



**T.C.
BURSA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

E-TİCARET SİTELERİNE ÖNERİ SİSTEMİNİN UYGULANMASI

YÜKSEK LİSANS TEZİ

Abdullah ÖZKILIÇ

Bilgisayar Mühendisliği Anabilim Dalı

EYLÜL 2022

**T.C.
BURSA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

E-TİCARET SİTELERİNE ÖNERİ SİSTEMİNİN UYGULANMASI

YÜKSEK LİSANS TEZİ

**Abdullah ÖZKILIÇ
(161082323)**

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Volkan ALTUNTAŞ

EYLÜL 2022

BTÜ, Lisansüstü Eğitim Enstitüsü'nün 20376482063 numaralı Yüksek Lisans Öğrencisi Abdullah ÖZKILIÇ, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “E-TİCARET SİTELERİNE ÖNERİ SİSTEMİNİN UYGULANMASI” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı :Dr. Öğr. Üyesi Volkan ALTUNTAŞ
Bursa Teknik Üniversitesi

Jüri Üyeleri :Prof. Dr. Turgay Tugay BİLGİN
Bursa Teknik Üniversitesi

Prof. Dr. Murat GÖK
Yalova Üniversitesi

Teslim Tarihi :
Savunma Tarihi : 20 EKİM 2022

20.04.2016 tarihli Resmi Gazete’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, Bursa Teknik Üniversitesi’nin abonesi olduğu intihal yazılım programı kullanılarak Lisansüstü Eğitim Enstitüsü’nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.



İNTİHAL BEYANI

Bu tezde görsel, işitsel ve yazılı biçimde sunulan tüm bilgi ve sonuçların akademik ve etik kurallara uyularak tarafımdan elde edildiğini, tez içinde yer alan ancak bu çalışmaya özgü olmayan tüm sonuç ve bilgileri tezde kaynak göstererek belgelediğimi, aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.

Öğrencinin Adı Soyadı: Abdullah ÖZKILIÇ

İmzası:

XXXXXX

ÖNSÖZ

Öneri sitemleri müşterilere kişiselleştirilmiş deneyimler sunarak verimlik artışı sağlamaktadır. Her geçen gün büyüyen elektronik ticaret pazar payı için önemli bir araçtır. Bu çalışmada öneri sistemleri tanımları, geliştirmesi ve uygulanması incelenip değerlendirilmiştir.

Tez çalışma süresince bilgi, donanım ve tecrübeleri ile desteğini esirgemeyen, Yüksek Lisans eğitimim boyunca bana yol gösteren Sayın Dr. Öğr. Üyesi Volkan ALTUNTAŞ teşekkür ederim. Eğitim hayatım boyunca desteğini bir an bile esirgemeyen sevgili annem Semiha ÖZKILIÇ'a sonsuz sevgi ve saygılarımı sunarım.

Eylül 2022

Abdullah Özkılıç
Yazılım Mühendisi

İÇİNDEKİLER

	<u>Sayfa</u>
ÖNSÖZ.....	vi
İÇİNDEKİLER	vii
KISALTMALAR	x
SEMBOLLER	xi
ÇİZELGE LİSTESİ.....	xii
ŞEKİL LİSTESİ.....	xiv
ÖZET.....	xvi
SUMMARY	xvii
1. VERİ MADENCİLİĞİ	1
1.1 Tanım	1
1.2 Veri Madenciliği Süreçleri	2
1.2.1 Problemin tanımlanması.....	2
1.2.2 Verilerin hazırlanması	3
1.2.3 Model kurma ve değerlendirme	3
1.2.4 Modelin kullanılması.....	3
1.2.5 Modelin izlenmesi	3
1.3. Veri Madenciliği Modelleri.....	4
1.3.1 Sınıflama ve regresyon modelleri	4
1.3.2 Kümeleme modelleri	5
1.3.3 Birliktelik kuralları ve ardışık zamanlı örüntüler	6
2. ÖNERİ SİSTEMLERİ.....	7
2.1 Giriş	7
2.2 Kullanım Amaçları ve Faydaları	7
2.2.1 Müşteri memnuniyetini artırmak	8
2.2.2 Popüler ürünlerin/hizmetlerin görünürlüğü artırmak	8
2.2.3 Satış miktarını artırmak.....	8
2.3 Kullanılan Yöntemler	8
2.3.1 İçerik tabanlı filtreleme	8
2.3.2 İş birlikçi filtreleme.....	9
2.3.2.1 Hafıza tabanlı filtreleme	10
2.3.2.2 Model tabanlı filtreleme	11
2.4 Öneri Sistemlerinin Zorlukları	12
2.4.1 Soğuk başlangıç (Cold start) problemi.....	12
2.4.2 Seyrek veri problemi.....	12
2.4.3 Gri koyun problemi.....	12
2.4.4 Ölçeklenebilirlik problemi	13
2.5 E-Ticarette Öneri Sistemleri.....	13
2.5.1 Uygulanan alanları	13

2.6 Çalışmanın Amacı ve Önemi	16
3. LİTERATÜR ARAŞTIRMASI	17
4. MATERYAL YÖNTEM	21
4.1 Veri Seti.....	21
4.2 Sistem Verileri.....	21
4.2.1 Proje tasarımı	22
4.2.1.1 İstemciler	22
4.2.1.2 Uygulama tasarımı.....	22
4.2.1.3 Mikro servis mimarisi.....	22
4.2.2 Modüller.....	23
4.2.2.1 Katalog / Kategori modülü	23
4.2.2.2 Müşteri modülü	24
4.2.2.3 Ürün modülü.....	24
4.2.2.4 Sipariş modülü.....	24
4.2.3 Proje gereksinimleri.....	27
4.2.3.1 İşlevsel gereksinimler	27
4.2.3.2 İşlevsel olmayan gereksinimler	27
4.2.4 Proje analizi	28
4.2.4.1 Veri sözlüğü.....	28
4.2.4.2 İş modeli	28
4.2.4.3 Kullanım senaryosu kısa tanımlar	28
4.2.4.4 Kullanım senaryoları ve iş birliği diyagramları.....	32
4.2.4.5 Sınıf tasarımları	44
4.3 Aktivite Verileri	46
4.3.1 Proje tasarımı	46
4.3.1.1 İstemciler	47
4.3.1.2 Modüller	47
4.3.1.3 Modüller geliştirilirken uygulanan tasarımları	48
4.3.1.4 Uygulamanın teknolojik incelenmesi	49
4.3.1.5 İstek modelleri	50
4.3.2 Proje gereksinimleri	53
4.3.2.1 İşlevsel gereksinimler	53
4.3.2.2 İşlevsel olmayan gereksinimler	53
4.3.3 Proje analizi	54
4.3.3.1 Veri sözlüğü.....	54
4.3.3.2 İş modeli	54
4.3.3.3 Kullanım senaryolarının kısa tanımları	54
4.3.3.4 Kullanım senaryoları ve iş birliği diyagramları.....	55
4.3.3.5 Sınıf tasarımları	58
4.4 Öneri Sistemi Uygulamaları.....	59

4.4.1 Müşteri segmentasyonu	60
4.4.1.1 RFM analizi	60
4.4.2 Birliktelik analizi	64
4.4.3 İçerik tabanlı filtreleme	70
4.4.4 İşbirlikçi filtreleme.....	73
4.4.4.1 Öğe tabanlı işbirlikçi filtreleme.....	73
4.4.4.2 Kullanıcı tabanlı işbirlikçi filtreleme.....	76
4.4.4.3 Model tabanlı işbirlikçi filtreleme	80
5. SONUÇ VE ÖNERİLER.....	87
KAYNAKLAR	91
EKLER.....	95
ÖZGEÇMİŞ.....	99



KISALTMALAR

API	: Uygulama Programlama Arayüzü
DF	: Data Frame
İBD	: İş Birliği Diyagramı
İBF	: İş Birlikçi Filtreleme
İTF	: İçerik Tabanlı Filtreleme
Token	: Erişim Anahtarı

SEMBOLLER

R_u	: Kullanıcı değerlendirme ortalaması
$R_{u,j}$	: U kullanıcısının J ögesindeki değerlendirmesi
$R_{u,i}$	: U kullanıcısının İ ögesindeki değerlendirmesi
U_j	: J ögesine puan veren tüm kullanıcılar
U_i	: İ ögesine puan veren tüm kullanıcılar



ÇİZELGE LİSTESİ

Sayfa

Çizelge 4.1 : Katalog/Kategori modülü veri örneği.....	24
Çizelge 4.2 : Müşteri modülü veri örneği.....	24
Çizelge 4.3 : Ürün modülü veri örneği.....	24
Çizelge 4.4 : Sipariş modülü veri örneği.....	24
Çizelge 4.5 : Siparişe ait ürün veri örneği.....	25
Çizelge 4.6 : Sistem verileri veri sözlüğü.....	28
Çizelge 4.7 : Kullanım senaryoları ve aktörler.....	28
Çizelge 4.8 : Sistemden token alma kullanım senaryosu	29
Çizelge 4.9 : Sisteme yeni kayıt ekleme kullanım senaryosu.....	30
Çizelge 4.10 : Sisteme yeni kayıt güncelleme kullanım senaryosu.....	30
Çizelge 4.11 : Sistemden kayıt silme kullanım senaryosu.....	31
Çizelge 4.12 : Sistemden token alma kullanım senaryosu için senaryo 1.....	32
Çizelge 4.13 : Sistemden token alma kullanım senaryosu için senaryo 2.....	33
Çizelge 4.14 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 1.....	34
Çizelge 4.15 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 2.....	35
Çizelge 4.16 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 3.....	36
Çizelge 4.17 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 4.....	36
Çizelge 4.18 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 1..	37
Çizelge 4.19 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 2..	38
Çizelge 4.20 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 3..	39
Çizelge 4.21 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 3..	40
Çizelge 4.22 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 1.....	40
Çizelge 4.23 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 2.....	41
Çizelge 4.24 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 3.....	42
Çizelge 4.25 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 4.....	43
Çizelge 4.26 : Aktivite verilerinin toplanması için geliştirilen uygulamanın genel istek modeli.....	50
Çizelge 4.27 : Görüntüleme metodu istek modeli.....	51
Çizelge 4.28 : Arama metodu istek modeli.....	51
Çizelge 4.29 : Sepet metodu istek modeli.....	52
Çizelge 4.30 : Sepet metodu BasketInfo nesne modeli.....	52
Çizelge 4.31 : Favori/İstek metodu istek modeli.....	53
Çizelge 4.32 : Favori/İstek metodu WishInfo nesne modeli.....	53
Çizelge 4.33 : Aktivite verileri veri sözlüğü.....	54
Çizelge 4.34 : Kullanım senaryoları ve aktörler.....	54
Çizelge 4.35 : Sisteme veri ekleme kullanım senaryosu.....	55
Çizelge 4.36 : Sisteme veri ekleme kullanım senaryosu için senaryo 1.....	55
Çizelge 4.37 : Sisteme veri ekleme kullanım senaryosu için senaryo 2.....	56
Çizelge 4.38 : Sisteme veri ekleme kullanım senaryosu için senaryo 3.....	57
Çizelge 4.39 : Sisteme veri ekleme kullanım senaryosu için senaryo 4.....	58
Çizelge 4.40 : RFM analizinde kullanılan veri özeti.....	62

Çizelge 4.41 : RFM analiz sonucu özet görünümü.....	62
Çizelge 4.42 : Birliktelik kuralı kullanılan veri setinin incelenmesi.	65
Çizelge 4.43 : ARL veri yapısı.	65
Çizelge 4.44 : Birliktelik kuralı veri setinin son yapısı.	66
Çizelge 4.45 : Oluşturulan birliktelik kuralı.	67
Çizelge 4.46 : Ürün bazlı birliktelik kuralı.	68
Çizelge 4.47 : İçerik tabanlı filtreleme 'de kullanılan veri setinin incelenmesi.	70
Çizelge 4.48 : 'CINTHYA ELBISE 2586' ürünü için öneriler.	72
Çizelge 4.49 : Öğe tabanlı İF'de kullanılan veri setinin incelenmesi.....	74
Çizelge 4.50 : Öğe tabanlı İF'de kullanılan pivot tablosu.	75
Çizelge 4.51 : Öğe tabanlı İF'de ProductID:5605 ürünü için öneriler.	75
Çizelge 4.52 : Öğe tabanlı İF'de ProductID:4034 ürünü için öneriler.	75
Çizelge 4.53 : Kullanıcı tabanlı İF'de kullanılan veri setinin incelenmesi.....	77
Çizelge 4.54 : '340094' Id'li kullanıcının pivot tablosu.....	78
Çizelge 4.55 : Kullanıcı tabanlı İF' de benzer ürünlere değerlendirme yapan kullanıcılar.....	78
Çizelge 4.56 : Kullanıcı tabanlı İF'de korelasyon DF'i.	79
Çizelge 4.57 : Kullanıcı tabanlı İF'de önerilecek ürünler listesi.....	79
Çizelge 4.58 : Kullanıcı tabanlı İF'de kullanılan veri setinin incelenmesi.....	81
Çizelge 4.59 : Model tabanlı İF'de kullanılan pivot tablosu.	82
Çizelge 4.60 : Oluşturulan modelin tahmin özeti.	85

ŞEKİL LİSTESİ

Şekil 1.1 : Veri madenciliğin birçok disiplinle birleşimi [5].	1
Şekil 1.2 : Basit bir veri madenciliği süreci [5].	2
Şekil 2.1 : İTF şeması [24].	9
Şekil 2.2 : İBF şeması [26].	10
Şekil 2.3 : perspective.com.tr arama çubuğu önerileri.	14
Şekil 2.4 : oxxo.com.tr açılır pencere önerileri.	14
Şekil 2.5 : perspective.com.tr katalog/kategori önerileri.	15
Şekil 2.6 : trendyol.com birlikte alınan ürün önerileri.	15
Şekil 2.7 : trendyol.com benzer ürünler.	15
Şekil 4.1 : Sistem verilerinin toplanması için geliştirilen uygulama mimarisi.	23
Şekil 4.2 : Sistemden token alma kullanım senaryosu	29
Şekil 4.3: Sisteme yeni kayıt ekleme kullanım senaryosu.	29
Şekil 4.4 : Sistemdeki kayıtlar güncelleme kullanım senaryosu.	30
Şekil 4.5 : Sistemdeki kayıtları silme kullanım senaryosu.	31
Şekil 4.6: Sistem verileri kullanım senaryolarının diyagramı.	32
Şekil 4.7: Sistemden token alma kullanım senaryosu için normal senaryo İBD 1...	33
Şekil 4.8 : Sistemden token alma kullanım senaryosu için anormal senaryo İBD 1.	33
Şekil 4.9 : Sistemden token alma kullanım senaryosu için normal senaryo İBD 2...	34
Şekil 4.10 : Sistemden token alma kullanım senaryosu için normal senaryo İBD 2.	34
Şekil 4.11 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 1..	34
Şekil 4.12 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 1.	35
Şekil 4.13 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 2..	35
Şekil 4.14 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 2.	35
Şekil 4.15 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 3..	36
Şekil 4.16 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 3.	36
Şekil 4.17 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 4..	37
Şekil 4.18 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 4.	37
Şekil 4.19 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 1.	38
Şekil 4.20 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 1.	38
Şekil 4.21 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 2.	38
Şekil 4.22 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 2.	39
Şekil 4.23 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 3.	39
Şekil 4.24 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 3.	39
Şekil 4.25 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 4.	40
Şekil 4.26 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 4.	40

Şekil 4.27 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD	
1.....	41
Şekil 4.28 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD	
1.....	41
Şekil 4.29 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD	
2.....	42
Şekil 4.30 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD	
2.....	42
Şekil 4.31 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD	
3.....	43
Şekil 4.32 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD	
3.....	43
Şekil 4.33 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD	
4.....	43
Şekil 4.34 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD	
4.....	44
Şekil 4.35 : Sistemden token alma kullanım senaryosu sıralama diyagramı.....	44
Şekil 4.36 : Sisteme kayıt ekleme kullanım senaryosu sıralama diyagramı.....	45
Şekil 4.37 : Sistemdeki kayıtları güncelleme kullanım senaryosu sıralama diyagramı.	
.....	45
Şekil 4.38 : Sistemdeki kayıtları silme kullanım senaryosu sıralama diyagramı.	46
Şekil 4.39 : Aktivite verilerinin toplanması için geliştirilen uygulama yapısı.	48
Şekil 4.40 : Sisteme veri ekleme kullanım senaryosu.	55
Şekil 4.41 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 1....	56
Şekil 4.42 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 1..	56
Şekil 4.43 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 2....	56
Şekil 4.44 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 2..	57
Şekil 4.45 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 3....	57
Şekil 4.46 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 3..	57
Şekil 4.47 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 4....	58
Şekil 4.48 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 4..	58
Şekil 4.49 : Sisteme veri ekleme kullanım senaryosu sıralama diyagramı.....	59
Şekil 4.50 : Öneri sistemi uygulama tasarımı.	59
Şekil 4.51 : RFM analizinde gruplandırma.....	61
Şekil 4.52 : RFM analiz sonucu grup dağılımı.	63
Şekil 4.53 : Birliktelik kuralı uygulama tasarımı.	69
Şekil 4.54 : İçerik tabanlı filtreleme uygulama tasarımı.....	73
Şekil 4.55 : Öge tabanlı İF uygulama tasarımı.	76
Şekil 4.56 : Kullanıcı tabanlı işbirlikçi filtreleme uygulama tasarımı.	80
Şekil 4.57 : Model tabanlı İF’de Kullanıcı ürün matrisinin iki boyutlu matrise	
ayrıştırılması.....	83
Şekil 4.58 : Model tabanlı İF’de Kullanıcı-Ürün matrisini oluşturan faktörler.....	83
Şekil 4.59 : Model tabanlı İF uygulama tasarımı.	86

E-TİCARET SİTELERİNE ÖNERİ SİSTEMİNİN UYGULANMASI

ÖZET

Dijitalleşme sadece metalar ve maddi varlıklar doğrultusunda değil, kişilerin davranış ve alışkanlıkları üzerinden de mümkün bir hâle gelmiştir. Bu durum, birçok alan ve amaç için yeni veri depolarının doğmasına sebebiyet vermiştir. Söz konusu veri kaynaklarını kullanmak ise başta e-ticaret olmak üzere çeşitli alanlarda verimliliğin artmasında rol oynamıştır. Bu çalışmada, depolanan verilerin aktif bir şekilde kullanılabilmesi adına geliştirilmiş olan veri madenciliği, veri madenciliğinde maksimum fayda sağlayan öneri sistemleri ve öneri sistemlerinin verimli kullanımına katkı sağlayacak öneri sistemi uygulamaları ele alınmıştır.

Detaylandırılmış alt kısımlarla birlikte dört ana bölüme ayrılmış olan bu çalışmada öncelikle veri madenciliğine yer verilmiştir. Her bir ana bölüm içeriğine bağlı olarak alt kısımlara ayrılmış olup ana başlıkla ilgili tüm detayları içerecek şekilde hazırlanmıştır. Olası problemlere çözüm önerileri verilirken ihtiyaçlar birçok farklı açıdan da değerlendirilmiştir.

İlk bölümde yer verilen veri madenciliğinin tanımsal kavramlarıyla birlikte süreçleri ve modelleri, yol haritalarıyla desteklenip konu bağlamında anlatılmıştır. Bu bölüm, ana fikri destekleme noktasında önem taşımanın yanında çalışmanın üzerine oturtulduğu temellerden biri olarak da görev yapmaktadır.

İkinci bölüm ise veri madenciliği ile elde edilen verilerden yola çıkarak uygulamaya konulan öneri sistemlerine ayrılmıştır. Öneri sistemlerinin kullanım amaçları, uygulama yöntemleri ve faydalarının yanı sıra getirdiği zorluklar da incelenmiştir. Ayrıca e-ticaret sektöründeki aktif ve gelişmeye devam eden öneri sistemlerinden de bahsedilmiştir.

Literatür araştırmasının yer aldığı üçüncü bölümün ardından dördüncü bölüm olan materyal yönteme geçilmiştir. Burada da veri madenciliği temelinden ve öneri sistemleri bağlamından kopulmamış olup ilk olarak sistem verilerine değinilmiştir. Sistem verileri, aktivite verileri ile desteklenmiştir. Bu süreçte proje tasarımları ve yapılması gereken analizler de öne sürülmüştür.

Son kısım, veri madenciliğini doğru ve etkin bir şekilde uygulayıp majör faydalara erişimin stratejisini belirleme açısından önem taşımaktadır. Bu çalışma, öneri sistemi uygulamalarının mevcut verimini yukarı taşımak için bir ihtiyaç olarak doğan segmentasyon, analiz ve filtreleme süreçlerini yönetebilmek adına önermeler sunmaktadır. Söz konusu önermeler sunulurken istatistikler ve örnek çalışmaların performansları referans alınmıştır.

Anahtar kelimeler: Öneri sistemleri, E-Ticaret, İçerik Tabanlı Filtreleme, İş Birlikçi Filtreleme, Veri Madenciliği

APPLICATION OF SUGGESTION SYSTEM TO E-COMMERCE SITES

SUMMARY

Digitalization has become possible not only through commodities and tangible assets, but also through human behavior and habits. This has led to the emergence of new data stores for many fields and objectives. Productivity has increased across a variety of industries thanks to the use of these data sources, particularly in e-commerce. Data mining, a technology created to make active use of stored data, recommender systems that maximize data mining benefits, and applications for recommender systems that will help make use of them more effectively are discussed in this study.

This study, which has four main sections and detailed subsections, focuses primarily on data mining. Each main section is prepared to cover all the details pertinent to the main topic and divided into subsections based on its content. The needs are assessed from many different perspectives while providing solutions to potential problems.

The descriptive concepts, along with the data mining processes and models, which are included in the first chapter, are supported by road maps and described in the context of the subject. In addition to playing a crucial role in supporting the main idea, this part also acts as one of the foundations on which the study is based.

The second section is devoted to the recommender systems that are implemented using the data obtained by data mining. The issues that recommender systems present are also analyzed, along with their uses, applications, and advantages. In addition, active and developing recommender systems in the e-commerce industry are also mentioned.

The third section, which comprises the literature review, is followed by the fourth section that focuses on the material method. The system data is mentioned first in this case as well without deviating from the basis of data mining and the context of recommender systems. System data is corroborated by activity data. In this process, project designs and necessary analyzes are also presented.

The last section is crucial for correctly and effectively implementing data mining and determining a strategy to provide the major advantages. This study puts forward propositions suggestions to manage the segmentation, analysis and filtering processes that emerge as a need to improve the current efficiency of the recommender system applications. Statistics and performances of case studies are also considered as reference while presenting the propositions.

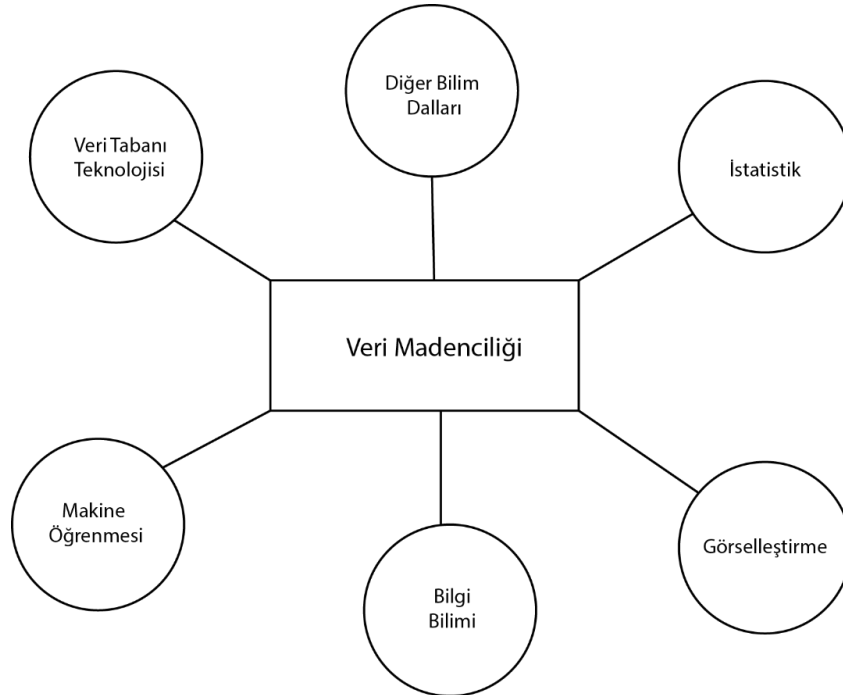
Keywords: Recommender System, E-Commerce, Content Based Filtering, Collaborative Filtering, Data Mining

1. VERİ MADENCİLİĞİ

1.1 Tanım

Bilgisayar teknolojilerinin gelişmesi ile son zamanlarda hem veri üretme hem de toplama süreçleri hızla gelişiyor ve artıyor. Birçok ticari ve devlet işlemlerinin online sistemler üzerinden gerçekleştirilmesi ve veri toplama araçlarındaki gelişmeler çok büyük miktarda veri depolarının oluşmasını sağlamıştır. Güçlü ve uygun fiyatlı veri tabanlarının sayısı hızla artmakta ve verilerdeki bu patlayıcı büyüme, toplanan verilerin otomatik olarak yararlı bilgi ve bilgiye dönüştürülebilen yeni teknikler ve araçlara ihtiyaç oluşturdu. Bu ve benzeri sebeplerden veri madenciliği önemi artan bir araştırma ve geliştirme alanı haline gelmiştir [1].

Veri tabanlarında bilgi keşfi olarak da adlandırılan veri madenciliği, veri tabanlarındaki verilerden örtük, önceden bilinmeyen ve potansiyel olarak yararlı bilgilerin çıkarılması süreci anlamına gelir. Şekil 1.1’de görüldüğü gibi veri madenciliği birçok disiplini içeren bir alandır.



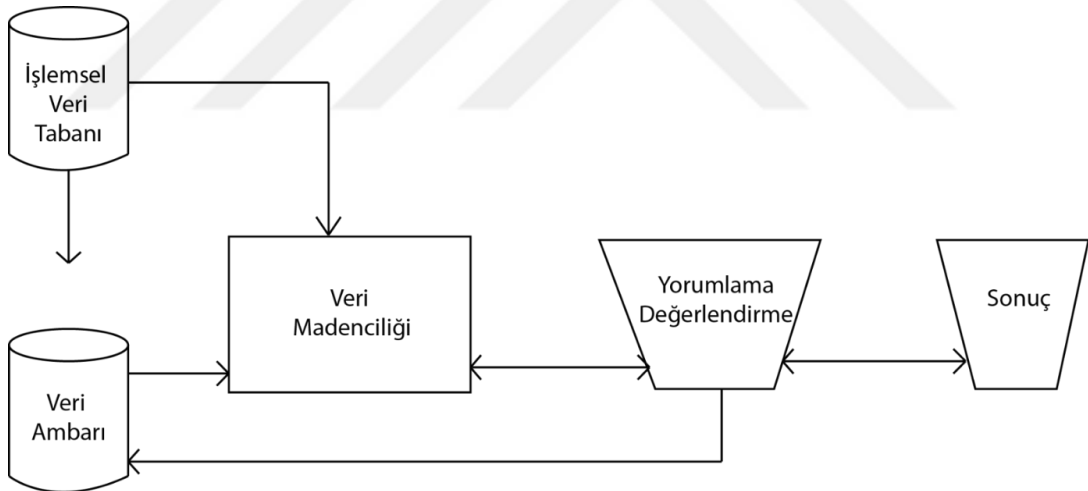
Şekil 1.1 : Veri madenciliğin birçok disiplinle birleşimi [5].

Veri Madenciliği belirli adımları takip eden bir süreçtir. Bu süreçler doğru bir şekilde izlenilmezse istenilen sonuca ulaşılması mümkün değildir. Veri madenciliği bilgi keşfi işlemidir, bu keşif aşamaları aşağıdaki gibi sıralanabilir [2].

1. Veri Temizleme
2. Veri Bütünleştirme
3. Veri Seçme
4. Veri Dönüşümü
5. Veri Madenciliği
6. Örüntü Değerlendirme
7. Bilgi Sunumu

1.2 Veri Madenciliği Süreçleri

Şekil 1.2’de basit bir veri madenciliği süreci görülmektedir. Bu başlığın devamında Şekil 1.2’de görülen işlemler alt başlıklar ile anlatılmıştır.



Şekil 1.2 : Basit bir veri madenciliği süreci [5].

1.2.1 Problemin tanımlanması

Problem açık bir şekilde ifade edilmiş ve uygulamanın kavramsal analizi iyi bir şekilde hazırlanmış olmalıdır.

1.2.2 Verilerin hazırlanması

- **Veri toplama**

Veriler birçok farklı ortamlarda depolanabilir. Birden çok veri tabanı, veri ambarı veya veri göllerinden saklanmış olabilir. İlk olarak geliştirilecek uygulama için uygun veriler veri saklama alanlarından toplanır. Bu işlemin ardında veriler test ve analiz olarak iki guruba ayrılır. Genellikle %80'i analiz %20'si test verisi olarak ayrılmaktadır [3].

- **Veri temizleme**

Veri temizleme işlemi, verileri içerisindeki uygun olmayan veya hatalı verileri ayıklanmasıdır. Bu işlemde genellikle eksik veriler uygun değerler ile doldurulur veya ilgili kayıt silinir [2].

- **Veri dönüştürme**

Veri depolarında veriler farklı formatlarda depolanmış olabilirler. Veri dönüşümünün amacı kullanıma uygun formatlara veya uygun değerlere dönüştürmektir [3].

- **Veri indirgenmesi**

Çok büyük veri kümelerinden daha küçük veri kümelerinin elde edilmesidir [7].

1.2.3 Model kurma ve değerlendirme

Veri madenciliğinin en önemli sürecidir. Modelin doğru kurulması için problemin doğru tanımlanmış olması gerekmektedir. Tanımlanan problem için en uygun modelin bulunabilmesi, olabildiğince çok sayıda model kurulup denenmesi ile mümkündür [4].

1.2.4 Modelin kullanılması

Kurulan model bir uygulama parçası olarak kullanılabileceği gibi yeni bir uygulama olarak da kullanılabilir [7].

1.2.5 Modelin izlenmesi

Zaman içerisinde değişen iş talepleri, veri akışları uygulamanın güncelliğini yitirmesine sebep olabilir. Değişkenlik gösteren süreçler karşısında modelin performansı incelenmeli gerekirse tekrar kurgulanmalıdır.

1.3. Veri Madenciliği Modelleri

Veri Madenciliğinde kullanılan modeller tahmin edici (Predictive) ve tanımlayıcı (Descriptive) olmak üzere iki ana başlık altında incelenmektedir [8].

Tahmin Edici (Predictive) Modeller: Sonuçları bilinen verilerden hareket edilerek model kurulması ve kurulan modelden yararlanılarak bilinmeyen veri kümelerinin sonuç değerlerinin tahmin edilmesini amaçlamaktadır [4].

Tanımlayıcı (Descriptive) Modeller: Veri setindeki bağıntı ve profilleri belirleyen ve yorumu kullanıcıya bırakan modellerdir [9].

1.3.1 Sınıflama ve regresyon modelleri

Sınıflama ve regresyon, önemli veri sınıflarını ortaya koyan veya gelecek veri eğilimlerini tahmin eden modelleri kurabilen iki veri analiz yöntemidir. Sınıflama kategorik değerleri tahmin ederken, regresyon süreklilik gösteren değerlerin tahmin edilmesinde kullanılır [5].

Kullanılan başlıca teknikler şunlardır [10]:

- **Karar ağaçları:** Ağaç yapısı ile kolay anlaşılabilen kurallar yaratabilen, bilgi teknolojileri işlemleri ile kolay entegre olabilen en popüler sınıflama tekniğidir [11]. Veri tabanı sistemleri ile kolayca entegre olabilmesi güvenilirliklerinin yüksek olması, kurulması ve yorumlanmasının kolay olması gibi avantajları ile veri madenciliğinde sınıflama modelleri arasında kullanılan yaygın bir tekniktir [4].
- **Yapay sinir ağları:** Beynin bir işlemi yerine getirme yöntemini modellemek için tasarlanan bir sistem olarak tanımlanabilir. Yapay sinir hücrelerini birbirleri ile çeşitli şekiller ile bağlanmasından oluşur. Beynin bilgi işleme yöntemine uygun olarak, bir öğrenme sürecinden sonra bilgiyi toplama, hücreler arasındaki bağlantı ağları ile bu bilgiyi saklama ve genelleme yeteneğine sahip paralel dağıtılmış bir işlemcidir [12].
- **Genetik algoritmalar:** Karmaşık düzenli problemlerin çözümünü gerçekleştirmek amacı ile kromozomların yeni diziler yürütme esasını temel alan sezgisel bir araştırma yöntemidir. Matematiksel fonksiyonların global optimizasyonu hedeflenir. Diğer araştırma yöntemlerinden ayıran özellik ise,

bir çözüm seti ile başladıktan sonra, geliştirme için biyolojik evrimi esas alan bir prosesin kullanılmasıdır [13].

- **K-En yakın komşu:** Kayıtlar, bir veri uzayındaki noktalar olarak düşünülürse, birbirine yakın olan kayıtlar, birbirinin yakın komşusu olur. Eğer belirli bir kişinin davranışı tahmin edilmek isteniyorsa, veri uzayında o kişiye yakın, 'n' kadar kişinin davranışına bakılır. Bu 'n' kişinin davranışlarının ortalaması hesaplanır ve bu ortalama belirlenen kişi için tahmin olur [12].
- **Bellek temelli nedenleme:** Ampirik bir sınıflandırma yöntemidir. Yeni sınıflandırılmamış kayıtları bilinen örnekler ve kalıplarla karşılaştırarak çalışır. Yeni kayıtla en yakından eşleşen durum, bir dizi farklı olası önlemden biri kullanılarak belirlenir. Bu yöntem, sayısal verilere olduğu kadar metinsel verilere de uygulanabilir [14].
- **Naive-Bayes:** Niteliklerin sınıfa göre koşullu olarak bağımsız olduğuna dair güçlü bir varsayımla birlikte Bayes kuralını kullanan basit bir öğrenme algoritmasıdır. Bu bağımsız varsayımı uygulama esnasında sıklıkla ihlal edilse de saf Bayes yine de sıklıkla rekabetçi sınıflandırma doğruluğu sağlar. Hesaplama verimliliği ve birçok özelliği ile bu saf Bayes'in pratikte yaygın olarak kullanılmasını sağlar [15].

1.3.2 Kümeleme modelleri

Kümeleme, veriyi sınıflara veya kümelere ayırma işlemidir [16]. Kümeleme modellerinde amaç, küme üyelerinin birbirine çok benzediği, ancak özellikleri birbirinden çok farklı kümelerin oluşturulmasıdır. Literatürde pek çok kümeleme algoritması bulunmaktadır. Kullanılacak olan algoritma seçimi probleme ve veri tipine bağlıdır. Genel olarak başlıca kümeleme yöntemleri aşağıdaki gibi sınıflandırılabilir [5].

- Bölme Yöntemleri
- Hiyerarşik Yöntemler
- Yoğunluk Tabanlı Yöntemler
- Izgara Tabanlı Yöntemler
- Model Tabanlı Yöntemler

1.3.3 Birliktelik kuralları ve ardışık zamanlı örüntüler

Büyük veri kümeleri arasında birliktelik ilişkileri bulurlar [9,17]. En tipik kullanım örneği market sepeti uygulamasıdır [9,18]. Birliktelik kuralları özel bir kural formunda verilir [6]. Bu özel form sol ve sağ kısım olmak üzere iki kısımdan oluşur. Bu bölümlerde iş veya nesneler yer alır ve veri arasındaki ilişkiler eğer-sonra ifadeleri ile gösterilir [7]. Eğer <bazı şartlar sağlanırsa> sonra <bazı niteliklerin değerlerini tahmin et>. En yaygın birliktelik kuralı algoritmaları arasında GRI (The Generalized Rule Induction), Apriori, sayılabilir [19].

Ardışık zamanlı örüntüler birbirleri ile ilişkisi olan ancak birbirini izleyen dönemlerde gerçekleşen ilişkilerin tanımlanmasında kullanılır [10].



2. ÖNERİ SİSTEMLERİ

2.1 Giriş

Öneri Sistemleri, bir kullanıcının geçmiş verilerini kullanarak bir öğe/içeriğin beğenmesini/tercih etmesini tahmin eden sistemlerdir. Öneriler, hangi öğelerin/ürünlerin satın alınacağı, hangi podcastin dinleneceği veya hangi içeriğin tıklanacağı gibi çeşitli karar verme süreçleriyle ilgilidir [20].

Günlük yaşantımızın neredeyse her alanında karşımıza çıkmaktadır. Online sistemler üzerinde izlediğimiz filmler, gezindiğimiz sosyal medya içerikleri, okuduğumuz haberler, satın aldığımız ürünler vb. birçok alanda etkileşimde bulunduğumuz veriler öneri/tavsiye sistemleri ile eğitilerek ilgimizi çekebilecek içerikleri bize sunmaktadır.

Öneri Sistemleri kullanıcıyı tercihlerine tanıyarak, kullanıcıların tercih edebileceği ürünleri/hizmetleri tespit edip kullanıcıya bu öğeleri sunarak sistem üzerindeki deneyimini iyileştirmeyi hedeflemektedir.

2.2 Kullanım Amaçları ve Faydaları

Online sistemlerde, öneri sistemleri kullanıcılara ilgilenebilecekleri ürün/hizmet öğelerine ulaşımını kolaylaştırarak müşteri deneyimi ve sistem etkileşim/satış rakamlarının arttırması hedeflemektedir.

Online sistemlerde kullanıcı profillerine uygun içerik/ürün/hizmet özelleştirilmesi, kullanıcıların deneyimlerini iyileştirmesine yardımcı olmaktadır. Kullanıcı profillerini tanıyan sistemlerin geliştirilmesi;

- Müşteri kayıplarını azaltmak
- Müşteri memnuniyetini yükseltmek
- Ürün/içerik erişimini kolaylaştırmak
- Satış ve etkileşimi arttırmak

Online sistemlere sağladığı temel faydalarındandır.

2.2.1 Müşteri memnuniyetini artırmak

Kullanıcı profillerine uygun ürün tavsiyeleri, arama sonuçları ve liste sıralamaları sistem üzerinde kullanıcının aradığı ürün/hizmete daha hızlı ulaşmasına olanak tanır. Kullanıcının her etkileşiminde yeni verilerin tekrar algoritmalar ile eğitilerek kullanıcıya yeni bir deneyim sunabilir. Kullanıcıyı tanıyan ve kullanıcıya özgün içerik sunan sistemler müşteri memnuniyeti ve sadakatini arttırabilir.

2.2.2 Popüler ürünlerin/hizmetlerin görünürlüğü artırmak

Sistem üzerinden en çok görüntülenen, satılan ürünlerin/hizmetlerin öneri ve/veya sıralama aracılığı ile kullanıcılara ulaşılabilirliğini arttırabilir. Bu tarz önerilerde demografik filtre kullanılarak, kullanıcının yaşadığı bölgede trendleri takip etmesi ve bu trendlere kolay ulaşımı kullanıcı sistem deneyimini iyileştirme bilmektedir.

2.2.3 Satış miktarını artırmak

Kişiselleştirilmiş tüketime uygun hale getirilmiş ürünlere/hizmete kullanıcının aradığı ürüne daha kolay ulaşmasını, kullanıcının davranış ve tercihleri dikkate alınarak ilgilenebileceği ürünlerin önerilmesi, trendlere kolay ulaşımı sipariş ve sipariş ürün miktarlarını arttırabilir.

2.3 Kullanılan Yöntemler

Öneri sistemleri Bilgi Erişimi, Veri Madenciliği ve Makine Öğrenimi sınıfına aittir. 3 temel sınıf altında incelenebilir. Her biri farklı tekniklere göre alt yöntemlere ayrılmaktadır.

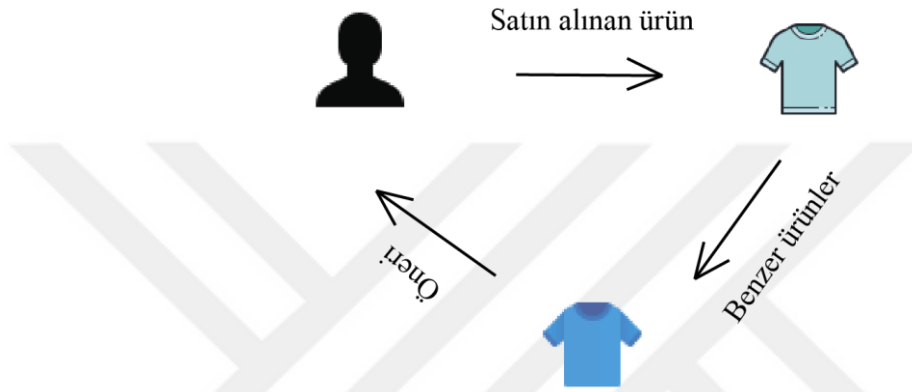
2.3.1 İçerik tabanlı filtreleme

Ürün tanımı ve kullanıcı yöneliminin profili, İTF de önemli bir rol oynar. İTF algoritmaları, benzerlik sayısına göre öğeler önermeye çalışır. En iyi eşleşen öğeler, çeşitli aday öğelerin daha önce kullanıcı tarafından derecelendirilen öğelerle karşılaştırılmasıyla önerilir [21].

Birçok güncel İTF sistemi, kullanıcıların profilini ve öğelerin profilini oluşturur. Bir kullanıcı profili, kullanıcıların anketlerinden alınabilecek veya zaman içindeki işlem davranışlarından öğrenilebilecek kullanıcının zevkleri, tercihleri ve ihtiyaçları hakkında bilgi içerirken, bir öğe profili bir dizi öğe niteliği içerir. Sistem daha sonra

kullanıcı profilinin ve tüm öğelerin her profilinin benzerliğini hesaplar ve kullanıcıların ihtiyaçlarını veya zevklerini karşılayabilecek öğeleri önerir [21].

İTF kolay, güvenilir ve maliyeti düşüktür. Ürün içerik bilgisi temel alınarak öneri sunduğu için görüntülenen ve/veya aksiyon alınan ürüne benzer ürünleri önerir. Bu yaklaşım için ürünler hakkında kapsamlı bilgi gereklidir. Kullanıcının aksiyon aldığı ürünün içeriği diğer ürünler ile eşleştirilerek en yüksek eşleşme yakalanan ürünleri kullanıcıya önerir. Bu yaklaşımın E-Ticaret sitelerindeki uygulama alanı sıklıkla ürün detay sayfalarında ‘Benzer ürünler’ listesi olarak karşımıza çıkmaktadır.



Şekil 2.1 : İTF şeması [24].

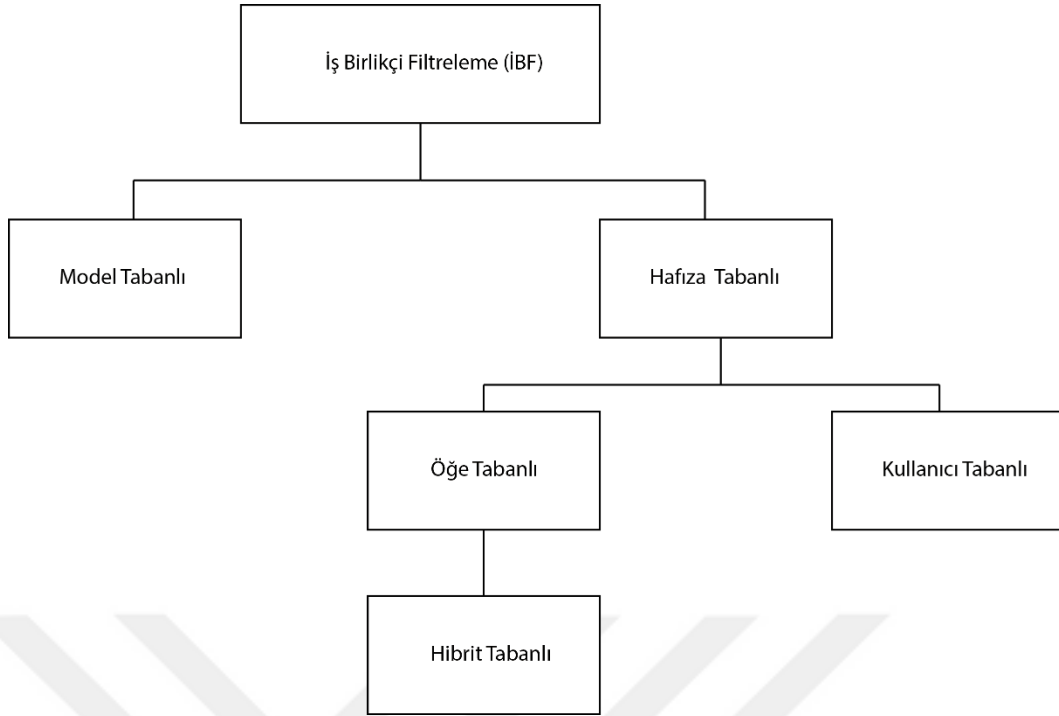
Şekil 2.1’de görüldüğü gibi kullanıcı A ürününü satın aldıysa, A ürününün kol tipi, kumaş tipi, kesim tipi, model tipi vb. bilgileri diğer ürünler ile kıyaslanarak A ürününe en yakın olan ürünler önerilir.

İçerik Tabanlı filtreleme, ürünlerin meta verilerine dayanarak kullanıcı değerlendirmeleri olmadan da öneriler sunulabilir [25]. Sistemde yeni katılan üyelere öneriler sunmak için oldukça faydalı bir yöntemdir. Ürünlerin meta verilerin doğruluğu ve yoğunluğu öneri kalitesine ile doğru orantılıdır.

2.3.2 İş birlikçi filtreleme

Kullanıcıların davranış, aktivite veya tercihlerine dayalı ve o kullanıcının beğenisini tahmin eden büyük miktarda bilgiyi diğer kullanıcılarla olan benzerliklerini kullanarak toplamak ve incelemek üzerine kuruludur [22].

İBF Filtreleme Şekil 2.2 ’de gösterildiği gibi 3 temel kategoride sınıflandırılabilir. Model tabanlı, hafıza tabanlı ve hibrit tabanlı yöntemler ile yürütülen araştırmalar vardır [23].



Şekil 2.2 : İBF şeması [26].

2.3.2.1 Hafıza tabanlı filtreleme

Hafıza tabanlı yöntemleri, bilinmeyen derecelendirmeyi tahmin etmek için önceki verileri kullanmayı amaçlar. Sonuçları tahmin etmek için tüm kullanıcı veri tabanı üzerinde çalışırlar. Belleğe dayalı CF yöntemleri, temel olarak benzer ilgi alanlarına sahip bir grup insanın kullanıcı üyelerine bağlıdır. Çeşitli uzaklık ölçüleri kullanarak en yakın komşular üzerinden öneri üretmektedir [27]. Büyük veri kümelerinde gerçek zamanlı öneriler üretmek zorlu bir süreçtir [23].

Öğeler tabanlı filtreleme yöntemi

Öğeler tabanlı İBF iki aşama vardır. İlk aşamada, sayı benzerlik ölçülerinden biri kullanılarak öğeler arasındaki benzerlikler bulunur. İkinci aşamada, bilinmeyen öğe için tahmin derecelendirmelerini kullanarak benzerlik değerleri değerlendirilir. [26]

Kosinüs tabanlı benzerlik yöntemi öğeler arasındaki benzerliği hesaplamının yaygın bir yoludur [28].

$$item(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u) \cdot (R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \cdot \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}} \quad (2.1)$$

$R_{u,i}$ ve $R_{u,j}$ sırasıyla i ve j öğelerindeki U kullanıcısının derecelendirmesini temsil ederken, R_u , kullanıcının derecelendirmelerinin ortalamasıdır ve $U_{i,j}$ i ve j öğelerine puan veren tüm kullanıcıları temsil eder. Ayrıca U kullanıcısı ve J öğesi için madde bazlı en yakın komşu yöntemi için tahmin hesaplaması aşağıdaki denklemdeki gibidir [27].

$$(u_t, j) = \frac{\sum_{i \in R_{u_t, j}} \text{Itemsim}(i, j) \cdot R_{u_t, j}}{\sum_{i \in R_{u_t, j}} \text{Itemsim}(i, j)} \quad (2.2)$$

Kullanıcı tabanlı filtreleme yöntemi

Geçmiş derecelendirme davranışı güncel kullanıcının kine benzer olan başka bir kullanıcı bulunur ve mevcut kullanıcının neyi seveceğini tahmin etmek için diğer öğelerdeki değerlendirmelerini kullanılır. Güncel kullanıcının değerlendirdiği öğeler ile veri tabanındaki kullanıcıların değerlendirmeleri arasında yüksek benzerlik aranır. Bu kullanıcıların söz konusu olabilecek öğeye ilişkin değerlendirmeleri ağırlıklandırılır, güncel kullanıcıya bu ağırlıklara göre öneri yapılır [29].

Hibrit tabanlı filtreleme yöntemi

Tahminler veya öneriler yapmak için İBF tekniklerini birbiriyle birleştiren İBF üçüncü türü olarak kabul edilir. Hibrit tabanlı İBF, esas olarak hibridizasyonun veya bellek ve model tabanlı işbirlikçi kombinasyonunun sonucudur. Önceki iki yöntemin bir karışımıdır, bu nedenle belleğin özelliklerinden yararlanır ve dezavantajlarından kaçınır ve modelin özelliklerinden yararlanır [27].

2.3.2.2 Model tabanlı filtreleme

Model tabanlı öneri sistemlerinin arkasındaki temel fikir, veri kümesi derecelendirmelerinin yardımıyla bir "model" oluşturmaktır. Başka bir deyişle, veri kümesinden bazı bilgilerin çıkarılması ve bu bilgileri her seferinde tam veri kümesi kullanmadan önerilerde bulunmak için bir "model" olarak kullanma sürecidir diyebiliriz. Bu yaklaşım hem hız hem de ölçeklenebilirlik açısından faydalıdır. Model tabanlı yaklaşım, algoritmanın tahmin doğruluğunu da geliştirir [22].

Bu modeller oluşturulurken;

- Tekil Değer Ayırıştırma (Singular Value Decomposition – SVD)

- Boyutsallık Azaltma Tekniđi (Dimensionality Reduction Technique)
- Matris Tamamlama Tekniđi (Matrix Completion Technique)
- Saklı Anlamsal Yöntemler (Latent Semantic Methods)
- Regresyon (Regression)
- Kümeleme (Clustering)

gibi teknikler kullanılır [30].

2.4 Öneri Sistemlerinin Zorlukları

Öneri sistemlerinin temel amacı doğru ve güvenilir öneriler yapmaktır. Çeşitli tekniklerin getirdiđi avantajların yanında farklı problemleri de yanında getirmektedir.

2.4.1 Soğuk başlangıç (Cold start) problemi

İF’de kullanıcı b kullanıcısına benzer ise b kullanıcısının beğendiđi ürünleri/hizmetleri a kullanıcı da beğenebilir mantıđı ile öneri yapmaktadır. Sistemde yeni olan bir kullanıcı hakkında herhangi bir bilgiye sahip olmadığı için diğerkullanıcılar ile karşılaştırılmaz. Aynı sorun öğeler içinde geçerlidir. İçerik tabanlı filtreleme de öğenin özelliklerine göre öneri sunulduđu için böyle bir sorun ile karşılaşılmaz.

2.4.2 Seyrek veri problemi

Belirli bir öğe veya kullanıcı için daha az derecelendirme olduđu anlamına gelir. Kullanıcılar tüm öğeler hakkında yorum yapamaz, bu nedenle büyük miktarda öğe asla değerlendirilmez [25].

İF yönteminde kullanıcı ve öğe değerlendirmeleri ve İçerik bazlı filtreleme de öğe özellikleri ne kadar fazla ise doğru öneride bulunma ihtimalleri daha yüksek olacaktır. Veri miktarı ne kadar seyrek ise algoritmalar yeterli seviyede veri ihtiyacı karşılanmayacağında öneri sisteminin güvenilir ve doğru önerilerde bulunma ihtimali azalacaktır.

2.4.3 Gri koyun problemi

İF ’de bir kullanıcının hiçbir kullanıcı ile ortak bir grupta bulunmaması problemidir. Aynı değerlendirmeleri bulunan kullanıcıları bir grupta bir araya getirmek kolaydır.

Yeterince deęerlendirmesi bulunmayan kullanıcılar olabilir. Bu tür kullanıcıları gri koyun kullanıcılar denir. Böyle bir durumda İF doğru önerilerde bulunması mümkün olmayabilir.

2.4.4 Ölçeklenebilirlik problemi

Öneri sistemleri, aynı anda birden çok kullanıcıya hizmet vermeyi hedefler. Aynı anda binlerce kullanıcıya hizmet vermesine ek olarak sistemde milyonları aşkın kullanıcı ve öęe bulunabilir. Böyle bir sistemin verimli çalışabilmesi için sistemi bu yükü taşıyabilecek donanım ve yazılımla çalıştırılması gerekir. Ölçeklenebilirlik, yoğun işlemlerin olduęu dięer tüm sistemler gibi öneri sistemleri için de zorlayıcıdır [31].

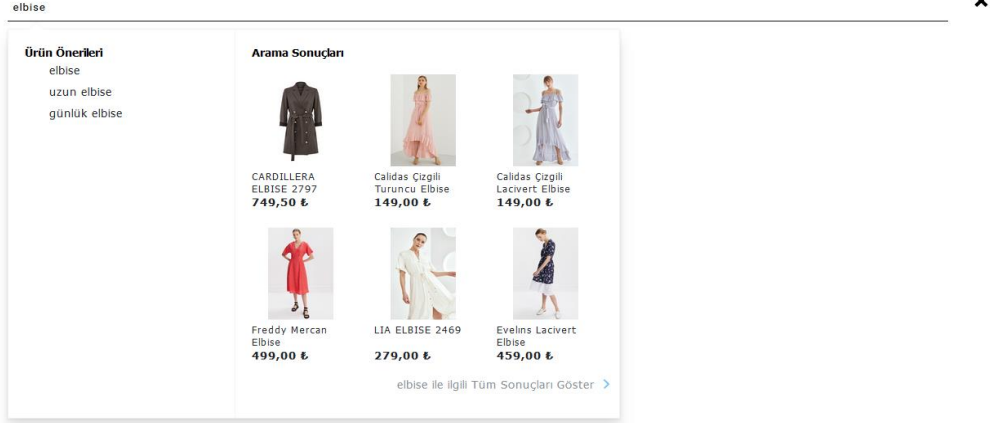
2.5 E-Ticarette Öneri Sistemleri

Günümüzde e-ticaret siteleri öneri sistemlerinin başlıca uygulama alanlarından biridir. E-ticaret sitelerinin doğru kullanıcıya doğru ürünü önerme hedefleri satış rakamlarının artmasında önemli yer edinmektedir [32].

Popüler e-pazar yerlerinin yanı sıra çoęu e-ticaret sitesi öneri sistemlerini kullanmaya başlamıştır. Öneri sisteminin geliştirilme ve yönetilme maliyeti orta ve küçük ölçekli firmalar için bir sorun oluştursa da bu gruptaki firmalar bu hizmeti 3. parti yazılım olarak farklı firmalardan hizmet alarak sistemlerine entegre etmektedirler.

2.5.1 Uygulanan alanları

- Arama çubuęu üzerinde yazılan kelime uygun ürün ve arama kelimelerinin önerilmesi, Şekil 2.3 'de 'www.perspective.com.tr' e-ticaret sitesinden kullanım örneęi görölmektedir.



Şekil 2.3 : perspective.com.tr arama çubuğu önerileri.

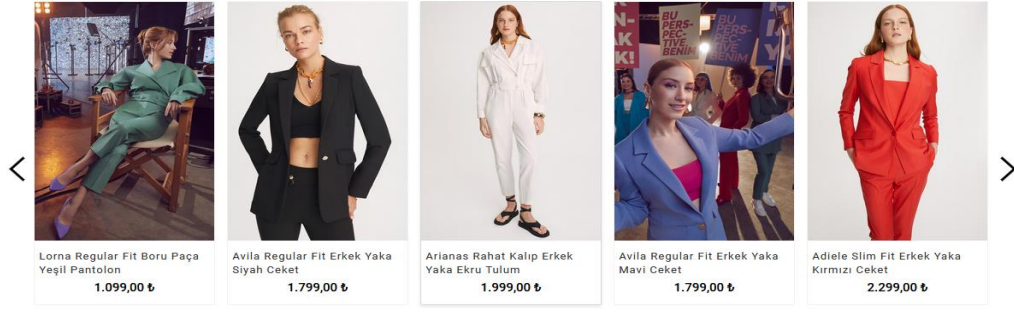
- Kullanıcının geçmiş tercihlerine göre açılır pencere ile ürün önerilmesi, Şekil 2.4’de ‘www.oxxo.com.tr’ e-ticaret sitesinden kullanım örneği görülmektedir.



Şekil 2.4 : oxxo.com.tr açılır pencere önerileri.

- Görüntülenen katalog/kategori sayfasının öneri ürünleri, Şekil 2.5’de ‘www.perspective.com.tr’ e-ticaret sitesinden kullanım örneği görülmektedir.

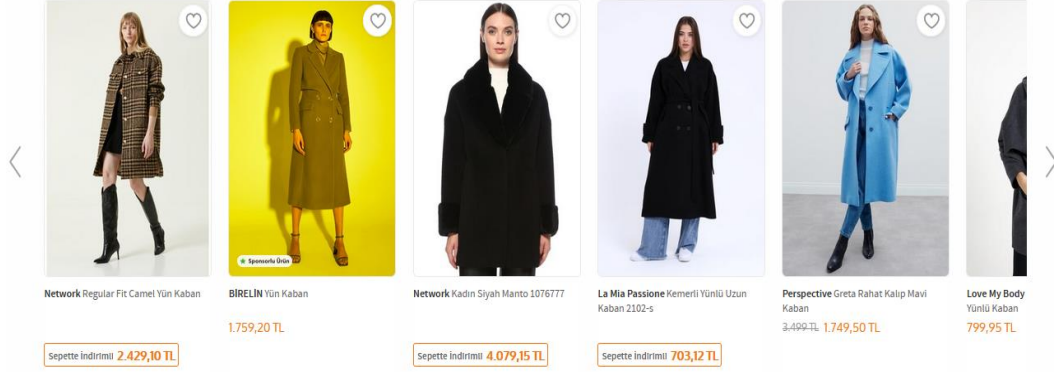
Kategorinin Öne Çıkanları



Şekil 2.5 : perspective.com.tr katalog/kategori önerileri.

- Birlikte alınan ürün önerileri, Şekil 2.6'de 'www.trendyol.com' e-ticaret sitesinden kullanım örneği görülmektedir.

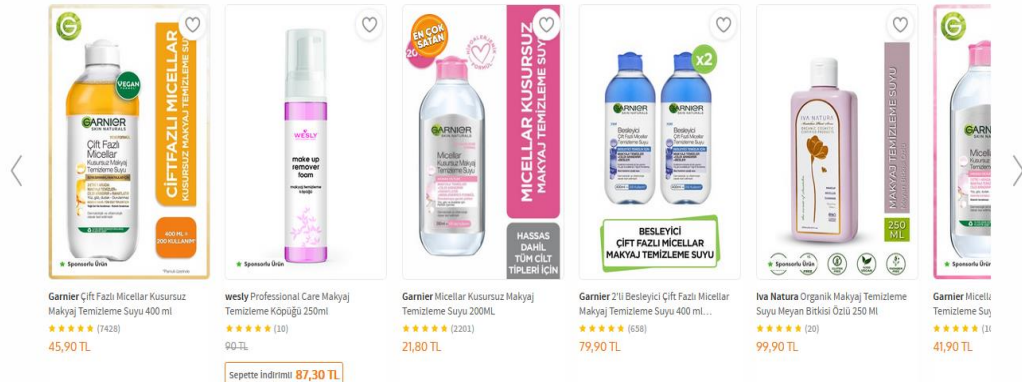
Bu Ürünü Alanlar Bunları da Aldı



Şekil 2.6 : trendyol.com birlikte alınan ürün önerileri.

- Benzer ürünler, Şekil 2.7'de 'www.trendyol.com' e-ticaret sitesinden kullanım örneği görülmektedir.

Benzer Ürünler



Şekil 2.7 : trendyol.com benzer ürünler.

Uygulanan bu yöntemler ve benzeri yöntemler ile aktif kullanıcının sistem üzerinde gezinirken aradığı öğelere ulaşmasını kolaylaştırarak ve beğenebileceği farklı ürünleri önererek, hizmet kalitesinin ve satış rakamlarının arttırabilirler.

2.6 Çalışmanın Amacı ve Önemi

Teknolojinin gelişmesiyle birlikte online alışveriş her geçen gün daha da yaygınlaşmaktadır. Fiziksel satış mağazalarının aksine online alışverişte fiziksel mağaza bulunmaması sebebi ile maliyet daha düşüktür. Bununla beraber herhangi bir mesafe sınırı bulunmadan tüm müşteri gruplarına hizmet oldukça basittir. Online alışverişin getirdiği sayısız avantajlar firmaların bu sektöre yönelmesini sağlamıştır.

Bu çalışmada da öneri sistemleri incelenmiş ve e-ticaret sitelerine nasıl uygulanacağı sorusuna cevap aranmıştır. Çalışma da kullanılacak veriler Kullanıcı aktivite verileri ve Sistem verileri olarak iki grupta incelenmiştir. Her iki gruptaki verilerin toplanması için uygulamalar geliştirilmiştir. Bu uygulamalar proje analizleri ve proje geliştirirken karşılaşılan problemlere yönelik çözümler sunulmuştur.

Çevrim içi alışveriş sitelerinden verilerin toplanması, toplanan verilerin öneri sistemi yöntemleri ile kullanılarak faydalı bilgiye dönüştürülmesi ve bu bilgilerin çevrim içi alışveriş sitesinde kullanılması bir bütün olarak ele alınmıştır. Öneri sistemi yöntemleri çevrim içi alışveriş sitesi üzerinde nasıl uygulanabilir sorusuna cevap aranarak tüm bu süreç adım adım işlenip mevcut bilginin faydalı bilgiye dönüştürülme süreci incelenmiştir.s

3. LİTERATÜR ARAŞTIRMASI

Şahin Güler 2019 yılında tez çalışmasında kullanılan öneri sitemlerini ve E-Ticarete öneri sistemlerinin kullanımını incelemiştir. 14.400 film verisini kullanarak İçerik bazlı İşbirlikçi ve Hibrit öneri sistemi uygulayarak performans kıyaslaması yapmıştır. Hibrit öneri sisteminin kullandığı veriler üzerinde daha iyi performans gösterdiği sonucuna varmıştır [30].

Metehan Uçar 2021 yılında tez çalışmasında Bulanık mantık ve makine öğrenmesi tabanlı bir model önermiştir. Amazon Reviews ve GoodReads veri setlerini kullanmıştır. Çok sayıda değişken ve veri içeren veri setlerinde oluşturulan bulanık modellerde karar ağacı yönteminin kuralları hızlı ve doğru bir şekilde oluşturabileceğini belirtmiştir [32].

Yusuf Özdemir 2021 yılında tez çalışmasında Kullanıcı ve öge temelli işbirlikçi filtrelerin birlikte kullanıp, kullanıcı özelliklerinden de faydalanan bir hibrit öneri sistemi modellenmiştir. MovieLens veri setini kullanarak 6 farklı yöntem ile oy tahmini yapılmıştır. Yapılan çalışma sonuçlar klasik yapıya oran ile daha iyi sonuçlar verilmiştir [33].

2010 yılında Moreno ve ark. kullanıcılar için TV programları önerileri sağlayan bir sistem sunmuştur. Metot olarak içerik filtreleme tekniği ve işbirlikçi filtrelemeyi birleştiren hibrit bir yaklaşım sunulmuştur. Çalışmada belirlenen amaç kullanıcıların izleyeceği programı seçmesini olabildiğince basite indirgemektir [34].

Lian 2011 yılında e-ticarete kişiselleştirilmiş web sayfası üzerine bir çalışma gerçekleştirmiştir. Çalışmada kullanıcıların etkileşimleri, arama verileri, sunucu verileri ve kullanıcıların yapmış olduğu değerlendirmelerin verileri alınarak kümeleme algoritması ile değerlendirmeler yapılmıştır. Öneri sistemi yöntemi olarak işbirlikçi filtreleme tekniği kullanılmıştır [35].

2013 yılında Zhang ve ark. mobil ürünleri ve hizmetleri için ve bulanık mantık tabanlı kişiselleştirilmiş hibrit bir yaklaşım sunmuşlardır. Çalışmada kullanıcı ve ürün tabanlı filtreleme teknikleri ile bulanık kümeleme teknikleri kullanılmıştır. Önerilerdeki

doğruluğu ve performansı artırmak için verilerdeki belirsizlikleri azaltmak için çalışılmıştır. Fuzzy-based Telecom Product Recommender System (FTCP RS) yazılımı üzerinden yapılan çalışma sonucunda kullanıcılara sunulan önerilerin daha iyi sonuçlar verdiği belirtilmiştir [36].

2018 yılında Hyunwoo Hwangbo ve ark. Tipik ürün tabanlı işbirlikçi filtreleme algoritmasını genişleten, K-RecSys adlı yeni bir sistem önerdi. K-RecSys, müşterilerin çevrimiçi ve çevrimdışı tercihlerini yansıtmak için ağırlıklandırılmış çevrimiçi ürün tıklama verilerini ve çevrimdışı ürün satış verilerini birleştirir [37].

2015 yılında Faryal Ali ve ark. Öneri yapmak için işbirlikçi ve içerik tabanlı filtreleme kullanarak bir kullanıcı tavsiye sistemi tanıttı. Sosyal ağ sitelerinin veri ayıklanması, soğuk başlangıç ve aşırı kişiselleştirme sorunlarına çözümler önermiştir. Sonuçlar, sırasıyla bilgi alma matrisleri Precision, Fallout ve F1 Skoru kullanarak değerlendirdi [38].

Elif Şafak Sivri 2015 yılında Veri Madenciliği/E-Ticaret Sitesi İçin Ürün Tavsiye Sistemi Geliştirilmesi isimli tez çalışmasında birliktelik analizi uygulamıştır. Bu çalışmada bir e-ticaret sitesini ait giyim alışveriş verilerini kullanmıştır. FP-Growth ve Apriori algoritmalarını kullanmış ve her biri için ayrı kurallar bulmuştur. Cinsiyet ve yaş değişkenlerini kullanarak k-means ile kümeleme analizi uygulamıştır. Benzerliklerin tespiti için Jaccard, Cosine ve Pearson algoritmaların kullanmıştır [50].

Waleed Abdullah 2019 yılında E-Ticaret İçin Ürün Tavsiye Sisteminin Geliştirilmesi isimli tez çalışmasında market sepet analizi ve işbirlikçi filtreleme tekniklerini uygulamıştır. Uygulama için inşaat sektöründe perakende hizmeti veren özel bir firmanın 3 yıllık satış verilerini kullanmıştır. Çalışmasında Apriori, FP-Growth algoritmasını ve sık örüntü madenciliği üzerine değerlendirmelerde bulunmuştur. Benzerlik hesaplaması için Kosinüs, Pearson ve Jaccard kullanmış ve Jaccard benzerliği ile FP-Growth algoritmasının en iyi performansı verdiğini belirtmiştir [51].

2012 yılında Fidan Kaya İşbirlikçi Filtrelemeye Dayalı Web Tabanlı Öneri Sistemi Geliştirilmesi isimli tez çalışmasında ‘Dekorasist’ isimli bir öneri sistemi geliştirmiştir. Öneri sunma algoritmasında ilk olarak Doğrusal Eşleştirme algoritması uygulanmış, daha sonra işbirlikçi Filtreleme algoritmasını uygulamıştır. Doğrusal Eşleştirme algoritması ile yapılan ön filtreleme işlemi klasik İşbirlikçi Filtreleme Yöntemi’nin performansını arttırmış ve yöntem sonucunda elde edilen değerler ile

gerçek değerler arasında %0,0035 oranında farklılık tespit edilmiştir. Geliştirilen modelin gerçeğe yakın sonuçlar ürettiği ve İşbirlikçi Filtreleme Yöntemi'nin tasarlanan öneri sistemine uygulanabilirliği gösterilmiştir [52].

Miyahara ve Pazzani hem kullanıcı hem de öge tabanlı işbirlikçi filtreleme algoritması üzerinde Bayesian sınıflandırıcılarını kullanan bir yöntem geliştirmiştir. Biri kullanıcıların benzerliklerine dayalı tahminler yapan kullanıcı tabanlı filtreleme, diğeri öğelerin benzerliklerine dayalı tahminler yapan öge tabanlı işbirlikçi filtrelemedir. Önerilen Bayes yaklaşımları tipik korelasyon tabanlı işbirlikçi filtreleme algoritmalarından daha iyi performans gösterdiğini göstermişlerdir. Ayrıca öneri performansı iyileştirmek için kullanıcı tabanlı ve öge tabanlı işbirlikçi filtrelemeyi Simple Bayesian Classifier ile birleştiren bir model önerilmiş, bu yöntemin tek işbirlikçi öneri yönteminden daha iyi performans gösterdiğini göstermişlerdir [53].

1999 yılında Jon Herlocker ve ark. İşbirlikçi Filtreleme yöntemlerinin tahmin performansını arttırmak için Framework geliştirmiştir. Farklı veri setleri üzerinden Spearman bağıntısının Pearson bağıntısından daha iyi sonuçlar verdiğini gözlemlemişlerdir [54].

Rogh Hu ve Yansheng kullanıcı ve öge tabanlı İşbirlikçi filtreleme yöntemlerini birleştirerek bir model önermişlerdir. Geliştirmiş oldukları hibrit modele HSPA adını vermişlerdir. Veri yumuşatma için öge tabanlı, veri seyrekliği ve tahmin doğruluğu için hem kullanıcı hem öğelerin özelliklerine dayalı tahmin modeli oluşturur. Gerçek veri kümelerinden elde edilen deneysel sonuçlar, HSPA'nın seyrek veriler üzerine tahminin iyileştirilmesine etkin bir şekilde katkı sağladığını göstermişlerdir [55].

Anıl Utku ve Muhammet Ali Akçayol değişken kullanıcı tercihlerine göre öneri sunma yaklaşımlarını adaptif olarak geliştiren ve kullanıcı tercihlerini öğrenebilen tavsiye sistemleri üzerinde incelemelerde bulunmuşlardır. Klasik yöntemler veri madenciliği, makine öğrenmesi ve yapay zekâ teknikleri kullanılarak hibritleştirilmiştir. Hibrit yöntemlerin daha yüksek başarı elde ettiğini ortaya koymuşlardır [56].

Öneri sistemleri literatürde kendine birçok alanda yer bulmuştur. Çevrim içi alışveriş bu alanların içinde rağbet görenlerden biridir. Bir çok çalışmada öneri sistemleri açıklanmış ve öneri türleri için farklı veri setleri örnekler mevcuttur. Çalışmalar öneri sistemi yöntemlerinin, çalışmamamızın yapıldığı alan özelinde performans

kıyaslamaları üzerinde yoğunlaşmıştır. Model performansı arttırmak için hibrit yöntemler üzerinde farklı fikirler ortaya atılmıştır. Bunun yanı sıra araştırmamızın konusu özelinde veri madenciliği yöntemlerinden birliktelik analizi ve rfm analizi sıklıkla literatürde yer edinmiş konulardandır.

Araştırmamızda çevrim içi alışverişte sitelerinde öneri sistemi uygulanması özelinde yoğunlaşmıştır. Araştırmamızda kullanılacak veri seti için Türkiye'de hizmet veren özel bir kadın giyim sektöründe faaliyet gösteren bir firmanın çevrim içi alışveriş sitesinin verileri kullanılmıştır. Literatürü takiben öneri sistemleri incelenmiş, araştırma konusu özelinde kendinde literatürde yer bulan veri madenciliği yöntemleri incelenmiştir. Çevrim içi alışveriş sitelerinden, öneri sisteminin kullanılacağı verilerin toplanması ve depolanma süreçleri incelenmiştir. Verilerin toplanması için 2 ayrı uygulama geliştirilmiş ve uygulama geliştirilirken öngörülen problemlere yönelik önerilerde bulunulmuştur.

Öneri sistemlerin kullanılan yöntemler toplanan veriler ile uygulama örnekleri gerçekleştirilmiş ve araştırmamızın konusu özelinde uygulanabilirliği değerlendirilmiştir. Öneri sisteminin bir çevrimiçi alışveriş sitesi ile birlikte çalışması bir bütün olarak incelenip sırası ile aşağıdaki adımları takip etmiştir.

- Çevrim içi alışveriş sitesinden verilerin toplanması ve depolanması
- Depolanan verilerin öneri sistemi yöntemleri ile kullanılıp faydalı bilginin üretilmesi
- Üretilen bilgilerin çevrim içi alışveriş sitesi ile kullanıcıya sunulması

Takip adım her adım için uygulama örnekleri ve analizleri incelenmiş olup, bu süreçler için önerilerde bulunulmuştur. Çevrim içi alışveriş deneyimini arttırmayı hedefleyen bu süreçlerin çevrim içi alışveriş siteleri ile çalışması kapsamlı bir şekilde incelenerek literatüre katkıda bulunmak hedeflenmiştir.

4. MATERYAL YÖNTEM

4.1 Veri Seti

Çalışmada Hazır Giyim sektöründe hizmet veren E-Ticaret siteleri için öneri sistemi geliştirmek istenmektedir. Bu neden ile hali hazırda E-Ticaret hizmeti veren birkaç özel giyim firmasının verileri kullanılacaktır. Kullanılacak verinin çeşitliliği ve güncelliği geliştirilen algoritmanın performansını etkileyeceği için, iki farklı başlık altında incelenmiştir.

4.2 Sistem Verileri

Sistem verileri e-ticaret sitesine ait ürün, kategori, üye vb. bilgileri barındıran, sistemin içerdiği temel verileri kapsamaktadır. Önerilen ürünlerin güncelliği önemlidir. Stoğu biten, satışa kapanmış ürünlerin kullanıcılara önerilmesi hata oluşturacaktır. Öneri algoritmalarının güncel veriler ile beslenmesi, veri akışının anlık olarak sağlanmasını yanlış ve/veya hatalı ürün önermenin önüne geçecektir.

Sürekli bir veri akışı oluşması, E-Ticaret sistemine ve/veya veri kaynağına yük oluşturulabilir. Sistem performansını ve müşteri deneyimini olumsuz etkilemekte kaçınılmalıdır. Bu ve benzeri problemlerin yaşanmaması için öneri sistemini besleyen veri kaynağı sistem veri kaynağından ayrı tutulmalı ve güncel olmalıdır.

Verilerin güncel tutulması için yaygın olarak zamanlanmış görevler kullanılmaktadır. Sistem üzerindeki yoğunluk göz önünde bulundurularak 1-2 saat veya daha kısa zaman aralıklarında sistem verileri güncellenmektedir. Her güncelleme işleminde toplu aktarım yapılmasından kaçınılmalı sistemde son güncelleme tarihinden sonra değişen/güncellenen verilerin aktarımı yapılmalıdır.

Bu hedef doğrultusunda uygulama geliştirilmiştir. Aşağıdaki bölüm de geliştiren uygulama anlatılmaktadır.

4.2.1 Proje tasarımı

Uygulama birden çok kullanıcıya/firmaya aynı anda yanıt verebilecek altyapıda tasarlanmıştır. Bu uygulama son kullanıcı verileri ile değil, E-Ticaret sisteminde kayıtlı olan ürün, müşteri, satış vb. bilgilerin aktarımı için kullanılması planlanmıştır. Web sitesi üzerinden geliştirilen uygulamaya API aracılığı ile erişilebilir. Her bir modül için veri örnekleri paylaşılmıştır.

4.2.1.1 İstemciler

Geliştirilen uygulama HTTP protokolünü benimser ve ağ üzerinden hizmet verir. HTTP ile haberleşen herhangi bir uygulama ile sisteme erişilebilir. E-Ticaret sistemleri Web sitesi ve mobil uygulamalar ile son kullanıcıya hizmet vermektedir.

- **Web uygulamaları**

Web uygulaması üzerinde kullanıcıların görüntülediği, sepetine eklediği veya beğendiği ürün bilgileri kayıt altına alınabilir. Kullanıcının web uygulaması üzerinden yapmış olduğu işlemin anlık olarak kaydedilmesi mümkündür.

- **Mobil uygulamalar**

Web uygulamalarında olduğu gibi mobil uygulamalarda da aynı işlemler gerçekleştirilebilir. Kullanıcıların kayıt altına alınmak istenen son aktivite verisi anlık olarak geliştirilen uygulama ile kaydedilebilir.

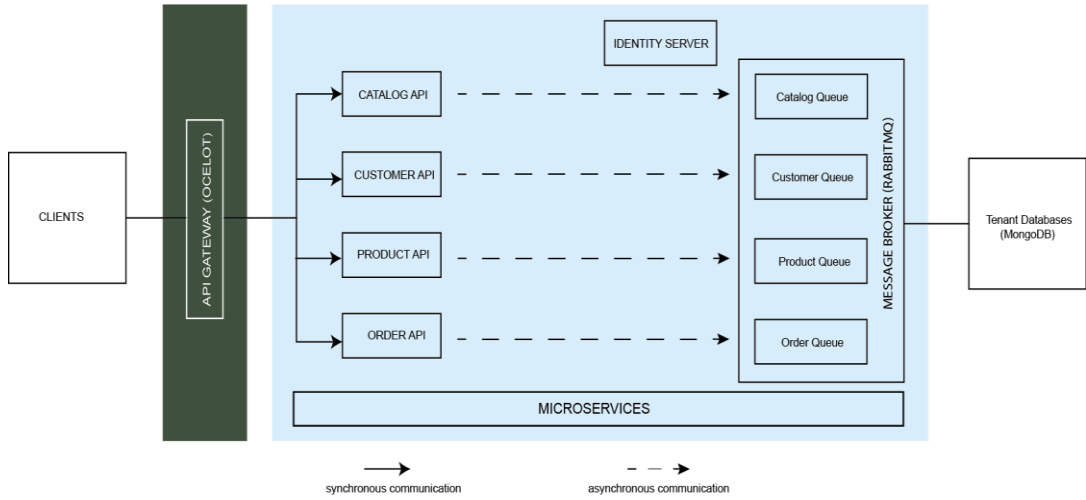
4.2.1.2 Uygulama tasarımı

Uygulama Mikro servis mimarisi ile geliştirilmiştir. Uygulama mimarisi Şekil 4.1’de gösterilmiştir.

4.2.1.3 Mikro servis mimarisi

Servis odaklı mimarinin (SOA) yapısal stiline bir çeşididir. Bir uygulamayı hafif bağlı hizmetler topluluğu olarak düzenler. Hizmetler küçük parçalıdır ve protokolleri hafiftir [50]. Servis odaklı mimariye göre daha küçük, sınırları belirli ve ihtiyaca göre tasarlanmaktadır. Bir diğer farkı ise merkezi mesajlaşma biriminin bulunmamasıdır [51].

Şekil 4.1’de Sistem verilerinin toplanması için geliştirilen uygulamanın mimarisi görülmektedir.



Şekil 4.1 : Sistem verilerinin toplanması için geliştirilen uygulama mimarisi.

E-Ticaret sitelerine ait sitem verilerinin, geliştirilecek öneri sistemi için ayrı bir depolama alanında saklanması ve site üzerindeki anlık gelişmelerin bu depolama alanında da sürekli güncellenmesi amacı ile geliştirilmiş bir Web API projesidir. Bu uygulamanın geliştirilmesindeki temel amaç E-Ticaret sitesi ile Öneri Sisteminin veri depolarını ayırmaktır.

Veri depoları ayrılarak;

- Öneri sisteminin ihtiyaç duyacağı verilerin depolanması, gereksiz veriler ile veri deposunun şişirilmemesi
- Öneri sistemi ile E-Ticaret sitesinin farklı veri depolarından beslenmesi, herhangi bir uygulamadaki aksaklığın bir diğerine etki etmemesi
- E-Ticaret sitesinin performansına etki edilmek istenilmemesi
- Öneri Sisteminin ihtiyaç duyacağı büyük boyutlu verilerin erişim hızı

stabil ve performanslı bir uygulama altyapısı hedeflenmiştir.

4.2.2 Modüller

Bu bölümde uygulamadaki modüllerin açıklamaları veri örnekleri ile açıklanmıştır.

4.2.2.1 Katalog / Kategori modülü

Sistemde kayıtlı olan ürün liste bilgilerini aktarmak için kullanılan modüldür. Geliştirilen uygulamada ‘Catalog API’ aracılığı ile toplanır/güncellenir.

Toplanan veri örneği Çizelge 4.1’de gösterilmiştir.

Çizelge 4.1 : Katalog/Kategori modülü veri örneği.

ID	Name	Url	Parent	Status
1	New Season	/newseason	Null	True
2	Outlet	/outlet	Null	False
3	Best Seller	/bestseller	1	True

4.2.2.2 Müşteri modülü

Sistemde kayıtlı olan müşteri/kullanıcı verilerinin aktarılması için kullanılan modüldür. Geliştirilen uygulamada ‘Customer API’ aracılığı ile toplanır/güncellenir.

Toplanan veri örneği Çizelge 4.2’de gösterilmiştir.

Çizelge 4.2 : Müşteri modülü veri örneği.

ID	Name	Email	Status
1	User Name	user@user.com	True
2	User Name	user@user.com	False

4.2.2.3 Ürün modülü

Sistemde kayıtlı olan ürün verilerinin aktarılması için kullanılan modüldür. Geliştirilen uygulamada Product API aracılığı ile toplanır/güncellenir.

Toplanan veri örneği Çizelge 4.3’te gösterilmiştir.

Çizelge 4.3 : Ürün modülü veri örneği.

ID	Name	ProductCode	Price	Url	Pictures	Catalogs	Available	Attributes
1	Product1	3A312F	136.99TL	/3a312f	Picurls	1,22,67	True	
2	Product2	3N431D	998.99TL	/3a312f	Picurls	3,43,65	True	
3	Product3	6B8SD3	69.99TL	/3a312f	Picurls	8,14,7	True	

4.2.2.4 Sipariş modülü

Sistemde kayıtlı olan sipariş verilerinin aktarılması için kullanılan modüldür. Geliştirilen uygulamada ‘Order API’ aracılığı ile toplanır/güncellenir.

Toplanan veri örneği Çizelge 4.4’te gösterilmiştir.

Çizelge 4.4 : Sipariş modülü veri örneği.

ID	UserID	OrderedItems	TotalAmoun t	TotalDiscount
1	7123	Object	1575.00 TL	0 TL
2	5245	Object	1345.50 TL	114.50 TL

3	36456	Object	235.00 TL	15.00 TL
---	-------	--------	-----------	----------

Siparişe ait ürünler için (OrderedItems) toplanan veri örneği Çizelge 4.5'te gösterilmiştir.

Çizelge 4.5 : Siparişe ait ürün veri örneği.

ID	ProductID	ProductCode	Price	Quantity	Discount	Size	Color	IsFree
1	PR123	3A312F	136.99TL	2	0	S	Blue	False
2	PR643	3N431D	998.99TL	1	9.89 TL	36	Black	False
3	PR432	6B8SD3	69.99TL	1	0	S	Yellow	False

Uygulama teknolojik incelenmesi

Bu bölümde uygulama geliştirilirken tercih edilen araçlar hakkında bilgiler verilmiştir. Bu araçların tercih edilme sebebi ve uygulamadaki rolleri açıklanmıştır.

C# programlama dili

2000 yılında Microsoft'tan Anders Hejlsberg tarafından tasarlandı ve daha sonra 2002'de Ecma ve 2003'te ISO/IEC tarafından uluslararası bir standart olarak onaylandı. C# çok paradigmalı bir programlama dilidir. Statik yazım, güçlü yazım, sözcüksel kapsamlı, zorunlu, bildirimsel, işlevsel, genel, nesne yönelimli (sınıf tabanlı) ve bileşen odaklı programlama disiplinlerini kapsar [44].

Bu çalışmada C# tercih edilmesinin temel nedeni, tüm modülleri tek çatı altında inşa etmek ve .Net Core 'un getirmiş olduğu avantajları kullanarak performanslı, ölçeklenebilir bir servis inşa edilmesidir.

.net Core yazılım çerçevesi

Windows, Linux ve macOS işletim sistemleri için ücretsiz ve açık kaynaklı, yönetilen bir bilgisayar yazılım çerçevesidir [45].

Avantajları;

- Kolay test edilebilir
- Windows, MacOS ve Linux üzerinde geliştirme yapabilme
- Açık kaynak ve topluluk odaklı
- Ortam tabanlı yapılandırma sistemi
- Yerleşik bağımlılıklar
- Basit, yüksek performanslı ve modüler bir http istek işlem hattı

RabbitMq mesaj aracı

Başlangıçta Gelişmiş Mesaj Kuyruklama Protokolünü (AMQP) uygulayan ve o zamandan beri Akışlı Metin Yönelimli Mesajlaşma Protokolünü (Stomp) desteklemek için bir eklenti mimarisiyle genişletilen açık kaynaklı bir mesaj aracı yazılımıdır [46]. Earlang dili ile yazılmış Mozilla Kamu Lisansı altına yayınlanmıştır. Herhangi bir kaynaktan gelen mesajı, başka bir kaynağa sırası geldiği anda iletilmesi için kullanılır. Kullanıcı aktivite verilerin veri depolarına kayıtlarının anlık olarak yapılması şart bir konu değildir. Burada önemli olan kayıt için gelen verinin sıraya alınarak işlemin asenkron bir şekilde yapılması, hem çalışan uygulamanın boş yere bekletilmemesi hem de sunucu üzerindeki maliyeti minimuma indirmesinden dolayı tercih edilmiştir.

MongoDB veri tabanı

MongoDB Inc. Tarafından ölçeklenebilir, doküman tabanlı, C++ ile geliştirilmiş açık kaynak, NoSQL veri tabanı uygulamasıdır. Verileri JSON tipinde dokümanlarda saklamaktadır [47].

- Esnek veriler, veriler belgelerde depolanır. Veri yapısı zaman içinde değişebilir.
- Dağıtılmış veri tabanı, yüksek kullanılabilirlik ve yatay ölçeklendirme
- Ücretsiz, Sunucu Tarafı Kamu Lisansı (SSPL)

Uygulama birden çok kullanıcıya/firmaya eş zamanlı hizmet sunacağı için, her bir kullanıcı/firma verilerinin farklılık gösterme durumu göz önünde bulundurularak doküman tabanlı veri tabanı tercih edilmiştir.

API ağ geçidi (Gateway)

API ağ geçidi, bir istemci ile arka uç hizmetleri koleksiyonu arasında yer alan bir API yönetim aracıdır. API çağrılarını kabul etmek, bunları yerine getirmek için gereken hizmetleri toplamak ve uygun sonucu göndermek temel görevleridir [52].

Uygulamada .Net ile uyumlu Ocelot API Ağ geçidi kullanılmıştır.

Ocelot API ağ geçidi

Sistemlerinde birleşik bir giriş noktası ihtiyacı duyan mikro servisler için .Net yazılım çerçevesinde hizmet veren bir ara katman yazılımıdır. Gelen isteklerin mikro servisler

içerisinde hangi servise yönlendirileceği tanımlanır. Bu sayede tek bir noktadan tüm servislere erişim sağlanmış olur.

Kimlik sunucusu (Identity Server)

Mikro servislere erişimin korunması ve kimlik doğrulaması gerekmektedir. 'IdentityServer', spesifikasyonlarla uyumlu OpenID Connect ve OAuth 2.0 uç noktalarını .Net Core uygulamasına ekleyen bir ara yazılımdır. Standart protokolleri kullanarak istemci uygulamalarının uygulama ile konuşması için protokol başlıklarını ekler [53].

Uygulamada .Net Core altyapısına uygun 'IdentityServer4' ara yazılımı tercih edilmiştir.

4.2.3 Proje gereksinimleri

4.2.3.1 İşlevsel gereksinimler

- Kullanıcı sistemden kullanıcı adı, parolası ve firma özel anahtarı ile 'Token' alabilecektir.
- Kullanıcı sistemdeki her bir modül üzerinden sisteme yeni kayıtlar ekleyebilecektir.
- Kullanıcı sistemdeki her bir modül üzerinden sistemdeki kayıtlarını güncelleyebilir
- Kullanıcı sistemdeki her bir modül üzerinden sistemdeki kayıtlarını silebilir.

4.2.3.2 İşlevsel olmayan gereksinimler

- Sistem bir Web API servisi olarak hizmet verecektir.
- Sistemin bir ara yüzü yoktur.
- Sistem Mikro servis mimarisi ile geliştirilmiştir.
- Sistem 7/24 aktif olacaktır.
- Sisteme ait her bir modül geliştirilebilir ve güncellenebilir.
- Sisteme yeni modül eklenmesi kolay olacaktır.

4.2.4 Proje analizi

4.2.4.1 Veri sözlüğü

Proje raporu içerisinde geçen terimler ve bunların açıklamaları Çizelge 4.6’da Veri Sözlüğünde gösterilmektedir.

Çizelge 4.6 : Sistem verileri veri sözlüğü.

Terim	Açıklama
Use Case	Kullanım senaryosu, sistemlerin fonksiyonel ihtiyaçlarını kapsamak için yazılım ve sistem mühendisliğinde kullanılan bir tekniktir. Aktörlerin sistemle etkileşimini tanımlayan senaryolardır.
API	Uygulama programlama ara yüzü, bir yazılımın bir başka yazılımda tanımlanmış işlevlerini kullanabilmesi için oluşturulmuş bir tanım bütünüdür.
End-point	API üzerinde belirli bir amaç için oluşturulmuş metoda verilen isim
Token	İki yazılım arasındaki veri alışverişi yapılırken kimlik doğrulaması için kullanılan süreli veya süresiz özel anahtarlardır.
Kullanıcı	Sistemi kullanan herhangi bir kimse
Control	Kontrol sınıfı, yapılacak işlem için belirlenen doğrulama/geçerleme işlemleri ile ilgilenen sınıftır.
Object	
API Gateway	İstemcilerle sunucular / mikro servisler arasında duran bir API yönetim aracıdır.

4.2.4.2 İş modeli

Bu bölümde projenin kullanım senaryoları ve aktörleri Çizelge 4.7’te görülmektedir. Devam eden başlıklarda kullanım senaryoları incelenmektedir.

Use Case’ler ve Aktörler;

Çizelge 4.7 : Kullanım senaryoları ve aktörler.

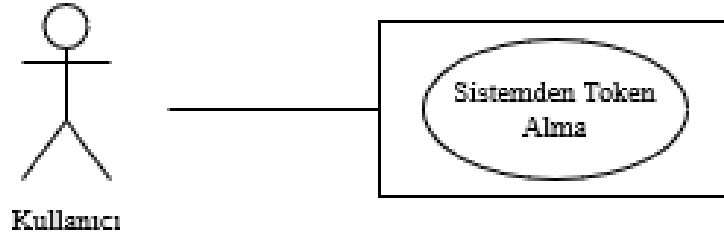
KSİD	Kullanım Senaryosu Adı	Aktörler
USC1	Sitemden Token Alma	Kullanıcı
USC2	Sisteme yeni kayıt ekleme	Kullanıcı
USC3	Sistemdeki kayıtları güncelleme	Kullanıcı
USC4	Sistemden kayıt silme	Kullanıcı

4.2.4.3 Kullanım senaryosu kısa tanımlar

Çizelge 4.7’de kullanım senaryoları ve aktörlerde belirtilen kullanım senaryolarının adım-adım kısa tanımları aşağıdaki Çizelge 4.7’den başlayarak Çizelge 4.11’e kadar KSİD sırasıyla gösterilmektedir.

USC1 Sistemden token alma

Sistemden token alma kullanıcının sisteme erişebilmesi için, sistemin kullanıcıya özel eşsiz bir anahtar üretmesi ve bu üretilen anahtarın kullanıcıya bir sonraki isteklerinde sistem verilerine erişebilmesi için yetkilendirme işlemidir. Şekil 4.2’de Sistemden token alma kullanım senaryosu gösterilmektedir. Çizelge 4.8’de kullanım senaryosunun kısa tanım ve adım adım tanımı açıklanmıştır.



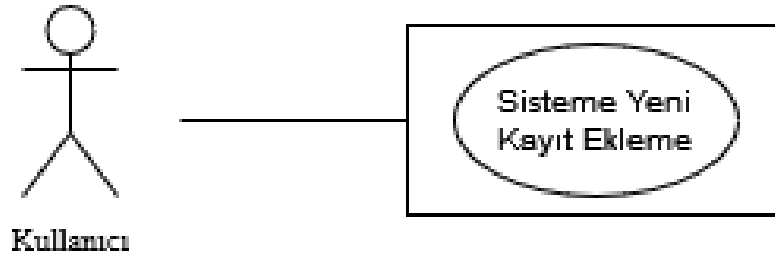
Şekil 4.2 : Sistemden token alma kullanım senaryosu

Çizelge 4.8 : Sistemden token alma kullanım senaryosu

Kısa Tanım:
Token alma Use case’i kullanıcı aktörünün kullanıcı adı, parola ve firma özel anahtar bilgileri ile sistemden token alması ile gerçekleşir.
Adım Adım Tanım:
Kullanıcı ilgili bilgiler ile sistemin token alma end-point’ine istek yapar. Sistem gönderilen bilgileri kayıtlar ile karşılaştırır. Girilen bilgiler doğru ve kullanıcı aktif ise kullanıcıya token bilgisi gönderir.

USC2 Sisteme yeni kayıt ekleme

Kullanıcıların sisteme kayıt ekleyebilmesi için önce USC1 de anlatılan token alma işlemini gerçekleştirmesi gerekmektedir. Alınan bu token ile sisteme yeni kayıt ekleyebilirler. Şekil 4.3’te Sisteme yeni kayıt ekleme kullanım senaryosu gösterilmektedir. Çizelge 4.9’da bu kullanım senaryosunun kısa tanım ve adım adım tanımları açıklanmıştır.



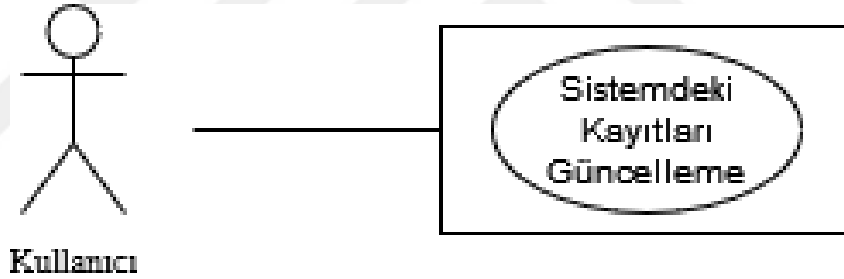
Şekil 4.3: Sisteme yeni kayıt ekleme kullanım senaryosu.

Çizelge 4.9 : Sisteme yeni kayıt ekleme kullanım senaryosu.

Kısa Tanım: Kayıt ekleme Use case'i kullanıcın verileri sisteme kaydetme işlemidir.
Adım Adım Tanım: Kullanıcının almış olduğu token bilgisi ve sisteme eklemek istediği veriler ile kayıt ekleme end-poin'tine istek yapar Sistem token bilgisinin geçerliliğini kontrol eder. Token geçerli ise kullanıcın verilerini kuyruğa gönderir. İşlem sırası geldiğinde kullanıcın veri tabanına kayıtlar eklenir.

USC3 Sistemdeki kayıtları güncelleme

Kullanıcıların sistemdeki kayıtları güncelleme bilmesi için USC1 de anlatılan token alma işlemini gerçekleştirmesi gerekmektedir. Alınan bu token ile sistemdeki kayıtları güncelleyebilirler. Şekil 4.4'te Sistemdeki kayıtları güncelleme kullanım senaryosu gösterilmektedir. Çizelge 4.10'da bu kullanım senaryosunun kısa tanım ve adım adım tanımları açıklanmıştır.



Şekil 4.4 : Sistemdeki kayıtlar güncelleme kullanım senaryosu.

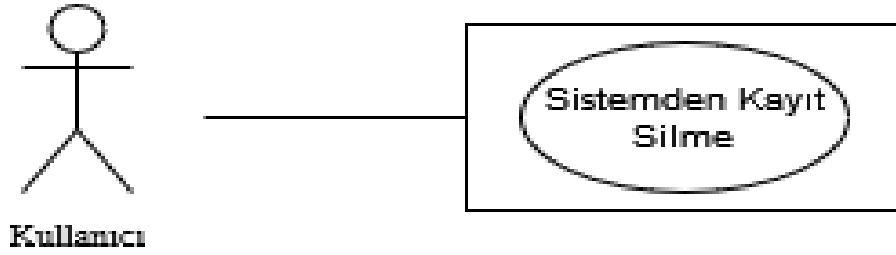
Çizelge 4.10 : Sisteme yeni kayıt güncelleme kullanım senaryosu.

Kısa Tanım: Kayıt güncelleme Use case'i kullanıcın sistemdeki verileri üzerinde değişiklik yapma işlemidir.
Adım Adım Tanım: Kullanıcının almış olduğu token bilgisi ve sisteme eklemek istediği veriler ile kayıt güncelleme end-poin'tine istek yapar Sistem token bilgisinin geçerliliğini kontrol eder. Token geçerli ise kullanıcın verilerini kuyruğa gönderir. İşlem sırası geldiğinde kullanıcın veri tabanında ilgili kayıtlar güncellenir.

USC4 Sistemden kayıt silme

Kullanıcıların sistemdeki kayıtları silebilmesi için USC1 de anlatılan token alma işlemini gerçekleştirmesi gerekmektedir. Alınan bu token ile sistemdeki kayıtları silebilirler. Şekil 4.5'te Sistemdeki kayıtları güncelleme kullanım senaryosu

gösterilmektedir. Çizelge 4.11’da bu kullanım senaryosunun kısa tanım ve adım adım tanımları açıklanmıştır.

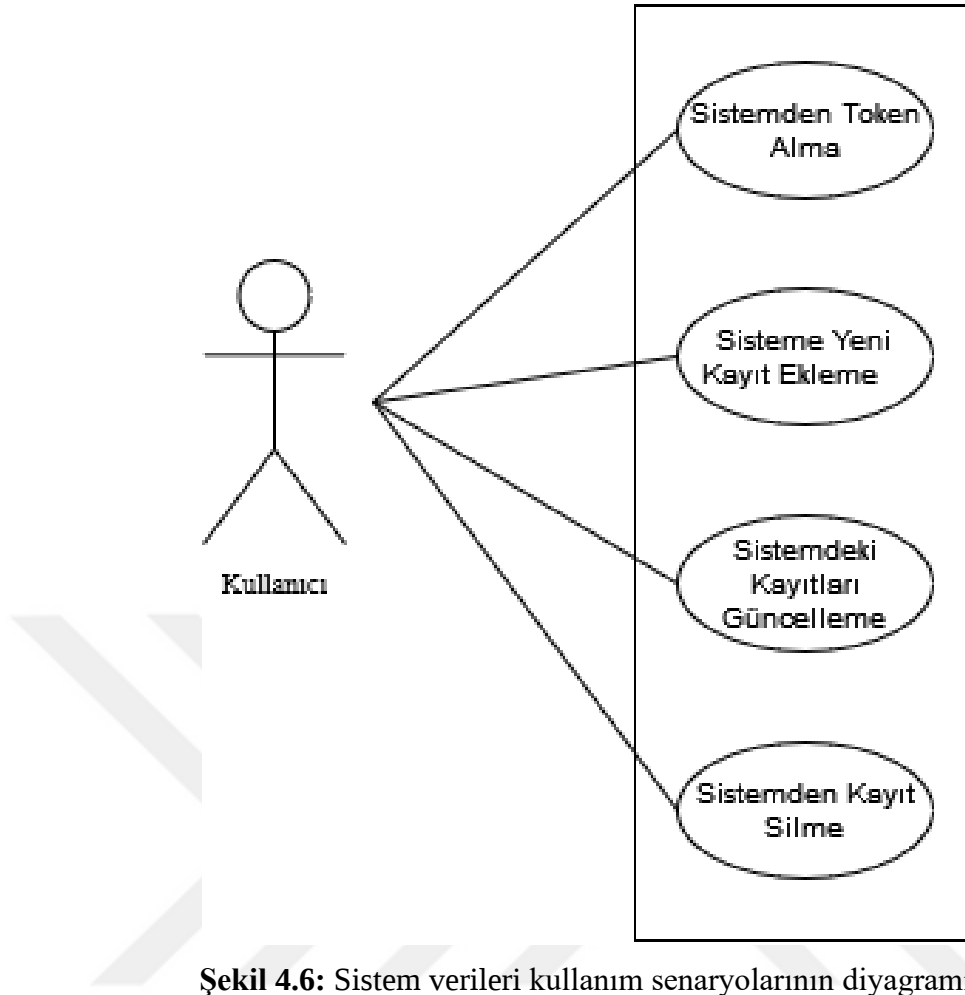


Şekil 4.5 : Sistemdeki kayıtları silme kullanım senaryosu.

Çizelge 4.11 : Sistemden kayıt silme kullanım senaryosu.

Kısa Tanım: Kayıt silme Use case’i kullanıcın sistemdeki verileri sistemden silme işlemidir.
Adım Adım Tanım: 1. Kullanıcının almış olduğu token bilgisi ve sistemden silmek istediği veriler ile kayıt güncelleme end-poin’tine istek yapar 2. Sistem token bilgisinin geçerliliğini kontrol eder. 3. Token geçerli ise kullanıcın verilerini kuyruğa gönderir. 4. İşlem sırası geldiğinde kullanıcın veri tabanında ilgili kayıtlar silinir.

Kullanıcıların sistem verileri için geliştirilen uygulamadaki kullanım senaryoları yukarıda açıklanıştır. Şekil 4.6’da kullanıcının sistem üzerindeki kullanım senaryolarının diyagramı gösterilmiştir.



Şekil 4.6: Sistem verileri kullanım senaryolarının diyagramı.

4.2.3.4 Kullanım senaryoları ve iş birliği diyagramları

Bu bölümde sırası ile her kullanım senaryosu için karşılaşılabilecek normal ve anormal senaryolardan bahsedilmiştir. Bu senaryolar iş birliği diyagramları ile desteklenmiştir.

Sistemden token alma kullanım senaryosu için senaryolar ve iş birliği diyagramları

Çizelge 4.12’de kullanıcının sistemden token alması için normal senaryo örneği ve hatalı bilgi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.7’de ve anormal senaryonun iş birliği diyagramı Şekil 4.8’de gösterilmiştir.

Çizelge 4.12 : Sistemden token alma kullanım senaryosu için senaryo 1.

Normal Senaryo-1:

Kullanıcı kullanıcı adı, parola ve firma özel anahtar bilgileri ile Token alma end-point’ine istek yapar.

1. Sistem gelen bilgileri kayıtlı kullanıcılar ile kıyaslar.

2. Bilgiler eşleşirse sistem kullanıcıya istek başarılı mesajı ile Token bilgisini geri gönderir.

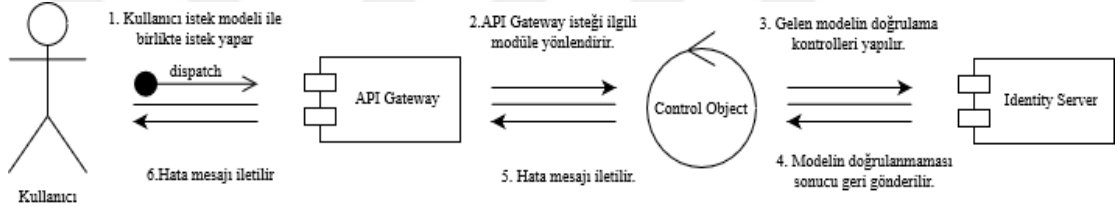
Anormal Senaryo:

Kullanıcı kullanıcı adı, parola ve firma özel anahtar bilgileri ile Token alma endpoint'ine istek yapar.

1. Sistem gelen bilgileri kayıtlı kullanıcılar ile kıyaslar.
2. Bilgiler eşleşmez sistem kullanıcıya istek başarısız mesajı geri gönderir.



Şekil 4.7: Sistemden token alma kullanım senaryosu için normal senaryo İBD 1.



Şekil 4.8 : Sistemden token alma kullanım senaryosu için anormal senaryo İBD 1.

Çizelge 4.13'te kullanıcının sistemden token alması için normal senaryo örneği ve hizmet süresinin dolması ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.9'da ve anormal senaryonun iş birliği diyagramı Şekil 4.10'de gösterilmiştir.

Çizelge 4.13 : Sistemden token alma kullanım senaryosu için senaryo 2.

Normal Senaryo-2:

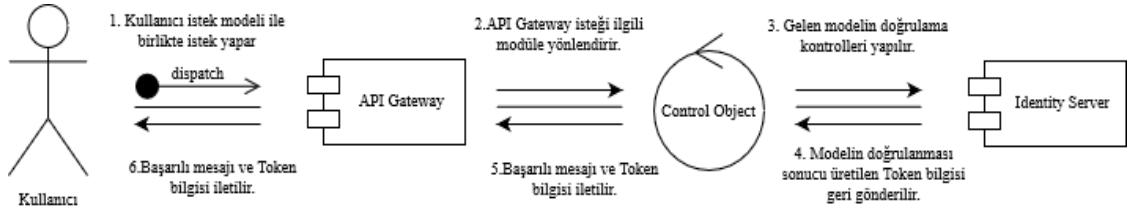
Kullanıcı kullanıcı adı, parola ve firma özel anahtar bilgileri ile Token alma endpoint'ine istek yapar.

1. Sistem gelen bilgileri kayıtlı kullanıcılar ile kıyaslar. Bilgiler eşleşirse sistem kullanıcıya istek başarılı mesajı ile Token bilgisini geri gönderir.

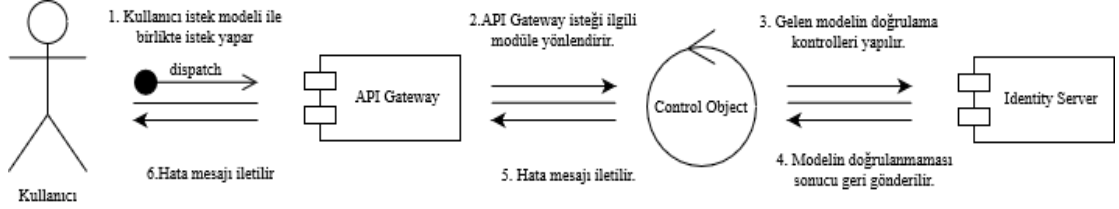
Anormal Senaryo:

Kullanıcı kullanıcı adı, parola ve firma özel anahtar bilgileri ile Token alma endpoint'ine istek yapar.

1. Sistem gelen bilgileri kayıtlı kullanıcılar ile kıyaslar.
2. Bilgiler eşleşir ve firma hizmet süresi doluyorsa hata mesajı geri döner.



Şekil 4.9 : Sistemden token alma kullanım senaryosu için normal senaryo İBD 2.



Şekil 4.10 : Sistemden token alma kullanım senaryosu için normal senaryo İBD 2.

Sisteme yeni kayıt ekleme kullanım senaryosu için senaryolar ve iş birliği diyagramları

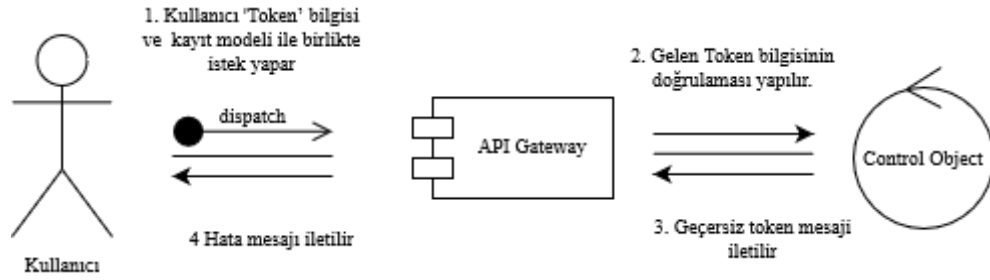
Çizelge 4.14'te kullanıcının sistemden kayıt eklemesi için normal senaryo örneği ve geçersiz token bilgisi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.11'da ve anormal senaryonun iş birliği diyagramı Şekil 4.12'de gösterilmiştir.

Çizelge 4.14 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 1.

Normal Senaryo-1: Kullanıcı 'Token' bilgisi ve kayıt modeli ile istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Kayıt başarılı mesajı geri gönderir.
Anormal Senaryo: Kullanıcı 'Token' bilgisi ve kayıt modeli ile istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder 2. Geçersiz Token bilgisi sebebi ile hata mesajını geri gönderir.



Şekil 4.11 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 1.

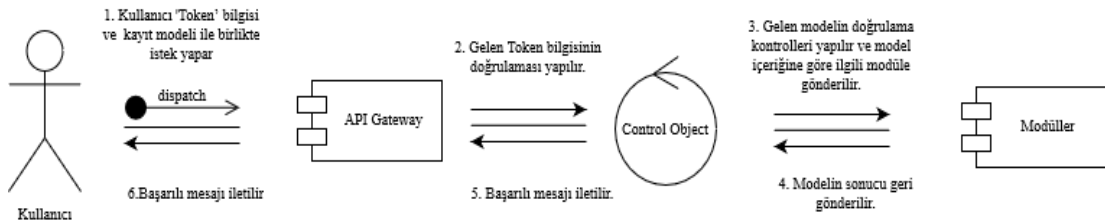


Şekil 4.12 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 1.

Çizelge 4.15’te kullanıcının sistemden kayıt eklemesi için normal senaryo örneği ve hatalı model sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.13’te ve anormal senaryonun iş birliği diyagramı Şekil 4.14’te gösterilmiştir.

Çizelge 4.15 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 2.

Normal Senaryo-2: Kullanıcı ‘Token’ bilgisi ve kayıt modeli ile birlikte istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Kayıt başarılı mesajı geri gönderir.
Anormal Senaryo: Kullanıcı ‘Token’ bilgisi ve kayıt modeli ile birlikte istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Gönderilen Token geçerlidir. 3. Model doğrulama işlemi hatalı olduğu için hata mesajı gönderilir.



Şekil 4.13 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 2.



Şekil 4.14 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 2.

Çizelge 4.16’te kullanıcının sistemden kayıt eklemesi için normal senaryo örneği ve eksik token sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde

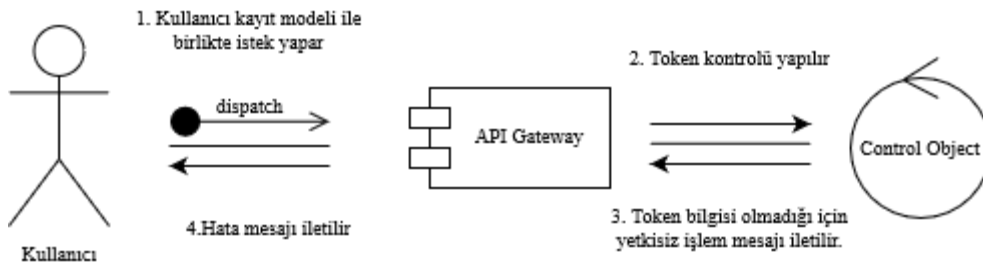
bu normal senaryonun iş birliği diyagramı Şekil 4.15'te ve anormal senaryonun iş birliği diyagramı Şekil 4.16'da gösterilmiştir.

Çizelge 4.16 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 3.

Normal Senaryo-3: Kullanıcı 'Token' bilgisi ve kayıt modeli ile birlikte istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Kayıt başarılı mesajı geri gönderir.
Anormal Senaryo: Kullanıcı 'Token' bilgisi olmadan istek yapar 1. Sistem gelen isteğin token bilgisi olmadığı için geriye yetkisiz işlem mesajı döndürür.



Şekil 4.15 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 3.



Şekil 4.16 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 3.

Çizelge 4.17'te kullanıcının sistemden kayıt eklemesi için normal senaryo örneği ve eksik model sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.17'te ve anormal senaryonun iş birliği diyagramı Şekil 4.18'da gösterilmiştir.

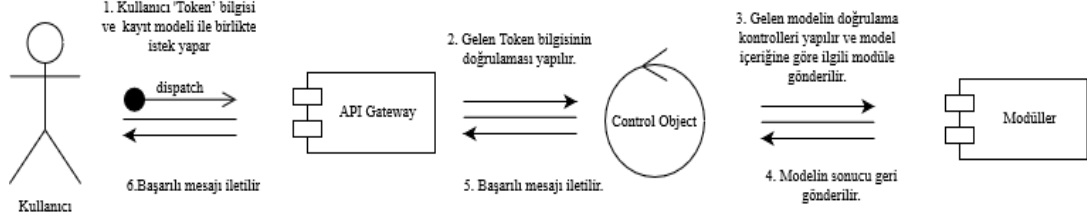
Çizelge 4.17 : Sisteme yeni kayıt ekleme kullanım senaryosu için senaryo 4.

Normal Senaryo-4: Kullanıcı 'Token' bilgisi ve kayıt modeli ile birlikte istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Kayıt başarılı mesajı geri gönderir.
--

Anormal Senaryo:

Kullanıcı kayıt modeli olmadan istek yapar.

1. Sistem gelen isteğin token bilgisini kontrol eder.
2. İsteğe model olmadığı için eksik model hatası geri gönderir.



Şekil 4.17 : Sisteme kayıt ekleme kullanım senaryosu için normal senaryo İBD 4.



Şekil 4.18 : Sisteme kayıt ekleme kullanım senaryosu için anormal senaryo İBD 4.

Sistemdeki kayıtları güncelleme kullanım senaryosu için senaryolar ve iş birliği diyagramları

Çizelge 4.18’te kullanıcının sistemden kayıt güncelleme işlemi yapması için normal senaryo örneği ve hatalı model bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.19’te ve anormal senaryonun iş birliği diyagramı Şekil 4.20’de gösterilmiştir.

Çizelge 4.18 : Sistemdeki kayıtları güncelleme kullanım senaryosu için senaryo 1.

Normal Senaryo-1:

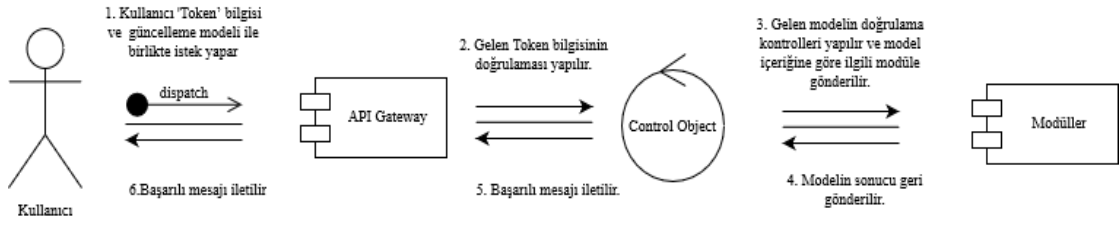
Kullanıcı ‘Token’ bilgisi ve güncelleme modeli ile birlikte istek yapar.

1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.
2. Kayıt başarılı mesajı geri gönderir.

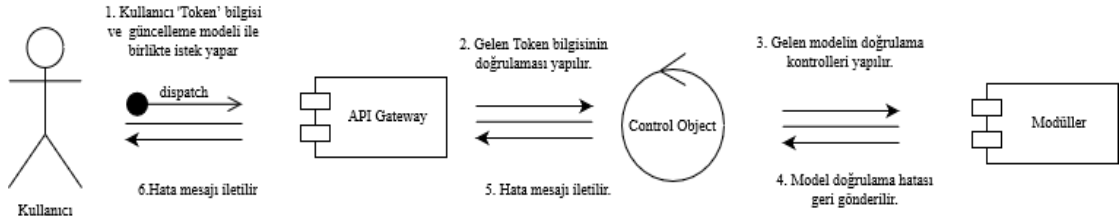
Anormal Senaryo:

Kullanıcı ‘Token’ bilgisi ve güncelleme modeli ile birlikte istek yapar.

1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.
2. Gönderilen Token geçerlidir.
3. Model doğrulama işlemi hatalı olduğu için hata mesajı gönderilir.



Şekil 4.19 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 1.



Şekil 4.20 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 1.

Çizelge 4.19’te kullanıcının sistemden kayıt güncelleme işlemi yapması için normal senaryo örneği ve geçersiz token bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.21’te ve anormal senaryonun iş birliği diyagramı Şekil 4.22’de gösterilmiştir.

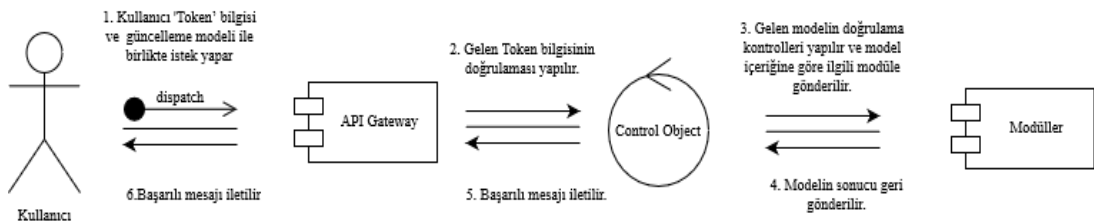
Çizelge 4.19 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 2.

Normal Senaryo-2:

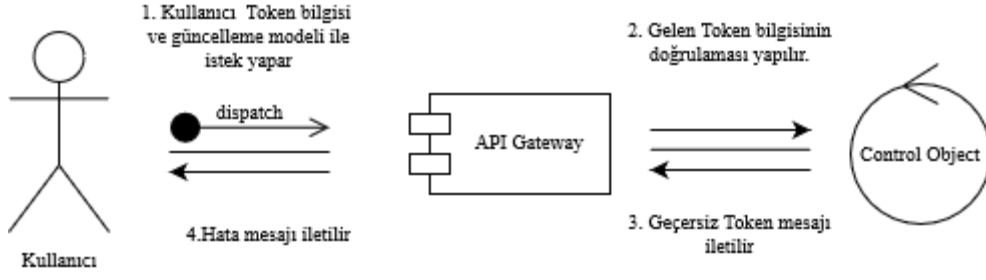
Kullanıcı ‘Token’ bilgisi ve güncelleme modeli ile birlikte istek yapar.
Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.
Kayıt başarılı mesajı geri gönderir.

Anormal Senaryo:

Kullanıcı ‘Token’ bilgisi ve güncelleme modeli ile birlikte istek yapar.
Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. Geçersiz Token bilgisi sebebi ile hata mesajını geri gönderir



Şekil 4.21 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 2.



Şekil 4.22 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 2.

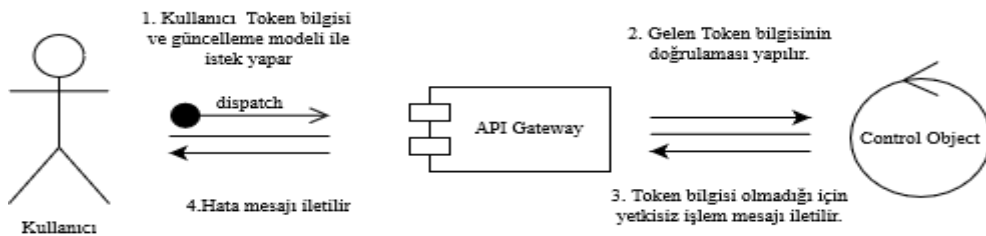
Çizelge 4.20’te kullanıcının sistemden kayıt güncelleme işlemi yapması için normal senaryo örneği ve eksik token bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.23’te ve anormal senaryonun iş birliği diyagramı Şekil 4.24’te gösterilmiştir.

Çizelge 4.20 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 3.

Normal Senaryo-3: Kullanıcı ‘Token’ bilgisi ve güncelleme modeli ile birlikte istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Kayıt başarılı mesajı geri gönderir.
Anormal Senaryo: Kullanıcı ‘Token’ bilgisi olmadan istek yapar 1. Sistem gelen isteğin token bilgisi olmadığı için geriye yetkisiz işlem mesajı döndürür.



Şekil 4.23 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 3.



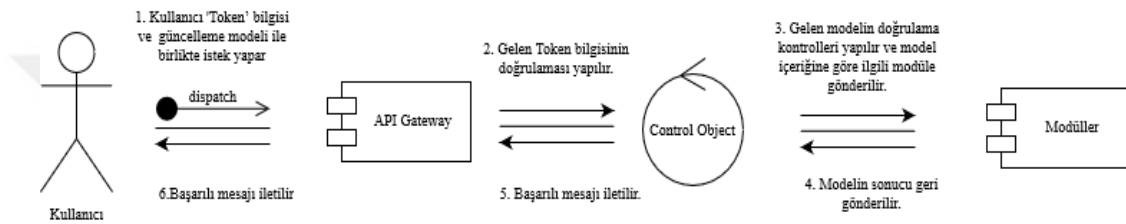
Şekil 4.24 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 3.

Çizelge 4.21’te kullanıcının sistemden kayıt güncelleme işlemi yapması için normal senaryo örneği ve eksik model bilgisi sebebi ile oluşan anormal senaryo örneği

anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.25'te ve anormal senaryonun iş birliği diyagramı Şekil 4.26'da gösterilmiştir.

Çizelge 4.21 : Sitemdeki kayıtları güncelleme kullanım senaryosu için senaryo 3.

Normal Senaryo-4: Kullanıcı 'Token' bilgisi ve güncelleme modeli ile birlikte istek yapar. 3. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 4. Kayıt başarılı mesajı geri gönderir.
Anormal Senaryo: Kullanıcı güncelleme modeli olmadan istek yapar. 1. Sistem gelen isteğin token bilgisini kontrol eder. 2. İsteğe model olmadığı için eksik model hatası geri gönderir.



Şekil 4.25 : Sistemdeki kayıtları güncelleme kullanım senaryosu için normal senaryo İBD 4.



Şekil 4.26 : Sistemdeki kayıtları güncelleme kullanım senaryosu için anormal senaryo İBD 4.

Sistemden kayıt silme kullanım senaryosu için senaryolar ve iş birliği diyagramları

Çizelge 4.22'te kullanıcının sistemdeki kayıtları silme işlemi yapması için normal senaryo örneği ve hatalı model bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.27'te ve anormal senaryonun iş birliği diyagramı Şekil 4.28'da gösterilmiştir.

Çizelge 4.22 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 1.

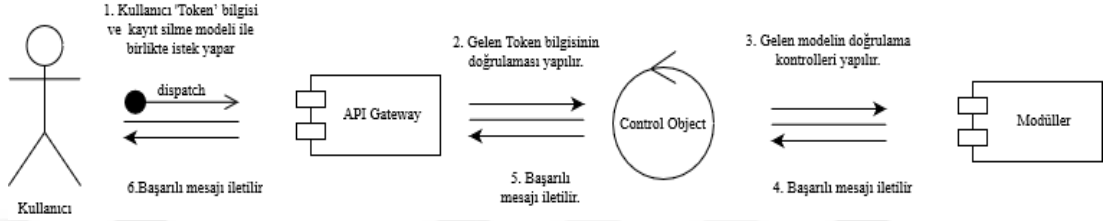
Normal Senaryo-1: Kullanıcı 'Token' bilgisi ve kayıt silme modeli ile birlikte istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.

2. Kayıt başarılı mesajı geri gönderir.

Anormal Senaryo:

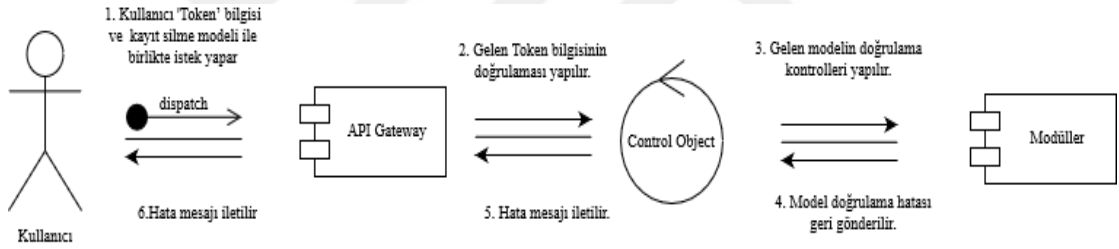
Kullanıcı 'Token' bilgisi ve kayıt silme modeli ile birlikte istek yapar.

1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.
2. Gönderilen Token geçerlidir.
3. Model doğrulama işlemi hatalı olduğu için hata mesajı gönderilir.



Şekil 4.27 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD

1.



Şekil 4.28 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD

1.

Çizelge 4.23'te kullanıcının sistemdeki kayıtları silme işlemi yapması için normal senaryo örneği ve geçersiz token bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.29'te ve anormal senaryonun iş birliği diyagramı Şekil 4.30'da gösterilmiştir.

Çizelge 4.23 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 2.

Normal Senaryo-2:

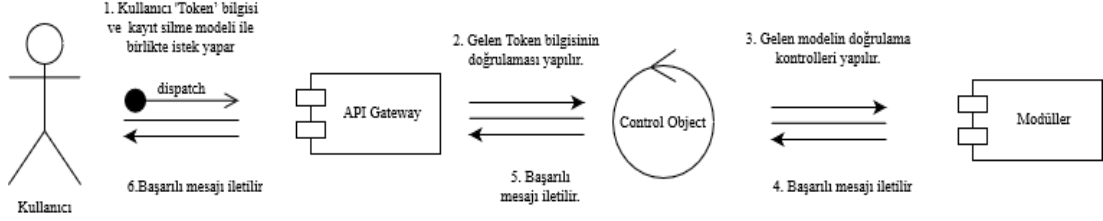
Kullanıcı 'Token' bilgisi ve kayıt silme modeli ile birlikte istek yapar.

1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.
2. Kayıt başarılı mesajı geri gönderir.

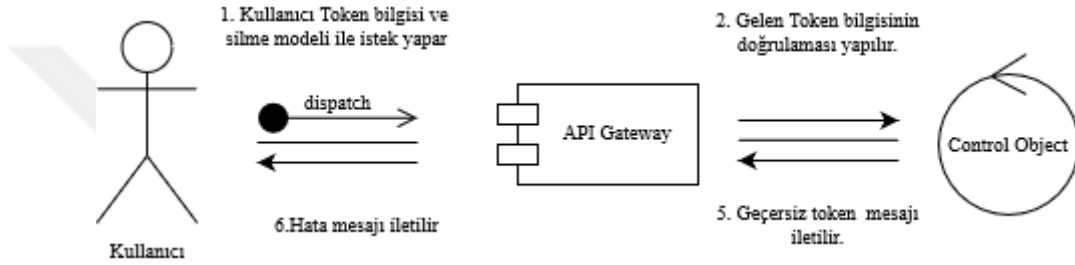
Anormal Senaryo:

Kullanıcı 'Token' bilgisi ve kayıt silme modeli ile birlikte istek yapar.

1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder
2. Geçersiz Token bilgisi sebebi ile hata mesajını geri gönderir.



Şekil 4.29 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD 2.



Şekil 4.30 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD 2.

Çizelge 4.24'te kullanıcının sistemdeki kayıtları silme işlemi yapması için normal senaryo örneği ve eksik token bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.31'te ve anormal senaryonun iş birliği diyagramı Şekil 4.32'da gösterilmiştir.

Çizelge 4.24 : Sistemdeki kayıtları silme kullanım senaryosu için senaryo 3.

Normal Senaryo-3:

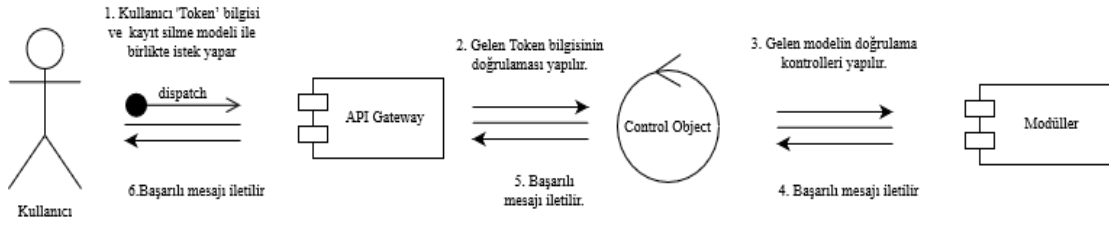
Kullanıcı 'Token' bilgisi ve güncelleme modeli ile birlikte istek yapar.

1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder.
2. Kayıt başarılı mesajı geri gönderir.

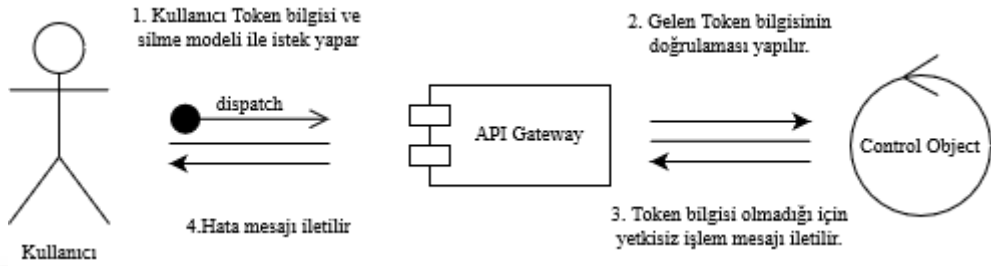
Anormal Senaryo:

Kullanıcı 'Token' bilgisi olmadan istek yapar.

1. Sistem gelen isteğin token bilgisi olmadığı için geriye yetkisiz işlem mesajı döndürür.



Şekil 4.31 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD 3.



Şekil 4.32 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD 3.

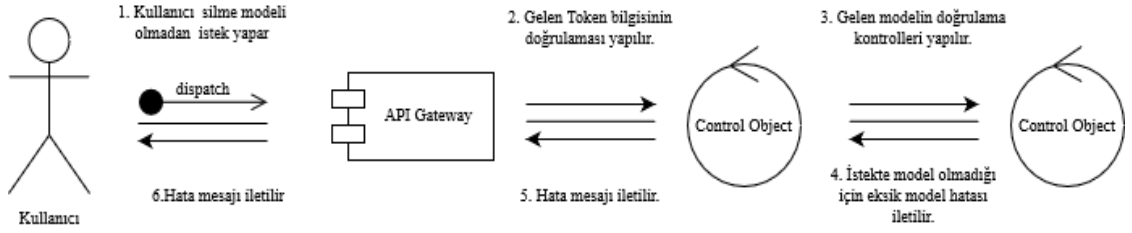
Çizelge 4.25’te ullanıcının sistemdeki kayıtları silme işlemi yapması için normal senaryo örneği ve eksik model bilgisi sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.33’te ve anormal senaryonun iş birliği diyagramı Şekil 4.34’te gösterilmiştir.

Çizelge 4.25 : Sitemdeki kayıtları silme kullanım senaryosu için senaryo 4.

Normal Senaryo-4: Kullanıcı ‘Token’ bilgisi ve güncelleme modeli ile istek yapar. 1. Sistem gelen Token bilgisinin geçerliliğini ve model doğrulama işlemlerini kontrol eder. 2. Kayıt başarılı mesajı geri gönderir.
Anormal Senaryo: Kullanıcı silme modeli olmadan istek yapar. 1. Sistem gelen isteğin token bilgisini kontrol eder. 2. İstekte model olmadığı için eksik model hatası geri gönderir.



Şekil 4.33 : Sistemdeki kayıtları silme kullanım senaryosu için normal senaryo İBD 4.



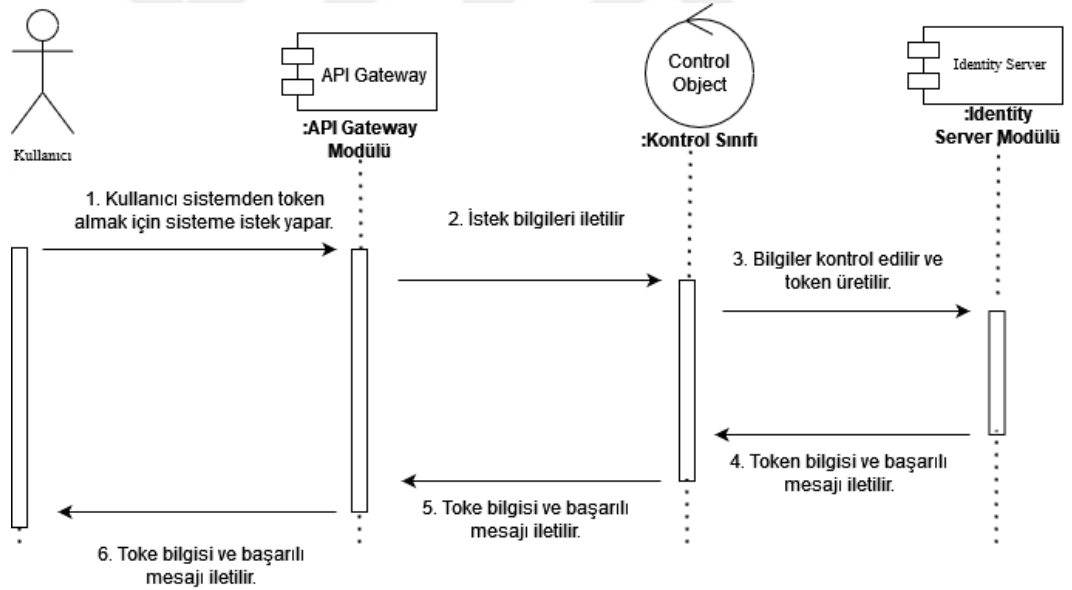
Şekil 4.34 : Sistemdeki kayıtları silme kullanım senaryosu için anormal senaryo İBD

4.2.3.5 Sınıf tasarımları

Bu bölümde sırası ile her bir kullanım senaryosu için sıralama diyagramları oluşturulmuştur. Bu diyagramlar kullanıcı sistemden ilgili senaryo için istekte bulunduğu sistem üzerinde gerçekleşen adımları göstermektedir.

Sistemden token alma kullanım senaryosu sıralama diyagramı

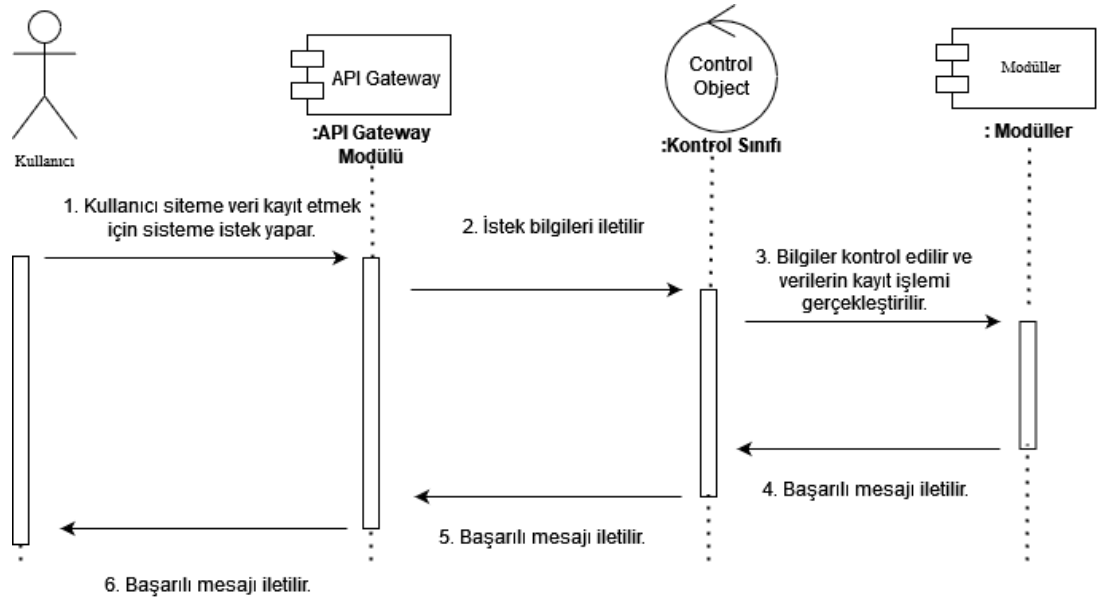
Kullanıcı sistemden token alman için istek yaptığında sistem üzerinde gerçekleşen adımlar Şekil 4.35'te gösterilmiştir.



Şekil 4.35 : Sistemden token alma kullanım senaryosu sıralama diyagramı.

Sisteme kayıt ekleme kullanım senaryosu sıralama diyagramı

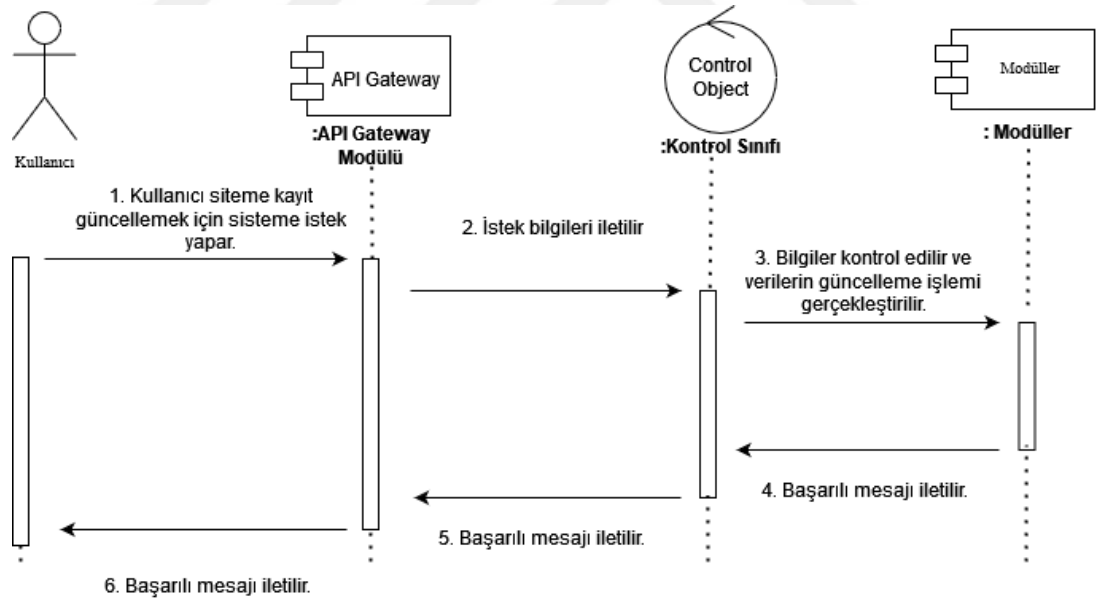
Kullanıcı kayıt eklemek için istek yaptığında sistem üzerinde gerçekleşen adımlar Şekil 4.36'te gösterilmiştir.



Şekil 4.36 : Sisteme kayıt ekleme kullanım senaryosu sıralama diyagramı.

Sistemdeki kayıtları güncelleme kullanım senaryosu sıralama diyagramı

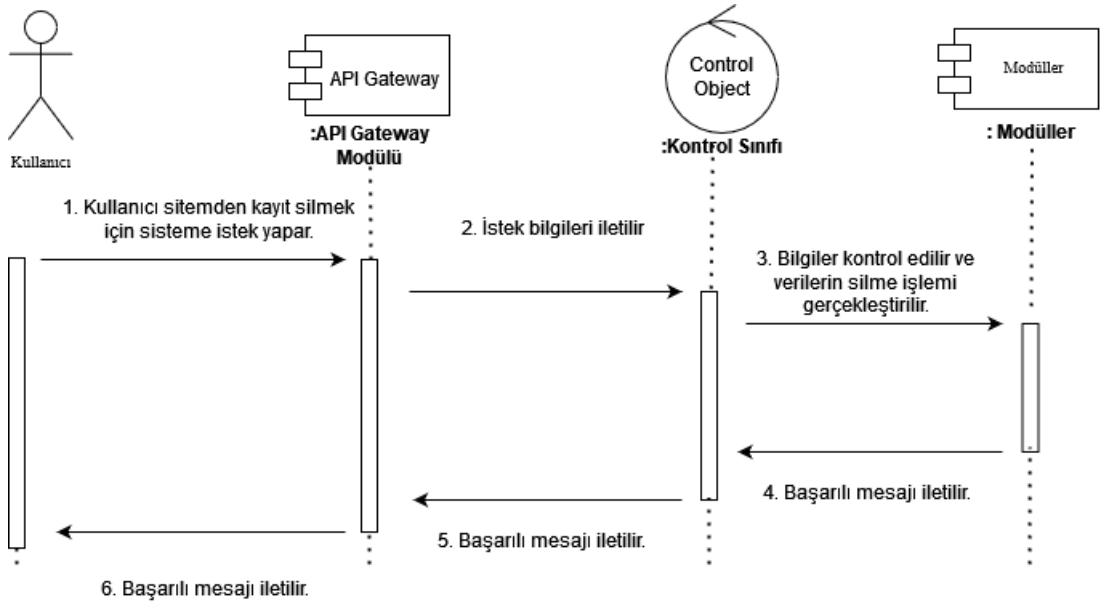
Kullanıcı sistemdeki kayıtları güncellemek için istek yaptığında sistem üzerinde gerçekleşen adımlar Şekil 4.37’te gösterilmiştir.



Şekil 4.37 : Sistemdeki kayıtları güncelleme kullanım senaryosu sıralama diyagramı.

Sistemdeki kayıtları silme kullanım senaryosu sıralama diyagramı

Kullanıcı sistemdeki kayıtları silmek için istek yaptığında sistem üzerinde gerçekleşen adımlar Şekil 4.38’te gösterilmiştir.



Şekil 4.38 : Sistemdeki kayıtları silme kullanım senaryosu sıralama diyagramı.

4.3 Aktivite Verileri

Aktivite verileri kullanıcının ilgilendiği ürün, katalog, renk vb. bilgilerin toplanmasını kapsamaktadır. Kullanıcılar anlık hareketleri kayıt altına alınarak öneri algoritmasının ilgili kullanıcı/kullanıcı segmentine uygun önerilerde bulunması hedeflenmektedir.

Bu hedef doğrultusunda uygulama geliştirilmiştir. Aşağıdaki bölüm de geliştiren uygulama anlatılmaktadır.

4.3.1 Proje tasarımı

E-Ticaret sistemleri son kullanıcıya Web sitesi ve mobil uygulama ile hizmet vermektedir. Geliştirilen proje her iki platform için desteklemesi için Hiper-Metin Transfer Protokolü (HTTP) tercih edilmiş ve *RESTful* mimarisini benimseyen Web API uygulaması geliştirilmiştir. Uygulama için bir önyüz tasarımı yoktur, arka uç olarak hizmet vermektedir. Web API önyüz olmadan tüm işlevleri yerine getirebilecek şekilde tasarlanmıştır. Web uygulamalarının entegrasyonun kolaylaştırılması için javascript dili ile script geliştirilmiştir. Geliştirilen script ile Web uygulamaları üzerinden tek bir fonksiyon ile veri toplama işlemi kolayca gerçekleştirilebilir. Tez içeriğine ek belge olarak örnek script uygulaması eklenmiştir.

4.3.1.1 İstemciler

Geliştirilen uygulama HTTP protokolünü benimser ve ağ üzerinden hizmet verir. HTTP ile haberleşen herhangi bir uygulama ile sisteme erişilebilir. E-Ticaret sistemleri Web sitesi ve mobil uygulamalar ile son kullanıcıya hizmet vermektedir.

- **Web uygulamaları**

Web uygulaması üzerinde kullanıcıların görüntülediği, sepetine eklediği veya beğendiği ürün bilgileri kayıt altına alınabilir. Kullanıcının web uygulaması üzerinden yapmış olduğu işlemin anlık olarak kaydedilmesi mümkündür.

- **Mobil uygulamalar**

Web uygulamalarında olduğu gibi mobil uygulamalarda da aynı işlemler gerçekleştirilebilir. Kullanıcıların kayıt altına alınmak istenen son aktivite verisi anlık olarak geliştirilen uygulama ile kaydedilebilir.

4.3.1.2 Modüller

- **Üretici**

Şekil 4.47’de ‘Producer’ (Üretici) isimli modüldür. Bir Web API uygulamasıdır ve modüllerin dışı açılan kapısıdır. Bu Modül C# dili ile .Net Core yazılım çerçevesi ile geliştirilmiştir.

- **Kuyruklar**

Üretici modülüne gelen verinin, veri içerisindeki bilgilere göre gönderildiği kuyruklardır. Bu modül C# dili ile RabbitMq mesaj-aracısı kullanılarak geliştirilmiştir.

- **Tüketici**

Kuyruklara gelen veriyi dinleyen ve bu veriyi tüketen modüldür. Her kuyruk için bir tüketici vardır. Gelen verinin içerisindeki bilgilere göre, veriyi ait olduğu veri tabanına kayıt eder.

- **Veri tabanları**

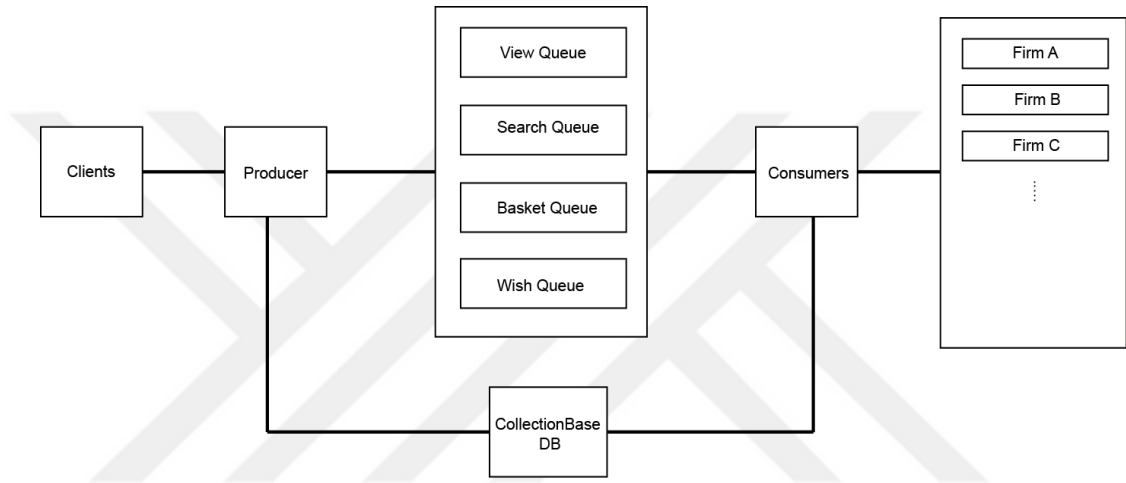
Gelen verilerin saklandığı depolama birimleridir. Depolama birimleri MongoDB ile inşa edilmiş NoSQL veri tabanı uygulamalarıdır.

- **CollectionBaseDB**

Uygulamayı kullanabilecek firma/kişi bilgilerini barındıran veri tabanıdır.

4.3.1.3 Modüller geliştirilirken uygulanan tasarımları

Geliştirilen uygulamanın mimarisi Şekil 4.39’da görülmektedir. Şekilde görülen ‘Producer’ isimli modül Web API hizmeti vermektedir. E-Ticaret sitelerinden gelen verilerin kontrollerini sağlar ve hangi modüle ait ise o modüle gönderir. ‘Consumer’ isimli modülde her kuyruğa ait bir küçük modüller barındırmaktadır. Bu modüller bağlı oldukları kuyrukları dinlerler ve yeni veri geldiği zaman veriyi okuyup ilgili kullanıcının veri tabanına kaydeder.



Şekil 4.39 : Aktivite verilerinin toplanması için geliştirilen uygulama yapısı.

Temsili durum aktarımı

Web protokollerini ve teknolojilerini kullanan bir dağıtık sistemdir. Rest prensiplerini sağlayan sistemler RESTful olarak adlandırılır [48]. Restful sistemler çoğunlukla http protokolü üzerinden HTTP fiilleri (GET, POST, PUT, DELETE) ile haberleşirler. Aktarılan verinin formatında bir kısıtlama getirmez. Verinin tipi ve özellikleri istemci ve sunucu tarafından http protokolünde yer alan içerik tipi (content-type) metaveri ile tanımlanır [49].

Kaynak kod yönetimi

Kaynak kodların yönetilmesi yazılım uygulamalarındaki sürdürülebilirliği arttırmaktadır. Eş zamanlı çalışma, versiyon takibi, geçmiş versiyonlara dönebilme gibi avantajlar getirmektedir. Bu çalışmada kaynak kod yönetimi için Git kaynak kod yönetim sistemi kullanılmıştır.

4.3.1.4 Uygulamanın teknolojik incelenmesi

Bu bölümde uygulama geliştirilirken tercih edilen araçlar hakkında bilgiler verilmiştir. Bu araçların tercih edilme sebebi ve projedeki rolü açıklanmıştır. Uygulama Şekil 4.47’de görülebileceği gibi 3 farklı modülden oluşmaktadır. Bu bölümde 3 modülde de kullanılan araçlar hakkında bilgi verilmiştir.

C# programlama dili

2000 yılında Microsoft’tan Anders Hejlsberg tarafından tasarlandı ve daha sonra 2002’de Ecma ve 2003’te ISO/IEC tarafından uluslararası bir standart olarak onaylandı. C# çok paradigmalı bir programlama dilidir. Statik yazım, güçlü yazım, sözcüksel kapsamlı, zorunlu, bildirimsel, işlevsel, genel, nesne yönelimli (sınıf tabanlı) ve bileşen odaklı programlama disiplinlerini kapsar [44].

Bu çalışmada C# tercih edilmesinin temel nedeni, tüm modülleri tek çatı altında inşa etmek ve .Net Core’un getirmiş olduğu avantajları kullanarak performanslı, ölçeklenebilir bir servis inşa edilmesidir.

.Net Core yazılım çerçevesi

Windows, Linux ve macOS işletim sistemleri için ücretsiz ve açık kaynaklı, yönetilen bir bilgisayar yazılım çerçevesidir [45].

Avantajları;

- Kolay test edilebilir
- Windows, MacOS, ve Linux üzerinde geliştirme yapabilmek
- Açık kaynak ve topluluk odaklı
- Ortam tabanlı yapılandırma sistemi
- Yerleşik bağımlılıklar
- Basit, yüksek performanslı ve modüler bir http istek işlem hattı

RabbitMq mesaj aracı

Başlangıçta Gelişmiş Mesaj Kuyruklama Protokolünü (AMQP) uygulayan ve o zamandan beri Akışlı Metin Yönelimli Mesajlaşma Protokolünü (Stomp) desteklemek için bir eklenti mimarisiyle genişletilen açık kaynaklı bir mesaj aracı yazılımıdır [46]. Erlang dili ile yazılmış Mozilla Kamu Lisansı altına yayınlanmıştır.

Herhangi bir kaynaktan gelen mesajı, başka bir kaynağa sırası geldiği anda iletilmesi için kullanılır. Kullanıcı aktivite verilerin veri depolarına kayıtlarının anlık olarak yapılması şart bir konu değildir. Burada önemli olan kayıt için gelen verinin sıraya alınarak işlemin asenkron bir şekilde yapılması, hem çalışan uygulamanın boş yere bekletilmemesi hem de sunucu üzerindeki maliyeti minimuma indirmesinden dolayı tercih edilmiştir.

MongoDB

MongoDB Inc. Tarafından ölçeklenebilir, doküman tabanlı, C++ ile geliştirilmiş açık kaynak, NoSQL veri tabanı uygulamasıdır. Verileri JSON tipinde dokümanlarda saklamaktadır [47].

- Esnek veriler, veriler belgelerde depolanır. Veri yapısı zaman içinde değişebilir.
- Dağıtılmış veri tabanı, yüksek kullanılabilirlik ve yatay ölçeklendirme
- Ücretsiz, Sunucu Tarafı Kamu Lisansı (SSPL)

4.3.1.5 İstek modelleri

Aşağıda Uygulamam kabul ettiği veri modeli açıklanmıştır.

Genel istek modeli, yapılan her istekte bulunması gerek model parametrelerini içermektedir. Çizelge 4.26’da gerekli parametreler açıklamıştır.

Çizelge 4.26 : Aktivite verilerinin toplanması için geliştirilen uygulamanın genel istek modeli.

İsim	Tip	Zorunluluk	Açıklama
Action	String	Zorunlu	Aksiyonu tipi; alabileceği 4 değer vardır. ‘View’, ‘Basket’, ‘Wish’, ‘Search’
AppKey	String	Zorunlu	Üye sistem birincil anahtarı
Payload	Object	Zorunlu	Aksiyona ait özel model

Çizelge 4.26’da ki genel istek modeli tüm kuyruklar için geçerlidir. Action nesnesi mesajın hangi kuyruğa gideceğini belirtmektedir. AppKey nesnesi aktif mesajın hangi sistem/firma ‘ya ait olduğu bilgisini barındırır. Payload nesnesi her aksiyona ait modeli belirtmektedir. Aşağıdaki adımlar da her aksiyonun modelleri detaylı incelenmiştir.

Not: Aşağıda belirtilen her bir model genel istek modelin ‘Payload’ objesini kapsamaktadır. Her istek için Genel İstek Modeli ile istek yapılmalıdır. ‘UserID’, ‘SessionID’ nesnelerinin zorunluluk bilgileri ‘Opsiyonel’ olarak belirtilmiştir. Bireysel olarak zorunlu değildir, her bir aksiyon için iki nesneden biri dolu olmak zorundadır.

Görüntüleme kuyruğu (View Queue): Bu kuyruk ile kullanıcıların görüntülediği ürün ve liste sayfa (katalog) bilgileri toplanması amaçlanmaktadır. Model bilgileri Çizelge 4.27 de açıklanmıştır.

Çizelge 4.27 : Görüntüleme metodu istek modeli.

İsim	Tip	Zorunluluk	Açıklama
UserID	String	Opsiyonel	Kullanıcı sisteme üye ise sistemde kayıtlı olan birincil anahtarı
SessionID	String	Opsiyonel	Kullanıcı sisteme kayıtlı değilse, sistem üzerinde tutulan oturum anahtarı
Type	String	Zorunlu	Görüntülenen sayfa bilgisi. İki farklı değer alabilir. “Product” or “Catalog”
Value	String	Zorunlu	Görüntülenen sayfanın birincil anahtarı

Arama kuyruğu (Search Queue): Sistem üzerinde bulunan ‘arama çubuğu’ aracılığı ile kullanıcıların aratmış oldukları kelime/cümle bilgilerinin toplanması amaçlanmaktadır. Model bilgileri Çizelge 4.28 de açıklanmıştır.

Çizelge 4.28 : Arama metodu istek modeli.

İsim	Tip	Zorunluluk	Açıklama
UserID	String	Opsiyonel	Kullanıcı sisteme üye ise sistemde kayıtlı olan birincil anahtarı
SessionID	String	Optionally	Kullanıcı sisteme kayıtlı değilse, sistem üzerinde tutulan oturum anahtarı
Value	String	Zorunlu	Aranan kelime

Sepet kuyruğu (Basket Queue): Kullanıcının sepete üzerindeki işlemlerinin (Ekle/Sil) toplanması amaçlanmaktadır. Model bilgileri Çizelge 4.29 ve Çizelge 4.30 da açıklanmıştır.

Çizelge 4.29 : Sepet metodu istek modeli.

İsim	Tip	Zorunluluk	Açıklama
UserID	String	Opsiyonel	Kullanıcı sisteme üye ise sistemde kayıtlı olan birincil anahtarı
SessionID	String	Opsiyonel	Kullanıcı sisteme kayıtlı değilse, sistem üzerinde tutulan oturum anahtarı
Type	String	Zorunlu	İşlem tipi. İki farklı değer alabilir “Add” or “Remove”
ProdutID	String	Zorunlu	İlgili ürün birincil anahtarı
BasketInfo	Object	Eğer ekleme işlemi ise zorunlu	Eklenen ürün bilgisi.

BasketInfo Object;

Çizelge 4.30 : Sepet metodu BasketInfo nesne modeli.

İsim	Tip	Zorunluluk	Açıklama
CurrentPrice	String	Zorunlu	Ürün satış/sepete ekleme fiyatı
OldPrice	String	Optionally	Piyasa satış fiyatı / İndirim uygulandıysa indirimli fiyat
InStock	Boolean	Zorunlu	Ürünün stok sayısı belirlenen alt limitten fazla ise ‘true’ değil ise ‘false’
Quantity	Int	Zorunlu	Eklenen ürün adedi

Favori/İstek kuyruğu (Wish Queue): Kullanıcının sistem üzerinde beğendiği ürünlerin verilerinin toplanması amaçlanmaktadır. Model bilgileri Tablo 4.38 ve 3.39 de açıklanmıştır.

Çizelge 4.31 : Favori/İstek metodu istek modeli.

İsim	Tip	Zorunluluk	Açıklama
UserID	String	Opsiyonel	Kullanıcı sisteme üye ise sistemde kayıtlı olan birincil anahtarı
SessionID	String	Opsiyonel	Kullanıcı sisteme kayıtlı değilse, sistem üzerinde tutulan oturum anahtarı
Type	String	Zorunlu	İşlem tipi. İki farklı değer alabilir “Add” or “Remove”
ProdutID	String	Zorunlu	İlgili ürün birincil anahtarı
WishInfo	Object	Eğer ekleme işlemi ise zorunlu	Eklenen ürün bilgisi.

WishInfo Object nesnesi Favori/İstek metadonuna ait bir nesnedir. Bu nesne işlem ekleme işlemi ise zorunlu bir alan olmaktadır. Çizelge 4.32’de bu nesnenin kabul ettiği parametreler açıklanmıştır.

Çizelge 4.32 : Favori/İstek metodu WishInfo nesne modeli.

İsim	Tip	Zorunluluk	Açıklama
CurrentPrice	String	Zorunlu	Ürün satış/sepete ekleme fiyatı
OldPrice	String	Optionally	Piyasa satış fiyatı / İndirim uygulandıysa indirimsiz fiyat
InStock	Boolean	Zorunlu	Ürünün stok sayısı belirlenen alt limitten fazla ise ‘true’ değil ise ‘false’
Quantity	Int	Zorunlu	Eklenen ürün adedi

4.3.2 Proje gereksinimleri

Bu bölümde projenin işlevsel ve işlevsel olmayan gereksinimleri listelenmiştir.

4.3.2.1 İşlevsel gereksinimler

- Kullanıcı sisteme veri ekleyebilecektir.

4.3.2.2 İşlevsel olmayan gereksinimler

- Sistem bir Web API servisi olarak hizmet verecektir.
- Sistemin bir ara yüzü yoktur.
- Sistem 7/24 aktif olacaktır.

- Sisteme ait her bir modül geliştirilebilir ve güncellenebilir.
- Sisteme yeni modül eklenmesi kolay olacaktır.

4.3.3 Proje analizi

Bu bölümde projenin analizi yapılmıştır. Alt başlıklarda proje ile ilgili detayları sırası ile takip edilmektedir.

4.3.3.1 Veri sözlüğü

Proje raporu içerisinde geçen terimler ve bunların açıklamaları Çizelge 4.33 Veri Sözlüğünde gösterilmektedir.

Çizelge 4.33 : Aktivite verileri veri sözlüğü.

Terim	Açıklama
Use Case	Kullanım senaryosu, sistemlerin fonksiyonel ihtiyaçlarını kapsamak için yazılım ve sistem mühendisliğinde kullanılan bir tekniktir. Aktörlerin sistemle etkileşimini tanımlayan senaryolardır.
API	Uygulama programlama ara yüzü, bir yazılımın bir başka yazılımda tanımlanmış işlevlerini kullanabilmesi için oluşturulmuş bir tanım bütünüdür.
'Clientkey'	API'a erişim yapan E-Ticaret sitesinin özel kimlik anahtarı
HTTP Durum Kodu	İstemci ile sunucu arasındaki aktarım işlemi sırasında neler olduğunu açıklayan bir sayı kümesidir.
Kullanıcı	Sistemi kullanan herhangi bir kimse
Control Object	Kontrol sınıfı, yapılacak işlem için belirlenen doğrulama/geçerleme işlemleri ile ilgilenen sınıftır.

4.3.3.2 İş modeli

Bu bölümde projenin kullanım senaryoları ve aktörleri Çizelge 4.34'te görülmektedir. Devam eden başlıklarda kullanım senaryoları incelenmektedir.

Use Case'ler ve Aktörler

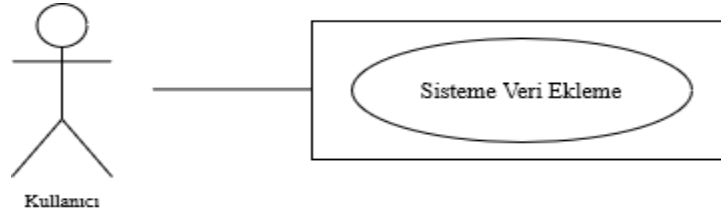
Çizelge 4.34 : Kullanım senaryoları ve aktörler.

KSİD	Kullanım Senaryosu Adı	Aktörler
USC1	Sisteme Veri Ekleme	Kullanıcı

4.3.3.3 Kullanım senaryolarının kısa tanımları

USC1 sisteme veri ekleme

Kullanıcıların sisteme genel istek modeli ile veri ekleyebilirler. Şekil 4.40'te Sisteme veri ekleme kullanım senaryosu gösterilmektedir. Çizelge 4.35'da bu kullanım senaryosunun kısa tanım ve adım adım tanımları açıklanmıştır.



Şekil 4.40 : Sisteme veri ekleme kullanım senaryosu.

Çizelge 4.35 : Sisteme veri ekleme kullanım senaryosu.

Kısa Tanım: Veri ekleme Use case'i kullanıcı aktörünün veri modeli ile istek yaparak sisteme veri eklemesidir.
Adım Adım Tanım: 1. Kullanıcı ilgili bilgiler ile sisteme istek yapar. 2. Sistem gönderilen modeli doğrulamasını yapar ve veri kaydını gerçekleştirir. 3. Sistem 200 HTTP Durum kodunu döner.

4.3.3.4 Kullanım senaryoları ve iş birliği diyagramları

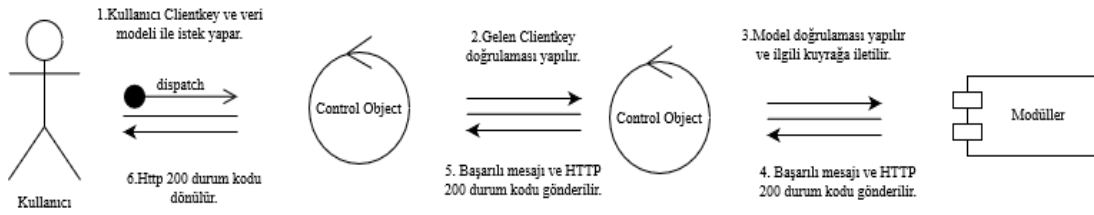
Bu bölümde sırası ile her kullanım senaryosu için karşılaşılabilecek normal ve anormal senaryolardan bahsedilmiştir. Bu senaryolar iş birliği diyagramları ile desteklenmiştir.

Sisteme veri ekleme kullanım senaryosu için senaryolar ve iş birliği diyagramları

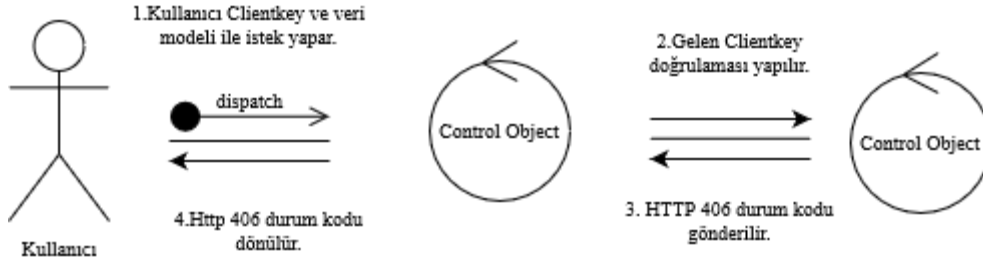
Çizelge 4.36'da kullanıcının sisteme veri eklemesi için normal senaryo örneği ve hatalı "Clientkey" sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.41'de ve anormal senaryonun iş birliği diyagramı Şekil 4.42'de gösterilmiştir.

Çizelge 4.36 : Sisteme veri ekleme kullanım senaryosu için senaryo 1.

Normal Senaryo-1: Kullanıcı 'Clientkey' ve veri modeli ile birlikte sisteme istek yapar. 1. Sistem gelen 'Clientkey' bilgisini kayıtlar ile kıyaslar. 2. Kayıtlar eşleşirse gelen modeli kuyruğa gönderir ve 200 HTTP Durum kodunu döner.
Anormal Senaryo: Kullanıcı 'Clientkey' ve veri modeli ile birlikte sisteme istek yapar. 1. Sistem gelen 'Clientkey' bilgisi kayıtlı kullanıcılar ile kıyaslar. 2. Kayıtlar eşleşmez ve 406 HTTP Durum kodunu döner.



Şekil 4.41 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 1.

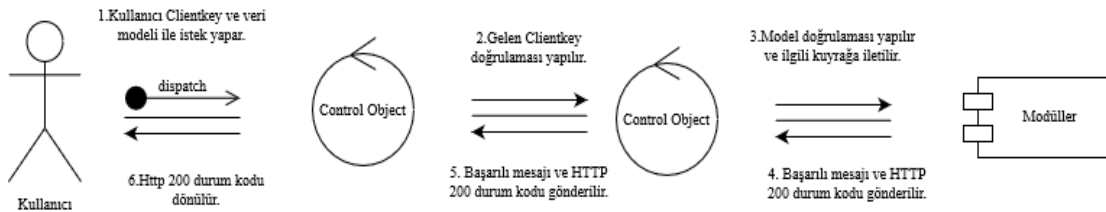


Şekil 4.42 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 1.

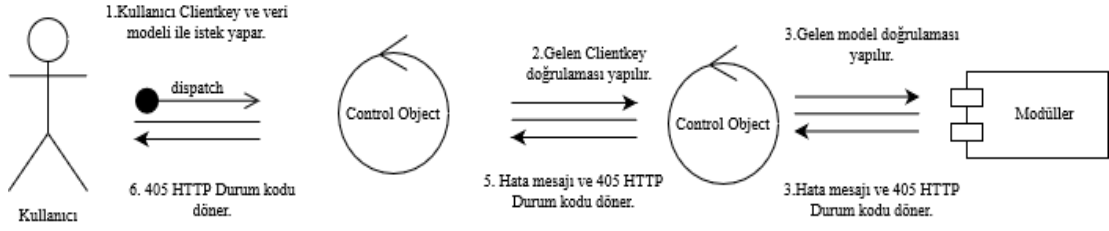
Çizelge 4.37’da Kullanıcının sisteme veri eklemesi için normal senaryo örneği ve model doğrulama işleminin başarısız olması sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.43’de ve anormal senaryonun iş birliği diyagramı Şekil 4.44’de gösterilmiştir.

Çizelge 4.37 : Sisteme veri ekleme kullanım senaryosu için senaryo 2.

Normal Senaryo-2: Kullanıcı ‘Clientkey’ ve veri modeli ile birlikte sisteme istek yapar. 1. Sistem gelen ‘Clientkey’ bilgisini kayıtlar ile kıyaslar. 2. Kayıtlar eşleşirse gelen modeli kuyruğa gönderir ve 200 HTTP Durum kodunu döner.
Anormal Senaryo: Kullanıcı ‘Clientkey’ ve veri modeli ile birlikte sisteme istek yapar 1. Sistem gelen ‘Clientkey’ bilgisi kayıtlı kullanıcılar ile kıyaslar. 2. Model doğrulaması yapılamaz ve 405 HTTP Durum kodunu döner



Şekil 4.43 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 2.

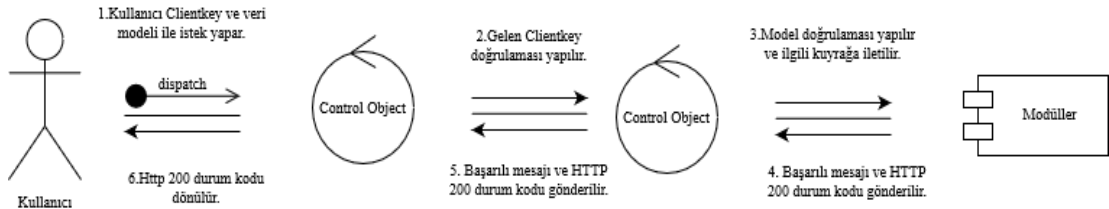


Şekil 4.44 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 2.

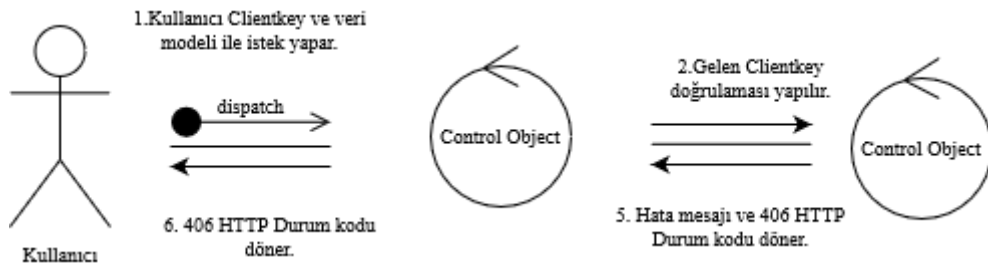
Çizelge 4.38’de kullanıcının sisteme veri eklemesi için normal senaryo örneği ve ‘Clientkey’ bilgisinin eksik olması sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.45’te ve anormal senaryonun iş birliği diyagramı Şekil 4.46’da gösterilmiştir.

Çizelge 4.38 : Sisteme veri ekleme kullanım senaryosu için senaryo 3.

Normal Senaryo-3: Kullanıcı ‘Clientkey’ ve veri modeli ile birlikte sisteme istek yapar. 1. Sistem gelen ‘Clientkey’ bilgisini kayıtlar ile kıyaslar. 2. Kayıtlar eşleşirse gelen modeli kuyruğa gönderir ve 200 HTTP Durum kodunu döner.	
Anormal Senaryo: Kullanıcı veri modeli ile birlikte sisteme istek yapar 1. Sistem gelen modelde ‘Clientkey’ bulamadığı için 406 HTTP Durum kodunu döner	



Şekil 4.45 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 3.



Şekil 4.46 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 3.

Çizelge 4.39’da kullanıcının sisteme veri eklemesi için normal senaryo örneği ve model bilgisinin eksik olması sebebi ile oluşan anormal senaryo örneği anlatılmıştır. Hemen akabinde bu normal senaryonun iş birliği diyagramı Şekil 4.47’de ve anormal senaryonun iş birliği diyagramı Şekil 4.48’de gösterilmiştir.

Çizelge 4.39 : Sisteme veri ekleme kullanım senaryosu için senaryo 4.

Normal Senaryo-4:

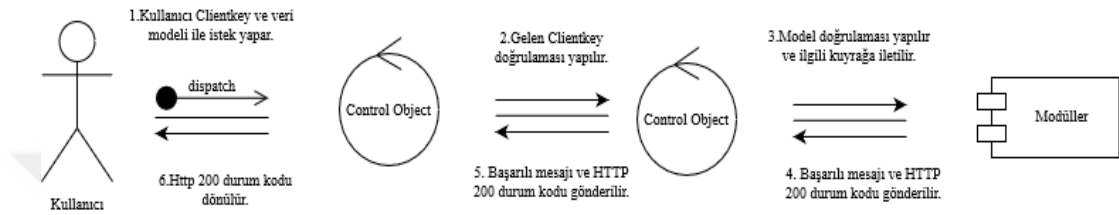
Kullanıcı 'Clientkey' ve veri modeli ile birlikte sisteme istek yapar.

1. Sistem gelen 'Clientkey' bilgisini kayıtlar ile kıyaslar.
2. Kayıtlar eşleşirse gelen modeli kuyruğa gönderir ve 200 HTTP Durum kodunu döner.

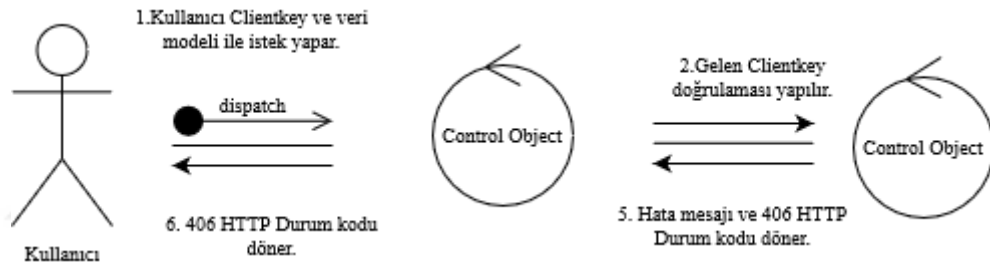
Anormal Senaryo:

Kullanıcı 'Clientkey' ile birlikte sisteme istek yapar

1. Sistem gelen 'Clientkey' bilgisini kayıtlar ile kıyaslar.
2. Model olmadığı için doğrulama yapılamaz ve 405 HTTP Durum kodunu döner.



Şekil 4.47 : Sisteme veri ekleme kullanım senaryosu için normal senaryo İBD 4.



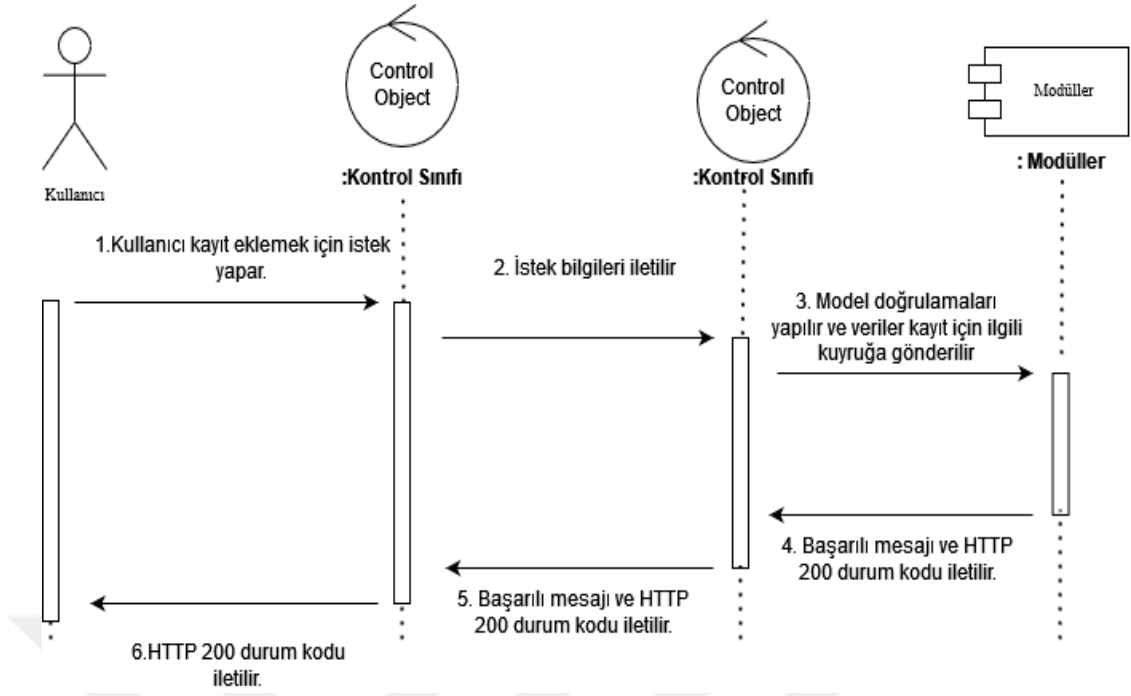
Şekil 4.48 : Sisteme veri ekleme kullanım senaryosu için anormal senaryo İBD 4.

4.3.3.5 Sınıf tasarımları

Bu bölümde veri ekleme kullanım senaryosu için sıralama diyagramı oluşturulmuştur. Bu diyagram kullanıcı sistemden ilgili senaryo için istekte bulunduğu anda sistem üzerinde gerçekleşen adımları göstermektedir.

Sisteme veri ekleme kullanım senaryosu sıralama diyagramı

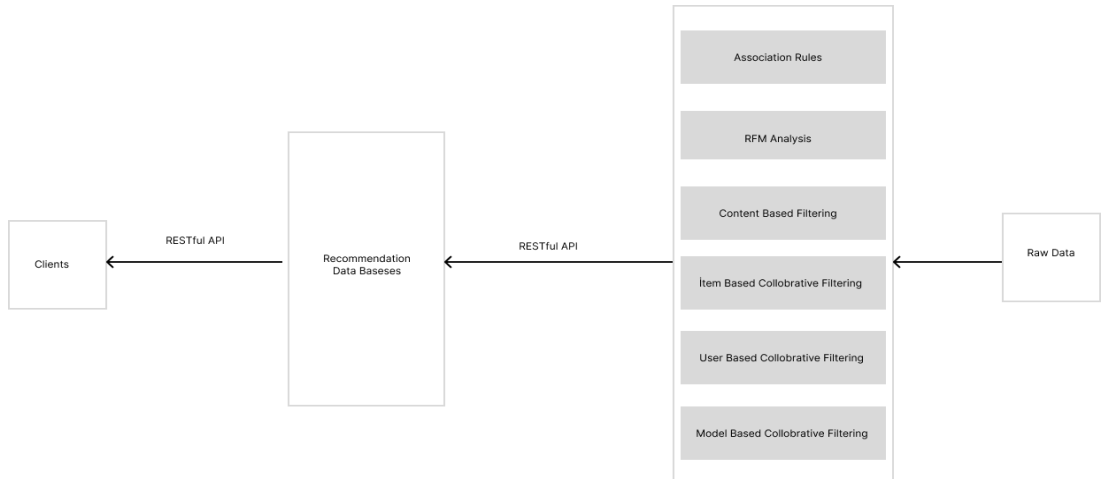
Kullanıcı sisteme veri eklemek için istek yaptığı anda sistem üzerinde gerçekleşen adımlar Şekil 4.49'te gösterilmiştir.



Şekil 4.49 : Sisteme veri ekleme kullanım senaryosu sıralama diyagramı.

4.4 Öneri Sistemi Uygulamaları

Bu bölümde sırasıyla öneri yöntemlerinin uygulama örnekleri verilmiştir. Bu yöntemler E-Ticaret uygulamalarında kullanıldıkları alanlar ve iş problemleri ile açıklanmıştır. Her uygulamada kullanılan veri seti ve uygulama adımları detaylı bir şekilde incelenmiştir.



Şekil 4.50 : Öneri sistemi uygulama tasarımı.

Şekil 4.50’de Veri Toplama Uygulamalarından toplanan veriler ‘Raw Data’ isimli modül ile ifade edilmektedir. Bu modül E-Ticaret sitelerinden toplanan işlenmemiş verileri içermektedir. Her bir öneri yöntemi için 3 uygulayacağı temel 3 adım vardır.

- İşlenmemiş verilere erişim
- Verilerin işlenip bilgi elde edilmesi
- Elde edilen bilgilerin veri tabanının Web API aracılığı ile kaydedilmesi

Bu süreç sonunda E-Ticaret Uygulamaları 'Recommendation Data Bases' isimli modülden ilgili Web API aracılığı ile veri tabanından faydalı bilgilere erişerek öneri işlemini gerçekleştirebilir.

4.4.1 Müşteri segmentasyonu

E-ticaret pazarlama faaliyetlerinde önemli bir sorun, her iki tarafın hedeflerini tatmin eden fikir, ürün ve hizmet alışverişini kolaylaştırmak için çevrimiçi faaliyetler aracılığıyla bir müşterinin tam olarak ne istediğini ve ihtiyaç duyduğunu nasıl bulacağıdır [39]. Çevrim içi mağazaların, müşterilerini gereksinimlerine göre memnun etmek ve onlar ile iyi bir ilişkiye sahip olmak için müşterilerin davranışlarını analiz etmesi gerekir [40]. İnternette alışveriş yapmayı tercih eden müşterilerin devamlılığını sağlamak için bireyin ihtiyaçlarının karşılanması; Bunun için öncelikler ve talepler dikkate alınmalıdır.

Müşteri segmentasyonu pazarlamanın vazgeçilmez alanlarından biridir. Müşterilerin tercihlerine göre sınıflandırılıp pazarlama stratejilerinin geliştirilmesi, müşteri sadakatini ve memnuniyetini artış olması kaçınılmazdır.

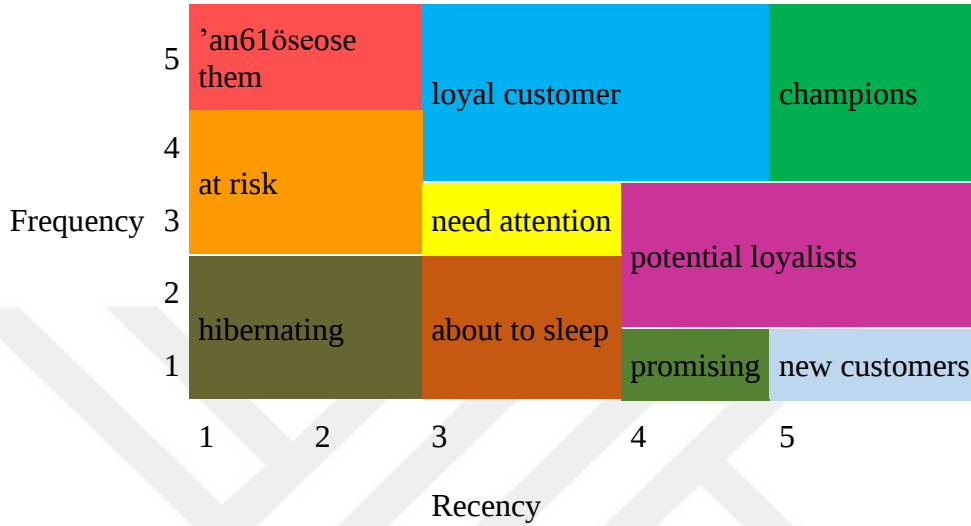
4.4.1.1 RFM analizi

Yenilik, sıklık ve parasal (RFM) analizi, veritabanı pazarlamasında güçlü ve tanınan bir tekniktir. Müşterileri önceki satın alma geçmişlerine göre sıralamak için yaygın olarak kullanılır. RFM analizi, e-ticaret, perakendecilik vb. gibi çok sayıda müşteriyi içeren çok çeşitli uygulamalarda kullanım alanı bulur. Bu yöntem, müşterileri yenilik(R), sıklık (F) ve parasal (M) olmak üzere üç boyuta göre gruplandırır [41].

Bu yöntem müşteriyi 3 farklı değişkenle 5 eşit miktara sınıflandırır. Genellikle (5, 5, 5) müşteri sonucunun kurum için en değerli sonuç olduğu düşünülürse, kurumun bu müşteri grubunun bakımına daha fazla dikkat etmesi gerekir. Müşterileri sıraladıktan sonra, yöntem her müşterinin 'FM' ini hesaplar ve hesaplama sonucunu sıralar, ardından müşterileri %20, %60, %20 oranlarına göre sınıflandırır, son olarak şirket farklı müşteri grupları ile karşılaştığında farklı stratejiler uygular. Eski %20 en değerli müşteridir, şirket bu insan grubunu elinde tutmak için elinden gelenin en iyisini

yapmalıdır, '60'lık grup için şirket onlara biraz yatırım yapmalı, mümkün olduğunca eski '20'lik gruba devretmelidir; ikinci '20'lik grup, şirketin vazgeçebileceği gruptur, çünkü bu insanların yarattığı değer, işletmenin onlara yaptığı yatırımdan daha azdır [42].

RFM skorlarına göre gruplanan müşteriler yaygın olarak Şekil 4.51'deki gibi ayrılır.



Şekil 4.51 : RFM analizinde gruplandırma.

Kullanılan veri seti ve uygulama

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır.

- **İş problemi tanımı;**

Firma mevcut kullanıcılarını belirli gruplara ayırarak, pazarlama stratejisi geliştirmek istemektedir.

- **Veri seti hikayesi;**

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır. Çizelge 4.40’ta RFM analizi uygulamasında kullanılan veri seti içeriğinden 3 adet örnek gösterilmektedir.

- **Uygulama geliştirilirken kullanılan araçlar;**

- Python (Versiyon:3.9.7) dili ile geliştirilmiştir.
- ‘pandas (Versiyon : 1.3.4)’ ve ‘numpy (Versiyon : 1.20.3)’ kütüphaneleri kullanılmıştır.

- **Veri setindeki değişkenler;**

- **OrderNumber:** Sipariş numarası (Fatura numarası) benzersiz sipariş numarası.
- **Barcode:** Ürün barkod değeri, bir ürünün her rengi için ayrı bir barkod değeri mevcuttur.
- **Quantity:** İlgili ürünün sipariş içerisindeki adedi
- **Date:** Sipariş tarihi
- **UserID:** Benzersiz müşteri numarası
- **Price:** İlgili ürünün satış fiyatı

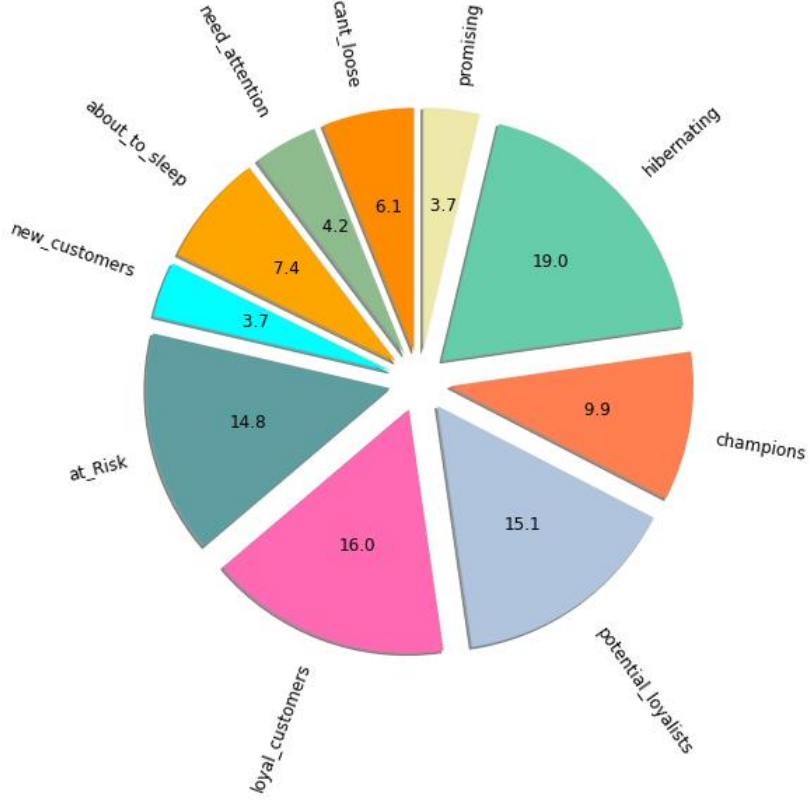
Çizelge 4.40 : RFM analizinde kullanılan veri özeti.

OrderNumber	Barcode	Quantity	Date	UserID	Price
85862	8802946510269	1	3/6/2022	12157	106.20TL
85862	8802946510443	1	3/6/2022	12157	95.40TL
85861	868024613286	1	3/5/2022	11563	110.00TL

Kullanılan veri setinde ürünün adet bilgisi ve fiyat bilgisi mevcuttur lakin toplam tutar bilgisi mevcut değildir. Bu bilgi adet ve fiyat bilgisinin çarpılması ile veri setine eklenmiştir. Veri seti üzerine RFM analizi uygulanmıştır, analiz sonucu özet görünümü Çizelge 4.41 da görülmektedir.

Çizelge 4.41 : RFM analiz sonucu özet görünümü.

	Recency			Frequency			Monetary		
	mean	count	max	mean	count	max	mean	count	max
Segment									
About_to_sleep	23.0	1209	27	1.0	1209	1	328.0	1209	2879.0
At_Risk	37.0	2425	51	1.0	2425	2	406.0	2425	3259.0
Cant_loose	38.0	989	51	4.0	989	33	1302.0	989	18492.0
Champions	4.0	1614	8	3.0	1614	32	1086.0	1614	16325.0
Hibernating	41.0	3101	51	1.0	3101	1	330.0	3101	11377.0
Loyal_customers	18.0	2620	27	3.0	2620	26	947.0	2620	40681.0
Need_attention	22.0	694	27	1.0	694	1	297.0	694	2949.0
New_customers	4.0	599	8	1.0	599	1	336.0	599	3519.0
Potential_loyalists	8.0	2472	17	1.0	2472	1	325.0	2472	3509.0
Promising	13.0	610	17	1.0	610	1	338.0	610	3219.0



Şekil 4.52 : RFM analiz sonucu grup dağılımı.

Şekil 4.52’de RFM analizinin grup dağılımı görülmektedir. Gruplar ile ilgili olarak aşağıdaki gibi yorumlar yapılarak faydalı aksiyon kararları alınabilir.

Grup Yorumları

- **Champions**

Müşteri portföyünün %9,9’unu oluşturan bu segment, ortalama 1086 TL ciro üreten 1614 müşteriyi içermektedir.

- **Loyal customer**

Müşteri portföyünün %16’sını oluşturan bu segment, ortalama 947 TL ciro üreten 2620 müşteriyi içermektedir. Yaklaşık 2.5 hafta önce alışveriş yapmışlardır.

- **Need attention**

Müşteri portföyünü %4.2 sini oluşturan bu segment, ortalama 247 TL ciro üreten 694 müşteriyi içermektedir.

- **At risk & hibernating**

Müşteri portföyünü % 33.8'ini (at_risk: %14.8 & hibernating: %19) oluşturan bu segmentler, uzun zamandır sisteme uğramamış kayıp ve kayıp müşteri adayları olarak görülebilir. Bu müşteri segmentleri için, müşteri geri kazanım aksiyonları alınmalıdır. Bu tarz müşterileri geri kazanma maliyet bazen yeni müşteri kazanma maliyetinden daha fazla olabilir.

Özet olarak analiz sonuçları yorumlanarak müşteri segmentleri için çeşitli pazarlama aksiyonları düzenlenerek müşteri memnuniyeti ve satış rakamları artırılabilir. Segmentler özel aksiyonlar uygulanmadan önce yatırım dönüşü (ROI) hesaplaması dikkate alınarak yapılması faydalı olacaktır. E-ticaret alanında uygulanması çeşitli kampanya süreçlerinde müşterilerin bulundukları segmente özel içerikler üretmek için faydalı olabilir. Hedefler doğrultusunda segmentlerdeki müşterileri bir üst segmente taşımak için çeşitli karar süreçleri alınıp, müşteri sadakati, memnuniyeti ve ciro artırılabilir.

4.4.2 Birliktelik analizi

Birliktelik kuralı uygulamalarından biri olan Pazar sepeti analizi gerçekleştirilecektir. Birliktelik Kuralı için farklı algoritmalar vardır, bu uygulamada Apriori Algoritması kullanılmıştır.

- **İş problemi tanımı;**

Kullanıcının sepetine eklediği ürün/ürünler ile ilişkisi bulunan ürünleri sepet sayfasında kullanıcılara önererek sipariş içeriğinin artırılması.

- **Veri seti hikayesi;**

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır. 20000 adet kayıt mevcuttur. Ürün kataloğun da kadın giyim eşyaları bulunmaktadır. Çizelge 4.42’de birliktelik analizi uygulamasında kullanılan veri setinde 3 adet örnek gösterilmektedir.

- **Veri setindeki değişkenler;**

- **Invoice:** Fatura numarası,
- **ProductID:** Ürün birincil anahtarı, her ürün için benzersiz bir değerdir.
- **Quantity:** İlgili ürünün sipariş içerisindeki adedi
- **Name:** Ürünün Adı

- **InvoiceDate:** Fatura Tarihi
- **UnitPrice:** İlgili ürünün adet fiyatı
- **CustomerID:** Benzersiz müşteri numarası
- **Category:** Ürünün ana kategorisi
- **Uygulama geliştirilirken kullanılan araçlar;**
 - Python (Versiyon:3.9.7) dili ile geliştirilmiştir.
 - ‘mlxtend (Versiyon:0.19.0)’ kütüphanesi içerisinde ‘apriori’ ve ‘association_rules’ metotları kullanılmıştır.
- **Veri ön işleme;**

Çizelge 4.42 : Birliktelik kuralı kullanılan veri setinin incelenmesi.

Invoice	Invoice	Quantity	Date	UserID	Price
85862	24976	1	4/19/2021	36539	374.50TL
85862	20628	1	4/19/2021	36452	109.00TL
85861	24447	1	4/19/2021	35327	299.50TL

- Verilerin tür dönüşümleri gerçekleştirilmiştir
- Adeti ve Fiyatı ‘0’ a eşit ve küçük olanlar veri setinden çıkartılmıştır.
- Veri setinde Boş (‘Null’) değerler mevcuttur. Bu değerler sahip kayıtlar veri setinden çıkartılmıştır.
- **ARL veri yapısı**

Birliktelik kuralı için ARL Veri yapısı oluşturulur. Çizelge 4.43 de ARL veri yapısı görülmektedir. ARL Veri yapısı fatura numarası ve veri setindeki tüm ürünleri sütunlara dağıtır. Her bir satırda fatura numarası ve o fatura numarasına ilgili ürün var ise hücre ‘1’ ile yok ise ‘0’ değeri ile doldurulur.

Çizelge 4.43 : ARL veri yapısı.

Invoice	Product1	Pronduct2	Product3	ProductN
124123124	0	0	1	1
521512413	1	1	0	0
3242345	1	0	0	1

Veri seti 'InvoiceNumber' ve 'ProductID' değişkenine göre gruplanır ve 'Quantity' değişkeninin toplamı alınır. Bu sayede her bir satır faturadaki her bir ürününe indirgenmiş olur. Veri setinde elde edilen son görünüm Çizelge 4.44'teki gibidir.

Çizelge 4.44 : Birliktelik kuralı veri setinin son yapısı.

ProductID	20490	20821	20493	20782	...
Invoice					
527405	1	0	0	1	...
527293	0	1	0	1	
...	.	.	.	.	..

Apriori algoritması kullanılarak sıklıkla birlikte görülen ürünler tespit edilir. Algoritmanın support değeri '0.001' olarak belirlenmiştir.

- `frequent_itemsets=apriori(dataframe,min_support=0.001,use_colnames=True)`

Mlxtend kütüphanesinden 'association_rules' fonksiyonu kullanılarak birliktelik kuralları oluşturulur.

- `rules = association_rules(frequent_itemsets,metric='support',min_threshold=0.1)`

Oluşturulan birliktelik kuralı Çizelge 4.45'te gösterilmiştir ve devamında ilgili parametrelerin açıklaması yapıp tabloda yorumlanmıştır.

Çizelge 4.45 : Oluşturulan birliktelik kuralı.

antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
Bluz	Elbise	0.26979953 992770295	0.38859677 949392046	0.09037134 406835359	0.33495736 90621193	0.8619664 051213777	- 0.014471888 256493176	0.9193444 029954052
Bluz	Pantolon	0.26979953 992770295	0.22707854 091357213	0.06588892 540256326	0.24421437 271619975	1.0754621 362885612	0.004623239 536627431	1.0226728 733440085
Etek	Elbise	0.16332566 546171542	0.38859677 949392046	0.06358856 391718698	0.38933601 60965794	1.0019023 230290836	0.000120736 31006292995	1.0012105 456063312
Elbise	Pantolon	0.38859677 949392046	0.22707854 091357213	0.05915215 248110418	0.15221987 31501057	0.6703401 939157333	- 0.029089837 210088405	0.9117003 744336168
Etek	Bluz	0.16332566 546171542	0.26979953 992770295	0.04814327 965823201	0.29476861 16700201	1.0925467 543384544	0.004078090 258275262	1.0354055 025846838
Elbise	T-Shirt	0.38859677 949392046	0.15626026 947091687	0.04518567 2034176796	0.11627906 976744186	0.7441371 383855427	- 0.015536565 445073675	0.9547581 161250152
T-Shirt	Bluz	0.15626026 947091687	0.26979953 992770295	0.04025632 599408478	0.25762355 415352256	0.9548702 50047709	- 0.001902622 818147473	0.9835986 367262811
Jogger	Sweat Shirt	0.10926717 055537298	0.11781137 035819915	0.03976339 1390075585	0.36390977 443609024	3.0889189 500949024	0.026890476 291784024	1.3868922 90098812
Bluz	Kazak	0.26979953 992770295	0.10138021 689122577	0.03581991 455800197	0.13276492 082825822	1.3095742 43372414	0.008467578 682978521	1.0361893 847363814
Elbise	Gömlük	0.38859677 949392046	0.12586263 55570161	0.03549129 148866251	0.09133192 389006344	0.7256476 355025144	- 0.013418523 347410949	0.9619985 420696403
Gömlük	Pantolon	0.12586263 55570161	0.22707854 091357213	0.03401248 76766349	0.27023498 694516973	1.1900507 456934175	0.005431784 0388110016	1.0591374 555638706
Pantolon	T-Shirt	0.22707854 091357213	0.15626026 947091687	0.03351955 30726257	0.14761215 62952243	0.9446557 131574501	- 0.001963800 9215917074	0.9898542 509263097
Elbise	Küpe	0.38859677 949392046	0.11140322 050607952	0.03319093 000328623	0.08541226 21564482	0.7666947 30802572	- 0.010100002 710627352	0.9715817 769316328
Küpe	Kolye	0.11140322 050607952	0.07952678 278015117	0.03220506 0795267824	0.28908554 572271383	3.6350715 52207513	0.023345521 077071555	1.2947735 296163019
Bluz	Gömlük	0.26979953 992770295	0.12586263 55570161	0.03138350 3121919155	0.11632155 907429961	0.9241945 281020724	- 0.002574178 0454519372	0.9892029 995970915
Bluz	Sweat Shirt	0.26979953 992770295	0.11781137 035819915	0.03105488 005257969	0.11510353 22777101	0.9770154 776041055	- 0.000730573 4683146631	0.9969399 379709821
Elbise	Atlet	0.38859677 949392046	0.09267170 555372987	0.03056194 544857049	0.07864693 446088795	0.8486617 785974541	- 0.005449980 8798178055	0.9847780 708423263
Bluz	Atlet	0.26979953 992770295	0.09267170 555372987	0.03006901 084456129	0.11144945 188794153	1.2026265 322517946	0.005066227 321849415	1.0211330 085543355
Elbise	Jogger	0.38859677 949392046	0.10926717 055537298	0.03006901 084456129	0.07737843 551797041	0.7081581 331764931	- 0.012391859 737669583	0.9654368 201817337

Tablodaki parametreler;

- **support:** X ürünü ile Y ürününün birlikte satın alınma olasılığı
- **confidence:** X ürünü satın alındığında Y ürününün satın alınma olasılığı
- **lift:** X ürünü satın alındığında Y ürününün satın alınma olasılığı ‘Z’ kat artmaktadır.

Oluşturulan kurallar sonucunda aşağıdaki yorumlar yapılabilir.

- Bluz kategorisindeki ürünler ile Elbise kategorisindeki ürünlerin birlikte satın alınma olasılığı 0. 09037134406835359
- Bluz kategorisinden ürün satın alındığında, Elbise kategorisinden ürün satın alınma olasılığı %3,34'tür.
- Bluz kategorisinden ürün satın alındığında, Elbise kategorisinden ürünün satın alınma olasılığı 0,8 kat artmaktadır.

Oluşturulan birliktelik kuralı ürünlerin kategorilerine göre eşleştirilmiştir. Çizelge 4.45'teki değerler okunarak yukarıdaki gibi yorumlar yapıp çeşitli aksiyon kararları alınabilir. Mevcuttaki veriler ile ürün tabanlı birliktelik kuralı oluşturduğumuzda sonuçlar Çizelge 4.46'daki gibidir.

Çizelge 4.46 : Ürün bazlı birliktelik kuralı.

Antecedents	consequences	antecedent support	consequent support	support	confidence	lift	leverage	conviction
20386	20559	0.01462372 658560631	0.00525796 9109431482 5	0.00213604 9950706539 6	0.1460674 157303370 7	2,78 E+1 5	0.00205915 8848054649 7	1,16 E+15
19981	20023	0.00591521 524811041 7	0.00492934 6040092014	0.00230036 1485376273 4	0.3888888 888888889	7,89 E+1 5	0.00227120 3342516708 4	1,63 E+16
19808	19825	0.00755833 059480775 5	0.00328623 0693394676	0.00246467 3020046007	0.3260869 565217391 4	9,92 E+1 5	0.00243983 4602054525 7	1,48 E+16
19955	19949	0.02070325 336838646	0.01840289 1883010187	0.00279329 6089385475	0.1349206 349206349	7,33 E+1 6	0.00241229 6356020492 6	1,13 E+16
20083	20093	0.00427209 990141307 9	0.00410778 8366743345	0.00279329 6089385475	0.6538461 538461539	1,59 E+1 5	0.00277574 7207108885	2,88 E+15
19880	19866	0.00427209 990141307 9	0.00525796 9109431482 5	0.00328623 0693394676	0.7692307 692307692	1,46 E+1 5	0.00326376 8124080641	4,31 E+15
19904	19903	0.00640814 985211961 9	0.00870851 1337495893	0.00427209 9901413079	0.6666666 666666666	7,66 E+1 5	0.00421629 4455773523	2,97 E+15
19841	19834	0.00837988 826815642 4	0.00739401 9060138022	0.00443641 1436082813 4	0.5294117 647058824	71,6	0.00437445 0382506238	2,11 E+15

Tabloda görüldüğü gibi birlikte görülme olasılıkları oldukça düşüktür. Bu çalışma giyim sektörü için ürün özelinde sağlıklı sonuçlar üretemeyebilir. Kullandığımız veri seti ile faydalı birliktelikler elde edilememiştir. Bunun başlıca sebepleri;

- Eğitime katılan ürün sayısının fazlalığı
- Ürün kataloğunun birliktelik analizine uygun olmaması

- Sadece güncel sezona ait satış / ürünler içermemesi

gibi bazı etkenler sonuçları olumsuz etkileyebilir.

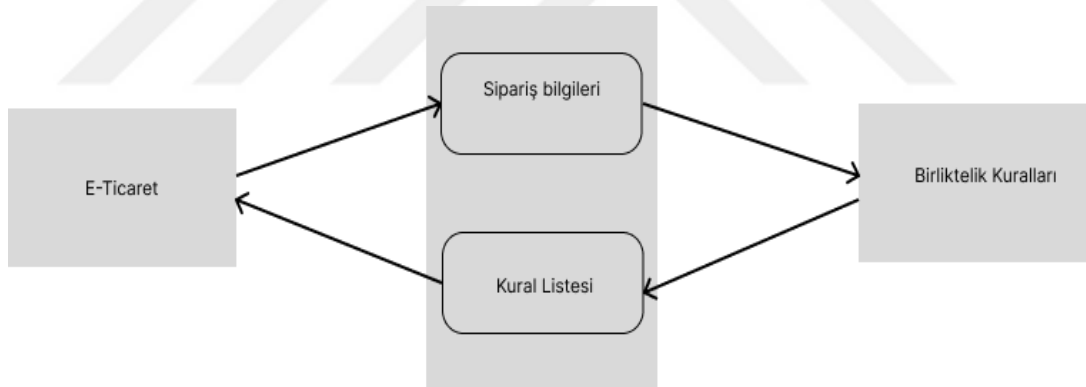
Sektörler özelinde değerlendirilerek E-Ticaret üzerinde kullanılıp fayda üretilebilecek bir yöntemdir. Kategoriler özelinde birliktelik kurallarından da fayda modelleri çıkartılabilir.

Uygulama ve öneriler

Sepet analizi E-Ticaret sitelerinde genellikle kullanıcının siparişini tamamlamak için Sepet sayfasına gittiğinde kullanıcıya gösterilen ürünler olarak karşımıza çıkmaktadır. Öneri başarısı için kural tablosundaki ‘confidence’ parametresinin iyi yorumlanması önemlidir. Faydalı birliktelik kurallarının elde edilmesi için ürün kataloğu ve satış verileri iyi incelenmelidir. Gerekirse veri seyrekleştirme işlemine tabi tutulmalıdır.

Tasarım

Şekil 4.53’te Birliktelik kuralı uygulamasının bir e-ticaret ile kullanılması için tasarım örneği gösterilmiştir.



Şekil 4.53 : Birliktelik kuralı uygulama tasarımı.

Site üzerinde alınan her aksiyonda algoritmanın yeniden çalıştırılması ciddi maliyetler doğurabilir. Maliyetten kaçınmak için veri yoğunluğu ve site trafiği dikkate alınarak algoritma belirli aralıklar ile çalıştırılıp sonuçları bir veri tabanına kaydedilip, web sitesi üzerinde kullanıcıya öneri yapılırken bu veri tabanı üzerinden işlemler gerçekleştirilmesi maliyeti ciddi oranda azaltabilir. E-Ticaret sitesinden depolama alanı erişimi için API servisleri üzerinden erişilmesi güvenli ve hızlı bir iletişim kanalı oluşturabilir.

4.4.3 İçerik tabanlı filtreleme

- **İş problemi tanımı;**

Sisteme yeni üye olmuş veya üyeliği bulunmayan kullanıcının ilgilendiği ürüne benzer ürünler önerilmesi.

- **Veri seti hikayesi;**

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır. Bu veri setinde 3733 adet ürüne ait veri mevcuttur. Çizelge 4.47’de içerik tabanlı filtreleme uygulamasında kullanılan veri setine ait 3 adet örnek gösterilmektedir.

- **Uygulama geliştirilirken kullanılan araçlar;**

- Python (Versiyon:3.9.7) dili ile geliştirilmiştir.
- ‘sklearn(Versiyon:0.24.2)’ kütüphanesinden ‘TfidfVectorizer’ ve ‘cosine_similarity’ metotları kullanılmıştır.

- **Veri setindeki değişkenler;**

- **ProductName:** Ürün Adı
- **Material:** Ürün materyalleri
- **Detail:** Ürün hakkında bilgi

Çizelge 4.47 : İçerik tabanlı filtreleme ’de kullanılan veri setinin incelenmesi.

ProductName	Material	Detail
Abelardo Elbise 2381	%100 Tencel	Tencel kumaştan,v yakalı,kruvaze kapamalı,kısa kollu,düğme detaylı,kloş etek formlu diz altı elbise.Tencel kumaştan,v yakalı,kruvaze kapamalı,kısa kollu,düğme detaylı,kloş etek formlu diz altı elbise.Tencel kumaştan,v yakalı,kruvaze kapamalı,kısa kollu,düğme detaylı,kloş etek formlu diz altı elbise.
ABINA PANTOLON 1104	%49 Poliester %47 Pamuk %4 Elastan%	Yüksek belli,yandan gizli cepli,patlı veagraf kapamalı,fit kesim pantolon.Yüksek belli,yandan gizli cepli,patlı veagraf kapamalı,fit kesim pantolon.Yüksek belli,yandan gizli cepli,patlı veagraf kapamalı,fit kesim pantolon.Yüksek belli,yandan gizli cepli,patlı veagraf kapamalı,fit kesim pantolon.Yüksek belli,yandan gizli cepli,patlı veagraf kapamalı,fit kesim pantolon.
ADAN BLUZ 2247	%100 Pamuk	Pamuklu kumaştan,V şeklinde açıklığı bulunan gömlek yakalı,omuz nakışlı,kısa kollu ve

- **Veri ön işleme**

- Çift kayıtları önlemek için Ürün adına (ProductName) göre gruplama işlemi yapılmıştır.
- ‘Detail’ nesnesi boş olan kayıtlar veri setinden silinmiştir.
- ‘Detail’ nesnesi için metin vektörleştirme işlemi yapılmıştır.

- **Metinlerin vektörleştirilmesi**

Metin vektörleştirme kelimelerin sınıfı içerisinde sayısal veriler içeren vektörlere dönüştürülmesidir. Kelimelerin birbirine yakınlığı ya da kelimelerin sınıf içerisindeki sıklığı gibi farklı yöntemler kelime vektörleştirmede yaygın olarak kullanılmaktadır [43]. Metin vektörleştirmesi için TF-IDF yöntemi kullanılmıştır.

TF-IDF matrisinin oluşturulması

Dokümanları makine öğrenmesi algoritmaları tarafından algılanması için sayısal değerler ile ifade edilmesi gerekmektedir [15]. Bu işlem için sklearn kütüphanesinden TfidfVectorizer sınıfı kullanılmıştır. ‘Overview’ değişkeni üzerinde uygulanmıştır

- TF (Term Frequency): Dokümanda geçen terim ağırlıklarını hesaplamak için kullanılan yöntemdir.

$TF = (t \text{ teriminin ilgili dokümandaki frekansı} / \text{dokümandaki toplam terim sayısı})$

- IDF (Inverse Document Frequency): Ters doküman sıklığı

$1 + \text{Log}_e ((\text{toplam doküman sayısı} + 1) / (\text{içinde } t \text{ terimi olan doküman sayısı} + 1))$

Dokümanlardaki bilgileri TF-IDF matrisi oluşturarak sayısal değerlere dönüştürüldü. İki film arasındaki benzerliği bulmak için, iki sayı arasındaki benzerliği ölçüsü olan kosinüs benzerliği kullanılmıştır. Benzerlik/uzaklık ölçüsü için farklı teknikler mevcuttur. Aşağıda birkaçı listelenmiştir.

- Manhattan Distance
- Euclidean Distance

- Pearson Correlation

Kosinüs benzerliği kullanarak oluşturduğumuz listelenen ürün indeks bilgisi ile filtre oluşturup, diğer ürünler ile olan benzerlik skoruna ulaşabiliriz. Skoru en yüksek olan belirli bir sayıdaki ürünler kullanıcıya önerilebilir. “CINTHYA ELBISE 2586” isimli ürün için bir filtreleme yapıp skora göre sıraladığımız Çizelge 4.48’deki sonuç alınmıştır.

Çizelge 4.48 : ‘CINTHYA ELBISE 2586’ ürünü için öneriler.

Ürün Birincil Anahtar	Ürün Adı
115	BRITON ELBISE 2338
1473	ALVERA ELBISE 2365
1472	GARBINA ELBISE 2364
148	KAMINA ETEK 0793
22	BECKA ELBISE 2268

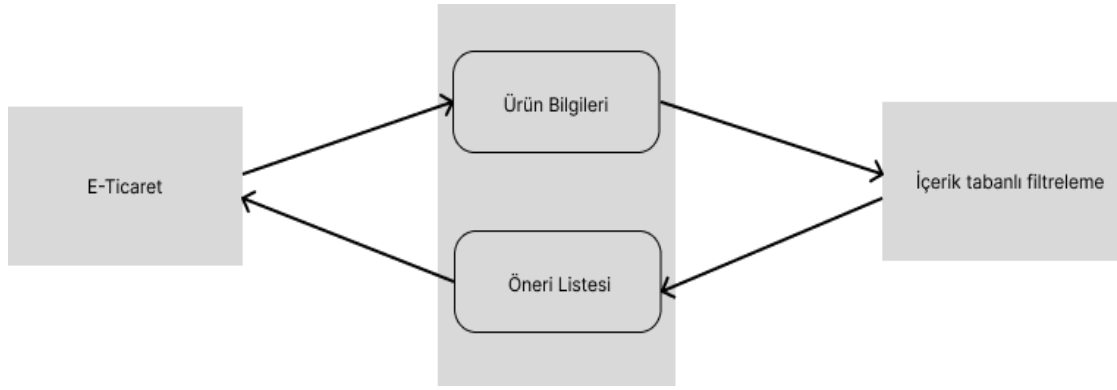
Bu öneri metodunu için sisteme eklenen yeni ürünler olduğu zaman çalıştırmak ve harici bir veri tabanında saklanması, maliyet düşürüp ve performansı arttıracaktır.

Uygulama ve öneriler

Bu filtreleme yöntemi ile E-Ticaret sitelerinde kullanıcının görüntülediği/ilgilendiği ürüne benzer ürünlerin tespit edilmesinde kullanılabilir. Kullanıcı hakkında herhangi bir bilgi gerektirmez. Sistem üzerindeki ürün bilgilerinin güncel ve temiz olması öneri performansını arttıracaktır. Uygulanan bu yöntem E-Ticaret sisteminden ayrı çalışmalı ve ürün için öneriler farklı bir depolama alanında tutulup, E-Ticaret sisteminin bu depolama alanından beslenmesi performansı arttıracaktır.

Tasarım

Şekil 4.54’te İçerik tabanı filtreleme uygulamasının e-ticaret sitesi ile çalışması için uygulama tasarım örneği gösterilmiştir.



Şekil 4.54 : İçerik tabanlı filtreleme uygulama tasarımı.

İçerik tabanlı filtreleme belirli aralıklar ile veya bir tetiklenme yardımı ile tekrar çalıştırılmalıdır. Burada sistem üzerindeki veri güncellenme hızı, yeni ürün ekleme sıklığı gibi çeşitli parametreler dikkate alınarak bir tercih yapılmalıdır. İçerik tabanlı filtreleme her çalıştırılma sonrasında veri tabanındaki liste güncellenmelidir. E-Ticaret sitesinden depolama alanı erişimi için API servisleri üzerinden erişilmesi güvenli ve hızlı bir iletişim kanalı oluşturabilir.

4.4.4 İşbirlikçi filtreleme

Bu bölümde öge tabanlı işbirlikçi, kullanıcı tabanlı işbirlikçi ve model tabanlı işbirlikçi filtreleme için uygulama geliştirilmiştir.

4.4.4.1 Öge tabanlı işbirlikçi filtreleme

- **İş problemi tanımı;**

Sistem kullanıcılarının ürünleri beğenme alışkanlıkları dikkate alınarak, benzer beğenme yapısı gösteren ürünlerin tespit edilmesi.

- **Veri seti hikayesi;**

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır. 6830 adet kayıt mevcuttur. Bu kayıtlar içerisinde 2405 adet farklı ürün için değerlendirme bulunmaktadır. Çizelge 4.49’da öge tabanlı işbirlikçi filtreleme uygulamasında kullanılan veri setine ait 3 adet örnek veri gösterilmektedir.

- **Uygulama geliştirilirken kullanılan araçlar;**

- Python (Versiyon:3.9.7) dili ile geliştirilmiştir.

- ‘pandas (Versiyon : 1.3.4)’ kütüphanesi kullanılmıştır.

- **Veri setindeki değişkenler;**

- **UserID:** Kullanıcı benzersiz anahtarı
- **ProductID:** Ürün benzersiz anahtarı
- **Rate:** Kullanıcının yapmış olduğu oylama puanı, 1-5 arasındadır.
- **Comment:** Kullanıcı yorumu

Çizelge 4.49 : Öğe tabanlı İF’de kullanılan veri setinin incelenmesi.

UserID	ProductID	Rate	Comment
325241	7928	4	İçi giysilerde çok pamuk bırakıyor ama birkaç yıkama sonra geçiyor. Kalın, sıcacık, kırıışmıyor ve topaklanma yapmıyor. Kapşonunun için de kumaş kaplı olsa mükemmel bir ürün olurdu. Bej ve beyaz rengi güzel ama kırmızı ve mavi renklerde feci akma ve solma sorunu var.
446695	6986	3	Onceden aldığım daha esnek ve rahattı şu an aldığım kumaşı daha dar
854074	10695	5	Bu modelin siyahını aldım. Fotoğrafta biraz küçük görünmüştü gözüme o yüzden tereddüt ettim ama yine de aldım iyi ki de almışım :D Beklediğimden de iyiydi kaliteli duruyor. Tavsiye ederim.

Seyreklik Problemi: Bir ürün yalnızca bir kullanıcı tarafından değerlendirilmiş olabilir. Oluşturulacak kullanıcı-ürün DF’inde bu işlem yer alacaktır. Bu oylamaların DF’in büyümesine, yapılacak olan işlemlerin gecikmesine ve performans problemlerine yol açabilmektedir. ‘Benzer beğeni’ ürünlerin tespiti yapılırken az oy almış ürünlerin veri setinden çıkarıp, daha sağlıklı bir veri setine indirgeme yapılması performansı arttıracaktır.

- **Veri ön işleme**

- Seyreklik problemi için 5 ‘den daha az oy alan ürünler veri setinden çıkartılmıştır.
- Verilerin tür dönüşümleri gerçekleştirilmiştir.

- Belirli bir sayıdan az oylama yapan kullanıcılar için veri indirgeme tercih edilebilir. Veri setimizin küçük olması nedeni ile tercih edilmemiştir.

- **Pivot tablonun oluşturulması**

Pivot tablosu kısaca daha büyük bir tabloyu özetleyen küçük istatistik tablosudur. 'Index' değerimiz 'UserID' isimli değişken, sütunlara 'ProductID' isimli değişken bunların kesişiminde 'Rate' isimli değişken kullanılarak oluşturulmuştur. Oluşturulan bu tablonun özet görünümü Çizelge 4.50'de gösterilmektedir.

Çizelge 4.50 : Öğe tabanlı İF'de kullanılan pivot tablosu.

UserID	1929	2022	22879	3771	3965
195658	0	0	5	4	0
216251	5	0	0	0	0
218365	0	0	0	3	5

Seçilen bir film tablo içerisindeki diğer filmler ile olan korelasyonuna bakılarak beğenilme benzerliği bulunabilir. Veri setindeki her ürün için bu işlem yapılarak ürün bazlı öneri listeleri elde edilir. Çizelge 4.51'de 5605 ProductID'li ve Çizelge 4.52 'de 4034 ProductID'li ürün için öneriler gösterilmektedir.

Çizelge 4.51 : Öğe tabanlı İF'de ProductID:5605 ürünü için öneriler.

	ProductID
1	12094
2	8853
3	7579
4	6986
5	5605

Çizelge 4.52 : Öğe tabanlı İF'de ProductID:4034 ürünü için öneriler.

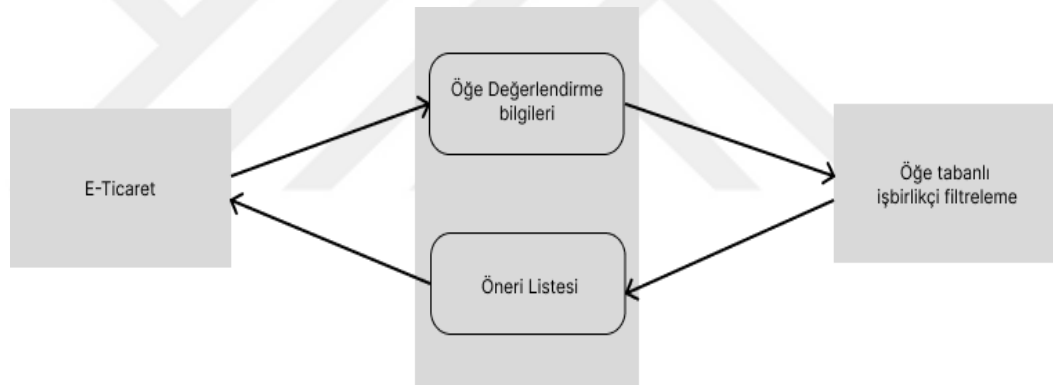
	ProductID
1	3847
2	1929
3	2202
4	2394
5	2798

Uygulama ve öneriler

Bu filtreleme yöntemi ile E-Ticaret sitelerinde ilgili ürün ile beğenme benzerliği yakınlık gösteren ürünleri tespit edebilir ve kullanıcılara önerebiliriz. Sistemdeki kullanıcıların beğeni hareketleri dikkate alınarak bir öneri yapılması, mevcut müşterilerin benzerlikleri bu öneri sistemini avantajlı kılabilir. Geniş ürün yelpazesi ve müşteri gruplarının bulunması öneri performansını etkileyebilir. Veri indirgemesi yapılırken kriterler veri setine göre yorumlanmalıdır. Oluşturulan pivot tablosunun büyüklüğü ve seyreklik problemi dikkate alınmalıdır. Seyreklik problemi için bir eşik değeri belirlenmelidir. E-Ticaret sitesi üzerinde ürün değerlendirmenin kullanım yoğunluğuna göre kullanılan veri güncellenmeli ve tekrar eğitilmelidir.

Tasarım

Şekil 4.55'te Öğe tabanlı İF uygulamasının e-ticaret sitesi ile çalışması uygulama tasarım örneği verilmiştir.



Şekil 4.55 : Öğe tabanlı İF uygulama tasarımı.

Ürünlerin değerlendirme sıklığına göre öneriler güncellenmelidir. E-Ticaret sitesi üzerinden öneri yapılmadan önce öneri listeleri hazırlanmalı ve farklı bir veri deposunda tutulup, site üzerinden bu depodan önerilere erişilmelidir. E-Ticaret sitesinden depolama alanı erişimi için API servisleri üzerinden erişilmesi güvenli ve hızlı bir iletişim kanalı oluşturabilir.

4.4.4.2 Kullanıcı tabanlı işbirlikçi filtreleme

- İş problemi tanımı;

Sistem kullanıcılarının beğenme alışkanlıkları dikkate alınarak, birbirine en benzer kullanıcıları tespit edilerek, öneri kalitesinin artırılması

- **Veri seti hikayesi;**

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır.6830 adet kayıt mevcuttur. Bu kayıtlar içerisinde 2405 adet farklı ürün için değerlendirme bulunmaktadır. Çizelge 4.53’te kullanıcı tabanlı işbirlikçi filtreleme uygulamasında kullanılan veri setine ait 3 adet veri örneği gösterilmektedir.

- **Uygulama geliştirilirken kullanılan araçlar;**

- Python (Versiyon:3.9.7) dili ile geliştirilmiştir.
- ‘pandas (Versiyon : 1.3.4)’ kütüphanesi kullanılmıştır.

- **Veri setindeki değişkenler;**

- **UserID:** Kullanıcı benzersiz anahtarı
- **ProductID:** Ürün benzersiz anahtarı
- **Rate:** Kullanıcının yapmış olduğu oylama puanı, 1-5 arasındadır.
- **Comment:** Kullanıcı yorumu

Çizelge 4.53 : Kullanıcı tabanlı İF’de kullanılan veri setinin incelenmesi.

UserID	ProductID	Rate	Comment
325241	7928	4	İçi giysilerde çok pamuk bırakıyor ama birkaç yıkama sonra geçiyor. Kalın, sıcacık, kırıışmıyor ve topaklanma yapmıyor. Kapşonunun için de kumaş kaplı olsa mükemmel bir ürün olurdu. Bej ve beyaz rengi güzel ama kırmızı ve mavi renklerde feci akma ve solma sorunu var.
446695	6986	3	Önceden aldığım daha esnek ve rahattı şu an aldığım kumaşı daha dar
854074	10695	5	Bu modelin siyahını aldım. Fotoğrafta biraz küçük görünmüştü gözüme o yüzden tereddüt ettim ama yine de aldım iyi ki de almışım :D Beklediğimden de iyiydi kaliteli duruyor. Tavsiye ederim.

Seyreklik problemi: Bir ürün yalnızca bir kullanıcı tarafından değerlendirilmiş olabilir. Oluşturulacak kullanıcı-ürün DF’inde bu işlem yer alacaktır. Bu oylamaların DF’in büyümesine, yapılacak olan işlemlerin gecikmesine ve performans problemlerine yol açabilmektedir. ‘Benzer beğeni’ ürünlerin tespiti yapılırken az oy almış ürünlerin veri setinden çıkarıp, daha sağlıklı bir veri setine indirgeme yapılması performansı arttıracaktır.

- **Veri ön işleme**

- Belirli bir sayıdan az değerlendirme alan ürünler veri setinden çıkartılmıştır. Bu sayede yaygın değerlendirme alan ürünler üzerinde işlem yapılmıştır.
- Verilerin tür dönüşümleri gerçekleştirilmiştir.
- ‘Rate’ değişkeni ‘null’ olan kayıtlar veri setinden silinmiştir.

- **Uygulama adımları**

Veri setinden rastgele bir kullanıcı (Öneri yapılacak) seçilmiştir. Bu kullanıcının değerlendirme yaptığı ürünler bir pivot tablosunda tutulur.

- **Pivot tablonun oluşturulması**

Pivot tablosu kısaca daha büyük bir tabloyu özetleyen küçük istatistik tablosudur. ‘Index’ değerimiz ‘UserID’ isimli değişken, sütunlara ‘ProductID’ isimli değişken bunların kesişiminde ‘Rate’ isimli değişken kullanılarak oluşturulmuştur. Çizelge 4.54’te örnek bir pivot tablosu gösterilmektedir.

Çizelge 4.54 : ‘340094’ Id’li kullanıcının pivot tablosu.

UserID	10063	10130	10173	10296	10097
340094	5	0	0	4	0

- **Aynı ürünleri değerlendiren diğer kullanıcılar**

Seçilen kullanıcının (Öneri yapılacak) değerlendirdiği ürün değerlendiren, diğer kullanıcılar bulunur ve bu değerlendirilen filmlerden kaçar adet değerlendirdikleri bulunarak bir DF’e aktarılır. Bu DF’den belirlenen bir adet (eşik değeri) üzerinde değerlendirme yapan kullanıcılar farklı bir DF’e alınarak benzer ürünlere değerlendirme yapan kullanıcılar listesi elde edilir. Çizelge 4.55’te benzer ürünlere değerlendirme yapan kullanıcıların listesinin bir kısmı görülmektedir.

Çizelge 4.55 : Kullanıcı tabanlı İF’ de benzer ürünlere değerlendirme yapan kullanıcılar.

	UserId
1	178653
2	768433
3	871185
4	207643
5	187354

- **Seilen kullanıcı (öneri yapılacak) ile benzer davranış gösteren kullanıcıların belirlenmesi**

Seilen kullanıcı ile diğerk kullanıcıların verileri bir araya getirilir ve korelasyon DF’i oluşturulur. Oluşturulan bu DF korelasyon değeriine göre sıralanır. Çizelge 4.56’da korelasyon DF’i görölmektedir.

Çizelge 4.56 : Kullanıcı tabanlı İF’de korelasyon DF’i.

	UserID	Korelasyon
1	190641	0.36413
2	324563	0.24513
3	768433	0.14352
4	871185	0.11974

- **Skor hesaplaması (Ağırlıklı ortalama)**

Elde edilen korelasyon ve veri setinde bulunan ‘Rate’ parametresi ile çarpılarak ‘ağırlıklı derecelendirme’ değeri elde edilir. ‘ProductID’ parametresine göre gruplandırma işlemi yapılır. Bu işlem sonrasında ‘ağırlıklı derecelendirme’ parametresine için bir eşik değeri belirlenerek önerilecek ürünler listesi oluşturulur. Çizelge 4.57’de ürünlerin ‘ProductID’ parametresine göre ağırlıklı ortalama listesinin bir kısmı görölmektedir.

Çizelge 4.57 : Kullanıcı tabanlı İF’de önerilecek ürünler listesi.

	ProductID	Weighted_rating
1	12789	4.08123
2	13374	3.80323
3	13522	3.70043
4	13674	3.02567

Uygulama ve öneriler

Bu filtreleme yöntemi ile E-Ticaret sitelerinde kullanıcıların diğerk kullanıcılar ile olan değerlendirme benzerlikleri tespit edilir. Bu yöntemin tercih edilebilmesi için büyük veri setine ihtiyaç duyulmaktadır. Öneri yapılacak kullanıcının belirli bir miktarda ürüne değerlendirme yapmış olması gerekmektedir. Bunun yanı sıra diğerk

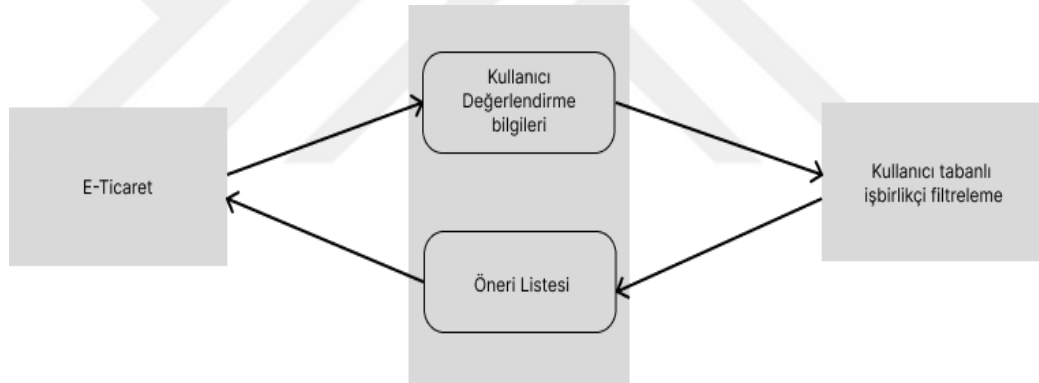
kullanıcıların da değerlendirme sayılarının yüksek olması öneri performansını arttıracaktır. E-Ticaret sistemi üzerindeki kayıtlı tüm kullanıcılar için bu filtreleme yönteminin kullanılması öneri performansını düşürüp ve maliyeti arttıracaktır. Veri indirilmesi yapılarak, belirli kullanıcılar için bu yöntemin kullanılması maliyeti düşürüp öneri performansını arttırabilir. Elde edilen bu öneriler E-Ticaret sitelerinde;

- Size özel ürünler sayfası
- E-posta pazarlama içerikleri
- Açılır-Kapanır pencereler
- Kullanıcıya özel indirim ürünleri

ve kullanıcıya özel sunulan içerik alanlarında kullanılabilir

Tasarım

Şekil 4.56’da Kullanıcı tabanlı işbirlikçi filtreleme uygulamasının e-ticaret sitesi ile çalışması için uygulama tasarım örneği gösterilmiştir.



Şekil 4.56 : Kullanıcı tabanlı işbirlikçi filtreleme uygulama tasarımı.

Bu filtreleme yöntemi belirli aralıklar ile tekrar eğitilerek öneri listesi güncellenmelidir. E-Ticaret sistemi üzerindeki kullanıcıların aktifliği, ürün değerlendirme aksiyonlarının sıklığı gibi faktörler göz önünde bulundurularak eğitme işlemi tekrarlanabilir. Elde edilen öneri listesi harici bir depolama alanında saklanmalıdır. E-Ticaret sitesi üzerinden bu depolama alanından önerilere erişilmelidir. E-Ticaret sitesinden depolama alanı erişimi için API servisleri üzerinden erişilmesi güvenli ve hızlı bir iletişim kanalı oluşturabilir.

4.4.4.3 Model tabanlı işbirlikçi filtreleme

- İş problemi tanımı;

Kullanıcının daha önceden değerlendirmedeği ürünlere, ilgili kullanıcının verebileceği değerlendirme derecesinin tahmin edilmesi

- **Veri seti hikayesi;**

Türkiye’de hizmet veren özel bir hazır giyim firmasının e-ticaret veri tabanından toplanan veriler kullanılmıştır. 6830 adet kayıt mevcuttur. Bu kayıtlar içerisinde 2405 adet farklı ürün için değerlendirme bulunmaktadır. Çizelge 4.58’de model tabanlı işbirlikçi filtreleme uygulamasında kullanılan veri setine ait 3 adet örnek veri gösterilmektedir.

- **Uygulama geliştirilirken kullanılan araçlar;**

- Python (Versiyon:3.9.7) dili ile geliştirilmiştir.
- ‘pandas (Versiyon : 1.3.4)’ ve ‘surprise(Versiyon:1.1.1)’ kütüphanesi kullanılmıştır. Kütüphanelerde kullanılan metotlar uygulama açıklanırken belirtilmiştir.

- **Veri setindeki değişkenler;**

- **UserID:** Kullanıcı benzersiz anahtarı
- **ProductID:** Ürün benzersiz anahtarı
- **Rate:** Kullanıcının yapmış olduğu oylama puanı, 1-5 arasındadır.
- **Comment:** Kullanıcı yorumu

Çizelge 4.58 : Kullanıcı tabanlı İF’de kullanılan veri setinin incelenmesi.

UserID	ProductID	Rate	Comment
325241	7928	4	İçi giysilerde çok pamuk bırakıyor ama birkaç yıkama sonra geçiyor. Kalın, sıcacık, kırışmıyor ve topaklanma yapmıyor. Kapşonunun için de kumaş kaplı olsa mükemmel bir ürün olurdu. Bej ve beyaz rengi güzel ama kırmızı ve mavi renklerde feci akma ve solma sorunu var.
446695	6986	3	Onceden aldığım daha esnek ve rahattı şu an aldığım kumaşı daha dar
854074	10695	5	Bu modelin siyahını aldım. Fotoğrafta biraz küçük görünmüştü gözüme o yüzden tereddüt ettim ama yine de aldım iyi ki de almışım :D Beklediğimden de iyiydi kaliteli duruyor.Tavsiye ederim.

- **Veri ön işleme**

- Seyreklik problemi için 5 ‘den daha az oy alan ürünler veri setinden çıkartılmıştır.

- Verilerin tür dönüşümleri gerçekleştirilmiştir.
- Belirli bir sayıdan az oylama yapan kullanıcılar için veri indirgeme tercih edilebilir. Veri setimizin küçük olması nedeni ile tercih edilmemiştir.

Uygulama adımları

- **Kullanıcı-Ürün matrisinin oluşturulması (pivot tablosu)**

Pivot tablosu kısaca daha büyük bir tabloyu özetleyen küçük istatistik tablosudur. 'Index' değerimiz 'UserID' isimli değişken, sütunlara 'ProductID' isimli değişken bunların kesişiminde 'Rate' isimli değişken kullanılarak oluşturulmuştur. Oluşturulan pivot tablo Çizelge 4.59'da gösterilmektedir.

Çizelge 4.59 : Model tabanlı İF'de kullanılan pivot tablosu.

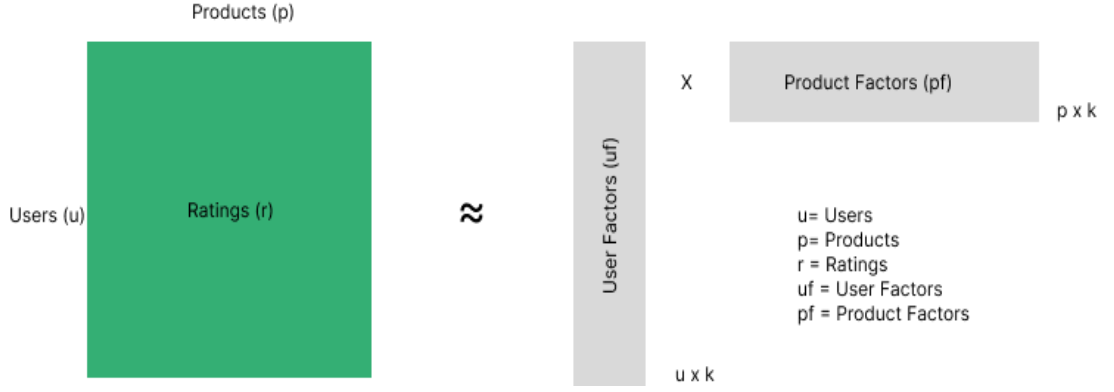
UserID	1929	2022	22879	3771	3965
195658			5	4	
216251	5				
218365				3	5

Matris çarpanlarına ayırma:

Kullanıcının daha önce değerlendirmedeği öğenin, kullanıcının bu öğeye verebileceği puanının tahmin edilmesinde kullanılır. Var olan değerlendirmeler kullanılarak, gizli özelliklerin ağırlıkları var olan veriden bulunur. Bulunan bu ağırlıklar ile var olmayan değerlendirmeler için tahmin yapılır.

Oluşturulan bu Kullanıcı-Ürün matrisi (Çizelge 4.59) iki tane daha az boyutlu matrise ayrıştırılır. Bu iki matristen Kullanıcı-Ürün matrisine gidişin gizli faktörler ile gerçekleştiği varsayımında bulunulur.

Gizli faktörler: Bir kullanıcı bir ürünü beğeniyorsa, bu kullanıcının bu ürünü beğenmesini etkileyen bazı sebepler var olduğu düşünülür. Bu sebepler ölçülebilir değildir fakat her bir kullanıcının vermiş olduğu puanı etkileyen gizli faktörler olarak kabul edilir. Örneğin bir kullanıcı bir ürünü beğenme sebebi, ürünün kumaş tipi, kol tipi, kesim tipi, rengi vb. faktörlere dayanabilir.



Şekil 4.57 : Model tabanlı İF’de Kullanıcı ürün matrisinin iki boyutlu matrise ayrıştırılması.

Kullanıcı-Ürün matrisinin, kullanıcının belirli kriterler göre davranış sergilediği gizli faktörler ürünlerin içerisinde de var olduğu kabul edilir ve bu ikisinin kesişmesinden meydana geldiği varsayılır.

Çizelge 4.59’da görülen boşlukları doldurmak için kullanıcılar ve ürünler için var olduğu varsayılan gizli faktörlerin ağırlıkları var olan veri üzerinden bulunur ve bu ağırlıklar ile var olmayan gözlemler için tahmin yapılır.

- Kullanıcı-Ürün matrisinin iki faktör matrisin çarpımı ile oluştuğu varsayılır.
- Faktör matrisler; Şekil 4.57’ de görüldüğü gibi Kullanıcı Faktörleri (User Factors(uf)) ve Ürün Faktörleri (Product Factors (pf)) ‘dir.
- Kullanıcı ve ürünlerin gizli faktörler için skorlara sahip olduğu varsayımında bulunulur.
- Bu skorlar (ağırlıklar) önce var olan veri üzerinden bulunur ve sonra veri üzerindeki boş alanlara bu ağırlıklara göre doldurulur.

Ratings(r)			
	P1	P2	P3
U1	3		5
U2		1	2
U3	4	5	

User Factors (p)		
	F1	F2
U1	p11	p12
U2	p21	p22
U3	p31	p32

Product Factors (q)				
	P1	P2	P3	P4
F1	q11	q12	q13	q14
F2	q21	q22	q23	q24

Şekil 4.58 : Model tabanlı İF’de Kullanıcı-Ürün matrisini oluşturan faktörler.

U1 Kullanıcısının P1 ürününe vermiş olduğu ‘3’ derecesi Kullanıcı faktörlerinden p_{11} ağırlığı ile ürün faktörlerinden q_{11} ağırlığının çarpımı ile elde edildiği varsayımında bulunulur.

$$r_{11} = p_{11} * q_{11} + p_{11} * q_{21}$$

$$3 = p_{11} * q_{11} + p_{11} * q_{21}$$

p ve q değerlerinin bulunması

- Başlangıçta rastgele p ve q değerleri ile Kullanıcı-Ürün matrisindeki var olan değerler tahmin edilmeye çalışılır.
- Her iterasyonda hatalı tahminler düzenlenerek Kullanıcı-Ürün matrisindeki değerlere yaklaşılmaya çalışılır.
- Belirli bir iterasyon sonucunda p ve q matrisleri doldurulur.
- Elde edilen p ve q matrisleri ile Kullanıcı-Ürün matrisindeki boş değerler doldurulur.

‘pandas’ kütüphanesinin `pivot_table` metodu (`index='userId', columns='title', values='rating'`) parametreleri ile pivot tablosu oluşturulur.

‘surprise’ sci-kit’i parametreleri için;

- Reader fonksiyonu ile değerlendirme aralığı belirtilir.
- Kullanılacak veri ‘Dataset’ sınıfının ‘load_from_df’ fonksiyonu kullanılarak ‘surprise’ kütüphanesinin destekleyeceği formata dönüştürülür.
- ‘train_test_split’ sınıfı kullanılarak veri eğitim ve test verisi olarak 2 ye ayrılır. Verinin %25 i test %75 eğitim verisi olarak ayrılmıştır.
- SVD fonksiyonu standart parametreleri kullanılarak model kurulmuş ve eğitilmiştir. Kurulan modelin başarı ölçümü için ‘accuracy’ sınıfı kullanılarak kök ortalama kare hatası (rmse) 1.2434 bulunmuştur.

Model optimizasyonu:

Optimizasyon için GridSearchCV sınıfı kullanılmıştır. Metot parametreleri

- ‘n_epoch’:[5,10,20]
- ‘lr_all’:[0.002,0.005,0.007]

- measures:['rmse','mea'] (Ölçüm yönetimi)
- cv:3, (Çapraz doğrulama)

cv: çapraz doğrulama : Veri 3 parçaya bölünür, bölünen parçanın 2 parçası eğitim 1 parçası test için kullanılır. Bu 3 parça kendi arasında değişerek her parçanın test için kullanımı gerçekleşir.

Optimizasyon sonucunda bulunan değerler;

{'n_epochs': 5, 'lr_all': 0.002}

rmse: 0.9308 bulunmuştur.

Çizelge 4.60 model optimizasyonu ile elde edilen tahmin özeti gösterilmektedir.

Çizelge 4.60 : Oluşturulan modelin tahmin özeti.

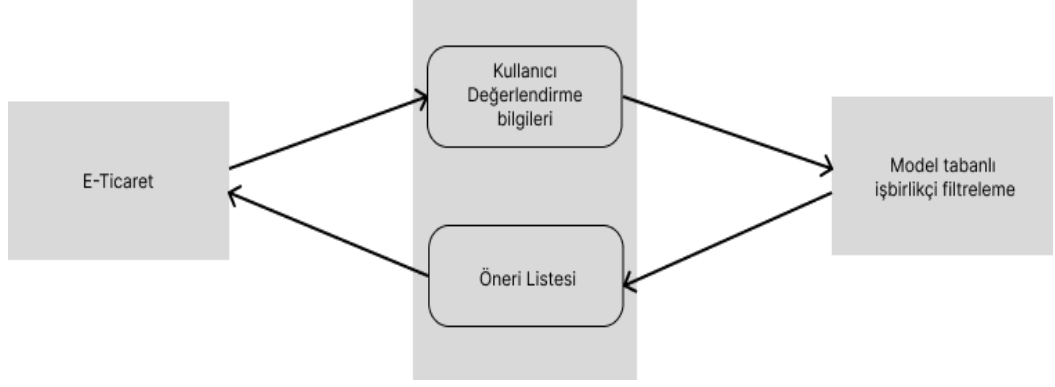
ProductId	UserId	rating	estimated
12789	20	2	2.76
12789	340094	4	4.14
13522	190641	5	4.69

Uygulama ve öneriler

Bu filtreleme yöntemi ile kullanıcıların daha önce değerlendirmedeği ürünlere verebilecekleri değerlendirme puanı tahmin edilebilir. Bu sayede yüksek puan verebileceği düşünülen ürünler kullanıcılara önerilebilir. Diğer işbirlikçi filtreleme yöntemlerinde olduğu gibi veri seyrekleştirme yoluna gidilmesi daha verimli sonuçlar için önemlidir.

Tasarım

Şekil 4.59'da Model tabanlı İF uygulamasının e-ticaret sitesi ile çalışması için uygulama tasarım örneği gösterilmektedir.



Şekil 4.59 : Model tabanlı İF uygulama tasarımı.

E-Ticaret sitesi üzerinde ürün değerlendirmenin kullanım yoğunluğuna göre kullanılan veri güncellenmeli ve tekrar eğitilmelidir. E-Ticaret sitesi üzerinden öneri yapılmadan önce öneri listeleri hazırlanmalı ve farklı bir veri deposunda tutulup, site üzerinden bu depodan önerilere erişilmelidir. E-Ticaret sitesinden depolama alanı erişimi için API servisleri üzerinden erişilmesi güvenli ve hızlı bir iletişim kanalı oluşturabilir.

5. SONUÇ VE ÖNERİLER

Bu çalışma; e-ticarete öneri sistemleri nasıl kullanılmalıdır, bu süreçte veri madenciliğinden nasıl faydalanılmalıdır ve verimli bir öneri sistemi uygulaması nasıl olmalıdır sorularına yanıt aramıştır. Söz konusu sorularla ilgili literatürde 15 farklı çalışmaya rastlanmıştır. Literatürdeki çalışmaların birçoğu öneri sistemi yöntemleri arasında karşılaştırmaya gitmiş bulunmaktadır. Hangi yöntemin daha faydalı olduğuyla ilgili fikirler bu çalışmalardan alınabilmektedir. Spesifik yöntemlere dayanan çalışmalarda ise artılar ve eksiler görülebilmekte, fakat gelişim süreci devam eden bir konu için yeterli sonuçlara ulaşamamaktadır.

Literatürdeki çalışmaları e-ticaret bağlamında incelendiğinde, sürdürülebilir çözümler elde edebilmek pek mümkün olmamaktadır. Bu çalışma, literatürdeki hâkim yaklaşımlardan farklı olarak öneri sistemleri ile veri madenciliğini birlikte kullanıp e-ticaret sektöründe fayda sağlayabilecek uygulama fikirleri vermektedir. Uygulama fikirlerini ise sektörde varlığını sürdüren firmalardan elde edilen verilerden faydalanarak ve geçerli öneri sistemlerini bir arada kullanarak savunmaktadır.

Öneri sistemi uygulamaları, hazır giyim sektöründe hizmet veren e-ticaret sitelerine yönelik bir şekilde tasarlanmıştır. Uygulamaların faaliyet prensibi anlatılırken de tablo ve grafiklerle çalışmaya eklenmiş olan ve hazır giyim sektöründe hizmet veren e-ticaret sitelerinden alınan verilerden faydalanılmıştır. Müşteri segmentasyonu, RFM analizi, birliktelik analizi, içerik tabanlı filtreleme ve öge tabanlı/kullanıcı tabanlı/model tabanlı işbirlikçi filtreleme için farklı tip ve amaçta veriler incelenmiştir.

Sistem verilerinin toplanması için geliştirilen uygulama tasarlanırken birden çok sisteme aynı anda hizmet verebilmek amaçlanmıştır. Alt yapı çalışmaları da bu gayeye bağlı kalınarak planlanmıştır. Ağ bağlantısıyla hizmet verecek olan uygulama HTTP protokolünü benimseyerek çalışacaktır. İlk görevlerinden biri, ilgili e-ticaret firmasının mobil uygulamasında ve web sitesinde veri madenciliği yapmaktır. Kullanıcıların incelediği, sepete eklediği, görüntülediği, mevcutsa beğeni butonuna tıkladığı ürünler; kişiselleştirilebilir ve anlık bir şekilde çekilerek uygulamanın kaynağını yaratacaktır.

Uygulama mimarisi Şekil 4.1 ile çalışmaya eklenmiştir. Söz konusu ek incelendiğinde, uygulamanın servis odaklı mimarinin yapısal bir stil çeşidi olan mikro servis mimarisi ile geliştirildiği görülebilmektedir. Hafif protokollü, sınırları belirli, ihtiyaca göre tasarlanmış ve merkezî mesajlaşma sistemi mevcut olması, uygulamanın fark yaratan detayları arasında sıralanabilir. Bu uygulama, öneri sistemlerinin veri deposunu genel veri tabanından ayırarak büyük bir fayda sağlamaktadır. Öneri sistemi veri deposunun ayrı olması ise gereksiz verilerin depolanmasının, deponun şişirilmesinin, doluluktan doğan aksaklıkların, performans düşüklüklerinin ve erişim hızının stabilitesini kaybetmesinin önüne geçmektedir.

Uygulamada katalog/kategori modülü, müşteri modülü, ürün modülü ve sipariş modülü bulunacaktır. Modüllerin veri örnekleri Çizelge 4.1 (katalog/kategori), Çizelge 4.12 (müşteri), Çizelge 4.3 (ürün) ve Çizelge 4.4 (sipariş) olarak çalışmaya eklenmiştir. Söz konusu modülleri tek çatı altında inşa etmek ve buna ek olarak .net Core yazılım çerçevesinin avantajlarından faydalanabilmek için uygulamanın programlama dili C# olarak belirlenmiştir.

Mesaj aracı olarak Stomp destekleyen ve AMQP uygulayan RabbitMq seçilmiştir. Uygulamada C++ ile geliştirilmiş NoSQL özelliği taşıyan MongoDB veri tabanı kullanılacaktır. Uygulamanın API formunda verimli olması; ileti alma, görev yapma, sonuçlandırma, sonuç bildirme gibi eylemleri sorunsuz tamamlaması için Ocelot API ağ geçidi tercih edilmiştir. Erişim devamlılığı ve kimlik doğrulaması için ise IdentityServer 4 ara yazılımı kullanılmıştır.

Aktivite verilerinin toplanması için geliştirilen uygulamanın yapısı, Şekil 4.47 ile çalışmaya eklenmiştir. RESTful mimarisini benimseyen bu Web API uygulaması, javascript ile geliştirilmiştir. Uygulamanın görev tanımı; kullanıcıların ilgilendiği katalog, ürün, renk vb. bilgilerin toplanmasını kapsamaktadır. Uygulama üretici, kuyruklar, tüketici, veri tabanları ve CollectionBaseDB modüllerinden oluşmaktadır. Bu uygulamada da C# programlama dili ve .net Core yazılım çerçevesi tercih edilmiştir. Uygulamanın istek modelleri Çizelge 4.26, 4.27, 4.28, 4.29, 4.30, 4.31 ve 4.32 ile çalışmaya eklenmiştir.

Öneri sistemi uygulaması, sistem verilerinin toplanması için geliştirilen uygulama ile aktivite verilerinin toplanması için geliştirilen uygulamanın topladığı verilerden faydalanarak çalışmaktadır. Öneri sistemi uygulama tasarımı, Şekil 4.46 ile çalışmaya

eklenmiştir. Veri toplama uygulamalarından gelen veriler ise Raw Data isimli modül ile bu uygulamaya girecektir. Uygulamaya giriş yapan işlenmiş veriler, bu uygulama sayesinde öneri yöntemlerine uygun olarak işlenip bilgiye dönüştürülecektir.

Öneri sistemi uygulaması, müşteri segmentasyonu ile birincil faydalarından birini göstermektedir. E-ticaretin her iki tarafının aynı anda tatmin olamama problemine bir çözüm olarak sunulan müşteri segmentasyonu sayesinde müşteriler tercihlerine göre sınıflandırılmakta ve pazarlama stratejilerinin belirlenmesinde etkili olmaktadır. Uygulama, RFM analizi ile de müşterileri önceki satın alma geçmişine göre sınıflandırabilmektedir. Buna göre müşteriler yenilik, sıklık ve parasal açıdan belirlenen parametreler doğrultusunda sınıflara ayrılmaktadır. Sınıflandırma örneği, Şekil 4.48 ile çalışmaya eklenmiştir. Dolayısıyla bu uygulamayı kullanan e-ticaret firmaları, mevcut ve sınıflandırma dahilindeki potansiyel müşterilere nasıl davranacağını ve hangi önerileri sunması gerektiğini önceden bilirken farklı müşteri grupları için de farklı stratejiler üretebilme fırsatını elde edebilmektedir.

Öneri sistemi uygulaması; probleme yani öneri amacına ve müşteri grubuna bağlı olarak birliktelik analizi, içerik tabanlı filtreleme ve işbirlikçi filtreleme yöntemlerinden bir ya da biracını bir arada kullanarak çözüm üretmekte ve öneri sunmaktadır.

Birliktelik analizi, ortalama sepet tutarının düşük kalması problemine çözüm olarak sunulmaktadır. Bu analiz, birlikte satın alınabilecek ürünleri tespit etmekte faydalıdır. Uygulama, bu analizden aldığı veriler doğrultusunda, sepet ekranındaki kullanıcıya, sepetindeki ürünler ile satın alabileceği benzer ve farklı kategorilerdeki ürünleri göstermektedir. Kullanıcıyı sepete yeni bir ürün daha eklemeye teşvik eden bu yöntem ile ortalama sepet tutarı artırılabilir. Ürün bazlı birliktelik kuralı Şekil 4.52, birliktelik kuralı uygulama tasarımı ise Çizelge 4.42 ile çalışmaya eklenmiştir.

Uygulamada yer verilen içerik tabanlı filtreleme, e-ticaret sitelerinin üye toplama problemine bir çözüm olarak geliştirilmiştir. Özellikle sektöre yeni giriş yapan firmalar için oldukça verimlidir. Bu öneri yöntemi, ilgili e-ticaret sitesini ya da mobil uygulamasının zenginliğini göstererek müşteriye cazip bir mağaza sunmaktadır. Bu yöntemde henüz üye olmayan kullanıcılara ve yeni üyelere, ilgilendikleri ürünlerin benzerleri önerilmektedir. Çizelge 4.43 ile yer verilen veri seti incelemesi ve Çizelge

4.44'teki öneriler, yöntemin verimi hakkında fikir vermektedir. İçerik tabanlı filtreleme uygulama tasarımı ise Çizelge 4.45 ile çalışmaya eklenmiştir.

Öneri sistemi uygulamalarının geniş kapsamlı çalışabilmesinin örnekleri, işbirlikçi filtreleme yöntemlerinde görülebilmektedir. İşbirlikçi filtreleme, tekrar eden sipariş sayısının düşük olma problemine karşı bir çözüm olarak sunulmaktadır. Aynı zamanda üye kaybının da önüne geçmektedir. Kullanıcılar beğendikleri ürünlerin benzerlerini de beğenme eğiliminde oldukları için bu yöntem, alışveriş devamlılığı yaratma amacı içeren stratejiler için son derece faydalıdır. Öğe tabanlı işbirlikçi filtrelemede kullanıcıların beğenme alışkanlıkları dikkate alınmakta ve benzer beğenme yapısı gösteren ürünler tespit edilmektedir. Kullanıcı tabanlı işbirlikçi filtreleme ile de beğenme alışkanlıkları üzerinden benzer kullanıcıların tespit edilmesi ve bu doğrultuda öneri kalitesinin artırılması hedeflenmektedir.

Model tabanlı işbirlikçi filtreleme ise satış sonrası sürecin planlanmasına yardımcı olmaktadır. Model tabanlı İF, bir kullanıcının, değerlendirme yapmadığı ürünlere vereceği değerlendirme derecesini tahmin etmektedir. Bu yöntem, özellikle yoruma açık ürünler satan firmalar için önem arz etmektedir. Satış artışına destek olması amacıyla uygulama tarafından önerilen bu yöntem, ürünlerdeki olumlu yorum ve yüksek puan oranlarının iyileşmesine katkı sağlamaktadır. Bu iyileşme, belirli bir müşteri kitlesine sahip e-ticaret firmalarının farklı potansiyel müşteri gruplarına hitap etme sürecinde sorun yaşaması karşısında etkili bir çözüm olarak sunulmaktadır. Model tabanlı İF uygulama tasarımı, Şekil 4.59 ile çalışmaya eklenmiştir.

Bahsi geçen öneri sistemi uygulamalarının etkili bir şekilde çalışabilmesi için gerekli alt yapı sağlanmalıdır. Öncelikli olarak bağımsız bir veri deposu edinilmeli ve site üzerinden uygulamaya, bu veri deposuna erişim imkânı verilmelidir. Güvenli ve hızlı bir iletişim kanalı oluşturulduktan sonra uygulamalar sorunsuz bir şekilde çalışabilecektir. Bu süreçte verilerin sık bir periyot ile sürekli olarak güncellenmesi ve tekrar eğitilmesi gerektiği unutulmamalıdır.

KAYNAKLAR

- [1] **Chen, M. S., Han, J. & Yu, P. S.** (1996). Data mining: an overview from a database perspective, *IEEE Transactions on Knowledge and data Engineering*, 8(6), 866-883.
- [2] **Tekerek, A.** (2011). Veri madenciliği süreçleri ve açık kaynak kodlu veri madenciliği araçları, *Akademik Bilişim*, 11, 2-4.
- [3] **Tang, Z. & Maclennan, J.** (2005). *Data mining with SQL Server 2005*. John Wiley & Sons.
- [4] **Ayık, Y. Z., Özdemir, A. & Yavuz, U.** (2007). Lise türü ve lise mezuniyet başarısının, kazanılan fakülte ile ilişkisinin veri madenciliği tekniği ile analizi. *Atatürk Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 10(2), 441-454.
- [5] **Han, J., Pei, J. & Kamber, M.** (2011). *Data mining: concepts and techniques*. Elsevier.
- [6] **Oğuzlar, A.** (2004). *Veri Madenciliğine Giriş*. Bursa: Ekin Kitabevi.
- [7] **Tüzüntürk, S.** (2010). *Veri Madenciliği ve İstatistik*.
- [8] **Bigus, J. P.** (1996). *Data Mining With Neural Networks: Solving Business Problems From Application Development To Decision Support*. McGraw-Hill, Inc.
- [9] **Baysal, A. C.** (2008). *Bayi değerlendirmesi için veri madenciliği uygulaması*. (Doctoral dissertation). Fen Bilimleri Enstitüsü.
- [10] **Akpınar, H.** (2000). Veri tabanlarında bilgi keşfi ve veri madenciliği, *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, C.
- [11] **Curtarolo, S., Morgan, D., Persson, K., Rodgers, J. & Ceder, G.** (2003). Predicting crystal structures with data mining of quantum calculations. *Physical review letters*, 91(13), 135503.
- [12] **Aynekin, G.** (2006). *İnternet içerik madenciliğinde yapay sinir ağları ve bir uygulama*. (Doctoral dissertation). Bursa Uludag University, Bursa.
- [13] **Özçakar, N.** (1998). Genetik algoritmalar. *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, 27(1), 69-82.
- [14] **Olson, D. L. & Delen, D.** (2008). Memory-Based Reasoning Methods. In *Advanced data mining techniques* (pp. 39-52). Springer: Berlin, Heidelberg.
- [15] **Webb, G. I., Keogh, E. & Miikkulainen, R.** (2010). Naïve Bayes. In *Encyclopedia of machine learning* (Vol 15, pp. 713-714).
- [16] **Karypis, G., Han, E. H. & Kumar, V.** (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32(8), 68-75.

- [17] **Han, J., & Fu, Y.** (1999). Mining multiple-level association rules in large databases. *IEEE Transactions on Knowledge and Data Engineering*, 11(5), 798-805.
- [18] **Özekes, S.** (2003). Veri madenciliği modelleri ve uygulama alanları.
- [19] **Albayrak, A. S., & Yilmaz, S. K.** (2009). Veri madenciliği: karar ağacı algoritmaları ve imkb verileri üzerine bir uygulama. *Suleyman Demirel University Journal of Faculty of Economics & Administrative Sciences*, 14(1).
- [20] **Ricci, F., Rokach, L. & Shapira, B.** (2011). Introduction to recommender systems handbook. In *Recommender Systems Handbook* (pp. 1-35). Springer : Boston, MA.
- [21] **Sarno, R., Dewandono, R. D., Ahmad, T., Naufal, M. F. & Sinaga, F.** (2015). Hybrid Association Rule Learning and Process Mining for Fraud Detection. *IAENG International Journal of Computer Science*, 42(2).
- [22] **Thorat, P. B., Goudar, R. M. & Barve, S.** (2015). Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4), 31-36.
- [23] **Su, X., & Khoshgoftaar, T. M.** (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*.
- [24] **Döşlü, A.** (2008). *Veri madenciliğinde market sepet analizi ve birliktelik kurallarının belirlenmesi.* (Yüksek lisans tezi). Fen Bilimleri Enstitüsü.
- [25] **Vaidya, N. & Khachane, A. R.** (2017). Recommender systems-the need of the ecommerce ERA. In *2017 International Conference on Computing Methodologies and Communication (ICCMC)*, (pp. 100-104). IEEE, July.
- [26] **Zuva, T., Ojo, S. O., Ngwira, S. & Zuva, K.** (2012). A survey of recommender systems techniques, challenges and evaluation metrics. *International Journal of Emerging Technology and Advanced Engineering*, 2(11), 382-386.
- [27] **Mustafa, N., Ibrahim, A. O., Ahmed, A. & Abdullah, A.** (2017). Collaborative filtering: Techniques and applications. In *2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE)*, (pp. 1-6). IEEE, January
- [28] **Park, S. T., & Pennock, D. M.** (2007). Applying collaborative filtering techniques to movie search for better ranking and browsing. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 550-559), August
- [29] **Ekstrand, M. D., Riedl, J. T. & Konstan, J. A.** (2011). *Collaborative Filtering Recommender Systems*. Now Publishers Inc.
- [30] **Güler, Ş.** (2019). *Öneri sistemleri ve E-ticarette öneri sistemlerinin kullanımı.* (Yüksek lisans tezi). Sakarya Üniversitesi.
- [31] **Bozkurt, M. B. & AÇI, Ç. İ.** (2021). Öneri Algoritmalarının Film Önerme Problemi Üzerinde Karşılaştırılması: MovieLens Örneği. *Bilgisayar Bilimleri ve Teknolojileri Dergisi*, 1(2), 36-42.

- [32] Uçar, M. (2021). *E-ticaret siteleri için bulanık mantık ve makine öğrenmesi tabanlı bir öneri sistemi*. (Master's thesis). Sakarya Üniversitesi.
- [33] Özdemir, Y. (2021). *Kullanıcı ve öge temelli yöntemlerin birlikte kullanıldığı işbirlikçi filtreleme tekniği geliştirilmesi* (Yüksek lisans tezi). Fatih Sultan Mehmet Vakıf Üniversitesi, Lisansüstü Eğitim Enstitüsü.
- [34] Sánchez-Moreno, D., Gil, A. B. & Moreno, M. N. (2013). TV-Seriesrec: a recommender system based on fuzzy associative classification and semantic information. *Trends In Practical Applications Of Agents And Multiagent Systems*, 201-208.
- [35] Lian, R. (2011). The construction of personalized Web page recommendation system in e-commerce. In *2011 International conference on computer science and service system (CSSS)* (pp. 2687-2690). IEEE, June
- [36] Zhang, Z., Lin, H., Liu, K., Wu, D., Zhang, G. & Lu, J. (2013). A hybrid fuzzy-based personalized recommender system for telecom products/services. *Information Sciences*, 235, 117-129.
- [37] Hwangbo, H., Kim, Y. S. & Cha, K. J. (2018). Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, 28, 94-101.
- [38] Faryal, A., Tauqir, A., Martinez-Enriquez, A. M. & Aslam, M. (2015, December). Data mining based recommendation system using social websites. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 1, pp. 365-368). IEEE.
- [39] Lee, H., Choi, S. Y. & Kang, Y. S. (2009). Formation of e-satisfaction and repurchase intention: Moderating roles of computer self-efficacy and computer anxiety. *Expert Systems with Applications*, 36(4), 7848-7859.
- [40] Chugh, S. & Baweja, V. R. (2020). Data mining application in segmenting customers with clustering. In *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)* (pp. 1-4). IEEE, February.
- [41] Christy, A. J., Umamakeswari, A., Priyatharsini, L. & Neyaa, A. (2021). RFM ranking—An effective approach to customer segmentation. *Journal of King Saud University-Computer and Information Sciences*, 33(10), 1251-1257.
- [42] Zhou, X., Zhang, Z. & Lu, Y. (2011). Review of Customer Segmentation method in CRM. In *2011 International Conference on Computer Science and Service System (CSSS)* (pp. 4033-4035). IEEE, June
- [43] Chouseinoglou, O. & Şahin, İ. (2019). Metin madenciliği, makine ve derin öğrenme algoritmaları ile web sayfalarının sınıflandırılması. *Yönetim Bilişim Sistemleri Dergisi*, 5(2), 29-43.
- [44] Hejlsberg, A., Wiltamuth, S. & Golde, P. (2003). *C# language specification*. Addison-Wesley Longman Publishing Co. Inc.
- [45] Url-1 < <https://github.com/dotnet/core/blob/main/LICENSE.TXT>>, erişim tarihi 18.07.2022.

- [46] **Url-2** < <https://www.rabbitmq.com/protocols.html>>, erişim tarihi 18.07.2022.
- [47] **Url-3** < <https://tr.wikipedia.org/wiki/MongoDB#MongoDB>>, erişim tarihi 18.07.2022.
- [48] **Url-4** < <https://www.techopedia.com/definition/1312/representational-state-transfer-rest>>, erişim tarihi 18.07.2022.
- [49] **Url-5** < <https://tr.wikipedia.org/wiki/REST>>, erişim tarihi 18.07.2022.
- [50] **Sivri, E. Ş.** (2015). *Veri madenciliği: e-ticaret için ürün tavsiye sistemi geliştirilmesi*. (Master's thesis). İstanbul Ticaret Üniversitesi.
- [51] **Abdullah, W.** (2019). *E-ticaret için ürün tavsiye sistem geliştirilmesi*. (Master's thesis). Fen Bilimleri Enstitüsü.
- [52] **Kaya, F.** (2012). *İşbirlikçi filtrelemeye dayalı web tabanlı öneri sistemi geliştirilmesi*. (Master's thesis). Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü.
- [53] **Miyahara, K. & Pazzani, M. J.** (2002). Improvement of collaborative filtering with the simple Bayesian classifier. *Information Processing Society of Japan*, 43(11).
- [54] **Herlocker, J. L., Konstan, J. A., Borchers, A. & Riedl, J.** (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 230-237). August.
- [55] **Hu, R. & Lu, Y.** (2006). A hybrid user and item-based collaborative filtering with smoothing on sparse data. In *16th International Conference on Artificial Reality and Telexistence--Workshops (ICAT'06)* (pp. 184-189). IEEE, November.
- [56] **Anıl, U. T. K. U. & Akcayol, M. A.** (2017). Öğrenebilen ve Adaptif Tavsiye Sistemleri İçin Karşılaştırmalı ve Kapsamlı Bir İnceleme. *Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi*, 33(3), 13-34.

EKLER

EK A: Aktivite Verileri Dokümanı



EK A

Aktivite Verilerinin Toplanması Dokümanı

Çizelge 0.1’da genel istek modeline ulaşılabilir.

Aktivite verileri tek bir model ile toplanmaktadır. Model içerisindeki “Action” parametresi ile hangi eylem ait olduğu ayrımı yapılmaktadır. “AppKey” parametresi verinin toplandığı site için eşsiz anahtar değeridir. “Payload” parametresi her bir metot için özel modeli temsil etmektedir. Her bir aksiyonun için model örnekleri tez içerisinde mevcuttur.

Örnek istekler;

Oluşturulan genel istek fonksiyonu ile tüm aksiyonlar için istek yapılabilir. Belge sonunda genel istek fonksiyonları paylaşılmıştır.

Ürün görüntüleme aksiyonu ile yapılan istek;

```
dataCollection("view",{ "Payload": {  
    "UserID":"","  
    "SessionID":"85948asfv8784GFAS84418AS",  
    "type":"Product",  
    "value":"889526854845",  
    "ViewRange":10  
}  
})
```

Arama çubuğu ile yapılan istek;

```
dataCollection("search",{ "Payload": {  
    "UserID":"444",  
    "SessionID":"","  
    "value":"T-sihrt"  
}})
```

Sepete ekleme aksiyonu ile yapılan istek;

```
dataCollection("basket",{ "Payload": {  
    "UserID":"799",  
    "ProductID":"5566",  
    "Type":"Add",  
    "BasketInfo":{  
        "CurrentPrice":"29.90",  
        "OldPrice":"39.90",
```

```
"InStock":true,  
"IsQuickLook":false,  
"Quantity":3  
}  
}))
```

Favorilere ekleme işlemi ile yapılan istek;

```
dataCollection("wish",{ "Payload": {  
  "UserID":"799",  
  "SessionID":"T5435SftSC2345",  
  "ProductID":"7744",  
  "Type":"Add",  
  "WishInfo":{  
    "CurrentPrice":"229.90",  
    "OldPrice":"344.90",  
    "InStock":true,  
    "IsQuickLook":false  
  }  
}))
```

Genel istek fonksiyonları;

Bir “JavaScript” belgesi olarak kaydedilip web sitesi üzerindeki tüm sayfalardan erişilebilir konumda olmalıdır. Yukarıda verilen örnekler ile kullanılarak geliştirilen uygulamaya veri aktarımı yapılabilir.

```

const baseUrl = "domain/api/v1/Request"

async function dataCollection(appkey,method, payload) {
  const reqData = {
    "AppKey": appkey,
    "Action": method,
    "Payload": payload.Payload
  }
  activity(reqData);
}

async function activity(reqData) {
  if (reqData.AppKey == "" || reqData.AppKey == undefined) {
    var err = { success: false, errors: ["Please set init func."] };
    console.log(err)
  } else {
    await sendHttpRequest('POST', baseUrl, reqData)
      .then(responseData => {
        if (responseData.success != true) {
        }
      })
      .catch(err => {
        console.log(err);
      });
  }
}

const sendHttpRequest = (method, url, data) => {

  const promise = new Promise((resolve, reject) => {
    const xhr = new XMLHttpRequest();
    xhr.open(method, url);
    xhr.responseType = 'json';
    if (data) {
      xhr.setRequestHeader('Content-Type', 'application/json');
    }
    xhr.onload = () => {
      if (xhr.status >= 400) {
        reject(xhr.response);
      } else {
        resolve(xhr.response);
      }
    };
    xhr.onerror = () => {
      reject('Something went wrong!');
    };
    xhr.send(JSON.stringify(data));
  });
  return promise;
};

```

ÖZGEÇMİŞ

Ad-Soyad: Abdullah Özkılıç

Doğum Tarihi ve Yeri:

E-posta:

ÖĞRENİM DURUMU:

- **Lisans** : 2018, Celal Bayar Üniversitesi, Teknoloji Fakültesi, Yazılım Mühendisliği
- **Yüksek Lisans** : Bursa Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği

MESLEKİ DENEYİM VE ÖDÜLLER:

- Bursalı Tekstil – Software Specialist (2022-Halen)
- Sorsware – Software Developer (2020-2022)

TEZDEN TÜRETİLEN ESERLER, SUNUMLAR VE PATENTLER:

- **Özkılıç A., Altuntaş V.** (2021, September) Data Acquisition On E-Commerce sites. *In 2021 Umteb International Conference on Vacation & Technical Sciences-XI*