

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



KIYAFET ÖNERİ SİSTEMİ GELİŞTİRME PROJESİ

19011501 – Doğa GÜNDOĞAR
19011502 – Gülsüm İrem BAŞ

BİLGİSAYAR PROJESİ

Danışman
Prof. Dr. Banu DİRİ

Haziran, 2023

TEŞEKKÜR

Proje sürecinde bizlere her daim destek olan, alanındaki bilgi birikimi ve uzmanlığını bizlerle paylaşarak projenin gerçekleştirme aşamasında bizlere yol gösteren danışman hocamız Prof. Dr. Banu Diri'ye teşekkür ederiz.

Doğa GÜNDÖĞAR
Gölsüm İrem BAŞ

İÇİNDEKİLER

KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
TABLO LİSTESİ	vii
ÖZET	viii
ABSTRACT	ix
1 Giriş	1
1.1 Proje Motivasyonu ve Kapsam	1
1.2 Yapım Aşamasında Karşılaşılabilecek Zorluklar	2
1.3 Literatür Taraması	2
2 Ön İnceleme	4
2.1 Projeye Duyulan İhtiyaç	4
2.2 Proje Kapsamı	5
2.3 Yol Haritası	5
3 Fizibilite	6
3.1 Teknik Fizibilite	6
3.1.1 Yazılım Fizibilitesi	6
3.1.2 Donanım Fizibilitesi	7
3.2 Zaman Fizibilitesi	7
3.3 Yasal Fizibilite	9
3.4 Ekonomik Fizibilite	9
4 Sistem Analizi	11
4.1 Sistemin Temel Bileşenleri	11
4.2 Kullanım Senaryosu	11
4.3 Kullanım Senaryosu Diyagramı	12
4.4 Veri Akış Diyagramı	12

5	Sistem Tasarımı	14
5.1	Yazılım Tasarımı	14
5.2	Veri Tabanı Tasarımı	15
5.3	Girdi-Çıktı Tasarımı	15
6	Uygulama	17
6.1	RestNet50 Modeli ile Özellik Çıkarma	17
6.1.1	RestNet50 Mimarisi	17
6.1.2	RestNet50'nin Kullanımı	17
6.2	Pickle Dosyalarının Oluşturulması	18
6.3	Görüntü Özelliklerini Çıkarma Fonksiyonunun Oluşturulması	18
6.4	Benzer Ürünlerin Önerilmesi	18
6.5	Kullanıcı Arayüzü Geliştirilmesi	18
6.5.1	Yükleme Sayfası	19
6.5.2	Sonuçlar Sayfası	19
7	Performans Analizi ve Sonuçlar	21
7.1	Model Seçimi ve Karşılaştırılması	21
7.1.1	ResNet50 Modeli Analizi	21
7.1.2	Öneri Geliştirme Algoritmaları Analizi	22
7.2	Sistem Başarısının Ölçümü	23
7.3	Sistem Başarısının Ölçümü	24
8	Sonuçlar	26
	Referanslar	27
	Özgeçmiş	28

KISALTMA LİSTESİ

ResNet	Residual Networks
API	Application Programming Interface
PIL	Python Imaging Library
REST	Representational State Transfer
KNN	K-En Yakın Komşular
LSTM	Long Short-Term Memory
NumPy	Numerical Python
Pandas	Python Data Analysis Library
SSD	Solid State Drive
Softmax	Soft Maximum
VGG	Visual Geometry Group

ŞEKİL LİSTESİ

Şekil 3.1	Gant Diyagramı	8
Şekil 4.1	Kullanım Senaryosu Diyagramı	12
Şekil 4.2	Veri Akış Diyagramı	13
Şekil 5.1	Ardışıl Diyagram	14
Şekil 5.2	Giriş Arayüz Tasarımı	15
Şekil 5.3	Çıktı Arayüz Tasarımı	16
Şekil 6.1	Yükleme Sayfası	19
Şekil 6.2	Yükleme Sayfası	20
Şekil 7.1	Veri Seti	23
Şekil 7.2	Başarı Ölçümü	24
Şekil 7.3	Algoritmaların Karşılaştırılması	24
Şekil 7.4	Her Bir Özellik İçin Algoritmaların Karşılaştırılması	25

TABLO LİSTESİ

Tablo 3.1	Diller Arasında Karşılaştırma	6
Tablo 3.2	Donanım Özellikleri	7
Tablo 7.1	Modellerin Karşılaştırılması	22
Tablo 7.2	Algoritmaların Test Sonuçları	24

KIYAFET ÖNERİ SİSTEMİ GELİŞTİRME PROJESİ

Doğa GÜNDOĞAR

Gülsüm İrem BAŞ

Bilgisayar Mühendisliği Bölümü

Bilgisayar Projesi

Danışman: Prof. Dr. Banu DİRİ

Online alışveriş günümüzde oldukça yaygın bir şekilde kullanılmaktadır. Sektörün büyümesiyle birlikte şirketler kullanıcıların ilgisini çekmek için diğer şirketlerle yarış haline girmiştir. Her bir firma, kullanıcıları kendi ürünlerinden almaya teşvik etmenin yollarını aramaktadır. Bu kapsamda öneri sistemleri büyük rol oynamaktadır. Kullanıcının ilgisini çeken ürünlerin benzerlerini göstererek onları sistemde tutma, zevkine göre seçtiği ürünlerin muadillerini sunarak daha fazla ürün almalarını sağlamak amaçlanmaktadır. Projemizde bu doğrultuda bir kıyafet öneri sistemi tasarlanmıştır. Potansiyel müşterinin yüklediği kıyafet resmine yakın benzerlikte ürünler müşteriye sunulurken beğenisini ve ilgisini kazanmak amaçlanmıştır. Sisteme yüklenen her türden resmin işlem görmesi için öncelikle görüntü işleme yöntemleri kullanılarak özellik çıkarımı yapılmıştır. Sonrasında farklı benzerlik metrikleri denenerek en uygun örneklerin sunulması sağlanmıştır.

Anahtar Kelimeler: Kıyafet Öneri Sistemi, Benzerlik Algoritmaları, Görüntü İşleme

ABSTRACT

FASHION RECOMMENDATION SYSTEM DEVELOPMENT PROJECT

Doğa GÜNDOĞAR
Gülsüm İrem BAŞ

Department of Computer Engineering
Computer Project

Advisor: Prof. Dr. Banu DİRİ

Online shopping has become very popular these days. As the industry grows, companies are competing with others for user attention. Every company is looking for ways to persuade users to buy their products. In this context, recommendation systems play an important role. Its purpose is to keep users in the system by showing them similar products that are of interest to them and to encourage them to buy more products by offering products that are similar to the ones they have selected according to their preferences. It is to get In our project, a fashion recommendation system was designed in this direction. The goal is to arouse the potential customer's appreciation and interest by presenting products that closely resemble the clothing image that the potential customer has uploaded. To process all kinds of images uploaded to the system, feature extraction was first performed using image processing techniques. A successful proposal presentation was then ensured by trying different similarity measures.

Keywords: Fashion Recommendation System, Similarity Algorithms, Image Processing

Günümüzde moda endüstrisi sürekli farklılaşmakta ve yeni ürünlerle birlikte sürekli büyümektedir. Bu büyüyen endüstriye kullanıcıların ayak uydurması, yeni çıkan ürünlere yetişmeleri zorlaşmaktadır. Bundan dolayı kullanıcılar doğru ürünlere ulaşmak için çok büyük bir vakit harcamak zorunda kalmaktadırlar. Aynı zamanda yeterli miktarda müşteri ile buluşamayan moda üreticileri ve satıcıları da bu durumdan olumsuz bir şekilde etkilenmektedir. Bu bağlamda, kıyafet öneri sistemleri, tüketicilere en uygun ürünleri önermek ve daha kişiselleştirilmiş hizmetler sunmak için büyük öneme sahiptir.

Gelişen teknoloji ile birlikte piyasada çoğu şirket kıyafet öneri sistemlerini kullanmaya başlamıştır. Örneğin Amazon şirketi müşterilerine daha kişiselleştirilmiş ürünleri sunmak için gelişmiş öneri algoritmaları kullanmaktadır. Bu sistemler derin öğrenme, görüntü işleme gibi teknolojileri kullanarak müşterilere daha fazla ürün keşfetme ve alışveriş deneyimlerini kişiselleştirerek aynı zamanda müşteri memnuniyetini arttırmayı amaçlamaktadırlar [1].

1.1 Proje Motivasyonu ve Kapsam

Kıyafet Öneri Sistemi projesi, potansiyel müşterilerin dikkatini çekmek ve onlara kişiselleştirilmiş öneriler sunarak alım oranlarını arttırmaktır. Projede gerçekleştiriminde kullanılan teknolojiler aşağıdaki gibidir:

Projede kullanıcı tarafından verilen görüntüden özellik çıkarma işlemi gerçekleştirilmektedir. Bu işlem için popüler olan ve karşılaştırmalarında en yüksek beğeniyi yakalamış olan resnet50 mimarisi kullanılmaktadır. ResNet-50, derin sinir ağları ve görüntü tanıma alanında önemli bir atılım olan Residual Networks (ResNet) mimarisine dayanan popüler bir modeldir. ResNet özellikle görüntü sınıflandırma ve nesne tanıma gibi görevlerde yüksek başarı sağlamıştır [2]. Resim özellikleri çıkarıldıktan sonra veri üzerinde KNN, Pearson katsayısı

ve kosinüs benzerliği algoritmaları çalıştırılmıştır. En yüksek başarı oranını elde eden algoritma öneri sunumunda kullanılmıştır. Son olarak da kullanıcının sistemi kolaylıkla çalıştırabilmesi için basit bir arayüzü tasarlamıştır.

1.2 Yapım Aşamasında Karşılaşılabilecek Zorluklar

- Veri Seti Hazırlığı ve Ön İşleme: Büyük veri setleriyle işlem yapmak zaman gerektirir. Verileri temizlemek ve eksik veya hatalı verileri düzeltmek gerekebilir. Ön işlem süreci, resimlerin boyutlarındaki farklılıklar nedeniyle zorlaşabilir.
- Özellik Çıkarma Süreci: Resimlerden özelliklerin çıkarılması, işlemci ve belleğin çok kullanıldığı bir süreçtir. Çok büyük bir veri seti üzerinde çalışılıyorsa, derin öğrenme kullanarak bu özellikleri çıkarmak ve pickle dosyaları olarak kaydetmek saatler sürer.
- Model Eğitimi: Derin öğrenme modelleri için uygun hiperparametrelerin seçilmesi ve modelin optimizasyonu uzun sürebilir. İşlem, yüksek işlem gücü gerektirdiği için yüksek performanslı bir donanıma da ihtiyaç duyar.
- Ölçüm Metriklerinin Seçimi: Uygun benzerlik algoritmasının seçilmesi, modelin performansı için çok önemlidir.
- Hafıza Yönetimi: Yüksek boyutlu veri setleri ile çalışmak performans sorunlarına neden olabilir.
- Kullanıcı Arayüzü ve Kullanılabilirlik: Basit ve kullanımı kolay bir kullanıcı arayüzü oluşturmak kullanıcı memnuniyeti için önemlidir.
- Sunucu Konfigürasyonları: Uygulamanın canlı ortamda çalışır duruma getirilmesi ve sunucu konfigürasyonları gibi teknik sorunlarla karşılaşılabılır.

1.3 Literatür Taraması

Moda ve ürün öneri sistemleri, kullanıcıların kişisel zevk ve isteklerine uygun, beğenebilecekleri ürünleri önermek için bazı Derin Öğrenme ve Bilgi Tabanlı Sistemler gibi sürekli olarak gelişmekte olan teknolojileri kullanmaktadır. E-ticaret platformlarının yükselişe geçmesiyle, insanların mağazaya gidip alışveriş yapma alışkanlıklarının azalmasıyla bu alana olan yatırım ve ilginin arttığı görülmektedir. Çevrimiçi alanda yarışmaya başlayan mağazalar ve E-ticaret platformları, kullanıcıların önceki satın alımlarını ve kişisel tercihlerini toplayarak daha kişiselleştirilmiş öneriler sunmaya başlamışlardır. Bu sayede kıyafet öneri

sistemlerinin ortaya çıkışı ve ivmelenmesi, bu bağlamda farklı farklı teknolojilerin kullanılmaya başlanması kısa bir sürede gerçekleşmiştir. Bu alanda yapılan çalışmalardan bir tanesi, 2016 yılında gerçekleştirilen konvolüsyonel sinir ağı kullanarak moda ürünlerini ve stillerini anlayabilen bir moda öneri sistemi geliştirilmesidir [3].

Bir başka geliştirilen çalışmada, moda ürünlerini anlayan sistem bir LTSM ve bir otokodlayıcı kullanarak gerçeklenmiştir [4]. 2018 yılında gerçekleştirilen bu çalışmadan önce 2017 yılında yine LTSM kullanarak moda öneri sistemi geliştirildi. Bu çalışmada var olan ürünlerle uyum yakalayan ürünleri bulmak üstüne ve kullanıcıların girdiği ürünlere veya açıklamalara benzeyen görünümeler elde edilmesi üstüne çalışılmıştır [5].

Bu gibi sistemlerin gelişmesini sağlayan çalışmalardan bir tanesi ise "VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback" başlıklı çalışmadır. Bu çalışma ile öneri sistemlerinde görsel bilgilerin daha doğru önerileri getireceği ve kullanıcıların kişisel tercihlerini daha iyi anlayacağı vurgulanmıştır [6].

Sonuç olarak, literatürden anlaşılacağı üzere kullanıcılara kişiselleştirilmiş, doğru ve kapsamlı öneriler sunabilmek için görsel bilgiler ve derin öğrenme algoritmaları önem kazanmıştır.

2 Ön İnceleme

Günümüzde sürekli değişen ve çeşitlenen moda endüstrisi ile beraber kullanıcıların istedikleri ürünlere daha kolay ulaşması için kıyafet öneri sistemlerine artan bir ihtiyaç bulunmaktadır. Bu projede kullanıcıların bu ihtiyaçlarına cevap vermek amacıyla seçilen ürün doğrultusunda farklı seçenekleri gösteren bir Kıyafet Öneri Sistemi oluşturulması amaçlanmıştır. Ön incelemeler doğrultusunda, derin öğrenme tabanlı çeşitli moda tavsiye sistemleri ve ilgili literatür incelenmiştir. "Image-based Recommendations on Styles and Substitutes" çalışmasında, orijinal görüntülerden özellikler Caffe derin öğrenme çerçevesi kullanılarak hesaplanmış ve ImageNet'teki 1.2 milyon görüntü üzerinde eğitim sağlanmıştır [7]. Bir diğer çalışma olan "DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations" ise "FashionNet" adı verilen, VGG-16 modeline dayanan bir ağı kullanmıştır [3]. "Learning Fashion Compatibility with Bidirectional LSTMs" adlı çalışmada, moda öğeleri verildiğinde, uyumluluk ilişkilerini öğrenmek için bir çift yönlü LSTM (Bi-LSTM) modeli kullanılmıştır [5]. Bu literatür taraması, projede kullanılması düşünülen yöntemler ve teknikler için bir temel oluştururken aynı zamanda moda öneri sistemlerinin mevcut durumunu ve potansiyel gelişim yollarını da göstermektedir.

2.1 Projeye Duyulan İhtiyaç

1. Kullanıcıların kişisel tercihlerine ve zevklerine göre kıyafet önerileri sunma
2. Kullanıcıların yaşadığı alışveriş deneyimini daha kullanışlı ve verimli hale getirme
3. Müşteri memnuniyetini arttırarak, markaların satışlarını ve müşteri güvenliğini arttırma

2.2 Proje Kapsamı

Bu projede yapılması amaçlanan moda öneri sistemi aşağıdaki bileşenlerden oluşmaktadır:

1. Ürün meta verilerinin toplanması ve işlenmesi: Ürünlerin özellikleri, görselleri ve markaları gibi meta verilerin toplanması ve düzenlenmesi
2. Öneri algoritması: Ürün meta verileri kullanarak, kullanıcılara en uygun kıyafet önerilerini sunan bir algoritma geliştirme
3. Kullanıcı arayüzü: Kullanıcıların önceki tercihine benzer ürünleri görebileceği basit bir arayüz tasarlama

2.3 Yol Haritası

1. Proje gereksinimlerinin ve kapsamının belirlenmesi
2. Kullanıcı profili ve ürün meta verisi için veri seti araştırılması ve düzenlenmesi
3. Öneri algoritması için gerekli makine öğrenimi ve veri analitiği tekniklerinin araştırılması ve seçilmesi
4. Kullanıcı arayüzü tasarlanması ve geliştirilmesi
5. Tüm bileşenlerin entegrasyonu ve test edilmesi
6. Sistem iyileştirmeleri

Bu kısım projenin yazılımsal, donanımsal, zamansal, yasal ve ekonomik durumunun gözden geçirildiği ve gerçekleştirme aşaması için gereksinimlerin belirtildiği yerdir.

3.1 Teknik Fizibilite

Bu bölümün amacı, projenin tasarımına, gereksinimlerine, kaynaklarına ve teknik uyumluluğuna odaklanarak projenin teknik fizibilitesini belirlemektir.

3.1.1 Yazılım Fizibilitesi

Bu bölümde, projemizin yazılım kısmı ve bunun projenin başarısını nasıl etkileyeceği incelendi. Uygulanabilir programlama dilleri, veri setleri, web çerçeveleri, API'ler ve sürüm kontrol araçları incelenmiştir.

Tablo 3.1, Python, R ve C# gibi olası programlama dillerinin karşılaştırmasını içerir. Her dilin kullanım kolaylığını, gerektirdiği programcı becerilerini, görüntü işleme ve makine öğrenmesi kütüphanelerini ve çalışma hızını değerlendirir.

Tablo 3.1 Diller Arasında Karşılaştırma

	Python	R	C#
Kullanım kolaylığı	Kolay	Kolay	Kolay
Programcı bilgisi	İyi	Orta	Az
Görüntü işleme kütüphaneleri	İyi	Az	Az
Makine öğrenmesi kütüphaneleri	Çok iyi	Çok iyi	Orta
Çalışma hızı	Orta	Orta	Hızlı

1. Programlama Dilleri ve Kütüphaneleri: Python, şekilde gösterilen fizibilite sonucunda projede kullanılacak ana programlama dili olacaktır. Python, popüler makine öğrenimi kütüphaneleri ve veri analizi araçlarıyla uyumlu bir dildir. Scikit-learn ve TensorFlow gibi makine öğrenmesi kütüphaneleri ve numpy,

pandas, matplotlib gibi veri analizi kütüphaneleri kullanılacaktır. Aynı zamanda görüntü işleme için PIL kütüphanesi vardır.

2. Veriseti: Bu projede önceden hesaplanmış özellik vektörlerini ve görüntü dosyalarını içeren pickle dosyaları kullanılacaktır.
3. Web Framework ve API: Kullanıcı arayüzünün geliştirilmesi ve öneri sistemi ile entegrasyonu için Python tabanlı bir web framework olan Flask kullanılacaktır. RESTful API'lar, istemci ve sunucu arasındaki veri iletişimini sağlamak için kullanılacaktır.
4. Sürüm Kontrolü ve İş birliği: Projenin geliştirilmesi sırasında, kodun kontrolü ve iş birliği için Kaggle kullanılacaktır. Bu, projede çalışan ekip üyelerinin kodu paylaşmasına, güncellemeleri takip etmesine ve geri almasına olanak tanır. Aynı zamanda bulunan veriseti ile direkt olarak işlem yapılmasına olanak sağlar.

3.1.2 Donanım Fizibilitesi

Proje için kullanılacak bilgisayarların donanım özellikleri Tablo 3.2'de verilmiştir.

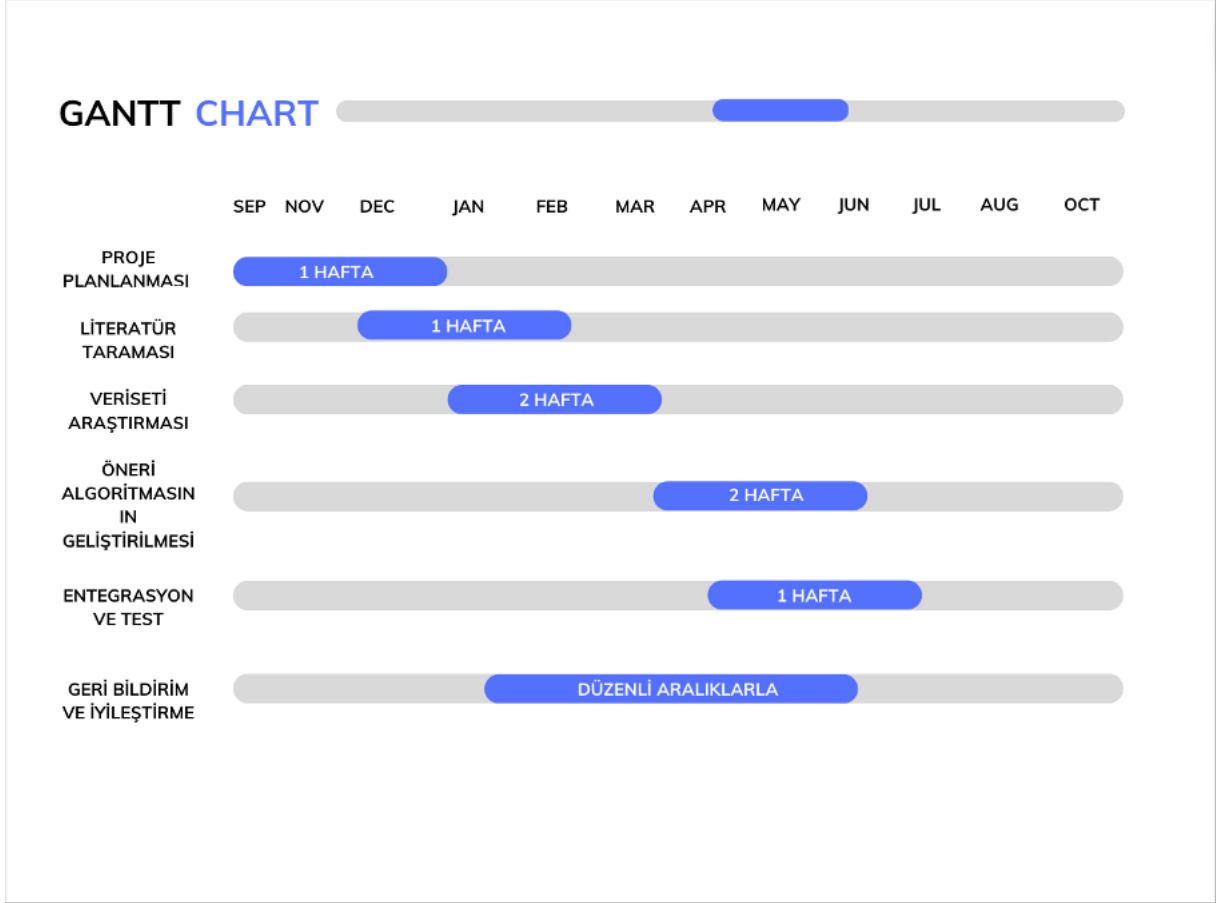
Tablo 3.2 Donanım Özellikleri

	1. Bilgisayar	2. Bilgisayar
RAM	16 GB	16 GB
Depolama	256 GB SSD	256 GB SSD
İşlemci	Apple M2	3,1 GHz Çift Çekirdekli Intel Core i5
İşletim Sistemi	macOS Ventura	macOS Big Sur

1. Veri Depolama: Ürünlerin görsellerini içeren veriseti için depolama birimlerine ihtiyaç duyulacaktır. Dosya boyutu 658,5 MB'dır.
2. Hesaplama Gücü: Öneri algoritmasının eğitimi ve uygulanması, yeterli işlemci gücü ve belleğe sahip bir sunucunun gerekli olduğundan emin olmak için gereklidir. Makine öğrenimi modellerinin eğitimi, GPU destekli sunucular tarafından daha hızlı ve verimli olacaktır.

3.2 Zaman Fizibilitesi

Sürecin adımlarını ve zaman aralıklarını gösteren gantt çizelgesi Şekil 3.1'de gösterilmiştir.



Şekil 3.1 Gant Diyagramı

1. Proje Planlaması ve Araştırma (1 hafta): Projenin gereksinimleri, kapsamı ve hedeflerinin belirlenmesi; öneri algoritması için gerekli makine öğrenimi ve veri analitiği tekniklerinin araştırılması aynı zamanda görüntü işleme ile özellik çıkarımı konusunda bilgi edinimi.
2. Literatür Taraması(1 hafta): Benzer projelerin makale ve içeriklerinin incelenmesi, yöntemlerin araştırılması.
3. Öneri Algoritmasının Geliştirilmesi (2 hafta): Kullanıcı profilleri ve ürün meta verileri kullanarak, kullanıcılara en uygun moda önerilerini sunan bir algoritma geliştirme ve test etme süreci.
4. Kullanıcı Arayüzü Geliştirme (2 hafta): Kullanıcı dostu bir arayüz tasarlanması ve geliştirilmesi.
5. Entegrasyon ve Test (1 hafta): Tüm bileşenlerin entegrasyonu ve test edilmesi.
6. Geri Bildirim ve İyileştirme (Sürekli): Kullanıcılar tarafından sağlanan geri bildirimler doğrultusunda sistem iyileştirmelerinin gerçekleştirilmesi.

Toplam süre: Yaklaşık 7-8 hafta (süre kaynaklara bağlı olarak değişebilir)

3.3 Yasal Fizibilite

1. Veri Gizliliği ve Kullanıcı Bilgilerinin Korunması: Proje içerisinde kullanıcıların bazı kişisel verileri kullanılıyor ise, gizlilik ve veri koruma yasaları doğrultusunda kullanıcılar verilerin kullanımı ve saklanması hakkında bilgilendirilmeli ve onay alınması gerekmektedir.
2. Fikri Mülkiyet Hakları: Ürün görselleri, açıklamaları ve markaları gibi meta verilerin kullanımı sırasında, fikri mülkiyet haklarına dikkat edilmelidir. İlgili içeriklerin kullanımı için gerekli izinlerin alınması ve kaynaklara atıfta bulunulması önemlidir.
3. Yasa Dışı İçerik ve Kısıtlamalar: Öneri sistemi, yasa dışı veya kısıtlanmış içerikleri (ör. kopyalanan tasarımlar, yasa dışı malzemeler) önermemelidir. Bu tür içeriklerin filtrelenmesi ve sistemden çıkarılması önemlidir.

3.4 Ekonomik Fizibilite

Kıyafet öneri sistemi için öngörülen maliyetler aşağıda verilmiştir:

1. Donanım ve Sunucu Maliyetleri: Veri depolama ve hesaplama gücü için gerekli sunucuların ve donanımın maliyeti. Bulut tabanlı hizmetler kullanılması durumunda, bu maliyetler abonelik ücretleri şeklinde çıkabilir. Fakat kullanılması düşünülmediğinden maliyeti yoktur.
2. Yazılım Lisansları ve API Kullanım Ücretleri: Bu proje doğrultusunda kullanılan yazılımlar, kütüphaneler ve uygulamalar ücretsiz olduğu için maliyet olarak sadece işletim sistemi ücreti bulunmaktadır.
3. Pazarlama ve Tanıtım: Kıyafet öneri sisteminin tanıtılması ve kullanıcıların dikkatini çekmek için yapılan pazarlama ve tanıtım faaliyetlerinin maliyeti henüz böyle bir aşamadan bahsedilmediği için belirlenmemiştir.

4

Sistem Analizi

Bu bölüm, projenin başarılı bir şekilde tamamlanmasını sağlamak için sistemin gerekli olan özelliklerini ve tasarımını ayrıntılı bir şekilde açıklamaktadır. Bu aşamada belirlenen gereksinimler, mevcut sistemleri analiz ederek projenin genel iş süreçlerini daha verimli hale getirecektir.

4.1 Sistemin Temel Bileşenleri

- Donanım Kaynakları
 - Sistem: MacOS İşletim Sistemi
 - Gerekli RAM: 16GB
 - Gerekli Disk: 256MB
- Yazılım Kaynakları
 - Kaggle
 - Spyder
- İnsan Kaynakları
 - Gülsüm İrem BAŞ | Sistem analisti & Programcı
 - Doğa GÜNDOĞAR | Sistem tasarımcısı & Programcı
 - Kullanıcı

4.2 Kullanım Senaryosu

Birincil Aktör: Kullanıcı

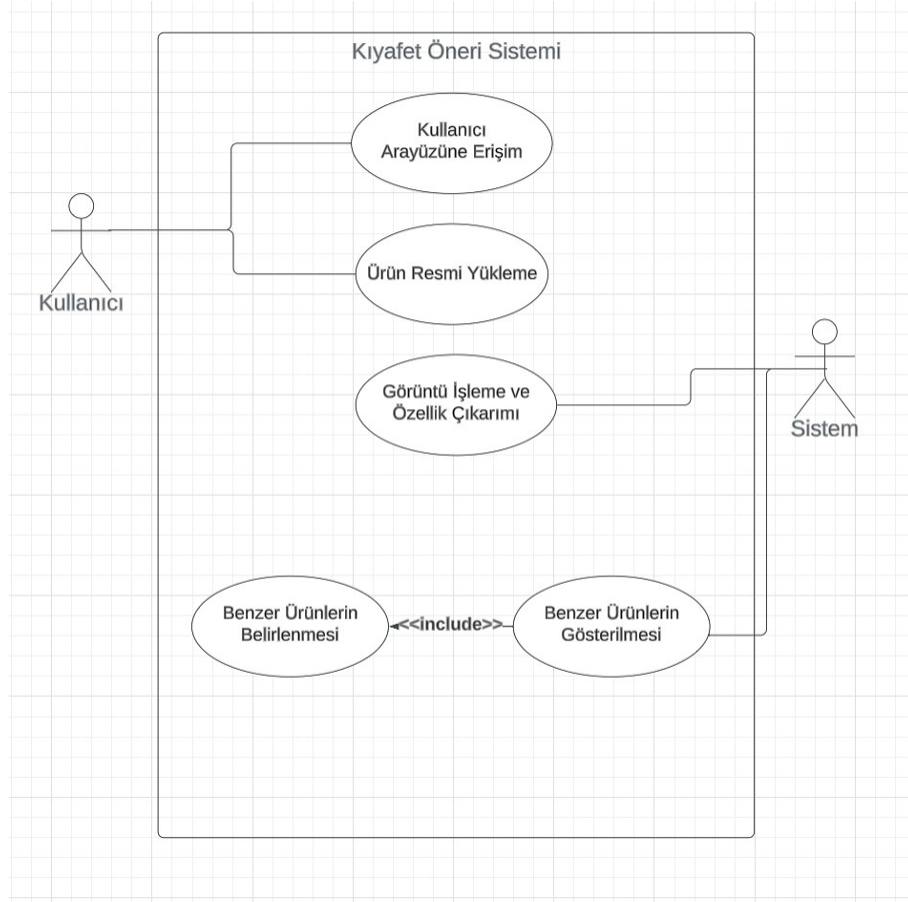
İlgililer ve İlgi Alanları: Kullanıcı sisteme yüklediği kıyafete göre yeni öneriler almak ister.

Ana Senaryo:

1. Kullanıcı web sitesine girer
2. Kullanıcı sisteme kıyafet resmi yükler
3. Sistem kullanıcıya benzer kıyafet önerileri sunar

4.3 Kullanım Senaryosu Diyagramı

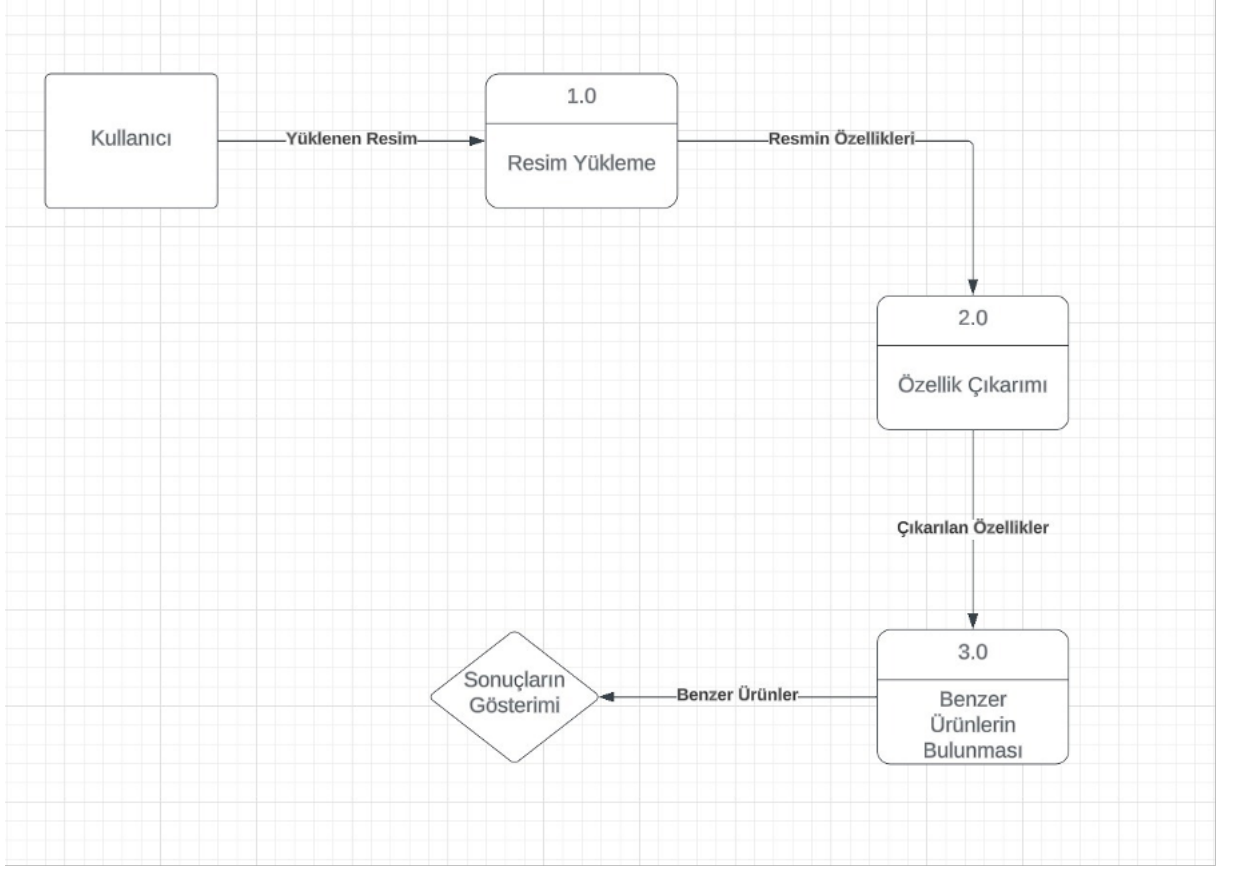
Önceki bölümde tasvir edilen kullanım senaryosunun diyagramı Şekil 4.1’de verilmiştir.



Şekil 4.1 Kullanım Senaryosu Diyagramı

4.4 Veri Akış Diyagramı

Şekil 4.2’de projenin veri akış diyagramı gösterilmiştir.



Şekil 4.2 Veri Akış Diyagramı

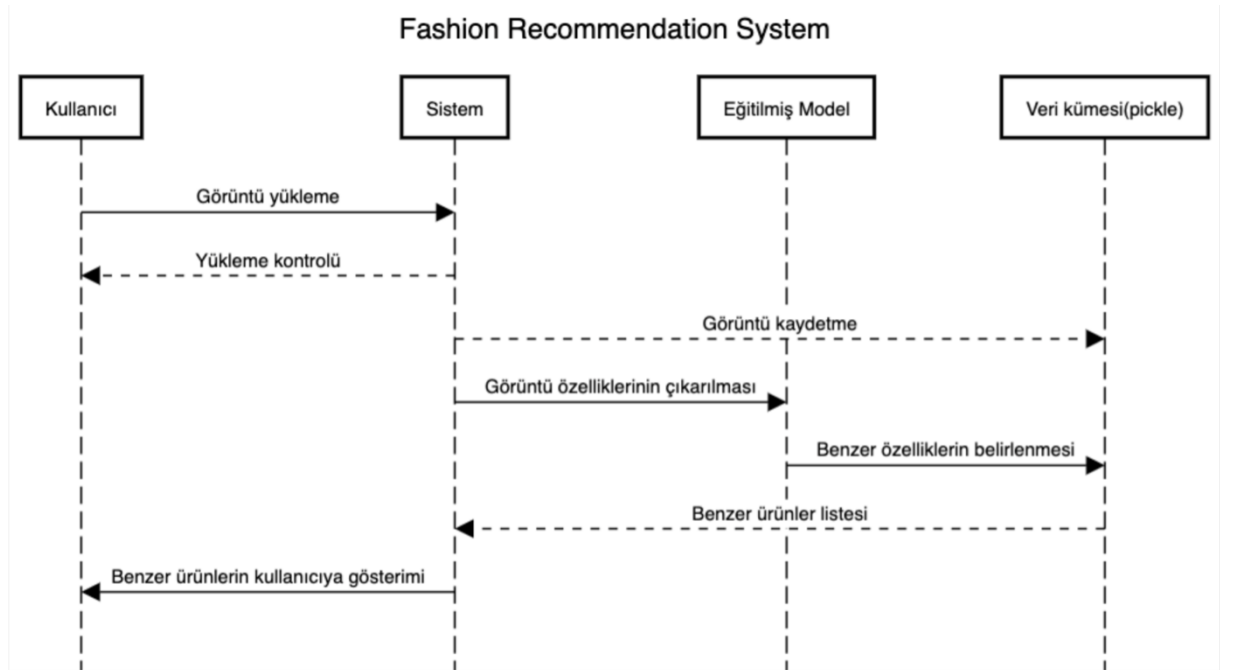
5

Sistem Tasarımı

Projenin teknik yapısı ve işleyişi bu bölümde kapsamlı bir şekilde ele alınmaktadır. Bu aşamada, gereksinim analizi sürecinden elde edilen veriler kullanılarak sistem bileşenleri, veri yapıları, iş süreçleri ve kullanıcı arayüzleri gibi unsurlar tasarlanır ve düzenlenir. Projenin uygulanabilirliği ve başarısı, sistem tasarımı sürecinden doğrudan etkilenir.

5.1 Yazılım Tasarımı

Kıyafet Öneri Sistemi Projesinin ardışıl diyagramı ve açıklaması Şekil 5.1’de sunulmuştur.



Şekil 5.1 Ardışıl Diyagram

1. Kullanıcı, bir moda ürününün görüntüsünü sisteme yükler.

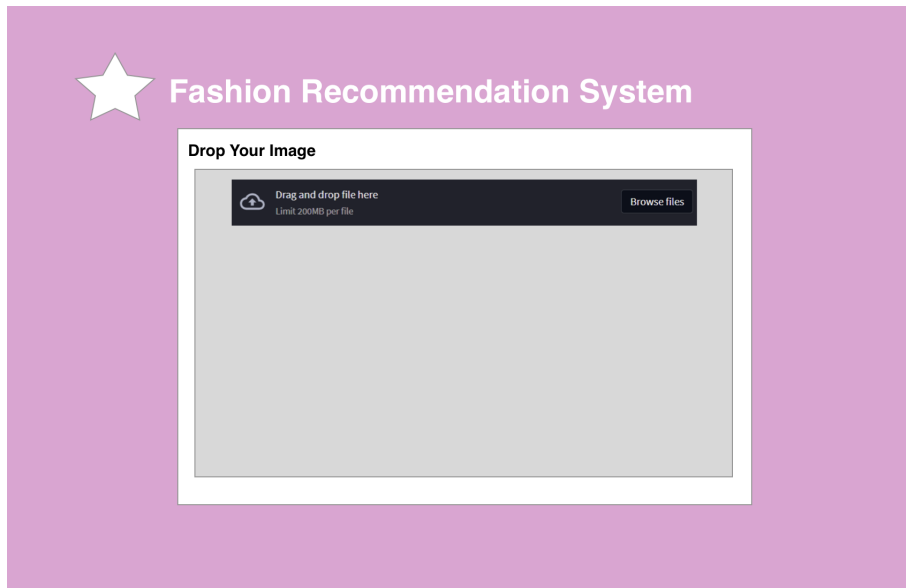
2. Sistem, görüntünün başarıyla yüklendiğini kullanıcıya bildirir ve görüntüyü veri kümesine (pickle dosyalarında) saklamak için işleme alır.
3. Eğitilmiş Model, verilen görüntünün özelliklerini çıkarır ve benzer özelliklere sahip diğer ürünleri belirlemek için veri kümesine (pickle dosyalarında) başvurur.
4. Veri kümesi (pickle dosyaları), belirlenen benzer ürünlerin listesini sisteme geri gönderir.
5. Sistem, benzer moda ürünlerinin listesini kullanıcıya gösterir.

5.2 Veri Tabanı Tasarımı

Her seferinde alınan güncel veri setleri, veri işleme ve analiz için kullanılır. Sonuç olarak, daha önce yapılan işlemlerde kullanılan veri setlerinin kalıcı bir veri tabanında saklanması gerekmez. Bu yöntem, sistem kaynaklarının daha verimli kullanılmasını sağlarken aynı zamanda veri saklama ve güvenlik sorunlarının ortadan kaldırılmasını da sağlar.

5.3 Girdi-Çıktı Tasarımı

Uygulamanın giriş ekranı Şekil 5.2 olarak tasarlanmıştır. Tasarımda sayfanın üstünde logo ve sistem adının yer aldığı yazı yer almaktadır. Altında kullanıcıya "Ürününüzü Bırakın" talimatıyla resim yükleyeceği yer gösterilir. Kullanıcı gerekli noktaya basarak resimini sisteme yükleyebilecektir.



Şekil 5.2 Giriş Arayüz Tasarımı

Uygulamanın çıktı ekranı Şekil 5.3 olarak tasarlanmıştır. Tasarımda sayfanın sol üstünde logo ve sistem adı yer almaktadır. Orta üstte kullanıcının yüklediği resme yer verilmiştir. Altında ise "Bunları Da Beğenebilirsiniz" yazısıyla birlikte dört adet öneri yan yana sıralanmıştır.



Şekil 5.3 Çıktı Arayüz Tasarımı

Bu projede, kıyafet öneri sistemi oluşturmak için görüntü işleme ve derin öğrenme teknikleri kullanılmıştır. Önceden eğitilmiş ResNet50 modelini kullanarak ürün görüntülerinden özellik vektörleri çıkarılmış ve bu özellik vektörleri ile benzer ürün önerileri sunulması amaçlanmıştır. Uygulamanın arayüzü Python Flask kütüphanesi kullanılarak gerçekleştirilmiştir. Aşağıda projenin uygulama kısımları sırasıyla gösterilmektedir.

6.1 RestNet50 Modeli ile Özellik Çıkarma

Residual Networks ailesinin bir parçası olan RestNet50, büyük ölçekli görüntü sınıflandırma problemlerini çözmek için yaygın olarak kullanılan bir derin öğrenme modelidir. RestNet50, ImageNet veri setindeki milyonlarca görüntü üzerinde önceden eğitilmiştir ve özellikle transfer öğrenme sürecinde kullanılır. Genellikle yüksek performans gösterir.

6.1.1 RestNet50 Mimarisi

RestNet50, toplam 50 katmandan oluşmaktadır. Model, çeşitli evrişimli katmanlardan oluşan residual blokları içerir. Mimariyi küçültmek için iki aşamalı bir örnekleme süreci kullanılır ve küresel bir ortalama havuz katmanı kullanılır. Aktivasyon fonksiyonu olarak Softmax kullanılır. RestNet mimarisi, konvolüsyonel, aktivasyon, pooling ve fully-connected katmanlardan oluşur [8].

6.1.2 RestNet50'nin Kullanımı

RestNet50, görüntü verilerinden özellikleri çıkarmak için aktarım öğreniminde kullanılır. Önceden eğitilmiş modeller zamandan ve kaynaklardan tasarruf sağlar. Bu kapsamda projede kullanılmak üzere RestNet50 tercih edilmiştir.

İlk olarak `'include_top = False'` seçeneği ile tam bağlantılı katmanların

çıkarılması sağlanır ve modelin sadece evrişimli katmanları kullanılır. Ardından GlobalAveragePooling2D katmanı eklenerek model düzenlenir.

6.2 Pickle Dosyalarının Oluşturulması

Pickle, Python için veri serileştirme ve ters serileştirme modülüdür. Bir nesnenin durumunu bayt akışına dönüştürme süreci serileştirme olarak bilinir. Bu bayt akışı, aynı program veya başka bir program tarafından orijinal nesneye dönüştürülebilmektedir. Pickle, bir Python nesnesinin iç yapısını diske kaydederek "dondurur". Bu nesne daha sonrasında aynı durumda geri yüklenerek kullanılabilir. *'image_features_embedding.pkl'* ve *'img_files.pkl'* adlı pickle dosyalarına, sırasıyla görüntü özellik vektörleri ve görüntü dosya yolları kaydedilir. Bu dosyalar, daha sonrasında öneri sistemi için kullanılmıştır. Bu sayede uzun süre işlem gerektiren model eğitimi aşaması sürekli olarak tekrarlanmamıştır.

6.3 Görüntü Özelliklerini Çıkarma Fonksiyonunun Oluşturulması

'extract_img_features' fonksiyonu, verilen bir görüntü dosya yolunu ve modeli alarak, görüntünün özelliklerini çıkarmak için kullanılır. Görüntüyü öncelikle ResNet50 modeline uygun hale getirir ve modelin öngörülen özelliklerini elde eder. Son olarak, özelliklerin normalize edilmiş halini döndürür.

6.4 Benzer Ürünlerin Önerilmesi

Görüntü üzerinde özellik çıkarımı sonrasında sistemin yüklenen görüntüye benzer olan ürünleri kullanıcıya gösterme kısmı için benzerlik algoritmaları denenmiş ve test edilmiştir. Bu kapsamda denenilen algoritmalar Kosinüs benzerliği, Pearson benzerliği, KNN algoritmalarıdır. Kullanılan algoritmalar arasında en başarılı sonuç veren benzerlik algoritması Pearson algoritmasıdır. Bunun sonucunda öneri sistemi geliştirilmesinde Pearson benzerliğinin ölçüt alınmasına karar verilmiştir.

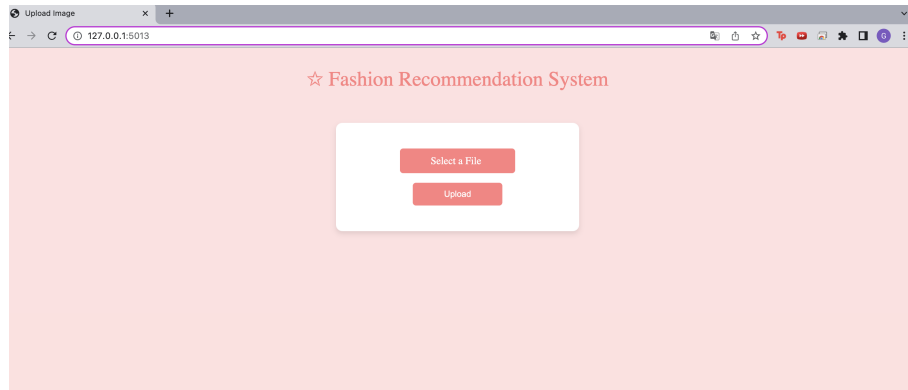
6.5 Kullanıcı Arayüzü Geliştirilmesi

Bir Python web çerçevesi olan Flask, bu projede dinamik bir web uygulaması oluşturmak için kullanılmıştır. Flask, hızlı ve basit bir şekilde işlevsel bir web arayüzü oluşturmak için gereken esnekliği ve basitliği sağlamıştır. Uygulama için tasarlanan arayüz iki ana sayfadan oluşmaktadır. Kullanıcının karşısında çıkan ilk ekran resim yükleyebileceği ekrandır. Burada kullanıcı benzer kıyafet önerilerini

almak istediği kıyafetin resmini sisteme yükler. Yükleme yapıldıktan sonra sistem sonuçların gösterildiği sayfaya geçiş yapar. Sonuç sayfasında ise kullanıcının yüklediği resme en benzer 5 resim özellikleriyle birlikte ekranda gösterilir.

6.5.1 Yükleme Sayfası

Kullanıcın bilgisayarından bir ürün resmi seçerek yüklemesini sağlayan sayfa ('uploads.html'), yükleme sayfasıdır. Flask kütüphanesinin "request" modülü, yüklenen dosyanın kabul edilebilir bir format olup olmadığını (yani 'jpg', 'jpeg' veya 'png') kontrol eder ve dosyanın sunucuya başarıyla yüklendiğini doğrular. Aşağıda yükleme sayfasının bir görüntüsü verilmiştir.

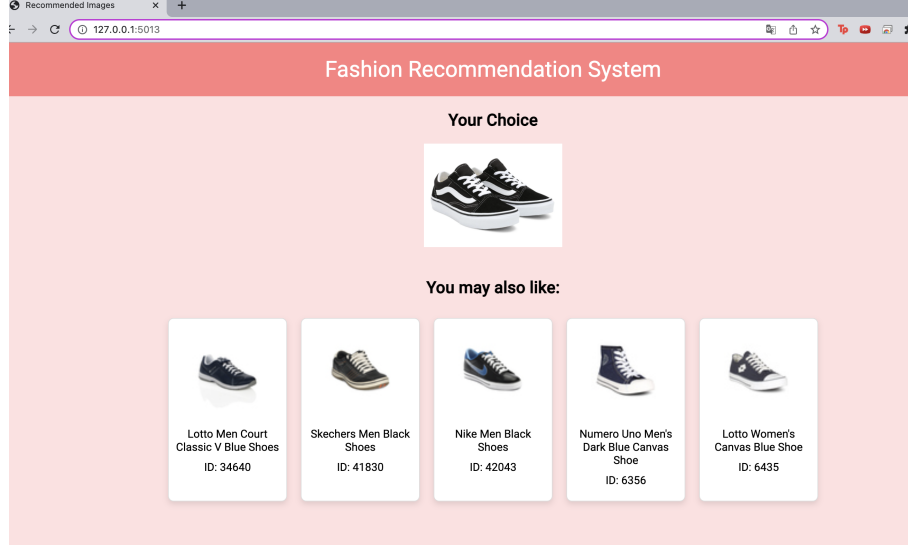


Şekil 6.1 Yükleme Sayfası

Şekil 6.1'de Görülen butonlardan 'Select a File' butonu sisteme görüntü yüklenmesini sağlarken, 'Upload' butonu ile sonuçların gösterildiği sayfaya geçilmektedir.

6.5.2 Sonuçlar Sayfası

Bir önceki aşamada yükleme işleminin başarılı gerçekleşmesi durumunda, ResNet50 modeli aracılığıyla yüklenen ürünün özellikleri çıkarılarak, Pearson benzerlik algoritması ile yüklenen ürüne benzer 5 ürün sonuçlar sayfası ('results.html') üzerinden gösterilmektedir. Her önerinin yanında ürünün markası modeli ve id numarası gibi ek bilgiler de yer almaktadır. Kullanıcı bu sayede beğendiği benzer ürüne kolay bir şekilde ulaşabilmektedir.



Şekil 6.2 Yükleme Sayfası

Şekil 6.2’de Görüldüğü üzere verilen ayakkabı resmine en çok benzeyen 5 ürünün resim, marka ve modelleri listelenmiştir.

Projenin geliştirme aşamasında seçilen algoritmaların analizi bu bölümde yapılmıştır ve sonuçlar ayrıntılı olarak ele alınmıştır. Daha sonra uygulamanın başarısını değerlendirmek için kullanılan yöntem ve test işlemleri gösterilmiştir.

7.1 Model Seçimi ve Karşılaştırılması

Bu projede görüntü işleme ile özellik çıkarımı için ResNet50 modeli kullanılmıştır. Ayrıca uygulamanın Öneri geliştirme kısmı için üç farklı algoritma test verisi üstünde denenerek en yüksek başarı oranına sahip olan algoritma sistem önerisi için kullanılmıştır. Aşağıda ResNet50 seçilme analizi ve sistem önerisi için kullanılan algoritmaların detaylı analizi verilmiştir.

7.1.1 ResNet50 Modeli Analizi

Bu çalışmada kullanılan temel model ResNet50 olarak seçilmiştir. ResNet50, önceden eğitilmiş bir görüntü sınıflandırma modeli olup, derinliği (50 katman) ve artık bağlantılarla (residual connections) ayırt edilir. Bu modelin seçilmesindeki temel faktörler, ImageNet veri kümesi üzerinde eğitilmiş olması ve 1000 farklı nesne sınıfını tanıyabilme yeteneğidir [9]. Bu özellikler, derin öğrenme modellerinde sıklıkla karşılaşılan gradyan kaybolma problemine bir çözüm sağlar. ResNet50, önceden eğitilmiş ağırlıklarla birlikte sıklıkla kullanılmaktadır çünkü daha az veri ile bile iyi performans gösterebilmektedir. Bu özellik, özellikle verinin sınırlı olduğu durumlarda çok faydalı olmuştur [10]. ResNet50 seçilmeden önce karşılaştırılabilir diğer modeller de incelenmiştir. Bu modeller Xception, InceptionV3, VGG16 ve VGG19'dur. Bununla birlikte, ResNet50'nin tercih edilmesinin ana nedeni, artık bağlantılar ve derinliğin sunduğu avantajlar olmaktadır [11]. Farklı modellerin seçilmeme nedenleri kısaca açıklanacak olursa, VGG16 ve VGG19 modelleri, ResNet50 ile karşılaştırıldığında daha az doğruluk oranlarına sahiptir ayrıca daha fazla parametre gerektirmektedirler. InceptionV3 ve Xception modelleri ise daha yüksek doğruluk oranlarına sahip

olmalarına rağmen daha karmaşık yapıda ve daha yüksek hesaplama maliyetlerine sahiptirler. Aşağıda modellerin karşılaştırmasını içeren bir tablo verilmiştir. Bu bilgiler doğrultusunda bu proje için en uygun görüntü işleme modeli olarak ResNet50 seçilmiştir [12]. Tablo 7.1’de modellerin karşılaştırmaları anlaşılır bir şekilde verilmiştir.

Tablo 7.1 Modellerin Karşılaştırılması

Model Adı	Derinlik	Parametre Sayısı	En Yüksek-1 Doğruluk	En Yüksek-5 Doğruluk
VGG16	16	138 Milyon	%71.5	%90.1
VGG19	19	144 Milyon	%72.7	%91.2
InceptionV3	48	23.8 Milyon	%78.8	%94.4
Xception	71	22.9 Milyon	%79.0	%94.5
ResNet50	50	25.6 Milyon	%76.2	%92.9

7.1.2 Öneri Geliştirme Algoritmaları Analizi

Öneri sistemlerinde, kullanıcıların amaçları doğrultusunda onlara ürün veya hizmet önermeyi hedefleyen algoritmalar kullanılır. Kıyafet Öneri Sistemi kullanıcılara önceki tercihlerine göre benzer fakat farklı ürün önerileri yapmaktadır. Bu kapsamda projeye en yüksek başarıyı getirebilecek çeşitli öneri algoritmaları incelenmiştir. Bu çalışmada K-En Yakın Komşular (KNN), Pearson Korelasyon Katsayısı ve Kosinüs Benzerliği algoritmaları incelenmiştir. Bu bölüm, bu algoritmaların tanımlanmasını ve performanslarını karşılaştırmayı ele almaktadır.

- **K-Nearest Neighbors** - K-En Yakın Komşular (KNN) algoritması, bir ürünü değerlendiren kullanıcıların ne kadar benzer olduğunu belirlemek için kullanılır. KNN algoritması, bir kullanıcının bir öğeyi nasıl değerlendireceğini tahmin etmek için diğer kullanıcıların değerlendirmelerini inceler. Tipik olarak, benzerlik kullanıcıların önceki tercihlerine dayanmaktadır. KNN, özellikle çok sayıda öğe ve kullanıcının olduğu büyük veri setlerinde etkilidir. KNN algoritmasının denklemi 7.1’de verilmiştir.

$$P(A|B) = \frac{I(A,B)}{n(B)} \quad (7.1)$$

- **Pearson Korelasyon Katsayısı** - Pearson Korelasyon Katsayısı, iki değişken arasındaki doğrusal ilişkiyi bulmak için kullanılır. Kullanıcıların bir öğeye verdikleri puanlar arasındaki ilişkiyi belirlemek için öneri sistemlerinde sıklıkla kullanılır. İki kullanıcının da aynı öğeleri değerlendirme alışkanlıkları benzer ise, aralarında yüksek bir Pearson Korelasyon Katsayısı olacaktır. Pearson Korelasyon Katsayısı, iki kişinin çok benzer olduğu durumlar dışında sıklıkla düşük sonuçlar verir. Denklemi 7.2’de verilmiştir.

$$r = \frac{\sum (x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum (x_i - \mu_x)^2 \sum (y_i - \mu_y)^2}} \quad (7.2)$$

- **Kosinüs Benzerliği** - Kosinüs Benzerliği, iki kişinin bir öğeye verdikleri puanlar arasındaki açıyı ölçen bir ölçümdür. Bu metrik, iki kullanıcının bir öğeyi değerlendirme biçimlerinin çok benzer olduğu durumlarda yüksek bir değer verir. Kosinüs Benzerliği, özellikle büyük veri setlerinde hızlı ve etkili bir şekilde benzerlik hesaplaması yapabilir. Denklemi 7.3'de verilmiştir.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (7.3)$$

7.2 Sistem Başarısının Ölçümü

Sistemin başarısını ölçmek için projede kullanılan veri setindeki bazı özelliklerden yararlanılmaktadır. Sistem başarısını ölçerken kullanılan bazı metrikler önerilen öğelerin doğruluğunu ve verilen ürün ile olan benzerliği üstünden bir ölçüm yapmaktadır. Şekil 7.1'de veri setinde kullanılan özellikler verilmiştir.

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName
24922	12532	Women	Apparel	Topwear	Jackets	Black	Fall	2011.0	Casual	Puma Women Solid Black Jackets
35553	3469	Unisex	Accessories	Headwear	Headband	Black	Fall	2010.0	Sports	ADIDAS Unisex Black White Headband
31700	24742	Women	Accessories	Bags	Handbags	Green	Winter	2015.0	Casual	Murcia Women Casual Handbag
42694	28477	Women	Apparel	Topwear	Kurtas	Purple	Summer	2012.0	Ethnic	Urban Yoga Women Printed Purple Kurta

Şekil 7.1 Veri Seti

Yukarıda verilen özelliklere göre kullanıcının verdiği ürünün önerilen ürünlerle olan benzerliği, özelliklerinin ne kadarının aynı olduğu üstünden ölçülmüştür. Veri seti eğitim ve test olarak ikiye ayrıldıktan sonra, 'cosine', 'pearson' ve 'knn' benzerlik algoritmaları kullanılarak sırasıyla sistem gerçekleştirilmiştir. Ayrılan test verisindeki her bir resim, kullanıcı ürünü olarak sisteme sokulmuş ve öneriler gözlemlenmiştir. Her bir girdinin veri setindeki özelliklerinin, önerilen ürünlerin özelliklerinin ne kadarıyla eşleştiğinin bilgisi tutulmuş ve tüm test verisi için hesaplanarak ortalaması alınmıştır. Başarı oranı, önerilen öğenin belirli özelliklerinin (örneğin, 'gender', 'masterCategory', 'subCategory', 'articleType', 'baseColour', 'season', 'year', 'usage') kullanıcı tarafından tercih edilen özelliklere ne kadar yakın olduğunu belirlemektedir. Her bir özellik için ayrı bir başarı oranı hesaplanır ve bunların ortalaması genel başarı oranını verir. Şekil 7.2'de bu işlemin uygulandığı kod bloğu verilmiştir.

```
def compute_success_rates(recommended_items, input_img_path, styles_df, features_to_check, model):
    input_img_features = styles_df[styles_df['image'] == input_img_path][features_to_check].values[0]
    counters = [0] * len(features_to_check)

    for item in recommended_items:
        item_image_path = "/Users/gulsumirembas/Downloads/project/" + item['image_path']
        item_features = styles_df[styles_df['image'] == item_image_path][features_to_check].values[0]
        for i in range(len(features_to_check)):
            if item_features[i] == input_img_features[i]:
                counters[i] += 1

    success_rates = [count / len(recommended_items) for count in counters]
    return success_rates

def compute_overall_success_rate(success_rates):
    return sum(success_rates) / len(success_rates)
```

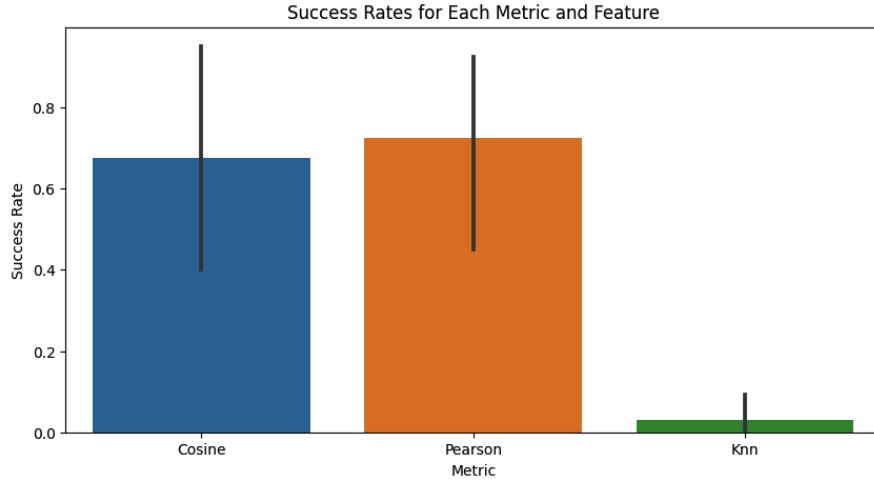
Şekil 7.2 Başarı Ölçümü

7.3 Sistem Başarısının Ölçümü

Sistemin başarısı için kullanılan algoritma 7.2'nci kısımda anlatılmıştır. Bu kapsamda K-En Yakın Komşu, Pearson Korelasyon ve Cosine Benzerliği algoritmaları test dataseti üzerinde denenmiştir. Sonuçlara dair grafikler aşağıda verilmiştir.

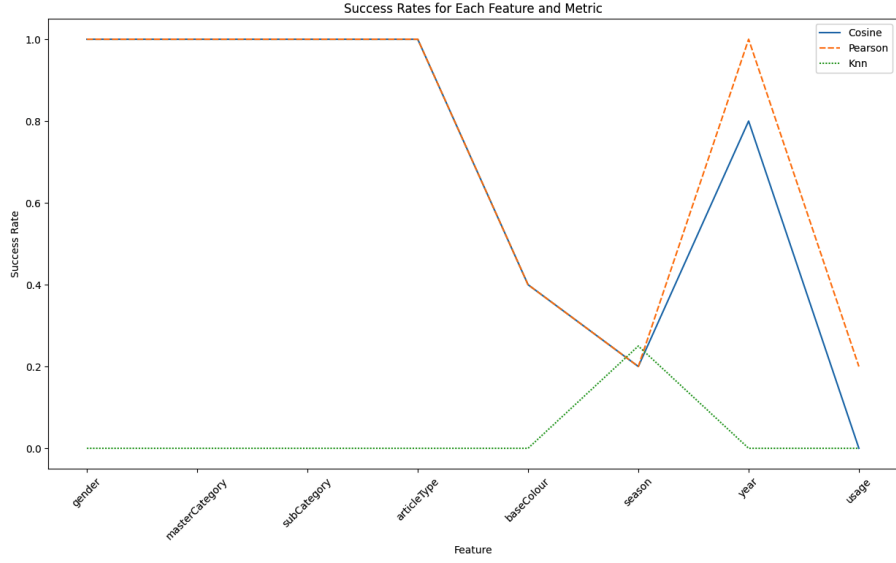
Tablo 7.2 Algoritmaların Test Sonuçları

Algoritma	Değer
Cosine	0.675
Pearson	0.725
Knn	0.03125



Şekil 7.3 Algoritmaların Karşılaştırılması

Şekil 7.3 'de Algoritmaların aynı test verisinde genel başarı oranları ölçülmüştür. Genel başarı oranlarına bakıldığında Pearson Öneri metriğinin Cosine ve KNN metriklerine göre daha yüksek bir başarı oranı sergilediği görülmektedir. Şekil 7.2'den de görüleceği üzere, Pearson başarı oranı için 0,725 gibi yüksek bir sonuç elde etmiştir. Bu sonuçlar doğrultusunda Pearson öneri metriğinin diğer algoritmalarla göre daha tutarlı olduğu ve kullanıcının verdiği ürüne daha benzer ürünler önerdiği görülmüştür.



Şekil 7.4 Her Bir Özellik İçin Algoritmaların Karşılaştırılması

Şekil 7.4'e bakıldığında, her bir özellik için Pearson ve Kosinüs algoritmalarının genellikle benzer ürünleri önerdiği fakat KNN'in çoğu özellikte yüksek doğruluk oranı yakalayamadığı gözlemlenmiştir. Pearson algoritması ise tüm özellikler için ortalamanın üzerinde bir başarı oranı sergilemiştir. Bu algoritmanın veri setindeki çeşitli özelliklerle iyi uyum sağladığı, daha dengeli ve geniş kapsamlı öneriler yaptığı anlaşılmaktadır. Bu nedenlerden dolayı, Pearson Benzerliği algoritması bu öneri sistemi için en iyi seçenek olarak belirlenmiştir.

8 Sonular

Bu projede, moda rnlerine iliřkin fotoęraflardan zellik ıkarmak ve benzer rnler nermek iin derin ęrenme tabanlı bir sistem geliřtirildi. Sistem, eřitli lm metriklerine dayalı olarak resimlerden zellikler ıkarmak ve ResNet50 adlı derin ęrenme modelini kullanarak benzer rnler nermektedir.

Proje srecinde veri toplama, n iřleme, model eęitimi, zellik ıkarma ve neri yapma ařamaları gerekleřtirildi. Proje sresince yksek boyutlu verilerle alıřma ve bu verilerle zellik ıkarma gereklilięi yapım ařamasında zorluklara yol amıřtır. Aynı zamanda bařarılı olunması iin performansı deęerlendirmesi iin doęru metriklerin seilmesi ve uygulanması da nemlidir.

Geliřtirilen sistem, moda endstrisi iin olduka faydalı olabilir. Markaların, rnlerini mřterilerin ilgi alanlarına ve zevklerine gre uyarlayarak alıřveriř deneyimini zelleřtirmelerine yardımcı olabilir. Bununla birlikte, sistemin daha verimli ve etkili hale getirilmesi iin neri motorunun gerek verilerle test edilmesi ve kullanıcı yorumlarının alınması gerekir.

Daha karmařık derin ęrenme modellerinin kullanılması, farklı lm metriklerinin denenmesi, lklenebilirlik ve hafıza ynetimi konularında iyileřtirmeler yapılması, ileride yapılacak alıřmalarda ele alınması nerilen konulardır. Bunun yanında arayzn daha interaktif olması iin alıřmalar yapılması da deęerlendirilebilir.

Son olarak, bu proje derin ęrenme ve makine ęreniminin moda sektrnde nasıl kullanılabileceęini gsteren bir rnektir. Teknolojinin srekli geliřmesiyle birlikte, modellemelerin ve algoritmaların daha da geliřtirilmesi endstriye nemli bir katkı saęlayacaktır.

- [1] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *IEEE Internet computing*, vol. 7, no. 1, pp. 76–80, 2003.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [3] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," pp. 1096–1104, 2016.
- [4] M. Vasileva, B. Plummer, A. Kovashka, D. P. Kumar, A. Farhadi, and R. Kumar, "Learning type-aware embeddings for fashion compatibility," pp. 390–405, 2018.
- [5] X. Han, Z. Wu, Z. Wu, R. Yu, and L. S. Davis, "Learning fashion compatibility with bidirectional lstms," pp. 1078–1086, 2017.
- [6] R. He and J. McAuley, "Vbpr: Visual bayesian personalized ranking from implicit feedback," 2016.
- [7] S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan, "Image-based recommendations on styles and substitutes," *arXiv preprint arXiv:1506.04757*, 2015.
- [8] T. Muhammed, "Meme kanseri histopatolojik görüntülerinin konvolüsyonel sinir ağları ile sınıflandırılması," *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 31, no. 2, pp. 391–398, 2019.
- [9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. [Online]. Available: <http://arxiv.org/abs/1512.03385>.
- [10] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1251–1258, 2017. [Online]. Available: <http://arxiv.org/abs/1610.02357>.
- [11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. [Online]. Available: <http://arxiv.org/abs/1409.1556>.
- [12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, 2016. [Online]. Available: <http://arxiv.org/abs/1512.00567>.

BİRİNCİ ÜYE

İsim-Soyisim: Doğa GÜNDOĞAR

Doğum Tarihi ve Yeri: 02.07.2001, İstanbul

E-mail: doga.gundogar@std.yildiz.edu.tr

Telefon: 0535 842 00 41

Staj Tecrübeleri: Softtech Açık Sistem Geliştiricisi

İKİNCİ ÜYE

İsim-Soyisim: Gülsüm İrem BAŞ

Doğum Tarihi ve Yeri: 02.06.2000, İstanbul

E-mail: gulsum.bas@ystd.yildiz.edu.tr

Telefon: 0531 336 43 61

Staj Tecrübeleri: Vodafone Devops System Engineer

Proje Sistem Bilgileri

Sistem ve Yazılım: MacOS İşletim Sistemi, Python

Gerekli RAM: 16GB

Gerekli Disk: 256MB