

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BİR DUYGU ANALİZ YÖNTEMİ OLARAK METİNDEN
EMOJİ TAHMİNİ

20011701 – Muhammet Ali Şen
20011901 – Muhammet Kayra Bulut

BİLGİSAYAR PROJESİ

Danışman
Doç. Dr. Ali Can Karaca

Aralık, 2022

İÇİNDEKİLER

KISALTMA LİSTESİ	iv
ŞEKİL LİSTESİ	v
TABLO LİSTESİ	vi
1 Giriş	1
1.1 Doğal Dil İşleme Nedir?	1
1.2 Doğal Dil İşleme Uygulamaları	2
1.3 Projenin Amacı	4
2 Ön İnceleme	6
2.1 Projeye Olan İhtiyaç	6
2.2 Proje Kapsamı	6
2.3 Projenin Gereksinimleri	7
3 Fizibilite	8
3.1 Teknik Fizibilite	8
3.1.1 Yazılım Fizibilitesi	8
3.1.2 Donanım Fizibilitesi	8
3.2 İş Gücü ve Zaman Fizibilitesi	9
3.3 Ekonomik Fizibilite	9
3.4 Yasal Fizibilite	10
4 Sistem Analizi	11
4.1 Gereksinimler	11
4.2 Hedefler	12
4.3 Performans Metrikleri	13
5 Sistem Tasarımı	14
5.1 Yazılım Tasarımı	14
5.1.1 LSTM	15
5.1.2 Naive Bayes	16

6 Uygulama	17
6.1 Modellerin Özellikleri	17
6.2 Modellerin Eğitilmesi İçin Kullanılan Yöntem	18
6.3 Modellerin Eğitilmesi	18
Referanslar	19

KISALTMA LİSTESİ

AI	Artificial Intelligence
CL	Computational Linguistics
CPU	Central Processing Unit
DL	Deep Learning
GPU	Graphics Processing Unit
IT	Information Technology
LSTM	Long Short-Term Memory
ML	Machine Learning
NLG	Natural Language Generation
NLP	Natural Language Processing
NLU	Natural Language Understanding
SVM	Support Vector Machine
TPU	Tensor Processing Units

ŞEKİL LİSTESİ

Şekil 2.1	Google Colab ve Python	7
Şekil 3.1	Proje İş/Zaman Çizelgesi	9
Şekil 4.1	Örnek Emoji Oranları	11
Şekil 4.2	Şekil 4.1'in Emoji Karşılıkları	12
Şekil 4.3	Örnek uygun emoji bulma oranları	13
Şekil 4.4	Örnek emojiler [3]	13
Şekil 5.1	Metin Sınıflandırma Akış Şeması	14
Şekil 5.2	Bayes Teroemi	16
Şekil 6.1	SVM ile elde edilen başarımlar	17
Şekil 6.2	Naive Bayes ile elde edilen başarımlar	18

TABLO LİSTESİ

Tablo 1.1	Dijital Mecralarda Dakikalık Üretilen İçerik Sayısı [1]	5
Tablo 3.1	Ekonomik Fizibilite Tablosu	10

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

1.1 Doğal Dil İşleme Nedir?

Doğal dil işleme (NLP) yapay zekanın (AI) bir koludur ve bilgisayarların, insan dilinin yapısını kavramasını ve insan dilini anlayarak çıktılar verebilmesini sağlar. 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 ardındaki temel teknoloji doğal dil işlemedir. Bu sistemler ve asistanlara sorular sorulduğunda hem kullanıcının talebini anlayıp hem de doğal bir dille yanıt vermesini sağlayan doğal dil işleme 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, duygu analizi ve dil bilgisi/yazım denetimi gibi araçlar örnek olarak verilebilir. Mesela, kimi e-posta programları mesaj içeriğine göre uygun yanıt tahminlerinde bulunabilir. Bu araçlar, mesajınızı okumak, anlamlandırmak ve yanıtlamak için doğal dil işleme 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 anlama demektir.

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

dil bilgisi" olarak bilinen bir kavram vasıtasıyla anlamlı bilgileri özütleyerek metne dönüştüren "dil çıktısı" da denir.

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

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 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 bugünlerde 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.

Aramayı optimizasyonu:

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

Arama motoru iyileřtirmesi:

Doęal dil iřleme arama sonularında daha st sıralarda grnmeniz iin ok elveriřli bir aratır. Arama motorları sonularını dzenlemek amacıyla doęal dil iřleme yntemlerinden faydalanır. Aynı zamanda bu yntemleri nasıl daha efektif řekilde kullanılacaęını bilmek arama sonucundan st sıralarda ıkmak aısından nem arz etmektedir. Bu vesileyle istenilen sonular daha yksek dzeyde fark edilirlie lařır.

Byk veri analizi: Belge sınıflandırma ve ierik modelleme gibi doęal dil iřleme teknikleri iřletme ii raporlar, haber sunumları veya bilimsel makaleler gibi byk belge koleksiyonlarında ierik karmařıklieını analiz etme misyonunu kolaylařtırır.

Sosyal aę analizleri:

Doęal dil iřleme kullanıcı incelemelerini ve sosyal medya ieriklerini yorumlayarak ve anlamlandırarak byk veriler konusunda ok iyi bir anlayıřa kavuřmanızı saęlayabilir. Duygu analizi, sosyal medyada yer alan ieriklerdeki bařta olumlu/olumsuz ierikleri dahası dięer duyguları tespit ederek kullanıcıların isteklerinin doęrultusunda anlık doęru lm gerekleřtirir. Bu, zamanla ok iyi edinimler saęlayabilir.

Pazar tahminleri:

Hedef pazarınızın dilini analiz etmek zere doęal dil iřleme'den faydalanarak hedef kitlenizin taleplerini daha doęru tespit eder ve aynı zamanda hedef kitlenizle nasıl daha iyi etkileřim kurabileceęinizi anlamıř olursunuz. İlgi doęrultulu duygu analizi, sosyal mecralardaki belli bařlı ilgi alanları veya rnlerle alakalı duyguyu (rneęin, "kulaklık muhteřem ama mikrofonu cızırtılı") tespit ederek rn modellemesi ve pazarlama iin temel veriler sunar.

İerięi ynetme:

Kurumunuz ok fazla mřteri yorumu veya mail gibi doęrudan metinsel mesajlar alıyorsa doęal dil iřleme, kelimelerle birlikte metinsel mesajların da duygusunu ve hedefini de deęerlendirerek ierięi dzenlemenize imkan verir.

1.3 Projenin Amacı

Gündelik hayatta dijital teknolojilerin yaygınlaşmasıyla birlikte dijital dönüşümler hız kazanmaktadır. Artık birçok işletme stratejilerini dijital platformlara taşımakta ve dijital platformlar aracılığıyla yapılan geri bildirimlerin önemi şirketler için çok önemli hale gelmektedir. Aynı zamanda şirketlerin dijital mecralara önem vermesiyle, dijital mecraların kullanıcı sayısı beklenmedik ölçüde artmaktadır ve kullanıcıların dijital mecraları tercih oranını artmasıyla da, şirketlerin dijital mecraları diğer şirketler de tercih etmeye ve bu yönde geliştirmeler yapmaya başlamıştır. Bu şekilde beslemeli bir döngüye sahip olan dijital mecralarda kullanıcı sayısının artmasıyla ve web 2.0 teknolojilerinin güncel hayata 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 hayatta 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 uydurmayan şirketler tedricen pazar paylarını kaybetmektedir. Kullanıcıların içerik üretebileceği bir platform olan Twitter’da bile dakikada yaklaşık 200.000[1] 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. Bu kadar yüksek hacimli verilerin sadece metin içeren kısımları dahi çok yekûn tutmaktadır. Doğal dil işleme yöntemleriyle bu verilerin anlamlandırılması kolaylaşmıştır. Bunun dışında bilişim sistemlerinin insan dilinde konuşulan veya yazılan içerikleri en yakın anlamıyla birebire yakın şekilde anlaması ve sonuç üretmesi çok önemlidir. Bilişim sistemlerinin bunu daha doğru anlaması ve dönüt vermesi için makine öğrenimi ve derin öğrenme gibi yöntemlerden faydalanılmaktadır. İletişimde verilmek istenen mesajın sadece %7’si kelimelerden oluşmaktadır[2] . Bunun haricinde beden dili, ses tonları ve mimikler iletilmek istenen mesajın çoğunu oluşturmaktadır. Kısacası 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. Bu çalışmamızda emoji ile duygulandırılmış metinler üzerinde çeşitli makine ve derin öğrenme yöntemleri kullanılarak metinlerin duyguları tahmin edilmek istenmiş ve bu şekilde yapılan duygu analiz yöntemiyle başarı oranı hesaplanmak istenmiştir.

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

Tablo 1.1 Dijital Mecralarda Dakikalık Üretilen İçerik Sayısı [1]

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 yansıtmaması nedeniyle emojiiler kullanılmakta ve metne duygu verilmek istenmektedir. Ancak bu şekilde kullanılmamış sadece metinlerde ibaret yazıların olumlu/olumsuz başta olmak üzere hangi duygu içerdiğini hesaplamak oldukça zordur. Bu duygu etiketleme işlemi için genelde işletmeler kullanıcı yorumlarını okuyup tasnifleyen çalışanlar istihdam etmektedir. Duygu etiketleme işlemlerini yapay zekâ yöntemleriyle makineye öğretmek ve öğrenilen, modellenen makine üzerinden çıkarımlarda bulunma ihtiyacı doğmuştur.

2.2 Proje Kapsamı

Projemizde yapılması amaçlanan sistemden beklenenler şunlardır:

- Sistemin cümleyi/cümleleri/kelimeleri anlamlandırması.
- Sistemin çeşitli makine ve derin öğrenme algoritmalarıyla sınıflandırılmış/kümelenmiş veriler benzerlik tespit etmesi.
- Sınıflandırılmamış veya etiketlenmemiş metinlere emoji tahmininde bulunması.
- Birden fazla makine, derin öğrenme yöntemiyle tasarlanan sistemin en optimum sonuçlara ulaşması için kullanılan yöntemlerin karşılaştırılması.

2.3 Projenin Gereksinimleri

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



Şekil 2.1 Google Colab ve Python

3.1 Teknik Fizibilite

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

3.1.1 Yazılım Fizibilitesi

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

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

3.1.2 Donanım Fizibilitesi

Model eğitimi çok fazla işlem gücü gerektiren bir durum olduğundan dolayı, işlem hacmi fevkalade GPU'ların kullanılması elzemdir. 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 GPU ve TPU kullanılması büyük avantaj sağlayacaktır. Projede ise içinde iyi işlem hacmine sahip bir CPU ve GPU bulunduran bir bilgisayarın yanı sıra, TPU ve GPU desteği sağlayan çevrimiçi hizmetler de kullanılmıştır. Bu projede birden fazla insan emek sarfedeceğinden dolayı, yapılan işlemleri ve yazılan kodları hızlı ve efektif olarak tutabilmek için bulut sistemi kullanılmıştır. Bu sistemde kodlardaki ilerlemeler de eşzamanlı olarak tutulabilmektedir.

3.2 İş Gücü ve Zaman Fizibilitesi

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

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												

Şekil 3.1 Proje İş/Zaman Çizelgesi

3.3 Ekonomik Fizibilite

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

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

Projede kullandığımız veri seti de açık kaynaklı ve ücretsizdir.

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

Detaylı maliyet tablosu Tablo 3.1’de gösterilmiştir.

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
Acet Nitro AN515-44	1	9500 TL	9500 TL
Toplam Maliyet			11500 TL

Tablo 3.1 Ekonomik Fizibilite Tablosu

3.4 Yasal Fizibilite

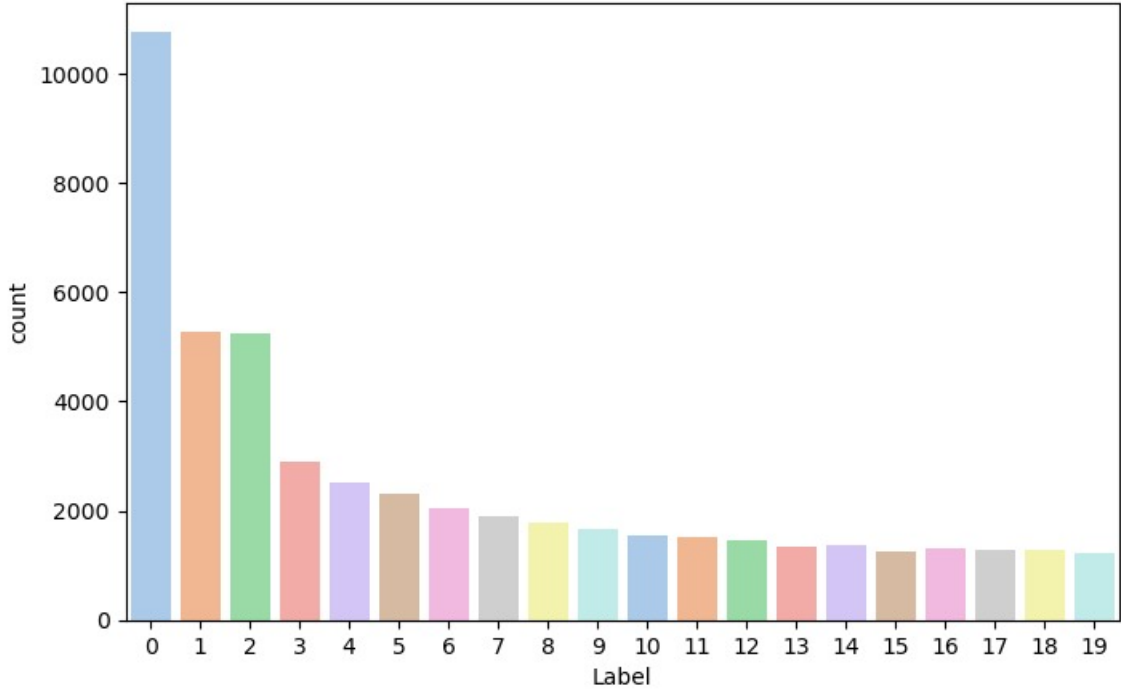
Projede kullanılan veri seti CodaLab [3] üzerinden alınmıştır. Açık kaynaklı olduğu için, kullanımı herhangi bir yasal sorun teşkil etmemektedir.

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. Projede kullanılan veri seti de yine açık kaynaklı veri setidir.

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

4.1 Gereksinimler

Metinlerin içindeki kelimelerin, vektörel karşılıkları kullanılarak, metinlerin kelime kelime ayrıştırılarak, önce kelimelerin ayrı ayrı duygu analizi çıkarımlarıyla ve çeşitli eğitim metotlarıyla emojiyle 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.



Şekil 4.1 Örnek Emoji Oranları

0	❤️	Red heart
1	😍	Smiling face with hearteyes
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

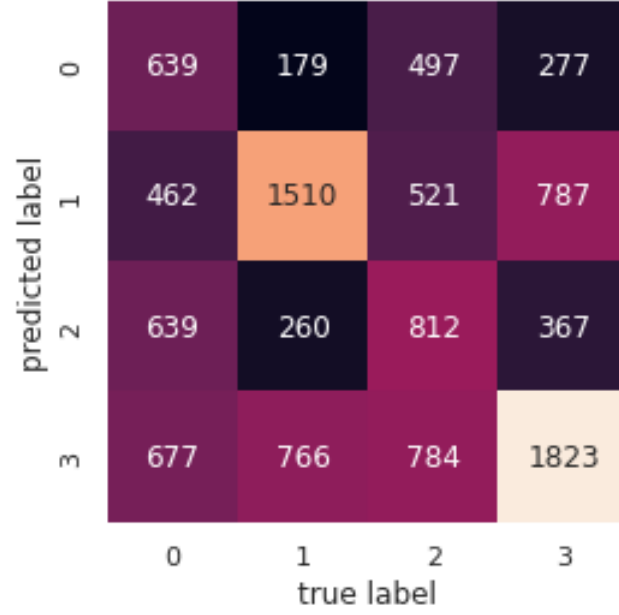
Şekil 4.2 Şekil 4.1'in Emoji Karşılıkları
[3]

4.2 Hedefler

Sınıflandırılmış metinlerin yardımıyla, yeni gelen metinlere en uygun emojiyi bulmak.

Emojilerden yola çıkarak, metinleri duygusal olarak sınıflandırmak.

Şekil 4.3'de örnek uygun emoji bulma oranları gösterilmiştir.



Şekil 4.3 Örnek uygun emoji bulma oranları

4.3 Performans Metrikleri

Modellerden beklenen şey, metinlerin emoji karşılıklarını düzgün şekilde gösterebilmesidir. Bu metriği, modelleri kıyaslamada kullanacağız. Şekil 4.3’de örnek emoji gösterilmiştir.

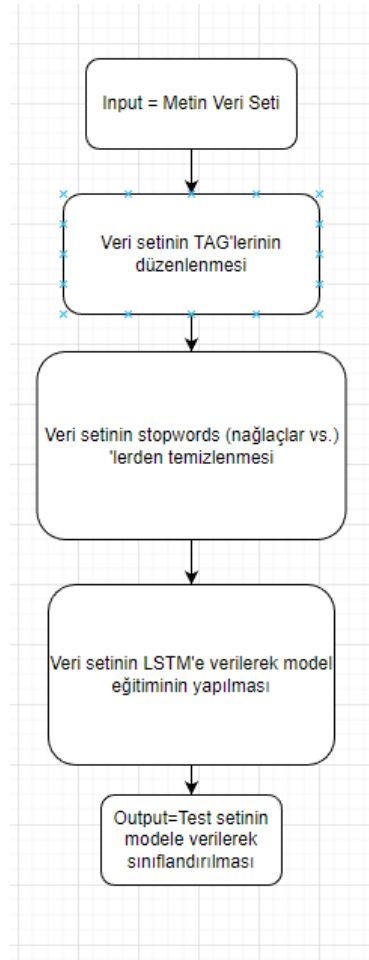
0	0	😄	0
1	1	📺	1
2	2	😄	2
3	3	😄	3
4	4	😄	4
5	5	🌲	5
6	6	📺	6
7	7	🔥	7
8	8	😄	8
9	9	❤️	9
10	10	😄	10

Şekil 4.4 Örnek emoji [3]

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 gibi birçok öğrenme yöntemi kullanılmıştır.



Şekil 5.1 Metin Sınıflandırma Akış Şeması

5.1.1 LSTM

LSTM derin öğrenme alanında kullanılan yapay bir yinelemeli sinir ağı (RNN) mimarisidir. LSTM geri beslemeli çalışır. Anlık verinin yanında veri dizilerini de işleyebilmektedir. Sıradan bir LSTM ünitesi giriş, çıkış ve unut kapısından oluşmaktadır[4].

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 karşılaşılabilecek unutulmuş ve yok olan gradyan sorunlarını çözmek için üretilmiştir

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

LSTM bu problemi dört adımda çözüyor[5], bu adımlar:

- Forget Gate (Unutma Kapısı):

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

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

- Input Gate (Girdi Kapısı):

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

- Cell State:

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

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

- Output Gate(Çıktı Kapısı):

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

5.1.2 Naive Bayes

İsmi matematiği 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[6]. Naive Bayes algoritmasının temeli olan Bayes Teoreminin matematiksel karşılığı şekil 5.4 de mevcuttur.

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

Şekil 5.2 Bayes Teroemi

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

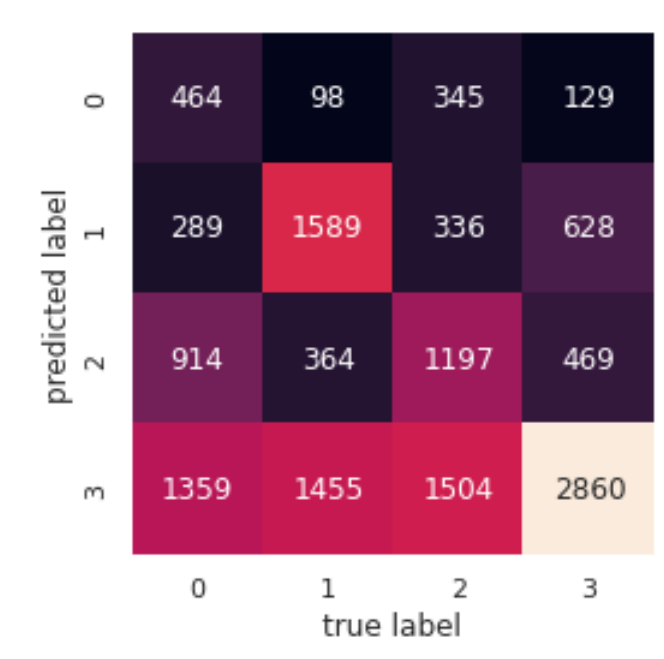
Naive Bayes algoritması öncelikle kullanım kolaylığı nedeniyle tercih sebebi olmaktadır. İkinci olarak ise diğer sınıflandırma yapan diğer makine öğrenmesi algoritmalarına karşılık eğitim verisinin yalnızca bir kez taranması yeterlidir. Ayrıca empty (kayıp - boş) veriler de olasılık hesaplarına katılmayarak ele alınabilmektedir. Basit ilişkilerin olduğu durumlarda genellikle iyi sonuç çıkaran bir yöntemdir[7].

Bu bölümde modelin özellikleri ve projenin aşamalarıyla alakalı bilgi verilmektedir.

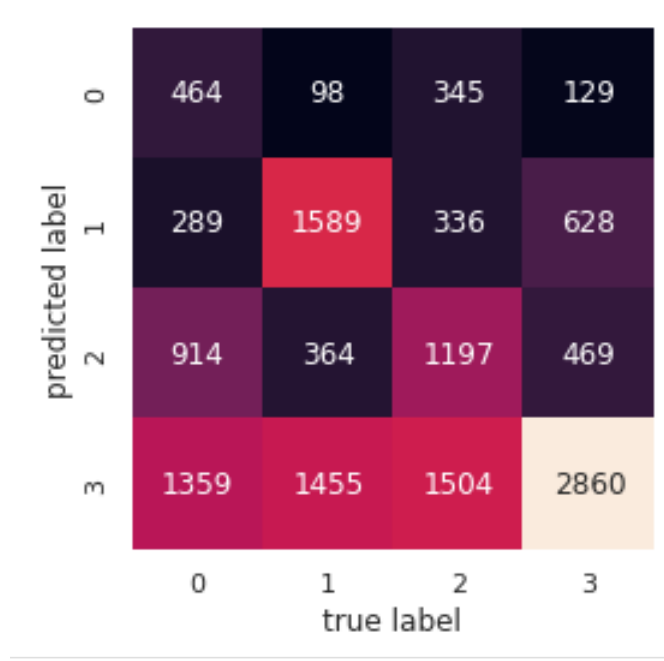
6.1 Modellerin Özellikleri

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

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



Şekil 6.1 SVM ile elde edilen başarımlar



Şekil 6.2 Naive Bayes ile elde edilen başarımlar

6.2 Modellerin Eğitilmesi İçin Kullanılan Yöntem

Eğitim sürecinde 50.000 adet veri setinin bir kısmı test (

Bu yöntemler elde ettikleri doğruluk oranlarına göre kıyaslanmıştır.

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

6.3 Modellerin Eğitilmesi

Eğitim aşaması çok fazla işlem gücü gerektirdiğinden, google colab'ın sunduğu GPU kullanılmıştır

Eğitim sırasında overfit olduğu görüldüğünden, epoch değeri belli bir sayıyla tutularak, data setindeki sınıf sayısı düşürülerek veya doğrulama veri setimizin rastgele alınması gibi yöntemlerle sorun aşılma çalışılmıştır.

Referanslar

- [1] A. Haberturk, *İnternette 1 dakikada neler oluyor?* <https://www.haberturk.com/internette-1-dakikada-neler-oluyor-3148965/>, [Online; accessed 25 November 2022], 2021.
- [2] Y. Kocabaş, “Etkili iletişimin sözsüz adımı olan beden dili ve türkçe eğitimindeki rolü,” *HAYEF Journal of Education*, vol. 4, no. 1,
- [3] F. R. Francesco Barbieri Luis-Espinoza-Anke and H. Saggion, *SemEval-2018 Task 2, Multilingual Emoji Prediction*, https://competitions.codalab.org/competitions/17344#learn_the_details-overview, [Online; accessed 25 November 2022], 2018.
- [4] 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.
- [5] M. Bozdemir, “Makine çevirisi ile türkçe sözel ifadelerin python sözdiziminin oluşturulması,” M.S. thesis, Bursa Uludağ Üniversitesi, 2022.
- [6] E. Günçe and A. Carus, “Twitter platformu üzerinden makine öğrenmesi algoritmaları ile cinsiyet ve ilgi alanı analizi,” M.S. thesis, Trakya Üniversitesi Fen Bilimleri Enstitüsü, 2021.
- [7] T. Demirhan, “Makine öğrenmesi algoritmalarının karmaşıklık ve doygunluk analizinin bir veri kümesi üzerinde gerçekleştirilmesi,” 2015.