

**T.C.
İSTANBUL TİCARET ÜNİVERSİTESİ
MÜHENDİSLİK VE TASARIM FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

İÇERİĞE DAYALI FİLM ÖNERİ SİSTEMLERİ

Koray YALÇIN

Bilgisayar Mühendisliği Programına Hazırlanan

BİTİRME PROJESİ

İSTANBUL, 2020

**T.C.
İSTANBUL TİCARET ÜNİVERSİTESİ
MÜHENDİSLİK VE TASARIM FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

İÇERİĞE DAYALI FİLM ÖNERİ SİSTEMLERİ

Koray YALÇIN

Bilgisayar Mühendisliği Programına Hazırlanan

BİTİRME PROJESİ

Proje Danışmanı : Dr. Arzu KAKIŞIM

İSTANBUL, 2020

ÖNSÖZ

Tez çalışmamın her aşamasında sürekli ilgilenen ve yol gösteren, sabırla beni her zaman motive eden ve güven veren değerli danışman hocam Dr. Arzu KAKIŞIM'a, jüri üyelerine ve her zaman beni destekleyen aileme sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖNSÖZ.....	i
İÇİNDEKİLER.....	ii
TABLOLAR LİSTESİ.....	iv
ŞEKİLLER LİSTESİ.....	v
KISALTMALAR LİSTESİ.....	vi
ÖZET.....	vii
ABSTRACT.....	viii
1. GİRİŞ.....	1
2. VERİ MADENCİLİĞİ.....	3
2.1. VERİ MADENCİLİĞİ MODELLERİ.....	4
2.1.1. TAHMİN EDİCİ MODEL.....	4
2.1.2. TANIMLAYICI MODEL.....	5
2.2 VERİ MADENCİLİĞİ TEKNİK VE ALGORİTMALARI.....	6
2.2.1. KARAR AĞAÇLARI.....	6
2.2.2. REGRESYON ANALİZİ.....	7
2.2.3. LOJİSTİK REGRESYON.....	7
2.2.4. BAYES.....	7
2.2.5. YAPAY SİNİR AĞLARI.....	8
2.2.6. K EN YAKIN KOMŞU.....	9
2.2.7. K-ORTALAMALAR.....	9
2.2.8. DESTEK VEKTÖR MAKİNESİ (SVM).....	9
3. VERİ MADENCİLİĞİ UYGULAMA ALANLARI.....	10
3.1. BANKACILIK.....	10
3.2. PAZARLAMA.....	10
3.3. SİĞORTACILIK.....	10
3.4 SAĞLIK.....	11
4. ÖNERİ SİSTEMLERİ.....	11
4.1. İŞBİRLİKÇİ FİLTRELEME.....	12
4.1.1. KULLANICI-KULLANICI İŞBİRLİĞİNE DAYALI FİLTRELEME.....	12
4.1.2. ÖĞE-ÖĞE İŞBİRLİĞİNE DAYALI FİLTRELEME.....	13
4.2. İÇERİĞE DAYALI FİLTRELEME.....	13
4.3 HİBRİT FİLTRELEME.....	14
4.4. ÖNERİ SİSTEMLERİNDE VERİNİN ANALİZİ.....	15
4.4.1. VERİ ÖN İŞLEME.....	15
4.4.2. VERİ ANALİZİ.....	17
4.4.3. SONUÇ YORUMLAMA.....	17
5. BENZER ÇALIŞMALAR.....	18
6. UYGULAMA.....	19
6.1. VERİNİN TOPLANMASI.....	19
6.2. SİSTEMİN TASARLANMASI.....	20
6.2.1. KOSİNÜS BENZERLİĞİ İLE ÖNERME.....	21
6.2.2. WORD2VEC İLE ÖNERME.....	25
7. SONUÇ.....	30
KAYNAKLAR.....	31
EKLER.....	34
EK-1 İkili filtrelemeye göre öneri.....	34
EK-2 Üçlü filtrelemeye göre öneri.....	34

EK-3 Tüm parametlere göre öneri.....	35
ÖZGEÇMİŞ.....	36

TABLÖLAR LİSTESİ

Tablo 6.1 Filmin adına göre benzerlik değerleri.....	22
Tablo 6.2 Filmin türüne göre benzerlik değerleri.....	23
Tablo 6.3 Filmin oyuncularına göre benzerlik değerleri.....	24
Tablo 6.4 Filmin yönetmenine göre benzerlik değerleri.....	25
Tablo 6.5 İki yöntemin karşılaştırılması.....	28

ŞEKİLLER LİSTESİ

Şekil 2.1 Veri madenciliğindeki bazı farklı disiplinler.....	3
Şekil 4.1 İşbirlikçi ve İçeriğe dayalı filtreleme arasındaki fark.....	14
Şekil 4.2 Verinin analizindeki süreçler.....	15
Şekil 6.1 Kullanılan kütüphaneler.....	20
Şekil 6.2 İki vektör arasındaki kosinüs benzerliği.....	21
Şekil 6.3 Filmin adına göre önerilen filmler.....	22
Şekil 6.4 Filmin türüne göre önerilen filmler... ..	23
Şekil 6.5 Filmin oyuncularına göre önerilen filmler.....	23
Şekil 6.6 Filmin yönetmenine göre önerilen filmler.....	24
Şekil 6.7 Veri setinin komşuluk listesi formatlı hali.....	26
Şekil 6.8 Veri setinin kenar listesi formatlı hali.....	26
Şekil 6.9 Graph yapısı.....	27
Şekil 6.10 Word2Vec yöntemine göre film önerisi	27
Şekil 6.11 Anlamsal benzerliğe örnek.....	28
Şekil 6.12 Anlamsal benzerliğe örnek.....	29

KISALTMALAR LİSTESİ

IMDb : Internet Movie Database

API : Application Programming Interface

YSA : Yapay Sinir Ağları

SVM : Support Vector Machine

ÖZET

Son yıllarda internetin de hayatımıza iyice yerleşmesiyle birlikte film izlemeyi seven kullanıcılar için de çeşitlilik miktarı fazlasıyla artmıştır. Kullanıcıların artan bu film çeşitliliğiyle birlikte büyük veri içerisinde kendisine hitap eden en uygun filmi bulup izlemesi uzun zaman almaktadır. Bu nedenle herhangi bir filmi, bu filmi izleyecek en uygun kullanıcıya önermek için öneri sistemleri geliştirilmiştir. Öneri sistemleri, kullanıcıların hareketleri, hobileri ve ayıt edici niteliklerinden faydalanarak kullanıcılar ile filmler arasında ilişkiler kurup, en uygun filmi kullanıcıya önerilebilir.

Bu tezde kullanıcıların izledikleri filmlerin adına, türüne, oynayan oyuncularına ve yönetmenine göre veri seti içerisinde benzer filmler önerilecektir. Veri seti film, dizi, animasyon gibi ürünleri veri tabanında bulunduran IMDb sitesinden alınmıştır. Öneride bulunurken kosinüs benzerliği ve Word2Vec olmak üzere iki ayrı yöntemde önerilerde bulunup sonuçlar karşılaştırılmıştır. Öneri sistemleri, öneride bulunurken içeriğe dayalı filtreleme, işbirlikçi filtreleme ve hibrit filtreleme olmak üzere üç tekniğe dayanır. Bu tezde içeriğe dayalı filtreleme tekniği kullanılmıştır.

ABSTRACT

In recent years, as the internet has settled in our lives, the amount of diversity has increased for users who like watching movies. With this increasing variety of movies, it takes a long time for users to find and watch the most suitable movie that appeals to them from big data. Therefore, recommendation systems have been developed to recommend any movie to the most suitable user to watch this movie. By using the recommendation systems, the actions of the users, their hobbies and their distinctive qualities, the most appropriate movie can be recommend to the user by establishing relationships with the users.

In this thesis, similar films from the data set will be proposed according to the name, genre, actors and director of the films watched by users. The data set was taken from the IMDb, which contains products such as movies, series, and animation in the database. While making the recommend, the cosine similarity and Word2Vec made recommends in two different methods and the results were compared. Recommendation systems are based on three techniques: content-based filtering, collaborative filtering and hybrid filtering when making suggestions. In this thesis, content based filtering technique is used.

1. GİRİŞ

Zamanla veri depolama birimlerinin ucuzlaması, bulut depolama alanlarının ortaya çıkması ve kapasite boyutlarının iyice artmasıyla birlikte elektronik ortamda bulunan veri miktarı her geçen gün artmıştır. Bu büyük veri içerisinde istenilen bilgiye ulaşmak ve onu işlemek önemli bir problem olmaya başlamıştır [44]. Bu sayede öneri sistemleri hızla popüler bir hale gelerek, kullanıcılardan toplanan verilerin veri madenciliği algoritmaları yardımıyla işlenip büyük veriden yalnızca istenilen anlamlı bilgilerin elde edilmesini sağlamıştır. Öneri sistemleri büyük veriyle başa çıkmak amacıyla geliştirilmiş bir tekniktir [39].

İnternetin iyice yaygınlaşmasıyla birlikte film izlemeyi seven kullanıcılar içinde film çeşitliliği epeyce artış göstermiştir. Kullanıcının internette gördüğü her filmi izleyip kendisine uygun olup olmadığını denemesi imkansız hale gelmiştir. Öneri sistemleri, kullanıcının daha öncesinden izlediği film bilgilerini işleyerek daha önce izlemediği ama izlediği filmlerle benzer nitelikleri taşıyan filmleri kullanıcıya önermeyi amaçlar [40].

Öneri sistemleri, eski dönemlerde yalnızca sorgu amaçlı çalışmaktaydı ve filtreleme tekniklerinden yalnızca içeriğe dayalı filtreleme kullanılmaktaydı [45]. Günümüzde ihtiyaçların artmasıyla birlikte içeriğe dayalı filtrelemenin yanında işbirlikçi bazlı filtreleme ve hibrit filtreleme yöntemleri geliştirilmiştir. İçeriğe dayalı öneri sistemlerini kısaca tanımlamak istersek, kullanıcıların daha önceden izlediği filmin türü, oyuncular, yönetmeni, çıkış yılı gibi bilgilerine benzer filmler önermeyi amaçlar. İşbirlikçi filtreleme sistemlerinde ise, kullanıcıların daha öncesinden izledikleri filmin bilgileriyle birlikte bu filmi izleyen diğer kullanıcıların bilgilerinden de faydalanarak öneri yapmayı amaçlar. Hibrit filtreleme sistemleri ise, bu iki yöntemin olumlu yönlerini birleştirerek öneri sunmayı amaçlamıştır [24].

Tez yedi ana başlıktan oluşmaktadır. Bölüm 1 tezin giriş bölümüdür. Burada öneri istemlerinin kullanım amaçları, hangi ihtiyaçlardan doğduğu ve hangi tekniklerden oluştuğu anlaşılmıştır. Bölüm 2’de veri madenciliğinin ne olduğu, hangi algoritmaları ve yöntemleri kullandığına değinilmiştir. Bölüm 3’te veri madenciliğini sık kullanan bazı uygulama alanları anlatılmıştır. Bölüm 4’te öneri sistemleri detayları olarak hangi amaçla geliştirildiği anlatılmış ve hangi tekniklere dayanarak öneride bulunulduğu detaylandırılmıştır. Bölüm 5’te daha önceki öneri sistemleriyle ilgi yapılan çalışmalar incelenmiştir. Bölüm 6’da iki farklı öneri yöntemi kullanarak

kullanıcının belirttiđi filme g re kendisine en uygun olan filmleri sıralayan bir uygulama yazılmıřtır. B l m 7 ise sonu  kısmıdır. Bu kapsamda  ıkan sonu lar incelenmiř ve  nerilerde bulunulmuřtur.

2. VERİ MADENCİLİĞİ

Son yıllarda teknolojinin gelişmesiyle birlikte artık çoğu firma işlerini bilgisayar kullanarak yapmaktadır. Firmaların topladıkları verilerin sürekli artması ve veriyi saklayacak kapasitenin hızla azalmasıyla birlikte yeni çözümler orta atılmıştır. Günümüzde veriler tek bir bilgisayarın işleyebileceğinden çok fazladır. Verilerin kısa zamanda çok fazla büyümesinden dolayı yeni yöntemlere ihtiyaç olmuştur. Geleneksel sorgu veya raporlama araçlarının çok miktardaki veriler karşısında yetersiz kalmasından dolayı veri madenciliği gelişmiştir [3].

Veri madenciliği, büyük veriden (big data) anlamlı, doğru, temiz ve kullanışlı verileri ayıklamak ve bu veriler üzerinden gelecekteki olayları tahmin etme amacıyla kullanılan bir yöntemdir. Bu sayede firmalar kullanıcılarının geçmişte tükettiği ürünlere göre gelecekte nasıl bir ürün tüketeceğinin bilgisini öğrenmiş olur. Veri madenciliği genel olarak, büyük boyutlardaki verilerin birbirleriyle arasındaki ilişkilerden faydalanarak aralarında bir bağlantı kurmayı amaçlar. Bu sayede veri tabanı içinde bulunan anlamlı bilgilerin ayıklanarak veri analizi yapılmasına olanak sağlar. Bu işlemler birçok alanda kullanılır. Bu alanlardan bazıları Şekil 2.1’de gösterildiği gibidir [1].



Şekil 2.1. Veri madenciliğindeki bazı farklı disiplinler

Veri madenciliği, büyük veriden, yalnızca kullanıcının ihtiyacı olduğu verinin çıkarılması için kullanılır. Ayrıca geçmiş verilerden yararlanarak, gelecek ile ilgili anlamlı ve tutarlı tahminler yapmakta kullanılabilir [1].

2.1. VERİ MADENCİLİĞİ MODELLERİ

Veri madenciliğinde tanımlayıcı ve tahmin edici olmak üzere iki model vardır [2].

2.1.1. TAHMİN EDİCİ MODEL

Tahmin edici model, sonuçları daha önceden bilinen verilerden yararlanarak bir model oluşturmayı ve oluşturulan bu modelden faydalanarak sonuçları bilinmeyen veri kümelerinin tahmin edilmesi için kullanılır [4]. Örneğin İstanbul'da ağustos ayının son birkaç yıldaki hava durumunun tutulduğu bir veri tabanı olsun. Bu veri tabanına tahmin edici model uygulayarak gelecek yıllardaki ağustos ayının nasıl geçeceği tahmin edilebilir. Kendi içinde sınıflama, regresyon ve zaman serisi olmak üzere üçe ayrılır.

Sınıflandırma, veri madenciliğinde genellikle sonuçları tahmin etmek amacıyla kullanılır. Gözetimli (supervised) öğrenme şekline sahiptir. Bu yöntemde amaç, etiketlenmemiş ve saklı kalmış verileri tespit ederek sınıflarına ve etiketlerine göre atamaktır [5]. Sınıflandırma işlemi, öğrenme sürecini bitirdiğinde bazı kurallar oluşturur. Bu kurallara göre belirlenen sınıflara atama yapılır. Örneğin havaya atılan bir paranın yazı mı tura mı geleceğini tahmin eder. Yazı ve tura etikettir.

Regresyon, veriler içerisindeki değişkenlerin aralarındaki ilişkileri inceleyerek matematiksel çıkarımlar yapan bir yöntemdir. Bu yöntemde, veriler bağımsız değişken olarak tanımlanır. Amaç, tanımlanan bağımsız değişken ile sonuç verileri arasında ilişki kurularak tahmin yapmayı amaçlar. Regresyon yönteminde girdiler birden çok değişken olabilir. Bu nedenle girdilerin sonuca etkisine bakılarak az etki eden değişkenler çıkartılabilir. Bunları gerçekleştirmek için bazı algoritmalar kullanılır.

Bunlardan bazıları [6]:

- Bayes Sınıflandırması
- Karar Destek Makinaları
- Zaman Serisi Analizi
- Doğrusal Regresyon Analizleri

- Yapay Sinir Ağları
- En Yakın Komşu
- Hatayı Geri Yayma
- Karar Ağaçları

Zaman serisi, zaman içerisinde tekrarlayan ölçümlerden elde edilen değerlerden oluşur. Bu değerler eşit zaman aralıklarında ölçülür. Zaman serisi; ekonomik tahminler, bütçe analizi, kar tahmini ve pazar payı hesaplamak gibi birçok alanda kullanılan bir yöntemdir [18].

2.1.2. TANIMLAYICI MODEL

Tanımlayıcı modeller daha öncesinden bir bilgiye sahip olmadan, veri kümesindeki ilişkileri tanımayı amaçlar. Büyük veri tabanlarında bilgileri incelemek ve örüntüleri tanımlamak için kullanılır [3]. Kendi içinde kümeleme ve birliktelik kuralı olmak üzere ikiye ayrılır.

Kümeleme yöntemi birbirine benzer olan öğelerin aynı kümede olmasını, farklı öğelerin ise farklı kümelerde olmasını amaçlar. Gözetimsiz (Unsupervised) bir öğrenme şekline sahiptir. Kümeleme yönteminin amacı, veri tabanındaki oluşan alt sınıfları tespit etmektir. Verilerin birbirlerine olan benzerlikleri göz önüne alınarak bu yöntem uygulanır [6]. Örneğin bir gazetede ekonomi haberleri ve spor haberlerinin farklı sayfalarda olması buna örnek gösterilebilir.

Birliktelik kuralı, bir veri tabanında birbiriyle ilişkili verileri ve aralarındaki bağlantının büyüklüğünü belirlemek amacıyla kullanılır. Elimizdeki veriyle veri tabanında kaç veriyle birlikte bulunduğunu tespit eder ve bu bilginin orantılanmasını yapmaya yarar. Bu yöntem özellikle marketlerde çok yaygın kullanılan bir yöntemdir [6]. Örneğin markete giden bir müşterinin ekmeğin yanında hangi oranda yumurta aldığını bulmayı amaçlar. Bu orana göre marketler geliştirecekleri stratejiye göre çok oranda olan ürünleri yan yana koymayı seçerek satış rakamlarını yükseltmek isteyeceklerdir.

2.2. VERİ MADENCİLİĞİ TEKNİK VE ALGORİTMALARI

Veri madenciliği algoritmaları, verinin toplanması ve bu verileri uygun algoritmalara göre modellemek amacıyla kullanılır. Veri tabanında yer alan her bir veriye yaklaşım verinin türüne ve istenilen sonuca göre değişiklik gösterebilir. Bu nedenle veri madenciliğinde birçok teknik ve algoritma kullanılır [6]. Bu tekniklerden bazıları şunlardır:

- Karar Ağaçları
- Regresyon Analizi
- Lojistik Regresyon
- Bayes
- Yapay Sinir Ağları
- K-En Yakın Komşular
- K-Ortalamalar
- Destek Vektör Makinesi (SVM)

2.2.1. KARAR AĞAÇLARI

Karar ağaçları, belli bir sınıfa atanmış verilerin tümevarım yöntemi kullanarak ağacın dallanmasına benzer bir şekilde gerçekleştirmesidir [7]. Karar ağaçlarının amacı, çok miktardaki veriyi, basit karar verme aşamalarına sokarak veriyi küçültmeyi amaçlar. Bu sayede sonuç verileriyle diğer veriler daha benzer hale gelir [8].

Karar ağaçları dallar, yapraklar ve düğümlerden oluşur. Bu teknikte ağaç meydana geldikten sonra kökten yaprağa doğru uzayan kurallar bu yapıya tanımlanır. Bu yöntem veri madenciliğinde çok sık kullanılan bir yöntemdir. Karar ağaçlarının bazı avantajları [9,10]:

- Analiz etmesi kolaydır,
- Eski ya da şimdiki duruma göre gelecekteki olayları tahmin edebilmek için kurallar oluşturur,
- Yalnızca alt gruba ait verilerin birbirleriyle ilişkilerini tanımlar,

- Matematiksel veriler üzerinde işlem yapılabilir.

Karar ağaçlarının avantajları olduğu gibi dezavantajları da vardır:

- Ezbere öğrenme (over-fitting) problemi oluşabilir,
- Başta karmaşık bir ağaç üretilirse dallar da onu takip edeceğinden iyice karmaşık bir hale gelebilir.

2.2.2. REGRESYON ANALİZİ

Regresyon analizinde, bir bağımlı değişken olmakla birlikte bir veya birden fazla bağımsız değişkenler olabilir. Bu değişkenler sayılıp ölçülebilen değişkenlerdir [6]. Regresyon analizinin, tek değişkenli doğrusal, tek değişkenli doğrusal olmayan ve çoklu regresyon gibi sınıfları vardır. Bu yöntem, iki değişkenin birbirleri arasındaki ilişkinin türünü, oranını ve gelecekteki halini tahmin etme gibi ihtiyaçlarda kullanılır [11].

2.2.3. LOJİSTİK REGRESYON

Lojistik regresyon, genellikle finansal alanlarda başarı oranını tahmin etmek için kullanılır. Bu yöntem bağımlı değişkenin, bağımsız değişkenlere neden-sonuç bağıntısını belirlemek için kullanılır [12]. Lojistik regresyon analizinin kullanım amacı bağımlı ve bağımsız değişkenler arasındaki ilişkiyi en az değişken kullanarak tahmin etmektir. Finansal alanda bu tahminler genellikle başarılı ise “1” başarısız ise “0” şeklinde kodlanır [6].

2.2.4. BAYES

Bayes, verilerin belirli sınıflara ait olmasını tahmin etmeye dayanır [6]. Bir olayın oluşması, birden çok koşulun olmasına bağlı olduğunda, bu koşulların hangilerinin tetiklendiğini hangilerinin tetiklenmediğini gösterir. Bayes kuralı, koşullu olasılıkların hesabı için kullanılan bir yaklaşımdır. Örneğin insanların evlerinden dışarı çıkmaları için korona virüs için gerekli önlemler alınması ve sokağa çıkma yasağının olmaması gereklidir. Bu örnekte vaka sayılarının oranı ve sokağa çıkma yasağının olmaması koşulları bayes yöntemiyle hesaplanarak olasılık belirlenebilir. Günümüzde bu yöntem birçok alanda kullanılmaktadır. Bunlardan bazıları şöyledir [13,14,15]:

- Metinlerin sınıflandırılmasında
- Şirketlerin log analizinde
- Ses tanıma sistemlerinde
- Şifre kontrolü sistemlerinde

2.2.5. YAPAY SİNİR AĞLARI

Yapay sinir ağları (YSA), insan beynini baz alarak tasarlanmış bir yöntemdir. İnsanlar gibi öğrenme, hatırlama ve düşünebilme gibi davranışları yapabilen çok sayıda sinir hücresinden oluşur. Bu sinir hücreleri arasında sayısız ölçüde/boyutta bağ vardır [16]. Öğrenme işlemi nöronların birbirleri arasındaki bağlantılara bağlıdır. Tıpkı insan beyni gibi zamanla sürekli gelişir. Bu sayede yapay sinir ağlarında öğrenme işlemi sürekli devam eder. Bu ağlarda öğrenme işlemi, eğitime kullanarak oluşur [6].

YSA'yı genel olarak tanımlamak istersek, ağırlıklandırılmış çok sayıda nörondan oluşan matematiksel sistemler bütünüdür. Bu nöronlar insan beynindeki nöronlara denk gelir ve bir ağda birbirlerine bağlanır. Bu yapı genel olarak YSA'yı oluşturur [6].

Yapay sinir ağlarının kullanılmasının birçok avantaj ve dezavantajı vardır. Avantajları [17]:

- YSA bir kez eğitildiği zaman yeni veri kümelerine karşılık verebilir,
- YSA matematiksel modeller olmadan da çalışır,
- YSA'da bir veya birden fazla nöronun zarar görmesi sonuca ulaşmayı engellemez,
- YSA bilinmeyen ilişkileri veriler sayesinde tespit edebilir.

Dezavantajları ise şöyledir [17]:

- Farklı sistemlere uyarlanması zordur,
- Maliyeti yüksektir,
- Düğüm sayısındaki artış sistemde yavaşlamaya neden olabilir.

2.2.6. K EN YAKIN KOMŞU

K-en yakın komşu tekniğinin çalışma mantığı, veri tabanı içerisindeki verileri kendi aralarında kümeleme işlemi yaparak bu kümenin dışında kalan veya yeni gelen veriyi kendisine en uygun olan kümeye sınıflandırarak çalışır. Bu yöntem diğer tüm eğitim programını hatırlar. Bu sayede yeni gelen bir veri olduğu zaman yalnızca bu eğitim programında birbirleriyle eşleşirse sınıflandırma yapar [23].

Bu yöntemin k komşu sayısı, eşik değeri ve benzerlik hesabı gibi parametrelere bağlı olarak performansı değişiklik gösterir [29].

2.2.7. K-ORTALAMALAR

K-ortalamlar tekniğinin çalışma mantığı, n adet veriden oluşan veri kümesini, parametre olarak girilen k adet kümeye bölmek olarak tanımlanabilir. Bu yöntemde amaç, bölme işlemi sonucunda oluşan kümelerin birbirleri arasındaki benzerliklerini arttırıp diğer kümelere benzeme oranını azaltmayı amaçlar.

Bu yöntemin k küme sayısı, benzerlik değeri ve başta belirlenen küme merkezi değerlerine bağlı olarak performansı değişiklik gösterir [29].

2.2.8. DESTEK VEKTÖR MAKİNESİ (SVM)

SVM yönteminin amacı, verilerin birbirleri arasındaki mesafeleri maksimum dereceye getirerek ayırma işlemi yaparak karar sınırını elde etmektir. SVM, veriler arasındaki aralığı maksimuma çıkaran karar sınırını elde ederse, buna göre gelecekte bilinmeyen verileri doğru bir şekilde sınıflandırma imkanı sunar. Eğer elde edilen karar sınırı değeri doğrusal değilse verileri Kernel numarası adı verilen matematiksel bir yöntemle verileri daha yüksek kapasiteli alanlara dönüşümü sağlanır.

SVM'ler; istatistik, sinir ağları ve makine öğrenmesinde kullanılan pek çok tekniği içinde barındırır. Ayrıca SVM'ler son yıllarda popülerliğinin artmasıyla birlikte pek çok alanda kullanılmaya başlanmıştır [30]. Bu alanlardan biri de öneri sistemleri olduğunu söylenebilir.

3. VERİ MADENCİLİĞİ UYGULAMA ALANLARI

Çok miktarda verinin bulunduğu bir veri tabanı için karar verme süresini kısaltmak için veri madenciliği çok sık kullanılan bir yöntemdir. Veri madenciliği, verinin yoğun olarak üretildiği her alanda kullanılabilir [19,20,21]. Bu alanlardan bazıları şunlardır:

3.1. BANKACILIK

- Dolandırıcıların tespiti ,
- Müşterilerin kredi taleplerinin değerlendirilmesi,
- Borç tahmini,
- Döviz kuru tahmini,
- Faiz oranı tahmin etmede kullanılır.

3.2. PAZARLAMA

- Müşterilerin satın aldığı ürünlere göre bir örüntü oluşturulması,
- Sepeti analizi,
- Satış tahmini,
- Çok satan ürünleri belirleme,
- Hangi ürünlerin birlikte satın alındığını belirler.

3.3. SİGORTACILIK

- Riskli müşterilerin tespiti,
- Yeni tarifelere geri dönüş yapacak müşterilerin tahmini,
- Dolandırıcıların tespiti,
- Fiyat belirlenmesi,
- Mevcut veriye göre ilerde ki gelir/gider oranını hesaplama.

3.4 SAĞLIK

- Test sonuçlarının tahmin edilmesi,
- Tedavi süresinin belirlenmesi,
- Hastaları, hastalıklarına göre sınıflama,
- İlaç geliştirme,
- Kalp krizi riski oranının belirlenmesi için kullanılır.

4. ÖNERİ SİSTEMLERİ

Günümüzde birçok platformda neredeyse her insana hitap eden sayısız ürün bulunmaktadır. Bu ürünler arasından kullanıcılar isteklerine uygun ürünleri bulmakta git gide zorlanmaktadır. Bu nedenle, bu sorunu çözmek için öneri sistemleri hızla popüler bir hale gelmiştir [22].

Öneri sistemleri, birçok verinin bulunduğu veri tabanı içinde her bir kullanıcıya hitap eden doğru ürünü öneren ve filtreleme yaptığı için veri boyutunu azaltan sistemler olarak tanımlanabilir [23]. Bir öneri sistemi genel anlamda üç ana başlıktan oluşur. Bunlar; veri seti, öneri motoru ve kullanıcı ara yüzü olarak adlandırılır. Veri setleri adından da anlaşılacağı üzere kullanıcının tercihlerine göre şekillenen ürünlerden oluşmaktadır. Öneri motorları, bu veri seti içerisindeki ürünlerin kullanıcıya en uygun olanlarını öneren sistem olarak çalışır. Kullanıcı arayüzü ise önerilerin kullanıcıya iletildiği ve geri dönüşlerin yapıldığı kısımdır.

Öneri sistemleri, hareket ve davranışlarına göre kullanıcıya ona en uygun olan ürünü önermeyi amaçlar. Bu ürünler müzik, dizi, film, oyun, kitap gibi birçok ürün olabilir. Bu sistemler, sayısız ürünün bulunduğu veri tabanları içerisinde kullanıcısının karar ve tercihlerine en yakın ürünlerin önerilmesini sağlar. Genel olarak öneri sistemlerinin kullanım amaçları:

- Kullanıcının zamandan tasarruf etmesini sağlamak,
- Kullanıcının sadece ilgilendiği ürünlere odaklanmasını sağlamak,
- Kullanıcının para kaybını önlemek,
- Şirketlerin müşteri sayısını arttırmak,

- Müşteri memnuniyetini sağlamak,
- Gelişmiş derecede filtreleme yapmak için kullanılır [23].

Öneri sistemleri genel olarak işbirlikçi filtreleme, içeriğe dayalı filtreleme ve hibrit filtreleme olmak üzere üç yöntemeye dayanır [24].

4.1. İŞBİRLİKÇİ FİLTRELEME

İşbirlikçi filtreleme yöntemi, öneri sistemlerinde en çok kullanılan yöntemdir. Bu yöntemde amaç, ürünleri veya kullanıcıları gruplayarak bu grupların birbirleri arasındaki benzerlikleri bulmayı amaçlar. Bu sayede bulunan benzerlikler kullanıcıya öneri olarak iletilir [23]. Kullanıcı ürünlere olumlu geri dönüş yaptıkça öneri sisteminin önereceği ürün de kullanıcının bu geri dönüş yaptığı ürünlere en benzerleri olacaktır [25]. Bu ürünler ile kullanıcı arasında bir matris oluşur. Önerme işlemi bu matris baz alınarak gerçekleştirilir.

İşbirlikçi filtrelemede erken puanlama, seyreklik ve gri koyun gibi problemler yaşanmaktadır. Erken puanlamada, sisteme yeni giriş yapan kullanıcılar veya geri dönüş yapılmayan ürünler için önerilerin zorunlu olduğu durumdur. Bu durum bir hatadır çünkü ne ürün ne de kullanıcı hakkında sistem bir bilgi yoktur. Seyreklik hatasında ise ürüne olan geri dönüşlerin kullanıcıların birbirleri arasındaki benzerlikleri bulmak için yetersiz kalması sonucu yanlış ürün önerileri oluşur. Gri koyun hatası ise kararsız kalmış kullanıcılara istedikleri türde ürün önerememesi sonucu ortaya çıkmıştır [26].

İşbirlikçi filtreleme yöntemi kullanıcı-kullanıcı işbirliğine dayalı filtreleme ve öge-öge işbirliğine dayalı filtreleme olmak üzere iki tekniğe dayanır.

4.1.1. KULLANICI-KULLANICI İŞBİRLİĞİNE DAYALI FİLTRELEME

Sistemdeki benzer kullanıcıları ilişkilendirir ve bu kullanıcılara onların seçtiği ürünlere benzer ürünleri önerir. Bu yöntemin en büyük dezavantajı, ilişkilendirilecek kullanıcıların bilgilerini analiz etmeyi gerektirmesidir. Bu sebeple, büyük platformlarda bu yöntemi uygulamak zordur.

4.1.2. ÖĞE-ÖĞE İŞBİRLİĞİNE DAYALI FİLTRELEME

Bu filtreleme yöntemi, kullanıcı-kullanıcı işbirliğinden farklı olarak kullanıcılar arasındaki benzerlikten değil öğeler arasındaki benzerlikten faydalanır. Bu sayede, kullanıcının ilgi alanındaki öğelere benzer öğeler önermeyi amaçlar.

4.2. İÇERİĞE DAYALI FİLTRELEME

İçeriğe dayalı filtreleme yönteminde, tüm kullanıcı ve ürünler için profil oluşturulur. Ürünün profilinde anahtar kelimeler oyuncuları, yönetmen olabilir. Kullanıcıda ise, bazı kişisel bilgileri veya geri bildirimleri olabilir. Bu filtreleme yönteminde, kullanıcının daha öncesinden tükettiği ürünlerle karşılaştırılma yapılır. Bu sayede benzer olanlar kullanıcıya önerilir [24].

İçeriğe dayalı filtreleme, işbirlikçi filtrelemenin tersine yalnızca kullanıcıların kendisinin geçmişine ve kararlarına bakarak önerilerde bulunur. Yani kullanıcının başta açtığı profilden veya zaman içinde oluşan dinamik profillerden yararlanır [27]. Bu filtreleme yönteminin çalışması prensibi, kullanıcıların kararlarına göre öneride bulunulacak ürün arasında matrissel bir bağ kurularak öneride bulunmaya dayanır.

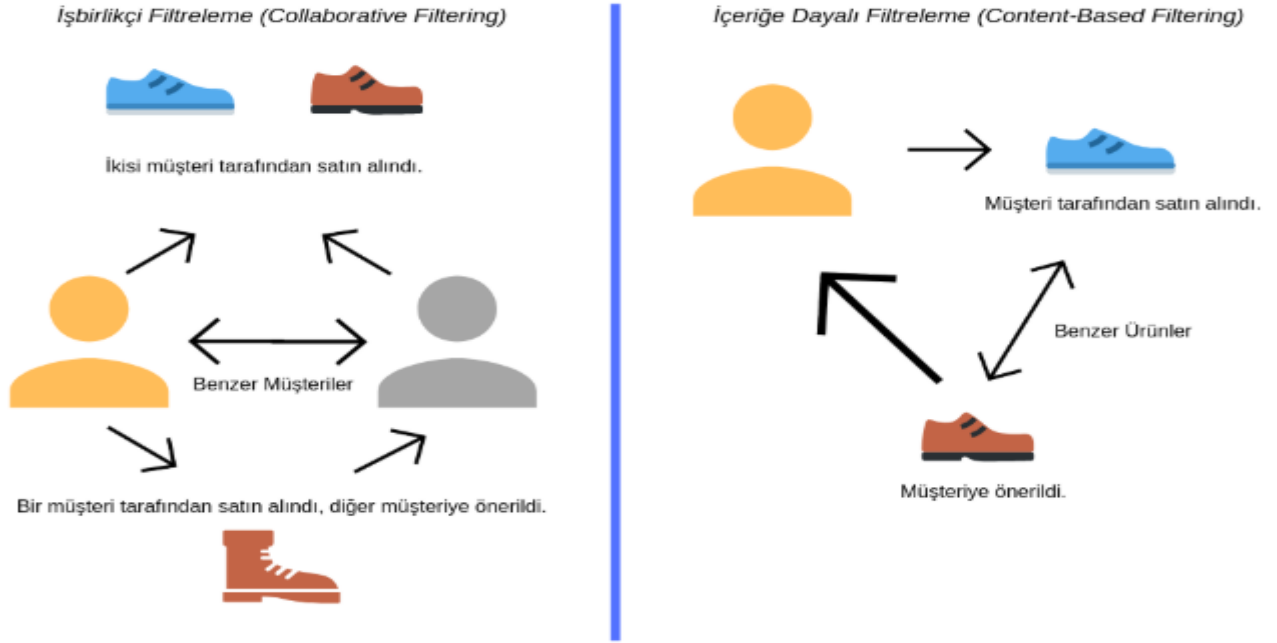
Bu filtreleme yönteminin avantajları:

- Yalnızca kullanıcının kararlarına göre öneri yaptığından işbirlikçi filtreleme kadar fazla hesap gerektirmez,
- Sistemde az kullanıcı varsa daha güvenilir önerilerde bulunur,
- Birbirlerine benzer kullanıcıları bulma sırasında yaşanacak hatalı sonuçları görmezden gelir.

Dezavantajları:

- Veri tabanına yeni girilmiş bir ürün hakkında yeterli miktarda bilgi yoksa hatalı önerme yapabilir,
- Yeni bir kullanıcı geldiğinde henüz bir ürün tüketmediği için hatalı öneriler olabilir.

İşbirlikçi ve İçeriğe dayalı filtreleme arasındaki bazı farklar Şekil 4.1’de gösterilmiştir.



Şekil 4.1 İşbirlikçi ve İçeriğe dayalı filtreleme arasındaki fark

4.3 HİBRİT FİLTRELEME

Hibrit filtreleme yöntemi kabaca, içeriğe dayalı filtreleme ve işbirlikçi filtreleme yöntemlerinin ağırlıklı puanlarının toplanması sonucu meydana gelen yöntemdir.

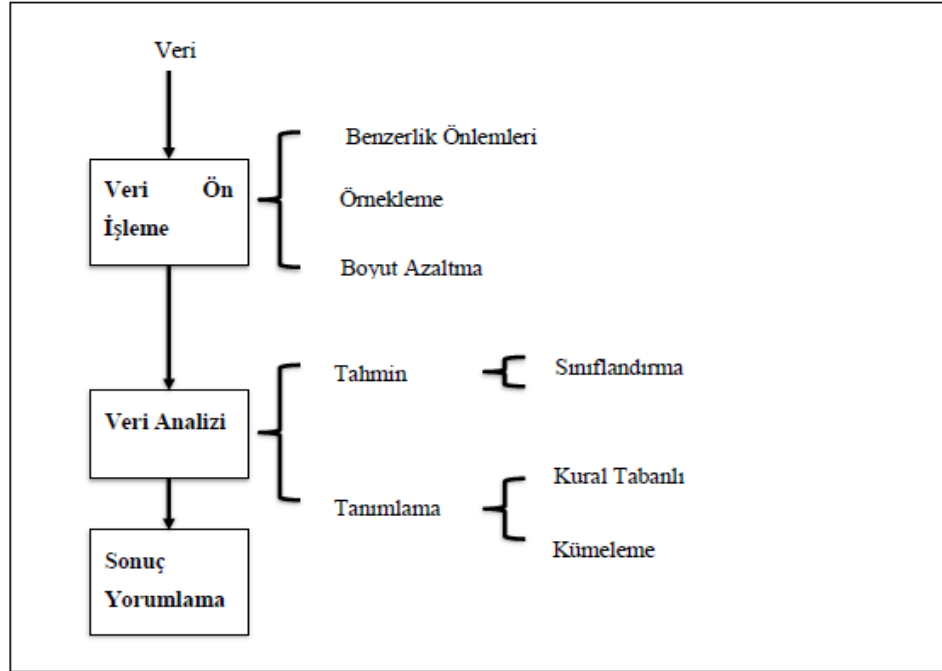
Hibrit öneri sistemi kullanıcıların seçtiği ürünler hariç, o ürünleri tüketen diğer kullanıcı davranışlarını da hesaplayarak öneri sunmaktadır. Bu sayede iki kullanıcının aynı anda tükettikleri ürün sayısı ne kadar fazlaysa bu kullanıcılar birbirlerine benzeme oranı diğer kullanıcılara göre daha fazla olduğu düşünülerek geliştirilmiş bir yöntemdir. Yalnızca kullanıcıların aynı anda tükettikleri ürünlere değil aynı zamanda kullanıcıların ürünlere geri dönüş puanlarına da bakılabilir [24].

Hibrit filtreleme tekniği, diğer iki öneri teknikleri ayrı ayrı gerçekleştirilip sonrasında birleştirilerek tek bir teknik haline getirilebilir. İşbirlikçi filtreleme yapan sisteme içeriğe dayalı filtreleme yapan teknik eklenerek ya da tam tersi içeriğe dayalı çalışan tekniğe işbirlikçi filtreleme yapan teknik eklenerek tek bir model haline getirilebilir. Performans olarak bakıldığında hibrit filtrelemenin, diğer iki teknikten daha doğru ve kesin sonuç verdiği ortaya konmuştur.

Bu öneri sistemini en çok kullanan uygulamalardan birinin Spotify olduğu söylenebilir. Spotify belli aralıklarla kullanıcılarına hazırladığı öneri çalma listeleri, kullanıcının dinlediği müziklere ve diğer kullanıcılarıyla benzer davranışları sergilediği için bu kullanıcılarına aynı türden çalma listesi önermektedir.

4.4. ÖNERİ SİSTEMLERİNDE VERİNİN ANALİZİ

Amatriani'in yaptığı çalışmaya göre [28], öneri sistemlerinin aslında veri madenciliğinin alt kümesi sayılabilecek algoritmalardan oluştuğunu belirtmiştir. Öneri sistemlerinde verinin analizi süreci veri ön işleme, veri analizi ve sonuç yorumlama olmak üç adımdan meydana gelir. Bu süreç Şekil 4.2'de gösterilmiştir [26].



Şekil 4.2 Verinin analizindeki süreçler

4.4.1. VERİ ÖN İŞLEME

Veri, bir objenin veya obje topluluğunun nitelik ve özelliklerinden meydana gelir. Verinin analiz edilebilmesi için öncelikle temizlenmesi gerekmektedir. Temizleme işlemlerine örnek olarak hatalı yazılmış verilerin düzeltilmesi, istenmeyen kelime veya sözcüklerin filtrelenmesi gibi

işlemler örnek gösterilebilir. Veri ön işleme için benzerlik önlemleri, örnekleme ve boyut azaltma olmak üzere üç farklı yöntem vardır [26].

Benzerlik Önlemleri

Burada verilerin birbirleri arasındaki mesafesi ve benzerliklerine göre ağırlıklandırma hesabı yapılır. Bu hesaplamayı yapmak için şu algoritmalar kullanılır:

- Öklid uzaklığı
- Minkowski uzaklığı
- Manhattan uzaklığı
- Kosinüs benzerliği
- Pearson korelasyonu

Bu tezde benzerlik hesabı için kosinüs benzerliği algoritması kullanılmıştır.

Örnekleme

Örnekleme, veri madenciliğinde çok veriden oluşan veri tabanındaki verilerin alt kümelerini oluşturmak için kullanılır. Bu teknik, veri analizi süreçlerinden hem veri ön işleme hem de yorumlama aşamalarını kapsar. Bu yöntemin kullanılmasındaki amaç, büyük bir veri varsa ve tüm verilerin işlenmesi gerekiyorsa bu işlem uzun zaman alacağından bu verilerin alt kümelerini bularak örnekleme işlemi yapılır. En bilindik örnekleme yönteminin, rastgele örnekleme olduğu söylenebilir [26].

Boyut Azaltma

Boyut azaltma tekniği, çok boyuta sahip alanı küçülterek daha az boyutlu bir alan haline getirmekte kullanılır. En çok kullanılan boyut azaltma tekniklerinin tekil değer ayrışımı ve temel bileşen analizi olduğu söylenebilir. Temel bileşen analizinde amaç, çok boyutlu veri setlerindeki kalıpları belirlemek için kullanılan istatistiksel tekniklerdir. Tekil değer ayrışımı ise, yeni niteliklerin kavramlar gibi temsil edilerek her bir kavramın değerinin hesaplanabilir olduğu daha az boyutlu nitelik alanları keşfetmekte kullanılır [26].

4.4.2. VERİ ANALİZİ

Verinin analiz edilmesinde en sık kullanılan yöntem sınıflandırma yöntemidir. Sınıflandırma yöntemi, önceden belirlenmiş niteliğin alanıyla niteliğin kendisini sınıflar. Veri kümesi içerisinde bu sınıfların belirlenmesi için etiketler kullanılır.

Bazı sınıflandırma yöntemleri:

- K-En yakın komşular
- Karar ağacı
- Bayes sınıflandırma
- Yapay Sinir Ağları
- Destek Vektör Makinesi (SVM)

Bir sınıflandırıcı modelinin doğruluk oranını hesaplamak için şu yöntem kullanılır:

- Gerçek Pozitif (TP): Tahmin edilen verinin olumlu ve gerçek olduğu durumdur.
- Gerçek Negatif (TN): Tahmin edilen verinin olumsuz ve doğru olduğu durumdur.
- Yanlış Pozitif (FP): Tahmin edilen verinin olumlu ve yanlış olduğu durumdur.
- Yanlış Negatif (FN): Tahmin edilen verinin olumsuz ve yanlış olduğu durumdur.

Buna bağlı olarak doğruluk oranı $(TP+TN) / (TP+TN+FP+FN)$ şeklinde hesaplanır.

4.4.3. SONUÇ YORUMLAMA

Veri analizinin son adımıdır. Bu adımda elde edilmiş sonuçlar değerlendirilerek, bu sonuçların tutarlılığına bakılır. Eğer tutarlı sonuçlar elde edildiyse analiz işlemi sonlanır. Tutarlı sonuç elde edilmediyse bir önceki adıma tekrar dönülür.

5. BENZER ÇALIŞMALAR

Şimdiye kadar yapılan film önerme çalışmalarından içeriğe dayalı filtreleme, işbirlikçi filtreleme ve hibrit filtreleme yaklaşımlarından pek çok çalışma yapılmıştır. Bu çalışmalardan birçoğu Netflix, MovieReco, Ringo gibi e-ticaret uygulamalarının temelini oluşturmaktadır [31].

Goldberg'in 1992 senesinde yaptığı çalışmanın literatürde yapılan ilk işbirlikçi filtreleme olduğu bilinir [32]. Goldberg bu çalışmasında "Tapestry" adı verilen bir önerme sistemi ortaya atmıştır. Tapestry kısaca manuel çalışma mantığıyla çalışan işbirlikçi sistem olarak nitelendirilebilir. Yani, yeni bir veri eklenmek istenildiğinde kullanıcı bunu kendi eklemesi gerekir. Bu sistem aynı zamanda içeriğe dayalı filtrelemenin başlangıç noktası olarak kabul edilir.

Tan ve Teo [33] 1998 yılında yaptığı çalışmada PIN adını verdikleri bir sistem öne atmışlardır. Bu sistem kullanıcıların profillerini, anahtar kelimeleri ve ilgilendikleri terimleri bir veri tabanında saklar. Bu sayede bu terimlerden yararlanarak kullanıcıya kendisine en uygun öneriyi sunmaya çalışır. Bu sistemin en büyük dezavantajı, kullanıcının başlangıçta seçtiği terimler zaman içinde değişiklik gösterirse sistem onu anlamayacağı için yanlış öneriler sunar.

Jiahui Liu ve arkadaşlarının [34] 2010 senesinde içeriğe dayalı filtreleme yöntemini kullanarak yaptığı çalışmaya göre, öneri sistemlerini kullanıcıların ilgi alanlarının bilinmesiyle meydana geldiği söylenmiştir. Kullanıcı profillerinin oluşabilmesi için kullanıcıların sistemi kullanması zorunluydu. Onlara göre kullanıcılar hakkında bilgi kazanabilmek için sistemi kullanmayı beklemeleri ciddi zaman kaybına yol açıyordu. Bu sebeple kullanıcıların tıklama hareketlerinin yanında sayfada kalma süreleri ve genel eğilimleri gibi karar verici özelliklerini de araştırarak daha doğru bir önerme sistemi ortaya atmışlardır.

Gong [35] işbirlikçi filtreleme yöntemi kullanarak yaptığı çalışmasında ürünleri ve kullanıcıları 2 farklı kümeye atayarak bu kümeler arasındaki ilişkilere göre öneri sunan bir sistem tasarlamıştır. Bu sistemin en büyük dezavantajı yeni gelen verilerin kümelenmesi ve bu işi yapan algoritmanın çok karmaşık olmasıdır.

Amini ve arkadaşlarının [36] 2014 senesinde yapılan çalışmasında, içeriğe dayalı filtreleme ve işbirlikçi filtrelemenin olumlu yanlarını alarak hibrit bir öneri sistemi geliştirmişlerdir. İçeriğe dayalı filtrelemeler, ürünlerin nitelikleri ve açıklamaları hakkında bilgiler verir. Bu sayede kullanıcılar ile ürünlerin nitelikleri arasında modelleme yapabilir. İşbirlikçi filtreleme tekniğinde ise, kullanıcının geri dönüşleri ve yorumları arasında bağlantı kurmaktır. Bu çalışmada ürünlerin niteliklerini modellemek için K-Ortalamalar algoritması kullanılmıştır [37].

Karypis ve Desrosiers'ın [38] 2011 senesinde yaptığı çalışmada, işbirlikçi filtreleme tekniğinde kullanılan verilerin birbirine olan komşuluğuna dayanan, ürün önerilerinde karşılaşılan problemlere değinmiştir. Bu problemleri ortadan kaldırmak için komşuluk tabanlı tekniklerin detaylı araştırmasını yapıp çözüm olarak sunmuşlardır. Bir diğer sorun ise, kullanıcılar hakkında bilginin kısıtlı olması ve genellikle büyük firmaların kullandığı öneri sistemlerinde görülen sınırlı kapsam problemine bu çalışmada değinilmiştir. Ayrıca boyut azaltma tekniklerinin öneri sistemlerinde yaşanan seyreklik ve kapsam problemlerine çözüm önerilerinde bulunulmuştur [37].

6. UYGULAMA

Günümüzde birçok platformda film izlemeyi seven kullanıcılar için sayısız sayıda seçenek bulunmaktadır. Kullanıcılar her gün kendilerine hitap eden filmleri bulup izlemek için çok fazla zaman harcamaktadır. Bu sorunu çözmek için veri madenciliği yöntemleri kullanılarak film önerme sistemleri geliştirilmiştir.

6.1. VERİNİN TOPLANMASI

Uygulamanın gerçekleşmesinde kullanılan veri seti oldukça popüler olan IMDb'nin veri tabanındaki filmlerden yararlanılmıştır. IMDb, 1990 yılında kurulan her kullanıcıya açık film, dizi, televizyon programları ve oyuncu bilgilerini bünyesinde bulunduran bir sitedir.

IMDb'nin film veri tabanına oluşabilmek için kendilerinin geliştirdiği, herkese açık hizmet veren IMDb API kullanılmıştır. Bu sayede 4802 tane farklı film csv formatında IMDb veri tabanından indirilmiştir. Filmlere ait bilgiler şu şekildedir:

- Bütçe
- Tür
- Çıkış tarihi
- Başlık
- İzlenme sayısı
- Web sitesi
- Orijinal dil
- Süre
- Puan
- Oyuncular
- Yönetmen

Bu çalışmada bu bilgilerden yalnızca filmin başlığı, türü, oyuncularını ve yönetmeni kullanılarak öneri yapılmıştır.

6.2. SİSTEMİN TASARLANMASI

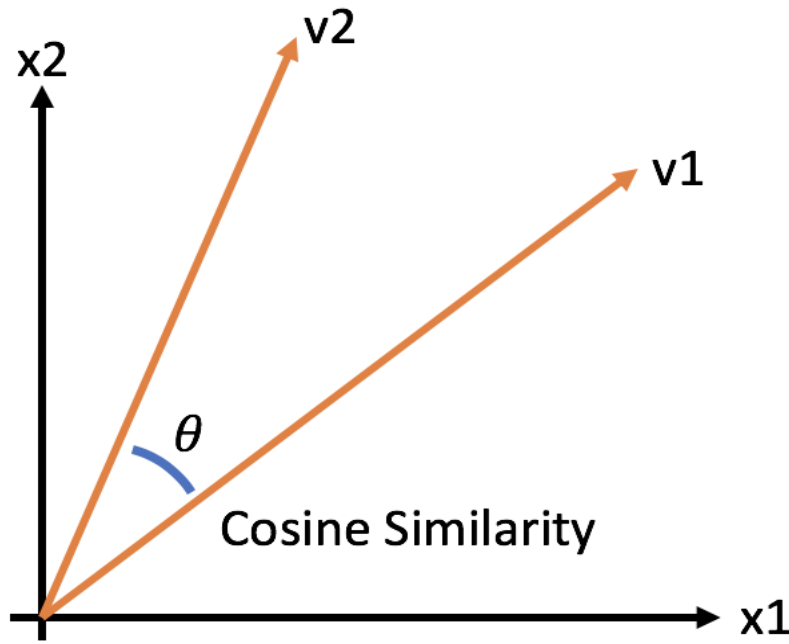
Bu çalışmada, filmlerin film adı, oyuncu, tür ve yönetmene göre benzerlikleri hesaplanıp birbirlerine en yakın filmlerin önerilmesi amaçlanmıştır. Bunun için iki adet ayrı yöntem kullanılmıştır. Bunlar kosinüs benzerliği ve Word2Vec yöntemleridir. Uygulamanın yazılımı için Python programlama dili kullanılmıştır. Bu kapsamda kullanılan bazı kütüphaneler Şekil 6.1’de gösterildiği gibidir:

```
import pandas as pd
import networkx as nx
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
import nltk
import gensim
from gensim import corpora, models, similarities
```

Şekil 6.1 Kullanılan kütüphaneler

6.2.1. KOSİNÜS BENZERLİĞİ İLE ÖNERME

Kosinüs benzerliği, kullanıcıların daha önceki değerlendirmelerinin vektörel değeri olarak tanımlanabilir. Böylece iki vektör arasındaki açı kosinüs değerini temsil eder. Kosinüs benzerliğinde kullanıcılar arasındaki benzerlik 0 ile 1 arasındadır. Sonuç 1'e ne kadar yakınsa vektörlerin o kadar birbirine benzer olduğu söylenir [31]. Şekil 6.2'de iki vektör arasında örnek bir kosinüs benzerliği gösterilmiştir.



Şekil 6.2 İki vektör arasındaki kosinüs benzerliği

Python'da kosinüs benzerliğinden yararlanmak için Şekil 6.1'de görüldüğü gibi "cosine_similarity" kütüphanesi kullanılmıştır.

Kosinüs benzerliği iki vektör arasında gerçekleşmektedir. Bu nedenle veri setindeki bilgileri vektörel bir gösterimle göstermek için Şekil 6.1'de görüldüğü gibi "CountVectorizer" kütüphanesi kullanılmıştır. Böylece kullanıcının izlediği filmin bilgileri ile veri setinde yer alan diğer filmlerin bilgileri iki ayrı vektör olarak kullanılabilir. Bu kütüphanelerden yararlanarak,

kullanıcının izlediği filme göre veri setinde ona en yakın film adı, tür, oyuncu ve yönetmenin önerildiği uygulama geliştirilmiş ve 0 ile 1 arasındaki benzerlik oranları tablo halinde bastırılmıştır. Bu bölümde filmler tek tek filtrelenmiştir. Ekler bölümünde ikili, üçlü ve tüm parametrelere göre filtrelenmiş örnekler mevcuttur.

Örnek olarak veri setinde yer alan “Fight Club” filmini film adı, tür, oyuncu ve yönetmenine göre öneri yapmak istersek;

Film adına göre öneri Şekil 6.3’te gösterilmiştir.

```
Top 5 similar movies to Fight Club are:  
  
Fight Valley  
Club Dread  
The Cotton Club  
The Emperor's Club  
Dallas Buyers Club
```

Şekil 6.3 Filmin adına göre önerilen filmler

Bu filmlerin kaçınıcı satırda oldukları ve benzerlik oranı Tablo 6.1’de gösterildiği gibidir.

most_similar_movies - List (4802 elements)

Ind.	Type	Size	Value
0	tuple	2	(2288, 0.4999999999999999)
1	tuple	2	(3235, 0.4999999999999999)
2	tuple	2	(1018, 0.408248290463863)
3	tuple	2	(2789, 0.408248290463863)
4	tuple	2	(3571, 0.408248290463863)

Save and Close Close

Tablo 6.1 Filmin adına göre benzerlik değerleri

Fight Club filminin türü drama olarak belirtilmiştir. Film türüne göre öneri Şekil 6.4’te gösterilmiştir.


```
Top 5 similar movies to Fight Club are:  
The Perfect Storm  
The Aviator  
Ali  
Ben-Hur  
Cold Mountain
```

Şekil 6.4 Filmin türüne göre önerilen filmler

Bu filmlerin kaçınıcı satırda oldukları ve benzerlik oranı Tablo 6.2’de gösterildiği gibidir.

Buradaki tüm filmler drama türünde oldukları için benzerlik oranı 1 yani en yüksek çıkmıştır.

most_similar_movies - List (4802 elements)

Indi	Type	Size	Value
0	tuple	2	(214, 1.0)
1	tuple	2	(250, 1.0)
2	tuple	2	(264, 1.0)
3	tuple	2	(357, 1.0)
4	tuple	2	(448, 1.0)

Save and Close Close

Tablo 6.2 Filmin türüne göre benzerlik değerleri

Bu veri setinde her film için 5 oyuncunun ismi yer almaktadır. Fight Club filminde oynayan oyuncular; Edward Norton, Brad Pitt, Meat Loaf, Jared Leto, Helena Bonham Carter şeklindedir. Buna göre sıradaki öneride bu oyunculara dair daha önce oynadıkları filmlere göre öneri yapılması beklenir. Oyuncuya göre öneri Şekil 6.5’te gösterilmiştir.

```
Top 5 similar movies to Fight Club are:  
Moonrise Kingdom  
Alice in Wonderland  
Alice Through the Looking Glass  
Big Fish  
Corpse Bride
```

Şekil 6.5 Filmin oyuncularına göre önerilen filmler

Bu filmlerin kaçıncı satırda oldukları ve benzerlik oranı Tablo 6.3’de gösterildiği gibidir.

most_similar_movies - List (4802 elements)

Indi	Type	Size	Value
0	tuple	2	(2461, 0.2860387767736777)
1	tuple	2	(32, 0.2727272727272727)
2	tuple	2	(105, 0.2727272727272727)
3	tuple	2	(583, 0.2727272727272727)
4	tuple	2	(1594, 0.2727272727272727)

Save and Close Close

Tablo 6.3 Filmin oyuncularına göre benzerlik değerleri

Fight Club filminin yönetmeni David Fincher’dır. Buna göre önerilecek film listesinin David Fincher’ın yönettiği filmler olması beklenir. Yönetmene göre öneri Şekil 6.6’da gösterilmiştir.

```
Top 5 similar movies to Fight Club are:  
The Girl with the Dragon Tattoo  
Zodiac  
Gone Girl  
Alien³  
The Game
```

Şekil 6.6 Filmin yönetmenine göre önerilen filmler

Bu filmlerin kaçıncı satırda oldukları ve benzerlik oranı Tablo 6.4’de gösterildiği gibidir. Buradaki tüm filmler David Fincher tarafından yönetildiği için benzerlik oranı 0.99 olarak hesaplanmıştır.

most_similar_movies - List (4802 elements)

Indi	Type	Size	Value
0	tuple	2	(354, 0.9999999999999998)
1	tuple	2	(421, 0.9999999999999998)
2	tuple	2	(662, 0.9999999999999998)
3	tuple	2	(693, 0.9999999999999998)
4	tuple	2	(838, 0.9999999999999998)

Save and Close Close

Tablo 6.4 Filmin yönetmenine göre benzerlik değerleri

6.2.2. WORD2VEC İLE ÖNERME

Word2Vec, Google çalışanı Tomos Mikolov [41] ve arkadaşlarının 2013 yılında geliştirdiği bir tekniktir. Bu teknikte metin verisi üzerinde işlem yapabilmek için tıpkı ilk önermede olduğu gibi sözcükleri sayısallaştırıp vektör haline getirmek gerekir. Word2Vec'te bu işlemler kelime gömme (Word embedding) sayesinde gerçekleşir. Kelime gömme, sözcükleri sayısallaştırıp vektör haline getiren bir dil modelleme tekniğidir. Word2Vec tekniği, sadece kelimelerin benzerlik oranını bulmak için değil aynı zamanda veriyi temizlemek, noktalama işaretlerini çıkartmak, text verisini küçük harfe çevirmek gibi kısacası veriye ön işlemde yapılan işleri de gerçekleştirebilir [42].

Python'da Word2Vec'ten yararlanmak için Şekil 6.1'de görüldüğü gibi "gensim" kütüphanesi kullanılmıştır.

Bu öneri tekniğinden yararlanmak için mevcut veri setinde yer alan veriler komşuluk listesi (Adjacency list) formatından kenar listesi (Edge list) formatına çevrilmiştir. Veri setinin komşuluk liste biçimi Şekil 6.7, kenar liste biçimi ise 6.8'de gösterilmiştir.

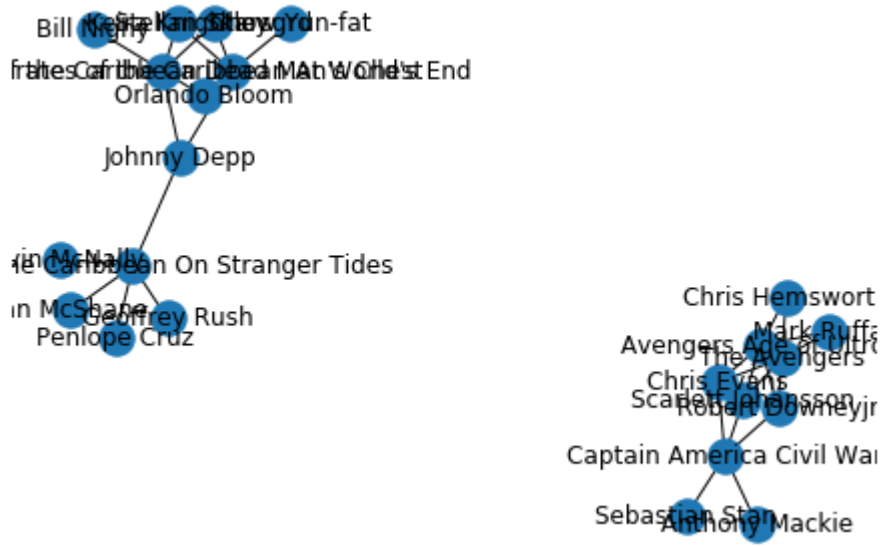
	A	B
1	title	cast
2	Avatar	Sam Worthington Zoe Saldana Sigourney Weaver Stephen Lang Michelle Rodriguez
3	Pirates of the Caribbean: At World's End	Johnny Depp Orlando Bloom Keira Knightley Stellan Skarsgrd Chow Yun-fat
4	Spectre	Daniel Craig Christoph Waltz Léa Seydoux Ralph Fiennes Monica Bellucci

Şekil 6.7 Veri setinin komşuluk listesi formatlı hali

	A	B	C	D	E	F
1	title,cast					
2	Avatar,Sam Worthington					
3	Avatar,Zoe Saldana					
4	Avatar,Sigourney Weaver					
5	Avatar,Stephen Lang					
6	Avatar,Michelle Rodriguez					
7	Pirates of the Caribbean At World's End,Johnny Depp					
8	Pirates of the Caribbean At World's End,Orlando Bloom					
9	Pirates of the Caribbean At World's End,Keira Knightley					
10	Pirates of the Caribbean At World's End,Stellan Skarsgrd					
11	Pirates of the Caribbean At World's End,Chow Yun-fat					
12	Spectre,Daniel Craig					
13	Spectre,Christoph Waltz					
14	Spectre,Léa Seydoux					
15	Spectre,Ralph Fiennes					
16	Spectre,Monica Bellucci					
17	The Dark Knight Rises,Christian Bale					
18	The Dark Knight Rises,Michael Caine					
19	The Dark Knight Rises,Gary Oldman					

Şekil 6.8 Veri setinin kenar listesi formatlı hali

Veri setini kenar liste formatına çevirdikten sonra hangi oyuncuların hangi filmlerde oynadığını tek bir yapıda gözlemlemek için graph yapısı oluşturulabilir. Graph yapısı oluşturmak için gerekli olan “networkx” kütüphanesi Şekil 6.1’de görüldüğü gibi tanımlanmıştır. Veri sayısı çok fazla olduğundan oluşan graph anlamsız bir görsel oluşturmuştur. Bunun yerine örnek olarak az sayıda veri kullanarak oluşturulan graph yapısı Şekil 6.9’da gösterilmiştir.



Şekil 6.9 Graph yapısı

Örnek olarak yine ilk önermede kullanılan “Fight Club” filmini ele alarak kendisine en yakın filmler listelendiğinde ilk önerideki filmlerden tamamen farklı filmlerin önerildiği Şekil 6.10’da gösterilmiştir.

```
In [45]: model.most_similar(positive=['FightClub'], topn=5)
C:\Users\lenovo\.spyder-py3\untitled1.py:1: DeprecationWarn
`most_similar` (Method will be removed in 4.0.0, use self.w
import pandas as pd
Out[45]:
[('TheBeyond', 0.6080554723739624),
 ('EscapefromNewYork', 0.5891255140304565),
 ('BetterLuckTomorrow', 0.5649175643920898),
 ('TheFamilyStone', 0.5548728704452515),
 ('ClashoftheTitans', 0.54128098487854)]
```

Şekil 6.10 Word2Vec yöntemine göre film önerisi

Kosinüs Benzerliği	Word2Vec
Fight Valley 0.49	The Beyond 0.60
Club Dread 0.49	Escape from New York 0.58
The Cotton Club 0.40	Better Luck Tomorrow 0.55
The Emporor's Club 0.40	The Family Stone 0.55
Dallas Buyers Club 0.40	Clash of the Titans 0.54

Tablo 6.5 İki yöntemin karşılaştırılması

Tablo 6.5’de görüldüğü gibi ilk önerilerden tamamen farklı filmlerin önerilmesinin nedeni Mikolov ve arkadaşlarının [42] yaptığı çalışmaya göre Word2Vec yönteminde kelimelerin sözdizimsel ve anlamsal benzerliklerini tespit ederek öneride bulunmayı amaçlamasıdır [43]. Anlamsal benzerliği daha iyi anlamak için Şekil 6.11 ve Şekil 6.12’de bir örnek gösterilmiştir.

```
In [46]: model.most_similar(positive=['tiger'], topn = 5)
C:\Users\lenovo\.spyder-py3\untitled1.py:1: DeprecationWarning
(Method will be removed in 4.0.0, use self.wv.most_similar
import os
Out[46]:
[('elephant', 0.8905155658721924),
 ('wild', 0.886576771736145),
 ('cow', 0.8775873184204102),
 ('rhino', 0.8762283325195312),
 ('shark', 0.8751325607299805)]
```

Şekil 6.11 Anlamsal benzerliğe örnek

```
In [9]: model.most_similar(positive=['Computer'], topn = 5)
C:\Users\lenovo\.spyder-py3\untitled1.py:1: DeprecationWarning
(Method will be removed in 4.0.0, use self.wv.most_similar())
import os
Out[9]:
[('Device', 0.9129914045333862),
 ('Robot', 0.9121586084365845),
 ('Complete', 0.910991370677948),
 ('Software', 0.9080201387405396),
 ('Leak', 0.9067715406417847)]
```

Şekil 6.12 Anlamsal benzerliğe örnek

Şekil 6.11’e bakarsak “tiger” kelimesine veri tabanında bulunan anlamca kendisine en yakın kelimeler önerilmiştir. Aynı sözleri Şekil 6.12’deki “computer” örneği için de söylemek mümkündür.

7. SONUÇLAR

Öneri sistemleri kullanıcıya büyük veri içerisinde yalnızca kendisine uygun olan veriyi önermeyi amaçlar. Öneri sistemleri içeriğe dayalı filtreleme, işbirlikçi filtreleme ve iki yöntemin olumlu taraflarından oluşan hibrit filtreleme yöntemlerinden oluşur. Bu tez çalışmasında içeriğe dayalı filtreleme yöntemi kullanılmıştır. Bölüm 6’de bahsedildiği gibi bu çalışmada 4802 adet film den oluşan veri seti IMDb sitesinden alınmış ve kullanıcının belirttiği filmin adına, türüne, oyuncularına ve yönetmenine göre öneride bulunulmuştur. Öneride bulunurken kosinüs benzerliği ve Word2Vec yöntemi olmak üzere iki farklı yöntem kullanılmış ve sonuçlar karşılaştırılmıştır.

Her iki yöntemde de kullanıcının belirttiği film ile veri setindeki filmler iki ayrı vektör olarak tanımlanarak aralarındaki mesafe ölçülmüştür. Bu mesafede 1’e en yakın olan değerler birbirlerine en benzer, 0’a yakın olanlar en farklı olarak tanımlanmıştır. Kosinüs benzerliğinde, kelimenin anlamına bakmaksızın sadece belirtilen filmin bilgileri, veri setinde başka bir filmle eşleşirse öneride bulunur. Word2Vec yönteminde ise kelimenin anlamına göre veri seti içerisinde o kelimeye en benzerleri bularak öneride bulunmayı amaçlar. İki yöntemin de kullanıcının amaçlarına göre performansı değişiklik gösterebilir. Fakat günümüz film önerme sistemlerine göre kosinüs benzerliği yöntemi daha kullanışlı bir yöntemdir. Ayrıca performansı daha da yükseltmek için veri setindeki film sayısını arttırarak benzerlik oranı arttırılabilir. Bu sayede kullanıcıya daha doğru öneriler sunulmuş olur.

KAYNAKLAR

- 1.Savaş, Serkan, Nurettin Topaloğlu, and Mithat Yılmaz. "Veri madenciliği ve Türkiye'deki uygulama örnekleri." (2012).
- 2.Şimşek, M. U. "Sosyal ağlarda veri madenciliği üzerine bir uygulama." Gazi Üniversitesi. Ulusal Tez Merkezi 321573 (2012): 5-21.
3. Arslan, H., "Web sitesi erişim kayıtlarının veri madenciliği ile analizi", Yüksek Lisans Tezi, **Sakarya Üniversitesi**, Sakarya, (2008).
4. Methodologies for Knowledge Discovery and Data Mining : Third Pacific-Asia Conference, Pakdd-99, Beijing, China, April 26-28, 1999 : Proceedings, Zhong, N. - Zhou, L., Springer Verlag, 1999.
- 5.Selim Tüzüntürk "Veri Madenciliği ve İstatistik" Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, Cilt XXIX, Sayı 1, 2010, s. 65-90
6. Özcan, Canan. Veri madenciliğinin güvenlik uygulama alanları ve veri madenciliği ile sahtekarlık analizi. Diss. İstanbul Bilgi Üniversitesi, 2014.
7. SPSS, "AnwerTree Algorithm Summary", SPSS White Paper, USA, 1999.
8. SUN, Jie ve Hui LI, "Data Mining Method for Listed Companies, Financial Distress Prediction", Knowledge-Based Systems, 21, No. 1, 2008.
9. Mustafa Aykut GÖRAL "Kredi Kartı Başvuru Aşamasında Sahtecilik Tespiti İçin Bir Veri Madenciliği Modeli – Yük. Lisans Tezi İ.T.Ü. Fen Bilimleri Ens.. Ocak 2007-
10. Roiger, R.J. and Geatz, M.W., 2003. Data Mining: A Tutorial-Based Primer,Pearson Education Inc.,USA
11. Doç. Dr. Kaan Yaralıoğlu (Dokuz Eylül Ün. İktisadi ve İdari Bilimler Fakültesi) "Uygulamada Karar Destek Yöntemleri" Veri Madenciliği İlkem Ofset, İzmir, 2004
- 12.KAYGIN, Ceyda YERDELEN, Alper TAZEGÜL, and Hakan YAZARKAN. "İşletmelerin Finansal Başarılı ve Başarısız Olma Durumlarının Veri Madenciliği ve Lojistik Regresyon Analizi İle Tahmin Edilebilirliği." Ege Academic Review 16.1 (2016).
13. <http://bm.bilecik.edu.tr/Dosya/Arsiv/duyuru/bayesogrenmesi.pdf>
- 14.<http://web.itu.edu.tr/~sgunduz/courses/verimaden/slides/d3.pdf>
- 15.<http://ceng.gazi.edu.tr/~ozdemir/teaching/dm/slides/06.DM.C2.pdf>
16. Ayık, Y. Ziya, Abdülkadir Özdemir, and Uğur Yavuz. "LİSE TÜRÜ VE LİSE MEZUNİYET BAŞARISININ, KAZANILAN FAKÜLTE İLE İLİŞKİSİNİN VERİ MADENCİLİĞİ TEKNİĞİ İLE ANALİZİ." Atatürk Üniversitesi Sosyal Bilimler Enstitüsü Dergisi 10.2 (2007): 441-454.
- 17.Tolon, Metehan, and Nuray Güneri TOSUNOĞLU. "Tüketici tatmini verilerinin analizi: yapay sinir ağları ve regresyon analizi karşılaştırması." Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi 10.2 (2008).

18. Irmak, Sezgin, Can Deniz Köksal, and Özcan Asilkan. "Hastanelerin Gelecekteki Hasta Yoğunluklarının Veri Madenciliği Yöntemleri İle Tahmin Edilmesi." *Journal of Alanya Faculty of Business/Alanya İisletme Fakültesi Dergisi* 4.1 (2012).
19. Akgöbek, Ö. , Çakır, F., "Veri madenciliğinde bir uzman sistem tasarımı", Akademik Bilişim'09 - XI. Akademik Bilişim Konferansı Bildirileri Harran Üniversitesi, Şanlıurfa, (2009).
20. Arabacı, Gültekin. Veri Madenciliğinde Apriori, Tahminci Apriori ve Tertius Algoritmalarının WEKA ve YALE Programları ile Karşılaştırılması. Diss. Yüksek lisans tezi, İstanbul Ticaret üniversitesi, Sosyal Bilimleri Enstitüsü, İstanbul, 2007.
21. Baykal, Abdullah. "Veri madenciliği uygulama alanları." *Dicle Üniversitesi Ziya Gökalp Eğitim Fakültesi Dergisi* 7 (2006): 95-107.
22. Resnick, P., Varian, H. R., Recommender systems, *Communications of the ACM*, 40(3), 56-58, 1997.
23. Taşcı, Servet. İçerik bazlı medya takip ve haber tavsiye sistemi. MS thesis. Fen Bilimleri Enstitüsü, 2015.
24. Uluyağmur, Mahiye. Hibrit Film Öneri Sistemi. Diss. Bilişim Enstitüsü, 2012.
25. Su, X., Khoshgoftaar, T. M., A survey of collaborative filtering techniques, *Advances in artificial intelligence*, 2009.
26. Akgün, Muhammet. KARIYER PLANLAMA İÇİN KARAR DESTEK SİSTEMİ. MS thesis. Fen Bilimleri Enstitüsü, 2019.
27. Asanov, D., Algorithms and methods in recommender systems, Berlin Institute of Technology, Berlin, Germany, 2011.
28. X. Amatriain, A. Jaimes, N. Oliver, J. M. Pujol, *Recommender Systems Handbook*, (2015).
29. Çalışkan, Sibel Kırmızıgül, and İbrahim Soğukpınar. "KxKNN: K-Means ve K En Yakın Komşu Yöntemleri İle Ağlarda Nüfuz Tespiti." *EMO Yayınları* (2008): 120-24.
30. Tolun, Seda. Destek vektör makineleri: Banka başarısızlığının tahmini üzerine bir uygulama. İktisadî Araştırmalar Vakfı, 2008.
31. Bulut, Hasan, and Musa Milli. "İşbirlikçi filtreleme için yeni tahminleme yöntemleri." *Pamukkale University Journal of Engineering Sciences* 22.2 (2016).
32. Goldberg D, Nichols D, Oki BM, Douglas T. "Using collaborative filtering to weave an information tapestry". *Communications of the ACM*, 35(12), 61-70, 1992.
33. Tan, A. H., Teo, C., Learning user profiles for personalized information dissemination, *Neural Networks Proceedings, 1998, IEEE World Congress on Computational Intelligence, The 1998 IEEE International Joint Conference on* (Vol. 1, pp. 183-188), IEEE, 1998.

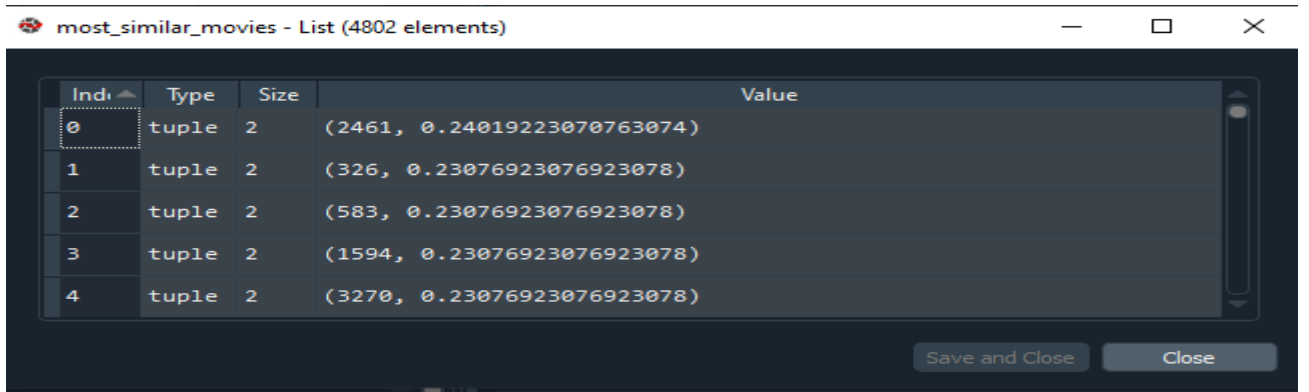
34. Liu, J., Dolan, P., Pedersen, E. R., Personalized news recommendation based on click behavior, Proceedings of the 15th international conference on Intelligent user interfaces (pp. 31-40), ACM, 2010.
35. Gong, S., A collaborative filtering recommendation algorithm based on user clustering and item clustering. Journal of Software, 5(7), 745-752, 2010.
36. Amini, M., Nasiri, M., and Afzali, M. 2014. Proposing a New Hybrid Approach in Movie Recommender System. International Journal of Computer Science and Information Security, 12(8), 4-45.
37. Anıl, U. T. K. U., and Muhammet Ali AKCAYOL. "Öğrenebilen ve adaptif tavsiye sistemleri için karşılaştırmalı ve kapsamlı bir inceleme." Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi 33.3 (2017): 13-34.
38. Desrosiers, C. and Karypis, G. 2011. A Comprehensive Survey of Neighborhood-based Recommendation Methods. Recommender systems handbook. İngiltere Springer. 107-144.
39. Vozalis E, Margaritis KG. "Analysis of recommender systems' algorithms". 6th Hellenic-European Conference on Computer Mathematics and its Applications, Athens, Greece, 25-27 September 2003.
40. Breese JS, Heckerman D, Kadie C. "Empirical analysis of predictive algorithms for collaborative filtering". 14th Conference on Uncertainty in Artificial Intelligence, San Francisco, CA, USA, 24-26 July 1998.
41. Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States, pages 3111–3119, 2013.
42. Sübay, Mehmet Turgut. Türkçe kelime vektörlerinde görülen anlamsal ve biçimsel yakınlıklar. MS thesis. Maltepe Üniversitesi, Fen Bilimleri Enstitüsü, 2019.
43. BİLGİN, Metin. "Kelime Vektörü Yöntemlerinin Model Oluşturma Sürelerinin Karşılaştırılması." Bilişim Teknolojileri Dergisi 12.2 (2019): 141-146.
44. Cleger-Tamayo, S., Fernández-Luna, J. M., Huete, J. F, Top-N news recommendations in digital newspapers, Knowledge-Based Systems, 27, 180-189, **2012**.
45. Li, Q., Kim, B. M., Clustering approach for hybrid recommender system, Web Intelligence, 2003. WI 2003, Proceedings IEEE/WIC International Conference on. IEEE, **2003**.

EKLER

EK-1 İkili filtrelemeye göre öneri

“Fight Club” filminin adına ve oyuncularına göre önerilen filmler ve benzerlik oranları

```
Top 5 similar movies to Fight Club are:  
  
Moonrise Kingdom  
Cinderella  
Big Fish  
Corpse Bride  
Howards End
```



The screenshot shows a Jupyter Notebook window with the title "most_similar_movies - List (4802 elements)". The window displays a table with the following data:

Indi	Type	Size	Value
0	tuple	2	(2461, 0.24019223070763074)
1	tuple	2	(326, 0.23076923076923078)
2	tuple	2	(583, 0.23076923076923078)
3	tuple	2	(1594, 0.23076923076923078)
4	tuple	2	(3270, 0.23076923076923078)

At the bottom of the window, there are two buttons: "Save and Close" and "Close".

EK-2 Üçlü filtrelemeye göre öneri

“Fight Club” filminin yönetmenine, oyuncularına ve türüne göre önerilen filmler ve benzerlik oranları

```
Top 5 similar movies to Fight Club are:  
  
Panic Room  
The Curious Case of Benjamin Button  
Down in the Valley  
Crazy in Alabama  
Se7en
```

most_similar_movies - List (4802 elements)

Ind	Type	Size	Value
0	tuple	2	(1010, 0.3450327796711771)
1	tuple	2	(100, 0.3241018617760822)
2	tuple	2	(3283, 0.3241018617760822)
3	tuple	2	(2634, 0.28571428571428575)
4	tuple	2	(1553, 0.2760262237369417)

Save and Close Close

EK-3 Tüm parametrelere göre öneri

“Fight Club” filminin adına, türüne, oyuncularına ve yönetmenine göre önerilen filmler ve benzerlik oranları

```
Top 5 similar movies to Fight Club are:
Panic Room
Down in the Valley
The Curious Case of Benjamin Button
Se7en
Fury
```

most_similar_movies - List (4802 elements)

Ind	Type	Size	Value
0	tuple	2	(1010, 0.30316953129541624)
1	tuple	2	(3283, 0.2727723627949905)
2	tuple	2	(100, 0.26064301757134345)
3	tuple	2	(1553, 0.25)
4	tuple	2	(456, 0.24253562503633297)

Save and Close Close

ÖZGEÇMİŞ

Doğum tarihi	31.08.1996	
Doğum yeri	İstanbul	
Lise	2011-2015	Çan Anadolu Lisesi
Lisans	2015-2020	İstanbul Ticaret Üniversitesi, Bilgisayar Mühendisliği Bölümü