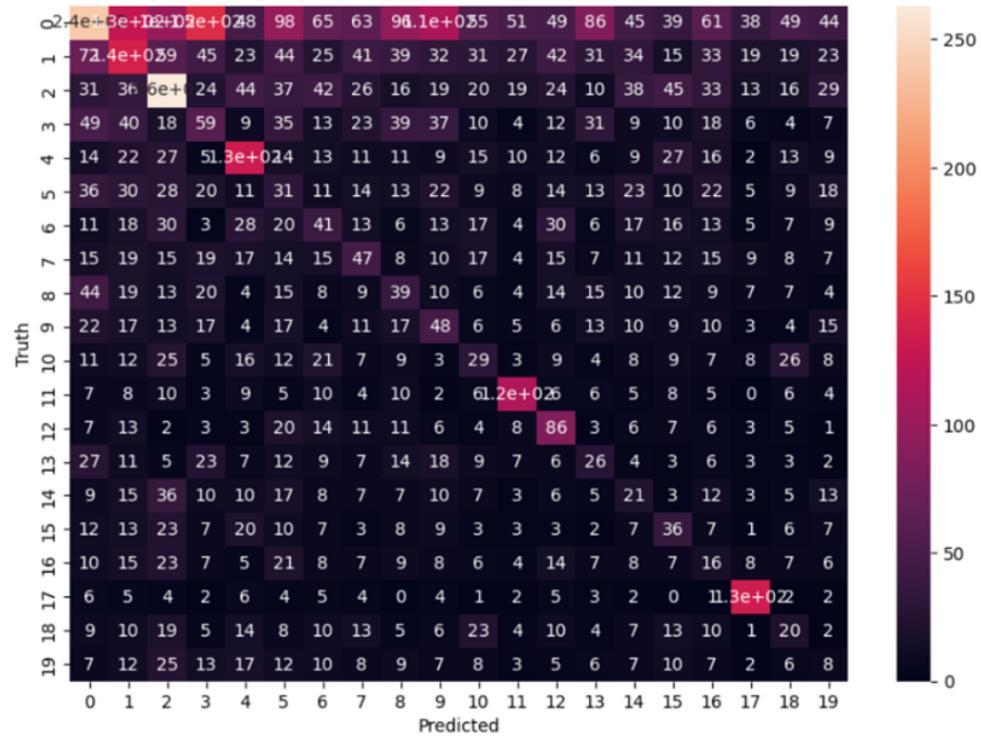
SVM SIGMOID yöntemi için sınıflandırma raporu Şekil 8.14'de gösterilmiştir.

Şekil 8.14 SVM SIGMOID ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.38	0.15	0.21	1614
1	0.24	0.17	0.20	792
2	0.36	0.34	0.35	785
3	0.14	0.14	0.14	433
4	0.31	0.35	0.33	377
5	0.07	0.09	0.08	347
6	0.12	0.13	0.13	307
7	0.14	0.17	0.15	284
8	0.11	0.14	0.12	269
9	0.12	0.19	0.15	251
10	0.10	0.12	0.11	232
11	0.40	0.50	0.44	229
12	0.23	0.39	0.29	219
13	0.09	0.13	0.11	202
14	0.07	0.10	0.09	207
15	0.12	0.19	0.15	187
16	0.05	0.08	0.06	196
17	0.50	0.70	0.58	192
18	0.09	0.10	0.10	193
19	0.04	0.04	0.04	182
accuracy			0.20	7498
macro avg	0.18	0.21	0.19	7498
weighted avg	0.24	0.20	0.21	7498

SVM SIGMOID yöntemi için doğrulama tablosu Tablo 8.13'de gösterilmiştir.

Tablo 8.13 SVM SIGMOID ROS Doğruluk tablosu



8.2.2.8 KNN

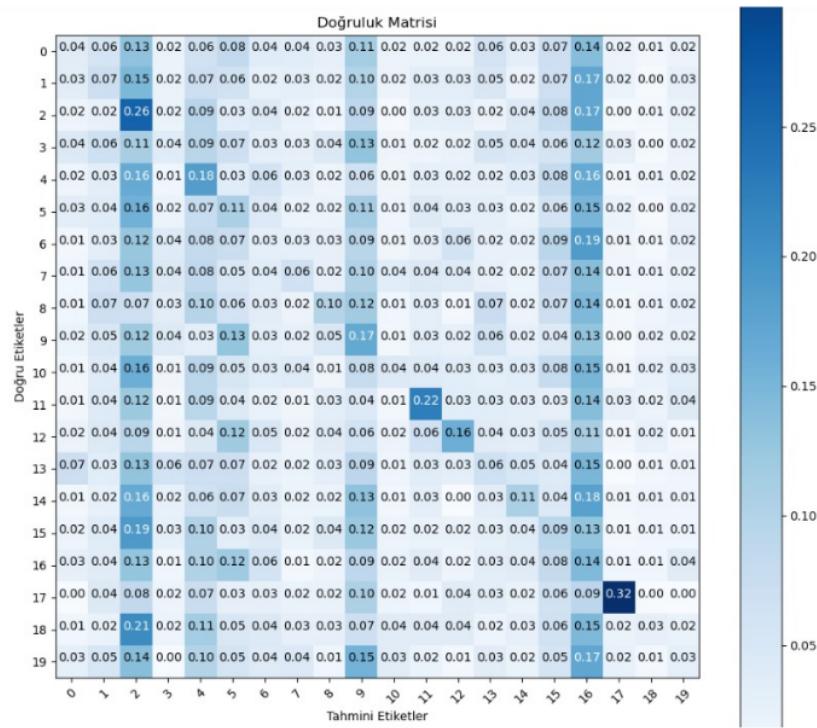
KNN yöntemi için sınıflandırma raporu Şekil 8.15'de gösterilmiştir.

Şekil 8.15 KNN ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.32	0.04	0.07	1614
1	0.16	0.07	0.09	792
2	0.18	0.26	0.22	785
3	0.09	0.04	0.05	433
4	0.11	0.18	0.14	377
5	0.07	0.11	0.09	347
6	0.03	0.03	0.03	307
7	0.09	0.06	0.07	284
8	0.13	0.10	0.11	269
9	0.05	0.17	0.08	251
10	0.08	0.04	0.06	232
11	0.20	0.22	0.21	229
12	0.16	0.16	0.16	219
13	0.04	0.06	0.05	202
14	0.09	0.11	0.10	207
15	0.03	0.09	0.05	187
16	0.03	0.14	0.04	196
17	0.39	0.32	0.35	192
18	0.07	0.03	0.04	193
19	0.03	0.03	0.03	182
accuracy			0.10	7498
macro avg	0.12	0.11	0.10	7498
weighted avg	0.16	0.10	0.10	7498

KNN yöntemi için doğrulama tablosu Tablo 8.14'de gösterilmiştir.

Tablo 8.14 KNN ROS Doğruluk tablosu



8.2.2.9 KNN-TFIDF

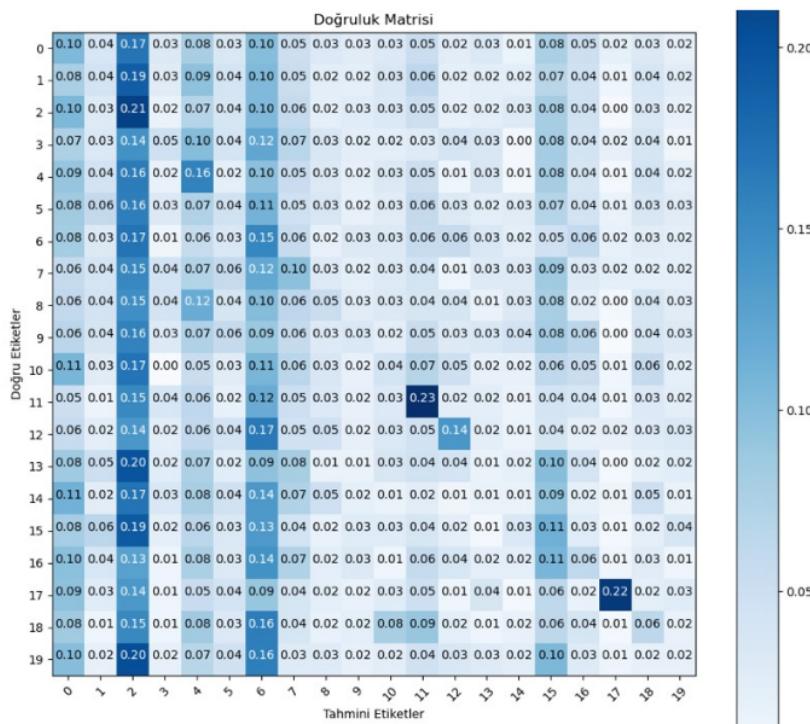
KNN yöntemiyle beraber TFIDF vektörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.16'da gösterilmiştir.

Şekil 8.16 KNN-TFIDF ROS Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.25	0.10	0.15	1600
1	0.12	0.04	0.06	788
2	0.13	0.21	0.16	784
3	0.11	0.05	0.06	432
4	0.10	0.16	0.12	376
5	0.06	0.04	0.05	346
6	0.06	0.15	0.08	306
7	0.07	0.10	0.08	282
8	0.07	0.05	0.06	268
9	0.05	0.03	0.04	249
10	0.04	0.04	0.04	231
11	0.12	0.23	0.16	225
12	0.14	0.14	0.14	218
13	0.02	0.01	0.02	202
14	0.02	0.01	0.02	206
15	0.04	0.11	0.06	187
16	0.04	0.06	0.05	196
17	0.33	0.22	0.26	190
18	0.05	0.06	0.05	192
19	0.02	0.02	0.02	182
accuracy			0.10	7460
macro avg	0.09	0.09	0.08	7460
weighted avg	0.13	0.10	0.10	7460

KNN yöntemiyle beraber TFIDF vekörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.15'de gösterilmiştir.

Tablo 8.15 KNN-TFIDF ROS Doğruluk tablosu



8.2.3 Çarpraz Doğrulama Skorları

8.2.3.1 Multinomial Naive Bayes

Veri seti Multinomial Naive Bayes ile Çarpraz Doğrulama'a sokulduğunda elde edilen sonuç Şekil 8.17'de gösterilmiştir.

Şekil 8.17 Multinomial Naive Bayes Çarpraz Doğrulama Skoru

```
In [18]: from sklearn.naive_bayes import MultinomialNB
NB_classifier = MultinomialNB()
scores = cross_val_score(NB_classifier, X_train_osm, y_train_osm, cv=5)
scores

Out[18]: array([0.69266732, 0.70155293, 0.70945429, 0.71598863, 0.71639873])
```

8.2.3.2 SVM

Veri seti MSVM ile Çarpraz Doğrulama'a sokulduğunda elde edilen sonuç Şekil 8.18'de gösterilmiştir.

Şekil 8.18 SVM Çarpraz Doğrulama Skoru

```
In [17]: from sklearn.model_selection import cross_val_score
from sklearn.svm import SVC
clf = SVC()
scores = cross_val_score(clf, X_train_osm, y_train_osm, cv=5)
scores

Out[17]: array([0.9113353 , 0.92156059, 0.93151247, 0.94206584, 0.94138233])
```

8.2.3.3 KNN

Veri seti KNN ile Çarpraz Doğrulama'a sokulduğunda elde edilen sonuç Şekil 8.19'da gösterilmiştir. Cross-Validation skorlarına bakıldığından, bölüntülenmiş başarımlar

Şekil 8.19 KNN Çarpraz Doğrulama Skoru

```
In [134]: knn_cv = cross_val_score(knn, X, y, cv=5)
knn_cv.mean()
knn_cv

Out[134]: array([0.1796539 , 0.18075423, 0.18565557 , 0.18535561, 0.17045114])
```

birbirlerine yakın çıkmıştır. Bu da aslında her yöntem için verinin kendi içerisinde homojen dağılığını ortaya koymaktadır.

8.2.4 SMOTE

8.2.4.1 Multinomial Naive Bayes

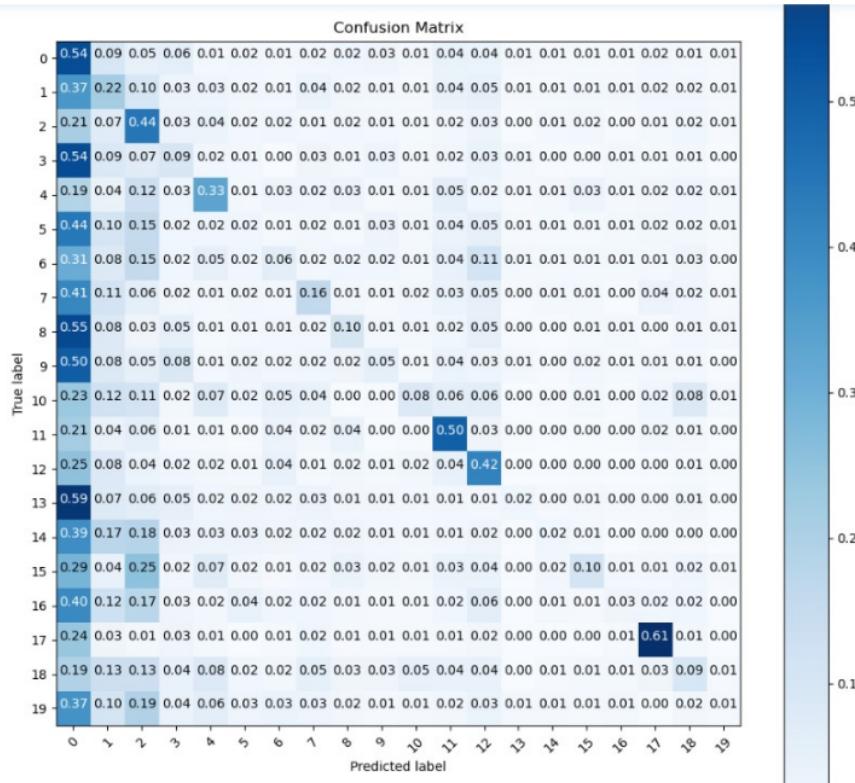
Multinomial Naive Bayes yöntemi için sınıflandırma raporu Şekil 8.20'de gösterilmiştir.

Şekil 8.20 Multinomial Naive Bayes SMOTE Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.30	0.54	0.39	1607
1	0.23	0.22	0.22	789
2	0.37	0.44	0.40	783
3	0.13	0.09	0.11	432
4	0.39	0.33	0.36	377
5	0.06	0.02	0.03	346
6	0.13	0.06	0.08	306
7	0.22	0.16	0.19	282
8	0.16	0.10	0.12	268
9	0.10	0.05	0.07	250
10	0.19	0.08	0.12	231
11	0.33	0.50	0.40	228
12	0.24	0.42	0.30	219
13	0.08	0.02	0.03	202
14	0.07	0.02	0.03	206
15	0.19	0.10	0.13	187
16	0.09	0.03	0.04	196
17	0.53	0.61	0.57	191
18	0.13	0.09	0.11	192
19	0.03	0.01	0.01	182
accuracy			0.27	7474
macro avg	0.20	0.19	0.19	7474
weighted avg	0.23	0.27	0.24	7474

Multinomial Naive Bayes yöntemi için doğrulama tablosu Tablo 8.16'da gösterilmiştir.

Tablo 8.16 Multinomial Naive Bayes SMOTE Doğruluk tablosu



8.2.4.2 SVM

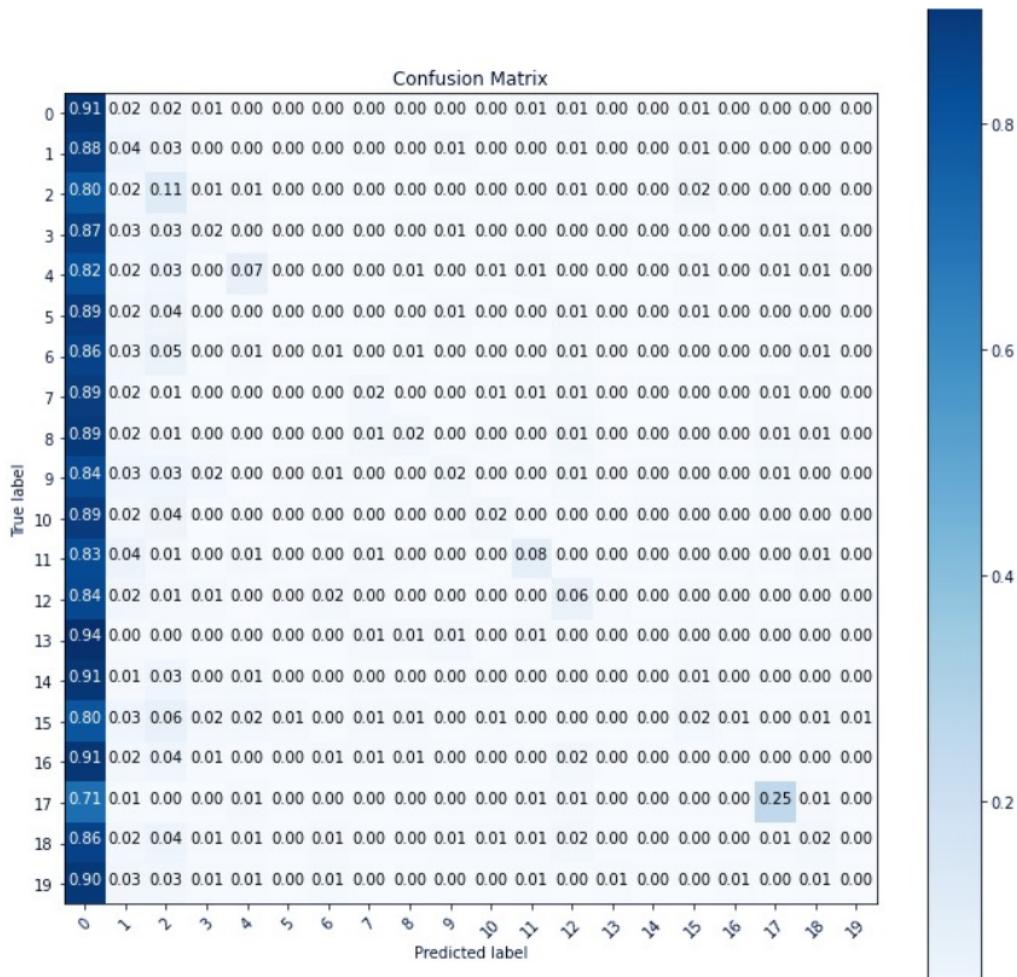
SVM yöntemi için sınıflandırma raporu Şekil 8.21'de gösterilmiştir.

Şekil 8.21 SVM SMOTE Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.22	0.91	0.36	1607
1	0.20	0.04	0.07	789
2	0.33	0.11	0.16	783
3	0.18	0.02	0.03	432
4	0.43	0.07	0.11	377
5	0.08	0.00	0.01	346
6	0.09	0.01	0.01	306
7	0.18	0.02	0.04	282
8	0.19	0.02	0.03	268
9	0.17	0.02	0.04	250
10	0.21	0.02	0.03	231
11	0.35	0.08	0.13	228
12	0.19	0.06	0.09	219
13	0.00	0.00	0.00	202
14	0.00	0.00	0.00	206
15	0.06	0.02	0.03	187
16	0.00	0.00	0.00	196
17	0.61	0.25	0.35	191
18	0.12	0.02	0.04	192
19	0.00	0.00	0.00	182
accuracy			0.23	7474
macro avg	0.18	0.08	0.08	7474
weighted avg	0.21	0.23	0.13	7474

SVM yöntemi için doğrulama tablosu Tablo 8.17'de gösterilmiştir.

Tablo 8.17 SVM SMOTE Doğruluk tablosu



8.2.4.3 KNN

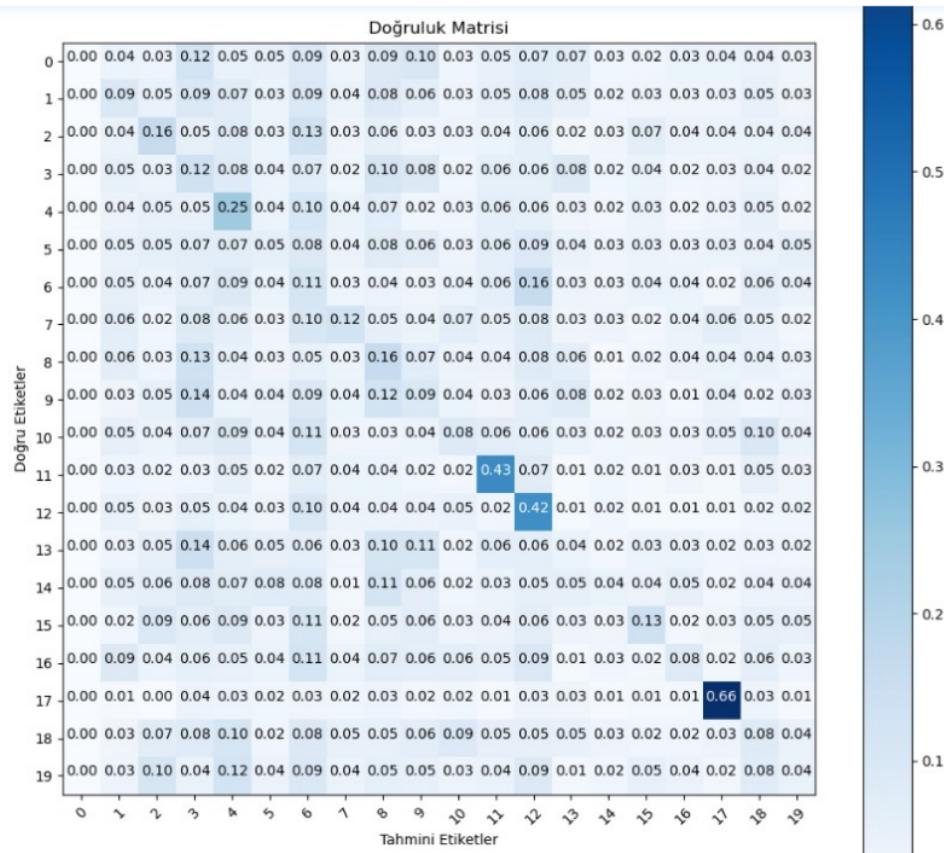
KNN yöntemiyle beraber TFIDF vektörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.22'de gösterilmiştir.

Şekil 8.22 KNN SMOTE Normal Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.00	0.00	0.00	1607
1	0.20	0.09	0.13	789
2	0.31	0.16	0.21	783
3	0.08	0.12	0.10	432
4	0.17	0.25	0.20	377
5	0.06	0.05	0.06	346
6	0.05	0.11	0.07	306
7	0.13	0.12	0.13	282
8	0.08	0.16	0.10	268
9	0.05	0.09	0.06	250
10	0.07	0.08	0.08	231
11	0.23	0.43	0.30	228
12	0.15	0.42	0.22	219
13	0.02	0.04	0.03	202
14	0.05	0.04	0.05	206
15	0.10	0.13	0.11	187
16	0.07	0.08	0.07	196
17	0.35	0.66	0.46	191
18	0.04	0.08	0.06	192
19	0.03	0.04	0.04	182
accuracy			0.12	7474
macro avg	0.11	0.16	0.12	7474
weighted avg	0.11	0.12	0.11	7474

KNN yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.18'de gösterilmiştir.

Tablo 8.18 KNN SMOTE Doğruluk tablosu



8.2.4.4 Multinomial Naive Bayes-TFIDF

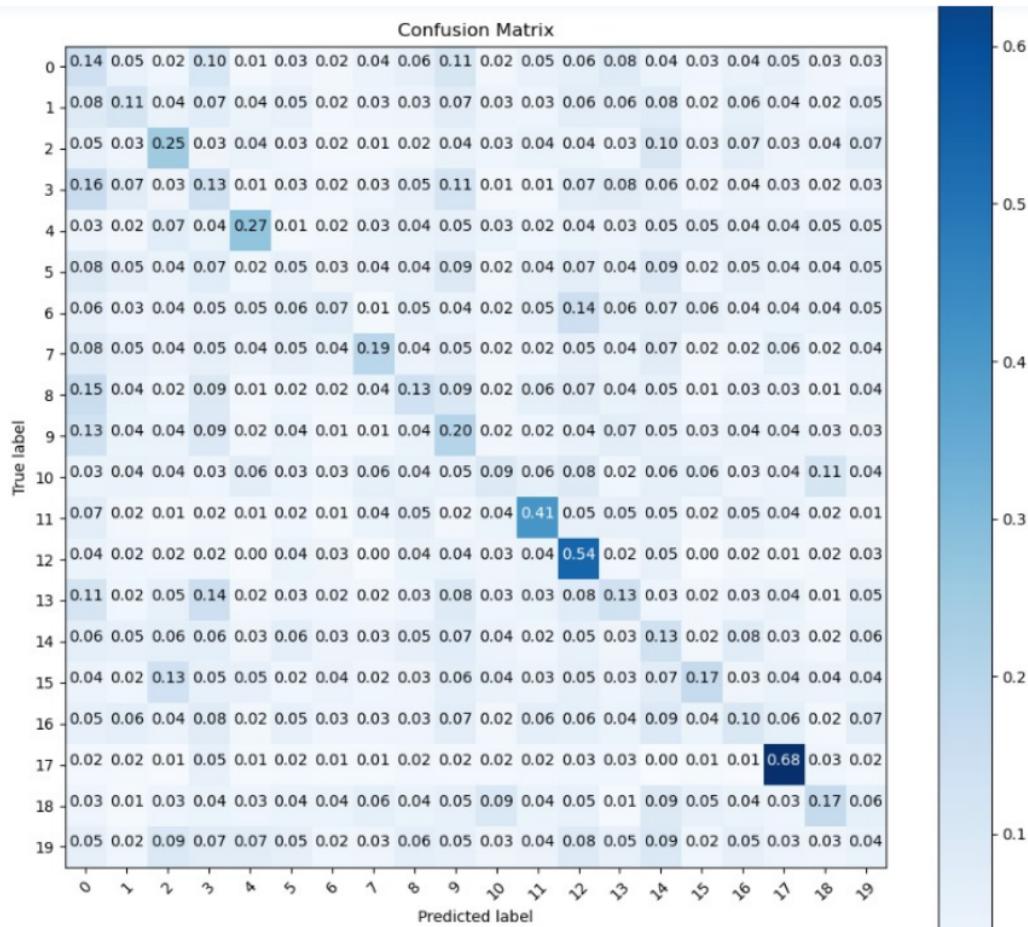
Multinomial Naive Bayes yöntemiyle beraber TFIDF vekötörize yöntemi kullanılmışlığı durumda sınıflandırma raporu Şekil 8.23'de gösterilmiştir.

Şekil 8.23 Multinomial Naive Bayes-TFIDF SMOTE Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.34	0.14	0.20	1600
1	0.25	0.11	0.16	788
2	0.43	0.25	0.32	784
3	0.11	0.13	0.12	432
4	0.35	0.27	0.30	376
5	0.07	0.05	0.06	346
6	0.12	0.07	0.09	306
7	0.20	0.19	0.20	282
8	0.11	0.13	0.12	268
9	0.09	0.20	0.12	249
10	0.09	0.09	0.09	231
11	0.25	0.41	0.31	225
12	0.21	0.54	0.30	218
13	0.07	0.13	0.09	202
14	0.06	0.13	0.08	206
15	0.14	0.17	0.15	187
16	0.06	0.10	0.07	196
17	0.31	0.68	0.42	190
18	0.13	0.17	0.14	192
19	0.03	0.04	0.03	182
accuracy			0.18	7460
macro avg	0.17	0.20	0.17	7460
weighted avg	0.23	0.18	0.18	7460

Multinomial Naive Bayes yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığında durumda doğrulama tablosu Tablo 8.19'da gösterilmiştir.

Tablo 8.19 Multinomial Naive Bayes-TFIDF SMOTE Doğruluk tablosu



8.2.4.5 KNN TF-IDF

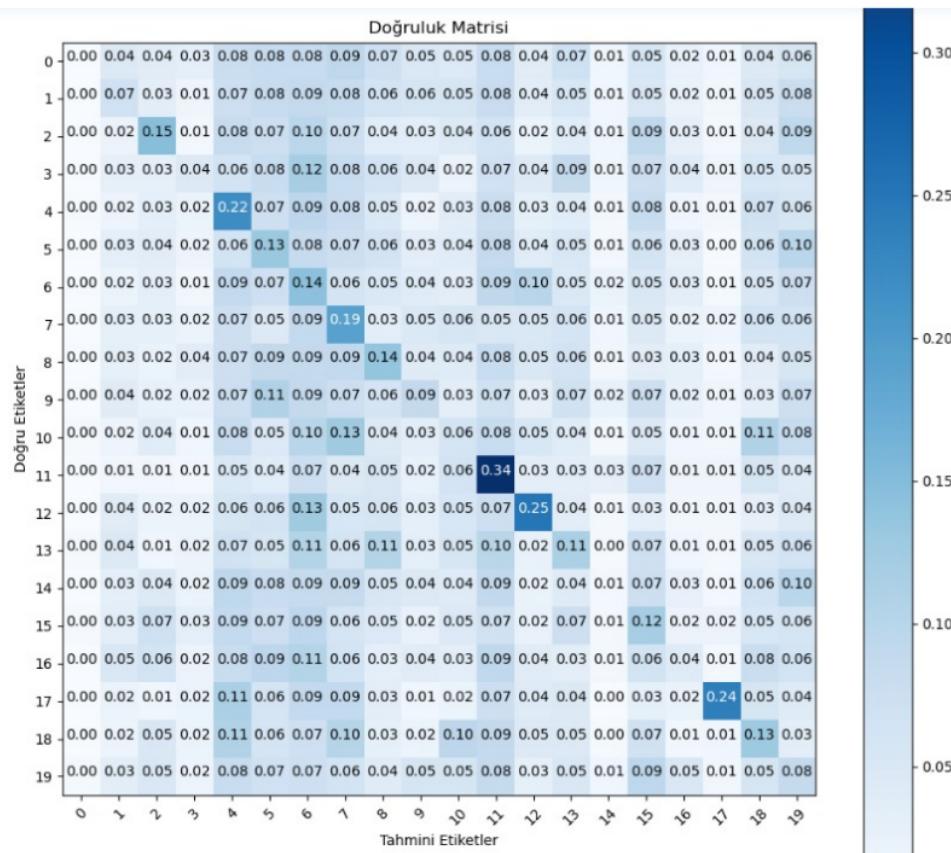
KNN yöntemiyle beraber TFIDF vektörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.24'de gösterilmiştir.

Şekil 8.24 KNN TF-IDF SMOTE Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.14	0.00	0.00	2128
1	0.20	0.07	0.10	1024
2	0.34	0.15	0.20	1062
3	0.12	0.04	0.06	589
4	0.13	0.22	0.16	487
5	0.08	0.13	0.10	476
6	0.06	0.14	0.09	400
7	0.09	0.19	0.12	374
8	0.09	0.14	0.11	390
9	0.07	0.09	0.08	335
10	0.04	0.06	0.05	299
11	0.12	0.34	0.18	294
12	0.16	0.25	0.20	285
13	0.05	0.11	0.07	255
14	0.02	0.01	0.01	269
15	0.05	0.12	0.07	261
16	0.05	0.04	0.04	265
17	0.40	0.24	0.30	273
18	0.06	0.13	0.08	242
19	0.03	0.08	0.04	238
accuracy			0.10	9946
macro avg	0.12	0.13	0.10	9946
weighted avg	0.14	0.10	0.09	9946

KNN yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.20'de gösterilmiştir.

Tablo 8.20 KNN TF-IDF SMOTE Doğruluk tablosu



8.2.4.6 SVM TF-IDF

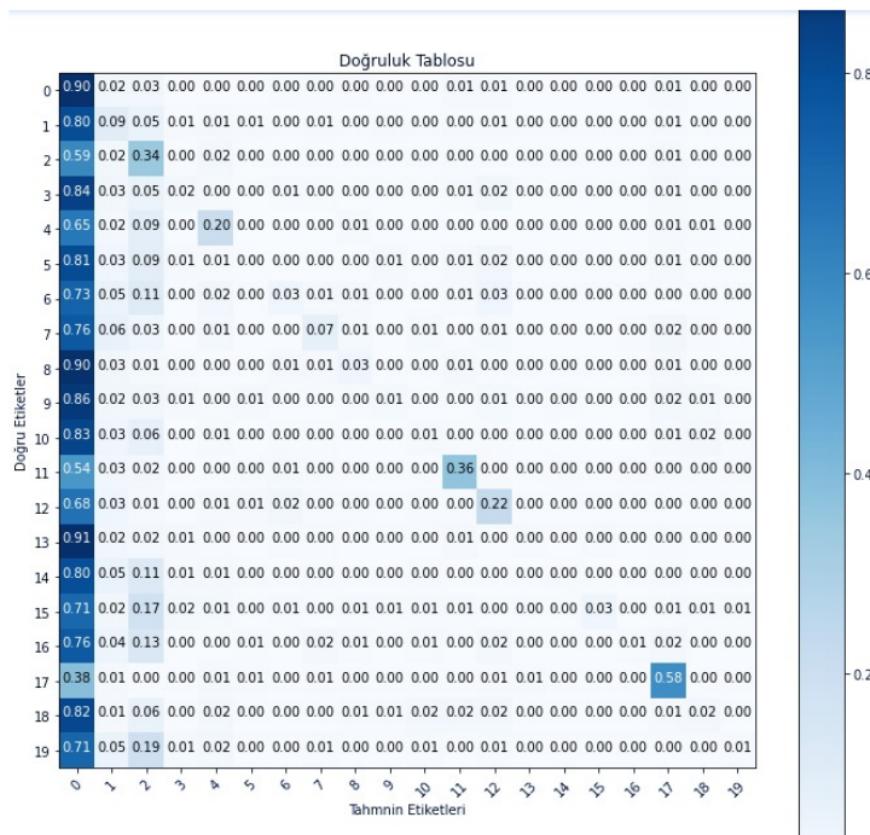
SVM yöntemiyle beraber TFIDF vekötörize yöntemi kullanılanlığı durumda sınıflandırma raporu Şekil 8.25'de gösterilmiştir.

Şekil 8.25 SVM TF-IDF SMOTE Sınıflandırma Raporu

	precision	recall	f1-score	support
0	0.25	0.90	0.39	1607
1	0.28	0.09	0.14	789
2	0.42	0.34	0.38	783
3	0.24	0.02	0.03	432
4	0.57	0.20	0.30	377
5	0.07	0.00	0.01	346
6	0.28	0.03	0.05	306
7	0.41	0.07	0.13	282
8	0.24	0.03	0.05	268
9	0.19	0.01	0.02	250
10	0.11	0.01	0.02	231
11	0.67	0.36	0.47	228
12	0.43	0.22	0.29	219
13	0.25	0.00	0.01	202
14	0.25	0.00	0.01	206
15	0.46	0.03	0.06	187
16	0.25	0.01	0.01	196
17	0.62	0.58	0.60	191
18	0.17	0.02	0.04	192
19	0.14	0.01	0.01	182
accuracy			0.29	7474
macro avg	0.32	0.15	0.15	7474
weighted avg	0.31	0.29	0.21	7474

SVM yöntemiyle beraber TFIDF vekötörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.21'de gösterilmiştir.

Tablo 8.21 SVM TF-IDF SMOTE Doğruluk tablosu



8.3 Derin Öğrenme

DL yöntemlerinin sonuçları yer almaktadır.

8.3.1 Normal Veri Kümesi Sonuçları

8.3.1.1 LSTM

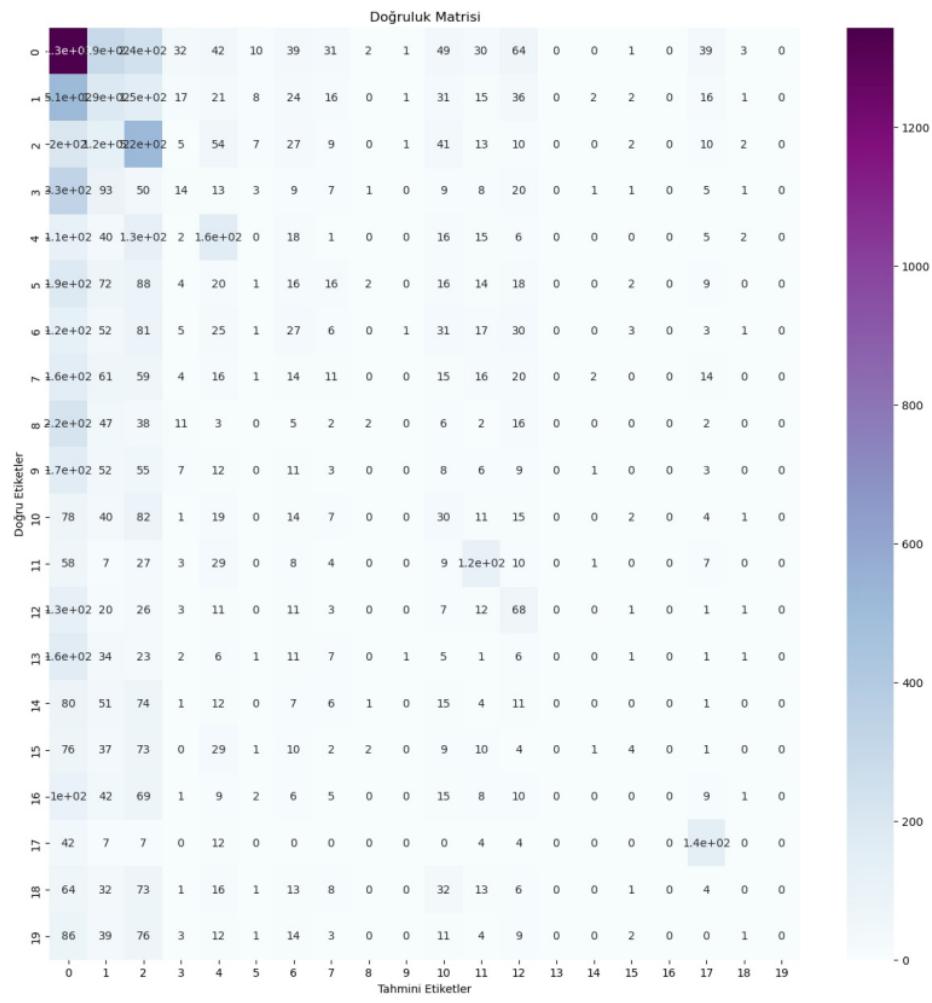
LSTM yöntemi için sınıflandırma raporu Şekil 8.26'da gösterilmiştir.

Şekil 8.26 LSTM Normal Veri Kümesi Sınıflandırma Raporu

313/313 [=====] – 5s 13ms/step				
	precision	recall	f1-score	support
0	0.32	0.61	0.42	2219
1	0.14	0.18	0.16	1047
2	0.27	0.50	0.35	1023
3	0.12	0.02	0.04	562
4	0.31	0.32	0.32	507
5	0.03	0.00	0.00	464
6	0.10	0.07	0.08	405
7	0.07	0.03	0.04	389
8	0.20	0.01	0.01	359
9	0.00	0.00	0.00	333
10	0.08	0.10	0.09	304
11	0.37	0.42	0.39	282
12	0.18	0.23	0.20	294
13	0.00	0.00	0.00	265
14	0.00	0.00	0.00	263
15	0.18	0.02	0.03	259
16	0.00	0.00	0.00	278
17	0.52	0.65	0.58	219
18	0.00	0.00	0.00	264
19	0.00	0.00	0.00	261
accuracy			0.26	9997
macro avg		0.14	0.16	9997
weighted avg		0.18	0.26	9997

LSTM yöntemi için doğrulama tablosu Tablo 8.22'de gösterilmiştir.

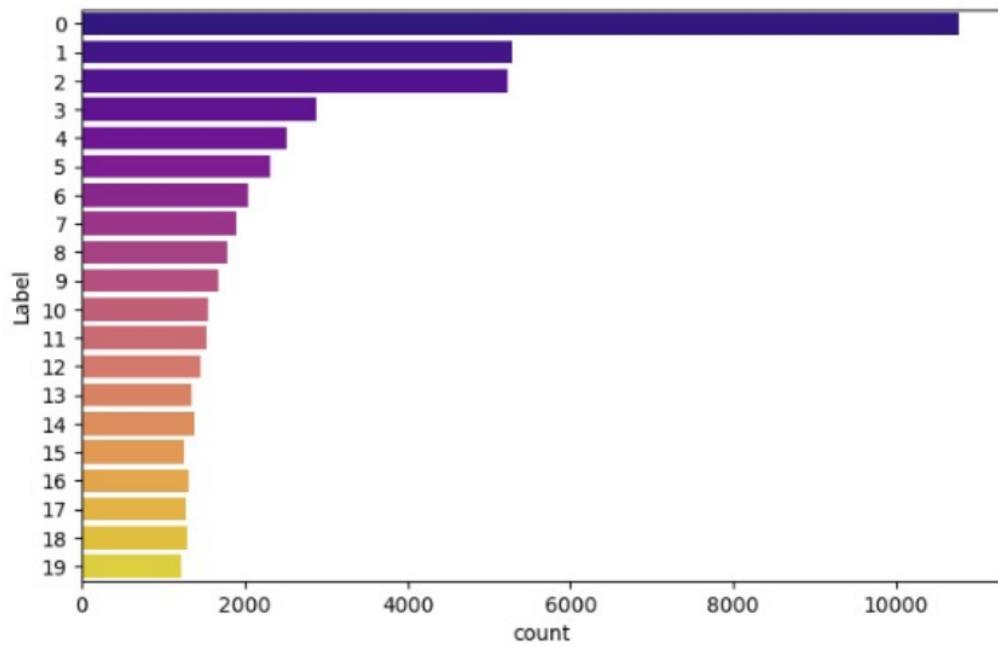
Tablo 8.22 LSTM Normal Veri Kümesi Doğruluk tablosu



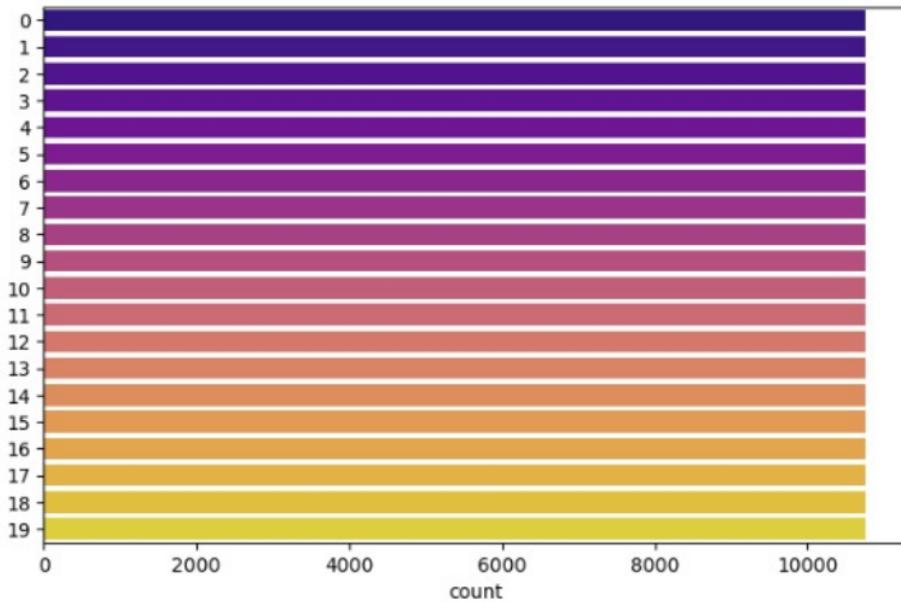
8.3.2 ROS Sonuçları

Veriye ROS uygulamadan önceki durumu Şekil 8.27'de gösterilmiştir. Veriye ROS uyguladıktan sonraki durumu Şekil 8.28'de gösterilmiştir.

Şekil 8.27 Normal Emoji-Veri Sayıları



Şekil 8.28 ROS Uygulanan Emoji-Veri Sayıları



8.3.2.1 LSTM

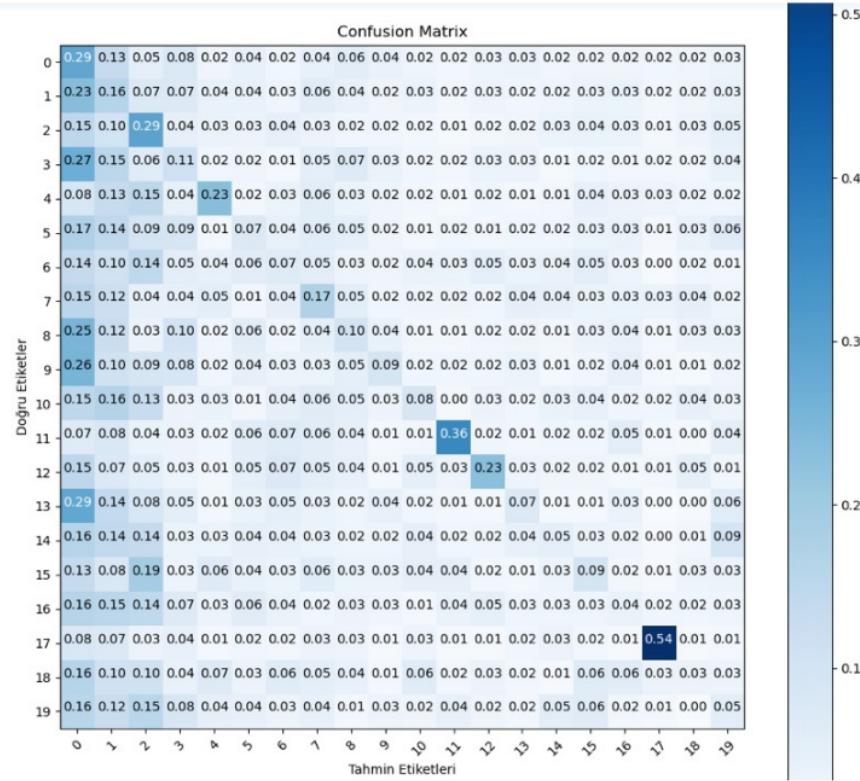
LSTM yöntemi için sınıflandırma raporu Şekil 8.29'da gösterilmiştir.

Şekil 8.29 LSTM ROS Sınıflandırma Raporu

234/234 [=====] - 5s 16ms/step				
	precision	recall	f1-score	support
0	0.31	0.29	0.30	1607
1	0.14	0.16	0.15	789
2	0.30	0.29	0.30	783
3	0.10	0.11	0.10	432
4	0.29	0.23	0.26	377
5	0.08	0.07	0.08	346
6	0.08	0.07	0.07	306
7	0.13	0.17	0.15	282
8	0.08	0.10	0.09	268
9	0.10	0.09	0.10	250
10	0.10	0.08	0.09	231
11	0.39	0.36	0.37	228
12	0.22	0.23	0.23	219
13	0.08	0.07	0.08	202
14	0.06	0.05	0.05	206
15	0.07	0.09	0.08	187
16	0.03	0.04	0.04	196
17	0.48	0.54	0.51	191
18	0.04	0.03	0.03	192
19	0.04	0.05	0.04	182
accuracy			0.19	7474
macro avg	0.16	0.16	0.16	7474
weighted avg	0.19	0.19	0.19	7474

LSTM yöntemi için doğrulama tablosu Tablo 8.23'de gösterilmiştir.

Tablo 8.23 LSTM ROS Doğruluk tablosu



9

Sonuç

Bu bölüm, kullanılan yöntemlerin nihai sonuçlarının karşılaştırmalı şekilde yorumlanması içermektedir.

9.1 ROS Sonuçları

9.1.1 CV

Yöntemlerin CV vektörize yöntemiyle ROS f-1 score sonuçları Tablo 9.1'de verilmiştir.

RANDOM OS	SVM (%)	M. Naive Bayes (%)	KNN (%)
0	38	17	7
1	22	17	9
2	38	34	22
3	9	15	5
4	37	37	14
5	7	10	9
6	9	11	3
7	13	16	7
8	13	14	11
9	8	13	8
10	11	14	6
11	41	35	21
12	32	31	16
13	3	9	5
14	1	7	10
15	11	14	5
16	3	6	4
17	62	51	35
18	7	12	4
19	0	2	3

Tablo 9.1 CV f-1 Score Sonuçları

Tablo 9.1'de de görüldüğü gibi ML algoritmaları arasındaki en yüksek başarımın SVM'de olduğu açıkça görülmektedir. Onun hemen ardından M. Naive Bayes gelmektedir. Öncelikle ML yöntemlerinin en başarılı olduğu emojiler sırasıyla 17 (Chirtmas Tree), 11 (United States), 0 (Red Heart), 2 (Face With Tears of Joy), 4 (Fire) ve 12 (Sun) numaralı emojilerdir. Bu emojilere baktığımızda 17 numaralı emoji diğer emojilere göre daha özel kelimelerle tanımlanabilmektedir. Bundan dolayı tahmin başarımı olarak en yüksek emoji bu emoji olmuştur. Bu emoji dikkate alındığında SVM ve M. Naive Bayes'in başarımı anlamındaki fark düzeyi, KNN yöntemine kıyasla oldukça azdır. 11 numaralı emojiye baktığımızdaysa, yine benzer şekilde bayrakla aynı bağlamda bulunan kelimeler görece daha özeldir. Buradaysa ML yöntemleri arasındaki başarım farkı görece azalmıştır. 0 numaralı emojiyse aslında veri setinde en fazla oranda bulunan emojidir ve bu emojiideki başarım büyük oranda ağırlıklı f-1 skora etki etmektedir. Buradaysa SVM yöntemi diğer yöntemlere göre oldukça yüksek başarım yakalamıştır. M. Naive Bayes yöntemi ise, KNN yöntemine göre oldukça başarılıdır. Aynı zamanda KNN yönteminin en başarısız olduğu emojilerden birisi 0 numaralı emojidir. 2 numaralı emojiye geldiğimizde, ML yöntemleri arasındaki başarım makasının daraldığı görülmektedir. Bu emojiye yöntemlerin başarılı olmasının en büyük sebebiyse, veri setimizde bu emojiye yakın emoji sayısının az olması ve 2 numaralı emojiye yakın olan emojilerin veri setinde görece çok az olmasıdır. Aynı zamanda 17 numaralı emojiyi saymazsa KNN yönteminin tahminlemede en başarılı olduğu emoji 2 numaralı emoji denilebilir. 4 numaralı emojiye gelindiğinde yöntemler arasındaki en başarılı yöntem olan SVM ile M. Naive Bayes yöntemlerinin sonuçları başa baş gitmektedir. KNN yöntemi ise yine görece daha başarısızdır. Yine yöntemlerin başarısındaki sebep, veri setinde bu emojiye yakın anlamlı emojilerin az olmasıdır. 12 numaralı emojiyeye yine her yöntem, diğer emojilerle başarımlarına kıyasla, iyi bir başarım elde etmiştir. Bundaki en büyük etken benzer emojilerin bulunmamasıdır.

9.1.2 TF-IDF

Yöntemlerin TD-IDF vektörize yöntemiyle ROS f-1 score sonuçları Tablo 9.2'de verilmiştir. Tablo 9.2'de de görüldüğü gibi ML algoritmaları arasındaki en yüksek başarımın SVM'de olduğu yine açıkça görülmektedir. Onun hemen ardından M. Naive Bayes gelmektedir. ML yöntemlerinin en başarılı olduğu emojiler CV vektörize yöntemine benzer şekilde sırasıyla 17 (Chirtmas Tree), 11 (United States), 2 (Face With Tears of Joy), 0 (Red Heart), 12 (Sun) ve 4 (Fire) numaralı emojilerdir. 17 numaralı emoji dikkate alındığında SVM ve M. Naive Bayes'in başarımı anlamındaki fark düzeyi, M. Naive Bayes yöntemiyle KNN yöntemi arasındaki farka kıyasla daha fazladır. 11 numaralı emojiyeye ML yöntemleri arasındaki başarım farkı görece

RANDOM OS	SVM (%)	M. Naive Bayes (%)	KNN (%)
0	38	11	15
1	19	17	6
2	43	33	16
3	8	12	6
4	32	30	12
5	3	9	5
6	6	12	8
7	13	18	8
8	10	12	6
9	6	13	4
10	3	8	4
11	46	27	16
12	35	30	14
13	5	9	2
14	1	7	2
15	5	11	6
16	0	8	5
17	64	42	26
18	9	13	5
19	1	4	2

Tablo 9.2 TF-IDF f-1 Score Sonuçları

azalmıştır. 2 numaralı emojiye geldiğimizde, SVM ve M. Naive Bayes yöntemleri arasındaki başarıım makasının daraldığı görülmektedir. Aynı zamanda 17 numaralı emojiyi saymazsaKNN yönteminin tahminlemede en başarılı olduğu emoji 2 numaralı emoji denilebilir. 0 numaralı emojiyedeyse SVM yöntemi diğer yöntemlere göre oldukça yüksek başarıım yakalamıştır. KNN yöntemi ise, M. Naive Bayes yöntemine göre oldukça başarılıdır. 12 numaralı emojiyedeyse yine her yöntem, diğer emojilerle başarımlarına kıyasla, iyi bir başarıım elde etmiştir. 4 numaralı emojiye gelindiğinde yöntemler arasındaki en başarılı yöntem olan SVM ile M. Naive Bayes yöntemlerinin sonuçları yine çok yakın çıkmıştır. KNN yöntemi ise oldukça daha başarısızdır.

9.2 Genel Doğruluk Oranları

Yöntemlerin test edilen tüm durumlardaki accuracy sonuçları Tablo 9.3'de verilmiştir.

ACCURACY	NORMAL (%)		RANDOM OS (%)		SMOTE (%)	
	CV	TF-IDF	CV	TF-IDF	CV	TF-IDF
SVM	29	29	28	29	23	29
M. Naive Bayes	27	24	19	17	27	18
LSTM	26		19			
KNN	18	19	10	10	12	10

Tablo 9.3 Accuracy Sonuçları

Tabloya bakıldığından, başarımlar olarak en başarılı yöntem ML algoritması olan SVM'dir. Makine öğrenmesi yöntemlerini kıyaslayacak olursak, aralarında en başarılı olan yöntem normal veri setinde ufak bir farkla SVM'dir. Normal veri setinde LSTM, M. Naive Bayes ve SVM yöntemlerinin sonuçları çok yakın çıkmıştır. Aynı zamanda her yöntemin tahmin etmekte en başarılı olduğu emoji 17 numaralı emojidir. Bunun sebebi, emojinin çok özel kelimelerle ifade edilebiliyor olmasıdır. SVM yöntemi ROS, SMOTE ve normal veri kümesi için yakın başarımlar verirken, M. Naive Bayes ise ROS sonuçlarına bakıldığından çok düşük başarımları almıştır. KNN ise hem SMOTE için hem ROS için normal duruma göre çok düşük başarımlar vermiştir. LSTM, ROS yapılmış veri setiyle eğitildikten sonra normal veri setine göre daha düşük başarımlar vermiştir. ROS ve SMOTE teknikleri uygulandıktan sonra genel olarak başarımların düşmesinin sebebi veri test ettiğimiz veri setinde büyük oranda 0 numaralı emoji bulunmasıdır. Gerçek dünyadan alınan ve heterojen dağılan emoji oranları, yapay şekilde eşitlendiğinde homojenize edilmiş veri setiyle modeller teste tabi tutulduğunda tahminleme oranı daha homojen şekilde olduğu için en yüksek orana sahip 0 numaralı emojiyi modeller daha az tahmin etmiştir. Bundan dolayı, veri kümesinde en fazla bulunan 0 numaralı emoji tahmin sayısı azalınca doğrudan elde edilen sonuçları da aşağı çekmiştir.

9.3 Vektörize Yöntemleri Kiyaslama

Yöntemlerin vektörize yöntemlerine göre 4 emoji için f-1 skor sonuçları Tablo 9.4'de verilmiştir.

Emoji	4 Fire		11 United States		12 Sun		17 Chirtmas Tree	
Yöntem	CV	TF-IDF	CV	TF-IDF	CV	TF-IDF	CV	TF-IDF
SVM (%)	37	32	41	46	32	35	62	64
M. Naive Bayes (%)	37	30	35	27	31	30	51	42
KNN (%)	14	12	21	16	16	14	35	26

Tablo 9.4 f-1 Skor Sonuçları

4 numaralı emojiyi dikkate aldığımızda, M. Naive Bayes yönteminin aldığı başarım TF-IDF vektörize yöntemiyle, CV yöntemine göre çok düşmüştür. M. Naive Bayes gibi, SVM ve KNN yöntemlerinde de başarım oldukça düşmüştür. 4 numaralı emojiyi tespit etmenin önemli olduğu bir durumda, CV vektörize yöntemi tercih edilebilir. 11 numaralı emojiye bakıldığında SVM yöntemiyle birlikte TF-IDF vektörize yöntemi kullanıldığından başarı %5 artmıştır. Diğer yöntemlerdeyse ciddi bir düşüş söz konusudur. 12 numaralı emojiye bakıldığımızda yine SVM ile TF-IDF vektörze yöntemi kullanıldığından başarı ufak da olsa artmıştır. M. Naive Bayes yöntemi ise hem CV, hem de TF-IDF vektörize yöntemleri için yakın sonuçlar vermiştir. KNN yöntemi de benzer şekilde gözüksé de, başarım oranı oldukça düşmüştür. Benzer şekilde 17 numaralı emojiye de SVM ve TF-IDF ikilisi daha iyi başarım verirken, diğer yöntemler CV ile daha yüksek başarım vermiştir. Bu dört emojiyi dikkate alacak olursak, SVM ile birlikte TF-IDF vektörize yöntemini kullanmak daha iyi olacaktır denilebilir. Aynı zamanda KNN ve M. Naive Bayes yöntemleriyle de CV vektörize yöntemi kullanmak daha iyi sonuç verir denilebilir.

Referanslar

- [1] S. F. Taşkıran, "Doğal dil işleme ile akademik metinlerin kümelenmesi," M.S. thesis, Konya Teknik Üniversitesi, 2021.
- [2] A. E. ÖZMUTLU, "Doğal dil İşleme," *Bilgisayar Bilimlerinde Teorik Ve Uygulamalı Araştırmalar*, p. 129,
- [3] A. Haberturk, *İnternette 1 dakikada neler oluyor?* <https://www.haberturk.com/internette-1-dakikada-neler-oluyor-3148965/>, [Online; accessed 25 November 2022], 2021.
- [4] Y. Kocababaş, "Etkili iletişim sözsüz adımı olan beden dili ve türkçe eğitimindeki rolü," *HAYEF Journal of Education*, vol. 4, no. 1,
- [5] D. Jurafsky and J. H. Martin, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*.
- [6] E. Saravia, H.-C. T. Liu, Y.-H. Huang, J. Wu, and Y.-S. Chen, "CARER: Contextualized affect representations for emotion recognition," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 3687–3697. DOI: 10.18653/v1/D18-1404. [Online]. Available: <https://aclanthology.org/D18-1404>.
- [7] F. Barbieri *et al.*, "Semeval 2018 task 2: Multilingual emoji prediction," in *Proceedings of the 12th international workshop on semantic evaluation*, 2018, pp. 24–33.
- [8] R. DiPietro and G. D. Hager, "Deep learning: Rnns and lstm," in *Handbook of medical image computing and computer assisted intervention*, Elsevier, 2020, pp. 503–519.
- [9] M. Bozdemir, "Makine çevirisi ile türkçe sözel ifadelerin python sözdiziminin oluşturulması," M.S. thesis, Bursa Uludağ Üniversitesi, 2022.
- [10] E. Günce and A. Carus, "Twitter platformu üzerinden makine öğrenmesi algoritmaları ile cinsiyet ve ilgi alanı analizi," M.S. thesis, Trakya Üniversitesi Fen Bilimleri Enstitüsü, 2021.
- [11] T. Demirhan, "Makine öğrenmesi algoritmalarının karmaşıklık ve doygunluk analizinin bir veri kümesi üzerinde gerçekleştirilmesi," 2015.

Özgeçmiş

BİRİNCİ ÜYE

İsim-Soyisim: Muhammet Ali ŞEN
Doğum Tarihi ve Yeri: 01.10.1989 İstanbul
E-mail: ali.sen@std.yildiz.edu.tr
Telefon: 0541 338 10 19
Staj Tecrübeleri: Mobil Programlama

İKİNCİ ÜYE

İsim-Soyisim: Muhammet Kayra BULUT
Doğum Tarihi ve Yeri: 07.11.2000 , Kayseri
E-mail: kayrabulut39@gmail.com
Telefon: 0552 477 27 33
Staj Tecrübeleri: Mobillium

Proje Sistem Bilgileri

Sistem ve Yazılım: Windows Operation System, Python
Gerekli RAM: 16GB
Gerekli Disk: 4096MB

ORIGINALITY REPORT



PRIMARY SOURCES

Rank	Source	Type	Similarity (%)
1	powerpoetry.org	Internet Source	9%
2	lib.unnes.ac.id	Internet Source	<1 %
3	"A Selection of Metaheuristics and Various Projects", Bioinformatics, 2008	Publication	<1 %
4	www.tcetmumbai.in	Internet Source	<1 %
5	repository.ihu.edu.gr	Internet Source	<1 %

Exclude quotes On

Exclude bibliography Off

Exclude matches Off