



Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Ağırlıklandırma Yöntemlerini Kullanarak Tahmin Performansının Arttırılması

Emre Sadıkoğlu¹, Burcu Demirelli Okkalıoğlu^{2*}

¹ Yalova Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Bölümü, Yalova, Türkiye (ORCID: 0000-0002-7341-4621)

² Yalova Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Yalova, Türkiye (ORCID: 0000-0003-2867-4667)

(Bu yayın 26-27 Haziran 2020 tarihinde HORA-2020 kongresinde sözlü olarak sunulmuştur.)

(DOI: 10.31590/ejosat.779171)

ATIF/REFERENCE: Sadıkoğlu, E. & Demirelli Okkalıoğlu, B. (2020). Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Ağırlıklandırma Yöntemlerini Kullanarak Tahmin Performansının Arttırılması. *Avrupa Bilim ve Teknoloji Dergisi*, (Special Issue), 110-121.

Öz

Öneri sistemleri son yıllarda e-ticaret, turizm, film, müzik ve restoran gibi çeşitli alanlarda popüler olarak uygulanmaya başlanmıştır. Araştırmacılar çeşitli algoritmalar geliştirmelerine rağmen işbirlikçi filtreleme öneri sistemlerinde en yaygın kullanılan algoritmalarından biridir. İşbirlikçi filtreleme ile kullanıcıların geçmiş tercihleri göz önünde bulundurularak gelecekte kullanıcıların beğenebileceği ürünleri önermesi hedeflenir. Mevcut tek kriterli sistemlerde kullanıcıların ürünler hakkında tek bir derecelendirme vermesi beklenir. Fakat tek bir kriter kullanıcının ürünler hakkında fikrini yansıtmayabilir. Bu nedenle çok kriterli işbirlikçi filtreleme sistemleri geliştirilmiştir. Özellikle film, otel ve restoran gibi alanlarda kullanıcılar tek bir kritere göre derecelendirme vermek yerine birden çok kritere göre beğenilerini sunmaları onlara gelecekte yeni ürünler önermekteki başarıyı arttırabilir. Çok kriterli işbirlikçi filtreleme sistemlerindeki ilk aşama öneri isteyen bir kullanıcıya benzer en yakın kullanıcıları veya ürünleri bulmaktır. Literatürde benzerlik hesaplamak için yaygın kullanılan benzerlik-tabanlı metotlar mevcuttur. Bu metotlarda benzerlik hesaplanırken kullanıcılar / ürünler arasında ortak derecelendirilen ürünlerin / kullanıcıların verileri kullanılır. Fakat ortak derecelendirilen ürünlerin / kullanıcıların sayısı çok az olmasına rağmen aralarındaki korelasyon çok yüksek hesaplanabilir. Yüksek korelasyon değerleri her zaman en iyi komşular olduğunu garanti etmeyebilir. Bu dezavantajlar göz önüne alındığında yüksek korelasyon değerleri her zaman güvenilir ve doğru tahminler elde edilmesini engelleyebilir. Makalemizde çok kriterli ürün-tabanlı işbirlikçi filtrelemede benzerlik tabanlı yaklaşımlardaki performans artışını sağlamak için mevcut benzerlik hesaplamalarını iyileştirmeyi hedefliyoruz. İki ürünü oylayan ortak kullanıcı sayısı arttıkça, iki ürün arasında hesaplanan benzerlikte daha güvenilir olacaktır. Bu nedenle, geleneksel işbirlikçi filtrelemede kullanılan Jaccard ve önem ağırlıklandırma yöntemlerini çok kriterli sistemlerde benzerlik hesaplama sürecinde ağırlıklandırma yöntemleri olarak kullanılması önerilir. Önerilen ağırlıklandırma yöntemleri ile amaç, her iki ürünü de derecelendirme veren kullanıcı sayısı azaldıkça hesaplanan benzerliği azaltmaktır. Ağırlıklandırma yöntemleri, mevcut benzerlik hesaplama işlemlerine entegre edilerek komşu seçimi ve tahmin performansı arttırılması hedeflenir. Önerilen yöntemler Yahoo!Movies veri setinin üç farklı versiyonu kullanılarak test edilmiştir. Yapılan deneyler gösteriyor ki, önerilen metotlar mevcut metotlara göre tahmin performansını ve kapsam değerlerini büyük oranda arttırmıştır.

Anahtar Kelimeler: Çok kriterli, İşbirlikçi filtreleme, Önem-ağırlıklandırma, Jaccard benzerliği, Ürün-tabanlı.

Increasing Prediction Performance Using Weighting Methods in Multi-Criteria Item-Based Collaborative Filtering

Abstract

Recommender systems have been popularly applied in various domains such as e-commerce, tourism, movie, music, and restaurants in recent years. Although researchers have introduced various algorithms, collaborative filtering is one of the most widely used algorithms in recommender systems. Collaborative filtering aims to recommend items that users might like in the future by taking into consideration the past preferences of users. In existing single criteria systems, users are expected to give a single rating about the items. However, a

* Sorumlu Yazar: Yalova Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Yalova, Türkiye, ORCID: 0000-0003-2867-4667, bdokkalioglu@yalova.edu.tr

single criterion may not reflect the user's opinion. Thus, multi-criteria collaborative filtering systems have been introduced. Multiple criteria rating instead of a single criterion can increase the accuracy of future recommendations especially in domains such as movies, hotels, and restaurants. The first step in multi-criteria collaborative filtering systems is to find users or items similar to a user who asks for a recommendation. There are similarity-based methods commonly used in the literature to calculate similarity. While calculating the similarity in these methods, the ones that are co-rated between users / items are used. Although the number of co-rated items / users is very small, the correlation between them might be calculated very high. High correlation values may not always guarantee the best neighbors. Given these disadvantages, high correlation values might prevent reliable and accurate predictions. We propose to improve existing similarity calculations in order to increase performance in similarity-based approaches in multi-criteria item-based collaborative filtering. As the number of common users rating two items increases, the similarity calculated between them becomes more reliable. Therefore, it is recommended to use Jaccard and significance-weighting methods used in traditional collaborative filtering as weighting methods in the similarity calculation process in multi-criteria systems. With the proposed weighting methods, the aim is to decrease the calculated similarity as the number of users who rated both items decreases. Weighting methods are integrated into existing similarity calculation processes and it is aimed to increase neighbor selection and prediction performance. The proposed methods have been tested using three different versions of the Yahoo!Movies dataset. Experiments show that the proposed methods have greatly improved the prediction performance and coverage values compared to existing methods.

Keywords: Multi-criteria, Collaborative filtering, Significance-weighting, Jaccard similarity, Item-based.

1. Giriş

Gelişen teknoloji ile birlikte Internet kullanıcıları artık çoğu ihtiyacı ve işini çevrimiçi olarak internet üzerinden gerçekleştirmektedir. Kullanıcılar interneti kullandıkça internet üzerindeki veri de devasa boyutlara ulaşmış ve artmaya devam etmektedir. Artan bu veri karşısında kullanıcılar çoğu zaman neyi tercih edeceğine karar veremez hâle gelmekte ve bilgi bombardımanı denilen sorun karşımıza çıkmaktadır (Schick, Gordon, ve Haka, 1990). Araştırmacılar bilgi bombardımanı karşısında çaresiz kalan kullanıcılar için yeni yöntemler geliştirmeye başlamışlar ve *öneri sistemleri* ortaya çıkmıştır (Goldberg, Nichols, Oki, ve Terry, 1992; Adomavicius ve Tuzhilin, 2005). Öneri sistemleri, kullanıcıların tercihlerini alan ve kullanıcıların muhtemelen tercih edeceği ürünleri tahmin edebilen bir sistem oluşturarak kullanıcıların bilgi bombardımanı problemini çözmelerini sağlarlar. Öneri sistemleri son yıllarda popüler hale gelmiştir. Günümüzde kitap (Linden, Smith, ve York, 2003), film (Gomez-Urbe ve Hunt, 2016), müzik (Perez-Marcos ve Batista, 2018) gibi e-ticaret ürünleri ve sosyal medya platformlarında (Arazy, Kumar ve Shapira, 2009) sıklıkla kullanılan öneri sistemleri, kullanıcının geçmiş verilerini analiz ederek gelecekte hangi ürün veya hizmeti tercih edebileceğini tahmin etmektedir (Kaleli, 2014). Öneri sistemlerinde çeşitli algoritmalar kullanılmasına rağmen, işbirlikçi filtreleme öneri sistemlerinde kullanılan en yaygın algoritmalarından biridir (Shi, Larson, ve Hanjalic, 2014). İşbirlikçi filtreleme sistemleri, bir kullanıcı-ürün matrisi üzerinden benzerlikler hesaplayarak birbirine benzeyen kullanıcıları veya ürünleri tespit etmekte ve daha sonra kullanıcıya beğenebileceği ürünü önermektedir (Herlocker, Konstan, Borchers ve Riedl, 1999). İşbirlikçi filtreleme algoritmaları iki ana grupta sınıflandırılabilirler. İlk grup hafıza-tabanlı işbirlikçi filtreleme, ikinci grup ise model-tabanlı işbirlikçi filtreleme olarak adlandırılırlar. Hafıza-tabanlı işbirlikçi filtrelemede bir kullanıcıya bir öneri üretebilmek için tüm matris üzerinde hesaplama yapılmaktadır (Adomavicius ve Tuzhilin, 2005; Ning, Desrosiers, ve Karypis, 2015). Model-tabanlı işbirlikçi filtrelemede ise eldeki matris kullanılarak bir makine öğrenmesi modeli aracılığıyla bir model üretilmekte ve kullanıcıya bir öneri üretilirken tüm matris yerine bu model kullanılarak hesaplama yapılmaktadır (Koren, Bell ve Volinsky, 2009; Aggarwall, 2016).

İşbirlikçi filtreleme sistemlerinde hafıza tabanlı yaklaşımlar, iki ayrı sınıfta incelenmektedir. İlk sınıf, kullanıcı-tabanlı yaklaşımlardır (Herlocker, Konstan, ve Riedl, 2002). Kullanıcı-tabanlı yaklaşımlarda, öneri üretilecek olan hedef kullanıcının komşuları yani hedef kullanıcıya benzeyen diğer kullanıcılar hesaba katılarak öneri üretilmektedir. İkinci sınıf ise ürün-tabanlı yaklaşımlardır (Linden, Smith, ve York, 2003). Ürün-tabanlı yaklaşımlarda ise kullanıcının değil hedef ürüne benzeyen diğer ürünler komşu seçilerek bir öneri üretilmektedir. Ürün-tabanlı işbirlikçi filtrelemedeki temel yaklaşım, ürünler arasında korelasyon saptamak ve bu korelasyona göre kullanıcılara öneri üretmektir (Sarwar, Karypis, Konstan ve Reidl, 2001). Kullanıcılar tarafından ürünlere verilmiş olan derecelendirmeler kullanılarak bir benzerlik hesaplama metoduna göre tüm ürünlerin birbiri ile arasındaki benzerlikler hesaplanmakta ve birbirine benzeyen ürünler saptanmaktadır. Daha sonra bu benzer ürünler üzerinden kullanıcının daha önce deneyimlemediği ve beğenebilme olasılığı en yüksek ürünler, kullanıcıya öneri olarak sunulmaktadır. Örneğin A ürünü ile B ürününün benzerlikleri en yüksek olsun. Buna göre A ürününe yüksek bir derecelendirme verip, B ürününü henüz oylamayan X kullanıcıya öneri olarak B ürünü sunulmaktadır. Literatürde, hafıza-tabanlı işbirlikçi filtreleme için birçok yaklaşım önerilmiştir. Bununla birlikte, k en yakın komşu algoritması kullanıcı-tabanlı ve ürün-tabanlı işbirlikçi filtrelemede en dominant ve yaygın kullanılan algoritmadır (Adomavicius ve Tuzhilin, 2005; Chae ve ark., 2018).

Geleneksel ürün-tabanlı işbirlikçi filtreleme sistemlerinde benzerlik hesaplamak ve öneri üretmek için ürünlere verilen derecelendirmelerin yer aldığı iki boyutlu kullanıcı-ürün matrisi kullanılmaktadır. Bu matris üzerinde bir kullanıcı bir ürüne sadece tek bir derecelendirme verebilmektedir. Özetle bu sistemler, tek bir kritere dayalı sistemlerdir. Oysa bir ürün için birden fazla kritere derecelendirmelerin verilmesi, ürünlerin ve kullanıcıların karakteristiğini daha iyi ortaya çıkarabilmektedir. Bu açığı gören Adomavicius ve Kwon (2007), tek kriterli sistemler yerine çok kriterli sistemlerin kullanılmasıyla daha iyi sonuçlar elde edilebileceğini gözler önüne sermiştir. Örneğin bir film için tek bir derecelendirme vermek yerine; senaryo, oyunculuk, yönetmenlik ve görsel efektler için ayrı ayrı derecelendirmeler filmin karakteristiğini özelliğini tek bir kritere göre daha iyi göstermektedir. Adomavicius ve Kwon (2007) geleneksel hafıza tabanlı yaklaşımları çok kriterli işbirlikçi filtreleme sistemlerine göre uyarlamışlardır. Geleneksel hafıza tabanlı işbirlikçi filtreleme yöntemlerinden farklı olarak birden çok kriter olduğundan benzerlik hesaplamaları için yeni metotlar önermişlerdir. Geleneksel benzerlik tabanlı yaklaşımlar kullanılabileceği gibi çok kriterli veriler çok boyutlu uzaydaki veriler olarak ele alınarak uzaklık metrikleri sayesinde aralarındaki benzerliklerin hesaplanabileceğini göstermişlerdir.

Makalemizde çok kriterli ürün-tabanlı işbirlikçi filtrelemede benzerlik tabanlı yaklaşımlardaki performans artışını sağlamak için mevcut benzerlik hesaplamalarını iyileştirmeyi hedefliyoruz. Kullanılan benzerlik fonksiyonları iki ürünü de ortak oylamış kullanıcılar üzerinden hesaplama yapmaktadır. Öneri sistemlerinin yapısı düşünüldüğünde, iki ürünü ortak oylayan kullanıcıların sayısı çok az olabilmektedir. Diğer yandan, sadece ortak kullanıcılara göre benzerlikler hesaplandığından, bazı durumlarda ortak kullanıcı sayısı çok az olmasına rağmen iki ürün arasındaki korelasyon çok yüksek olmaktadır. Bu durumda bir tahmin üreteceğimiz hedef ürüne seçilen en iyi komşular aslında hedef ürüne en benzeyen ürünler olmayabilir. İki ürün arasında hesaplanan yüksek korelasyon değeri her zaman iki ürünün birbirine benzediği sonucunu güvenilir kılmayabilir. Bu sorunların üstesinden gelmek için geleneksel işbirlikçi filtreleme sistemlerinde, ortak kullanıcıların sayıları kullanılarak farklı ağırlıklandırma yöntemleri mevcuttur. Kullanılan ağırlıklandırma yöntemlerini çok kriterli işbirlikçi filtreleme sistemlerine uyarlayarak, komşu seçimini iyileştirmeyi hedefliyoruz. İki ürünü ortak oylayan kullanıcıların sayısı ne kadar fazla ise mevcut benzerlik değerleri pozitif yönde etkilenirken, bu sayının çok az olması durumunda hesaplamalar negatif yönde etkilenir. Diğer bir deyişle, hesaplanan benzerlik değeri düşürülür. Mevcut veri setleri ile ağırlıklandırma yöntemlerinin çok kriterli ürün-tabanlı işbirlikçi filtreleme üzerindeki etkisi incelenmiştir.

Makalenin sonraki bölümleri şu şekilde ilerlemektedir. Bir sonraki bölüm olan ikinci bölümde, literatürde daha önce gerçekleştirilmiş olan ilgili çalışmaların neler olduğu ve bu çalışmalarda neler yapıldığından kısaca bahsedilmektedir. Üçüncü bölümde, kullanılan veri setleri, geleneksel ürün-tabanlı işbirlikçi filtreleme sistemleri, kullanılan benzerlik hesaplama metotları, öneri üretme işlemi, çok kriterli ürün-tabanlı işbirlikçi filtreleme sistemleri ve tarafımızca önerilen metotların içeriği ve hesaplama yöntemleri yer almaktadır. Dördüncü bölümde ise performans metrikleri, deney yöntem ve sonuçları yer almaktadır. Son bölümde ise sonuçlar bulunmaktadır.

2. İlgili Çalışmalar

Çok kriterli öneri sistemlerinin uygulanmaya başlanması 2000'li yılların başlarına dayanır. İlk olarak Plantié, Montmain, ve Dray (2005) metin madenciliği teknikleri ile çok kriterli analiz tekniklerini birleştiren bir karar verme destek sistemini film tabanlı öneri sistemleri geliştirmek için kullanılmasını önermiştir. Adomavicius ve Kwon (2007) geleneksel öneri sistemleri yerine çok kriterli öneri sistemlerinin kullanılmasının kullanıcılar arasında hesaplanan korelasyonların daha gerçekçi olma olasılığını arttırdığını belirtmişlerdir. Kullanıcılar ürünler hakkında tek bir derecelendirme vermek yerine daha çok kriter ile örneğin bir film veri setinde görsel efektler veya senaryo gibi, ürünleri derecelendirmenin aynı zamanda tavsiye doğruluğunu arttıracaklarını savunmuşlardır. Geleneksel hafıza tabanlı işbirlikçi filtreleme algoritmaları nasıl çok kriterli işbirlikçi filtreleme sistemlerine uyarlanabileceğini göstermişlerdir. İlk olarak geleneksel benzerlik tabanlı yaklaşımları kullanarak öneriler üretmişlerdir. Kullanıcılar arasında benzerlikleri hesaplarken her bir kriter için ayrı ayrı hesapladıktan sonra tüm kriter değerlerinin ortalaması veya en düşük değerlerin seçilmesini veya çok boyutlu uzaklık metrikleri kullanılarak benzerlik hesaplamalarının yapılabileceğini açıklamışlardır. Bu yaklaşım ile sadece geleneksel benzerlik hesaplama sürecinin değiştiğini geriye kalan sürecin aynı geleneksel işbirlikçi filtreleme algoritmalarında olduğu gibi sürdürülerek tahminler elde edileceği gösterilmiştir. Ayrıca, hafıza tabanlı işbirlikçi filtreleme algoritmalarının yanı sıra tek kriterli veya geleneksel öneri sistemlerinde kullanılan diğer tüm yöntemlerin de kullanılabileninden bahsetmişlerdir. Bilge ve Kaleli (2014) çok kriterli ürün tabanlı işbirlikçi filtreleme algoritmalarının geleneksel ürün-tabanlı işbirlikçi filtreleme sistemlerine göre performanslarının daha iyi olduğunu göstermişlerdir. Araştırmacılar, Adomavicius ve Kwon (2007) tarafından önerilen çalışmaları çok kriterli ürün-tabanlı işbirlikçi filtreleme sistemlerine uyarlamışlardır. Nilashi, Ibrahim, Ithnin, ve Sarimi (2015) tek kriterli sistemlerde gerçekleştirilen otel öneri sistemlerinin çok kriterli sistemlere uyarlandığında kullanıcıların birden çok açıdan tanınabildiğini ve daha doğru öneriler yapıldığını belirtmiş, yaptıkları deneylerde de yüksek doğruluk değerlerine ulaşmışlardır. Bilge ve Yargıç (2017) çok kriterli işbirlikçi filtreleme sistemlerinde sayısal verileri olduğu gibi değil normalizasyon işleminden geçirerek kullanmışlardır. Normalize edilmiş verilerin korelasyonu daha iyi saptadığını tespit etmişler ve yaptıkları deneylerle normalize edilmiş veriler kullanıldığında daha tutarlı sonuçlara ulaşıldığını göstermişlerdir. Nilashi ve ark. (2016) çok kriterli işbirlikçi filtreleme sistemleri ile sınıflandırma ve regresyon ağacı (CART) yöntemlerini uygulamışlardır. Beklenti maksimizasyonu (EM) ve Temel Bileşen Analizi (PCA) metotları ile veri boyutunu da indirgeyerek verileri daha düzenli hale getirmişlerdir. Yaptıkları deneyler ile yüksek doğruluk oranlarına ulaşmışlardır. Batmaz ve Kaleli (2019) çok kriterli işbirlikçi filtreleme ile nöral ağlarının bir türü olan oto-kodlayıcıları (AE) kullanmışlardır. Geliştirdikleri sistemde (AE-MCCF) katman sayısının artmasıyla doğruluk değerlerinin arttığını görmüşlerdir. Araştırmacılar önerdikleri sistemi, temel çok kriterli işbirlikçi filtreleme algoritmaları ile karşılaştırmışlar ve geliştirdikleri sistemin doğruluk değerlerinin daha yüksek olduğunu göstermişlerdir. Ayrıca kapsam değerleri önerilen bu sistem ile oylanmamış ürünlerin tümü için tahminler üretmenin olası olduğunu göstermiştir.

İşbirlikçi filtreleme sistemlerinde benzerlik tabanlı yaklaşımlar kullanılırken, kullanıcılar / ürünler arasında ortak derecelendirilen ürünlerin / kullanıcıların sayısı önemli bir faktördür. Öneri sistemlerinin yapısı düşünüldüğünde, çoğu ürünün derecelendirilmediği görülür. Bu durum ise, örneğin iki kullanıcı arasında bir benzerlik hesaplaması yaparken çok az sayıda ortak derecelendirilmiş ürün üzerinden bu hesaplamaların yapılmasına sebep olur. Çok az sayıda ortak ürün üzerinden hesaplanan değerler ise gerçek korelasyonu yansıtmayabilir. Bu nedenle Herlocker ve ark. (2002) benzerlik hesaplaması yaparken tek kriterli öneri sistemlerinde bir ağırlıklandırma yöntemi kullanılmasını önermişlerdir. Önem-ağırlıklandırma olarak adlandırılan yöntemde, bir eşik değeri belirlenir ve eğer iki kullanıcının ortak oylamış oldukları ürünlerin sayısı bu değerden küçük ise benzerlik değeri hesaplanan ağırlıklandırma ile çarpılır ve böylece iki kullanıcı arasındaki benzerlik belli oranda düşürülür. Amaç iki kullanıcı arasında ne kadar çok ortak derecelendirilmiş ürün varsa, bu iki kullanıcı arasındaki benzerliğin güvenilirliğini o kadar çok arttırmaktır. Ma, King, ve Lyu (2007) önem-ağırlıklandırma yöntemini (Herlocker ve ark., 2002) değiştirerek hem kullanıcılar arasında benzerliği hem de ürünler arasında benzerliği hesaplarken kullanılmasını önerirler. Polatidis ve Georgiadis (2016) benzerlik hesaplama sürecini Herlocker ve ark. (2002)'nin çalışmalarından esinlenerek çoklu seviyeye bölerler. Önerilen çalışmada dört seviye vardır. Her seviyede belirtilen koşulların gerçekleşmesi durumunda hesaplanan benzerlik değerlerine araştırmacılar tarafından belirlenmiş bir değer eklenir. Araştırmacıların amacı iki kullanıcı arasında ne

kadar çok ortak oylanmış ürün varsa bu kullanıcıların benzerlik değerlerinin artırılmasıdır. Benzer şekilde araştırmacıların yapmış olduğu bir diğer çalışmada ise yine dinamik çok seviyeli bir işbirlikçi filtreleme sistemi önerilir (Polatidis ve Georgiadis, 2017). Bu çalışmada belirtilen kriterlerin sağlanması durumunda farklı seviyeler için benzerlik hesabına farklı sayılar eklenerek korelasyonun pozitif yönde etkilenmesi hedeflenir. En son seviyede ise diğer bir deyişle tanımlanan hiçbir kriter sağlanmıyorsa, benzerlik hesabına negatif yönde bir etki eder. Negatif yöndeki etkiyi sağlamak için benzerlik değeri belirtilen değer ile çarpılarak değerin düşürülmesi sağlanır. Candillier, Meyer, ve Fessant (2008) Jaccard benzerliğini bir ağırlıklandırma yöntemi olarak mevcut benzerlik yöntemleri ile çarpılarak kullanılmasını önermişlerdir. Yapılan çalışmalarda önerilen metodun geleneksel metotlara göre daha iyi sonuçlar verdiği gösterilmiştir. Başka bir çalışmada Jaccard benzerlik hesaplamak için kullanılmıştır (Bag, Kumar, ve Tiwari, 2019). Araştırmacılar yeni önermiş oldukları Jaccard benzerlik modeli ile hedef kullanıcının minimum ortak oylanmamış ürünlerinin sayısına ve en yakın komşularının ise maksimum ortak oylanmış ve oylanmamış ürünlerinin sayısına öncelik vermişlerdir. Shambour ve Lu (2011) Jaccard benzerliğini bir ağırlıklandırma yöntemi olarak çok kriterli ürün-tabanlı işbirlikçi filtrelemede kullanılmasını önermişlerdir. İlk olarak ayarlanabilir kosinüs benzerliği ile her bir kriter için ürünler arasında benzerlikler bulunmuştur. Son benzerlik değeri ise ağırlıklı ortalama yöntemi ile hesaplanmıştır. Jaccard ile son hesaplanmış benzerlik değeri çarpılarak yeni bir benzerlik hesabı önerilmiştir. Shambour, Hourani, ve Fraihat (2016) Dice metriğini bir ağırlıklandırma yöntemi olarak çok kriterli ürün-tabanlı işbirlikçi filtreleme için önermişlerdir. Dice metriği, Jaccard benzerliği gibi iki ürün arasındaki ortak kullanıcılara göre bir benzerlik hesaplar. Her bir kriter için ürünler arasındaki benzerlikler hesaplanırken çok boyutlu uzaklık metriği olarak Öklid uzaklığı kullanılmıştır. Son benzerlik için ise uzaklık metriği kullanıldığından toplama fonksiyonu olarak en düşük değere sahip benzerlik son benzerlik olarak seçilmiştir. Daha sonra Dice metriği son benzerlik değeri ile çarpılarak çok kriterli ürün-tabanlı sistemlerde benzerlik hesaplama işlemini geliştirmişlerdir. Son iki çalışmaya benzer bir çalışmada çok kriterli kullanıcı-tabanlı işbirlikçi filtreleme sistemleri için önerilmiştir (Shambour, 2016). Araştırmacılar, Dice metriğini diğer çalışmalarda olduğu gibi bir ağırlıklandırma yöntemi olarak ayarlanabilir kosinüs benzerliğine entegre etmişlerdir.

3. Materyal ve Metot

3.1. Veri Setleri

Çok kriterli işbirlikçi filtreleme sistemlerinde en yaygın kullanılan veri seti, Yahoo!Movies, üç farklı versiyonu ile önerdiğimiz yaklaşımları test etmek için kullanılmıştır. Veri seti film alanında 4 alt kriter ve bir tane de genel kriter olmak üzere çok kriterli tercihleri içerir. Kullanıcıların yönetmenlik, oyunculuk, senaryo ve görsel efektler gibi kriterleri derecelendirmeleri istenir. Ayrıca 4 alt kriter verilen derecelendirmelerin ortalaması da genel derecelendirme olarak sistemde kaydedilir. Derecelendirme aralığı A+ ile F aralığında olup 13 farklı derecelendirme içerir. A+ en yüksek derecelendirmeyi ifade ederken F ise en düşük derecelendirmeyi gösterir. Öneri sistemlerinde çoğu veri seti 5 yıldız derecelendirme ölçeği kullandığından dolayı, veri setindeki harfler (A+, A, A-) sayısal değeri 5, (B+, B, B-) sayısal değeri 4, diğerleri 3, 2 ve F harfinin değeri 1 olacak şekilde dönüştürülür (Jannach, Karakaya, ve Gedikli, 2012). Tablo 1'de çok kriterli kullanıcı-ürün matrisine ait küçük bir örnek verilmiştir.

Tablo 1. Çok Kriterli Kullanıcı x Ürün Matrisi Örneği

	ürün ₁	ürün ₂	ürün ₃	ürün ₄	ürün ₅	ürün ₆
kullanıcı ₁	4 _{3,5,5,3}	-	3 _{2,4,3,3}	4 _{3,5,3,5}	-	4 _{4,4,5,3}
kullanıcı ₂	-	1 _{1,1,1,1}	2 _{3,1,2,2}	-	2 _{2,2,2,2}	-
kullanıcı ₃	5 _{5,5,5,5}	3 _{2,2,4,4}	-	5 _{5,5,5,5}	3 _{4,3,3,2}	-
kullanıcı ₄	4 _{4,5,4,3}	-	-	5 _{4,5,5,5}	-	4 _{3,5,5,3}

Veri setinin aşırı seyrek olması sebebiyle, bazı kısıtlamalar altında 3 veri seti oluşturulmuştur. İlk olarak en az 5 derecelendirmeye sahip kullanıcıların ve ürünlerin bulunduğu bir alt küme oluşturulmuştur ve Tablo 2'de YM5 olarak adlandırılmıştır. Benzer şekilde YM10 ve YM20 veri setleri de elde edilmiştir.

Tablo 2. Kullanılan Veri Setleri

	YM5	YM10	YM20
# kullanıcılar	4377	1293	202
# ürünler	2565	1164	247
# derecelendirmeler	63027	34846	8157

3.2. Geleneksel Ürün-Tabanlı İşbirlikçi Filtreleme

Geleneksel ürün-tabanlı işbirlikçi filtrelemede ürünler arasındaki benzerlik hesabı tek bir kriter üzerinden gerçekleştirilmektedir (Sarwar ve ark., 2001). Örneğin bir film önerisi için benzerlikler hesaplanırken her filme sadece tek bir derecelendirme verilmiş olan kullanıcı-ürün matrisi kullanılmaktadır. Bu kullanıcı verileri, kullanıcının ürüne verebileceği derecelendirmeyi tahmin etmek için aşağıdaki formülde kullanılmaktadır.

$$R: \text{kullanıcılar } x \text{ ürünler} \rightarrow R_0 \quad (1)$$

Denklem 1’de R_0 , geleneksel ürün-tabanlı işbirlikçi filtrelemede sistem tarafından üretilen, kullanıcının ilgili ürüne verebileceği tahmin değeridir.

3.2.1. Benzerlik Hesaplama

Geleneksel ürün-tabanlı işbirlikçi filtrelemede, tüm ürünlerin birbiri ile arasındaki benzerlik değerleri hesaplanmaktadır. Bu hesaplama işlemi için literatürde farklı hesaplama metotları kullanılmıştır. Bu hesaplama metotlarından biri *Pearson Korelasyon Katsayısı (PCC)* metotudur (Resnick ve ark., 1994). PCC’de iki ürün arasındaki benzerlik değeri $[-1,1]$ aralığında yer almaktadır. Birbirine en çok benzeyen iki ürün arasındaki benzerlik değeri 1 iken, birbirine hiç benzemeyen iki ürün arasındaki benzerlik değeri ise -1 olmaktadır. PCC’de iki ürün olan i ile j arasındaki benzerlik hesabı Denklem 2’de gösterilmiştir:

$$sim(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_j)^2}} \quad (2)$$

burada $R_{u,i}$, u kullanıcısının i ürününe verdiği derecelendirmeyi gösterirken, $R_{u,j}$ ise u kullanıcısının j ürününe verdiği derecelendirmeyi ifade eder. U ise i ve j ürünlerini oylayan ortak kullanıcıları gösterir. \bar{R}_i ve \bar{R}_j , i ve j ürünlerine verilen derecelendirmelerin ortalamasını ifade eder.

Diğer bir benzerlik hesaplama metodu ise *Ayarlanabilir Kosinüs Benzerliği (ACS)*’dir (Sarwar ve ark., 2001). ACS’de ise derecelendirmeler n - boyutlu uzayda vektör olarak görülmektedir. Benzerlik değeri de bu vektörler arasındaki açı aracılığı ile hesaplanmaktadır. Kosinüs benzerliği metodundan farklı olarak ortalama kullanıcı derecelendirme değerleri de kullanılmaktadır. ACS’de iki ürün arasındaki benzerlik değeri de PCC’de olduğu gibi $[-1,1]$ aralığında yer almaktadır. ACS’de iki ürün olan i ile j arasındaki benzerlik hesabı Denklem 3 kullanılarak gerçekleştirilmektedir:

$$sim(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}} \quad (3)$$

Denklem 3’te, $R_{u,i}$ ve $R_{u,j}$, u kullanıcısının i ve j ürünlerine verdiği derecelendirmeyi, \bar{R}_u ise u kullanıcısının derecelendirmelerinin ortalamasını göstermektedir. U ise i ve j ürünlerini oylayan ortak kullanıcıları gösterir.

3.2.2. Komşu Seçimi

Ürünler arasındaki benzerlik değerleri hesaplandıktan sonra öneri üretmek için komşu seçimi işlemi yapılmalıdır. Komşu seçimi işlemi için literatürde iki farklı metot kullanılmaktadır. İlk metotta önceden hesaplanan benzerlik değerleri sıralanarak bunlar içerisinde en büyük benzerlik değerlerine sahip k tanesi en yakın k komşu olarak seçilmektedir (Konstan ve ark., 1997). İkinci metotta ise önceden belirlenen bir eşik değeri ile komşu seçimi yapılmaktadır. Eğer hesaplanan benzerlik değeri, belirlenen eşik değerinden büyük ise komşu olarak seçilir, aksi durumlarda komşu olarak değerlendirilmezler (Kim ve Yang, 2007). Bu çalışma kapsamında k en yakın komşu metodu kullanılarak hedef ürüne benzer komşular seçilmiştir.

3.2.3. Öneri Üretme

Ürünler arasındaki benzerlik değerleri hesaplandıktan sonra kullanıcılara beğenebilecekleri ürünleri önerebilmek için benzerlik değerlerinin kullanıldığı bir formülle öneri üretme işlemi yapılmaktadır. a kullanıcısının p ürününe vereceği muhtemel derecelendirme değeri Denklem 4’te gösterildiği gibi hesaplanmaktadır (Adomavicius ve Kwon, 2007).

$$pred(a, p) = \frac{\sum_{i \in ratedItem(a)} sim(i, p) \times r_{a,i}}{\sum_{i \in ratedItem(a)} sim(i, p)} \quad (4)$$

Burada $sim(i, p)$, i ürünü ile p ürünü arasındaki benzerlik değerini, $r_{a,i}$ ise a kullanıcısının i ürününe vermiş olduğu derecelendirme değerini göstermektedir. Seçilen komşu ürünlerden a kullanıcının oylamış olduğu ürünler bu hesaplama katılır. Komşuluk seçim işlemi de önceden belirlenen k sayısına göre p ürününe en çok benzeyen ürünler kümesinin oluşturulması işlemidir. Bu k değerinin tek kriterli işbirlikçi filtreleme sistemlerinde 20 ile 50 arasında seçilmesinin daha tutarlı sonuçlar sağladığı daha önceki çalışmalarda bildirilmiştir (Herlocker ve ark., 2002).

3.3. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtreleme

Geleneksel ürün-tabanlı işbirlikçi filtrelemede benzerlik hesaplama işlemleri tek bir kriter üzerinden yapılmaktaydı. Daha sonra araştırmacılar tarafından ortaya atılan fikirle çok kriterli sistemlerin, kullanıcıların karakteristiğini daha iyi yansıtabileceği düşünülerek çok kriterli işbirlikçi filtreleme kavramı ortaya çıkmıştır (Planté ve ark., 2005; Adomavicius, Manouselis, ve Kwon, 2011). Bir ürünü değerlendirirken tek bir kriter yerine daha çok kriter üzerinden derecelendirmek kullanıcıya da esneklik sağlamaktadır. Örneğin bir kullanıcı bir filmin senaryosunu beğenirken, görsel etkilerini beğenmeyebilir. Bu durumda çok kriterli işbirlikçi filtrelemede, bu iki

kriteri ayrı ayrı derecelendirme şansına sahip olabilmekte ve kişisel eğilimlerini daha iyi yansıtabilmektedir. Böylece kullanıcılara beğenebileceği ürünleri önerirken daha tutarlı sonuçlar elde edilebilmektedir.

Tek kriterli işbirlikçi filtrelemede kullanılan tahmin formülü de çok kriterli işbirlikçi filtreleme için aşağıdaki şekilde güncelleştirilmiştir:

$$R: \text{kullanıcılar } x \text{ ürünler} \rightarrow R_0 \times R_1 \times \dots \times R_k \quad (5)$$

Denklem 5'te, R_0 , çok kriterli ürün-tabanlı işbirlikçi filtrelemede sistem tarafından üretilen, kullanıcının ilgili ürüne verebileceği genel derecelendirme tahmin değeridir. R_j , kullanıcının ilgili ürünün j . kriterine vereceği derecelendirme tahmin değeridir ve buradaki j kriter değeri $j = 1, 2, \dots, k$ değerlerini alır.

Çok kriterli ürün-tabanlı işbirlikçi filtrelemede, ürünler arasındaki benzerlik değerleri her bir kriter için geleneksel ürün-tabanlı işbirlikçi filtrelemede olduğu gibi PCC, ACS vb. benzerlik metotlarından biri kullanılarak hesaplanmaktadır. Tüm kriterler bitene kadar bu hesaplama işlemi devam etmektedir. Benzerlik hesaplama işlemi tamamlandıktan sonra, tahmin etme aşamasına geçilmektedir. Tahmin işlemi için benzerliklerin nasıl kullanılacağı belirlenmelidir. Bu belirleme işlemi için literatürde iki yaklaşım kullanılmıştır (Adomavicius ve Kwon, 2007). Bu yaklaşımlardan ilkinde tüm kriterlerin benzerlik değerlerinin ortalaması alınarak tahmin işleminde Denklem 6'da gösterildiği gibi kullanılmaktadır:

$$\text{sim}_{avg}(i, j) = \frac{\sum_{c=0}^k \text{sim}_c(i, j)}{k + 1} \quad (6)$$

Denklem 6'da, k , kriter sayısını tanımlarken, $\text{sim}_c(i, j)$ ise c . kriter için i ve j ürünleri arasındaki benzerlik değerini gösterir.

İkinci yaklaşımda ise en kötü-durum göz önüne alınmaktadır, diğer bir ifadeyle en düşük benzerlik değerleri seçilerek tahmin işleminde kullanılmaktadır. Bu seçim işlemi için Denklem 7 kullanılmaktadır:

$$\text{sim}_{min}(i, j) = \min_{c=0,1,\dots,k} \text{sim}_c(i, j) \quad (7)$$

Denklem 6 veya Denklem 7 kullanılarak iki ürün arasındaki son benzerlik değeri hesaplanır. Komşu seçimi ve tahmin üretme işlemleri ise aynı geleneksel ürün-tabanlı işbirlikçi filtrelemede olduğu gibi hesaplanır.

3.4. Önerilen Metotlar

PCC ve ACS ürün-tabanlı işbirlikçi filtrelemede benzerlik hesaplamak için en yaygın kullanılan metotların başında gelirler. Her iki metottaki temel işleyiş her iki ürünü de oylamış ortak kullanıcılar üzerinden benzerlikleri hesaplamaktır. İki ürünün ortak bir kullanıcı veya birden çok kullanıcı tarafından oylanması benzerlik hesaplaması için bir problem değildir. Diğer yandan birkaç kullanıcı tarafından oylanmış iki ürünün (a, b) benzerlik değerinin, 50 kullanıcı tarafından oylanmış iki ürünün (a, c) benzerlik değerinden fazla olması b ürününün c ürününe göre a 'ya daha benzer olduğu sonucunu çıkarmak doğru olmayabilir. Ne kadar çok bu iki ürünü oylayan ortak kullanıcılar varsa, o kadar çok benzerlik hesabının güvenilirliğinden bahsedilir. (Herlocker ve ark., 2002).

Ürün-tabanlı işbirlikçi filtrelemelerde aktif kullanıcı öneri istediği hedef ürünü sisteme gönderdikten sonra, sistem hedef ürün ile diğer ürünler arasındaki benzerliği hesaplar. En yüksek benzerliğe sahip ürünleri de komşu olarak seçer. Aktif kullanıcının istekte bulunduğu hedef ürün ile yüksek korelasyona sahip komşuları incelendiğinde bazı örneklerde iki ürünü de ortak oylayan kullanıcı sayısının düşük olmasına rağmen korelasyonlarının çok yüksek olduğu görülebilir. Ürünler arasında birkaç ortak kullanıcı gerçek korelasyonu yansıtmayabilir. Bu nedenle hesaplanmış benzerlik değerlerinin ağırlıklandırılması iki ürün arasındaki benzerliği hesaplarken daha güvenilir sonuçların elde edilmesini sağlayabilir.

Bahsedilen bu sorunun üstesinden gelmek için ilk olarak ağırlıklandırma yöntemi olarak, kullanıcı-tabanlı işbirlikçi filtrelemelerde, Herlocker ve ark. (2002) tarafından önerilen iki kullanıcı arasındaki benzerliği hesaplarken kullanılan *önem-ağırlıklandırma* yöntemini, ürün-tabanlı işbirlikçi filtrelemelerde iki ürün arasında benzerlik hesaplamasında kullanılmasını öneriyoruz. Buradaki amaç iki ürünü oylayan ortak kullanıcı sayısının önceden belirlenmiş bir eşik değerinden küçük olması durumunda aralarındaki benzerlik değerini ağırlıklandırma yöntemiyle düşürmektir. Denklem 8, ağırlıklandırma yöntemini gösterir.

$$\text{sim}'(i, j) = \begin{cases} \frac{|U_i \cap U_j|}{SW} \times \text{sim}(i, j), & \text{Eğer } |U_i \cap U_j| < SW \\ \text{sim}(i, j), & \text{diğer durumlarda} \end{cases} \quad (8)$$

Denklem 8'de, $\text{sim}(i, j)$, i ve j ürünleri arasındaki benzerliği verir. $|U_i \cap U_j|$, i ve j ürünlerini ortak oylayan kullanıcıların sayısını gösterir. SW ise eşik değerini gösterir. Eğer i ve j ürünlerini ortak oylayan kullanıcıların sayısı, SW değerinden küçükse, hesaplanan benzerlik değeri belli bir oranda ağırlıklandırılır veya başka bir deyişle benzerlik değerleri düşürülür. Diğer durumlarda ise hesaplanmış benzerlik değeri değiştirilmeden kalır. Bu durum her iki ürünü de eşik değerinden yüksek, ortak oylayan kullanıcıların var olduğunu gösterir.

İki ürün arasında ortak kullanıcıların sayısını bulmak için seçilen diğer yöntem ise Jaccard metriğidir. Jaccard iki ürün arasındaki ortak kullanıcıların sayısı ile ilgilidir. Oylamaların gerçek değerleri ile ilgilenmez. Ortak olarak oylayan ne kadar çok kullanıcı varsa,

bu iki ürün o kadar çok birbirine benzer sonucunu çıkarır. Jaccard metriği Denklem 9'da verilmiştir. Denklem 9'da $|U_i \cap U_j|$ i ve j ürünlerini ortak oylayan kullanıcıların sayısını gösterirken, $|U_i \cup U_j|$ ise i ve/veya j ürünlerini oylayan kullanıcıların toplam sayısını verir.

$$Jaccard(i, j) = \frac{|U_i \cap U_j|}{|U_i \cup U_j|} \quad (9)$$

Jaccard metriğini tek başına benzerlik ölçütü olarak kullanmak gerçek korelasyonu bulmak için yeterli değildir. Örneğin iki ürünü de oylayan 50 kullanıcı olduğunu varsayalım. Birinci ürüne yüksek derecelendirme yapıldığını ama ikinci ürüne ise en düşük derecelendirmelerin verildiğini kabul edelim. Verilen örneğe göre Jaccard benzerliğinin de yüksek çıktığını kabul edelim. Bu durumda bu iki ürün birbirine benzerdir sonucunu çıkarmak yanıltıcı olabilir. Bu nedenle, sadece Jaccard metriği benzerlik hesaplamasında yeterli değilken, ağırlıklandırma yöntemi olarak kullanmak ise korelasyonları hesaplarken az sayıda bulunan ortak kullanıcıların benzerlik değerini düşürebilir.

Bu çalışmamızda yukarıda bahsedilen ağırlıklandırma yöntemlerini mevcut çok kriterli ürün-tabanlı işbirlikçi filtrelemedeki benzerlik metriklerine ekleyerek, benzerlik hesaplamalarının iyileştirilmesini hedefliyoruz. İlk olarak Jaccard metriğini bir ağırlıklandırma yöntemi olarak kullanıyoruz ve onu geleneksel çok kriterli işbirlikçi filtreleme ile birleştiriyoruz. Daha önce de bahsedildiği gibi çok kriterli sistemlerde korelasyon hesaplamak için benzerlik-tabanlı yaklaşımlar kullanılabilir. Her bir kriter için iki ürün arasındaki benzerlikler geleneksel işbirlikçi filtrelemede olduğu gibi mevcut yöntemler kullanılarak hesaplanır. Son benzerlik değerini bulmak içinde ya tüm hesaplanmış değerlerin ortalaması alınır ya da en küçük hesaplanmış değer benzerlik değeri olarak seçilir (Adomavicius ve Kwon, 2007). Biz Jaccard metriğini ağırlıklandırma metriği olarak her bir kriterde ürünler arasında benzerlikler hesaplanırken kullanılmasını öneriyoruz. Böylece iki ürünü ortak oylayan kullanıcıların sayısı düştükçe onların benzerlik değerlerini de düşürüyoruz ve bu sayede daha güvenilir benzerliklerin hesaplanmasını amaçlıyoruz. Denklem 10 önermiş olduğumuz benzerlik hesaplamasını gösterir.

$$sim'_{avg}(i, j) = \frac{\sum_{c=0}^k sim_c(i, j) \times Jaccard_c(i, j)}{k + 1} \quad (10)$$

Denklem 10'da $sim_c(i, j)$, c kriteri için i ve j ürünleri arasındaki benzerliği gösterir. $Jaccard_c(i, j)$ ise c kriteri için i ve j ürünleri arasındaki ortak oylayan kullanıcıların toplam bu ürünler için oylayanlara sayısını verir. Her bir kriter için hesaplanmış benzerlik değeri ile Jaccard metriği çarpılır. Son olarak, tüm kriter değerlerinin ortalaması alınır.

Çok kriterli işbirlikçi filtreleme sistemlerinde son benzerlik değerini hesaplamak için bir diğer yöntem ise, hesaplanmış benzerlik değerlerinden en küçüğünü almaktır. Burada da benzerlikler hesaplanırken yine Jaccard metriği ağırlıklandırma metriği olarak kullanılır. Denklem 11 önermiş olduğumuz benzerlik hesaplamasını gösterir.

$$sim'_{min}(i, j) = \min_{c=0,1,\dots,k} sim_c(i, j) \times Jaccard_c(i, j) \quad (11)$$

Geleneksel tek kriterli kullanıcı-tabanlı işbirlikçi filtreleme sistemlerinde kullanılması önerilen önem-ağırlıklandırma yöntemini (Herlocker ve ark., 2002) ürünler arasında benzerlikleri hesaplamak için uyarlayarak (Denklem 8) ikinci yöntem olarak çok kriterli işbirlikçi filtreleme sistemlerine entegre ettik. Denklem 6 ve Denklem 7'de verilen formüllerde $sim_c(i, j)$ yerine her bir kriter için Denklem 8'de verilen formül kullanılarak $sim'_c(i, j)$ değerleri elde edilerek son benzerlik değerleri tüm kriter sonuçlarının ortalaması veya en küçüğü alınarak hesaplanır.

4. Araştırma Sonuçları ve Tartışma

4.1. Deney Metodolojisi ve Değerlendirme Metrikleri

Makalede önerilen yaklaşımları test etmek için seçilen üç veri seti (YM20, YM10 ve YM5) ile deneyler gerçekleştirilmiştir. Deneylerde her veri seti için k -kat çapraz doğrulama metodolojisi uygulanmıştır. k -kat çapraz doğrulama ile, her veri seti k eşit parçaya bölünmüş olup her seferinde farklı bir parça test verisi olarak seçilmiştir. Geri kalan veriler ise eğitim verilerini oluşturmuştur. Test verisindeki her bir test kullanıcının (öneri sistemlerinde aktif kullanıcı olarak adlandırılır) oylamış olduğu tüm ürünler için öneriler üretilmiştir. Burada kullandığımız metodoloji ise birini dışarıda bırak çapraz doğrulamadır. Her seferinde test kullanıcısının derecelendirmiş olduğu bir ürün seçilip onun değeri silinip sistemden tahmin etmesi istenir.

Ortalama mutlak hata (MAE), öneri sistemlerinde doğruluğu ölçmek için kullanılan en yaygın değerlendirme metriklerinden biridir. Tahmin edilen değerler ile gerçek değerler arasındaki hataların mutlak değerlerinin toplamının ortalamasıdır. Denklem 12'de, t_i tahmin edilen değeri, g_i ise gerçek değeri göstermektedir. R ise tahmin edilen test verilerinin sayısıdır. Unutulmamalıdır ki, düşük MAE sonuçları daha iyi sonuçları gösterir.

$$MAE = \frac{1}{R} \sum_{i=1}^R |t_i - g_i| \quad (12)$$

Kapsam, kullandığımız diğer değerlendirme metriğidir ve Denklem 13'te gösterilmiştir. *Kapsam*, öneri sisteminin bir tahmin üretebileceği ürünlerin yüzdesi olarak tanımlanır. Başka bir ifadeyle, tahmin istenen ürünlerin yüzde kaçına bir tahmin yapıldığını ölçer.

$$Kapsam = \frac{\text{bir tahmin üretilebilen ürünlerin sayısı}}{\text{tahmin için tüm istekte bulunulan ürünlerin sayısı}} \quad (13)$$

4.2. Deney Sonuçları

Önermiş olduğumuz iki yaklaşımı geleneksel çok kriterli ürün-tabanlı işbirlikçi filtreleme metodunun sonuçları ile karşılaştırdık. Daha önce de bahsettiğimiz gibi veri setlerine *k*-kat çapraz doğrulama metodolojisi uygulanmıştır. *k* değerimiz 10 olarak seçilmiştir. Tablo 3'te, *Geleneksel* olarak tanımlanan adından da anlaşılacağı üzere geleneksel çok kriterli ürün-tabanlı işbirlikçi filtreleme metodlarının sonuçlarını ifade ederken, Jaccard benzerliğini ağırlıklandırma yöntemi olarak çok kriterli ürün-tabanlı sistemlere entegre ettiğimiz birinci yöntem ise *Yöntem_{Jaccard}* olarak tabloda gösterilmektedir. Son olarak tabloda yer alan *Yöntem_{önem-ağırlıklandırma}* ise ikinci önermiş olduğumuz yaklaşımı temsil eder. Bu yöntemdeki Denklem 8'deki eşik değeri (SW) 50 olarak seçilmiştir. Komşu sayısı ise Bilge ve Kaleli (2014)'nin çalışmasında olduğu gibi 40 olarak tanımlanmıştır. İlk olarak YM20 veri seti kullanılarak, önerilen yaklaşımların MAE sonuçları geleneksel metod ile karşılaştırılmıştır.

Tablo 3. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Önerilen Yöntemler için MAE Sonuçları - YM20

		<i>Geleneksel</i>	<i>Yöntem_{Jaccard}</i>	<i>Yöntem_{önem-ağırlıklandırma}</i>
<i>PCC</i>	<i>avg</i>	0.6567	0.6123	0.6106
	<i>min</i>	0.6621	0.6290	0.6304
<i>ACS</i>	<i>avg</i>	0.5846	0.5514	0.5555
	<i>min</i>	0.5863	0.5643	0.5650

Makalede, işbirlikçi filtrelemede en yaygın kullanılan iki benzerlik hesaplama metriği (PCC ve ACS) için sonuçlar hesaplanmıştır. Tablo 3'te görüldüğü gibi önerilen yaklaşımlar klasik metottan daha iyi sonuçlar vermiştir. Sonuçlar değerlendirildiğinde, geleneksel yöntemde en iyi sonucun ACS kullanılarak ve son benzerlik değeri için tüm kriterler için hesaplanan benzerlik değerlerinin ortalaması (*avg*) en iyi sonucu vermiştir. Benzer şekilde önerilen her iki yöntemde de en iyi sonucun bu seçimlerde elde edildiği gösterilmiştir.

Öneri sistemlerinde doğruluk kadar kapsam da önemli bir metriktir. Bu nedenle Tablo 4, YM20 veri seti için geleneksel ve önerilen iki yöntem için kapsam değerlerini göstermektedir. Veri setinin, YM10 ve YM5 veri setlerine göre yoğun olması nedeniyle kapsam değerlerinin yüksek olduğu görülmüştür. Diğer yandan, sonuçlar dikkatlice incelendiğinde, önerilen yöntemlerin ufakta olsa kapsam değerlerini arttırdığı görülmüştür.

Tablo 4. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Önerilen Yöntemler için Kapsam Değerleri - YM20

		<i>Geleneksel</i>	<i>Yöntem_{Jaccard}</i>	<i>Yöntem_{önem-ağırlıklandırma}</i>
<i>PCC</i>	<i>avg</i>	0.9695	0.9730	0.9739
	<i>min</i>	0.9687	0.9695	0.9705
<i>ACS</i>	<i>avg</i>	0.9898	0.9948	0.9963
	<i>min</i>	0.9916	0.9930	0.9928

Deneylerimizi YM10 veri seti kullanarak tekrarladık. Tablo 5, üç yöntemin MAE sonuçlarını gösterir. Buradaki sonuçlar incelendiğinde, geleneksel yöntem ile önerilen yöntemler arasındaki farkların Tablo 3'ten daha belirgin olduğu görülüyor. En iyi MAE sonucunu veren yöntem, *Yöntem_{Jaccard}*, 0.6012, diğer yöntemimizin, *Yöntem_{önem-ağırlıklandırma}*, 0.6142, ve geleneksel yöntemde ise en iyi sonuç olarak 0.7609 MAE değeri hesaplanmıştır. Buradaki deneyler gösteriyor ki, önerilen yöntemler ile doğruluk sonuçları mevcut klasik yönteme göre oldukça arttırılmıştır. MAE değerlerinin küçük olması daha iyi doğruluk sonuçları olduğunu gösterir.

Tablo 5. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Önerilen Yöntemler için MAE Sonuçları - YM10

		<i>Geleneksel</i>	<i>Yöntem_{Jaccard}</i>	<i>Yöntem_{önem-ağırlıklandırma}</i>
<i>PCC</i>	<i>avg</i>	0.7609	0.6701	0.6780
	<i>min</i>	0.7731	0.7107	0.7094
<i>ACS</i>	<i>avg</i>	0.7757	0.6012	0.6142
	<i>min</i>	0.7773	0.6188	0.6210

Tablo 6, YM10 veri seti için kapsam değerlerini gösterir. Tablodaki değerler incelendiğinde, önerilen yöntemlerdeki kapsam değerleri klasik yönteme göre muazzam bir artış göstermektedir. Geleneksel yöntemde kapsam değeri en iyi yaklaşık %64 iken, önerilen ikinci yöntemde (*Yöntem_{önem-ağırlıklandırma}*) %91'e kadar bu değer yükselmektedir. İlk önerdiğimiz yöntemde ise bu değer %82

olarak hesaplanmıştır. Geleneksel yöntemde (ACS-AVG) %55 olarak hesaplanan kapsam değerinin önerilen yöntemler ile sırasıyla %82 ve %91 değerlerine ulaşması, önerilen yaklaşımların başarısını gösterir.

Tablo 6. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Önerilen Yöntemler için Kapsam Değerleri – YM10

		Geleneksel	Yöntem _{Jaccard}	Yöntem _{önem-ağırlıklandırma}
PCC	avg	0.6139	0.7456	0.7934
	min	0.6355	0.7277	0.7594
ACS	avg	0.5514	0.8230	0.9061
	min	0.5473	0.7854	0.8614

Tablo 3 ve Tablo 5'teki, MAE değerleri incelendiğinde ilk yöntemin (Yöntem_{Jaccard}) daha iyi sonuçlar verdiği görülür. Fakat öneri sistemlerinde, istekte bulunulan ürünlerin ne kadarına bir tahmin ürettiğimiz de büyük bir önem taşır. Bu nedenle, kapsamın değerli olduğu durumlarda Tablo 6'da görüldüğü gibi ikinci yöntemin kullanılması tavsiye edilir.

Veri setlerimiz arasında en seyrek olan YM5 veri setinde de deneylerimizi gerçekleştirdik. Tablo 7, MAE değerlerini gösterirken, Tablo 8 kapsam değerlerini içerir. Diğer iki veri setinde olduğu gibi önerilen yöntemlerin daha iyi sonuçlar verdiği görülmektedir. Veri setinin çok seyrek olması geleneksel yöntemdeki kapsam değerlerinin düşük olmasına neden olmuştur. Önerdiğimiz hipoteze göre benzerlik değerinin yüksek olması her zaman gerçek korelasyonu göstermez. İki ürün arasındaki benzerliği hesaplarken çok az sayıda kullanıcı tarafından oylanmaları ve benzerlik değerlerinin yüksek olması onların gerçekte benzer olduklarını güvenilir kılmaz. Bu nedenle önerilen yaklaşımlar ile iki ürün arasında ne kadar çok oylayan kullanıcı varsa onların benzerlik değerlerini olumlu yönde arttırmaya çalışırken, çok az ortak kullanıcıların olması durumunda ise benzerlik değerlerini belirli bir oranda azaltmaya çalışıyoruz. Böylelikle hem tahminlerin doğruluk değerini arttırmayı hedeflerken hem de ürünlere bir tahmin üretebilmek için yeterli ve güvenilir komşular seçebiliyoruz.

Tablo 7'de görüldüğü gibi en iyi sonuçları veren durum ACS ile birlikte tüm kriterlerin ortalama benzerliklerinin kullanılmasıdır. Geleneksel metotlarda en iyi MAE sonucu 0.7733 iken, önerilen ilk metot ile sonuç 0.5920'ye kadar düşer. Ayrıca ikinci metottaki sonuçlarında yine geleneksel metotlardan çok daha iyi olduğu görülür.

Tablo 7. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Önerilen Yöntemler için MAE Sonuçları – YM5

		Geleneksel	Yöntem _{Jaccard}	Yöntem _{önem-ağırlıklandırma}
PCC	avg	0.7761	0.6666	0.6780
	min	0.7851	0.7014	0.7046
ACS	avg	0.7733	0.5920	0.6132
	min	0.7857	0.5958	0.6138

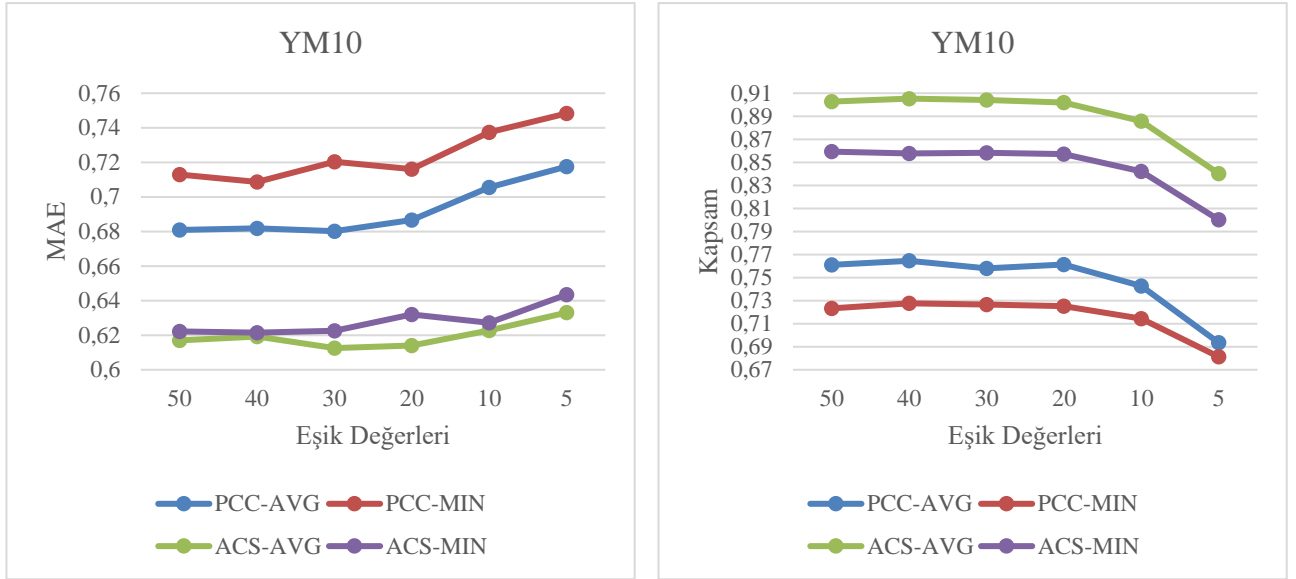
Tablo 7'deki değerler Tablo 5'e benzerlik göstermesine rağmen, iki veri seti arasındaki en büyük değişim kapsam değerlerinde görülmüştür. Tablo 6 ve Tablo 8 karşılaştırıldığında, geleneksel metotlardaki kapsam değerlerinin neredeyse yarı yarıya düştüğü görülür. Tablo 8'de geleneksel metotlardaki kapsam değerleri %34-38 aralığında iken önerilen yaklaşımlar ile %62-76'lara kadar yükseldiği görülür.

Tablo 8. Çok Kriterli Ürün-Tabanlı İşbirlikçi Filtrelemede Önerilen Yöntemler için Kapsam Değerleri – YM5

		Geleneksel	Yöntem _{Jaccard}	Yöntem _{önem-ağırlıklandırma}
PCC	avg	0.3656	0.5289	0.5938
	min	0.3761	0.5016	0.5489
ACS	avg	0.3520	0.6179	0.7580
	min	0.3415	0.5666	0.6954

Önermiş olduğumuz ikinci yöntemde eşik değeri (SW) 50 olarak seçilmişti. Farklı eşik değerlerine göre önermiş olduğumuz yöntemin sonuçlarını karşılaştırmak için yeni bir deney tasarladık. Seçilen eşik değerleri 50, 40, 30, 20, 10 ve 5'tir. Bu deneyimizde YM10 veri setini kullandık ve 5-kat çapraz doğrulama yöntemini uyguladık. Unutulmamalıdır ki, iki ürünü ortak oylayan kullanıcıların sayısı eşik değeri ile karşılaştırılır. Eğer bu sayı eşik değerinden küçükse benzerlik değerleri Denklem 8'deki gibi yeniden hesaplanır. Diğer durumlarda ise benzerlik değerleri değiştirilmez. Şekil 1, YM10 veri seti için MAE ve kapsam değerlerini gösterir.

Şekil 1'de görüldüğü gibi eşik değerleri azaldıkça, MAE değerlerinin yükseldiği görülmektedir. MAE ile doğruluk arasında ters orantı olduğu unutulmamalıdır. Daha önceki deneylerimizde en iyi sonuçları veren ACS-AVG metodunun eşik değeri 30 seçildiğinde tüm metotların içinde en iyi sonuçları verdiği şekilde gösterilmektedir. Genel olarak değerlendirildiğinde çok ufak değişimler göz önüne alınmadığında, bundan önceki deneylerde seçilen eşik değerinin (50) optimum bir değer olduğu sonucu çıkarılabilir. Benzer şekilde eşik değerleri düştükçe kapsam değerlerinin de düştüğünü ve küçük eşik değerlerinin sistemin performansı için seçilmemesi önerilir.



a) MAE Sonuçları

b) Kapsam Değerleri

Şekil 1. Önem-ağırlıklandırma Yönteminde Farklı Eşik Değerleri

5. Sonuç

Çok kriterli işbirlikçi filtreleme algoritmalarında kullanıcılar veya ürünler arasındaki benzerlik hesaplamasında yaygın olarak kullanılan PCC ve ACS metotlarının dezavantajlarını gidermek için benzerlik hesaplamalarına mevcut ağırlıklandırma yöntemlerinin eklenmesi önerilmiştir. PCC ve ACS benzerlik hesaplarırken ortak kullanıcılar veya ürünler üzerinden hesaplar. Ortak kullanıcıların veya ürünlerin toplamda kaç tane olduğuyla ilgilenmezler. İster bir tane isterse yüzlerce ortak kullanıcı veya ürün olması benzerlik hesaplamasına engel değildir. Diğer yandan az sayıda ortak kullanıcı veya ürünle hesaplanan korelasyonlar ise gerçek değerleri yansıtmayabilirler ve bu durumda komşu seçimini ve tahmin performansını etkiler. Bahsedilen sorunları hafifletmek için, mevcut ağırlıklandırma yöntemlerini çok kriterli ürün-tabanlı işbirlikçi filtreleme algoritmalarının benzerlik hesaplama sürecine entegre ettik. Ağırlıklandırma yöntemleri olarak Jaccard ve önem-ağırlıklandırma yöntemlerini seçerek mevcut benzerlik hesaplamalarının iyileştirilmesini hedefledik.

Önerdiğimiz yaklaşımları test etmek için üç veri setini kullandık. Yapılan deneyler gösteriyor ki, geleneksel yöntemlerde ve önerilen yaklaşımlarda en iyi sonucu veren ACS ile birlikte tüm kriter değerlerinin ortalamasını veren metottur (ACS-avg). YM20 veri setinde değişimler az olmasına rağmen YM10 ve YM5 veri setlerindeki sonuçlar önerilen yöntemlerin başarısını göstermektedir. YM5 veri seti kullanılarak yapılan deneylerde, geleneksel metotlarda en iyi MAE sonucu 0.7733 iken, önerilen ilk metot ($Yöntem_{Jaccard}$) ile sonuç 0.5920'ye kadar düşür. Yine benzer şekilde YM10 için MAE değerleri, geleneksel ACS-avg 0.7757 iken, $Yöntem_{Jaccard}$ 0.6012 olarak hesaplanmıştır. Sadece doğruluğun artması değil aynı şekilde kapsam değerlerinin artması da önerilen yöntemlerin başarısını gösterir. YM10 veri setinde geleneksel ACS-avg'nin kapsam değerleri %55 olarak hesaplanmıştır. $Yöntem_{Jaccard}$ ile kapsam değerleri %82 iken $Yöntem_{önem-ağırlıklandırma}$ ile bu değer %91'e kadar yükselmektedir. YM5 için sırasıyla %35, %62 ve %76 olarak hesaplanmıştır. Özetle, YM10 ve YM5 veri setleri ile yapılan deneyler incelendiğinde kapsam olarak yaklaşık olarak %36-41 bir yükselişten bahsedilir.

Makalemiz çok kriterli ürün-tabanlı işbirlikçi filtreleme algoritmalarını incelemiş ve onlar üzerinde performans arttırımı üzerinde durmuştur. Fakat önerilen yaklaşımlar kolaylıkla çok kriterli kullanıcı-tabanlı işbirlikçi filtreleme algoritmalarına da uyarlanabilirler. Ayrıca bundan sonraki çalışmalarımızda çok kriterli işbirlikçi filtreleme algoritmalarında benzerlik hesaplamak için kullanılan mevcut uzaklık metriklerinin tahmin performansını arttırmayı hedefliyoruz.

Kaynakça

- Adomavicius, G., ve Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6), 734-749. doi: 10.1109/TKDE.2005.99.
- Adomavicius, G., ve Kwon, Y. (2007). New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems*, 22(3), 48-55. <https://doi.org/10.1109/MIS.2007.58>.
- Adomavicius, G., Manouselis, N., ve Kwon, Y. (2011). *Multi-Criteria Recommender Systems. Recommender Systems Handbook*, 769-803. https://doi.org/10.1007/978-0-387-85820-3_24.
- Aggarwal, C. C. (2016). Model-Based Collaborative Filtering. *Recommender Systems*, 71-138. https://doi.org/10.1007/978-3-319-29659-3_3.
- Arazy, O., Kumar, N., ve Shapira, B. (2009). Improving Social Recommender Systems. *IT Professional*, 11(4), 38-44. <https://doi.org/10.1109/mitp.2009.76>.

- Bag, S., Kumar, S. K., ve Tiwari, M. K. (2019). An efficient recommendation generation using relevant Jaccard similarity. *Information Sciences*, 483, 53–64. <https://doi.org/10.1016/j.ins.2019.01.023>.
- Batmaz, Z., ve Kaleli, C. (2019). AE-MCCF: An Autoencoder-Based Multi-criteria Recommendation Algorithm. *Arabian Journal for Science and Engineering*, 44(11), 9235–9247. <https://doi.org/10.1007/s13369-019-03946-z>.
- Bilge, A., ve Kaleli, C. (2014). A multi-criteria item-based collaborative filtering framework. *11th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 18–22. <https://doi.org/10.1109/JCSSE.2014.6841835>.
- Bilge, A., ve Yargıç, A. (2017). Improving Accuracy Of Multi-Criteria Collaborative Filtering By Normalizing User Ratings. *Anadolu University Journal Of Science And Technology A- Applied Sciences and Engineering*, 18(1), 225–237. <https://doi.org/10.18038/aubtda.273802>.
- Candillier, L., Meyer, F., ve Fessant, F. (2008). Designing specific weighted similarity measures to improve collaborative filtering systems. *Advances in Data Mining. Medical Applications, E-Commerce, Marketing, and Theoretical Aspects*, 242–255. https://doi.org/10.1007/978-3-540-70720-2_19.
- Chae, D.K., Lee, S.C., Lee, S.Y., ve Kim, S.W. (2018). On identifying k -nearest neighbors in neighborhood models for efficient and effective collaborative filtering. *Neurocomputing*, 278, 134–143. <https://doi.org/10.1016/j.neucom.2017.06.081>.
- Goldberg, D., Nichols, D., Oki, B. M., ve Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 61–70. <https://doi.org/10.1145/138859.138867>.
- Gomez-Uribe, C. A., ve Hunt, N. (2015). The Netflix Recommender System. *ACM Transactions on Management Information Systems*, 6(4), 1–19. <https://doi.org/10.1145/2843948>.
- Herlocker, J. L., Konstan, J. A., Borchers, A., ve Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '99*. <https://doi.org/10.1145/312624.312682>.
- Herlocker, J., Konstan, J. A., ve Riedl, J. (2002). An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information retrieval*, 5(4), 287–310. doi: 10.1023/A:1020443909834.
- Jannach, D., Karakaya, Z., ve Gedikli, F. (2012). Accuracy improvements for multi-criteria recommender systems. *Proceedings of the 13th ACM Conference on Electronic Commerce- EC '12*, 674. <https://doi.org/10.1145/2229012.2229065>.
- Kaleli, C. (2014). An entropy-based neighbor selection approach for collaborative filtering. *Knowledge-Based Systems*, 56, 273–280. <https://doi.org/10.1016/j.knosys.2013.11.020>.
- Kim, T. H., ve Yang, S. B. (2007). An effective threshold-based neighbor selection in collaborative filtering. *In European Conference on Information Retrieval*, 712–715.
- Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., ve Riedl, J. (1997). GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3), 77–87. <https://doi.org/10.1145/245108.245126>.
- Koren, Y., Bell, R., ve Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8), 30–37. doi: 10.1109/MC.2009.263.
- Linden, G., Smith, B., ve York, J. (2003). Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1), 76–80. <https://doi.org/10.1109/mic.2003.1167344>.
- Ma, H., King, I., ve Lyu, M. R. (2007). Effective missing data prediction for collaborative filtering. *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval- SIGIR '07*, 39. <https://doi.org/10.1145/1277741.1277751>.
- Nilashi, M., bin Ibrahim, O., Ithnin, N., ve Sarmin, N. H. (2015). A multi-criteria collaborative filtering recommender system for the tourism domain using Expectation Maximization (EM) and PCA–ANFIS. *Electronic Commerce Research and Applications*, 14(6), 542–562. <https://doi.org/10.1016/j.elerap.2015.08.004>.
- Nilashi, M., Esfahani, M. D., Roudbaraki, M. Z., Ramayah, T., ve Ibrahim, O. (2016). A multi-criteria collaborative filtering recommender system using clustering and regression techniques. *Journal of Soft Computing and Decision Support Systems*, 3(5), 24–30.
- Ning, X., Desrosiers, C., ve Karypis, G. (2015). A Comprehensive Survey of Neighborhood-Based Recommendation Methods. *Recommender Systems Handbook*, 37–76., doi:10.1007/978-1-4899-7637-6_2.
- Pérez-Marcos, J., ve Batista, V. L. (2017). Recommender system based on collaborative filtering for spotify’s users. *In International Conference on Practical Applications of Agents and Multi-Agent Systems*, 214–220. doi: 10.1007/978-3-319-61578-3_22.
- Plantíé, M., Montmain, J., ve Dray, G. (2005). Movies recommenders systems: automation of the information and evaluation phases in a multi-criteria decision-making process. *Lecture Notes in Computer Science*, 633–644. https://doi.org/10.1007/11546924_62.
- Polatidis, N., ve Georgiadis, C. K. (2016). A multi-level collaborative filtering method that improves recommendations. *Expert Systems with Applications*, 48, 100–110. <https://doi.org/10.1016/j.eswa.2015.11.023>.
- Polatidis, N., ve Georgiadis, C.K. (2017). A dynamic multi-level collaborative filtering method for improved recommendations. *Computer Standards & Interfaces*, 51, 14–21. <https://doi.org/10.1016/j.csi.2016.10.014>.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., ve Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. *In Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 175–186. <https://doi.org/10.1145/192844.192905>.
- Sarwar, B., Karypis, G., Konstan, J., ve Reidl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the Tenth International Conference on World Wide Web- WWW '01*. <https://doi.org/10.1145/371920.372071>.
- Schick, A. G., Gordon, L. A., ve Haka, S. (1990). Information overload: A temporal approach. *Accounting, Organizations and Society*, 15(3), 199–220. [https://doi.org/10.1016/0361-3682\(90\)90005-f](https://doi.org/10.1016/0361-3682(90)90005-f).
- Shambour, Q. (2016). A user-based multi-criteria recommendation approach for personalized recommendations. *International Journal of Computer Science and Information Security*, 14(12), 657.

- Shambour, Q., ve Lu, J. (2011). A hybrid multi-criteria semantic-enhanced collaborative filtering approach for personalized recommendations. *IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, 71-78. <https://doi.org/10.1109/WI-IAT.2011.109>.
- Shambour, Q., Hourani, M., ve Fraihat, S. (2016). An item-based multi-criteria collaborative filtering algorithm for personalized recommender systems. *International Journal of Advanced Computer Science and Applications*, 7(8), 274-279. <https://doi.org/10.14569/IJACSA.2016.070837>.
- Shi, Y., Larson, M., ve Hanjalic, A. (2014). Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys*, 47(1), 1–45. <https://doi.org/10.1145/2556270>.