

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BİR DUYGU ANALİZ YÖNTEMİ OLARAK METİNDE
EMOJİ TAHMİNİ

20011701 – Muhammet Ali ŞEN

20011901 – Muhammet Kayra BULUT

BİLGİSAYAR PROJESİ

Danışman
Doç. Dr. Ali Can Karaca

Aralık, 2022

TEŞEKKÜR

Öncelikle danışmanlığımızı üstlenen, konu seçiminden araştırmmanın yürütülmesine dek bizleri sınırlamayıp özgür bırakın, bir bilgisayar mühendisliği öğrencisinin yetkinliğini göstermesine olanak tanıyan, bunun yanında bizlere rehberlik yaparak karşılaşlığımız problemleri aşmada bize yol gösteren Doç. Dr. Ali Can Karaca hocama teşekkürlerimi sunarım.

Bununla birlikte Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği bölümündeki tüm hocalarına bizlerin yetişmesindeki emekleri ayrı ayrı teşekkür ediyoruz.

Muhammet Ali ŞEN
Muhammet Kayra BULUT

İÇİNDEKİLER

KISALTMA LİSTESİ	vi
ŞEKİL LİSTESİ	vii
TABLO LİSTESİ	ix
ÖZET	xi
ABSTRACT	xii
1 Giriş	1
1.1 Doğal Dil İşleme	1
1.2 Doğal Dil İşleme Uygulamaları	4
1.3 Projenin Amacı	6
1.4 Benzer Çalışmalar	9
1.5 Veri Seti	10
2 Ön İnceleme	12
2.1 Projeye Olan İhtiyaç	12
2.2 Proje Kapsamı	12
2.3 Projenin Gereksinimleri	13
3 Fizibilite	14
3.1 Teknik Fizibilite	14
3.1.1 Yazılım Fizibilitesi	14
3.1.2 Donanım Fizibilitesi	14
3.2 İş Gücü ve Zaman Fizibilitesi	15
3.3 Ekonomik Fizibilite	15
3.4 Yasal Fizibilte	16
4 Sistem Analizi	17
4.1 Gereksinimler	17
4.2 Hedefler	18
4.3 Performans Metrikleri	19

5 Sistem Tasarımı	20
5.1 Yazılım Tasarımı	20
5.1.1 LSTM	21
5.1.2 Naive Bayes	22
5.1.3 KNN	22
5.1.4 Destek Vektör Makineleri	23
6 Uygulama	24
6.1 Modellerin Özellikleri	24
6.2 Modellerin Eğitilmesi İçin Kullanılan Yöntem	26
6.3 Modellerin Eğitilmesi	26
7 Deneysel Sonuçlar	27
7.1 Veri Seti	27
7.1.1 Veri Setinin Boyutunu Değiştirme	27
7.1.2 Veri Setinde Oran Değiştirme	28
7.2 Sonuç	28
8 Performans Analizi	29
8.1 TF-IDF ile CV Arasındaki Farklar	29
8.2 Makine Öğrenmesi	30
8.2.1 Normal Veri Kümesi Sonuçları	30
8.2.2 ROS Sonuçları	42
8.2.3 SMOTE	61
8.2.4 Çapraz Doğrulama (Cross-Validation) Skorları	73
8.3 Derin Öğrenme	75
8.3.1 Normal Veri Kümesi Sonuçları	76
8.3.2 ROS Sonuçları	78
9 Sonuç	81
9.1 Normal Sonuçlar	81
9.1.1 CV	81
9.1.2 TF-IDF	83
9.2 ROS Sonuçları	84
9.2.1 CV	84
9.2.2 TF-IDF	85
9.3 SMOTE Sonuçları	87
9.3.1 CV	87
9.3.2 TF-IDF	88
9.4 Genel Doğruluk Oranları	89

9.5 Vektörize Yöntemleri Kiyaslama	90
Referanslar	91
Özgeçmiş	93

KISALTMA LİSTESİ

AI	Yapay Zeka (Artificial Intelligence)
CL	Bilgisayarlı Dilbilim (Computational Linguistics)
CNN	Evrişimsel Sinir Ağı (Convolutional Neural Network)
CPU	Merkez İşlem Ünitesi (Central Processing Unit)
CV	Vektör Sayacı (Counter Vektorizer)
DDİ	Doğal Dil İşleme (Natural Language Processing)
DL	Derin Öğrenme (Deep Learning)
GPU	Graif İşlem Ünitesi (Graphics Processing Unit)
KNN	K-En Yakın Komşu (K-Nearest Neighbours)
LSTM	Uzun Kısa Süreli Bellek (Long Short-Term Memory)
ML	Makine Öğrenmesi (Machine Learling)
MNB	Çok Terimli Naive Bayes (Multinomial Naive Bayes)
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)
RNN	Yinelenen Sinir Ağları (Recurrent Neural Network)
SMOTE	Yapay Azınlık Aşırı Örnemleme (Synthetic Minority Oversampling Technique)
SVM	Destek Vektör Makineleri (Support Vector Machine)
TF-IDF	Terim Sıklığı - Ters Doküman Sıklığı (Term Frequency–Inverse Document Frequency)
TPU	Tensor İşlem Üniteleri (Tensor Processing Units)

ŞEKİL LİSTESİ

Şekil 1.1 NLP Aşamaları	2
Şekil 1.2 Temizlenmemiş Veri	10
Şekil 1.3 Temizlenmiş Veri	10
Şekil 1.4 Emoji Dağılımı	11
Şekil 4.1 Örnek Emoji Oranları	17
Şekil 4.2 Şekil 4.1'in Emoji Karşılıkları	18
Şekil 5.1 Metin Sınıflandırma Akış Şeması	20
Şekil 8.1 MNB Normal Sınıflandırma Raporu	30
Şekil 8.2 MNB-TFIDF Normal Sınıflandırma Raporu	32
Şekil 8.3 SVM Normal Sınıflandırma Raporu	34
Şekil 8.4 SVM TF-IDF Normal Sınıflandırma Raporu	36
Şekil 8.5 KNN Normal Sınıflandırma Raporu	38
Şekil 8.6 KNN-TFIDF Normal Sınıflandırma Raporu	40
Şekil 8.7 ROS Veri Seti Durumu	42
Şekil 8.8 Multinomial Naive Bayes ROS Sınıflandırma Raporu	43
Şekil 8.9 MNB TFIDF ROS Sınıflandırma Raporu	45
Şekil 8.10 SVM ROS Sınıflandırma Raporu	48
Şekil 8.11 SVM Kernel POLY ROS Sınıflandırma Raporu	50
Şekil 8.12 SVM Kernel RBF ROS Sınıflandırma Raporu	52
Şekil 8.13 SVM'in kernel parametresi olarak SIGMOID ROS Sınıflandırma Raporu	54
Şekil 8.14 SVM TF-IDF ROS Sınıflandırma Raporu	56
Şekil 8.15 KNN ROS Sınıflandırma Raporu	58
Şekil 8.16 KNN-TFIDF ROS Sınıflandırma Raporu	60
Şekil 8.17 Multinomial Naive Bayes SMOTE Sınıflandırma Raporu	62
Şekil 8.18 MNB TF-IDF SMOTE Sınıflandırma Raporu	64
Şekil 8.19 SVM CV SMOTE Sınıflandırma Raporu	66
Şekil 8.20 SVM TF-IDF SMOTE Sınıflandırma Raporu	68
Şekil 8.21 KNN CV SMOTE Normal Sınıflandırma Raporu	70
Şekil 8.22 KNN TF-IDF SMOTE Sınıflandırma Raporu	72
Şekil 8.23 Multinomial Naive Bayes Çapraz Doğrulama Skoru	73

Şekil 8.24 SVM Çapraz Doğrulama Skoru	74
Şekil 8.25 KNN Çapraz Doğrulama Skoru	74
Şekil 8.26 LSTM Normal Veri Kümesi Sınıflandırma Raporu	76
Şekil 8.27 Normal Emoji-Veri Sayıları	78
Şekil 8.28 ROS Uygulanan Emoji-Veri Sayıları	79
Şekil 8.29 LSTM ROS Sınıflandırma Raporu	79

TABLO LİSTESİ

Tablo 1.1 Dijital Mecralarda Dakikalık Üretilen İçerik Sayısı	7
Tablo 3.1 Proje İş / Zaman Çizelgesi	15
Tablo 3.2 Ekonomik Fizibilite Tablosu	16
Tablo 6.1 Naive Bayes ile elde edilen başarım	25
Tablo 6.2 SVM ile elde edilen başarım	25
Tablo 7.1 Yöntem Kiyaslama tablosu	27
Tablo 7.2 Yöntem Kiyaslama tablosu	28
Tablo 8.1 MNB Normal Doğruluk tablosu	31
Tablo 8.2 MNB-TFIDF Normal Doğruluk tablosu	33
Tablo 8.3 SVM Normal Doğruluk tablosu	35
Tablo 8.4 SVM TF-IDF Normal Doğruluk tablosu	37
Tablo 8.5 KNN Normal Doğruluk tablosu	39
Tablo 8.6 KNN-TFIDF Normal Doğruluk tablosu	41
Tablo 8.7 Multinomial Naive Bayes ROS Doğruluk tablosu	44
Tablo 8.8 Multinomial Naive Bayes-TFIDF ROS Doğruluk tablosu	46
Tablo 8.9 SVM ROS Doğruluk tablosu	49
Tablo 8.10 SVM Kernel POLY ROS Doğruluk tablosu	51
Tablo 8.11 SVM Kernel RBF ROS Doğruluk tablosu	53
Tablo 8.12 SVM Kernel SIGMOID ROS Doğruluk tablosu	55
Tablo 8.13 SVM TF-IDF ROS Doğruluk tablosu	57
Tablo 8.14 KNN ROS Doğruluk tablosu	59
Tablo 8.15 KNN TF-IDF ROS Doğruluk tablosu	61
Tablo 8.16 Multinomial Naive Bayes SMOTE Doğruluk tablosu	63
Tablo 8.17 Multinomial Naive Bayes-TFIDF SMOTE Doğruluk tablosu	65
Tablo 8.18 SVM CV SMOTE Doğruluk tablosu	67
Tablo 8.19 SVM TF-IDF SMOTE Doğruluk tablosu	69
Tablo 8.20 KNN SMOTE Doğruluk tablosu	71
Tablo 8.21 KNN TF-IDF SMOTE Doğruluk tablosu	73
Tablo 8.22 LSTM Normal Veri Kümesi Doğruluk tablosu	77
Tablo 8.23 LSTM ROS Doğruluk tablosu	80
Tablo 9.1 Normal CV f-1 Skorları	81

Tablo 9.2	Emoji İndeks Bilgileri	82
Tablo 9.3	Normal TF-IDF f-1 Skorları	83
Tablo 9.4	ROS CV f-1 Skorları	84
Tablo 9.5	ROS TF-IDF f-1 Skorları	86
Tablo 9.6	SMOTE CV f-1 Skorları	87
Tablo 9.7	SMOTE TF-IDF f-1 Skorları	88
Tablo 9.8	Doğrulama Sonuçları	89
Tablo 9.9	4 Emoji için f-1 Skor Sonuçları	90

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 anlayabileceği şekilde makinenin çıktı üretmesini sağlayabilmek bilgisayar bilimleri alanında trendler arasındadır. Artık yapay öğrenme metodlarıyla Doğal Dil İşleme (NLP) sistemleri sayesinde makineler, kelimelerin anımlarının yanında cümle anlamını hatta paragraf bağlamını anlayabilecek hale geleceklereidir.

Yapılan araştırmalara göre iletilmek istenen mesajın sözel kapsamı %7 ile %15 aralığındadır [1]. Anlatılmak istenen mesaj en çok görsel iletişim olan jest ve mimiklerle alıcıya ulaştırılmaktadır. Bu nedenle metin mesajlarında iletimin güçlendirilmesi için emojiler kullanılmaktadır. Emojiler sayesinde metinlere duyu kazanımı sağlanmakta ve iletilen mesajın doğruluğu artmaktadır.

Bu çalışmamız sayesinde emoji kullanılmayarak, duygusu yoksun metinlerin duygularını tahmin edebilecek bir sistemin, bir kaç farklı yapay öğrenme yöntemiyle karşılaşmalı olarak tasarlanarak çıktılarının analiz edilmesi hedeflenmiştir.

Anahtar Kelimeler: Doğal Dil İşleme (NLP), Makine Öğrenmesi (ML), Naive Bayes, Destek Vektör Makineleri (SVM), K-En Yakın Komşu (KNN), Derin Öğrenme (DL), Kısa-Uzun Süreli Bellek (LSTM), Emoji Tahmini, Yapay Sinir Ağları, Keras, TensorFlow, Duygu Analizi

ABSTRACT

Emojis for Sentiment Analysis 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 7-15% of the message to be transferred is conveyed through texts [1]. 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:

Natural Language Processing (NLP), Machine Learnig(ML), Naive Bayes, Support Vector Machine (SVM), K-Nearest Neighbours (KNN), Deep Learning (DL),Long Short-Term Memory (LSTM), 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 [2]. 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-Asistan 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 dile 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 özetleme, duygusal analizi ve dil bilgisi/yazım denetimi gibi araçlar örnek olarak verilebilir. Mesela,某些 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:

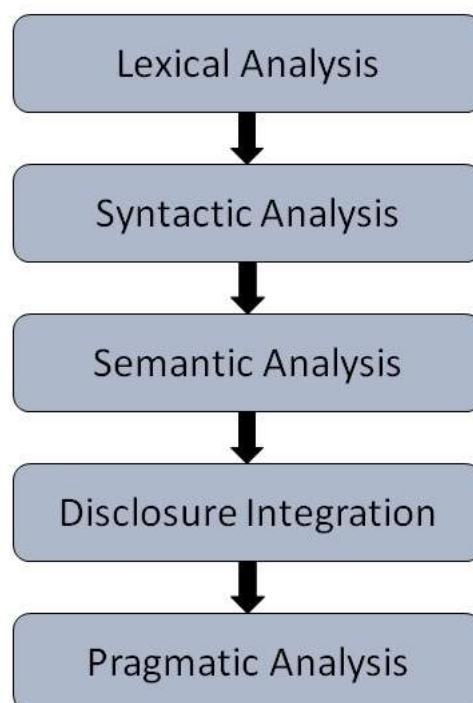
- Doğal dil oluşturma (NLG) üretme anlamına gelir.
- Doğal dil anlayışı (NLU) bilgisayarları kullanarak insan dilini anlamaya demektir.

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

"dil bilgisi" olarak bilinen bir kavram vasıtasyyla anlamlı bilgileri özütleyerek metne dönüştüren "dil çıktısı" da denir [3].

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 üremekle ilgilenen bir mühendislik koludur [3].

Doğal dil işleme aşamaları; biçimsel analiz (lexical) , sözcüksel analiz (syntax) , anlam analizi (semantics) , söylem analizi (discourse) , pragmatik analiz (pragmatics) olarak sınıflandırılmıştır [3]. Bunlar haricinde ses hece analizi olarak fonoloji de doğal dil işlemesi olarak sayılabilir. Biz bu çalışmamızda anlam analizi olarak metinden duyu çıkarımı yapmaktadır.



Şekil 1.1 NLP 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 örüntüleri öğrenebilen makine öğrenmesinin (ML) alt dallarından biri olan derin öğrenme (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 kullanıldığında bugünden önce 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 bağamlarda farklı anlamlarda olur), yakın anlamlı cümle ve kelimeleri ilişkilendirerek (örneğin, "gül" aramasında içinde "çiçek" geçen dokümanları göstererek) ve biçimbilimsel varyasyonları dikkate alarak sıkça sorulan sorular 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 edilirliğe 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 kavuşmanızı sağlayabilir. Duygu analizi sayesinde, sosyal medyada yer alan içerikler başta olumlu/olumsuz olarak sınıflandırılmaya tabi tutulabilir. Ayrıca diğer duyguları tespit ederek kullanıcıların isteklerinin doğrultusunda anlık doğru ölçüm gerçekleştirebilir.

Pazar tahminleri:

Hedef pazarınızın dilini analiz etmek üzere doğal dil işleme yöntemlerinden faydalananarak 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 duyguyu (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, dil tanımlama, sözcük anlam belirsizliğini giderme ve intihal tespiti doğal dil işlemenin uygulama alanları arasındadır [3].

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 platformları 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 sosyal mecralarda kullanıcı sayısının artırsıyla ve web 2.0 teknolojilerinin güncel hayatı bir hayli dahil olmasıyla 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 olguyu internet ortamına taşımakta ve yorumlamaktadır. Bu yorumlara göre, şirketler satış ve pazarlama stratejileri geliştirmekte ve kendilerini sürekli dinamik tutmaktadır. Bu değişime ayak uyduramayan şirketler tediçen pazar paylarını kaybetmektedir. Kullanıcıların içerik üretebileceği bir platform olan Twitter'da bile dakikada yaklaşık 200.000 [4] tweet atılmaktadır. Bunu saat veya gün bazlı düşündüğümüzde bu kadar çok verinin insanlar tarafından işlenmesi, anlamlandırılması ve yorumlanması imkansız hale gelmektedir. Diğer platformlar için istatistiksel bilgiler tablo 1.1'de gösterilmiştir [4]. Bu kadar yüksek hacimli verilerin sadece metin içeren kısımları dahi çok yekün tutmaktadır. Doğal dil işleme yöntemleriyle bu verilerin anlamlandırılması kolaylaşmıştır. Bunun dışında bilişim sistemlerinin insan dilinde konuşulan veya yazılan içerikleri en yakın anlamıyla yüzde yüz 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 faydalanailmaktadır. İletişimde verilmek istenen mesajın sadece %7-15'i kelimelerden oluşmaktadır. Bunun haricinde ses tonları ve konseptin %30, mimik veya yüz ifadelerinin ise %55 oranlarında olduğu hesaplanmıştır [1]. Kısacası istenen mesajın çoğunu duygusal görsel ifadeler oluşturmaktadır. Bundan dolayı internet ortamında üretilen metinsel içeriklerde anlam tam anlamıyla alıcıya iletilememektedir. Metin halinde üretilen içeriklerin duygudan yoksun olması nedeniyle emojiler geliştirilmiştir. Bu sayede içerik üreticileri iletmek istediği mesajın anlamını daha anlaşılır 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 netlige 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 Kaydırması/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 hedeflenmiştir. Duygu analizinde amaç yazarın hedefe karşı takındığı 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 sınıflandırılması hedeflenmiştir. Bunun için bir duygusal aktarım yöntemi olan emojilerden yararlanılmıştır. Buradaki amaç metne ait duygunun tespiti için emojinin kullanılmasıdır. Öncelikle emoji ile duygulandırılmış metinler açık kaynak ortamında temin edilmiştir [7]. Belirtilen Semeval 2018 Task-2 verileri bir çok bilimsel makaleye konu olmuş, twitter platformu üzerinden toplanmış gerçek ve ham verilerdir. Bu metinlerde sadece bir emoji ikonu olmasına kısacası yalnızca bir duygusal çıkarımıne 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 duygusal aktarımını sağlayan emojiler ise etiket (Label) olarak veri setindeki metinlerin yanına ekstra sütun açılarak işlenmiştir. Kısacası her cümleinin bir yanındaki sütunda o cümleye ait emoji bir etiket olarak kullanılmıştır.

Belirtilen veri setiyle model eğitilmeye çalışılmıştır. Veri setimiz öncelikle test ve eğitim datası olarak iki parçaya ayrılmıştır. Bunun için genel kabul görmüş oranlar baz alınmıştır (80 eğitim - 20 test). Çapraz Doğrulama (Cross Validation) yöntemleriyle verinin tutarlılığı hesaplanmıştır. Ayrıca eğitim verisi üzerinden doğrulama verisi (validation data) bölüntülenmiştir.

Öncelikle makine öğrenmesi algoritmalarında Naive Bayes, K-En yakın Komşu (KNN) ve SVM algoritmaları yardımıyla eğitim verisi üzerinde sınıflandırma yapılmaya çalışılmıştır. Bu şekilde eğitilmiş modelimizi test verimiz üzerinde çalıştırarak doğruluk oranlarımız hesaplanmıştır. Kısacası 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 modelimizin iyi eğitilip eğitilmediği doğrulama veri setiyle her eğitim döngüsünde kontrol edilmiştir.

Öncelikle 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.

1.4 Benzer Çalışmalar

Son yıllarda bir çok bilimsel araştırmaya konu olan sosyal ağ analizleri NLP araştırmacıları içinde ilgi çekici konular arasında yer almaktadır. Bu kapsamda bir çok araştırma yapılmıştır. Duygu analizi veya emoji tahminlemesi konularında Semeval verileri de bu araştırmaların birçoğunu veri seti olarak kullanılmıştır [8].

Veri setini toplayan ekibin yazmış olduğu makaleye göre Destek Vektör Makineleri (SVM)'nin diğer Evrişimsel Sinir Ağı (CNN) ve Uzun Kısa Süreli Bellek (LSTM) yöntemlerine göre daha doğru çıkarımlarda bulunduğu vurgulamıştır [9].

Aynı veri seti üzerinde çalışan Çağrı Çöltekin ve Taraka Rama tarafından yapılan çalışmanın başlığı ise "SVMs perform better than RNNs at Emoji Prediction" (SVM Yinelenen Sinir Ağları (RNN) yöntemlerinden daha mı başarılı?) olmuş ve SVM'nin RNN yöntemlerinden daha doğru hesaplama yaptığı sonucuna varmıştır [10]. Aynı veri seti üzerinde bir diğer çalışmada ise Naive Bayes'in RNN algoritmalarından daha başarılı sonuç ürettiğini tespit etmiştir [11].

Bunlar haricinde benzer çalışmalarında da Çok Terimli Naive Bayes (MNB) sonuçlarının kayda değer biçimde karmaşık derin öğrenme sistemlerine kıyasla daha başarılı sonuçlar ürettiğini vurgulayan [12] ve RNN yöntemlerinin daha yüksek başarıya ulaşması için bir dizi ön çalışma yapan ve karmaşık katmanlı derin öğrenme mimarileri modelleyen [13] çalışmalar da mevcuttur.

Proje başarımız aynı veri setini kullanan bir çok projenin başarılarıyla kıyaslanmıştır ve benzer sonuçlar alındığı görülmüştür. Bir DL yöntemi olan LSTM'in başarısının, bir ML yöntemi olan SVM'nin başarısından daha düşük olması şaşırtıcı olsa da, tüm projelere yakın sonuçlara ulaşması bizlerinde bu çalışmada hangi yöntemler kullanılarak daha başarılı sonuçlara ulaşabileceğin konusunda motive etmiştir.

1.5 Veri Seti

Veri setimiz bir çok bilimsel araştırmada, özellikle NLP çalışmalarında veri seti olarak kullanılan Semeval 2018 Task-2 [7] setidir. Veri setinin ham hali Şekil 1.2'de verilmiştir.

```
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
s e a s o n s g r e e t i n g s @ Farmers Branch Historical Park,17
myfavvvv @ Great Lakes Mall,13
"we're baaaaaaaaack @ 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
Ooooweeeeeeeeee!!!!!! #Kobe #MambaOut #KStateMBB #NikeBasketball @ Bramlage Coliseum,4
BOOM 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
```

Şekil 1.2 Temizlenmemiş Veri

Göründüğü üzere veriler bir çalışma için üretilmemiş sentetik olmayan doğal verilerden oluşmaktadır. Atılan tweetlerin hedef kitleleri için belki bir anlam ifade eden ancak konsepte uzak kişiler için anlaması veya analiz edilmesi zor olan verilerle çalışmak ve bunları modelimize doğru aktarabilmek bir hayli zor olmuştur. Veri setinin temizlenmiş hali Şekil 1.3'de verilmiştir.

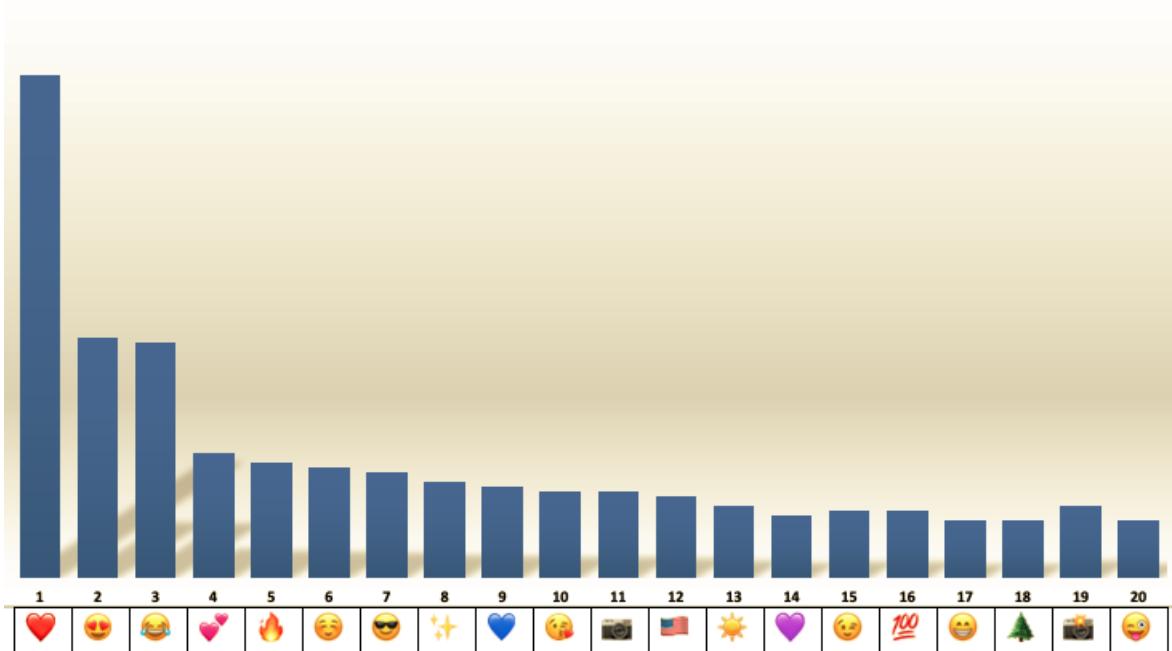
TEMİZLENMİŞ	TEMİZLENMEMİŞ
stop drinkin saint amp second,7	Can't stop drinkin' about you @ Saint & Second,7
ought ummmmm favorite season,0	"@OneRepublic: Ought. Ummmmm favorite season,0
happy b day mickey,0	#mickeymouse #88thbirthday #vivaorlando #waltdisneyworld Happy B-Day Mickey!!!!!! You!""I...",0
farmers branch historical park,17	s e a s o n s g r e e t i n g s @ Farmers Branch Historical Park,17
myfavvvv great lakes mall,13	myfavvvv @ Great Lakes Mall,13
baaaaaaaaaack fayetteville arkansas,6	we're baaaaaaaaack @ 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
oooooweeeeeeeeee bramlage coliseum,4	Ooooweeeeeeeeee!!!!!! #Kobe #MambaOut #KStateMBB #NikeBasketball @ Bramlage Coliseum,4
boom boom boom boomthe coolest party dmv tuesday,4	BOOM 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

Şekil 1.3 Temizlenmiş Veri

Veri setine bakıldığında, aslında veriler gerçek dünya verileri olduğu için elde edilen sonuçlar pek de sağlıklı değildir. Örneğin veri kümesinde "baaaack" şeklinde bir kelime vardır ama "back" ile "baaaack" kelimelerinin vektör değerleri (vectorization) veya sayısallaştırılmış sonuçlarının (tokenization) farklı olduğu bilinmektedir. Ama kelimeler aslında bağamları içinde benzer anlamda kullanılmış. Benzer şey "looooooooooveeee" ve "love" için de söylenebilir. Başka örneklerdeyse "ough" ya da "ummmmm" gibi anlamsız kelimeler mevcuttur. Bu kelimelerin pek bir anlamı olmadığı için yine sonuçlara etkisi negatif olacaktır. Yine Şekil 1.3'de görüldüğü gibi bazı cümlelerden geriye çok az kelime kalmıştır. Bu yine anlam çıkarma işini zorlaştıran bir durumdur. Belirtilen sebeplerden dolayı, veri seti doğal ve gerçek verilerindenoluştuğu için, bu verileri ML ve DL yöntemleriyle yorumlamak oldukça meşakkatli olacaktır.

Veri setimiz 20 sınıf üzerinden oluşmaktadır. Bu sınıflar emojilerimize karşılık gelmektedir. Veri seti incelediğinde 20 sınıf olmasına karşın sınıf dağılımlarının çok dengesiz olduğu Şekil 1.4'de görülebilmektedir. Bu tip dengesiz veriler üzerinde eğitimin doğru modellenebilmesi bir hayli zordur. Bu durumu aşmak için bir kaç aşırı örnekleme yöntemi denenmiştir.

EMOJİLER ORANLARI



Şekil 1.4 Emojî Dağılımı

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 içeriğini 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 eğitilen model üzerinden çıkarımlarda bulunma ihtiyacı doğmuştur.

2.2 Proje Kapsamı

Projemizde yapılması amaçlanan sistemden beklenenler şunlardır:

- Farklı vektörizasyon ve tokenizasyon yöntemleriyle sayısallaştırma yapılmaktadır. Bu nedenle sistemin önce modelleme işlemi için kelimeleri sayısallaştırması gerekmektedir. Kelimelerin sayısallaştırılması.
- Veri setinin dengesiz dağılımına yönelik birden çok örnekleme yöntemiyle eğitim setinin dengelenmeye çalışılması.
- 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 tasarılanması 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 kullanılabilirnesine imkan sağlayan geliştirme ortamlarına (Anaconda Navigator, Jupyter Notebook, Colab) ihtiyaç duyulmuştur.

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 belirlenmiştir. Bu kapsamında yazılım araçlarının açık kaynak kodlu olması, ürünün uygulanmasını kolaylaştırmıştır. Modelin eğitim sürecinde sistemden beklenen gereksinimleri karşıladığı görülmüştür.

Proje Windows 10 üzerinde geliştirilecektir. Python (3.9 ve üzeri) ve kütüphaneleriyle yazılacaktır. Sistem Jupyter Notebook veya Google Colab üzerinde derlenecektir bu nedenle ücretsiz olarak temin edilebilen NumPy ve Pandas gibi Python kütüphaneleri ile TensorFlow, Keras gibi AI işlemlerinde kullanılan kütüphaneler sistemde kullanılmıştır.

3.1.2 Donanım Fizibilitesi

Model eğitimi çok fazla işlem gücü gerektiren bir durum olduğundan dolayı, işlem hacmi fevkalade GPU'ların kullanılması elzemdir. Tabi 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 Grafik İşlem Ünitesi (GPU) ve Tensör İşlem Üniteleri (TPU) kullanılması büyük avantaj sağlayacaktır. Projedeyse içinde iyi işlem hacmine sahip bir Merkez İşlem Ünitesi (CPU) ve GPU bulunduran bir bilgisayarın yanı sıra, TPU ve GPU desteği sağlayan google colab gibi çevrimiçi hizmetler de kullanılmıştır. Bu projede birden fazla insan emek sarf edecekinden dolayı, yapılan işlemleri ve yazılan kodları hızlı ve efektif olarak tutabilmek için

bulut sistemi kullanılmıştır. Bu sistemde kodlardaki ilerlemeler de eşzamanlı olarak tutulabilmektedir.

3.2 İş Gücü ve Zaman Fizibilitesi

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

Tablo 3.1 Proje İş / Zaman Çizelgesi

GÖREVLER	EKİM				KASIM				ARALIK			
	H1	H2	H3	H4	H1	H2	H3	H4	H1	H2	H3	H4
ML ve DL araştırılması 03.10.2022												
Benzer projelerin incelenmesi 10.10.2022												
Veri seti analizi 17.10.2022												
Kodlama 08.11.2022												
Kodlama dili ve kütüphane belirlenmesi 01.11.2022												
Veri setinin hazırlanması 24.11.2022												
Ara Rapor 21.11.2022												
Optimizasyon Çalışmaları 05.12.2022												
Rapor 18.12.2022												

3.3 Ekonomik Fizibilite

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

Derin öğrenme ve makine öğrenmesi için kullandığımız kütüphaneler de ücretsiz olması sebebiyle, kullanılan yazılım araçlarına da bir ücret ödenmemiştir.

Projede kullandığımız veri seti de açık kaynaklı ve ücretsizdir.

Kişisel bilgisayarlarımızın donanımı da (Acer Nitro-5 ve Macbook Pro 2011) projeyi gerçekleştirmek açısından gayet yeteri seviyededir.

Detaylı maliyet tablosu Tablo 3.2'de gösterilmiştir.

Tablo 3.2 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 Semeval 2018 Task-2, CodaLab [7] üzerinden alınmıştır. Açık kaynaklı olduğu için, kullanımı herhangi bir yasal sorun teşkil etmemektedir.

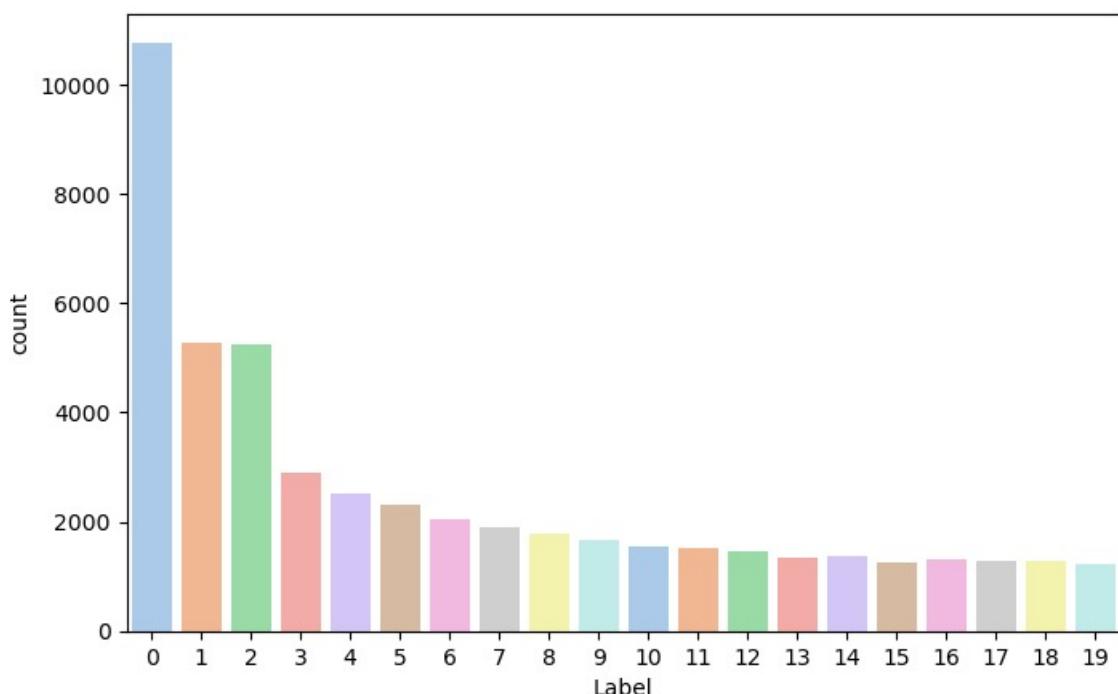
Proje herhangi bir şekil ve durumda yasa ve yönetmelikleri ihlal edecek bir veri tutmamaktadır. Aynı zamanda tüm kullanılan kütüphane ve frameworkler açık kaynaklı ve ücretsizdir. Bundan dolayı herhangi bir patent gibi koruyucu hak da kullanılmamıştır.

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 duyu analizi çıkarımlarıyla ve çeşitli eğitim metodlarıyla emojilerle 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 sınıflandırılmamış 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 yöntem ile ulaşıldığı karşılaştırılmak istenmiş ayrıca uygulanan ML ve DL algoritmalarının farklı metriklerle nasıl sonuçlar ürettiği tespit edilmeye çalışılmıştır.

Ayrıca veri seti üzerinde bir kaç ön işleme metodu tüm algoritmalar üzerinde uygulanarak farklılıklarını tespit edilmeye çalışılmıştır.

4.3 Performans Metrikleri

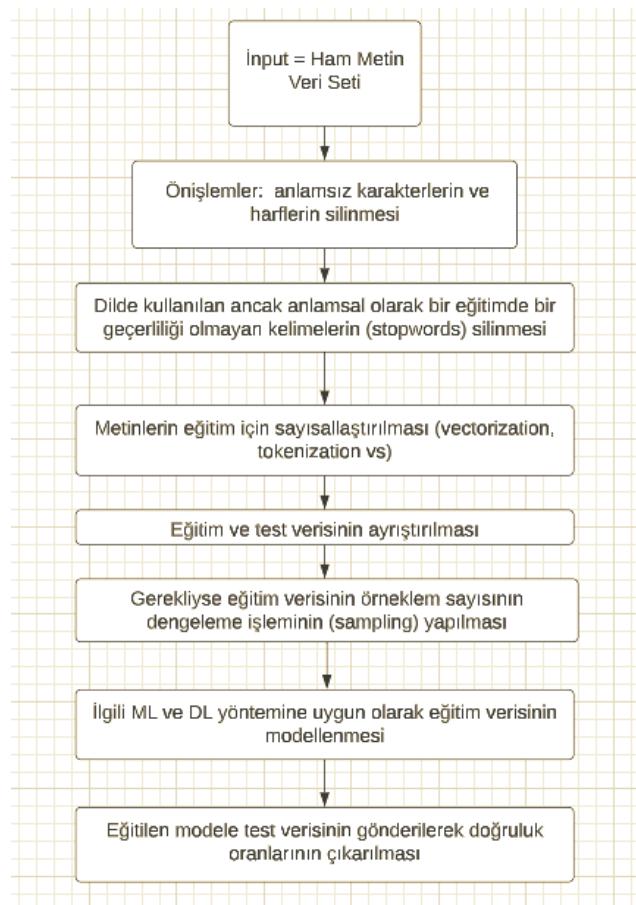
Eğitilen modellerden hedeflenen ise metinlerin emoji karşılıklarını en doğruya yakın şekilde gösterebilmesidir. Eğitilen modelimiz test veri seti üzerinde çalıştırılarak, test veri setindeki cümlelere uygun emojileri tahmin etmesi beklenmiştir. Yapılan tahmin sonucu çıkan emoji etiketler ile gerçekte 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. Sonuçlar literatürde kabul gören (avg accuracy, F-1 Score vb.) metrikler üzerinden çıkarılacaktır.

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, MNB kullanılmıştır.



Şekil 5.1 Metin Sınıflandırma Akış Şeması

5.1.1 LSTM

LSTM derin öğrenme alanında kullanılan yapay bir yinelemeli sinir ağısı (RNN) mimarisidir. LSTM geri beslemeli çalışır. Anlık verinin yanında veri dizilerini de işleyebilmektedir. Sıradan bir LSTM ünitesi giriş, çıkış ve unutma kapısından oluşmaktadır [14].

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 indirdiğimiz için inputumuzdaki başlıca bir farklılık aktivasyon fonksiyonunda gereği kadar önemli bir farklılığa sebep olmayıabilir. Bu sebeple türevi de minör olur ve türevi çok minörse, o seviye gereği miktarda öğrenemez.

Kıcacısı insan hafızası gibi önemli ve gerekli olanları tespit ederek onları kalıcı bir şekilde hafızasında tutmak, ancak önemli görmediklerini unutmak üzerine tasarlanan bir mimaridir.

LSTM bu problemi dört adımda çözüyor [15], 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. Unutma kapısı da benzer mantıkla bir durum gerçekleşiyor. Unutmak istediğimiz tüm girdilerin ağırlığına sıfır değeri vermiş oluyoruz.

Önceki gizli katmandan elde edilen bilgiler ve anlık bilgiler Sigmoid Fonksiyonu isimli bir fonksiyona girer. Sıfıra ne kadar yakınsarsa o kadar unutulmasının gerekli olduğu, bire ne kadar yakınsa bilginin o kadar unutulmayacağı anlamına gelir.

- Girdi Kapısı (Input Gate) :

Hücre durumunu 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) :

Hücrenin kapsadığı en önemli görevi bilgiyi hareket ettirmektir. Hareket ettirişmesi icap eden verileri alır ve hücre sonuna, hücre sonundan da farklı hücrelere hareket ettirir. Yani ağ üstündeki veri trafigini hücre durumu vesilesiyle sağlarız. Öncelikle unutma kapısından gelen sonuç ile ondan önceki katmanın sonucu çarpılır. Ardından giriş kapısından elde edilen değer ile toplanır.

- Çıktı Kapısı(Output Gate) :

Bir sonraki seviyeye gidecek değer seçilir. Bu değer, tahminleme amacıyla kullanılır. İlk olarak önceki değer ile anlık input Sigmoid fonksiyonunda işlenir. Hücre durumundan elde edilen değer Tanh fonksiyonunda işlendikten sonra iki değer çarpılır ve bir sonraki seviyeye “hemen önceki değer” olarak gönderilir. Hücre durumu 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 [16]. Naive Bayes algoritmasının temeli olan Bayes Teoreminin matematiksel karşılığı aşağıda görüldüğü gibidir.

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

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 kayıp/boş veriler de olasılık hesaplarına katılmayarak ele alınabilmektedir. Basit ilişkilerin olduğu durumlarda genellikle iyi sonuç çıkarır bir yöntemdir [17].

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 varsayarsak, 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 özelliklerini şunlardır:

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

5.1.4 Destek Vektör Makineleri

Destek vektör makinesi (SVM) algoritması, girdi olarak alınan bir veri seti üzerinde iki sınıf arasındaki arımı optimum düzeyde yapan hiperdüzlemi bulmayı hedefler. Girdi olarak alınan bir veri noktalarının düzlem üzerindeki yerlerinin (x, y) ile ifade edildiğini varsayarsak, hiperdüzlem verileri iki sınıfa optimum şekilde ayırabilen düzlemdir.

SVM algoritmasının özelliklerini şunlardır:

- SVM algoritması çoğunlukla yüksek performansla çalışır, SVM'den alınan sonuçlarsa 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

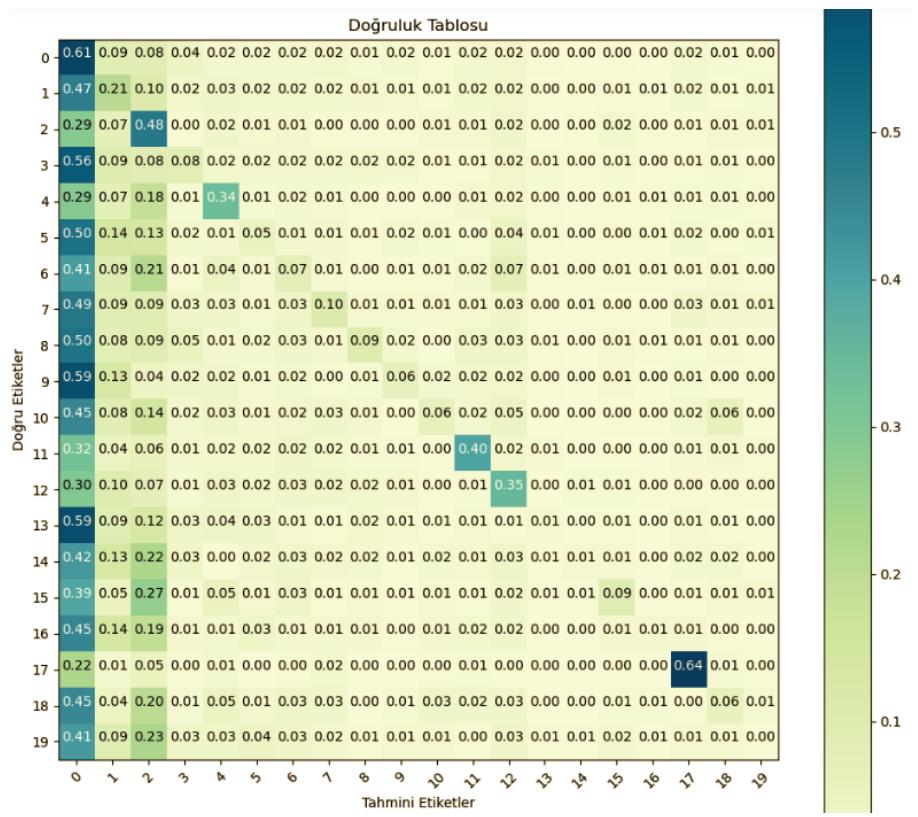
Modeller toplamda yaklaşık 50.000 emoji içeren metinle eğitilmiştir. Veri setinde mutlu surat, kalp , kalpli surat , alev de dahil olmak üzere toplam 20 emoji bulunmaktadır. Bu emojilerden, optimum sonuç alabilmek için veri temizleme işlemleriyle birlikte belli bir kısmı göz ardı edilmiştir.

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

Tablo 6.1 Naive Bayes ile elde edilen başarımlar



Tablo 6.2 SVM ile elde edilen başarımlar



6.2 Modellerin Eğitilmesi İçin Kullanılan Yöntem

Eğitim sürecinde 50.000 adet veri setinin bir kısmı test (%10-%15-%20-%30) bir kısmı da eğitim verisi olarak böülümlere ayrılmıştır. Aynı veri kümesi SVM, Naive Bayes, KNN, LSTM yöntemleriyle kullanılmıştır.

Bu yöntemler elde ettikleri doğruluk oranlarına göre kıyaslanmıştır.

Her makine öğrenmesi ve derin öğrenme yöntemi farklı sayıda sınıflara göre farklı oranda başarı elde etmektedir. Aynı zamanda yine her makine öğrenmesi ve derin öğrenme yöntemi farklı oranda eğitim-test veri setlerine göre farklı doğrulukta başarı 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 setimizin bir kısmı doğrulama veri seti olarak ayrılmıştır. Özellikle derin öğrenme yöntemlerinde doğrulama veri seti sayesinde eğitim aşamasında eğitim veri setinin doğru eğitilip eğitilmediği doğrulama veri setiyle sürekli olarak test edilmiştir. Eğitim sırasında aşırı öğrenme-ezberleme (overfit) olduğu görüldüğünden, eğitim tur sayısı (epoch) değeri durdurularak, 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 Veri Seti

Yapılan deneysel çalışmalarda açık kaynak ortamından temin edilen veri seti toplamda 70.000 tweetten oluşmaktadır. Naive Bayes ve SVM yöntemlerini deneysel olarak kıyaslayabilmek amacıyla bu veri setinin de dört ayrı sınıf üzerinden çalışma yapılmıştır. Ayrıca bu veri setini farklı oranlarda kullanarak, yöntemlerin veri seti büyülüğüne ve eğitim, test oranlarına göre başarım oranlarını kıyaslanmıştır.

7.1.1 Veri Setinin Boyutunu Değiştirme

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

Tablo 7.1 Yöntem Kıyaslama 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

7.1.2 Veri Setinde Oran Değiştirme

Tablo 7.2'de görüldüğü üzere, en yüksek başarım en büyük veri setinde sağlanmıştır. Bundan dolayı bu veri setindeki test ve eğitim verilerinin dağılım oranını değiştirilerek test edilmiştir. Test ve eğitim verisi sayıları toplam 70.000 iken oluşan sonuçlar Tablo 7.2'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

7.2 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 verisi olduğu zaman başarım ciddiye alınır biçimde değişmemektedir. Ama test ve eğitim kümesi çok küçültüldüğü zaman, gözle görülür bir azalma söz konusudur.

8

Performans Analizi

Bu bölümde Semeval 2018 verileri [7] üzerinde çeşitli DL ve ML yöntemlerinde kullanılarak elde edilen sonuçlar gösterilmiş ve yorumlanmıştır. Vektörize yöntemi olarak aksi belirtilmediği sürece Vektör Sayacı (CV) yöntemi kullanılmıştır. CV yöntemiyle alınan sonuçları kıyaslamak amacıyla aynı zamanda TF-IDF vektörize yöntemi de kullanılmıştır.

8.1 TF-IDF ile CV Arasındaki Farklar

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

TF-IDF, bir dokümanıçındaki herhangi bir kelimeyi diğer dokümanlarda arayarak, kelimenin dokümanıçındaki 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 CV yöntemine kıyasla daha doğru ölçümlenebilir.

CV ise, sadece dokümanıçındaki kelime sıklıklarına bakarak bir sayı vektörü oluşturur. Diğer dökümanlarla kıyaslama işine girişmemektedir. Bundan dolayı, CV 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, CV kelime için yalnızca sıklığı ölçer. Bundan dolayı, iki yöntem de farklı hedefler doğrultusunda tercih edilebilir.

8.2 Makine Öğrenmesi

Bu başlık altında MNB, SVM (farklı kernel parametreleriyle) ve KNN algoritmaları öncelikle normal veri setinde CV ve TF-IDF vektörizasyon yöntemleri uygulanarak başarımlar skorları karşılaştırılacaktır. Veri setimizin dengesiz olması nedeniyle sırasıyla Rastgele Aşırı Örneklemme (ROS) ve Yapay Azınlık Aşırı Örneklemme (SMOTE) teknikleriyle azınlık sınıflarımızın örneklem sayısı çoğaltılarak yine CV ve Terim Sıklığı - Ters Doküman Sıklığı (TF-IDF) vektörizasyon işlemleri yapılarak karşılaştırmalar yapılacaktır.

8.2.1 Normal Veri Kümesi Sonuçları

8.2.1.1 MNB CV

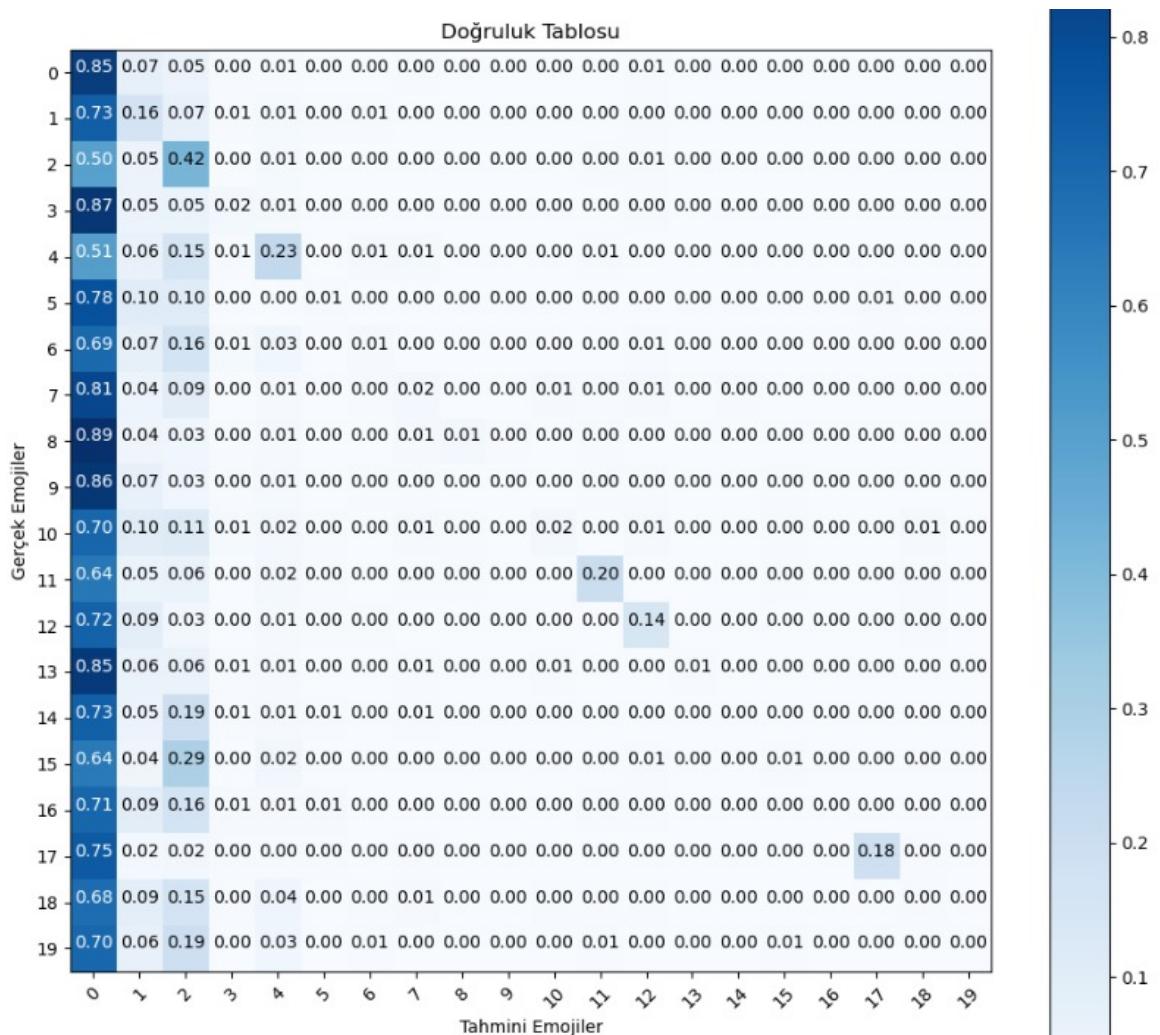
Multinomial Naive Bayes yönteminin CV ile vektörize işlemi sonrası sınıflandırma raporu Şekil 8.1'de gösterilmiştir.

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
macro avg				0.28
weighted avg				0.28
				7498
				7498
				7498

Şekil 8.1 MNB Normal Sınıflandırma Raporu

MNB yöntemi için doğrulama tablosu Tablo 8.1'de gösterilmiştir.

Tablo 8.1 MNB Normal Doğruluk tablosu



8.2.1.2 MNB TF-IDF

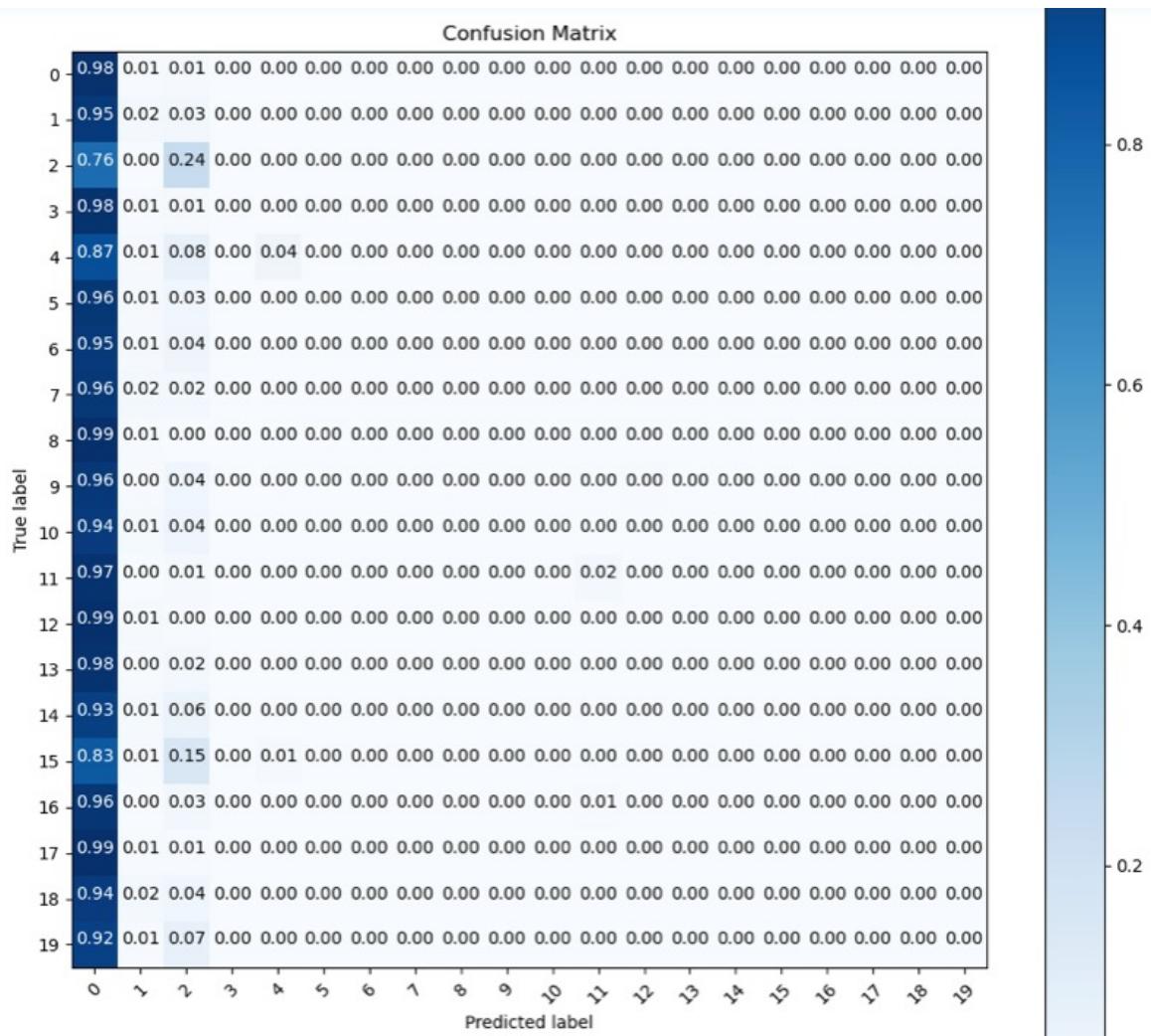
MNB yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.2'de gösterilmiştir.

	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

Şekil 8.2 MNB-TFIDF Normal Sınıflandırma Raporu

MNB yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.2'de gösterilmiştir.

Tablo 8.2 MNB-TFIDF Normal Doğruluk tablosu



8.2.1.3 SVM CV

SVM yöntemi CV vektörizasyonu ile sınıflandırma raporu Şekil 8.3'de gösterilmiştir.

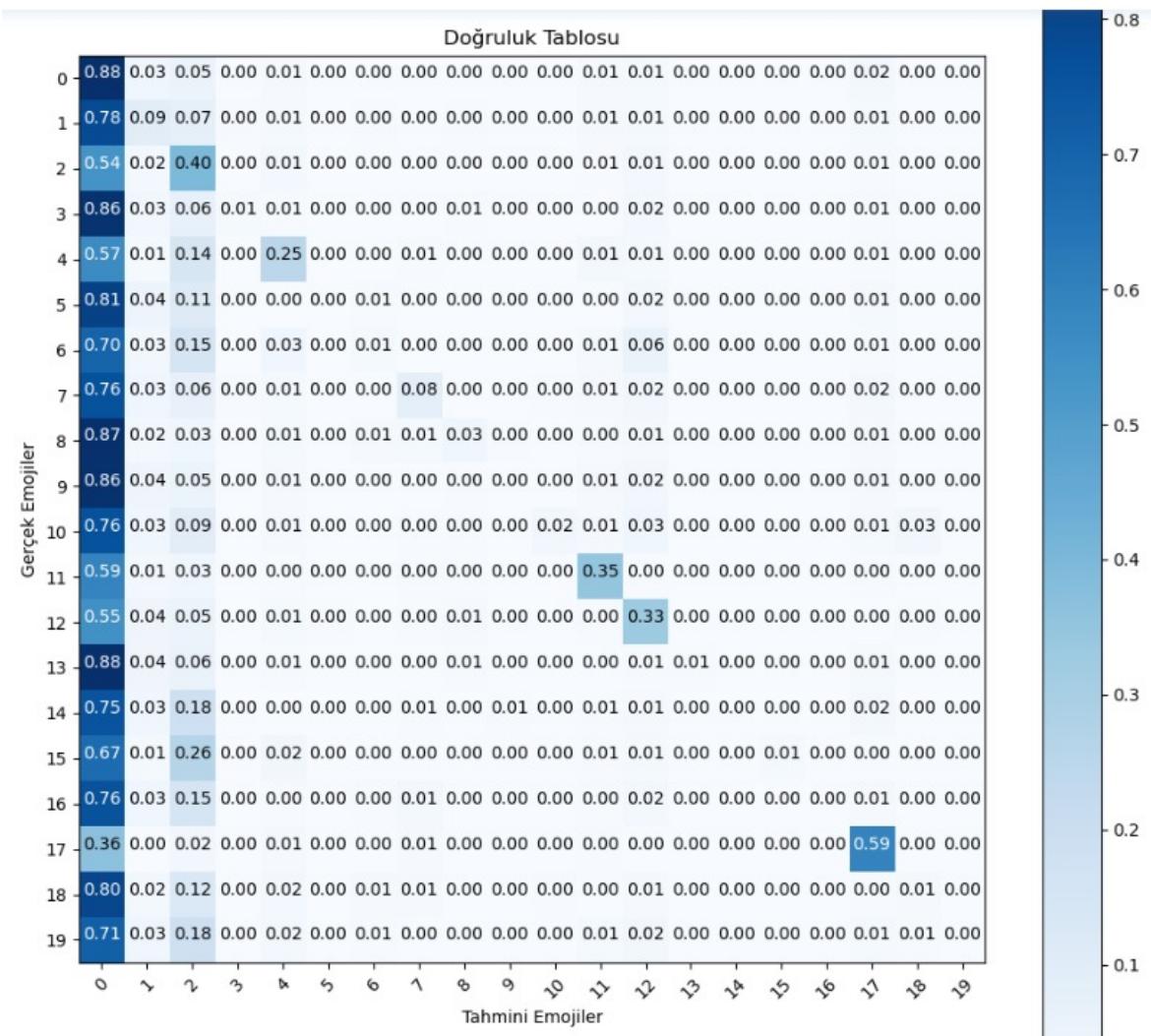
```
5]: print(classification_report(y_test, y_predict_test))
```

	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

Şekil 8.3 SVM Normal Sınıflandırma Raporu

SVM yöntemi için doğrulama tablosu Tablo 8.3'de gösterilmiştir.

Tablo 8.3 SVM Normal Doğruluk tablosu



8.2.1.4 SVM TF-IDF

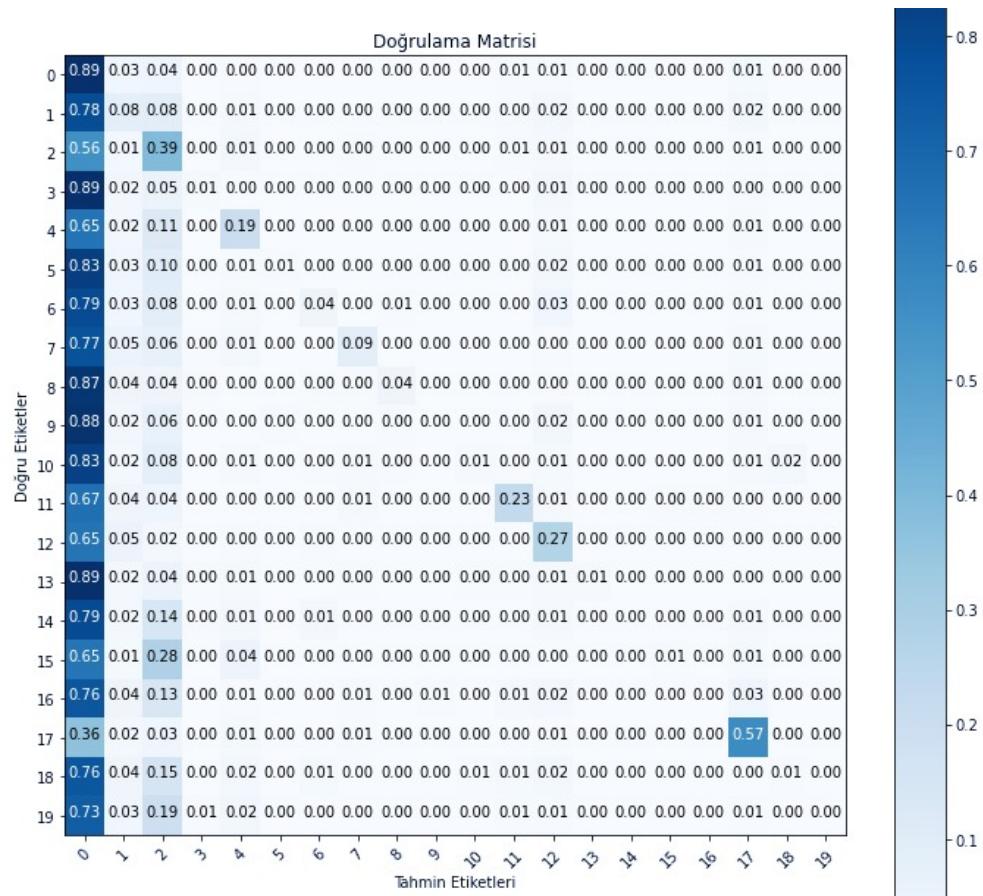
SVM yöntemiyle beraber TF-IDF vekktörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.4'de gösterilmiştir.

	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

Şekil 8.4 SVM TF-IDF Normal Sınıflandırma Raporu

SVM yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.4'de gösterilmiştir.

Tablo 8.4 SVM TF-IDF Normal Doğruluk tablosu



8.2.1.5 KNN CV

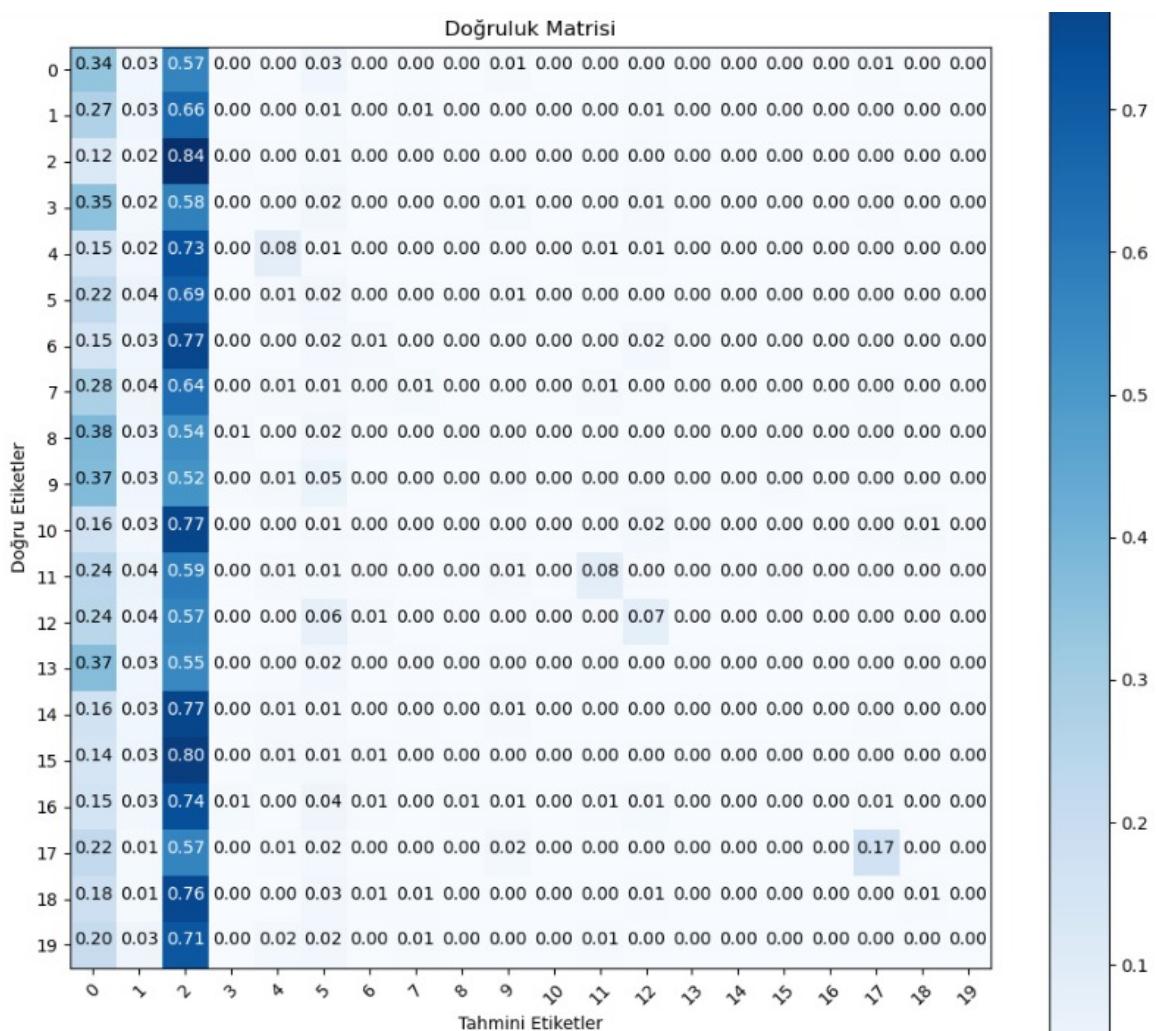
KNN yöntemi CV vektörizasyonu ile sınıflandırma raporu Şekil 8.5'de gösterilmiştir.

	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

Şekil 8.5 KNN Normal Sınıflandırma Raporu

KNN yöntemi için doğrulama tablosu Tablo 8.5'de gösterilmiştir.

Tablo 8.5 KNN Normal Doğruluk tablosu



8.2.1.6 KNN TF-IDF

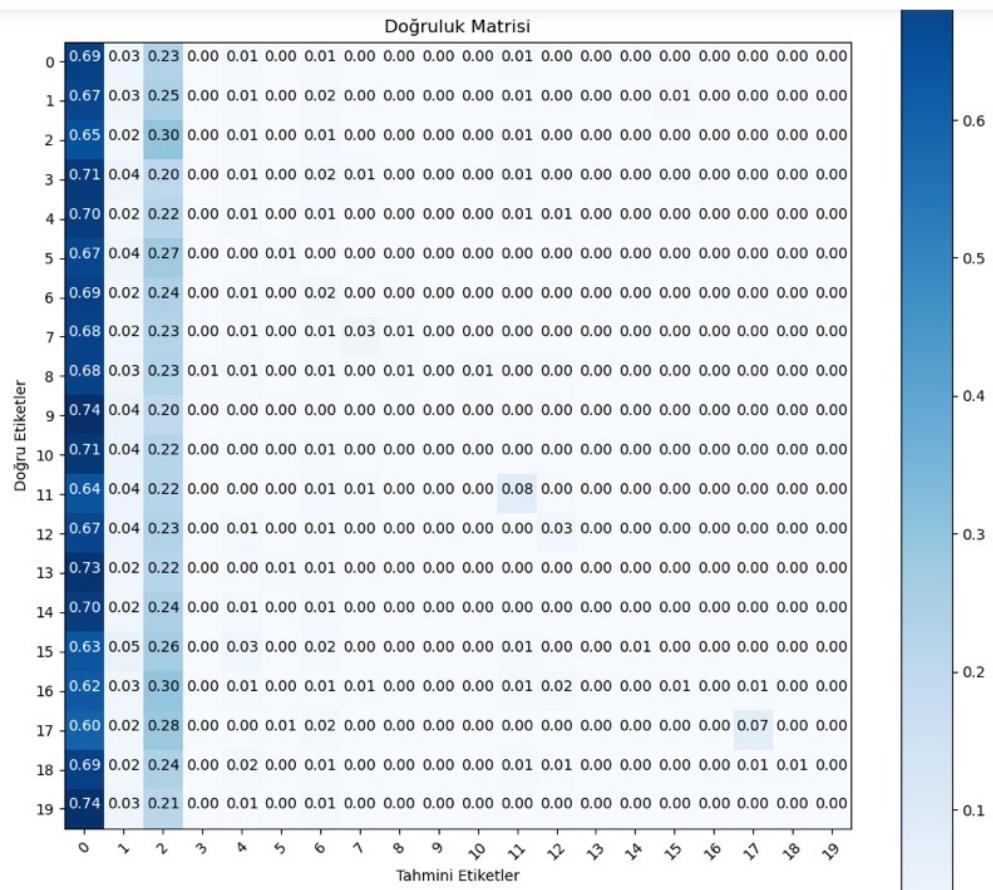
KNN yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.6'da gösterilmiştir.

	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

Şekil 8.6 KNN-TFIDF Normal Sınıflandırma Raporu

KNN yöntemiyle beraber TF-IDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.6'da gösterilmiştir.

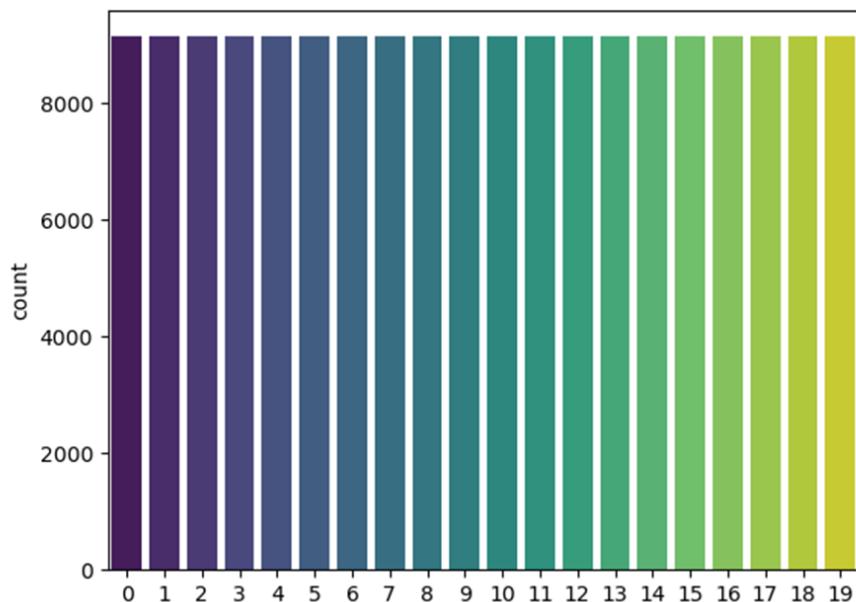
Tablo 8.6 KNN-TFIDF Normal Doğruluk tablosu



8.2.2 ROS Sonuçları

Bazı sınıfların aşırı az örneklememin olması hem eğitimde hem de tahminde sorunlara neden olduğunu göstermiştir. Bu nedenle bir aşırı örneklem yöntemi olarak ROS kullanılmıştır. Bu yöntemde eğitim setimizdeki azınlık sınıflar arasında rastgele şekilde kopya alarak üst üste eklenmiş ve eğitim setimizde yer alan tüm sınıfların sayısı aynı seviyeye getirilmeye çalışılmıştır. ROS Sonrası eğitim veri setinin durumu Şekil 8.7'de gösterilmiştir. Tüm sınıflar eşit seviyede veri sayısına sahiptir.

```
: sns.countplot(x=y_train_osm, palette="viridis")
plt.show()
```



Şekil 8.7 ROS Veri Seti Durumu

8.2.2.1 MNB CV

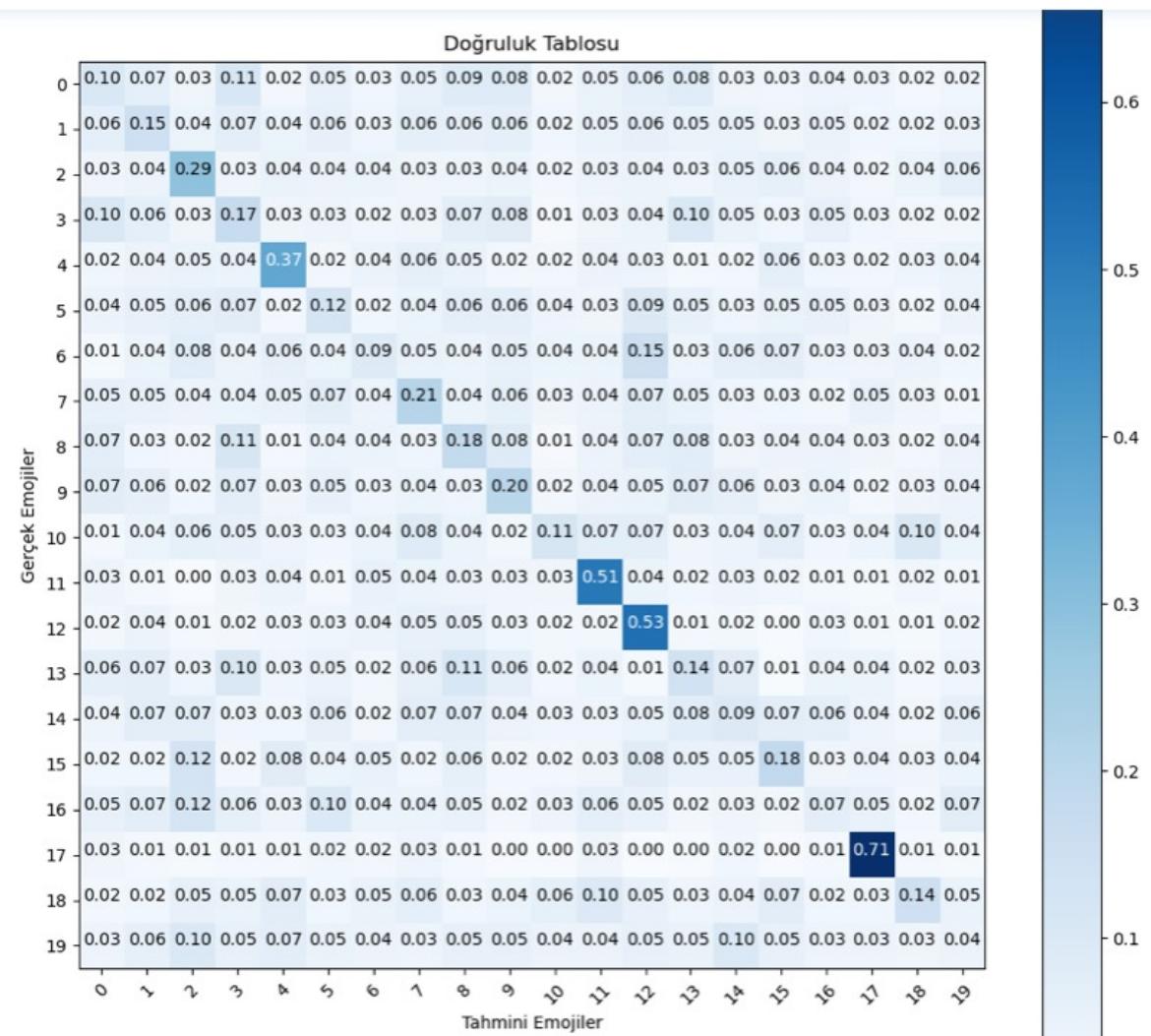
Multinomial Naive Bayes yönteminin CV ile vektörizasyonu sonrası sınıflandırma raporu Şekil 8.8'de gösterilmiştir.

	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

Şekil 8.8 Multinomial Naive Bayes ROS Sınıflandırma Raporu

MNB yöntemi için doğrulama tablosu Tablo 8.7'de gösterilmiştir.

Tablo 8.7 Multinomial Naive Bayes ROS Doğruluk tablosu



8.2.2.2 MNB TF-IDF

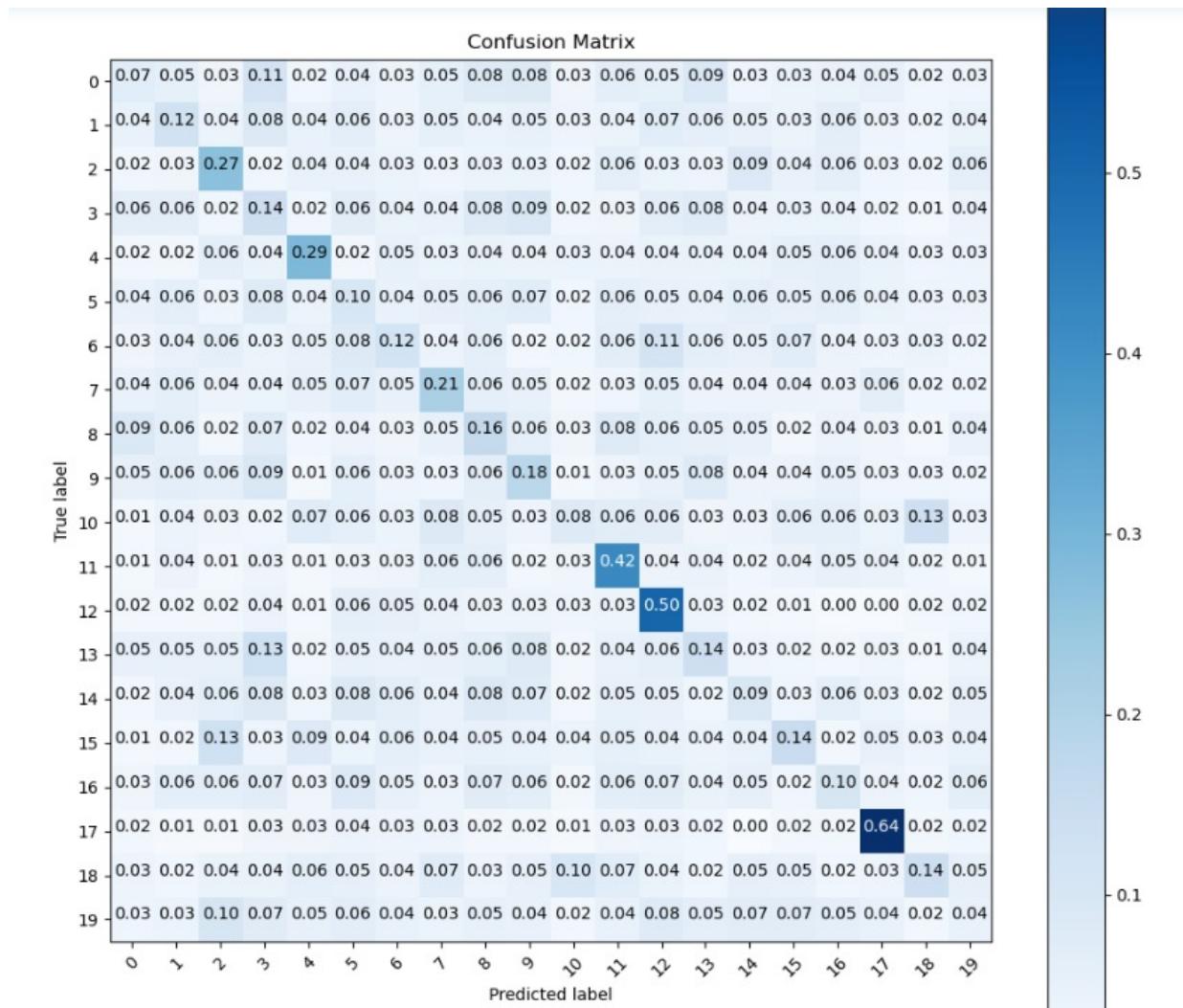
Multinomial Naive Bayes yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.9'da gösterilmiştir.

	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	7460
weighted avg		0.23	0.17	7460

Şekil 8.9 MNB TFIDF ROS Sınıflandırma Raporu

Multinomial Naive Bayes yöntemiyle beraber TF-IDF vekktörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.8'de gösterilmiştir.

Tablo 8.8 Multinomial Naive Bayes-TFIDF ROS Doğruluk tablosu



8.2.2.3 SVM CV

SVM yönteminde, kernel parametresi SVM'nin çalışma prensibini etkilemektedir. Bu parametre, veri seti özelliklerinin düz çizgi üzerinde nasıl ayırtılacağını belirtir. Kernel parametresi, veri setinin ayırtırılma işleminde kullanılacak olan "kernel fonksiyonu"nu belirtir. Kernel fonksiyonu, veri setindeki noktaların birbirleriyle nasıl ilişkilendirileceğini belirtir. SVM yönteminde kullanılabilecek kernel fonksiyonları arasında lineer kernel, polinom kernel ve RBF (Radial Basis Function) kernel gibi seçenekler vardır.

Lineer SVM, veri kümesi içindeki örnekleri düzgün bir şekilde iki sınıfa ayırabilecek bir lineer hiperdüzlem bulmaya çalışır. Bu hiperdüzlemin veri kümesi içindeki en yakın örneklerle olan mesafesi (margin) mümkün olduğunca büyük olmalıdır, bu sayede SVM'nin genelleştirme performansı artar.

Radyal tabanlı çekirdek (Radial Basis Function - RBF) SVM, veri kümesi içindeki örnekleri iki sınıfa ayıran hiperdüzlemi bulmak için veri kümesi içindeki her örnek için bir RBF fonksiyonu kullanır. Bu fonksiyonlar, veri kümesi içindeki her örnek için ayrı ayrı hesaplanır ve daha sonra bu fonksiyonlar kullanılarak veri kümesi içindeki örnekleri iki sınıfa ayıran hiperdüzlem bulunur.

Polinominal (Poly) SVM, veri kümesi içindeki örnekleri iki sınıfa ayıran hiperdüzlemi bulmak için veri kümesi içindeki örneklerin özniteliklerini polinom özelliklerine dönüştürür ve daha sonra bu polinom özellikleri kullanılarak veri kümesi içindeki örnekleri iki sınıfa ayıran hiperdüzlem bulunur.

Sigmoid SVM, veri kümesi içindeki örnekleri iki sınıfa ayıran hiperdüzlemi bulmak için veri kümesi içindeki örneklerin özniteliklerini sigmoid fonksiyonlarına dönüştürür ve daha sonra bu sigmoid fonksiyonları kullanılarak veri kümesi içindeki örnekleri iki sınıfa ayıran hiperdüzlem bulunur.

Çalışmamızda SCM yöntemi öncelikle lineer, sonrasında sırasıyla Poly, RBF ve Sigmoid kernel metotları denenecektir. Lineer hariç tüm kernel metodları sadece CV vektörizasyonu ile denenmiştir. Lineer kernel ise hem CV hem de TF-IDF ile çalıştırılarak sonuçları karşılaştırılmıştır.

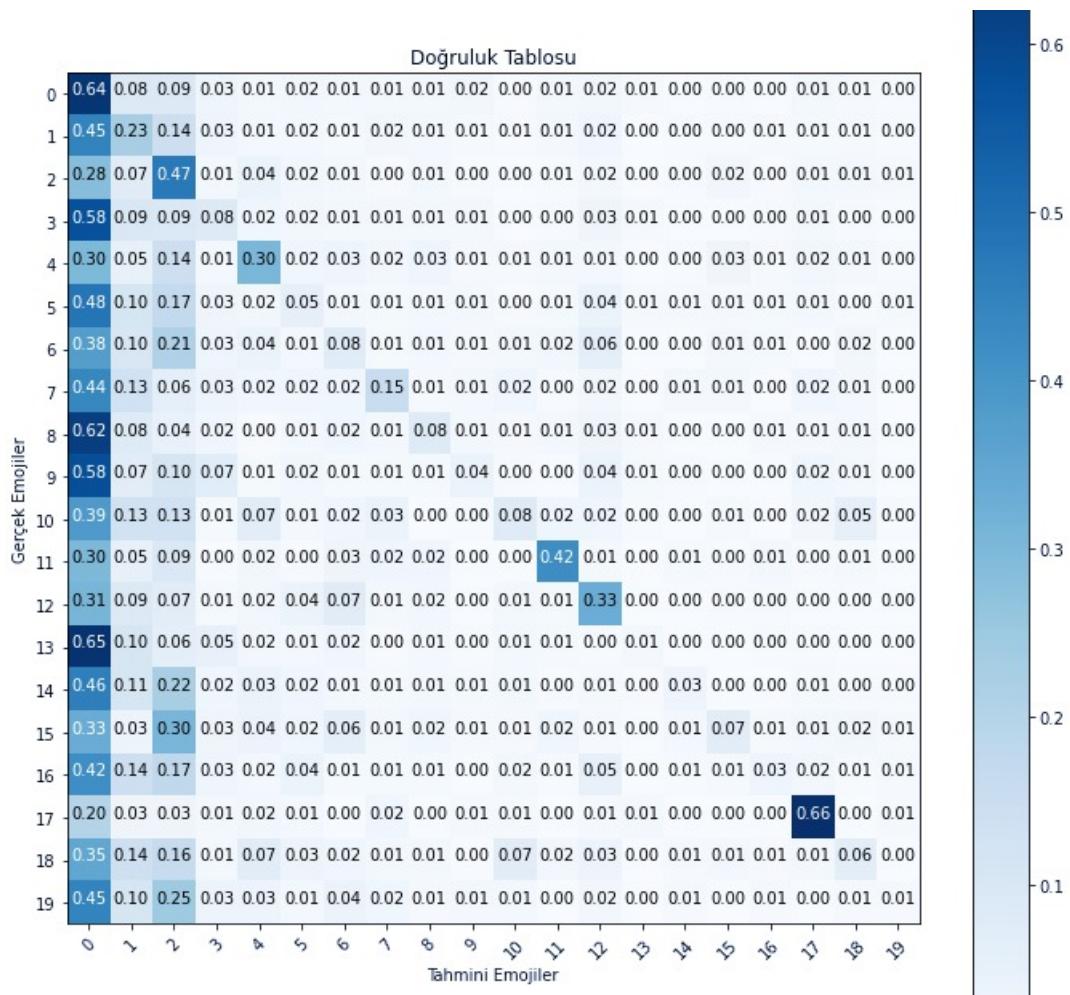
8.2.2.3.1 Kernel Lineer: SVM yöntemi kernel Lineer için CV ile vektörizasyonu sonrası sınıflandırma raporu Şekil 8.10'da gösterilmiştir.

	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

Şekil 8.10 SVM ROS Sınıflandırma Raporu

SVM lineer kernel ile CV doğrulama tablosu Tablo 8.9'da gösterilmiştir.

Tablo 8.9 SVM ROS Doğruluk tablosu



8.2.2.4 SVM CV

8.2.2.4.1 Kernel POLY: SVM'in kernel parametresi olarak POLY yöntemi için sınıflandırma raporu Şekil 8.11'de gösterilmiştir.

```
print(classification_report(y_test, y_predict_test))
```

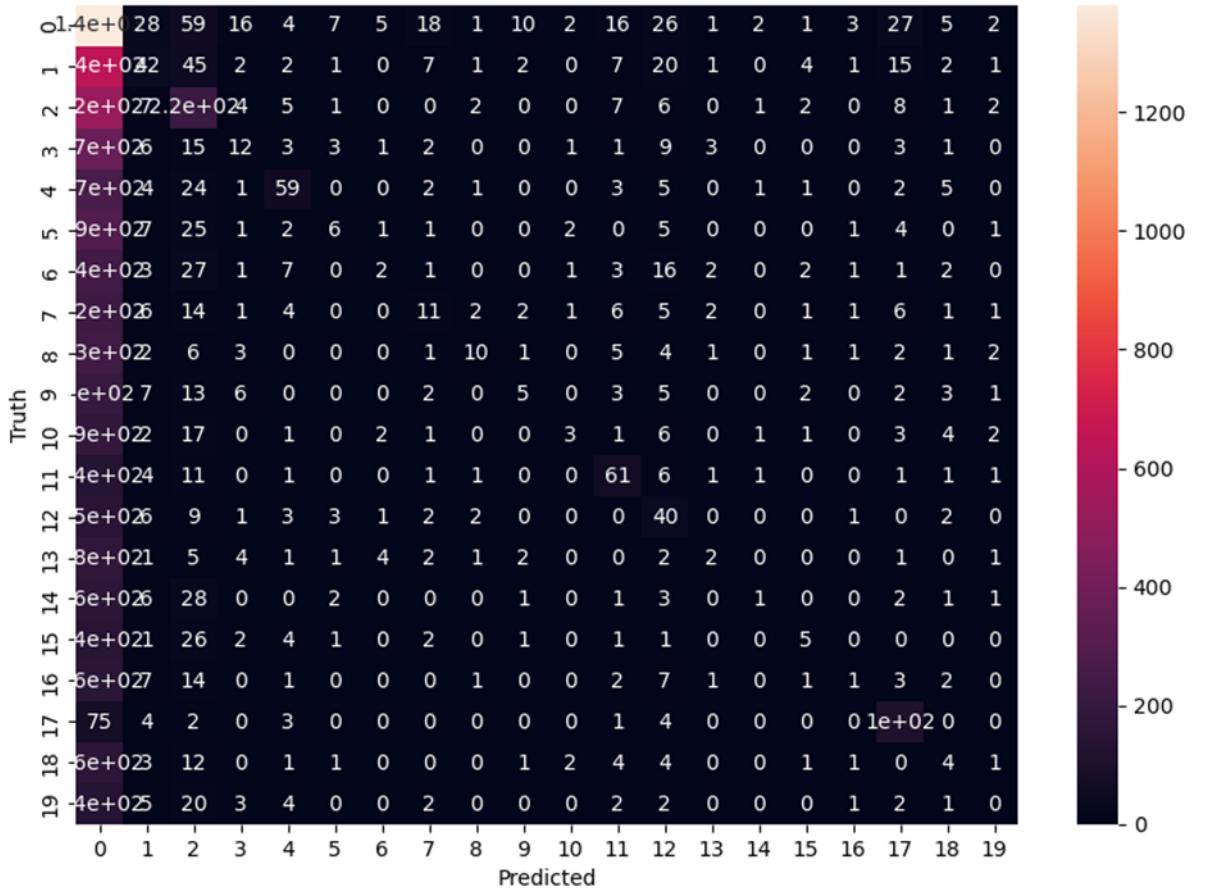
	precision	recall	f1-score	support
0	0.24	0.86	0.37	1614
1	0.28	0.05	0.09	792
2	0.37	0.27	0.31	785
3	0.21	0.03	0.05	433
4	0.56	0.16	0.24	377
5	0.23	0.02	0.03	347
6	0.12	0.01	0.01	307
7	0.20	0.04	0.06	284
8	0.45	0.04	0.07	269
9	0.20	0.02	0.04	251
10	0.25	0.01	0.02	232
11	0.49	0.27	0.35	229
12	0.23	0.18	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

Şekil 8.11 SVM Kernel POLY ROS Sınıflandırma Raporu

SVM'in kernel parametresi olarak POLY yöntemi için doğrulama tablosu Tablo 8.10'de gösterilmiştir.

Tablo 8.10 SVM Kernel POLY ROS Doğruluk tablosu

Text(95.72222222222221, 0.5, 'Truth')



8.2.2.5 SVM CV

8.2.2.5.1 Kernel RBF: SVM'in kernel parametresi olarak RBF yöntemi için sınıflandırma raporu Şekil 8.12'de gösterilmiştir.

```

print(classification_report(y_test, y_predict_test))

```

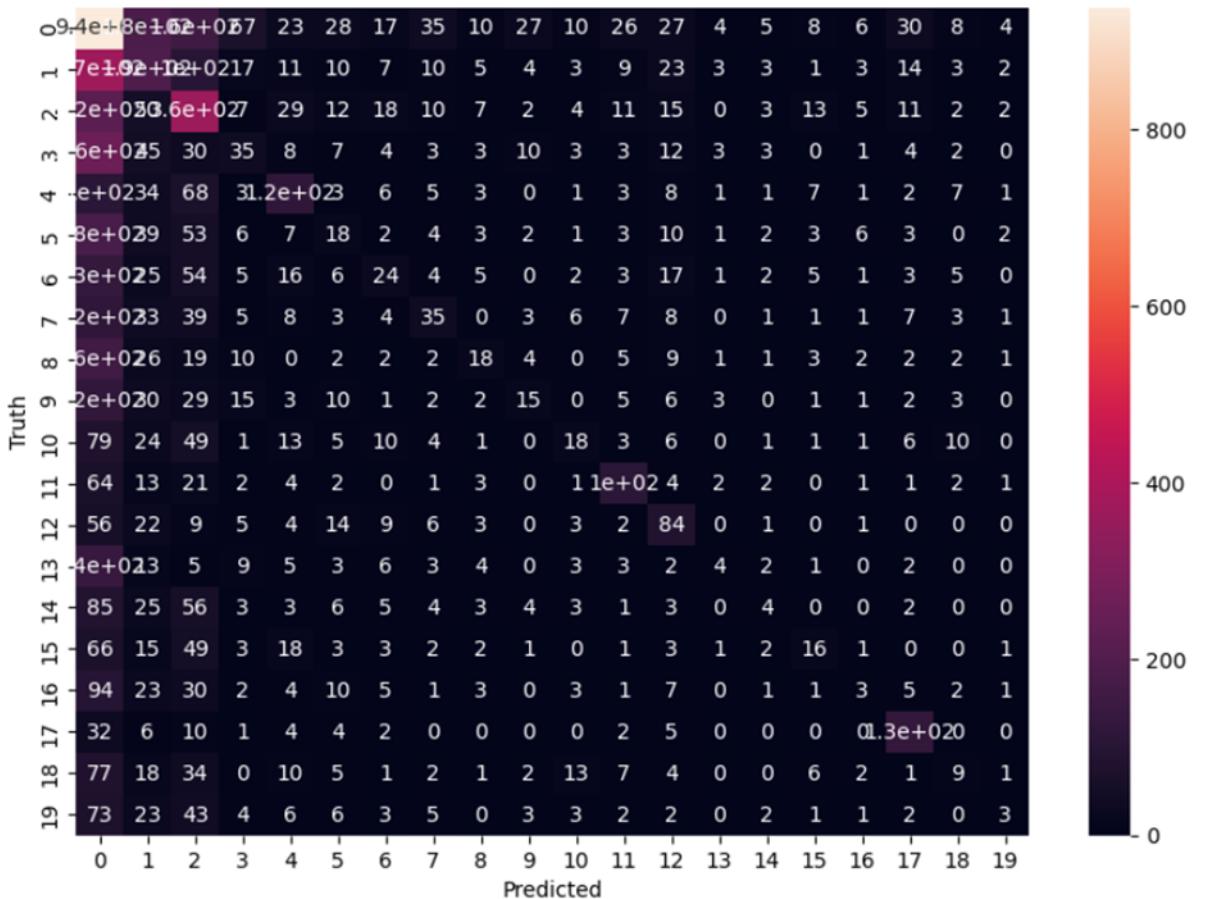
	precision	recall	f1-score	support
0	0.28	0.58	0.38	1614
1	0.23	0.24	0.24	792
2	0.30	0.46	0.36	785
3	0.17	0.08	0.11	433
4	0.40	0.32	0.35	377
5	0.11	0.05	0.07	347
6	0.19	0.08	0.11	307
7	0.25	0.12	0.17	284
8	0.24	0.07	0.10	269
9	0.19	0.06	0.09	251
10	0.23	0.08	0.12	232
11	0.52	0.46	0.49	229
12	0.33	0.38	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

Şekil 8.12 SVM Kernel RBF ROS Sınıflandırma Raporu

SVM'in kernel parametresi olarak RBF yöntemi için doğrulama tablosu Tablo 8.11'de gösterilmiştir.

Tablo 8.11 SVM Kernel RBF ROS Doğruluk tablosu

Text(95.7222222222221, 0.5, 'Truth')



8.2.2.6 SVM Kernel SIGMOID

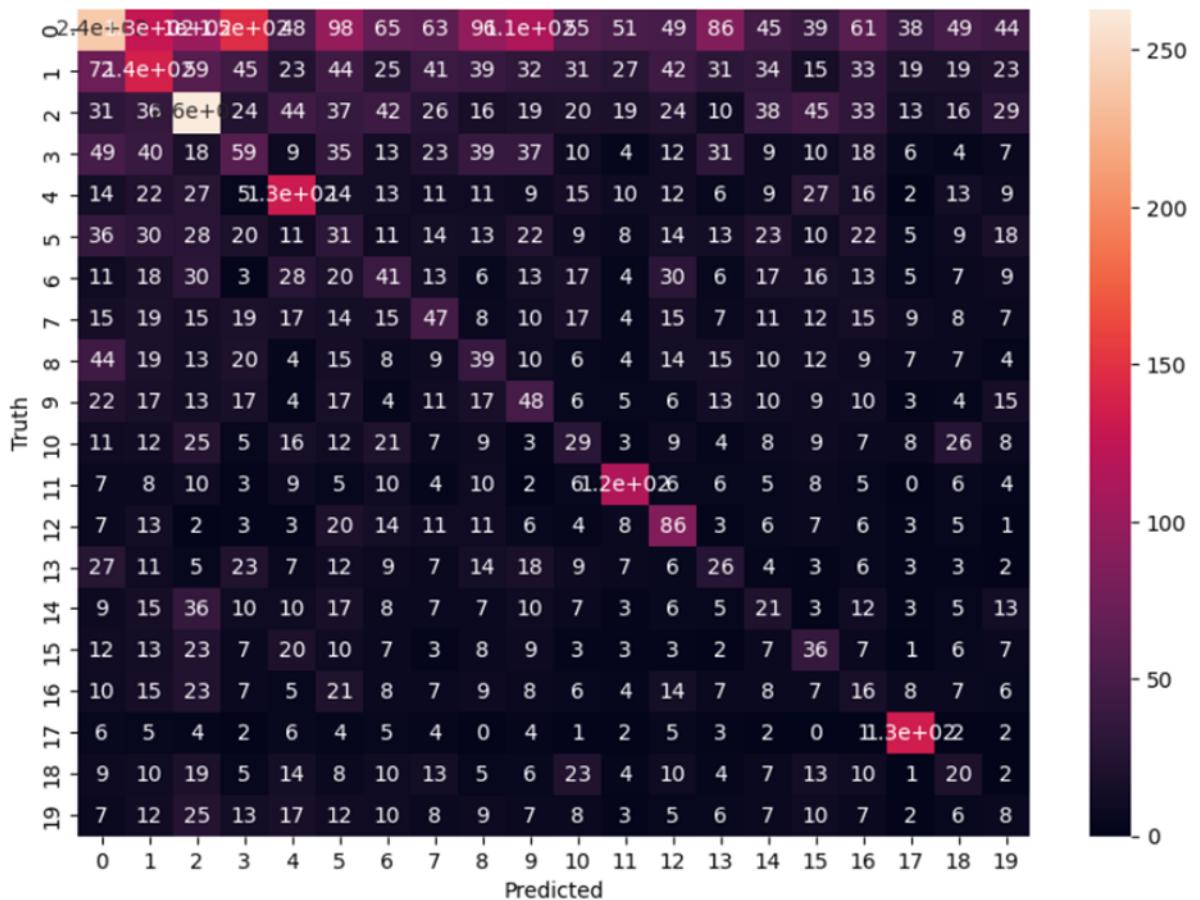
8.2.2.6.1 Kernel SIGMOID: SVM SIGMOID yöntemi için sınıflandırma raporu Şekil 8.13'de gösterilmiştir.

	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	7498
weighted avg		0.24	0.20	7498

Şekil 8.13 SVM'in kernel parametresi olarak SIGMOID ROS Sınıflandırma Raporu

SVM'in kernel parametresi olarak SIGMOID yöntemi için doğrulama tablosu Tablo 8.12'de gösterilmiştir.

Tablo 8.12 SVM Kernel SIGMOID ROS Doğruluk tablosu



8.2.2.7 SVM TF-IDF

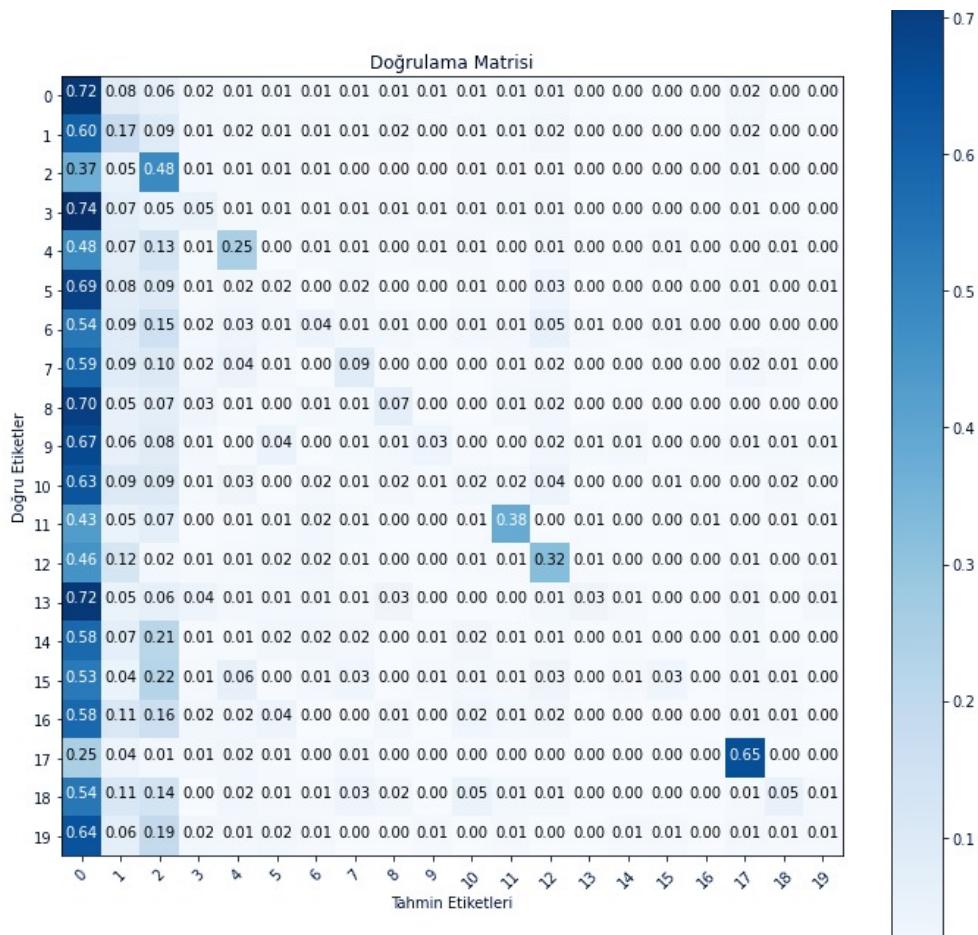
SVM lineer kernel yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.14'de gösterilmiştir.

	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

Şekil 8.14 SVM TF-IDF ROS Sınıflandırma Raporu

SVM yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.13'de gösterilmiştir.

Tablo 8.13 SVM TF-IDF ROS Doğruluk tablosu



8.2.2.8 KNN CV

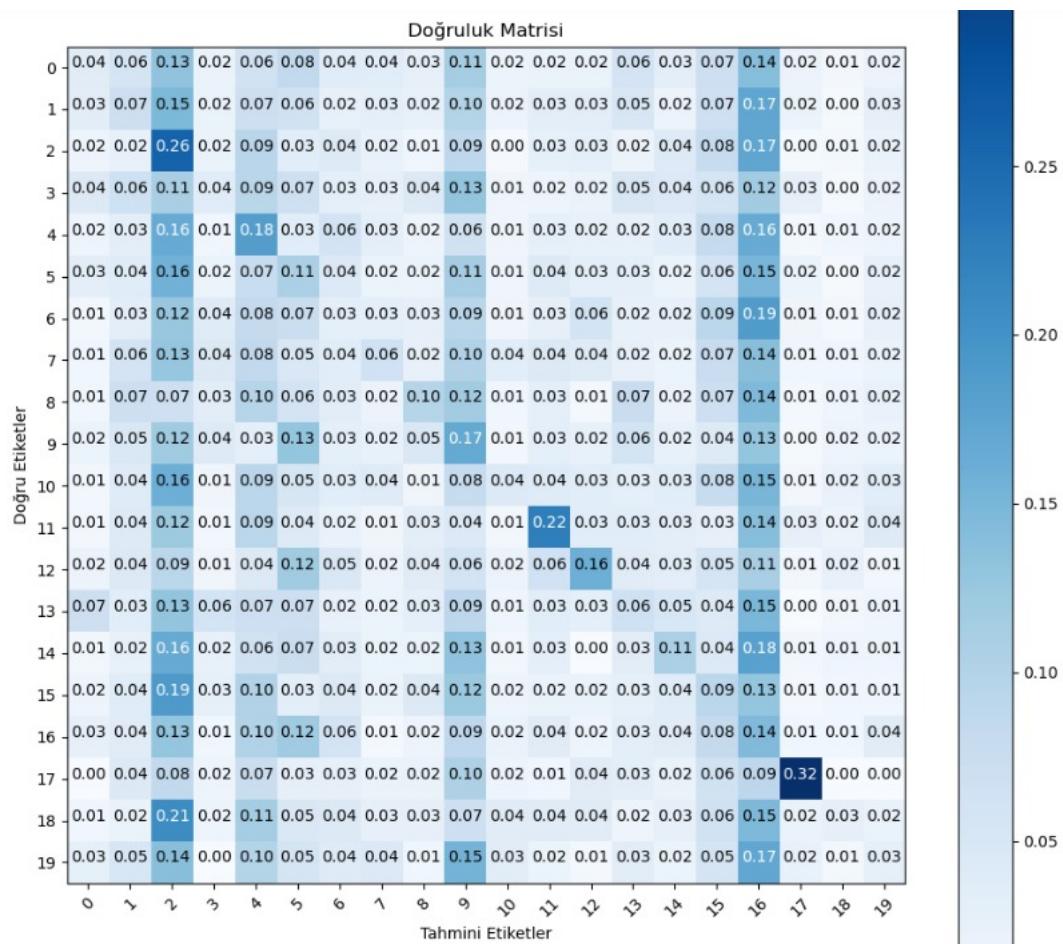
KNN yöntemi CV vektörizasyonu ile sınıflandırma raporu Şekil 8.15'de gösterilmiştir.

	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

Şekil 8.15 KNN ROS Sınıflandırma Raporu

KNN yöntemi için doğrulama tablosu Tablo 8.14'de gösterilmiştir.

Tablo 8.14 KNN ROS Doğruluk tablosu



8.2.2.9 KNN TF-IDF

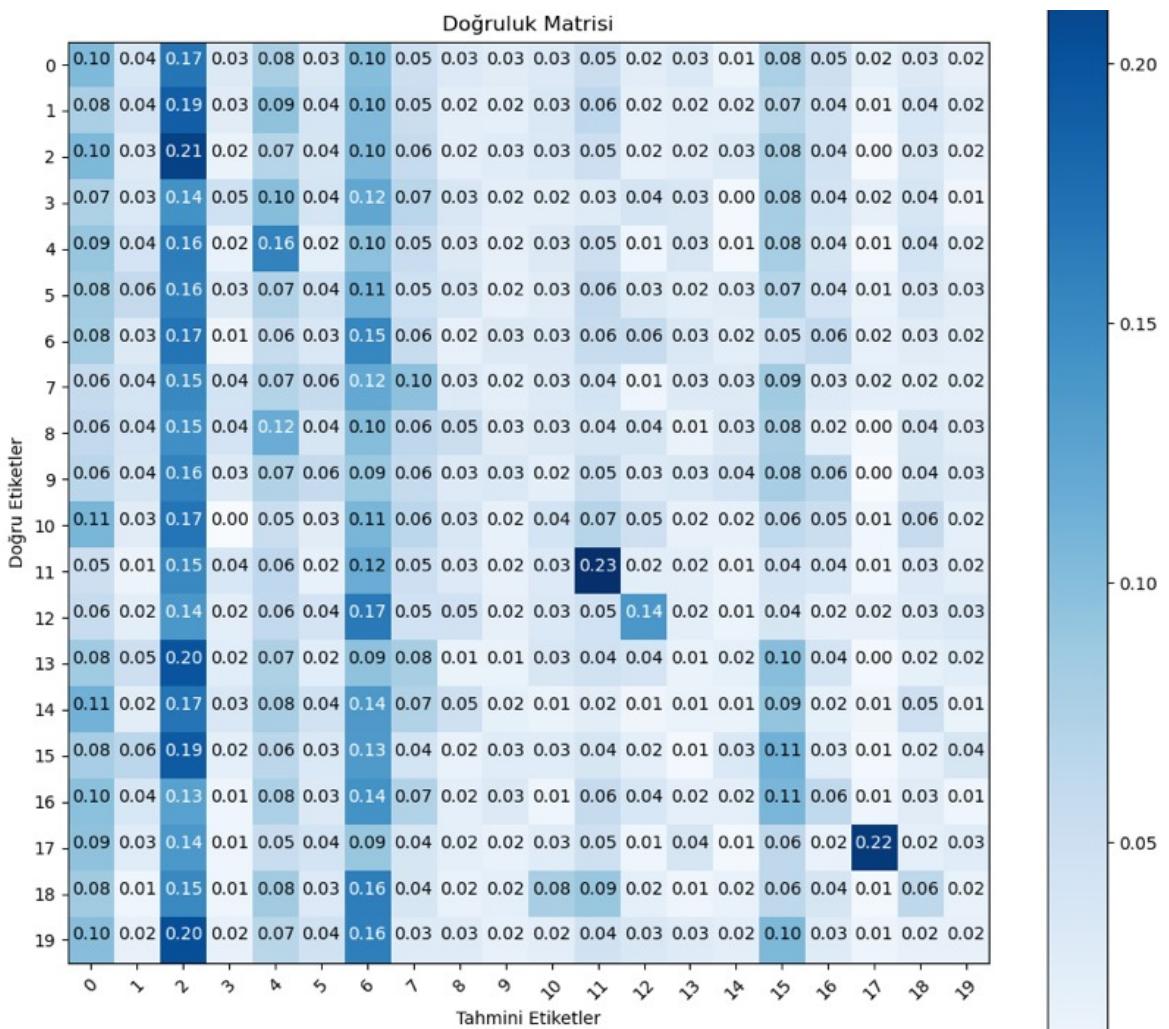
KNN yöntemiyle beraber TF-IDF vekktörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.16'da gösterilmiştir.

	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	7460
weighted avg		0.13	0.10	7460

Şekil 8.16 KNN-TFIDF ROS Sınıflandırma Raporu

KNN yöntemiyle beraber TF-IDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.15'de gösterilmiştir.

Tablo 8.15 KNN TF-IDF ROS Doğruluk tablosu



8.2.3 SMOTE

Görüldüğü üzere bazı sınıflarımızın ROS ile örneklenmesi istenilen sonuçlara ulaşırıamamıştır. Bu nedenle SMOTE yöntemi denenmiştir. SMOTE ROS'dan farklı olarak azınlık sınıflardaki aynı değerleri kopyalamak yerine, aynı azınlık sınıfına yeni veriler üretme esasına dayalıdır.

8.2.3.1 MNB CV

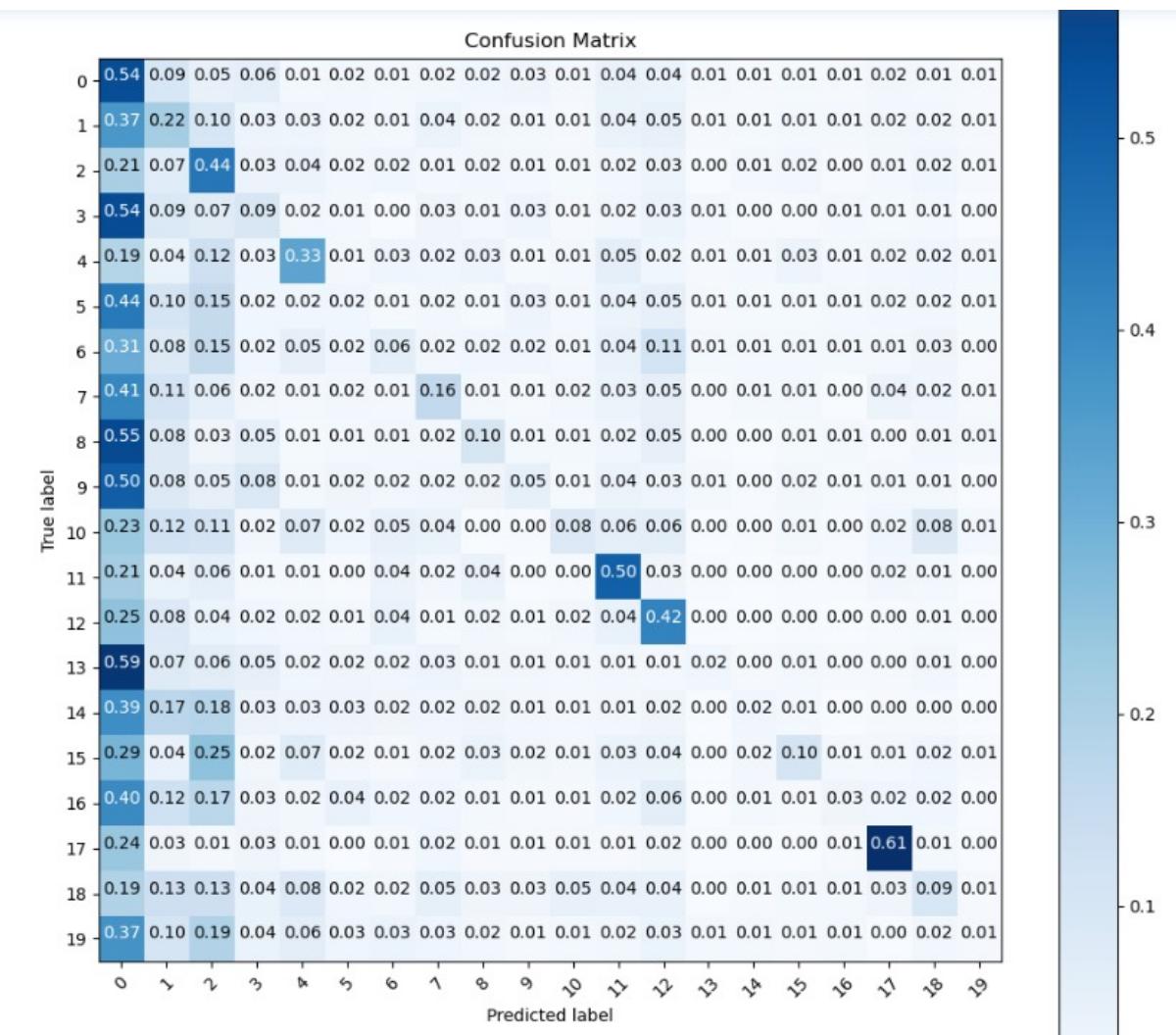
Multinomial Naive Bayes yöntemi CV vektörize işlemi sonrası sınıflandırma raporu Şekil 8.17'de gösterilmiştir.

	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

Şekil 8.17 Multinomial Naive Bayes SMOTE Sınıflandırma Raporu

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

Tablo 8.16 Multinomial Naive Bayes SMOTE Doğruluk tablosu



8.2.3.2 MNB TF-IDF

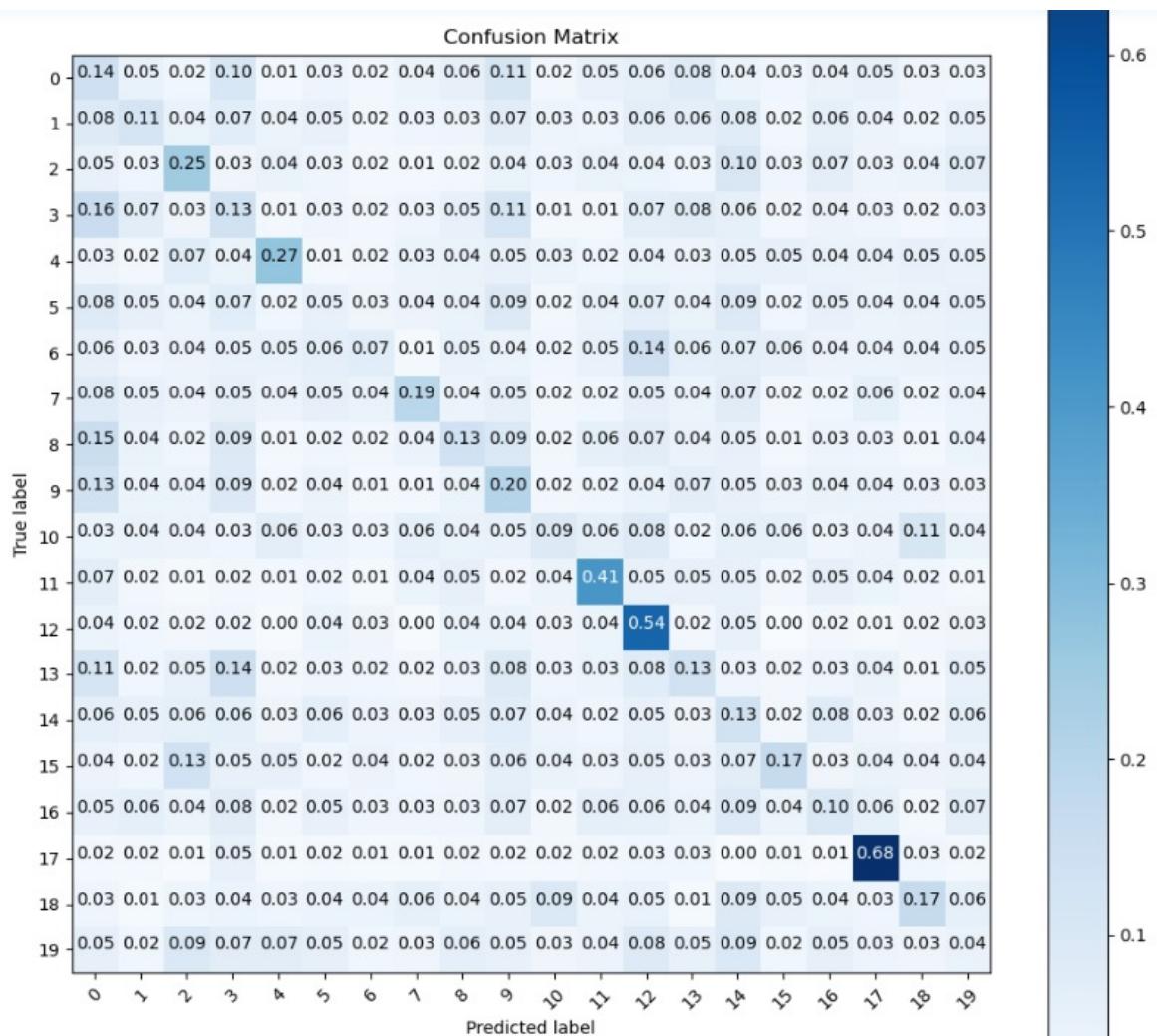
Multinomial Naive Bayes yöntemiyle beraber TF-IDF vekktörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.18'de gösterilmiştir.

	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

Şekil 8.18 MNB TF-IDF SMOTE Sınıflandırma Raporu

Multinomial Naive Bayes yöntemiyle beraber TF-IDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.17'da gösterilmiştir.

Tablo 8.17 Multinomial Naive Bayes-TFIDF SMOTE Doğruluk tablosu



8.2.3.3 SVM CV

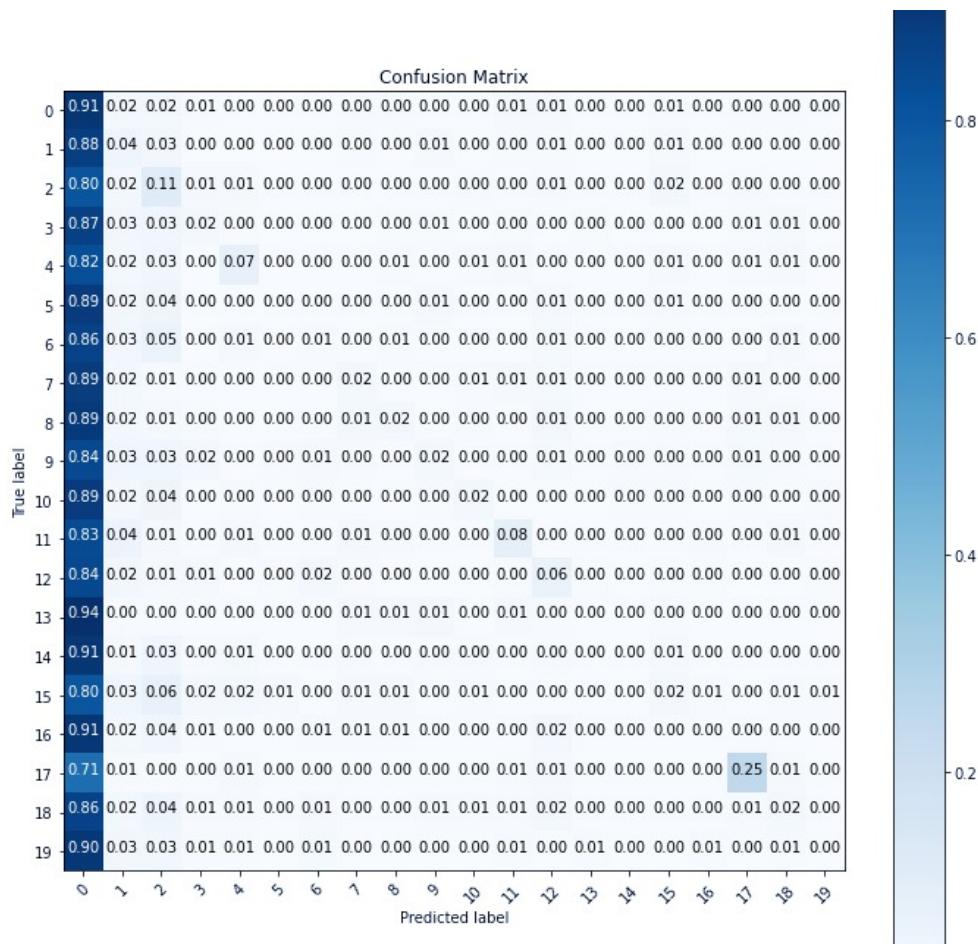
SVM yöntemi CV vektörize işlemi sonrası sınıflandırma raporu Şekil 8.19'de gösterilmiştir.

	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

Şekil 8.19 SVM CV SMOTE Sınıflandırma Raporu

SVM yöntemi için doğrulama tablosu Tablo 8.18'de gösterilmiştir.

Tablo 8.18 SVM CV SMOTE Doğruluk tablosu



8.2.3.4 SVM TF-IDF

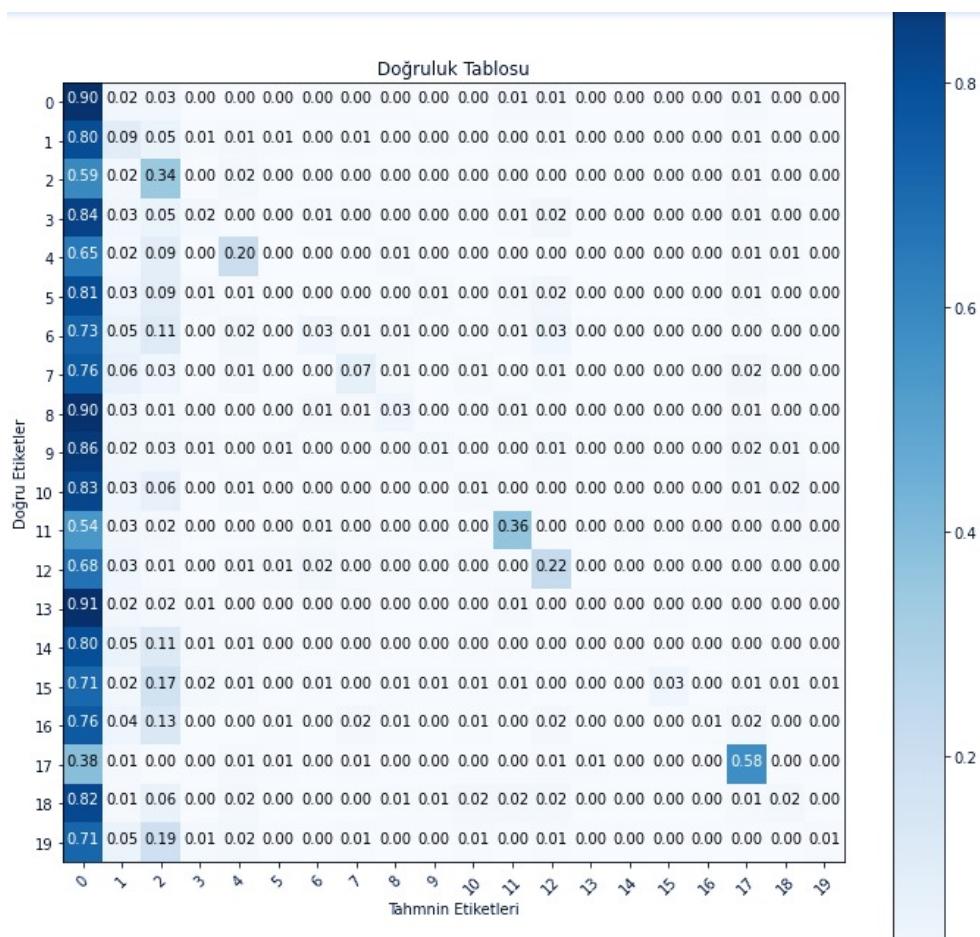
SVM yöntemiyle beraber TFIDF vekktörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.20'de gösterilmiştir.

	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

Şekil 8.20 SVM TF-IDF SMOTE Sınıflandırma Raporu

SVM yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.19'da gösterilmiştir.

Tablo 8.19 SVM TF-IDF SMOTE Doğruluk tablosu



8.2.3.5 KNN CV

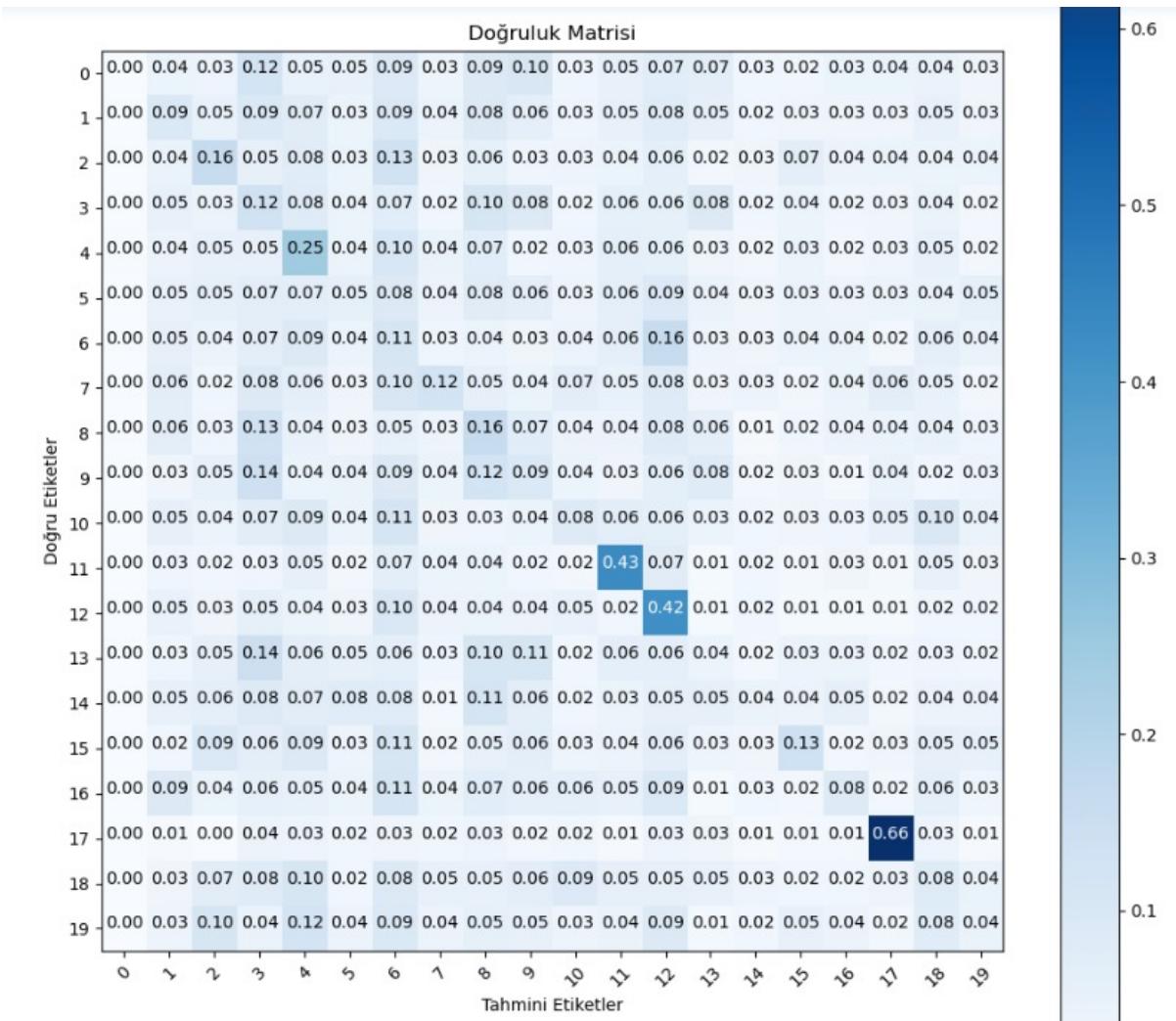
KNN yöntemiyle beraber CV vekötörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.21'de gösterilmiştir.

	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	7474
weighted avg		0.11	0.12	7474

Şekil 8.21 KNN CV SMOTE Normal Sınıflandırma Raporu

KNN yöntemiyle beraber CV vekötörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.20'de gösterilmiştir.

Tablo 8.20 KNN SMOTE Doğruluk tablosu



8.2.3.6 KNN TF-IDF

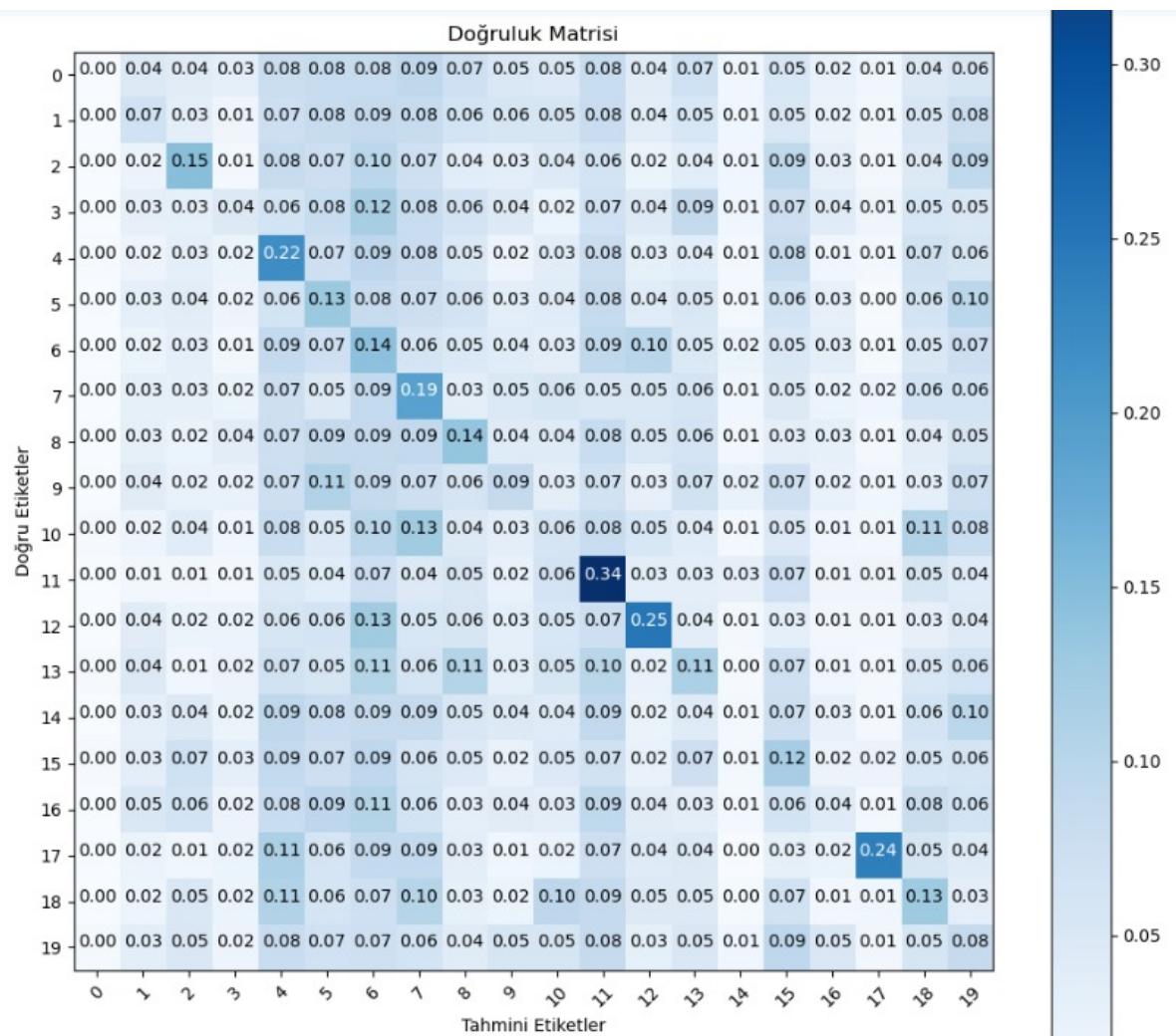
KNN yöntemiyle beraber TF-IDF vektörize yöntemi kullanıldığı durumda sınıflandırma raporu Şekil 8.22'de gösterilmiştir.

	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

Şekil 8.22 KNN TF-IDF SMOTE Sınıflandırma Raporu

KNN yöntemiyle beraber TFIDF vektörize yöntemi kullanıldığı durumda doğrulama tablosu Tablo 8.21'de gösterilmiştir.

Tablo 8.21 KNN TF-IDF SMOTE Doğruluk tablosu



8.2.4 Çapraz Doğrulama (Cross-Validation) Skorları

8.2.4.1 MNB

Veri seti Multinomial Naive Bayes ile Çapraz Doğrulama (Cross-Validation) işlemi uygulandığında elde edilen sonuç Şekil 8.23'de gösterilmiştir.

```
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])
```

Şekil 8.23 Multinomial Naive Bayes Çapraz Doğrulama Skoru

8.2.4.2 SVM

Veri seti SVM ile Çapraz Doğrulamaya sokulduğunda elde edilen sonuç Şekil 8.24'de gösterilmiştir.

```
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])
```

Şekil 8.24 SVM Çapraz Doğrulama Skoru

8.2.4.3 KNN

Veri seti KNN ile Çapraz Doğrulamaya sokulduğunda elde edilen sonuç Şekil 8.25'de gösterilmiştir.

```
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])
```

Şekil 8.25 KNN Çapraz Doğrulama Skoru

Çapraz Doğrulama skorlarına bakıldığından, bölüntülenmiş başarımlar birbirlerine yakın çıkmıştır. Bu da aslında her yöntem için verinin kendi içerisinde homojen dağıldığını ortaya koymaktadır.

8.3 Derin Öğrenme

Derin öğrenme, yapay sinir ağları kullanarak makine öğrenimi yöntemidir. Bu yöntemler, verilen bir veri setinde öğrenmeyi gerçekleştirirler ve veri setinde bulunan özellikleri ve öğrenme kurallarını öğrenirler. Öğrendiklerini yeni verilere uygulayarak, derin öğrenme yöntemleri çok sayıda veri setinde çok yüksek doğruluk oranları elde edebilirler. Derin öğrenme yöntemleri, özellikle ses, görüntü ve metin verilerinin işlenmesinde etkili bir şekilde kullanılmaktadır [18].

Yapay sinir ağları, bir makine öğrenimi modelidir. Bu model, bir çok sinir hücresinden oluşur ve bu sinir hücreleri birbirlerine bağlıdır. Yapay sinir ağları, sinir sistemine benzer şekilde çalışır ve verilen bir girdiye göre bir çıktı üretir. Yapay sinir ağının girdisi, özellikler veya nitelikler olarak adlandırılır ve çıktıları, modelin öğrendiği kurallar doğrultusunda üretilir. Yapay sinir ağları, makine öğrenimi yöntemlerinden biridir ve veri setlerinden öğrenmeyi gerçekleştirir [19].

RNN ve CNN iki farklı yapay sinir ağları modelidir. RNN, verilen girdi verisinin zamanla değişen bir sırası varsa kullanılır. Örneğin, bir dil modeli oluşturmak için kullanılır. RNN, dil modelinde kelime sırasını korur ve bu sayede anlamlı cümleler üretebilir [19].

CNN ise görüntülerin işlenmesi için kullanılır. CNN, girdi olarak verilen görüntülerin özelliklerini öğrenir ve bu özellikleri kullanarak görüntülerin sınıflandırılmasını yapar. Örneğin, görüntülerdeki nesnelerin tanımlanması için kullanılabilir [19].

RNN ve CNN arasındaki en önemli fark, girdi verilerinin tipidir. RNN, zamanla değişen veri sırasını işlerken, CNN görüntülerini işler. Bu nedenle, RNN dil modeli oluştururken kullanılırken, CNN ise görüntülerin sınıflandırılması için kullanılır.

LSTM (Long Short-Term Memory - Uzun Kısa Vadeli Bellek) bir RNN (Geri Beslemeli Sinir Ağısı) türüdür. LSTM, RNN'lerden daha ileri bir seviyedeki bir model olup, zamanla değişen veri sırasını daha etkili bir şekilde işleyebilir. LSTM, veri sırasının uzunluğunu ve zaman dilimlerini daha iyi anlayabilir ve bu sayede daha anlamlı çıktılar üretebilir. Örneğin, bir dil modelinde cümlelerin anlamını daha iyi anlayabilir ve daha anlamlı cümleler üretebilir. LSTM, RNN'den daha karmaşık bir model olup, daha iyi performans gösterebilir ancak bu aynı zamanda modelin eğitimi için daha fazla zaman ve kaynak gerektirir [20].

LSTM bir RNN türüdür ve bir yapay sinir ağları modelidir. LSTM katmanları, LSTM modelinin çalışma şeklini tanımlayan katmanlardır. Bir LSTM katmanı, bir dizi LSTM hücresinden oluşur ve bu hücreler birbirlerine bağlıdır. Her LSTM hücresi, girdi

verilerini işler ve çıktı verir. LSTM katmanı, bu hücrelerin çıktılarını birleştirerek bir sonuç üretir.

LSTM katmanları, bir çok veri setinde kullanılır ve özellikle zamanla değişen veri sırasını işlerken etkili bir şekilde kullanılır. Örneğin, bir dil modelinde kelime sırasını korur ve bu sayede anlamlı cümleler üretebilir. LSTM katmanları, ayrıca ses, görüntü ve metin verilerinin işlenmesinde de etkili bir şekilde kullanılabilir [20].

Derin öğrenme başlığı altında LSTM ile 3 katmanlı bir model tasarlayıp hem dengesiz konumda olan normal veri setimizi hem de ROS ile dengeleme işlemi yapılmış halinin LSTM başarı skorları paylaşılmıştır.

8.3.1 Normal Veri Kümesi Sonuçları

8.3.1.1 LSTM

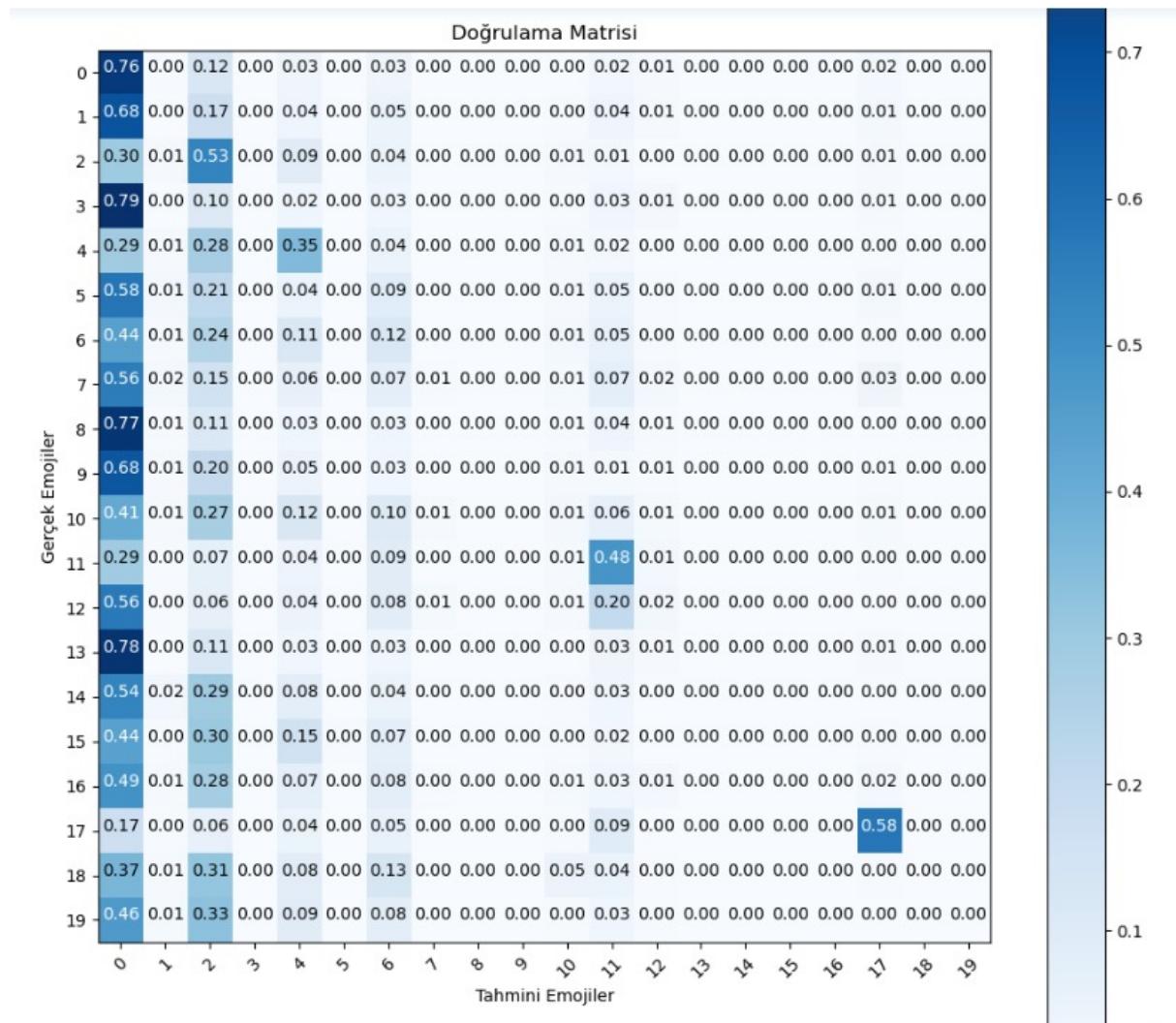
LSTM yöntemi için sınıflandırma raporu Şekil 8.26'da gösterilmiştir.

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

Şekil 8.26 LSTM Normal Veri Kümesi Sınıflandırma Raporu

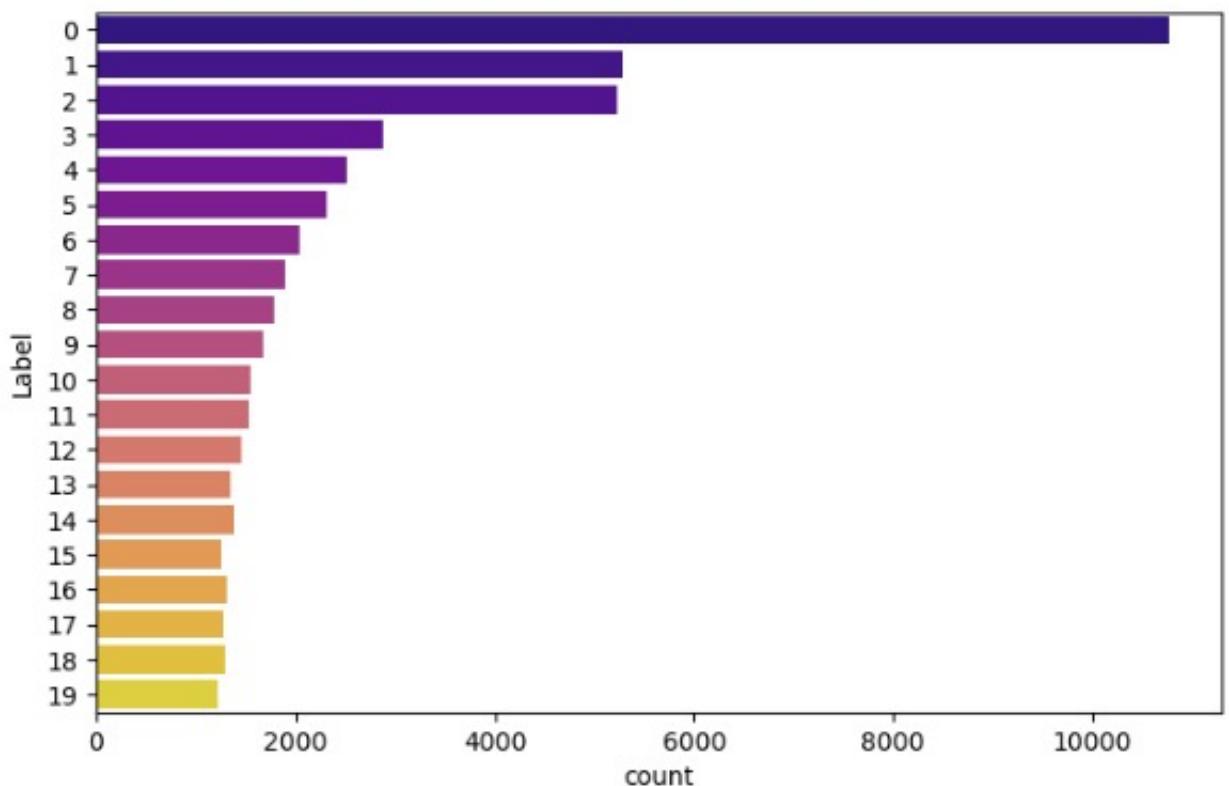
LSTM yöntemi için doğrulama tablosu Tablo 8.22'de gösterilmiştir.

Tablo 8.22 LSTM Normal Veri Kümesi Doğruluk tablosu

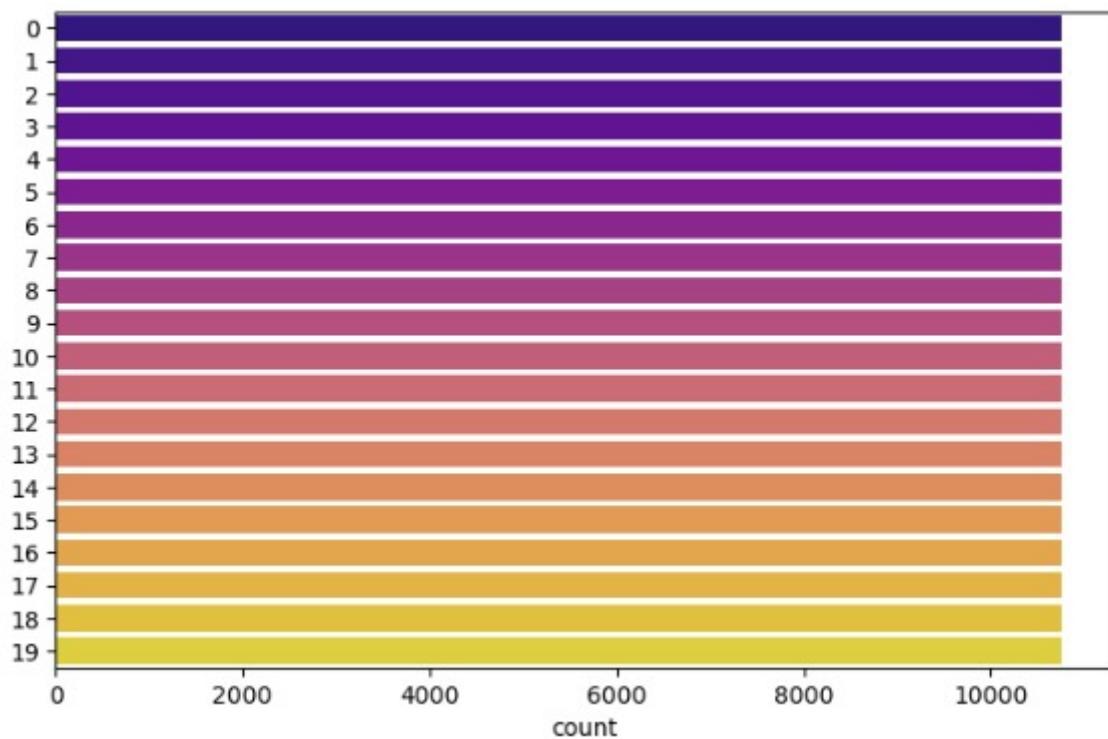


8.3.2 ROS Sonuçları

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

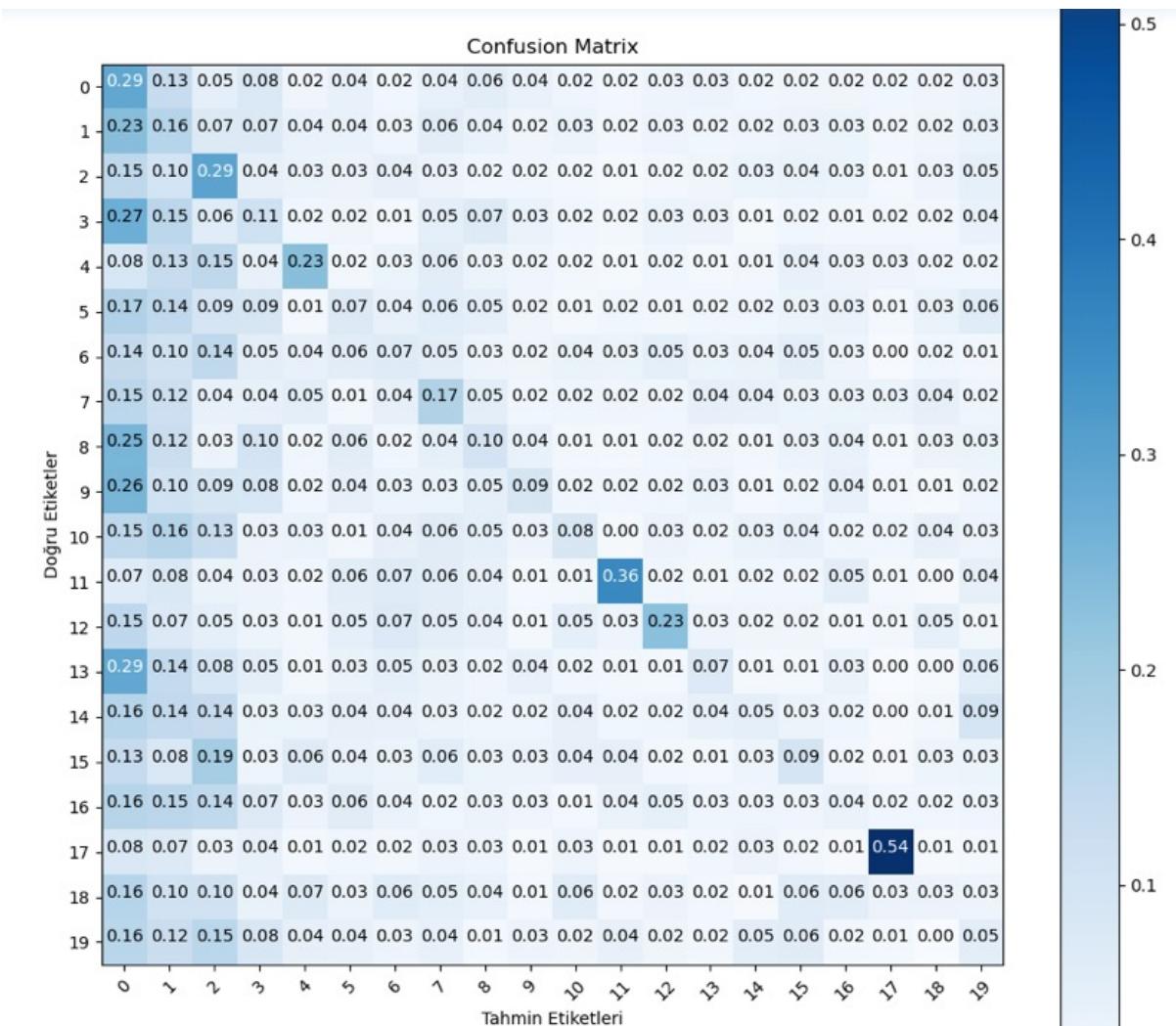
LSTM yöntemi için sınıflandırma raporu Şekil 8.29'da gösterilmiştir.

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
macro avg				0.16
weighted avg				0.19
				7474
				7474
				7474

Şekil 8.29 LSTM ROS Sınıflandırma Raporu

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 yorumlanmasılığını içermektedir.

9.1 Normal Sonuçlar

9.1.1 CV

Yöntemlerin CV vektörize yöntemiyle Normal f-1 skorları Tablo 9.1'de verilmiştir.

Tablo 9.1 Normal CV f-1 Skorları

<u>NORMAL</u>	SVM (%)	M. Naive Bayes (%)	KNN (%)	LSTM (%)
0	39	39	31	42
1	14	18	5	16
2	38	39	23	35
3	2	3	0	4
4	34	32	14	32
5	0	1	3	0
6	3	2	2	8
7	13	3	2	4
8	5	3	1	1
9	0	0	1	0
10	4	4	0	9
11	44	31	14	39
12	35	22	12	20
13	1	1	0	0
14	0	0	0	0
15	2	1	0	3
16	0	0	0	0
17	60	30	26	58
18	2	1	1	0
19	0	0	0	0

Genel olarak f-1 skorlar kıyaslandığında en yüksek başarımı SVM yöntemi elde etmiştir. Sonrasında MNB yöntemi gelmektedir ve ondan sonra da KNN yöntemi gelmektedir. MNB yöntemi çoğu emojide % 0 başarıım almasına rağmen, yüksek sayıya sahip emojilerde yüksek başarım aldığı için genel başarısı KNN yöntemine göre yüksektir. Emoji numara karşılıkları Tablo 9.2 de görülmektedir.

Tablo 9.2 Emoji İndeks Bilgileri

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
❤️	🙏	😂	💕	🔥	😊	😎	✨	💙	🥳	📷	🇺🇸	☀️	💜	☺️	💯	😊	🎄	🎁	😊

9.1.2 TF-IDF

Yöntemlerin TF-IDF vektörize yöntemiyle Normal f-1 Skorları Tablo 9.3'de verilmiştir.

Tablo 9.3 Normal TF-IDF f-1 Skorları

NORMAL	SVM (%)	M. Naive Bayes (%)	KNN (%)	LSTM (%)
0	39	39	31	42
1	14	18	5	16
2	38	39	23	35
3	2	3	0	4
4	34	32	14	32
5	0	1	3	0
6	3	2	2	8
7	13	3	2	4
8	5	3	1	1
9	0	0	1	0
10	4	4	0	9
11	44	31	14	39
12	35	22	12	20
13	1	1	0	0
14	0	0	0	0
15	2	1	0	3
16	0	0	0	0
17	60	30	26	58
18	2	1	1	0
19	0	0	0	0

Genel olarak f-1 skorlar kıyaslandığında en yüksek başarımı kıl payı SVM yöntemi elde etmiştir. Sonrasında MNB yöntemi gelmektedir ve ondan sonra da KNN yöntemi gelmektedir. Bu yöntemler arasında bakıldığından tüm yöntemler benzer emojilerde görece yüksek başarımlar yakalamışlardır. Yine benzer şekilde görece düşük başarım gösterdikleri emojilerde benzerdir.

9.2 ROS Sonuçları

Azılık sınıfımızın doldurulması için aynı değerler kopyalanarak çoğaltılması işlemine Random Over Sampling (ROS) denilmektedir.

9.2.1 CV

Yöntemlerin CV vektörize yöntemiyle ROS f-1 skorları Tablo 9.4'de verilmiştir.

Tablo 9.4 ROS CV f-1 Skorları

RANDOM OS	SVM (%)	M. Naive Bayes (%)	KNN (%)	LSTM (%)
0	38	17	7	30
1	22	17	9	15
2	38	34	22	30
3	9	15	5	10
4	37	37	14	26
5	7	10	9	8
6	9	11	3	7
7	13	16	7	15
8	13	14	11	9
9	8	13	8	10
10	11	14	6	9
11	41	35	21	37
12	32	31	16	23
13	3	9	5	8
14	1	7	10	5
15	11	14	5	8
16	3	6	4	4
17	62	51	35	51
18	7	12	4	3
19	0	2	3	4

Tablo 9.1'de de görüldüğü gibi ML algoritmaları arasındaki en yüksek başarımın SVM'de olduğu açıkça görülmektedir. Onun hemen ardından MNB 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 emojideki 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 MNB 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 emojideyse yine her yöntem, diğer emojilerle başarımlarına kıyasla, iyi bir başarım elde etmiştir. Bundaki en büyük etken benzer emojilerin bulunmamasıdır.

9.2.2 TF-IDF

Yöntemlerin TD-IDF vektörize yöntemiyle ROS f-1 skorları Tablo 9.5'de verilmiştir.

Tablo 9.3'de de görüldüğü gibi ML algoritmaları arasındaki en yüksek başarımın SVM'de olduğu yine açıkça görülmektedir. Onun hemen ardından MNB 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 MNB'nin başarımı anlamındaki fark düzeyi, MNB yöntemiyle KNN yöntemi arasındaki farka kıyasla daha fazladır. 11 numaralı emojideyse ML yöntemleri

Tablo 9.5 ROS TF-IDF f-1 Skorları

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

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

9.3 SMOTE Sonuçları

Azınlık sınıfı artırmaya yönelik bir diğer yöntem olan SMOTE ile azınlık olan sınıfın yönelik ROS'dan farklı olarak yeni değerleri üretip eklemeler yapılmıştır. ROS'da azınlık sınıfına aynı değerler kopyalanırken SMOTE de azınlık sınıfına yeni değerler eklenerek yapılmaktadır.

9.3.1 CV

Yöntemlerin CV vektörize yöntemiyle SMOTE f-1 skorları Tablo 9.6'de verilmiştir.

Tablo 9.6 SMOTE CV f-1 Skorları

<i>SMOTE</i>	SVM (%)	M. Naive Bayes (%)	KNN (%)
0	36	39	0
1	7	22	13
2	16	40	21
3	3	11	10
4	11	36	20
5	1	3	6
6	1	8	7
7	4	19	13
8	3	12	10
9	4	7	6
10	3	12	8
11	13	40	30
12	9	30	22
13	0	3	3
14	0	3	5
15	3	13	11
16	0	4	7
17	35	57	46
18	4	11	6
19	0	1	4

Genel olarak f-1 skorlar kıyaslandığında en yüksek başarımı bu sefer MNB yöntemi elde etmiştir. Sonrasında SVM yöntemi gelmektedir ve ondan sonra da KNN yöntemi gelmektedir. MNB yöntemi, en fazla adete sahip emojilerde ezici bir üstünlük sağlarken, KNN ise en fazla adete sahip olan 0 numaralı emojide çok düşük bir başarı yakalamıştır.

Yöntemlerin TF-IDF vektörize yöntemiyle SMOTE f-1 skorları Tablo 9.7'da verilmiştir.

9.3.2 TF-IDF

Tablo 9.7 SMOTE TF-IDF f-1 Skorları

<i>SMOTE</i>	SVM (%)	M. Naive Bayes (%)	KNN (%)
0	39	20	0
1	14	16	10
2	38	32	20
3	3	12	6
4	30	30	16
5	1	6	10
6	5	9	9
7	13	20	12
8	5	12	11
9	2	12	8
10	2	9	5
11	47	31	18
12	29	30	20
13	1	9	7
14	1	8	1
15	6	15	7
16	1	7	4
17	60	42	30
18	4	14	8
19	1	3	4

Genel olarak f-1 skorlar kıyaslandığında en yüksek başarımı SVM yöntemi elde etmiştir. Sonrasında MNB yöntemi gelmektedir ve ondan sonra da KNN yöntemi gelmektedir. SVM, en fazla adete sahip emojilerde daha yüksek bir başarı elde ederken, KNN yöntemi yine en fazla adete sahip olan 0 numaralı emojide çok düşük bir başarı yakalamıştır.

9.4 Genel Doğruluk Oranları

Yöntemlerin test edilen tüm durumlardaki doğruluk sonuçları Tablo 9.8'de verilmiştir.

Tablo 9.8 Doğrulama Sonuçları

ACCURACY	NORMAL (%)		RANDOM OS (%)		SMOTE (%)	
	Yöntem	CV	TF-IDF	CV	TF-IDF	CV
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

Tabloya bakıldığında, başarıım olarak en başarılı yöntem ML algoritması olan SVM'dir. Makine öğrenmesi yöntemlerini kıyaslayacak olursak, aralarında en başarılı olan yöntem normal veri setinde ufk bir farkla SVM olmuştur. 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, MNB ise ROS sonuçlarına bakıldığında çok düşük başarıım 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 başarıım vermiştir. ROS ve SMOTE teknikleri uygulandıktan sonra genel olarak başarımın düşmesinin sebebi veri test ettiğimiz veri setinde büyük oranda 0 numaralı emoji bulunmasıdır. Doğal 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.

9.5 Vektörize Yöntemleri Kıyaslama

Yöntemlerin vektörize yöntemlerine göre 4 emoji için doğrulama sonuçları Tablo 9.9'de verilmiştir.

Tablo 9.9 4 Emoji için f-1 Skor Sonuçları

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

4 numaralı emojiyi dikkate aldığımızda, MNB yönteminin aldığı başarı TF-IDF vektörize yöntemiyle, CV yöntemine göre çok düşmüştür. MNB gibi, SVM ve KNN yöntemlerinde de başarı 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 bakıldığında yine SVM ile TF-IDF vektörze yöntemi kullanıldığından başarı ufak da olsa artmıştır. MNB yöntemi ise hem CV, hem de TF-IDF vektörize yöntemleri için yakın sonuçlar vermiştir. KNN yöntemi de benzer şekilde gözükse de, başarı oranı oldukça düşmüştür. Benzer şekilde 17 numaralı emojiide de SVM ve TF-IDF ikilisi daha iyi başarı verirken, diğer yöntemler CV ile daha yüksek başarı vermiştir. Bu dört emojiyi dikkate alacak olursak, SVM ile birlikte TF-IDF vektörize yöntemini kullanmak daha iyi olacaktır denilebilir. Aynı zamanda KNN ve MNB yöntemleriyle de CV vektörize yöntemi kullanmak daha iyi sonuç verir denilebilir.

Referanslar

- [1] N. ÇALIŞKAN and R. Yeşil, “Eğitim sürecinde öğretmenin beden dili,” *Ahi Evran Üniversitesi Kırşehir Eğitim Fakültesi Dergisi*, vol. 6, no. 1, pp. 199–207, 2005.
- [2] S. F. Taşkıran, “Doğal dil işleme ile akademik metinlerin kümelenmesi,” M.S. thesis, Konya Teknik Üniversitesi, 2021.
- [3] A. E. ÖZMUTLU, “Doğal dil İşleme,” *Bilgisayar Bilimlerinde Teorik Ve Uygulamalı Araştırmalar*, p. 129,
- [4] A. Haberturk, *İnternette 1 dakikada neler oluyor?* <https://www.haberturk.com/internette-1-dakikada-neler-oluyor-3148965/>, [Online; accessed 25 November 2022], 2021.
- [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 (SemEval-2018)*, New Orleans, LA, United States: Association for Computational Linguistics, 2018.
- [8] 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.
- [9] F. Barbieri *et al.*, “SemEval 2018 task 2: Multilingual emoji prediction,” in *Proceedings of the 12th International Workshop on Semantic Evaluation*, New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 24–33. DOI: 10.18653/v1/S18-1003. [Online]. Available: <https://aclanthology.org/S18-1003>.
- [10] Ç. Cöltekin and T. Rama, “Tübingen-oslo at semeval-2018 task 2: Svms perform better than rnns in emoji prediction,” in *Proceedings of The 12th International Workshop on Semantic Evaluation*, 2018, pp. 34–38.
- [11] L. Alexa, A. B. Lorent, D. Gifu, and D. Trandabat, “The dabblers at semeval-2018 task 2: Multilingual emoji prediction,” in *Proceedings of The 12th International Workshop on Semantic Evaluation*, 2018, pp. 405–409.

- [12] S. Jin and T. Pedersen, “Duluth urop at semeval-2018 task 2: Multilingual emoji prediction with ensemble learning and oversampling,” *arXiv preprint arXiv:1805.10267*, 2018.
- [13] C. Baziotis, N. Athanasiou, G. Paraskevopoulos, N. Ellinas, A. Kolovou, and A. Potamianos, “Ntua-slp at semeval-2018 task 2: Predicting emojis using rnns with context-aware attention,” *arXiv preprint arXiv:1804.06657*, 2018.
- [14] 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.
- [15] M. Bozdemir, “Makine çevirisi ile türkçe sözel ifadelerin python sözdiziminin oluşturulması,” M.S. thesis, Bursa Uludağ Üniversitesi, 2022.
- [16] 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.
- [17] T. Demirhan, “Makine öğrenmesi algoritmalarının karmaşıklık ve doygunluk analizinin bir veri kümesi üzerinde gerçekleştirilmesi,” 2015.
- [18] A. Şeker, B. Diri, and H. H. Balık, “Derin öğrenme yöntemleri ve uygulamaları hakkında bir inceleme,” *Gazi Mühendislik Bilimleri Dergisi*, vol. 3, no. 3, pp. 47–64, 2017.
- [19] K. Öztürk and M. E. Şahin, “Yapay sinir ağları ve yapay zekâ’ya genel bir bakış,” *Takvim-i Vekayı*, vol. 6, no. 2, pp. 25–36, 2018.
- [20] C. Burcu, “Lstm ağları ile türkçe kök bulma,” *Bilişim Teknolojileri Dergisi*, vol. 12, no. 3, pp. 183–193, 2019.

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