İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

HİBRİT FİLM ÖNERİ SİSTEMİ

YÜKSEK LİSANS TEZİ

Mahiye ULUYAĞMUR

Bilgisayar Bilimleri Anabilim Dalı

Bilgisayar Bilimleri

Tez Danışmanı: Doç. Dr. Zehra ÇATALTEPE

TEMMUZ 2012

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

HİBRİT FİLM ÖNERİ SİSTEMİ

YÜKSEK LİSANS TEZİ

Mahiye ULUYAĞMUR
(704091021)

Bilgisayar Bilimleri Anabilim Dalı

Bilgisayar Bilimleri

Tez Danışmanı: Doç. Dr. Zehra ÇATALTEPE

TEMMUZ 2012

İTÜ, Bilişim Enstitüsü'nün 704091021 numaralı Yüksek Lisans Öğrencisi **Mahiye ULUYAĞMUR,** ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı **"HİBRİT FİLM ÖNERİ SİSTEMİ"** başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı :	**Doç.Dr. Zehra ÇATALTEPE** İstanbul Teknik Üniversitesi	.....
Jüri Üyeleri :	**Doç. Dr. Ali Taylan Cemgil** Boğaziçi Üniversitesi	.....
	Doç. Dr. A.Şima Etaner Uyar İstanbul Teknik Üniversitesi	.....

Teslim Tarihi : **11.05.2012**
Savunma Tarihi : **20.07.2012**

Anneme ve babama,

ÖNSÖZ

Yüksek lisans eğitimim boyunca bana her zaman destek olan, yol gösteren danışmanım Doç.Dr.Zehra Çataltepe'ye teşekkürü bir borç bilirim. Bu projede yer almamı sağladığı için de ayrıca teşekkür ederim.

Bana her zaman güvenen, maddi manevi desteklerini bir an olsun esirgemeyen ve aldığım kararların arkasında benimle birlikte duran sevgili aileme minnet duyuyorum.

Bu tez çalışması San-Tez destek programı kapsamında Bilim, Sanayi ve Teknoloji Bakanlığı ile Krea İçerik Hizmetleri ve Prodüksiyon A.Ş. tarafından 00966.STZ.2011-2 no'lu proje olarak desteklenmektedir.

Temmuz 2012

Mahiye Uluyağmur
(Bilgisayar Mühendisi)

İÇİNDEKİLER

Sayfa

KISALTMALAR

MAE **:** Mean Absolute Error (Ortalama Mutlak Hata)
RWNP **:** Rating Weighted Normalized Precision (Puanlarla Ağırlıklandırılmış Normalleştirilmiş Kesinlik)

ÇİZELGE LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

HİBRİT FİLM ÖNERİ SİSTEMİ

ÖZET

Sinema/televizyon ve müzik alanlarında, izlenebilecek ürün sayısı, türü ve bunları izleyebilecek izleyici sayısında büyük bir artış görülmektedir. Bu nedenle, herhangi bir ürünü, bu ürünü izlemekle en çok ilgilenebilecek izleyici kitlesine önermeye yarayacak öneri sistemleri de önem kazanmıştır. İçerik tabanlı öneri sistemleri kullanıcının şimdiye kadar izlediği ürünlerin içerik bilgisini kullanır ve içeriğin türünden etkilenir. Öte yandan, beraber filtreleme öneri sistemleri kulanıcıların ürünlere verdiği puanları (rating) kullanır ve içerik türünden bağımsızdır. İçerik tabanlı öneri sistemlerinin de, beraber filtreleme tabanlı öneri sistemlerinin de zayıf ve güçlü yönleri vardır. Hibrit öneri sistemleri, hem içerik hem de puanlama bilgisini kullanarak daha iyi öneriler üretmeyi amaçlar.

Bu çalışmada ürün olarak filmler kullanılmıştır. İçerik tabanlı bir öneri sistemi geliştirilmesi için film içeriği olarak oyuncu, yönetmen, tür gibi bilgilerin yanında her film hakkındaki özet dokümanlarından doküman işleme teknikleri ile üretilmiş vektörler ve kullanıcıların filmlere verdikleri puanlar kullanılmıştır. Ayrıca sadece kullanıcıların filmlere verdikleri puanları kullanan beraber öneri sistemi üzerinde çalışılmıştır. Bu iki sistemin doğrusal bir model ile birleştirilmesiyle kullanıcılara özel film önerileri yapabilmek için hem film içeriği hem de kullanıcıların puanlamalarını kullanan bir hibrit öneri sistemi geliştirilmiştir.

İçerik tabanlı öneri sisteminde kullanıcıların izlediği filmlerde geçen öznitelikleri, bu filmlere verdikleri puanlar ile ağırlıklandırarak her özniteliğin her kullanıcı için bir ağırlık değeri oluşturulmuştur. Böylelikle kullanıcıların hangi özniteliklere fazla hangilerine düşük ağırlık verdiği ortaya çıkmaktadır. Önerilecek filmin puanı kullanıcının verdiği ağırlıkların toplamının o kullanıcının eğitim kümesinde izlediği toplam film sayısına bölünmesiyle elde edilir. Öneri işlemi yüksek ağırlık değerine sahip öznitelikler içeren filmlerin kullanıcılara önerilmesi şeklinde yapılmaktadır. Sistemde filmlerin dört farklı özniteliğine göre puan üretilmektedir. Özniteliklerin çıkarılması işleminde öncelikle eğitim kümesindeki tüm kullanıcıların izledikleri tüm filmlerin öznitelikleri çıkarılmaktadır. Böylece film-öznitelik matrisi meydana getirilir. Bu işlem oyuncu, yönetmen, tür ve anahtar kelime öznitelik kümeleri için ayrı ayrı yapılmaktadır. Bir öznitelik kümesi içindeki bir özniteliğin ağırlığı kullanıcının o özniteliği bulunduran tüm filmlere verdiği puanların toplanması ve aynı kullanıcının eğitim kümesinde izlediği toplam film sayısına bölünmesiyle elde edilir. Öznitelikler için elde edilen bu ağırlıklar filmlere tahmini puan üretme işleminde kullanılır.

Test kümesinde *u* kullanıcısına önerilecek bir film geldiğinde öncelikle bu filmin özniteliklerine bakılır. Öznitelik türü olarak oyuncu seçildiğinde filmin hangi oyuncuları bulundurduğu ve bu oyuncuların, *u* kullanıcısının eğitim kümesinde ağırlık verdiği bir oyuncu olup olmadığı araştırılır. Kullanıcı *u'*nun bu oyuncuya ait

bir filmi önceden izleyip puan verdiği bulunursa önerilecek filmin puanı *u* kullanıcısının bahsi geçen oyuncuya ait ağırlık değeri olarak belirlenir. Önerilecek filmde geçen oyuncuların hangileri için eğitim kümesinde *u* kullanıcısının ağırlık değeri varsa bu değerlerin toplamı önerilecek film için verilecek puanı temsil eder. Oyuncu özniteliğine göre puan üretilirken filmlerin birden fazla oyuncusunun olmasıyla genelde ağırlıkların toplanması gerekir, ancak yönetmen özniteliğine göre puan üretilirkern genelde filmlerin tek yönetmeni olacağından ağırlıklar doğrudan puan olarak atanır. Yapılan deneylerde yönetmen özniteliğine göre öneri yapıldığında diğer öznitelik türlerine göre daha başarılı olunduğu görülmüştür. Tür özniteliğine göre alınan sonuçlarda iyi performans göstermektedir.

Beraber öneri sistemlerinde, genellikle, kullanıcıların sevdiği ve sevmediği ürünleri açık olarak derecelendirdiği dolaysız geri bildirimli öneri yöntemleri kullanılmaktadır. Öte yandan, TV program önerisi gibi çoğu alanda kullanıcıdan her program için derecelendirme istemek zordur. Derecelendirme yerine, kullanıcıların hangi ürünleri ne kadar süre ile izlediği bilgisinin toplanması ve dolaylı geri bildirimli öneri yöntemlerinin kullanılması daha uygundur. Bu çalışmada, kullanıcıların TV programı izleme süreleri normalleştirilerek üretilen beğeni değerleri puan gibi kullanılmıştır.

Bu çalışmada kullanıcıların filmlere verdikleri puanlarden oluşan puanlama matrisleri kullanılmıştır. Oldukça seyrek olan puanlama matrisi, matris ayrıştırma yöntemleri ile faktörlerine ayrılmıştır. Öneri yöntemi olarak dolaylı geri bildirimli öneri yöntemleri düzenli matris çarpanlarına ayırma yöntemi ile beraber kullanılmıştır. Dolaysız geri bildirimli yöntem ile de sonuçlar alınmış, ancak sistemimizde kullanıcılardan doğrudan puan alınamamasından dolayı dolaylı geri bildirimli yöntem esas alınmıştır. Matris çarpanlarının öğrenilmesi sırasında hem öğrenmenin hızlandırılması için uyarlamalı öğrenme hızı kullanılmış, hem de kullanıcı ve ürüne uyarlamalı düzenleme yöntemleri kullanılmıştır.

Beraber öneri sisteminde kullanıcılar arasındaki benzerliklerden yararlanan bir yöntem önerilmiştir. *u* kullanıcısına film önerisi yapılırken, önerilecek *i* filmini daha önceden eğitim kümesinde izleyen kullanıcılar araştırılır ve bu kullanıcıların *i* filmine verdikleri puanlar alınır. *i* filmini eğitim kümesinde izleyen kullanıcıların *u* kullanıcısı ile eğitim kümesinde ortak izledikleri filmlerin olup olmadığına bakılır. Eğer *u* kullanıcısı eğitim kümesinde bu filmi izlediyse, film *u* kullanıcısına direk önerilir. Hem *i* filmini izleyen hem de kullanıcı *u* ile aynı filmleri izleyen kullanıcıların u kullanıcısıyla izledikleri ortak film sayısının, bu kullanıcıların *i* filmine verdikleri puan ile çarpımlarının toplanmasıyla *u* kullanıcısına *i* filmi için puan üretilmiş olur. *u* kullanıcısının en fazla ortak film izlediği kullanıcıyla çok benzer oldukları yorumu yapılabilir.

Hibrit öneri sistemi, beraber ve içerik tabanlı önerilerin iki değişik şekilde birleştirilmesi ile oluşturulmuştur. Birinci sistemde beraber öneri puanı önerilecek filmi izleyen kullanıcıların, öneri verilecek kullanıcı ile ortak izlediği film sayısıyla orantılı olarak oluşturulmuştur. İkinci hibrit sistemde ise matris ayrıştırma sonuçlarından üretilen beraber öneri puanları kullanılmıştır. Birinci sistemle içerik tabanlı sistemin doğrusal olarak birleştirilmesi en iyi sonuçları vermiştir.

Tez çalışmasında ayrıca TV film önerilerinin değerlendirilmesinde kullanılabilecek, değişik performans ölçütleri kullanılmış ve yeni ölçütler önerilmiştir. Bütün yöntemlerin performansları 13 aylık bir veri kümesi üzerinden değerlendirilmiştir.

HYBRID MOVIE RECOMMENDATION SYSTEM

SUMMARY

The number and kind of available content and the number of users who can view them have increased tremendously in both movie/television and music domains. Therefore, recommendation systems that can accurately recommend to a certain user the set of products that he would most likely be interested and as fast as possible, have become important. While content based recommendation systems use features of products a user has viewed so far and they are domain dependent, domain independent collaborative filtering systems use only the ratings given to each product by a number of users. There are some shortcomings of both collaborative and content-based recommendation systems. Cold-start problem is one of the most important problem of the collaborative filtering systems. If a movie is not watched in the training set, this movie can not be recommended to any user. Content-based system can solve this problem. Moreover if a user is new in the system namely if s/he did not watch any movies, collaborative filtering system can not recommend any movies to this user either. In order to solve the new user problem user demographics can be used, however they tend to be not so reliable for many domains. In our system we first observe the watching behavior of a user for a number of movies and then do recommendations. Content-based recommendation systems rely on content features which need to be extracted. Rating matrices are generally sparse and high dimensional matrices, so it is costly to work with large matrices. In collaborative filtering system matrix factorization methods can generate low dimensional user and item factors to solve the sparsity problem. Content-based recommendation systems rely on content data gathered for a specific user and if too complex models are chosen they may suffer from overspecialization. Different hybrid recommendation systems that integrate content and collaborative recommendation systems have been proposed in the literature.

In this thesis, content-based, collaborative and hybrid TV movie recommendation methods are proposed and evaluated. In the content-based recommendation system as the content for a movie, we use information such as movie actor, producer, genre and also words obtained from the movie summaries. In addition to these fields, computed (implicit) ratings which users give to the movies are used in the content-based system. Another recommendation method used in this work is the collaborative filtering method. Collaborative filtering method uses only users' ratings for movies. In this project, we also propose a hybrid movie recommendation system which uses a linear combination of recommendations proposed by the content-based and collaborative filtering methods.

Recommendation systems need user ratings. However, for the TV recommendation problem, we do not have explicit ratings from the users. In this thesis, we used the implicit ratings of the movies, which are generated as the percentage of the movie watched by the user over all presentations of the movie. Therefore if a user watched

a movie multiple times or different parts in different sessions, the implict rating reflects that.

Another contribution of the thesis is the use of different performance evaluation criteria for TV movie recommendation. We evaluate performance of the movie recommendation system by using four evaluation measures. Two of them are the well known information retrieval performance measurements precision and recall. Precision is determined in our system as the number of movies watched by the user in top 10 recommendations divided by 10. High precision means system hits many correct movies in the top 10 recommendation. If a user has watched a lot of content, his/her precision is naturally high. Recall solves this problem since it divides the top 10 hits by the number of movies user *u* watched in the test set. In addition, two other performance evaluation measures are developed in this thesis: normalized precision and rating weighted normalized precision. Precision gets higher as the number of movies that a user watched in the test set increases and it also gets higher as the number of movies in the test set decreases. Normalized precision takes into account the number of the movies in the test set. Ratio of the number of movies watched by a user and the number of movies in the test set can be used as a normalization term for each user. Normalized precision is precision normalized by this ratio. This ratio is proportional to how much better a recommendation is compared to a uniform random recommendation system to a user who watches movies uniformly random. A recommendation system which recommends movies watched by the user with high ratings is more preferable to another system that recommends the same number of watched movies with low ratings. Rating weighted normalized precision (RWNP) performance measure takes into account the users ratings for the test movies. It is computed as the sum of the ratings of the watched movies in the top 10 recommendations and divided by the ratio of the number of movies that are watched by the user in the test set and the total number of the movies in the test set.

The content-based recommendation system uses actor, genre, director and keyword features of movies watched by a user. In the feature extraction phase, first of all a movie-feature matrix which contains the features of all movies in the training set, is created. For a particular user, an existing feature in a watched movie is scaled by the implicit rating for that movie and the sum of the user's weights for the movie's features divided by the number of movies that the user watched in the training set gives the weight of a feature for that user. These features are reference features for the recommendation of the test set movies. If a feature weight for a user is greater than the other feature weights it means that the user gives more importance to this feature than the others. This feature weight computation is done separately for four different feature sets: actor, genre, director and keyword. In the test set when movie *i* will be recommended to the user *u*, firstly features of the movie *i* are extracted. Assume that actor feature set is chosen, which actor features movie *i* contains and whether user *u* watched such a movie which contains one of these actor features is investigated. If user *u* watched a movie which contains the actors of the movie *i* in the training set, then user *u* rating for movie *i* is determined by summation of the actor features weights of the user *u*. While generating ratings according to actor feature set, since usually movies have more than one actor, all available feature weights are summed. On the other hand, according to the director feature set generally there is one director for each movie, so user weights for director features are used directly as ratings for movies. It is observed that ratings generated using the

director feature set are more successful than the others, while the genre feature set is also quite successful.

Content-based recommendations for each feature set are also combined using three different strategies. Before combination, all generated ratings are normalized to 0-1 range using min-max normalization. In the first combination scheme, different feature sets' ratings are summed directly to generate a new rating for a user to a movie. The second combination scheme takes a weighted sum of the ratings for each feature set. The weight of a feature set is determined as the exponential of the negative mean absolute training set error between the actual ratings and the predicted ratings for that feature set. Weighted sum combination gives better results than sum. The third strategy aims to use the feature set which is likely to be the most successful for a particular user. The feature set with the minimum mean absolute training error for the user is chosen as the feature set to be used for test recommendations.

In collaborative filtering, generally explicit feedback recommendation methods where users rank movies explicitly such as likes or dislikes or using scores, are used. However, in TV program recommendation problem, as in many other areas, it is difficult to request the explicit ratings from the user for the programs. Instead of ratings, there is information on how long the user watched an item. For such problems, instead of explicit recommendation methods, implicit methods should be used. In this work, we process the time durations for which users watch the programs to obtain implicit ratings and similar to prior work of others use these ratings for implicit recommendation. The user-movie matrix, which contains the users' ratings for movies can be used to assess similarities between users and movies and hence, for example, movies liked by users similar to the current user can be recommended. However, the user-movie matrix is a very sparse matrix and most user-user and item-item similarities may happen to be just zero. Matrix factorization techniques are used to represent each movie and user in a small number of reduced dimensions where user-item similarities are as close as possible to the ratings given in the training set. We first use the implicit computed ratings as if they are explicit ratings and use explicit matrix factorization methods. While learning the matrix factors, we introduce adaptive learning rate to speed up the learning and we also introduce user/item adaptive regularization. We also use implicit matrix factorization and compare it with the other recommendation methods.

Since matrix factorization is a costly procedure which involves many parameters, we also used count based collaborative filtering to measure user-user similarities. In this method when movie i will be recommended to the user u, first the set of users who watched movie i in the training set is obtained. For each of these users, the count of movies liked by user u and that user is used as a similarity between the users. Count based collaborative filtering predicts ratings as the similarity weighted ratings of the users similar to user u for movie i..

In this thesis, we propose two hybrid movie recommendation systems by combining content-based and collaborative recommendation system ratings linearly. The first hybrid system HybridCommonMovie is obtained by combining content-based system and count based collaborative filtering system. The second one HybridMF is generated by combining content-based system and matrix factorization based collaborative filtering system. A weight parameter is used to adjust the contribution of the methods in the linear combination.

Experiments were performed to assess the performance of the recommendation algorithms for thirteen months of data. Among all the methods experimented with, the best results are obtained with the HybridCommonMovie systems. For this recommendation system, averaged over all users, precision, recall, normalized precision and rating weighted normalized precision results are better than the other recommendation systems. HybridCommonMovie method also is the method which has the smallest number of parameters that need to be adjusted for different datasets, therefore is the preferred recommendation method for the TV recommendation dataset used in this thesis.

1. GİRİŞ

Günümüzde önerilebilecek ürünlerdeki çeşitliliğin artması ve kullanıcıların geçmişte hangi ürünleri tükettikleri bilgisinin tutulması sayesinde öneri sistemlerinin önemi artmaktadır. Genel olarak kullanıcıların her birinin farklı beğenileri olacağından doğru kullanıcıya doğru ürün önerisi yapılması kritik bir problem olarak karşımıza çıkmaktadır. Dijital televizyon yayınları gibi kullanıcılara çok büyük miktarlarda ve çok farklı içerikler sunabilen sistemlerde ise yapılan öneriler çok daha karmaşık olabilmektedir. Genel olarak tüketilen ürünlere benzer ürünlerin önerilmesi şeklinde bir yöntem izlense de, benzer olmayanlar arasından da bir öneri yapılıp yapılamayacağı incelenmelidir.

Öneri sistemlerinde genel olarak kullanıcıların ürünlere verdikleri puanları içeren, doğrudan geribildirimli (explicit feedback) kullanıcı-ürün matrisleri kullanılmaktadır. Puan verilmemiş olsa da, dolaylı geribildirimli (implicit feedback) sistemler ile kullanıcıların ürün alma tarihçesi, hangi ürünlere baktığı gibi veriler kullanılarak kullanıcının bir ürünle ilgili fikri hakkında bilgi edinilebilir. Beraber öneri sistemlerinde kullanıcı-ürün puanlama matrisleri ile ürünlerin kendi aralarındaki ve kullanıcıların kendi aralarındaki benzerlikler hesaplanabilmekte ve öneri için kullanılabilmektedir. Ancak puanlama matrisleri genelde seyrek yapıda olan matrislerdir. Tüm kullanıcıların tüm ürünlere puan vermiş olmaları çok muhtemel değildir. Bu nedenle doğrudan puan matrisleri ile benzerlik hesaplandığında, benzerlikler sıfır çıkabilir. Bu tezde beraber öneri sistemi matris ayrıştırma ile elde edilen matrisleri kullanmaktadır. Matris ayrıştırma yöntemleri ile kullanıcı ve içerik daha az boyutlu uzaylarda gösterilerek benzerliklerin daha doğru hesaplanması sağlanır.

İçerik tabanlı (Content-Based) öneri sistemleri her kullanıcıyı ayrı ayrı ele almaktadır ve kullanılan içerik tabanlı öneri sisteminde film bilgileri ve kullanıcıların önceki izleme bilgilerinden elde edilen dolaylı değerlendirme bilgileri kullanılmaktadır. Beraber (Collaborative) öneri sistemi ise bir kullanıcının izleme bilgileriyle birlikte ona izleme tercihi olarak yakın olan diğer kullanıcıların

bilgilerinden de faydalanır. Ayrıca içerik tabanlı öneri sistemi ve beraber öneri sistemleri birleştirilerek hibrit (Hybrid) bir öneri sistemi oluşturulmuştur.

Bu tez çalışmasında dijital televizyon yayını kullanıcılarına film öneri sistemleri sunulmuştur. Bölüm 1.1'de bu tez çalışmasında geliştirilmesi amaçlanan öneri sistemleri ve tez çalışması kapsamında yapılan özgün çalışmalardan bahsedilmiştir. Bölüm 2'de genel olarak öneri sistemleri incelenip, öneri sistemleri hakkında daha önce yapılan çalışmalardan bahsedilerek literatür araştırması yapılmıştır. Bölüm 3'te TV izleyicilerine film önerisinde karşılaşılan problemlerden bahsedilmiş ve kullanılan veri kümesi hakkında bilgi verilmiştir. Bölüm 4'te tez çalışmasında öneri yöntemi olarak kullanılan içerik tabanlı öneri sistemi, beraber öneri sistemi ve hibrit öneri sistemleri gerçekleme detaylarıyla birlikte anlatılmıştır. Bölüm 5'te geliştirilen öneri sistemlerinin değerlendirilmelerinde kullanılan performans ölçütlerinden bahsedilmiş olup, geliştirilmiş olan öneri sistemlerinin değerlendirmeleri, performans sonuçları ve analizleri gösterilmiştir. Bölüm 6'da sonuçlar değerlendirilmiştir.

1.1 Tezin Amacı

Bu tez çalışmasının amacı ürün öneri sistemi ile izleyicilerin ekran karşısında geçirdikleri zamanı ilgilenecekleri ve faydalanacakları ürünleri izleyerek geçirmelerini ve böylelikle memnuniyetlerini arttırmayı sağlamaktır. Öneri yapılacak filmler hakkında hem içerik hem de dolaylı olarak puanlama bilgisi mevcuttur. İçerik bilgisi, filmler hakkında oyuncu, yönetmen, tür, yapım yılı, filmin yayınlandığı kanal, gün içinde yayınlandığı zaman dilimi gibi bilgiler yanında filmler hakkındaki özet dokümanlarındaki kelimeleri de kapsayacağı için çok yüksek boyutlu olacaktır. Bütün içerik öznitelik boyutlarının kullanılması halinde, hibrit öneri sistemi hem yavaş olacak hem de başarımı düşecektir. Özniteliklerin tümü için ayrı ayrı başarımlar gözlenmiş ve başarımı olumlu yönde etkileyen oyuncu, tür, yönetmen, anahtar kelime özniteliklerinin kullanılmasına karar verilmiştir.

Tez çalışması içeriğinde diğer çalışmalardan özgün olarak, öneri sisteminde kullanılabilecek dolaylı (implicit) puanları kullanan bir puan hesaplama yöntemi ve oyuncu, yönetmen, tür, anahtar kelimeler gibi farklı öznitelik kümelerini kullanan içerik tabanlı öneri sistemi sunulmuştur. Bu aşamada farklı öznitelik kümelerinin davranışları karşılaştırılarak öneri sistemine olan etkileri incelenmiştir.

Tez çalışmasıyla birlikte yapılan bir diğer katkı da beraber filtrelemede kullanılabilen dolaysız matris ayrıştırma yöntemi üzerinde hız ve doğruluk açısından çeşitili iyileştirmeler olmuştur.

Beraber öneri sistemi ve içerik tabanlı öneri sistemine ek olarak, hem içerik hem de diğer kullanıcıların puan değerlerini kullanan bir hibrit öneri sistemi de sunulmuştur.

Ayrıca, öneri sistemlerinin performans değerlendirilmesinde kullanılan ölçütler incelenerek geliştirilen öneri sistemlerinin değerlendirilmesin için yeni ölçütler oluşturulmuştur.

2. ÖNERİ SİSTEMLERİ

Kullanıcılara sunulan ürünlerin çeşitliliğinin artması, kullanıcıların daha önceden hangi ürünleri tükettikleri bilgisinin tutulması sayesinde ürün öneri sistemlerinin önemi artmıştır. Sistemde var olan içeriklerin doğru kullanıcıya önerilebilmesi kritiktir. Kullanıcıların geçmişte izledikleri içerikler, gelecekte izleyecekleri hakkında bilgi verir. Sadece geçmişte izlediklerine benzer olanlar değil, sevme potansiyeli olan diğer içerikler de önerilebilmelidir. Kullanıcıların içerik beğenisi doğrudan kullanıcıdan alınan ya da izleme davranışlarından hesaplanan puanlar ile gösterilir.

Öneri sistemleri, kullanıcıların davranışlarını izleyerek, kullanıcılara doğru ürün önerilerinde bulunmayı amaçlayan sistemlerdir. Bu ürünler film, kitap, müzik ya da elektronik ticaret sitelerindeki herhangi bir ürün olabilir. Öneri sistemleri kullanıcılara var olan bu binlerce ürün içerisinden uygun olanları önererek doğru kullanıcıya doğru ürünün önerilmesini sağlarlar.

Öneri sistemleri temel olarak içerik filtreleme ve beraber filtreleme olmak üzere iki yönteme dayanmakla beraber, farklı özelliklerinin bir araya getirilmesiyle oluşan hibrit öneri sistemleri üzerinde de çalışmalar yapılmıştır.

2.1.1 İçerik Tabanlı Öneri Sistemi

İçerik tabanlı öneri sistemlerinde her kullanıcı ve ürün için birer profil üretilir. Bir ürün için, ürün profilinde tür, oyuncular, anahtar kelimeler gibi alanlar olabilir. Kullanıcı profilinde ise, kullanıcı hakkında toplanan demografik veriler, kullanıcının bazı anketlere verdiği cevaplar olabilir. İçerik filtreleme yöntemlerinde içerik ile ilgili kaliteli ve yeterli bilginin toplanması gereklidir. Bu içerik bilgileri kullanıcının daha önceden izlediği içeriklerin bilgileriyle karşılaştırılır. Benzer olanları tespit edilir ve kullanıcıya bu içerik-içerik benzerliğine göre öneri yapılır.

FIT adı verilen bir televizyon programı öneri sisteminde Goren-Bar ve Glimansky (2004) kullanıcılardan program türleri ve gün içinde hangi saat aralığında program izledikleri bilgisini alarak bir kullanıcı profili oluşturmaktadır. Gün içinde bulunulan saat aralığına göre hangi kullanıcının televizyon izlediği tahmin edilerek içerik

önerisi yapılmaktadır. Eğer sistem hangi kullanıcının TV'yi izlemeye başladığıyla ilgili hatalı bir tahmin yaparsa hatalı içerik önerilebilmektedir.

Cataltepe ve Altınel (2007, 2009) çalışmalarında, müzik önerisi için hem kullanıcıların geçmişte dinlediği parçalar, hem de müzik parçalarının öznitelikleri beraber kullanılmıştır. Ayrıca parçaların popülerlikleri de göz önüne alınmıştır. Değişik özniteliklere göre parçalar kümelendiğinde, kullanıcının beğendiği müzik parçalarının ne kadarının aynı küme içinde kaldığına bir entropi ölçütü kullanılarak bakılarak, her kullanıcının müzik zevkini temsil için farklı öznitelikler kullanılabilmiştir.

Kişisel dijital televizyonlar için içerikten haberdar öneri sistemlerini (context-aware recommendation systems) destekleyen bir çalışmada Santoz de Silva ve diğ. (2009) televizyon programlarının türü, kullanıcının kişisel profili ve kullanıcının içerik bilgilerinden oluşan bir nitelik kümesi kullanmıştır. Bu kişisel bilgiler kullanıcılara çeşitli sorular sorularak elde edilmiştir. Elde edilen özniteliklerin kullanıcılardan çeşitli sorular aracılığıyla elde edilmiş olması bu sistemi demografik bilgi içermeyen yöntemlere göre daha az güvenilir kılmaktadır.

Bir Japon video servis sağlayıcı için sunulan öneri sisteminde Ikawa ve diğ. (2010) kullanıcıların daha önceden izlemiş oldukları içeriklerdeki oyuncu bilgileri ve anahtar kelimeleri kullanmıştır. Ayrıca sunulan sistemi kullanan kullanıcıların gün içinde hangi saatlerde yayını takip ettikleri bilgisi de çalışmada dikkate alınmıştır. Kullanıcının bir oyuncuyu ne kadar süre izlediği bilgisinin, o oyuncunun tüm içerikler içinde ne kadar süreyle bulunduğuna oranı araştırılmıştır. Her bir oyuncu ve anahtar kelime için bu oranlar hesaplanıp her bir film için ortalama bir öznitelik bilgisi elde edilmiştir. Yapılan bu çalışmada değerlendirme ölçütü olarak anma (recall), kesinlik (precision) ve F-ölçütü (F-Measure) kullanılmıştır. Bu çalışmada da değerlendirme yapılırken kullanıcıların geri bildirim yapması sağlanmıştır. Bu tez çalışmasındaki içerik tabanlı öneri sisteminde Ikawa ve diğ. (2010) çalışmasına benzer, ama puanların ve benzer kullanıcıların da kullanıldığı bir yöntem izlenmiştir.

Pandora.com radyo servisinde kullanılan Music Genome Project, Westergren (2011) öneri sisteminde kullanıcılara sistemden istedikleri şarkı ve şarkıcıya uygun radyo istasyonları bulunmaktadır. Sistem şu şekilde çalışmaktadır; kullanıcı dinlemek

istediği şarkı veya şarkıcıyı sisteme girer. Bu bilgilere göre kullanıcıya, belirttiği özelliklere uyan şarkıları çalan radyolar önerilir.

Filmler hakkındaki bilgileri içeren özet dokümanları Mak ve diğ. (2003) çalışmasında kullanılmıştır. Özet dokümanlarındaki kelimeler önce köklerine ayrıştırılmış, tf-idf vektörleri oluşturularak her film bir öznitelik vektörü ile gösterilmiştir. Her kullanıcı için bir sınıflandırıcı oluşturulmuştur. Doküman Sıklığı (DF: Document Frequency) yöntemi ile gereksiz özniteliklerin elenmesi iyi sonuçlar vermiştir. Bu çalışma, filmleri tanımlayan özet kelimeleri yerine onların özelliklerini (tür, önde gelen oyuncular, yönetmen, kazanılan ödüller, yapım tarihi ve yılı) kullanan IMR yöntemi Mak ve diğ. (2003) ile karşılaştırılmıştır. Kullanıcının vermiş olduğu puan sayısının dokümanları tanımlamak için kullanılan kelime sayısına oranının yüksek olduğu durumlarda özete göre sınıflandırmanın daha başarılı olduğu gözlemlenmiştir. Li ve Kim (2004) çalışmasında da kullanıcının izlediği filmler hakkındaki özet bilgileri kullanıcıyı tanımlama amacı ile kullanılmıştır.

Literatürde Debnath ve diğ. (2008) çalışmasında IMDB' de film önerisi için öznitelik olarak film tipi, yazar ve yapım şirketini kullanmışlardır. İki içeriğin benzerliğini o içerikleri beğenen kullanıcı sayısı ile ölçmüşler ve benzerliği özniteliklerin doğrusal bir fonksiyonu olarak modellemişlerdir. Bizim çalışmamızda her kullanıcı için ayrı öznitelikler hesaplanırken Debnath ve diğ. (2008) çalışmasında global bir öznitelik kümesi kullanılmıştır. Luo ve diğ. (2008) çalışmasında beraber öneri sistemi için global ve yerel kullanıcı benzerlikleri kullanılmıştır. Bizim yöntemimizde içerik tabanlı öneri sisteminde yerel kullanıcı benzerlikleri öznitelik üretiminde kullanılmıştır.

2.1.2 Beraber Öneri Sistemi

Beraber filtrelemede, önce bir kullanıcının beğendiği ürünleri beğenen benzer diğer kullanıcılar bulunur. Benzer kullanıcıların beğendiği başka ürünler, en fazla kullanıcı tarafından beğenilenden başlanarak, kullanıcıya tavsiye edilir. Bu yöntemin iyi tarafı içerikten bağımsız olduğu için, herhangi bir ürünün önerisinde kullanılabilmesidir. Beraber filtreleme yöntemleri daha önce hiç karşılaşılmamış olan kullanıcı ya da ürünlere öneri verememe, yani soğuk başlatma (cold-start) problemine sahiptirler. Beraber filtreleme yöntemleri, komşuluk (neighborhood) ve gizli etmen (latent factoring) olmak üzere iki alanda incelenebilir.

Komşuluk yöntemleri kullanıcılar (user) ya da ürünler (item) arasındaki benzerlikleri bulmaya çalışırlar. Bunun için Pearson correlation, cosine similarity gibi benzerlik ölçme yöntemleri kullanılır (Sarwar ve diğ, 2000).

Gizli etmen yöntemlerinde kullanıcıların ürünleri puanlamaları, az sayıda faktörle açıklanmaya çalışılır. Bu faktörler, aksiyon miktarı, erkek/kadın izleyiciye yönelik olma ya da doğrudan anlaşılamayacak başka boyutlarda olabilir. Gizli etmenlerin bulunmasında çoğunlukla matris ayrıştırma (matrix factorization) yöntemleri kullanılır. Matris ayrıştırma yöntemlerinde kullanıcılar da ürünler de değişik faktörlerle belirtilir. Faktörleri aynı olan kullanıcı ya da ürünlerin benzerliği varsayılır.

Beraber filtreleme yöntemi, ilk öneri sistemlerinden olan Tapestry'de Goldberg ve diğ. (1992) kullanılmış olan yöntemdir. Bu yöntemde bir kullanıcının beğendiği içerikler bu kullanıcıya benzer olan diğer kullanıcılara önerilir. Beraber filtreleme yöntemleri için komşuluk ve gizli etmen yaklaşımları mevcuttur.

TiVo televizyon programı öneri sistemi Ali ve Stam (2004) beraber filtrelemenin öğe-öğe benzerliğine dayalı türünü kullanmaktadır. Kullanıcıdan beğendiği/beğenmediği içerikleri uzaktan kumandası ile değerlendirmesi istenir. Bu değerlendirmeler doğrudan ve dolaylı olarak iki farklı şekilde yorumlanabilir.

Koren ve diğ. (2009) çalışmasında Netflix 2006 verisinde, matris ayrıştırma yöntemlerini kullanarak öneri yapılmıştır. Matris ayrıştırma yöntemlerinde kullanıcı-ürün matrisi kullanılarak, kullanıcı ve ürünler daha az boyutlu bir uzayda, ama kullanıcı-ürün matrisindeki benzerlikler korunarak temsil edilir. Matris ayrıştırma yöntemleri, kullanıcı/ürün puanlama farklılıkları (bias) yanında, kullanıcının dolaylı geri bildirimleri, ürün popüleritesi, puanlama farklılıklarının zamana bağlı olarak hesaplanması ve girdilere değişik güven faktörlerinin verilmesini de hesaba katabilir. Koren ve diğ. (2009, 2011) çalışmasında Netflix 2006 verisinde, bütün bu faktörlerin hesaba katılmasının, hata oranını sadece matris ayrıştırmasının kullanılmasına göre düşürdüğü görülmüştür. Fakat sistemde bulunan parametre sayısı da, hesaba katılan değişkenle artmıştır. Matris ayrıştırma yöntemleri kullanıcı-ürün matrisinde bir boyut azaltma olarak görülebilir. TF-IDF gibi basit öznitelik seçimi yöntemleri ayrıca öneri sistemlerinde kullanıcı ya da ürün içeriklerinde boyut azaltma yöntemi olarak da kullanılmıştır (Mak, 2003). Bu tez çalışmasında Koren ve diğ. (2009) ve Mak ve diğ. (2003)'ın matris ayrıştırma yöntemleri temel alınmıştır. Miyahara ve Pazzani (2002)

çalışmasında Bayes sınıflandırıcısı ile öneri yapılmış ve entropiye dayalı bir ölçütle sınıf hakkında en fazla bilgi veren öznitelikleri seçmişlerdir. Film önerisi için, hibrit değil, ama beraber öğrenme için karşılıklı bilgiye dayalı öznitelik seçimi ve örnek ağırlıklandırılmasının Yu ve diğ. (2003) çalışmasında başarım ve hızı arttırdığı gözlemlenmiştir. Benzer şekilde, beraber öğrenmede öznitelik değil de ürün seçimini Baltrunas ve Ricci (2008) çalışmasında yapılmıştır. Görüntü önerisi için, yüksek boyutlu SIFT özniteliklerinin azaltılmasında, Gauss karışımı modellerini kullanarak eğitimsiz öznitelik seçimi yapan bir çalışma da mevcuttur (Boutemedjet ve diğ, 2007). Genel olarak kümeleme (clustering) işlemleri için boyut azaltma üzerine Dash ve diğ. (2002)'de olduğu gibi karşılıklı bilgiye dayalı filtreleme kullanan yöntemler de mevcuttur.

2.1.3 Hibrit Öneri Sistemleri

Farklı öneri sistemlerinin eksik yönlerini tamamlamak için birkaç öneri yönteminin bir arada kullanılmasıyla hibrit öneri sistemleri meydana gelmektedir. Genel olarak beraber öneri yönteminin yetersiz kaldığı noktalarda içerik bilgisini kullanan içerik tabanlı sistemin devreye girmesiyle veya içerik tabanlı sistemin yetersiz kaldığı yerde beraber filtrelemenin kullanılmasıyla, hibrit bir yöntem ortaya çıkmaktadır. Beraber filtrelemede sisteme yeni gelen bir ürünün hangi kullanıcılar tarafından izlendiği ve beğenildiği bilinmediğinden bu ürünün hangi kullanıcıya önerileceği bilinemez. Bu durumda ürünün içerik bilgisine başvurulur. Böylece sistemdeki hangi ürünlerle benzer olduğu bulunur ve ona göre kullanıcılara önerilebilir. İçerik tabanlı sisteme göre öneri yapılırken, sadece öneri yapılacak kullanıcının beğenileri yanında, bu kullanıcıya benzer diğer kullanıcıların beğenilerinden de yararlanılarak öneri yapılabilmektedir.

Hibrit öneri sistemleri Burke (2002), hem kullanıcıların verdiği puanları, hem de kullanıcı ve ürünler hakkındaki bilgileri önerilerde kullanmaya çalışır. Chen ve diğ. (2005) çalışmasında da içerik, kullanıcı puanı ve istatistiksel yöntemler aynı zamanda kullanılmıştır. Hem kullanıcı puanı hem de ürün içeriği bilgisi, Yoshii ve diğ. (2006) çalışmasında Bayes ağları ile kullanılmış ve hibrit sistemle sadece kullanıcı puanı ya da içerik kullanılmasından daha iyi sonuçlar alındığı gözlemlenmiştir.

Hibrit öneri sistemleri kullanıcılara film önerilmesi konusunda kullanılmıştır. Lekakos ve Caravelas (2008) çalışmasında kullanıcı puanlaması için "Movielens" öneri sisteminde toplanmış olan binlerce kullanıcının binlerce filme vermiş olduğu öneriler kullanılmıştır. Filmler hakkındaki içerik bilgileri ise "Internet Movie Database" sitesinde gezilerek toplanmıştır. Bu iki sistemin verdiği öneriler birleştirilerek hibrit bir öneri sistemi oluşturulmuştur.

2.2 Öneri Sistemlerinin Temel Problemleri

2.2.1 Soğuk başlatma problemi

Soğuk başlatma problemi sisteme yeni kullanıcı veya yeni film geldiğinde ortaya çıkan bir problemdir.

Yeni gelen kullanıcının hangi tür filmleri seveceği, ne tür bir izleme davranışı göstereceği bilinmediği için öneri yapmak mümkün olmamaktadır. Bu durumda yeni gelen kullanıcının ilk bir hafta sistemde hangi içerikleri izleyeceğine bakılıp ona göre öneri yapılması bir çözüm olabilir. Ayrıca en popüler içeriklerin önerilmesi yoluna da gidilebilir.

Sisteme yeni eklenen bir filmin hangi kullanıcılara önerileceği yine bir soğuk başlatma problemidir. Film, belli bir süre yayınlandıktan sonra, bu filmi izleyen kullanıcıların verdikleri puanlar ile özniteliklerine ağırlıklar atanır. Böylece film içerdiği özniteliklere ait ağırlık değeri olan kullanıcılara önerilebilir hale gelir.

2.2.2 Seyrek veri problemi

Beraber filtreleme yöntemlerinde kullanıcıların ürünlere verdikleri puanlardan oluşan kullanıcı-ürün matrisleri vardır. Bu matrisler genel olarak seyrek yapıdadırlar. Çünkü öneri sistemlerinde kullanıcılara önerilebilecek binler (1000) mertebesinde ürün bulunur. Ancak kullanıcılar bu ürünlerin ancak belli bir kısmına puan verirler. Bu nedenle kullanıcı-ürün matrisleri seyrek bir yapıya sahip olmaktadırlar. Komşuluk tabanlı yöntemlerde kullanıcılar ve ürünler arasındaki benzerlikler bu seyrek matris üzerinden hesaplandığında benzerlikler genel olarak sıfır çıkmaktadır.

2.2.3 Ölçeklenebilirlik problemi

Öneri sistemlerinde birçok kullanıcı için birçok ürün hakkında öneri üretilmeye çalışılır. Ürün ve kullanıcı sayılarının artmasıyla verinin ölçeklenebilirliği

zorlaşmaktadır. Beraber öneri sistemlerinde kullanılan komşuluk tabanlı yöntemlerde kullanıcılar *m* boyutlu vektorlerle, ürünlerde *n* boyutlu vektörlerle temsil edilir. Bu durumda kullanıcı-kullanıcı ve ürün-ürün benzerliklerini hesaplamak sisteme önemli miktarda yük getirecektir. Bu problemin önüne geçmek için beraber öneri sisteminde yöntem olarak matris ayrıştırma kullanılmaktadır (Koren ve diğ, 2009). Matris ayrıştırmada *m* toplam kullanıcı sayısı, *n* toplam ürün sayısı olmak üzere her kullanıcı ve ürün $f << m,n$ boyutunda vektörle temsil edilirler.

2.2.4 Aşırı özelleşme problemi

Öneri sistemlerinin etkilendiği problemlerden biri de aşırı özelleşme problemidir. Sistem kullanıcılara sadece önceden tükettiklerine benzer ürünler önerir. Bu durumda kişiye beğenme potansiyeli olabilecek diğer ürünler önerilemez. Oysaki öneri sistemlerinin hem kullanıcının geçmişteki izleme davranışına uygun içerikler önermesi hem de sevebileceği diğer ürünleri önermesi beklenir. Hibrit yöntemler kullanılarak aşırı özelleşme problemi çözülebilmektedir. İçerik tabanlı sistem kullanıcının kendi beğenilerine göre öneri yapılmasını sağlarken beraber filtreleme yöntemi bu kullanıcıya benzer başka kullanıcıların beğenilerinden de yararlanarak öneri yapabilmektedir. Bu iki sistemin bir arada kullanılmasıyla elde edilen hibrit sistemler aşırı özelleşme problemine çözüm getirmektedir.

3. TV İZLEYİCİLERİNE FİLM ÖNERİ PROBLEMİ

Öneri sistemlerinde kullanıcıların geçmişte tükettikleri ürünler gelecekte tüketecekleri hakkında fikir verir. Kullanıcıların geçmişte tükettikleri ürünler için yaptıkları değerlendirme bilgilerinden yararlanılır. Eğer beğenme/beğenmeme, 0-5 arasındaki puan değerleri gibi doğrudan değerlendirme bilgileri mevcut ise bu bilgiler kullanılarak bir öneri sistemi oluşturulabilir. Eğer doğrudan puan bilgileri mevcut değilse, bu değerlerin dolaylı olarak elde edilmesi gerekmektedir. Doğrudan veya dolaylı puan değerleri kullanılarak yapılan önerilerin sonuçları farklı olabilmektedir. Kullanıcı bir içeriğe doğrudan puan vererek beğenisini belirtmiş olsa da, bu puanı rastgele olarak vermiş de olabilir. Kullanıcının izleme davranışlarından yararlanılarak elde edilen puanlar da ise yine bir takım gürültüler vardır. Kullanıcının bir filmi izleme süresi oldukça fazla göründüğü halde kullanıcı bir başkasından dolayı bu filmi izliyor olabilir ya da televizyon açık kalmış ve kullanıcının zevkiyle alakasız bir sürü film tamamen izlendi ve dolayısıyla yüksek puan aldı diye kaydedilmiş olabilir.

TV programlarının önerilmesinde karşılaşılan bir diğer problem televizyonun ev halkının farklı üyeleri tarafından izleniyor olması ve dolayısıyla tek bir kişinin beğenilerinden ziyade ancak aile üyelerinin genel izleme bilgilerinin elde edilebiliyor olmasıdır. Bu durumda önerilerin ev halkının tamamını memnun edecek şekilde yapılabilmesi problemi karşımıza çıkmaktadır.

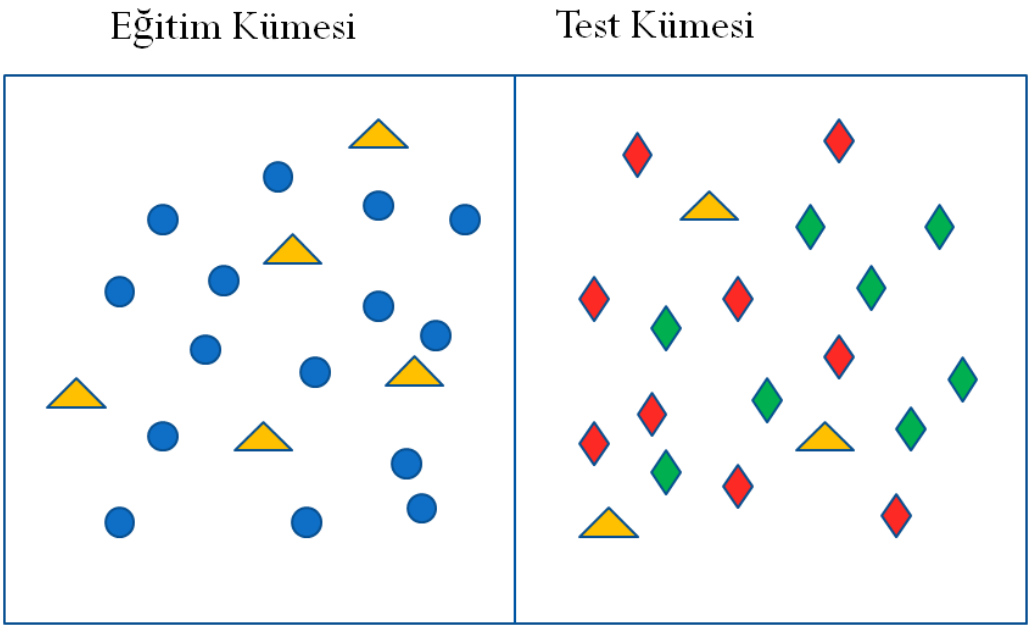
Özellikle dijital içerik yayıncılığı yapan sistemlerde her kullanıcı için önerilebilecek içerik kümesi farklı olabilmektedir. Bunun nedeni farklı kullanıcıların farklı içerikleri izleyebiliyor olmalarıdır.

3.1 Veri Kümesi

Kullanılan veri kümesi kullanıcıların on üç aylık izleme verilerinden oluşmaktadır. Bu verinin ilk on iki aylık kısmı eğitim kümesi için geriye kalan bir aylık kısmı da test kümesi için kullanılmıştır. Bu veri kümesi kullanıcıların izledikleri filmlerin

oyuncu, tür, anahtar kelime gibi içerik bilgileri ve bu filmlere dolaylı olarak verdikleri puanlardan oluşmaktadır. Esasında puan bilgisi hali hazırda mevcut değildir, kullanıcıların filmleri izleme sürelerinden yararlanılarak puan üretimi bizim tarafımızdan yapılmıştır. Kullanıcılardan her zaman doğrudan puan almak kolay değildir, alınsa bile bu veri yeterince güvenilir olmayabilir. Ayrıca kullanıcıların tüm ürünlere puan vermeleri de mümkün değildir. Dolayısıyla puanı alınan ürün sayısı kısıtlı kalır.

Şekil 3.1'de mavi daireler eğitim zamanında yayınlanmış ve en az bir kullanıcı tarafından izlenmiş filmlerdir. Sarı üçgenler hem eğitim hem test zamanında yayınlanan filmleri, kırmızı baklava dilimleri sadece test zamanında yayınlanmış ve en az bir kullanıcı tarafından izlenmiş filmleri, yeşil baklava dilimleri test zamanında yayınlanan ve hiç kimse tarafından izlenmemiş filmleri göstermektedir.



Şekil 3.1 : Eğitim ve test kümesi.

Kullanıcılara test kümesindeki filmler arasından öneri yapılmaktadır. Test kümesinde eğitim kümesinde olan filmler olabildiği gibi hiç kimse tarafından izlenmemiş filmler de bulunur. Test kümesindeki filmlerin bir kısmı yine kullanıcılar tarafından izlenmiştir. Ancak izlenmeyen filmler de vardır. Öneri aşamasında kullanıcılara test zamanında hem izledikleri hem de izlemedikleri filmler için öneri üretilir.

3.1.1 İçerik verisi

Filmlerin türü, oyuncuları, yönetmeni gibi bilgiler filmler için öznitelik olarak kullanılmıştır. Ayrıca filmlerin özet verilerinden de tanımlayıcı anahtar kelimeler çıkarılmıştır. Tüm bunlar film öznitelikleri olarak kullanılmaktadır. Filmlerin yapım yılı, yayınlandığı kanal, günün hangi saatinde yayınlandığı gibi bilgiler kullanılarak bazı sonuçlar alınmıştır; ancak öneri kalitesinde çok etkileri olmadığına karar verilmiş, sonraki testlerde göz ardı edilmişlerdir. Çizelge 3.1'de filmlere ait bazı öznitelik bilgileri gösterilmiştir.

Çizelge 3.1 : Film içerik bilgileri.

Filmler	Tür	Oyuncu	Yönetmen	Anahtar Kelime
Esaretin bedeli	Suç, Dram	Tim Robbins, Morgan Freeman and Bob Gunton	Frank Darabont	hapishane, dost, cinayet, esaret
Uzaylı Kuklalar	Aile	Dave Goelz, Bill Barretta	Tim,Hill	gezegen, kukla, macera, merak, sev, sevim, sevimli, uzay,
Üç Adam ve Bir Bebek	Komedi, Romantik	Tom Selleck, Steve Guttenberg, Ted Danson, Nancy Travis	Leonard Nimoy	apartman, bebek, bekar, daire, ekmek, haberdar

3.1.2 Özet verilerinden önemli kelimelerin çıkarılması

Özet verilerinden önemli kelimeleri çıkarabilmek için öncelikle filmlerin özet bilgileri Zemberek (Url-1) programına verildi. Zemberek programı açık kaynak kodlu, platform bağımsız, Türk dilleri için geliştirilmiş bir doğal dil işleme kütüphanesidir. Zemberek programı ile ek almış kelimelerin köklerini bulunur. Kök bulmak için programın içinde iki seçenek vardır; ilkinde kelimedeki her yapım ekinin teker teker atılmasıyla tüm kökler bulunurken diğerinde tüm yapım ve çekim

eklerinin atılmasıyla tek bir kök bulunur. Ancak bu durumda kelimenin asıl kullanım anlamından farklı bir anlama gelen bir kök seçilmiş olabilir. Sistemimizde bu duruma mahal vermemek için kelimelerin teker teker yapım eklerinin çıkartılmasıyla oluşan tüm köklerin elde edilmesini sağlayan seçenek kullanıldı ve özet verilerindeki kelimeler köklerine ayrıldı. Zemberek programı özet verilerindeki sadece Türkçe kelimeleri çözümleyebilmektedir. Bu nedenle özet verilerinde geçen Newyork, Bahamas gibi genelde yer isimlerini içeren yabancı özel isimler veri kaybı olmaması açısından anahtar kelimeler arasına katılmışlardır.

Kök halindeki kelimelerin özet verisinde bulunduğu film için ne ölçüde önemli olduğunu belirlemek için, kelime ağırlıkları tf-idf yöntemi Jones (2004) ile hesaplanır. Tf-Idf bir terimin bir dokümandaki değerinin istatistiksel bir ölçüsüdür.

$$idf_t = \log_{10} \frac{N}{df_t} \tag{3.1}$$

idf değeri çok sık kullanılan terimlerin bilgi içermediğini belirtir ve bu terimleri eleyerek filtre görevi görür. N toplam doküman sayısını, df_t t terimini içeren dokümanların sayısını, idf_t t teriminin ne kadar bilgi içerdiğini belirler.

$$(tf - idf)_{t,d} = tf_{t,d} * idf_t \tag{3.2}$$

$tf_{t,d}$ (term frequency) t teriminin d dokümanındaki frekansıdır. $tf_{t,d}$ ve idf_t terimlerinin birleştirilmesiyle her terimin her dokuman için ağırlığı belirlenmiş olur.

Tf-Idf değeri belli bir eşik değerinde olan kelimeler içeriklerimiz için anahtar kelime olarak seçilmiştir.

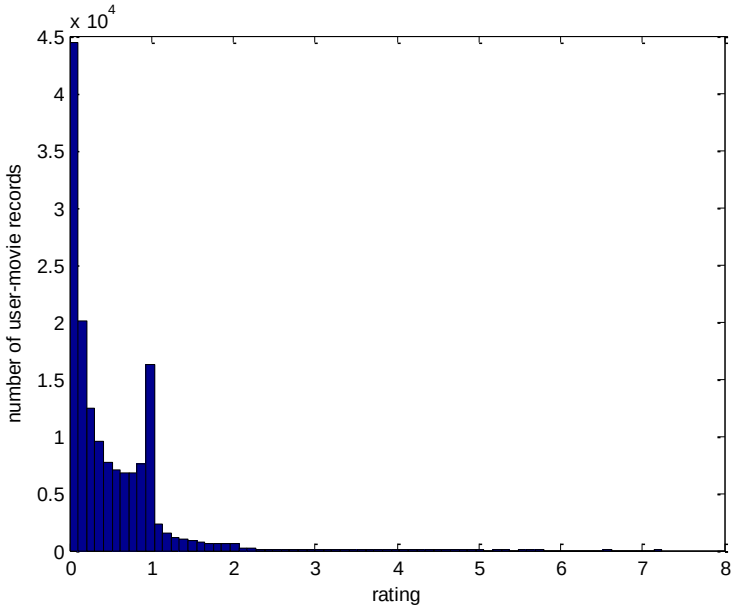
3.1.3 Film izleme süresinin derecelendirme ya da beğeniye dönüştürülmesi

Sistemde kullanıcıların filmlere doğrudan verdikleri puanlar bulunmamaktadır. Bu nedenle puanlamaların dolaylı olarak elde edilmesi gerekmiştir.

Bir ürünü (i) bir kullanıcının (u) bir yıl boyunca izleme süresi $t(u,i)$, ürünün toplam süresi ise t_i olsun. Kullanıcının bu ürün için normalleştirilmiş izleme süresini, Hu ve diğ. (2008)'e benzer şekilde şöyle tanımladık:

$$r(u,i) = \frac{t(u,i)}{t_i} \tag{3.3}$$

Eşitlik (3.3) testler için kullanılan veriye uygulandığında [0-7.2] arasında değişen değerler alınmaktadır.



Şekil 3.2 : Normalleştirilmiş izleme zamanlarının dağılımı.

Şekil 3.2'de $r(u,i)$ nin en yoğun değerlerinin bulunduğu aralıklar gösterilmiştir. İçeriklerin çoğu en fazla bir defa seyredilmektedir. Fakat daha sonraki defalar izlenen içerikler de mevcuttur.

Eğer bir kullanıcı ürünü hiç izlememiş ise, o ürün için $r(u,i)$ derecelendirme değerinin olmadığı varsayılmaktadır.

4. TEZDE KULLANILAN FİLM ÖNERİ SİSTEMLERİ

4.1 İçerik Tabanlı Öneri Sistemi

Geliştirilen içerik tabanlı öneri sisteminde kullanıcılara filmlerin öznitelikleri için ağırlıklar üretilmiştir. Şekil 4.1'de görülüğü gibi öznitelik olarak filmlerin oyuncu, yönetmen, tür, anahtar kelime gibi öznitelikleri yanında filmlere verdikleri puanlar da kullanılmıştır.

Şekil 4.1 : Film öznitelikleri ve puan.

4.1.1 Öznitelikler için ağırlık hesaplama

Film öneri sistemlerinin amacı kullanıcının sevebileceği filmleri tüm filmler arasından seçip kullanıcılara önermektir. Bu tez çalışmasının da amacı kullanıcı beğenilerini tahmin edebilmektir. Bunun için filmlerin oyuncu, tür, yönetmen, anahtar kelime, yapım yılı, yayınlandığı kanal, yayınlandığı zaman dilimi

özniteliklerini kullandık. Eğitim kümesindeki filmlerde oyuncu öznitelik kümesinde 4716 adet farklı oyuncu, yönetmen öznitelik kümesinde 1927 adet farklı yönetmen, tür öznitelik kümesinde 34 adet farklı tür, kanal öznitelik kümesinde 10 adet farklı kanal, zaman dilimi öznitelik kümesinde 5 adet farklı zaman dilimi, yapım yılı özniteliğinde 5 adet farklı yapım yılı özniteliği bulunmaktadır. Oyuncu öznitelik kümesinde Brad Pitt, Harrison Ford, Engin Günaydın gibi oyuncular varken; tür öznitelik kümesinde komedi, dram, korku, gerilim gibi film türleri bulunmaktadır. Yapım yılı öznitelik kümesi, var olan tüm yılları 1980öncesi, 1980'ler, 1990'lar, 2000'ler, 2010'lar olmak üzere 5'e bölerek elde edildi. Günün saatleri 00:00-06:59, 07:00-11:59, 12:00-16:59, 17:00-19:59 ve 20:00-23:59 şeklinde 5 zaman dilimine ayırıldı ve her biri bir öznitelik olarak kullanıldı. Yönetmen öznitelik kümesinde Frank Darabont, Francis Ford Coppola, Jonathan Demme gibi yönetmenler, anahtar kelime kümesinde filmlerin özetlerinden tf-idf yöntemiyle elde ettiğimiz anahtar kelimeler bulunmaktadır.

user u	j_0	j_1	j_2	j_3	$j_{4 .....}$	Puan
i_0	1	1	0	0	0	0.5
i_1	0	1	0	0	0	0.3
i_2	1	1	1	0	0	0.9
i_3	1	0	0	1	0	0.7
i_4	0	0	0	1	0	0.2
i_5	1	0	0	1	0	1.0
i_6	1	0	0	1	0	0.44
i_7	0	1	0	0	0	0.67
i_8	1	0	0	1	0	0.2
$w_k(u,j)$	0.42	0.26	0.1	0.26	0	-

Şekil 4.2 : Kullanıcı *u* için film-öznitelik matrisi.

Tf-idf hesaplamasında df (document frequency) değeri 3000 den fazla olan anahtar kelimeler kullanılmamıştır. Böylelikle 3000'den daha az sayıda dokümanda geçen

kelimelerin belirleyici olabileceği düşünülerek onların öznitelik olarak atanmasına karar verilmiştir.

Sistemde her bir kullanıcıya geçmişteki kendi beğenilerine dayanarak öneri yapılmaktadır. Her kullanıcıya, o kullanıcının beğenilerine göre öneri yapıldığı için her bir özniteliğin o kullanıcıya has öznitelik ağırlığının bulunması gerekmektedir. Bu ağırlığı belirlemek için ise kullanıcının o filme verdiği puanlar kullanılmıştır. Bu durum film-öznitelik matrisinin oluşturulmasını gerektirmiştir. Örnek olarak hazırlanan film-öznitelik matrisi Şekil4.2'de görülmektedir. Bu matriste eğitim kümesindeki tüm filmlerin $j_0, j_1, \ldots$ şeklinde gösterilen öznitelikleri bulunmaktadır. Seçilen kullanıcının izlediği $i_0, \ldots, i_8$ filmlerinde bu öznitelikleri bulunduran filmlerin matristeki değeri 1 olarak atanmaktadır, aksi halde bu değer 0 olarak belirlenir. Matrisin en son kolonunda ise *u* kullanıcısının izlediği filmlere verdiği puanlar bulunmaktadır. j_4 özniteliğinde olduğu gibi bazı kullanıcılar bazı öznitelikleri bulunduran hiçbir filmi izlememiş olabilirler. *u* kullanıcısının *k* öznitelik kümesi için, j öznitleiğine ait ağırlık şu şekilde belirlenir:

$$w_k(u, j) = \frac{1}{I_u^{train}} \sum_{i \in I_u^{train}} x_{k,u}(i, j) r(u, i) \qquad \textbf{(4.1)}$$

k öznitelik kümelerini göstermek üzere $k \in$ {oyuncu, tür, yönetmen, anahtar kelime, yapım yılı, yayınlandığı kanal, yayınlandığı zaman dilimi} dir ve $w_k(u, j)$ *u* kullanıcısı için *k* öznitelik kümesindeki *j* özniteliğinin ağırlığını ifade eder. $r(u,i) \in \Re$ kullanıcı *u*'nun film *i*'ye verdiği puandır. $x_{k,u}(i, j) \in \{0,1\}$ ise *i* filminde *j* özniteliği bulunuyorsa 1, bulunmuyorsa 0 değerini alan bir fonksiyondur. I_u^{train} *u* kullanıcısının eğitim kümesinde izlediği filmlerin listesidir. *u* kullanıcısı için her j özniteliğinin hesaplanan $w_k(u, j)$ ağırlıkları Şekil 4.2'in en alt satırında verilmiştir. j_0 özniteliğinin ağırlığı, içerisinde bu özniteliği bulunduran $i_0, i_2, i_3, i_5, i_6, i_8$ filmlerine kullanıcının verdiği puanlar toplanıp, kullanıcının izlediği toplam içerik sayısına bölünüp normalleştirilerek bulunur.

4.1.2 İçerik tabanlı puan tahmin etme yöntemi

Her bir kullanıcı için bulunan öznitelik ağırlıkları, önerilecek ürünlerin puan tahmininde kullanılır.

k öznitelik kümesinde, *u* kullanıcısına *i* filmi için tahmin edilen puan:

$$r_k(u,i) = \sum_{j \in D_{k,i}} w_k(u, j) \tag{4.2}$$

olarak hesaplanır.

$D_{k,i}$, *i* filminin *k* öznitelik kümesinde olan özniteliklerinin kümesidir. (4.2)'deki tahmini puan öznitelik sayısı ile normalleştirilebilir:

$$r'_k(u,i) = \frac{1}{\left|D_{k,i}\right|} \sum_{j \in D_{k,i}} w_k(u, j) \tag{4.3}$$

$\left|D_{k,i}\right|$ *i* filminde bulunan *k* öznitelik kümesine ait özniteliklerin sayısıdır. Bazı filmlerin öznitelik kümelerinden birden fazla öznitelik gelebilmektedir. Oyuncu öznitelik kümesi buna örnektir. Bir filmin genelde birden fazla oyuncusu olacağından her bir oyuncunun ağırlığı toplanarak puan elde edilebileceği gibi (4.2), bu toplam, geçen oyuncu sayına bölünerek ağırlık toplamı normalleştirilebilir (4.3). Bu iki puan hesaplama yönteminin karşılaştırılmasıyla elde edilen sonuçlar Deneyler kısmında bulunmaktadır. Deneylerde eşitlik (4.2)'ye göre daha iyi öneri yapıldığı görülmüştür. O nedenle sonraki işlemlerde puan tahmin formülü olarak (4.2) kullanılmıştır.

4.1.3 Farklı öznitelik kümelerinden gelen puanların birleştirilmesi

Her bir öznitelik kümesinde farklı sayıda öznitelik olduğundan, öznitelik kümelerine göre hesaplanan tahmini puanların $r_k(u,i)$ her biri farklı bir aralıkta olacaktır. Oysa farklı öznitelik kümelerinden gelen tahmini puanları birleştirmek ve her öznitelik kümesinin etkisini görmek için tahmini puanların aynı aralıkta olmaları şarttır. Bu şartı sağlayabilmek için tahmini puanlar Min-Max normalizasyon yöntemiyle normalleştirilirler. Normalleştirme işlemi şu şekilde yapılmıştır:

$$r_k^N(u,i) = (r_k(u,i) - mR_{u,k}) / (MR_{u,k} - mR_{u,k}) \tag{4.4}$$

$r_k(u,i)$, u kullanıcısının i filmine k özniteliğine göre vereceği puan tahminini ifade etmektedir. $mR_{u,k}$ u kullanıcısının k özniteliğine göre eğitim kümesinde vereceği en düşük puan tahminin, $MR_{u,k}$ ise en yüksek puan tahminini göstermektedir. Elde edilen $r_k^N(u,i)$ değeri, tahmini puanın normalleştirilmiş halidir.

Normalleştirilmiş puanlar üç yönteme göre birleştirme işlemine tabi tutulmuşlardır. İlk yöntemde u kullanıcısının i filminin oyuncu, yönetmen, tür, zaman dilimi, kanal, yapım yılı, anahtar kelime özniteliklerinden gelen puanları toplanır:

$$r_{sum}^N(u,i) = \sum_k r_k^N(u,i) \qquad \textbf{(4.5)}$$

Eşitlik (4.5)'te bulunan puan değeri ile yapılan doğru öneri miktarları Deneyler bölümünde grafiklerle gösterilmiştir.

Eğitim kümesindeki hata ile ters orantılı birleştirme yöntemine göre yapılan birleştirme işlemi de şu şekildedir:

$$r_{\exp sum}^N(u,i) = \sum_k r_k^N(u,i) * \exp(-E_{u,k}) \qquad \textbf{(4.6)}$$

$E_{u,k}$, eğitim kümesinde u kullanıcısının k özniteliğine göre hesaplanan puanlarının gerçek puanlarla arasındaki ortalama mutlak hatasıdır (Mean Absolute Error) (4.7).

$$E_{u,k} = \frac{1}{\left|I_u^{train}\right|} \sum_{i=1}^{\left|I_u^{train}\right|} \left| r^N(u,i) - r_k^N(u,i) \right| \qquad \textbf{(4.7)}$$

(3.3)'e göre hesaplanan u kullanıcısının i filmine verdiği puan (4.8) eşitliğine göre normalleştirilmiştir:

$$r^N(u,i) = (r\ (u,i) - mR_u)/(MR_u - mR_u) \qquad \textbf{(4.8)}$$

Ortalama mutlak hatanın negatif üssel değeri ile çarpılan farklı özniteliklere göre hesaplanmış normalleştirilmiş puanların toplamı ile oluşan puanlar, daha iyi öneriler yapılmasını sağlamıştır.

Puanların birleştirilmesi yerine, her kullanıcı için eğitim kümesinde en iyi performans veren öznitelik kümesine göre bulunan puanların kullanılması da başka bir yöntemdir:

$$r^{N}_{bestMAE} = r_{k^*}(u,i) \tag{4.9}$$

(4.9)'da k^*, u kullanıcısı için eğitim kümesinde en düşük MAE'ye sahip öznitelik kümesidir:

$$k^* = \arg\min_{k} E_{u,k} \tag{4.10}$$

4.2 Beraber Öneri Sistemi

Beraber öneri sisteminde yöntem olarak büyük boyutlu ve seyrek puanlama matrisini faktörlerine ayırma ve kullanıcı/ürün benzerliklerini daha sağlıklı hesaplama amacı ile matris ayrıştırma yöntemleri kullanılmıştır.

4.2.1 Matris ayrıştırma

Matris ayrıştırma yöntemlerinde kullanıcılar da ürünler de değişik faktörlerle belirtilir. Faktörleri aynı olan kullanıcı ya da ürünlerin benzer olduğu varsayılır.

Öneri sistemlerinde genel olarak kullanıcıların ürünlere verdikleri puanları içeren (dolaysız geri bildirimli) matrisler kullanılır. Puan verilmemiş olsa da, dolaylı geri bildirimli sistemler ile kullanıcıların ürün alma tarihçesi, hangi ürünlere baktığı gibi veriler kullanılarak kullanıcının bir ürün hakkındaki tercihi hakkında bilgi edinilebilir.

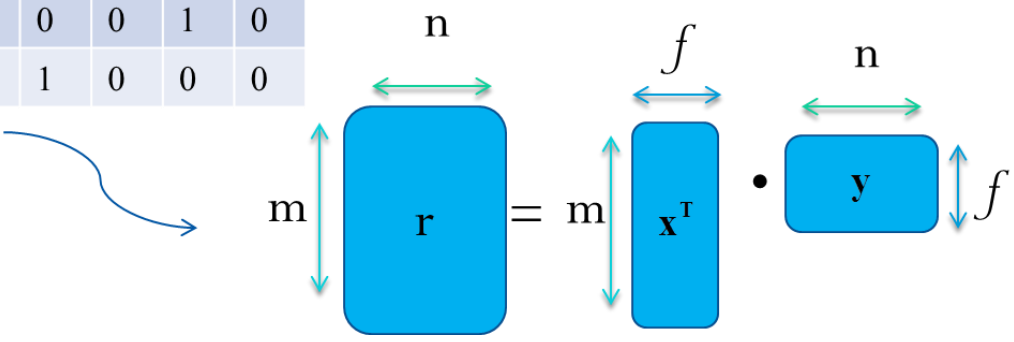
	i1	i2	i3	i4	i5
u1	1	1	1	0	0
u2	0	0	1	0	0
u3	0	1	0	1	1
u4	1	0	1	0	1
u5	0	0	0	1	0
u6	1	1	0	0	0

Şekil 4.3 : Kullanıcı-ürün puanlama matrisi.

Şekil 4.3'te görülen kullanıcı-ürün puanlama matrisleri öneri sistemlerinin önemli bir parçasıdır; fakat genellikle her kullanıcının az sayıda ürüne oy vermesi nedeni ile çok seyrektirler. Dolayısı ile beraber öneri sistemlerinde ürün-ürün, ürün-kullanıcı, ya da kullanıcı-kullanıcı benzerlik hesapları puanlama matrisi doğrudan kullanılarak yapılırsa genellikle sıfır çıkabilir.

Matris ayrıştırma yöntemlerinde, kullanıcı-ürün matrisi kullanılarak kullanıcı ve ürünler daha az boyutlu bir uzayda, ancak kullanıcı-ürün matrisindeki benzerlikler korunarak temsil edilir. Matris ayrıştırmasında, her kullanıcı ve ürün $f << m,n$ boyutlu kullanıcı $\mathbf{x}_u$ ve ürün faktörleri $\mathbf{y}_i$ ile özetlenmeye çalışılır. f boyutu büyüdükçe bilinen puanlamaların değerleri matris faktörü çarpanlarına daha yakın olmaktadır, fakat büyük f değerleri için faktörlerin eğitim kümesini ezberlemesi (overfitting) mümkündür. Büyük f değerleri sistemin yavaş çalışmasına da sebep olmaktadır. Çünkü bu durumda yine kullanıcılar ve ürünler çok büyük boyutlu vektörler ile temsil edilmiş olmaktadır.

	i1	i2	i3	i4	i5
u1	1	1	1	0	0
u2	0	0	1	0	0
u3	0	1	0	1	1
u4	1	0	1	0	1
u5	0	0	0	1	0
u6	1	1	0	0	0



Şekil 4.4 : Puan matrisinin kullanıcı ve film matrislerine ayrıştırılması.

Matris ayrıştırma yöntemlerine çeşitli eklemeler yapılabilmektedir. Bu eklemeler popüler olan ürünlerin seçilmesi, eğer kullanıcı doğrudan bir puan veriyorsa bu puanın kullanılması şeklinde olabilir. Ayrıca zamanla kullanıcıların değişen zevklerini belirten faktörlerin katılması da söz konusu olabilmektedir.

4.2.2 Dolaysız geri bildirimli matris ayrıştırma

Dolaysız geri bildirimli Koren ve diğ. (2009) yöntemlerinde $u \in \{1,..,m\}$ kullanıcısının, $i \in \{1,..,n\}$ ürünü için derecelendirmesini $r(u,i)$'nin belirttiği varsayılır. Dolaysız puan kullanıcıların doğrudan içeriklere verdikleri puandır. Bizim sistemimizde kullanıcıların filmlere doğrudan verdikleri puanlar bulunmamaktadır. Ancak dolaysız yöntemleri de deneyebilmek için Bölüm 3.1.3'te eşitlik (3.3)' e göre hesaplanan puanlar, belli bir eşik değerine göre dolaysız puan haline şu şekilde getirilmiştir:

$$r_{\text{dolaysiz}}(u,i) = \begin{cases} 1, r(u,i) < \theta \\ 2, r(u,i) \geq \theta \end{cases} \quad \textbf{(4.11)}$$

θ belirlenen eşik değeridir. *u* kullanıcısı bu eşik değerin altında kalacak şekilde bir izleme yaptıysa *i* filmine vereceği dolaysız puan 1, aksi durumda 2 olacaktır. Eğer bir kullanıcı ürünü hiç izlememiş ise, o ürün için derecelendirme değerinin olmadığı varsayılmaktadır.

K bu şekilde derecelendirilmiş (u,i) ikililerini göstersin. Kullanıcı ve ürün faktörleri öğrenilirken dolaysız geri bildirimli yöntemlerde şu hata fonksiyonunun en düşük değerinin bulunması gerekir:

$$E_{\text{exp}} = \min \sum_{(u,i)\in K} (r_{\text{dolaysiz}}(u,i) - \mathbf{x}_u^T \mathbf{y}_i)^2 + \lambda(\|\mathbf{x}_u\|^2 + \|\mathbf{y}_i\|^2) \quad \textbf{(4.12)}$$

(4.12)'de düzenleme katsayısı λ arttırıldıkça faktörlerin çok büyük değerler alması engellenerek çözüm daha düzenli hale gelmektedir. Bu parametre genellikle çapraz doğrulama yöntemi ile bulunmaktadır. (4.12)'nin $\mathbf{x}_u$ ve $\mathbf{y}_i$'ye göre türevlerinin alınması ve bayır inişi yöntemi ile $\mathbf{x}_u$ ve $\mathbf{y}_i$ vektörlerinin bir aşamadan diğerine nasıl güncelleneceği bulunur:

$$\begin{aligned} \mathbf{x}_u &\leftarrow \mathbf{x}_u + \gamma(e_{ui}.\mathbf{y}_i - \lambda.\mathbf{x}_u) \\ \mathbf{y}_i &\leftarrow \mathbf{y}_i + \gamma(e_{ui}.\mathbf{x}_u - \lambda.\mathbf{y}_i) \end{aligned} \quad \textbf{(4.13)}$$

Eşitlik (4.13)'te *(u,i)* ikilisinde yapılan hatayı $e_{ui} = r_{ui} - \mathbf{x}_u^T \mathbf{y}$, öğrenme hızı parametresini ise γ göstermektedir.

Öğrenme hızı parametresi γ, yapay sinir ağları, lojistik regresyon gibi modeller öğretilirken, bütün öğretim boyunca aynı seçilmek yerine adaptif olarak değiştirilmektedir. Bu çalışmada, öğrenme hızı parametresi şu şekilde değiştirilmiştir:

$$\gamma \leftarrow \begin{cases} \gamma + \in, E_{\exp} & azalma \\ \frac{\gamma}{2}, E_{\exp} & artma \end{cases} \quad \textbf{(4.14)}$$

(4.14) sayesinde değişik denemeler sırasında γ parametresinin değerinin en iyisinin bulunmasına gerek kalmamıştır. Böylece faktörlerin en iyilenmesinin hızlandırılması sağlanmıştır.

Eşitlik (4.12)'de düzenleme ile ilgili terimin katsayısı olan λ parametresi bütün kullanıcı ve ürünler *(u,i)* için eşit olarak alınmıştır. Öte yandan, eğer bir ürünü çok fazla sayıda kullanıcı izlemişse ya da bir kullanıcının izlediği ürün sayısı çok fazla ise, böyle ürün ya da kullanıcılar için düzenlemeye daha az ağırlık verilmesi yerinde olacaktır. Eğer kullanıcı ve ürün faktörlerinin boyutları uzun ise düzenleme daha çok, küçük ise daha az olmalıdır. Bu etkileri sağlamak için bu çalışmada eşitlik (4.12) yerine şu fonksiyonun en iyilenmesi önerilmektedir:

$$E_{\exp Norm} = \min_{x^*, y^*} \sum (r_{ui} - x_u^T y_i)^2 + \lambda f \left(\sum_u \frac{\|x_u\|^2}{I_u^{train}} + \sum_i \frac{\|y_i\|^2}{I_i^{train}} \right) \quad \textbf{(4.15)}$$

(4.15)'te I_u^{train} bir *u* kullanıcısının izlediği toplam içerik sayısını, I_i^{train} ise *i* içeriğinin kaç kullanıcı tarafından izlendiğini gösterir.

Dolaysız geri bildirimli yöntemde λ değerinin her kullanıcı için izlediği film sayısıyla ilişkili bir değer almasıyla ve öğrenme parametresinin uyarlamalı olarak değiştirilmesiyle alınan sonuçlar Koren ve diğ. (2009)'da önerilen yöntem ile aldığımız sonuçlardan daha iyi çıkmıştır (Cataltepe ve diğ, 2012). Ancak sistemde dolaylı puanların kullanılması söz konusudur. Bu nedenle bir sonraki bölümde detayları anlatılan dolaylı geri bildirimli matris ayrıştırma yöntemi kullanılmıştır.

4.2.3 Dolaylı geri bildirimli matris ayrıştırma

Matris ayrıştırmada *m* tane kullanıcının *n* tane film için beğenilerinin tutulduğu R matrisi:

$$R_{mxn} = X_{mxf} Y_{fxn} \tag{4.16}$$

şeklinde gösterilir. Herhangi bir *u* kullanıcısının *i* filmine vereceği puan :

$$r_{mfc}(u,i) = \mathbf{x}_u^T \mathbf{y}_i \tag{4.17}$$

şeklinde hesaplanabilir. Burada $\mathbf{x}_u$ ve $\mathbf{y}_i$ sırasıyla (4.21) ve (4.22)'teki gibi hesaplanır.

Dolaysız derecelendirme olmadığı zaman, kullanıcı hareketine göre hesaplanmış *u* kullanıcısının *i* ürünü hakkındaki fikrini gösteren ölçüm $r(u,i)$'dir (3.3). Dolaylı geri bildirimleri kullanarak yapılacak öneri için Hu ve diğ. (2008) bu değerlerden tercih $p(u,i)$ ve güven $c(u,i)$ değerlerinin şu şekilde hesaplanmasını önermiştir:

$$p(u,i) = \begin{cases} 1, r(u,i) > 0 \\ 0, r(u,i) = 0 \end{cases} \qquad c(u,i) = 1 + \alpha r(u,i) \tag{4.18}$$

Eğer kullanıcı *u* içerik *i*'yi tükettiyse ($r(u,i) > 0$ ise) $p(u,i) = 1$ olur ve kullanıcı *u*'nun içerik *i*'yi sevdiği, aksi durumda sevmediği ($p(u,i) = 0$) çıkarımında bulunuruz. Kullanıcının içeriği tüketmesinin, gerçekten sevmesi, başkası izlediği için izlemesi, TV'nin açık kalması gibi sebepleri olabilir. Benzer olarak izlememe sebebi de, sevmemesi veya haberi olmaması olabilir. $c(u,i)$ değişkeni, $p(u,i)$ hakkındaki güveni göstermektedir. Eğer kullanıcı bir ürünü defalarca seyretmiş ise güven yüksek, aksi halde düşük olacaktır.

Kullanıcı ve ürün faktörleri öğrenilirken dolaylı geri bildirimli yöntemde şu hata fonksiyonunun en düşük değerinin Bai ve diğ. (2011) bulunması gerekir:

$$\begin{aligned} E_{imp} &= \min_{x^* y^*} \sum_{u,i} c(u,i)(p(u,i) - b_u - d_i - \mathbf{x}_u^T \mathbf{y}_i)^2 \\ &+ \lambda(\sum_u \|\mathbf{x}_u\|^2 + \sum_i \|\mathbf{y}_i\|^2 + b_u^2 + d_i^2) \end{aligned} \tag{4.19}$$

(4.19)'de en iyileme sadece *K* kümesindeki değil bütün *(u,i)* ikilileri üzerinden yapıldığı için almaşık en küçük kareler yöntemi kullanılmaktadır. (4.19)'ye dikkatli bakıldığında, dolaylı geri bildirimde en iyilenen fonksiyonun şu olduğu görülür:

$$
\begin{aligned}
E_{imp} &= \min_{x^* y^*} \sum_{u,i} (1+\alpha r(u,i))(H(r(u,i)) - b_u - d_i - \mathbf{x}_u^T \mathbf{y}_i)^2 \\
&+ \lambda(\sum_u \|\mathbf{x}_u\|^2 + \sum_i \|\mathbf{y}_i\|^2 + b_u^2 + d_i^2) \\
&= \min_{x^* y^*} \sum_{u,i} (H(r(u,i)) - (b_u + d_i + \mathbf{x}_u^T \mathbf{y}_i))^2 + \alpha r(u,i)(1-(b_u + d_i + \mathbf{x}_u^T \mathbf{y}_i))^2 \\
&+ \lambda(\sum_u \|\mathbf{x}_u\|^2 + \sum_i \|\mathbf{y}_i\|^2 + b_u^2 + d_i^2)
\end{aligned}
\qquad \textbf{(4.20)}
$$

(4.20)'de H(x), x>0 ise 1, değilse 0 değeri alan step fonksiyonudur. Böylece bir *(u,i)* ikilisi için veri varsa faktörler çarpımının 1'e, yoksa 0'a yakınsaması hedeflenmekte; 1'e yakınsamaya verilen önem de kullanıcının ürün beğenisi ile orantılı olmaktadır.

$$\mathbf{x}_u = (\lambda II_d + Y^T C^u Y)^{-1} Y^T C^u p(u) \qquad \textbf{(4.21)}$$

$$\mathbf{y}_i = (\lambda UI_d + X^T C^i X)^{-1} X^T C^i p(i) \qquad \textbf{(4.22)}$$

Sırası ile kullanıcı faktörleri ve ürün faktörlerini en iyileyen değerler iteratif bir şekilde (4.21) ve (4.22)'teki gibi bulunmaktadır. C^u *mxm* ve C^i *nxn* boyutlarında, kullanıcı ve ürün için güven değerlerinin tutulduğu köşegen matrisler, Y bütün ürün faktörlerinin tutulduğu *nxf*'lik, X ise bütün kullanıcı faktörlerinin tutulduğu *mxf*'lik birer matristir. I film sayısını, U kullanıcı sayısını göstermektedir. I_d birim matristir.

$$b_u = \frac{e^T C_u (p_u - d - Y\mathbf{x}_u)}{e^T C_u e + \lambda I} \qquad \textbf{(4.23)}$$

$$d_i = \frac{e^T C^i (p^i - b - X\mathbf{y}_i)}{e^T C^i e + \lambda U} \qquad \textbf{(4.24)}$$

Beraber öneri sistemlerinde bazı kullanıcılar diğer kullanıcılardan daha yüksek oy verme şeklinde sistematik bir eğilim göstermektedirler (Koren ve diğ, 2009). Benzer şekilde bazı ürünler de diğerlerinden daha fazla tercih ediliyor olabilirler. Beraber öneri sistemlerinde ortak beğenilerin ürünler için puan üretmede önemli bir rol oynadığı göz önüne alındığında sistemli bir şekilde yüksek puan veren kullanıcıların ve yüksek puan alan ürünlerin model eğitimini yanlış yönlendirmesi söz konusu olabilmektedir. Bu durumu engellyebilmek için eşitlik (4.23)'te gösterilen b_u kullanıcı u için eğilim (bias), eşitlik (4.24)'te verilen d_i i filmi için eğilim formülleri kullanılmıştır (Bai ve diğ, 2011).

4.3 Hibrit Öneri Sistemi

Hibrit öneri üretme yollarından birisi içerik ve beraber öneri puanlarının ağırlıklandırılarak toplanmasıdır (Claypool ve diğ, 1999). Bu tezde içerik tabanlı puana, iki değişik türde elde edilen beraber öneri puanı eklenerek hibrit öneriler üretilmiştir.

4.3.1 Ortak içerik izlemeli hibrit öneri sistemi

Hibrit öneri sistemi sayesinde kullanıcıların kendi beğenilerinin yanısıra, aynı filmleri izleyen diğer kullanıcıların izleme davranışlarından da yararlanılarak öneri yapılmaktadır. Böylece iki kullanıcının ortak izledikleri film sayısı ne kadar çok ise birbirlerine o kadar benzerlerdir düşüncesinden yola çıkılmıştır. Bunun yanısıra öneri yapılacak kullanıcıyla aynı içeriği izlemiş iki kullanıcının izledikleri ortak filmlere verdikleri puanlar da göz önünde bulundurulmuştur. Böylece ortak film beğenisine sahip olmalarının yanısıra bu beğeninin miktarı da hesaba katılmış olmaktadır.

Elimizde u kullanıcısına i filminin önerilip önerilmeyeceği problemi olsun. Hibrit öneri sistemde öncelikle bu içeriği eğitim kümesinde izlemiş kullanıcılar bulunur. Bu kullanıcılardan u kullanıcısıyla ortak film izlemiş olan kullanıcılar seçilir ve bu kullanıcıların i filmine verdikleri puan alınır. Böylece u kullanıcısı ile ortak beğeniye sahip kullanıcının izlemelerinden yararlanılarak u kullanıcısına öneri yapılmış olur.

Hibrit öneri sistemiyle i filmini önerirken eğer u kullanıcısı eğitim kümesinde i filminin özniteliklerinden hiçbirini bulunduran bir film izlemediyse içerik tabanlı sistem i filmini u kullanıcısına öneremektedir. Ancak i filmini i ile ortak beğeniye

sahip kullanıcılardan biri izlediyse sistem artık *i* filmini *u* kullanıcısına önerebilir hale gelmektedir.

i içeriği test kümesinde olan bir film olsun. Bu filmi eğitim kümesinde izlemiş kullanıcılar kümesi U_i olsun. U_i kümesi içindeki her *v* kullanıcısının *i*'ye verdiği puan $r(v,i)$ olsun. *v* kullanıcısı ile *u* kullanıcısının eğitim kümesinde ortak izledikleri film sayısı $c(u,v)$ olsun. Bu durumda, *u* kullanıcısının *i* içeriğine vereceği beraber öneri (collaborative) puanı şöyle hesaplayabiliriz:

$$\mathrm{r_c}(u,i) = \sum_{v \neq u, v \in Ui} \frac{c(u,v)}{C} r(v,i) \qquad \textbf{(4.25)}$$

Burada normalizasyon katsayısı şu şekilde hesaplanmaktadır:

$$C = \sum_{v \neq u, v \in Ui} c(u,v) \qquad \textbf{(4.26)}$$

Beraber öneriden gelen puan ile *k* özniteliğine göre içerik tabanlı öneriden gelen puanlar birleştirildiğinde *k* özniteliğine göre hibrit puan şu şekilde üretilir:

$$r_{k,c}(u,i) = \alpha r_c(u,i) + (1-\alpha) r_k(u,i) \qquad \textbf{(4.27)}$$

Hibrit öneri formülünde, beraber öneriye verilen ağırlık α ile gösterilmiştir. Deneylerde α'nın 0.0,0.25,0.5,0.75 ve 1.0 değerleri için sonuç alınmış olup, hibrit sistem en iyi performansını α = 0.25 iken sergilemiştir. α = 1 değeri için bu öneri yöntemi beraber öneri yöntemi haline gelmektedir. Fakat bir filmi izleyen ortak kullanıcıların az olduğu durumlarda bu yönteme göre üretilen puanlar ile öneri yapıldığında iyi sonuçlar alınmamaktadır. Matris ayrıştırması bu ortak izleyici olmaması problemi için bir çözümdür.

4.3.2 Matris ayrıştırma kullanılarak hibrit öneri sistemi geliştirilmesi

Matris ayrıştırma ile elde edilen puanların eşitlik formülünün beraber öneri kısmında kullanılmasıyla yeni bir hibrit formül elde edilmiştir.

$$r_{k,mfc}(u,i) = \alpha r_{mfc}(u,i) + (1-\alpha) r_k(u,i) \qquad \textbf{(4.28)}$$

(4.28)'de görülen $r_{mfc}(u,i)$ matris ayrıştırma (matrix factorization) metodunu kullanan beraber öneri sisteminde üretilen puanı ifade eder.

(4.27) ve (4.28) kullanılarak alınan sonuçlar Deneyler bölümünde yer almaktadır.

5. PERFORMANS ÖLÇÜTLERİ ve DENEYLER

Bu bölümde sistemin değerlendirilmesi için kullanılan performans ölçütleri tanıtılmıştır. İçerik tabanlı öneri sisteminin, beraber öneri sisteminin ve hibrit öneri sisteminin performans değerlendirilmesi yapılmıştır.

5.1 Performans Ölçütleri

Sistemde kullanıcılara test kümesindeki içeriklerden öneri yapılmaktadır. Test kümesindeki filmler için üretilen puanlar büyükten küçüğe doğru sıralanmakta ve ilk 10 (Φ) tane film kullanıcıya önerilmektedir. Bu önerilen filmlerden kullanıcının izlediklerinin sayısı doğru öneri sayısı "*bilinen film miktarı"* ile ifade edilmiştir. Ayrıca ilk 10 öneride kullanıcının önerilen filmlerden kaç tanesini, ne kadar puan vererek izlediği de dikkate alınan diğer bir faktördür.

Sistemin performansını ölçmek için 4 adet performans ölçütü kullanılmıştır. Bunlardan kesinlik (precision) ve anma (recall) literatürde kullanılan ölçütlerdir. Diğer ölçütler ise kesinliğin normalleştirilmesiyle elde edilen normalleştirilmiş kesinlik (normalized precision) ve kullanıcının içeriğe verdiği puanlar kullanılarak üretilen puan ile ağırlıklandırılmış normalleştirilmiş kesinlik (RWNP: Rating Weighted Normalized Precision) ölçütleridir.

$$\text{Kesinlik} = \frac{\text{bilinen film miktarı}}{\Phi} \qquad \textbf{(5.1)}$$

Kesinlik ölçütü ile kullanıcının testte izlediği içeriklerden kaç tanesinin ilk 10 öneride doğru tahmin edildiği hesaplanmaktadır (5.1) $\Phi = 10$ olarak seçilmiştir.

$$\text{Anma} = \frac{\text{bilinen film miktarı}}{I_u^{test}} \qquad \textbf{(5.2)}$$

I_u^{test}, u kullanıcısının testte izlediği film sayısıdır. (5.2)'de görüldüğü üzere anma kullanıcının test zamanında izlediği film sayısını da baz alan bir ölçüttür. İlk 10'da

yapılan doğru öneri miktarının kullanıcının testte izlediği film sayısına oranı olarak belirtilmiştir.

Sistem genel olarak kullanıcının izlediği film sayısı arttıkça çok miktarda doğru öneri yapma eğilimindedir. Ancak izlenen film sayısı az iken de çok sayıda doğru öneri yapılabilen kullanıcılar mevcuttur.

Normalleştirilmiş kesinlik ölçütü test kümesindeki toplam film sayısını hesaba katan bir ölçüttür:

$$\text{Kesinlik}^{\text{N}} = \frac{\text{bilinen film miktarı}}{\Phi} \Bigg/ \frac{I_u^{test}}{I^{watch}} \qquad \textbf{(5.3)}$$

I^{watch}, test kümesindeki toplam film sayısıdır. Kullanıcının testte izlediği içerik sayısının test kümesindeki toplam içerik sayısına oranı üretilecek doğru önerileri etkileyen bir faktördür ve kesinlik değerinin bu oran ile normalleştirilmesiyle Normalleştirilmiş kesinlik (5.3) elde edilmiştir. (5.3)'ün paydasındaki bu oran, kullanıcıya rastgele olarak öneri yapılsaydı ne kadar doğrulukta tahmin yapılabileceğini belirler.

İki kullanıcı seçilsin ve sistemin kesinlik ölçütü bu iki kullanıcı için aynı olsun. Sistemin bu kullanıcılardan testte izlediği film sayısının testteki toplam film sayısına oranı daha küçük olan kullanıcı için yaptığı önerinin başarısı daha yüksektir. Çünkü birçok film içinde oldukça az film izleyen bu kullanıcının izlediklerini doğru tahmin etmek zor bir problemdir. Bu nedenle $\text{Kesinlik}^{\text{N}}$ değerinin yüksek olması o kullanıcı için yapılan önerinin de başarılı olduğunu gösterir. Önerilen filmlerin izlenmiş olması yanında kullanıcı tarafından verilen gerçek puanlarının yüksek olması da önemlidir. RWNP ölçütü bu durumu değerlendirmektedir:

$$\text{RWNP} = \frac{\sum_{i \in H(u)} r(u,i)}{\Phi} \Bigg/ \frac{I_u^{test}}{I^{watch}} \qquad \textbf{(5.4)}$$

H(u), ilk 10 da *u* kullanıcısına önerilen ve doğru tahmin edilen filmlerin kümesidir. Bu filmlere kullanıcının verdiği puanlar toplanarak RWNP ölçütü hazırlanmıştır (5.4). Eğer kullanıcının yüksek puan verdiği filmler ilk 10 da önerilen filmlerin arasına girdiyse RWNP değeri yüksek çıkacaktır. Bu ölçüt, ilk 10 da kaç tane doğru

tahmin yapıldığı bilgisinin yanısıra yüksek puan verilen filmler arasından öneri yapılıp yapılmadığını da göstermektedir.

5.2 Deney Sonuçları

5.2.1 İçerik tabanlı öneri sistemi için deney sonuçları

Filmlerin özniteliklerine göre hesaplanan puanlar büyükten küçüğe doğru sıralandıktan sonra, en büyük puanı alan film kullanıcıya önerilir.

Kullanıcılara önerilen filmlerin, doğru önerilip önerilmediğini değerlendiren ölçütlerin bulunduğu tablolar hazırlanmıştır. Çizelge 5.1'de ilk kolonda kullanıcıları tanımladığımız numaralar listelenmektedir. I_u^{test} u kullanıcısının test kümesinde izlediği film sayısıdır. I^{watch} ise test kümesindeki filmlerin toplam sayısıdır. Öneriler test kümesindeki filmler üzerinden yapılmaktadır. En son kolonda doğru tahmin edilen filmlere kullanıcının verdiği puanların toplamı görülmektedir.

Çizelge 5.1 : Kullanıcı bazında önerilerin doğruluk ölçümleri (tür).

Kullanıcı no	Kesinlik	Anma	Kesinlik^N	RWNP	I_u^{test}/I^{watch}	Toplam puan
23	0.9	0.18	18.13	4.59	0.05	2.28
45	0.9	0.07	7.31	1.19	0.12	1.46
128	0.9	0.5	10.80	2.17	0.08	1.81

Çizelge 5.1'de ilk 10 öneride 9 tane doğru tahmin yapılan üç kullanıcı için doğruluk ölçümleri görülmektedir. Kullanıcıların herbiri için Kesinlik değerleri aynıdır.

23 numaralı kullanıcıya bakıldığında, Kesinlik^N değeri rastgele olarak yapılacak öneriden 18 kat daha iyi bir öneri yapıldığını gösterir.

23 numaralı kullanıcı diğer kullanıcılarla RWNP ölçütüne göre karşılaştırıldığında; test/watch oranı düşük olan 23 kullanıcısı için yapılan öneri daha başarılıdır. Test kümesinde birçok içerik varken 23 kullanıcısının gayet az bir izleme yapması sistemi zorlayan bir durumdur. Çünkü sistemin bu kullanıcıya önerebileceği birçok film vardır, bu filmlerden kullanıcının izlemiş olduklarını tutturma ihtimali kullanıcının az izleme yapmasıyla düşmesi beklenen bir durumdur. Ancak sistem bu zorluğu aşmış ve bu kullanıcı için 10'da 9 doğru tahmin yapabilmiştir.

Anma değerlerine bakıldığında 128 numaralı kullanıcı için de başarılı bir öneri yapıldığı söylenebilir.

Eğer RWNP değeri yüksek ise, sistem kullanıcının yüksek puanlama verdiği içeriklerden önermiştir yorumu yapılabilir.

5.2.2 Doğru öneri sayısı ve puan sıralamasının incelenmesi

Oyuncu, yönetmen, tür, kanal, yıl, anahtar kelime öznitelik kümelerinin ayrı ayrı kullanarak kullanıcıların verdikleri puanlarla sistemin ürettiği puanlar karşılaştırıldığında, ilk 10 da doğru tahmin edilen içeriklerin kullanıcılar tarafından da öncelikle izlenmiş oldukları görüldü. Aşağıdaki tablolarda oyuncu, yönetmen ve türe göre hesaplanan puanların performansları görülmektedir.

Çizelge 5.2'de tür özniteliğine göre puan üretildiğinde Kesinlik^N değerleri eşit çıkan üç kullanıcı için yapılan önerilerin performans değerleri görülmektedir. Bu kullanıcılar için sistem sırasıyla 2, 4 ve 3 adet doğru tahmin yapmıştır.

Çizelge 5.2 : Tür özniteliğine göre performans ölçümleri.

Kullanıcı no	Kesinlik	Anma	Kesinlik^N	RWNP	I_u^{test} / I^{watch}	Toplam puan
1	0.2	0.2	20.46	2.102	0.0097	0.206
354	0.4	0.2	20.46	6.119	0.0195	1.196
89	0.3	0.2	20.46	4.485	0.0146	0.658

Doğru tahmin edilen filmler için kullanıcıların verdikleri puanlar ve sistemin ürettiği puanların sıralamaları Çizelge 5.3' te gösterilmiştir. Hem kullanıcının filmleri izleme sürelerinden elde edilen puanlar hem de sistemin ürettiği puanlar min-max normalleştirme işlemine tabi tutulmaktadır. Çizelge 5.3'te kullanıcıların izleme sürelerinden oluşan puanlar ve sistemin ürettiği puanların sıralamaları gösterilmiştir. Kullanıcının kendi puan skalasına göre üst sıralarda puan verdiği filmler, sistem tarafından da bulunmuş ve üst sıralarda önerilebilmiştir.

Çizelge 5.3 : Kullanıcı puan-sistem puan karşılaştırması (tür).

Kullanıcı no	İçerik no	Kullanıcı puan	Sistem puan	Kullanıcı puan sıra	Sistem puan sıra
1	371	0.13	0.75	3	4
1	365	0.08	0.78	5	5
354	330	0.37	1	7	1
354	370	0.12	1	12	1
354	718	0.23	0.73	10	6
354	210	0.47	0.65	3	10
89	350	0.32	0.92	2	3
89	378	0.22	0.86	3	4
89	334	0.12	0.8	6	6

Çizelge 5.4'te yönetmen özniteliğine göre yapılan önerilerin sonuçları görülmektedir. En yüksek Kesinlik^N değerine sahip 90 no'lu kullanıcı için yapılan öneri rastgele yapılacak bir önerinin 28.771 katıdır.

Çizelge 5.4 : Yönetmen özniteliğine göre performans ölçümleri.

Kullanıcı no	Kesinlik	Anma	Kesinlik^N	RWNP	I_u^{test} / I^{watch}	Toplam puan
90	0.4	0.29	28.771	20.738	0.014	2.88
112	0.3	0.27	27.464	22.969	0.011	2.51
400	0.1	0.25	25.575	12.75	0.004	0.5

Çizelge 5.5'te yönetmen özniteliği kullanılarak alınan performans sonuçlarının bulunduğu tablodaki kullanıcıların detaylı analizleri bulunmaktadır. Tür özniteliği ile karşılaştırıldığında daha üst sıralarda izlenmiş filmlerin önerildiği görülmektedir.

Yine yönetmen özniteliği kullanılarak; ancak bu sefer Kesinlik^N değerleri daha düşük olan diğer üç kullanıcı için performans sonuçları Çizelge 5.6'da gösterilmektedir.

Çizelge 5.5 : Kullanıcı puan-sistem puan karşılaştırması (yönetmen).

Kullanıcı no	İçerik no	Kullanıcı puan	Sistem puan	Kullanıcı puan sıra	Sistem puan sıra
90	352	1	0.95	1	2
90	354	0.68	0.64	2	4
90	3443	0.67	0.65	3	5
90	654	0.54	0.51	4	8
112	3541	0.92	1	1	1
112	558	0.83	0.78	2	5
112	368	0.77	0.72	3	6
4	3637	0.5	0.5	1	4

Kesinlik^N değerinin azalmasıyla, RWNP değerinin de düştüğü Çizelge 5.6'da görülmektedir. Dolayısıyla Kesinlik^N ve RWNP değeri yüksek öneriler yapmak sistemin ne kadar iyi çalıştığının göstergelerinden biri olmaktadır.

Çizelge 5.6 : 90, 112 ve 4 no'lu kullanıcılar için yönetmen özniteliğine göre performans ölçümleri.

Kullanıcı no	Kesinlik	Anma	Kesinlik^N	RWNP	I_u^{test} / I^{watch}	Toplam puan
194	0.5	0.1	10.07	4.827	0.049	2.4
44	0.1	0.1	10.07	8.133	0.009	0.81
250	0.4	0.1	10.07	2.165	0.04	0.86

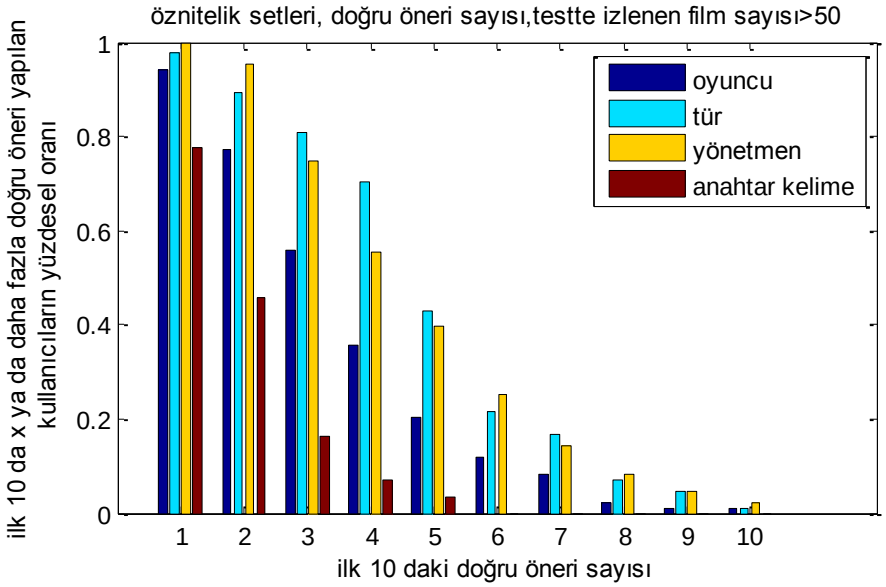
Tür ve yönetmen için yapılan detaylı analizler oyuncu ve anahtar kelime öznitelik kümeleri için de yapılmıştır. Çizelge 5.7'de tür, oyuncu, yönetmen, anahtar kelime öznitelik kümelerine göre yapılan önerilerin tüm kullanıcılar üzerinden ortalama kalite ölçümleri görülmektedir.

Çizelge 5.7 : Öznitelik kümeleri için performans ölçüm sonuçları.

Öznitelik Kümesi	Kesinlik	Anma	Kesinlik^N	RWNP
tür	0.195	0.073	6.076	1.633
oyuncu	0.177	0.076	6.405	3.42
yönetmen	**0.215**	**0.085**	**7.231**	**3.79**
anahtar kelime	0.065	0.032	2.473	1.214

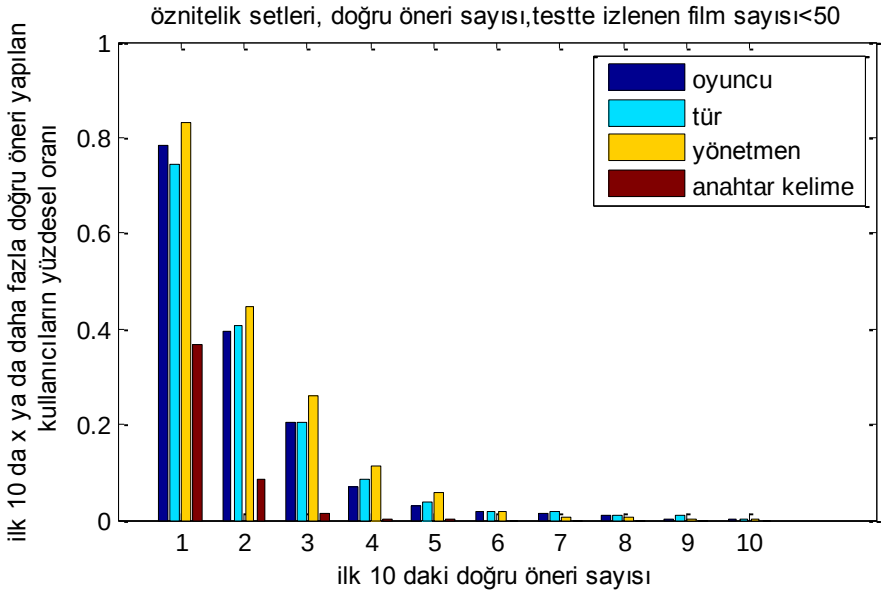
En başarılı sonuçlar yönetmen özniteliği kullanılarak yapılan önerilerde alınmıştır. Yönetmen özniteliği ile üretilen puanlarla yapılan öneriler rastgele yapılacak bir öneriden 7.2 kat daha doğru olmaktadır. Ayrıca ilk 10 öneride ortalama 2 doğru tahmin yapılabilmektedir. Yine kullanıcının izlediği filmlere verdiği puanları da performans ölçümünde kullanan RWNP ölçütü en yüksek değerini yönetmen özniteliğinde almıştır. Bu durum kullanıcıların büyük bir kısmının film tercihi yaparken yönetmen kriterini önemsediklerini göstermektedir.

Şekil 5.1'de ilk 10 öneride x ya da daha fazla doğru öneri yapılan kullanıcıların yüzdesel oranları görülmektedir. Bu kullanıcılar testte 50'den fazla içerik izlemiş kullanıcılardır. Genel olarak yönetmen veya türe göre yapılan önerilerde daha başarılı sonuçlar elde edilmiştir. Tür özniteliği kullanılarak yapılan önerilerde kullanıcıların %43'ü için 5 ya da daha fazla doğru film önerisi yapılabilmektdir. Yine yönetmen özniteliği kullanıldığında kullanıcıların %40'ı için 5 ya da daha fazla doğru film önerisi yapılabilmektdir. Tür veya yönetmen özniteliği kullanılarak öneri yapıldığında kullanıcıların yaklaşık %10'u için 9 ya da daha fazla doğru film önerisi yapılabilmektedir.



Şekil 5.1 : Testte 50'den fazla film izlemiş kullanıcılara farklı özniteliklerine göre yapılan önerilerin performansı.

Şekil 5.2'de test kümesindeki içeriklerden 50'den az film izlemiş kullanıcılara önerilen doğru film miktarlarının kullanıcıların yüzdesel miktarıyla karşılaştırılması görülmektedir.

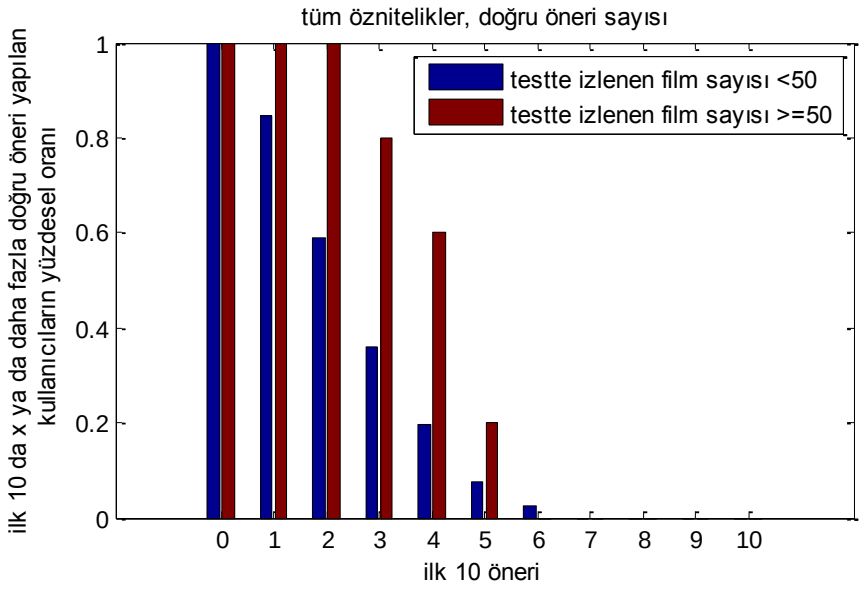


Şekil 5.2 : Testte 50'den az film izlemiş kullanıcılara farklı özniteliklerine göre yapılan önerilerin performansı.

Grafikten görüldüğü üzere, test kümesindeki içeriklerden 50'den az film izlemiş kullanıcılar için bile hiç doğru öneri yapılamayan kullanıcı oranı, tüm öznitelikler için yaklaşık %20'ler civarında kalmaktadır.

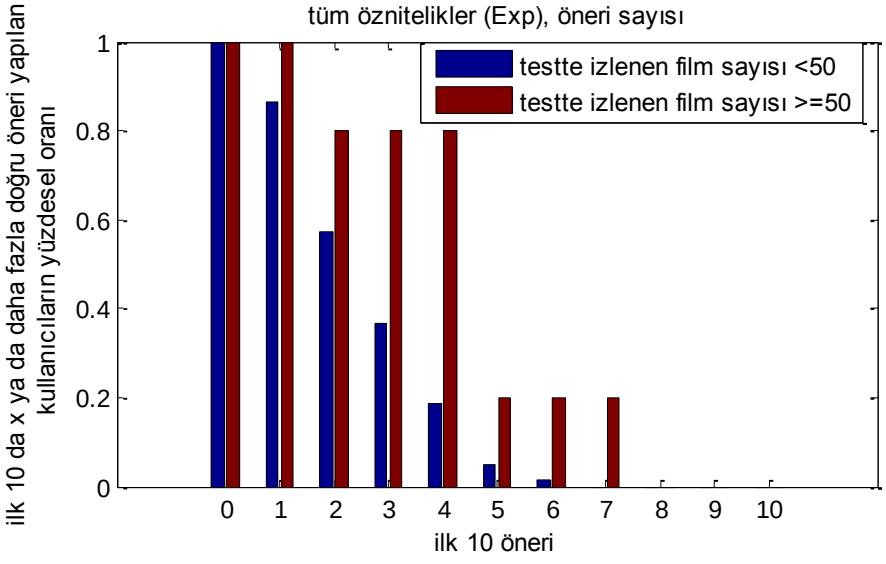
5.2.3 Öznitelik kümelerinin birleştirilmesiyle elde edilen sonuçlar

Bölüm 4.1.3'te anlatılan özniteliklerin (4.5)'e göre birleştirilmesiyle elde edilen sonuçlar Şekil 5.3'te görülmektedir. Şekil 5.3'te görüldüğü üzere tüm özniteliklerin birleştirilmesiyle kullanıcılara ilk 10 öneride 7, 8, 9, 10 adet doğru film önerisi yapılamamaktadır. Ancak ilk 10'da 4 veya daha fazla doğru öneri yapılan kullanıcıların oranı, 50'den az film izlemiş kullanıcılar için, sadece yönetmen, tür, oyuncu veya anahtar kelimeye göre doğru öneri yapılan kullanıcıların oranından daha iyidir. Özniteliklerin birleştirilmesiyle ilk 10'da 6'dan çok doğru tahmin edilen az sayıdaki filmler birleştirme işleminde yapılan toplamanın etkisiyle artık doğru tahminlerin arasında bulunmamaktadırlar.



Şekil 5.3 : Testte 50'den az ve 50'den fazla film izlemiş kullanıcılara tüm özniteliklerin birleştirilmesiyle yapılan önerilerin performansı.

Şekil 5.4'te Bölüm 4.1.3'te bahsedilen öznitelik birleştirmesinin eşitlik (4.6)'ya göre eğitim kümesindeki ortalama mutlak değer hatanın üssel değerinin alınmasıyla üretilen puanlara göre öneri yapılmasıyla elde edilen sonuçlar görülmektedir. İlk 10'da 7 doğru öneri yapılabilen kullanıcılar Şekil 5.3'te bulunmazken, öznitelikler üssel yönteme birleştirildiğinde 50'den fazla izleme yapmış kullanıcıların %20 sine 7 veya daha fazla öneri yapılabilmektedir.



Şekil 5.4 : Testte 50'den az ve 50'den fazla film izlemiş kullanıcılara tüm özniteliklerin üssel yönteme göre birleştirilmesiyle yapılan önerilerin performansı.

5.2.4 En düşük MAE'ye sahip öznitelik kümesi ile öneri üretilmesi

Eğitim kümesinde en küçük ortalama mutlak hata değeri elde edilen özniteliğe göre kullanıcılara öneri yapmak için, kullanıcıların filmlere verdiği puan ile üretilen puan arasındaki farka bakılır. Bu fark hangi öznitelik kümesinde daha düşük ise kullanıcı için o öznitelik kümesi öneri yapılırken kullanılmaktadır.

Çizelge 5.8 : Her bir öznitelik kümesinden elde edilen MAE ortalaması.

oyuncu_mae	yönetmen_mae	tür_mae	anahtar kelime_mae
0.115	0.081	0.262	0.187

Çizelge 5.8'de öznitelik kümelerinde tüm kullanıcıların ortalama MAE değerleri analizi görülmektedir. En yüksek hata oranı tür öznitelik kümesinde alınmaktadır. En düşük hata ise yönetmen özniteliğinde elde edilmiştir.

Çizelge 5.9'a, daha önce Çizelge 5.7'de gösterilen performans değerlerine her bir kullanıcı için hangi öznitelik kümesinde en küçük MAE değeri elde ediliyorsa ona göre öneri yapıldığında elde edilen performans sonuçları eklenmiştir. Üzerinde çalıştığımız 444 kullanıcı verisinin 379 tanesi için yönetmen, 48 tanesi için oyuncu, 16 tanesi için de anahtar kelime özniteliğinde en küçük MAE elde edilmiştir. Bu kullanıcılardan alınan ortalama değerler Çizelge 5.9'da min_train_MAE_oyuncu, min_train_MAE_yönetmen, min_train_MAE_anahtarkelime ile gösterilmiştir.

Çizelge 5.9 : Öznitelik kümeleri için performans ölçüm sonuçları (MAE'li sonuçlarla birlikte).

Öznitelik Kümesi	Kesinlik	Anma	Kesinlik^N	RWNP
tür	0.195	0.073	6.076	1.633
oyuncu	0.177	0.076	6.405	3.42
yönetmen	**0.215**	**0.085**	**7.231**	**3.79**
anahtar kelime	0.065	0.032	2.473	1.214
min_train_MAE_oyuncu	0.196	0.061	5.007	2.57
min_train_MAE_yönetmen	0.203	0.081	7.148	3.779
min_train_MAE_anahtarkelime	0.163	0.09	3.955	1.805

Performans ölçütlerinden görüldüğü üzere normal ve MAE'ye göre yapılan önerilerde yönetmen özniteliğine göre öneri yapmak rastgele yapılacak bir öneriden

7 kat daha doğru olmaktadır. Yine yüksek puan verilmiş filmlerin tahmin edilmesini ifade eden RWNP değeri en yüksek değerini yönetmen özniteliğinde almaktadır.

5.2.5 Matris ayrıştırmalı beraber öneri sistemi ile alınan deney sonuçları

Matris ayrıştırma yöntemine göre alınan sonuçlar Çizelge 5.10 da gösterilmiştir. Bu çizelgede Kesinlik^N değerinin 8.093 olması kullanıcılara rastgele yapılacak bir öneriden 8 kat daha iyi öneriler yapılabileceğini göstermektedir.

Çizelge 5.10 : MF yöntemi performans ölçüm sonuçları.

Kesinlik	Anma	Kesinlik^N	RWNP
0.123	0.112	8.093	1.874

5.2.6 Matris ayrıştırmalı hibrit öneri sistemi ile alınan deney sonuçları

Matris ayrıştırma yöntemi kullanılarak oluşturulan beraber öneri sistemi ile içerik tabanlı sistemin birarada kullanılmasıyla elde edilen hibrit yönteme göre alınan sonuçlar Çizelge 5.11' da görülmektedir.

Çizelge 5.11 : HibritMF yöntemi performans ölçüm sonuçları.

Öznitelik Kümesi	Kesinlik	Anma	Kesinlik^N	RWNP
tür	0.084	0.127	6.968	0.817
oyuncu	0.16	0.148	**9.981**	3.228
yönetmen	0.147	0.121	8.918	2.953
anahtar kelime	**0.161**	**0.147**	9.978	**3.29**

İçerik tabanlı sistemde alınan kesinlik değerlerine göre HibritMF yönteminin kesinlik değerinde düşme görülürken, Kesinlik^N ve RWNP değerlerinde yükselme görülmüştür.

5.2.7 Ortak içerik izlemeli hibrit öneri sistemi ile alınan deney sonuçları

Bölüm 4.3.1'de anlatılan ortak içerik izleyen kullanıcıların izleme bilgileri ve içerik tabanlı öneri sistemini baz alan yöntemde alınan sonuçlar diğer yöntemlerden daha yüksek çıkmıştır.

Çizelge 5.12 : HibritOrtakFilm yöntemi performans ölçüm sonuçları.

Öznitelik Kümesi	Kesinlik	Anma	Kesinlik^N	RWNP
tür	0.555	0.164	15.125	5.669
oyuncu	0.785	0.207	19.323	7.099
yönetmen	0.793	0.209	19.442	7.112
anahtar kelime	**0.875**	**0.269**	**25.828**	**7.213**

Çizelge 5.12'e bakıldığında kullanıcının test kümesindeki izlediği içerik sayısının test kümesindeki içeriklerin toplamına oranlanmasıyla yapılabilecek olasılıksal öneriden tür özniteliğine göre öneri yapıldığında 15 kat, oyuncu ve yönetmene göre öneri yapıldığında 19 kat anahtar kelimeye göre öneri yapıldığında ise yaklaşık 26 kat daha doğru öneriler yapılabilmektedir.

6. SONUÇLARIN DEĞERLENDİRİLMESİ

Bu tez çalışmasında çeşitli yöntemler kullanılarak kullanıcılara film önerileri üreten bir sistem geliştirilmiştir. Bu sistemde yöntem olarak içerik tabanlı öneri sistemi, beraber öneri sistemi bu iki sistemin birleşiminden oluşan iki hibrit öneri sistemi geliştirilmiştir.

İçerik tabanlı öneri sisteminde farklı öznitelik kümelerinin öneri kalitesindeki etkisi incelenmiştir. İçerik tabanlı yöntemde eğitim kümesinde kullanıcıların filmlere verdikleri puanlar özniteliklerin ağırlıklarının belirlenmesinde kullanılmıştır. Böylece hem filmlerin öznitelik bilgileri hem de kullanıcıların filmlere verdikleri puanlara göre öneriler üretilebilmiştir. Her bir öznitelik kümesi için kullanıcılara ayrı ayrı öneriler üretilmiştir. En başarılı sonuçlar yönetmen özniteliğinde alınmıştır.

Matris ayrıştırma kullanılarak yapılan beraber öneri sisteminde kullanıcıların filmlere dolaylı olarak verdikleri puanlar kullanılmıştır. Mevcutta kullanıcıların doğrudan beğenilerini ifade ettikleri bir skor bulunmamakta idi. Bu nedenle kullanıcı beğenisini saptayabilmek için filmlerin izleme sürelerini kullanarak dolaylı puan üretilmiştir. Matris ayrıştırma kullanan beraber öneri sisteminde alınan sonuçlara bakıldığında kesinlik değeri içerik tabanlı sistemden alınan sonuçlardan daha iyi olmasına rağmen, ortalama kesinlik miktarı içerik tabanlı öneri sisteminde daha yüksek değerler alabilmiştir.

Bu çalışmada iki adet performans ölçütü sunulmuştur ve yöntemlerin performans ölçümlerinde kullanılmıştır.

Sistemde iki adet hibrit öneri sistemi geliştirilmiştir. İki yöntem de P-Tango sisteminde Claypool ve diğ. (1999) olduğu gibi doğrusal yöntemlerdir. Bu sistemlerden ilkinde aynı eğitim kümesinde ortak içerik izlemiş kullanıcıların izleme davranışlarından yararlanılarak bir beraber öneri puanı üretme yöntemi sunulmuştur. Bu yöntem ile elde edilen puanların içerik tabanlı öneriden gelen puanlar ile birleştirilmesiyle oluşturulan hibrit ortak film yöntemi doğrusal bir yöntemdir. Normalleştirilmiş kesinlik değerleri bu yöntem ile oldukça yüksek çıkmıştır. Ayrıca doğru önerilen film sayısında da diğer yöntemlere göre artışlar gözlenmektedir.

Diğer hibrit yöntem matris ayrıştırma yöntemini kullanan beraber öneri sistemiyle üretilen puanları, yine içerik tabanlı sistemden gelen puanlar ile doğrusal bir modele göre birleştirmektedir. Bu yöntem ile alınan normalleştirilmiş kesinlik sonuçlarının hibrit ortak filtreleme yönteminden daha iyi olmasa da, içerik tabanlı öneri sistemiyle elde edilen normalleştirilmiş kesinlik sonuçlarından daha iyi olduğu deney sonuçlarında görülmüştür.

Bundan sonra yapılabilecek çalışmalar arasında öncelikle anahtar kelime kullanılarak öneri yapıldığında öznitelik seçimi için tf-idf'den başka, örneğin FCBF# gibi öznitelik seçimi yöntemleri kullanılması sayılabilir (Senliol ve diğ, 2009). Öneri üretilmesi gereken filmi izleyen kullanıcılar ile kendisi için öneri üretilecek kullanıcı arasında ortak izlenmiş film olmaması durumunda Luo ve diğ. (2008) olduğu gibi küresel kullanıcı benzerlikleri hesaplanabilir.

KAYNAKLAR

Albright, R., Cox, J., Duling, D., Langville, A. N. and C. D. Meyer, 2006: "Algorithms, initializations, and convergence for the nonnegative matrix factorization," presented at the 12th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, Aug. 2006.

Bai, X., Wu, J., Wang, H., Zhang, J., Yin W., Dong J., 2011: Recommendation Algorithms for Implicit Information, Service Operations, Logistics, and Informatics (SOLI), 2011 IEEE International Conference.

Baltrunas, L., ve Ricci, F., 2008: Locally Adaptive Neighborhood Selection for Collaborative Filtering Recommendations, Adaptive Hypermedia and Adaptive Web-Bases Systems, 2008, LNCS 5149, sayfa 22-31.

Boutemedjet, S. Ziou, D., Bouguila, N., 2007: Unsupervised Feature Selection for Accurate Recommendation of High-Dimensional Image Data, NIPS (Neural Information Processing Systems) 2007.

Bucak, S.S., Günsel, B., 2009: Incremental subspace learning via non-negative matrix factorization, Pattern Recognition, Volume 42, Issue 5, Mayıs 2009, sayfa 788-797.

Burke R., 2002: Hybrid recommender systems: survey and experiments. User Model User Adapt Interact 12:331-370, 2002.

Çataltepe, Z. ve Altınel B., 2007: Hybrid Music Recommendation Based on Different Dimensions of Audio Content and Entropy Measure, Eusipco (European Signal Processing Conference) 2007, 3-7 Eylül, Polonya.

Çataltepe, Z. ve Altınel B., 2009: Music Recommendation by Modeling User's Preferred Perspectives of Content, Singer/Genre and Popularity, Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling, M. Chevalier, C. Julien ve C. Soulé-Dupuy tarafından derlendi, IGI Global, sayfa 203-221, ISBN: 978-1-60566-306-7.

Çataltepe Z., Uluyağmur M., Tayfur E., 2012: Adaptif Düzenlemeli Dolaylı Geri Bildirim Kullanılarak TV Programı Önerisi, SIU 2012, Antalya, Turkiye.

Chen, H-C. ve Chen, A.L.P., 2005: "A music recommendation system based on music and user grouping," Journal of Intelligent Information Systems, Volume 24, Numbers 2-3, 113-132, 2005.

Cheung, K.-W., Kwok, J.T., Law, M.H., Tsui, K.C., 2003:Mining customer product ratings for personalized marketing, Decision Support Systems 35 (2003) 231–243.

Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D. and Sartin, M., 1999: "Combining Content-Based and Collaborative Filters in an Online Newspaper" *SIGIR'99 Workshop on Recommender Systems: Algorithms and Evaluation.* Berkeley, CA.

Cremonesi, P., Koren, Y., Turrin, R., 2010: Performance of recommender algorithms on top-n recommendation tasks. In Proc. 4th ACM Conference on Recommender Systems(RecSys'10), pages 39-46, 2010.

Dash, M., Choi, K., Scheuermann, P., Liu, H., 2002: Feature Selection for Clustering - A Filter Solution, Second IEEE International Conference on Data Mining (ICDM'02), 2002, sayfa 115-126.

Debnath S.,Ganguly N.,Mitra P., 2008: Feature Weighting in Content Based Recommendation System Using Social Network Analysis, WWW 2008, April 21-25, 2008 Beijing, China.

de Campos, L., Fernandez-Luna, J., Huete, J., Rueda-Morales, M., 2010: Combining Content-Based and Collaborative Recommendations: A Hybrid Approach Based on Bayesian Networks. Approximate Reasoning 51(7) (2010) 785–799

Ding,C., Peng,H., 2003: Minimum Redundancy Feature Selection from Microarray Gene Expression Data, Proceedings of the Computational Systems Bioinformatics conference (CSB'03), sayfa 523–529 (2003).

Fernández, Y. B., Pazos Arias, J. J., Nores, M. L., Solla, A. G., Cabrer, M. R. ve diğerleri, 2006: "AVATAR: An Improved Solution for Personalized TV based on Semantic Inference", IEEE Trans. on Consumer Electronics. Vol. 52(1), February, 2006.

Goldberg, D. ve diğerleri, 1992: Using Collaborative Filtering to Weave an Information Tapestry, Comm. ACM, vol. 35, 1992, sayfa 61-70.

Goren-Bar, D., Glinansky, O., 2004**:** FIT-recommending TV programs to family members. Computers & Graphics 28, 149-156

Gülgezen, G., Çataltepe, Z., Yu, L., 2009: Stable and Accurate Feature Selection, ECML/PKDD 2009, Bled, Slovenya.

Hu, Y.F., Koren, Y., and Volinsky, C., 2008: "Collaborative Filtering for Implicit Feedback Datasets", Proc. IEEE Int'l Conf. Data Mining (ICDM 08), IEEE CS Press, 263-272, 2008.

Jones, K. S., 2004: "A statistical interpretation of term specificity and its application in retrieval," Journal of Documentation,vol. 60, no. 5, pp. 493–502, 2004

K. Ali and W. van Stam., 2004: TiVo: making show recommendations using distributed collaborative filtering architecture, Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 394–401.

K. Ikawa, T. Fukuhara, H. Fujii and H. Takeda, 2010: Evaluation of a TV Programs Recommendation using the EPG and Viewer's Log Data, in C. Peng, P. Vuorimaa, P. Naranen, C. Quico, G. Harboe and A. Lugmayr eds., Adjunct Proceedings EuroITV, pp. 182–185, Tampre, Finland, Tampere University of Technology.

Kim, M., Ko, S., Mun, J., Ji, Y., Jung M., 2007: A Usability Study on Personalized EPG (pEPG) UI of Digital TV. Department of In-formation and Industrial Engineering, Yonsei University.

Koren, Y., Bell, R., Volinsky, C., 2009: Matrix Factorization Techniques for Recommender Systems, Computer, Volume 42, Issue 8, 30-37.

Koren, Y., Bell, R. B., 2011: Advances in Collaborative Filtering. In Ricci, F., Rokach, L., Shapira, B., Kantor, P. B. (Eds.) Recommender Systems Handbook, 145-186.

Lekakos, G., Caravelas, P., 2008: A hybrid approach for movie recommendation, Multimed Tools Appl (2008) 36:55–70.

Li, Q. ve Kim, M., 2004: Constructing User Profiles for Collaborative Recommender System, Advanced Web Technologies and Applications, Lecture Notes in Computer Science, 2004, Volume 3007/2004, 100-111.

Luo H., Niu C., Shen R.,Ullrich C., 2008, "A collaborative filtering framework based on both local user similarity and global user similarity", Mach Learn (2008) 72: 231–245.

Mak, H.,Koprinska, I., Poon, J., 2003: INTIMATE: a Web-based movie recommender using text categorization, Proc. Web Intelligence 2003, 602-605.

Melville, P., R. J. Mooney, and R. Nagarajan., 2002: Content-Boosted Collaborative Filtering for Improved Recommendations. In Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, Canada, 2002.

Miyahara, K. ve Pazzani, M. J., 2002: Improvement of Collaborative Filtering with the Simple Bayesian Classifier. IPSJ Journal, Vol.43, No.11, Information Processing Society of Japan, November, 2002.

Mobasher, B., Jin, X., And Zhou, Y., 2004: Semantically enhanced collaborative filtering on the web. In Web Mining: From Web to Semantic Web, B. B. et al., Ed. LNAI Volume 3209. Springer.

Mooney, R. J., Roy, L., 2000: Content-based book recommending using learning for text categorization. In DL '00: Proceedings of the fifth ACM conference on Digital libraries, pages 195–204. ACM, 2000.

Moshfeghi, Y., Agarwal, D., Piwowarski, B., Jose, J.M., 2009: Movie Recommender: Semantically Enriched Unified Relevance Model for Rating Prediction in Collaborative Filtering. In: Boughanem, M., et al. (eds.) ECIR 2009. LNCS, vol. 5478, pp. 54–65. Springer, Heidelberg (2009).

Santos da Silva, F., Alves, L. G. P., and Bressan, G., 2009: PersonalTVware: A proposal of architecture to support the context-aware personalized recommendation of TV programs. In Proceedings of the 7th European Conference on Interactive TV and Video.

Sarwar B., G. Karypis, J. Konstan, and J. Riedl., 2000: Application of dimensionality reduction in recommender systems – a case study. In Proc. of the ACM WebKDD Workshop, 2000.

Smyth, B., Cotter, P., 2000: A personalized televisionlistings service. Commun. ACM, 43(8):107–111, 2000.

Spangler, W. E., Gal-Or, M. and J. H. May., 2003: Using data mining to profile TV viewers. Commun. ACM, 46(12):66–72, 2003.

Steck., H., 2010: Training and testing of recommender systems on data missing not at random. In Proc. 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'10), pages 713-722, 2010.

Senliol, F., Aral, A. ve Cataltepe, Z., 2009: Feature Selection for Collective Classification, ISCIS 2009, Eylül 14-16, Kıbrıs.

Takacs, G., Pilaszy, I., Nemeth, B., Tikk, D., 2007: On the Gravity Recommendation System, KDD Cup Workshop at SIGKDD'07 San Jose, California USA.

Wang, J., de Vries, A.P., Reinders, M.J.T., 2006: Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion, SIGIR'06, Ağustos 6-11, 2006, Seattle, Washington, USA.

Westergren, T., 2011: The music genome project. Alındığı tarih: 08.05.2012 tarihinde erişildi, adres: http://www.pandora.com/

Yoshii, K. ve diğerleri, 2006: Hybrid Collaborative and Content- based Music Recommendation Using Probabilistic Model with Latent User Preferences, Proc. of the International Conference on Music Information Retrieval (ISMIR), 2006.

Yu, K., Xu, X., Ester, M., Kriegle, H.P., 2003: Feature Weighting and Instance Selection for Collaborative Filtering: An Information-Theoretic Approach, Information Systems, Volume 5, Number 2, 201-224

Url-1 < *http://code.google.com/p/zemberek/*>, alındığı tarih: 16.07.2012.

ÖZGEÇMİŞ

Ad Soyad: Mahiye Uluyağmur

Doğum Yeri ve Tarihi: Kayseri, 11 Ocak 1987

Adres: İstanbul Teknik Üniversitesi

E-Posta: muluyagmur@itu.edu.tr

Lisans: Erciyes Üniversitesi Bilgisayar Mühendisliği

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- Çataltepe Z., Uluyağmur M., Tayfur E., 2012 Adaptif Düzenlemeli Dolaylı Geri Bildirim Kullanılarak TV Program Önerisi, SIU 2012, Antalya, Türkiye.
- Uluyagmur M., Cataltepe Z., Tayfur E., 2012 Content-Based Movie Recommendation using Different Feature Sets, International Conference on Machine Learning and Data Analysis (ICMLDA'12), San Francisco, USA.