

**NAİVE BAYES SINIFLANDIRICI TABANLI
İKİLİ-VERİ ÇOKLU-ÖLÇÜTLÜ ÖNERİ
SİSTEMLERİ**

Yüksek Lisans Tezi

Emre YALÇIN

Eskişehir, 2016

**NAİVE BAYES SINIFLANDIRICI TABANLI İKİLİ-VERİ ÇOKLU-ÖLÇÜTLÜ ÖNERİ
SİSTEMLERİ**

Emre YALÇIN

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Yard. Doç. Dr. Alper BİLGE

Eskişehir

Anadolu Üniversitesi

Fen Bilimleri Enstitüsü

Temmuz, 2016

Bu Tez Çalışması BAP Komisyonunca kabul edilen 1508F588 no.lu proje kapsamında desteklenmiştir.

JÜRİ VE ENSTİTÜ ONAYI

Emre YALÇIN'ın "Naive Bayes Sınıflandırıcı Tabanlı İkili-Veri Çoklu-Ölçütlü Öneri Sistemleri" başlıklı tezi 21.07.2016 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği"nin ilgili maddeleri uyarınca, Bilgisayar Mühendisliği Anabilim dalında Yüksek Lisans tezi olarak kabul edilmiştir.

	<u>Unvanı-Adı Soyadı</u>	<u>İmza</u>
Üye (Tez Danışmanı) :	Yrd. Doç. Dr. Alper BİLGE
Üye :	Doç. Dr. Hüseyin POLAT
Üye :	Yrd. Doç. Dr. Efnan ŞORA GÜNAL

.....

Enstitü Müdürü

ÖZET

NAİVE BAYES SINIFLANDIRICI TABANLI İKİLİ-VERİ ÇOKLU-ÖLÇÜTLÜ ÖNERİ SİSTEMLERİ

Emre YALÇIN

Bilgisayar Mühendisliği Anabilim Dalı

Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Temmuz, 2016

Danışman: Yard. Doç. Dr. Alper BİLGE

Öneri sistemleri, kullanıcıların herhangi bir çabasına ihtiyaç duymadan onların kişisel özelliklerine ve geçmişteki tercihlerine uygun öğeler tavsiye edebilme yeteneğine sahiplerdir. Bazı ürün ve hizmet alımı durumlarında, ilgili ürün ya da hizmet alımının içerdiği ölçütler bazında ayrı ayrı beğeni değerleri toplamak daha verimli olabilmektedir. Bu doğrultuda araştırmacılar, kullanıcıların beğeni derecelerini daha ayrıntılı ve etkili ifade edebilecekleri çoklu-ölçütlü değerlendirme sistemlerini geliştirmişlerdir. Bu tür sistemlerde, alt-ölçüt sayısının çok olması nedeniyle nümerik veriler kullanmak yerine ikili-veri kullanmak tercih edilebilir. İkili-veri kullanan tek ölçütlü öneri sistemlerinde basit Bayes sınıflandırıcı algoritması ortak filtreleme amacıyla kullanılmaktadır. Literatürde, çoklu-ölçütlü sistemler için benzer bir çalışma mevcut değildir. Bu tezde, ikili veriye dayalı çoklu-ölçütlü öneri sistemlerinin uygulanabilirliği araştırılmaktadır. İlk olarak, genel beğeni ölçütü için kullanıcılara basit Bayes sınıflandırıcı kullanılarak öneriler üretilmiştir. Üretilen önerilerin doğruluğunu arttırmak amacıyla başarılı komşulukların belirlenmesini sağlayan kullanıcı ve ürün tabanlı benzerlik modelleri önerilmiştir. Ardından, bu modeller uyumluluk tekniği kullanılarak geliştirilmiştir. Uyumluluk tekniği, benzerlik hesabının daha yüksek kişiselleştirilme ile yapılmasını sağlamıştır. Son olarak, karma bir model önerilmiştir. Karma model, kullanıcı-tabanlı ve ürün-tabanlı benzerlik modellerinin birlikte kullanılmasını sağlamıştır. Bu modeller kullanılarak daha yüksek başarıma sahip önerilerin üretilmesi sağlanmıştır.

Anahtar Sözcükler: Öneri Sistemleri, Basit Bayes Sınıflandırıcı, Çoklu-Ölçütlü Öneri Sistemleri, Ortak Filtreleme, İkili-Veri

ABSTRACT
**BINARY-DATA MULTI-CRITERIA RECOMMENDER SYSTEMS BASED ON NAIVE
BAYES CLASSIFIER**

Emre YALÇIN

Department of Computer Engineering

Anadolu University, Graduate School of Sciences, July, 2016

Supervisor: Assist. Prof. Dr. Alper BİLGE

Recommender systems are specialized in suggesting appropriate items to users with respect to their personal characteristics and past preferences without requiring any effort of users. It might be more efficient to collect preferences of users based on multiple sub-criteria of corresponding product or service. For this purpose, researchers have proposed multi-criteria recommender systems that are convenient for more accurate and effective evaluation of items. In such systems, it might be preferable to collect binary ratings instead of numerical ones due to large number of sub-criteria. Naïve Bayes classifier is used for collaborative filtering purposes in single-criterion based recommender systems utilizing binary data. However, there is a gap in the literature in terms of a similar multi-criteria system. In this thesis, applicability of multi-criteria recommender systems based on binary data is investigated. Firstly, recommendations for users on overall preference criterion are produced employing naïve Bayes classifier. In order to improve quality of recommendations, user and item based similarity models are proposed enabling formation of more successful neighborhoods. Such models are further improved by integrating concordance measure between overall preference and sub-criteria ratings. Concordance measure provides the opportunity to calculate more personalized similarities among users. Finally, a hybrid model is proposed facilitating employing user and item based models together and statistically significantly improving quality of estimated binary referrals.

Keywords: Recommender Systems, Naïve Bayes Classifier, Multi-Criteria Recommender Systems, Collaborative Filtering, Binary Data

25/07/2016

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilmeyen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Anadolu Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı” ile tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçlara razı olduğumu bildiririm.

Emre YALÇIN

TEŐEKKÜR

Yüksek lisans tez çalışmam boyunca benden yardımlarını esirgemeyen, karşılaştığım sorunlarda bilgi ve deneyimlerini benimle paylaşan değerli tez danışmanım Yard. Doç. Dr. Alper BİLGE'ye teşekkürü bir borç bilirim.

Tez süresince desteklerini aldığım Anadolu Üniversitesi Bilgisayar Mühendisliği Bölümü'nde görev yapmakta olan tüm öğretim üyelerine ve çalışma arkadaşlarıma teşekkürlerimi sunarım.

Tez kapsamında yaptığım çalışmalarıma BAP projesi kapsamında maddi destek sağlayan Anadolu Üniversitesi'ne teşekkürlerimi sunarım.

Çalışma hayatım boyunca her zaman yanımda olan ve destek veren sevgili aileme teşekkür ederim.

Emre YALÇIN
Temmuz, 2016

İÇİNDEKİLER

	<u>Sayfa</u>
BAŞLIK SAYFASI	i
JÜRİ VE ENSTİTÜ ONAYI	ii
ÖZET	iii
ABSTRACT	iv
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	v
TEŞEKKÜR.....	vi
TABLolar DİZİNİ.....	ix
ŞEKİLLER DİZİNİ	x
SİMGELER VE KISALTMALAR DİZİNİ	xi
1. GİRİŞ.....	1
1.1. Öneri Sistemlerinde Kullanılan Yaklaşımlar.....	3
1.1.1. Ortak filtreleme	3
1.1.2. İçerik-tabanlı filtreleme	8
1.1.3. Karma yaklaşımlar	9
1.1.4. Demografik filtreleme	10
1.2. Çoklu-Ölçütlü Öneri Sistemleri.....	11
1.3. Öneri Sistemlerinde İkili-Veri Kullanımı.....	14
1.4. Basit Bayes Sınıflandırıcı Tabanlı Ortak Filtreleme	15
1.5. Önerilen Yaklaşımlar	16
2. İLGİLİ ÇALIŞMALAR	18
3. KURAMSAL BİLGİLER	21
3.1. Çoklu-Ölçütlü Ortak Filtreleme Yaklaşımları	21
3.1.1. Hafıza-tabanlı yaklaşımlar	22
3.1.2. Model-tabanlı yaklaşımlar.....	24

3.2. İkili-Veri İçin Kullanılan Benzerlik Metrikleri	25
3.3. Basit Bayes Sınıflandırıcı	28
3.4. Uyumluluk Tekniği	30
4. BASİT BAYES SINIFLANDIRICI TABANLI İKİLİ-VERİ ÇOKLU-ÖLÇÜTLÜ ÖNERİ SİSTEMLERİ	33
4.1. Kullanıcı-Tabanlı Benzerlik	34
4.2. Ürün-Tabanlı Benzerlik	35
4.3. Benzerlik Tabanlı Modellerin Uyumluluk Tekniği ile Kullanılması .	37
4.3.1. Uyumluluk tekniği ile kullanıcı-tabanlı benzerlik.....	38
4.3.2. Uyumluluk tekniği ile ürün-tabanlı benzerlik.....	40
4.4. Karma Model.....	41
5. DENEYSEL ÇALIŞMALAR.....	44
5.1. Veri Kümesi	44
5.2. Değerlendirme Ölçütleri.....	45
5.3. Metodoloji	46
5.4. Deney Sonuçları ve Analizi.....	47
5.4.1. Kullanıcı-tabanlı benzerlik modeli.....	48
5.4.2. Ürün-tabanlı benzerlik modeli.....	52
5.4.3. Uyumluluk tekniği ile kullanıcı-tabanlı benzerlik modeli	55
5.4.4. Uyumluluk tekniği ile ürün-tabanlı benzerlik modeli.....	58
5.4.5. Karma model	61
6. SONUÇLAR VE DEĞERLENDİRME	66
KAYNAKÇA	68
ÖZGEÇMİŞ	

TABLolar DİZİNİ

	<u>Sayfa</u>
Tablo 1.1. Örnek Film Öneri Sistemi.....	4
Tablo 1.2. Örnek Çoklu-Ölçütlü Öneri Sistemi Matrisi.....	13
Tablo 3.1. İkili Verilerden Oluşan Örnek Kullanıcı-Ürün Matrisi.....	26
Tablo 3.2. İkili-Veriden Oluşan i ve j Örnekleri İçin OTB Değerleri.....	27
Tablo 3.3. Kullanılan Benzerlik Metrikleri ve Formülleri	27
Tablo 3.4. İkili-Verilerden Oluşan Örnek Kullanıcı-Ürün Matrisi	29
Tablo 3.5. Bir Kullanıcının Beş Film için Beğeni Değerleri	31
Tablo 4.1. KB ve $ÜB$ Modelleri için İşlem Adımları.....	37
Tablo 5.1. YM10 ve YM20 Veri Kümeleri ile İlgili Ayrıntılı Bilgi	45
Tablo 5.2. İkili-Veri için Karışıklık Matrisi.....	45
Tablo 5.3. Salt-Bayes için SD ve $F1$ Değerleri	47
Tablo 5.4. KB Modeli/salt-Bayes için t -test Sonuçları.....	51
Tablo 5.5. $ÜB$ Modeli/salt-Bayes için t -test Sonuçları	55
Tablo 5.6. $U-KB$ Modeli/ KB Modeli için t -test Sonuçları.....	58
Tablo 5.7. $U-ÜB$ Modeli/ $ÜB$ Modeli için t -test Sonuçları.....	61
Tablo 5.8. Karma Model/ $U-KB$ ve $U-ÜB$ için t -test Sonuçları	65

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. İçerik-Tabanlı Filtreleme Mekanizması.....	9
Şekil 4.1. Kullanıcı-Tabanlı Benzerlik Yaklaşımlarının Çalışma Mekanizması	40
Şekil 4.2. Ürün-Tabanlı Benzerlik Yaklaşımlarının Çalışma Mekanizması	41
Şekil 4.3. Karma Model Yaklaşımının Çalışma Mekanizması.....	42
Şekil 4.4. Önerilen Modellere Genel Bakış	43
Şekil 5.1. KB Modeli $benz_{ort}$ için SD ve F1 Değerleri - YM20	48
Şekil 5.2. KB Modeli $benz_{min}$ için SD ve F1 Değerleri - YM20	49
Şekil 5.3. KB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM10.....	50
Şekil 5.4. KB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10	50
Şekil 5.5. ÜB Modeli $benz_{ort}$ için SD ve F1 Değerleri - YM20	52
Şekil 5.6. ÜB Modeli $benz_{min}$ için SD ve F1 Değerleri - YM20	52
Şekil 5.7. ÜB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM10.....	53
Şekil 5.8. ÜB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10	54
Şekil 5.9. U-KB Modeli $benz_{ort}$ için SD ve F1 Değerleri - YM20.....	55
Şekil 5.10. U-KB Modeli $benz_{min}$ için SD ve F1 Değerleri - YM20.....	56
Şekil 5.11. U-KB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM10	57
Şekil 5.12. U-KB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10.....	57
Şekil 5.13. U-ÜB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM20.....	59
Şekil 5.14. U-ÜB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM20.....	59
Şekil 5.15. U-ÜB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM10.....	60
Şekil 5.16. U-ÜB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10.....	60
Şekil 5.17. Karma Modeli için SD ve F1 Değerleri – YM20.....	63
Şekil 5.18. Karma Modeli için SD ve F1 Değerleri – YM10.....	64

SİMGELER VE KISALTMALAR DİZİNİ

BBS	: Basit Bayes Sınıflandırıcı
Ç	: Çağrışım
ÇÖÖS	: Çoklu-Ölçütlü Öneri Sistemleri
D	: Duyarlık
DN	: Doğru Negatif
DP	: Doğru Pozitif
F1	: F1 Metriği
GAİ	: Gizli Anlamsal İndeksleme
HRS	: Hotel Reservation Service
KB	: Kullanıcı-Tabanlı Benzerlik
KM	: Karma Model
OF	: Ortak Filtreleme
OTB	: Operasyonel Taksonomi Birimi
SD	: Sınıflandırma Doğruluğu
TBA	: Temel Bileşenler Analizi
TDA	: Tekil Değer Ayrışımı
U-KB	: Uyumluluk Tekniği ile Kullanıcı-Tabanlı Benzerlik
U-ÜB	: Uyumluluk Tekniği ile Ürün-Tabanlı Benzerlik
ÜB	: Ürün-Tabanlı Benzerlik
YM	: Yahoo! Movies
YN	: Yanlış Negatif
YP	: Yanlış Pozitif

1. GİRİŞ

Son yıllarda, İnternet kullanımının hızla artması ile birlikte birçok çevrimiçi sistem, İnternet kullanıcılarının hizmetine sunulmakta ve bu sistemler yoğun bir biçimde kullanılmaktadır. İnternet teknolojisindeki gelişimle paralel olarak sürekli geliştirilen ve çeşitlendirilen bu sistemler, kullanıcılara/tüketicilere çeşitli avantajlar sağlarken bazı dezavantajları da beraberinde getirmektedir. Bu dezavantajlardan birisi, İnternet kullanımının hızla artması neticesinde oluşan ve zaman içerisinde sürekli artan veri yoğunluğudur. İnternet kullanıcıları, bu kapsamda büyük bir veri kümesi içerisinde, ihtiyacına uygun olan ürünleri/hizmetleri seçmekte zorlanabilmektedirler. Örneğin bir e-ticaret sisteminde, müşteriler için binlerce ürün seçeneği mevcuttur. Bu durum, müşterilerin aradığı ve ihtiyaç duyduğu ürünleri doğru ve kolay bir şekilde tespit edebilmesi noktasında ciddi olumsuzluk teşkil etmektedir.

Bir tüketicinin seçeceği ürünün, o tüketicinin ihtiyaçlarını ve beklentilerini karşılaması gerekmektedir. Ürün çeşitliliğinin fazla olması, tüketicilerin doğru tercihler yapmasını zorlaştırabilmektedir. Ayrıca, tüketicilerin tercihinine sunulan ürünlerin/hizmetlerin miktarının çok fazla olması, tüketicilerin ihtiyaçları olan veya ilgilendikleri ürünleri bulabilmek için harcadıkları zamanın ve çabanın artmasına neden olmaktadır. Bu gibi problemlerin üstesinden gelebilmek ve kullanıcılara/tüketicilere, ürünler/hizmetler hakkında beğenip beğenmeyecekleri konusunda bir ön fikir verebilmek amacıyla öneri sistemleri yaygın biçimde kullanılmaktadır [1, 2]. Öneri sistemleri, kullanıcılara/tüketicilere, karar verme aşamasında yardımcı olarak, onların herhangi bir çabası olmadan, kişisel tercih ve özelliklerine uygun ürünlerin/hizmetlerin tavsiye edildiği sistemler olarak tanımlanabilir [3].

İlk olarak, 1990'lı yılların ortalarında araştırmacıların bağımsız bir alan olarak ele aldığı ve üzerinde çalışmalar yaptığı öneri sistemleri [4], günümüzde birçok popüler çevrimiçi İnternet sistemi ile birlikte çeşitli amaçlarla etkin bir biçimde kullanılmaktadır. Film, müzik, video, kitap, haber, araştırma makalesi, sosyal etiket, restoran, tatil, finansal servis vb. alanlar, öneri sistemlerinin yaygın biçimde kullanıldığı alanlara örnek olarak verilebilir. IMDb (film), YouTube (video),

Amazon.com (kitap ve birçok ürün), eBay (e-Ticaret), Google News (haber), Booking.com (konaklama), Tripadvisor (tatil), Moviefinder.com (film), Last.fm (müzik) ve Ness (restoran) öneri sistemlerinin etkin bir biçimde kullanıldığı popüler sistemlerden bazılarıdır [5]. Genel olarak, öneri sistemlerinin başlıca kullanım amaçları şu şekilde sıralanabilir:

- Sistemi kullanan kullanıcı/tüketici sayısını arttırmak ve daha fazla ticari kazanç elde etmek [6].
- Kullanıcıların/müşterilerin yaptığı aramaların kalitesini arttırmak [7].
- Müşterilere/kullanıcılara, ürünler/hizmetler hakkında ön fikirler vererek karar verme sürecinde onlara yardımcı olmak [8].
- Kullanım ihtiyacına göre, büyük miktardaki verileri daha az miktarda veriye indirgeyerek sistemin kullanılabilirliğini ve performansını arttırmak [6].
- Sistemi kullanan kullanıcıların/tüketicilerin memnuniyetini arttırarak, sisteme sadık kalmalarını sağlamak [6].
- Sistemin popüleritesini arttırmak [6].

Belirtilen amaçlar göz önünde bulundurulduğunda, öneri sistemlerinin, hem sistem sahiplerine hem de kullanıcılara/tüketicilere çeşitli avantajlar sağladığı söylenebilir. Sistem sahipleri açısından en önemli avantajlarından birisi ticari kazanç elde etmelerini sağlamak iken, sistemi kullanan kullanıcılar/tüketiciler için karar verme aşamasında destek almak, gereğinden fazla bilgi ile baş etmek yerine ihtiyaçları doğrultusunda daha az bilgi ile ilgilenmek ve memnuniyet örnek olarak verilebilir.

Giriş kısmının sonraki bölümleri şu şekilde organize edilmiştir: Öneri sistemlerinde kullanılan yaklaşımlar ayrıntılı olarak Bölüm 1.1'de anlatılmıştır. Çoklu-ölçütlü öneri sistemleri (ÇÖÖS) ile ilgili temel bilgiler Bölüm 1.2'de sunulmuştur. Öneri sistemlerinde ikili-verinin kullanımı Bölüm 1.3'te açıklanmıştır. Basit Bayes sınıflandırıcı (BBS) algoritmasının ortak filtreleme (OF) amacıyla kullanımı Bölüm 1.4'te anlatılmıştır. Son olarak, tez kapsamında araştırılan problem ve çözüm doğrultusunda önerilen metotlar hakkında genel bilgiler Bölüm 1.5'de sunulmuştur.

1.1. Öneri Sistemlerinde Kullanılan Yaklaşımlar

Tipik bir öneri sistemi, kullanıcılara/tüketicilere öneriler üretirken bazı temel yaklaşımlar kullanmaktadır. Bu yaklaşımlardan en çok kullanılanları aşağıda sunulmaktadır.

1.1.1. Ortak filtreleme

OF, öneri/tavsiye üretmek için kullanılan en yaygın ve en bilinen teknik olarak kabul edilmektedir. Bu tekniğin temel mantığı, kullanıcıların davranışlarını, aktivitelerini veya tercihlerini içeren büyük miktardaki veri kümesini bir araya getirip analiz ederek, kullanıcıların, ürünlerin veya belirli grupların birbirlerine olan benzerliklerini belirlemektir. Belirlenen benzerlikler, öneriler üretirken etkin bir biçimde kullanılmaktadır. Kullanıcıların/grupların, geçmişte yaptıkları tercihlerin birbirine benzer olması, gelecekte yapacakları tercihlerin de birbirine benzer olacağı varsayımını ortaya atmaktadır [9]. Yani bir ürün/hizmet hakkında Ayşe ile Ali aynı fikirde ise, başka bir ürün/hizmet hakkında Ayşe'nin fikrinin Ali'nin o konudaki fikriyle benzer olma ihtimalinin, diğer kullanıcıların o ürün/hizmet hakkındaki fikri ile benzer olma ihtimalinden daha fazla olduğu düşünülebilir.

Klasik bir *OF* senaryosunda, belirli sayıda kullanıcı ve ürün bulunur. Kullanıcılar, daha önce tercih ettikleri ürünler hakkındaki düşüncelerini ve değerlendirmelerini, belirli bir değerlendirme yelpazesine göre yaparlar. Yapılan bu değerlendirmeler, bir kullanıcı-ürün matrisi yardımı ile saklanır. *OF*'nin kullanıldığı öneri sistemlerinde genellikle çok fazla sayıda kullanıcı ve ürün bulunduğu için, bu matris çok büyük ve boşluklu yapıda olur. Çeşitli teknikler kullanılarak bu veriler işlenir ve kullanıcılar/ürünler arasındaki ilişkiler yardımı ile bir öneri sistemi oluşturulur. Son olarak sistem, bu benzerliklerden faydalanarak kullanıcılara, ilgili ürünler için öneriler üretir.

Daha teknik olarak ifade edilecek olursa, tipik bir *OF* senaryosunda, n adet kullanıcıyı içeren kullanıcı listesi $\{k_1, k_2, \dots, k_n\}$ ve m adet ürünü içeren ürün listesi $\{ü_1, ü_2, \dots, ü_m\}$ bulunur. Bu iki liste kullanılarak bir kullanıcı-ürün matrisi $[n \times m]$

oluşturulur. Bu matriste kullanıcıların, daha önce tercih ettikleri ürünler hakkında yapmış oldukları değerlendirmeler saklanır. Öneri sisteminin kullanıldığı sistemlerin çok sayıda kullanıcıya ve ürüne sahip olması, kullanıcı-ürün matrisinin devasa büyüklükte olmasına neden olmaktadır. Kullanıcıların çok az sayıda ürünü değerlendirmiş olması, kullanıcı-ürün matrisinin boşluklu yapıda olmasına neden olabilmektedir. Öneri sistemi tarafından öneri üretilecek kullanıcıya, aktif kullanıcı, ürüne ise hedef ürün denmektedir. Çeşitli benzerlik hesaplama algoritmaları ile aktif kullanıcıya en benzer kullanıcılar tespit edilerek komşulukların oluşturulması sağlanır. Daha sonra, aktif kullanıcıya benzer komşuların, hedef ürün için yapmış oldukları değerlendirmeler *OF* yaklaşımları yardımıyla kullanılarak, aktif kullanıcıya hedef ürün için bir öneri üretilir.

OF yaklaşımının daha iyi anlaşılabilmesi için örnek bir film öneri sistemi Tablo 1.1’de sunulmaktadır. Bu sistemde, beş adet kullanıcı, beş adet film için [1-5] sayısal değerlendirme yelpazesinde, çeşitli değerlendirmeler yapmıştır. Ayşe, “Başlangıç” filmi için öneri sisteminden bir öneri beklemektedir. Dolayısıyla, Ayşe aktif kullanıcı, “Başlangıç” filmi ise hedef ürün olarak tanımlanabilir. *OF* yaklaşımında, aktif kullanıcıya hedef ürün için öneri üretilirken, aktif kullanıcıya en benzer kullanıcıları doğru bir şekilde tespit edebilmek çok önemlidir. Kullanıcıların diğer filmlere yaptıkları değerlendirmeler gözlemlendiğinde, Ayşe ile en benzer değerlendirmeleri yapan iki kullanıcının Ali ve Mehmet olduğu görülebilmektedir. *OF* yaklaşımlarının dayandığı hipotez, Ayşe’nin diğer filmler için yaptığı değerlendirmelerle Ali’nin ve Mehmet’in yaptıklarının benzer olması nedeniyle, “Başlangıç” filmi için yapacakları değerlendirmelerin de benzer olacağı öngörüsüdür. Dolayısıyla, Ali ve Mehmet’in “Başlangıç” filmi için yaptıkları değerlendirmeler, Ayşe’ye yapılacak öneri için anlamlıdır ve öneri üretme aşamasında kullanılabilir.

Tablo 1.1. *Örnek Film Öneri Sistemi*

	Sihirbaz	Adalet	Kaçış	Düzenbaz	Başlangıç
Ali	5	1	1	2	5
Veli	2	-	3	-	5
Mehmet	4	2	-	1	5
Hasan	2	-	5	-	1
Ayşe	5	1	1	2	?

OF kullanan sistemlerin, kullanıcı tercihlerini toplarken kullandığı değerlendirme yelpazeleri farklılık gösterebilir. Örneğin kimi sistemlerde, kullanıcıların, sunulan ürünler/hizmetler hakkındaki düşünceleri görsel bir değerlendirme yelpazesi (kötüden iyiye doğru uzanan yuvarlak objeler, anket vb.) ile toplanabilir. Bazı işletmeler, müşterilerin ürünler/hizmetler hakkındaki değerlendirmelerini kendilerinin belirledikleri sayısal bir değerlendirme yelpazesinde toplayabilmektedir. Sistem, kullanıcıların ürünleri sadece beğenip beğenmemesi ile ilgilenebilir. Bu amaçla, müşterilerin değerlendirmeleri ikili-sistemde olabilmektedir. Bu sistemlerde, bir (1) ürünün beğenildiğini, sıfır (0) ürünün beğenilmediğini temsil etmektedir. Ek olarak, bazı işletmeler ise müşteri beğenilerini harf sembolleri (A+, A, B- vb.) ile toplamayı tercih etmektedir. Sistemler, tercih edilen veri toplama yöntemleri ile kullanıcı değerlendirmelerini topladıktan sonra, kullanıcılar arasındaki benzerlikleri tespit eder. Kullanıcılar arasında tespit edilen benzerlikler kullanılarak sistem tarafından kullanıcılara öneriler üretilir. *OF* yaklaşımlarında öneri üretilme amacıyla genellikle kullanılan teknikler aşağıda açıklandığı gibi üç farklı kategori altında toplanmaktadır [10];

1.1.1.1. Hafıza-tabanlı ortak filtreleme

Bu kategoride, kullanıcılar veya ürünler arasındaki benzerlikleri hesaplamak ve öneriler üretmek amacıyla, kullanıcıların, ürünler için yaptıkları değerlendirmeler kullanılmaktadır. Uygulanması kolay ve verimli bir tekniktir. Ancak, bu teknikte bir model oluşturulmadığından ve sisteme yeni dâhil olan bir kullanıcının herhangi bir ürün için geçmişte yaptığı bir değerlendirmesi bulunmadığından, bu kullanıcılar için öneri mekanizması çalışmayabilir. Kullanıcı-ürün matrisinin boşluklu yapıda olması, hafıza-tabanlı *OF* yaklaşımlarının istenilen performansta çalışmamasına neden olabilmektedir.

Hafıza-tabanlı *OF*, kullanıcılar/ürünler arasındaki benzerlikler üzerinde çalıştığı için komşuluk-tabanlı *OF* olarak da tanımlanmaktadır. Kullanıcılar/ürünler arasındaki benzerlikler Pearson ilgileşim katsayısı, kosinüs benzerliği, uzaklık-tabanlı benzerlik gibi çeşitli benzerlik metrikleri kullanılarak hesaplanabilmektedir [11, 12]. Hafıza-tabanlı *OF* yaklaşımlarında, kullanıcılar arasındaki benzerlikler

belirlendikten sonra, aktif kullanıcıya en benzer kullanıcılar, aktif kullanıcının komşuları olarak seçilmektedir. Daha sonra aktif kullanıcıya, hedef ürün için, bu komşuların yaptığı değerlendirmeler kullanılarak (ağırlıklandırılarak) öneri üretilmesi sağlanır [13]. Komşuluk seçimi işlemi, hedef ürün ile diğer ürünler arasındaki benzerliklere göre de yapılabilmektedir. Ek olarak, kullanıcı ve ürün-tabanlı benzerlik yöntemlerinin beraber kullanıldığı karma teknikler de mevcuttur. Ayrıca *eniyi-N* şeklinde tanımlanan ve aktif kullanıcının en çok ilgisini çekecek N tane ürünün tavsiye edilmesini sağlayan teknikler bulunmaktadır. *Eniyi-N* yaklaşımı hem kullanıcı hem de ürün-tabanlı olabilmektedir [14].

1.1.1.2. Model-tabanlı ortak filtreleme

Bu kategoride, eğitilmiş veri kümesinden faydalanarak kendini geliştirecek bir model oluşturulup kullanıcılara öneriler üretilir. Bu modelin oluşturulması başlangıçta zor ve karmaşık bir süreçtir. Buna ek olarak, performans açısından da çeşitli sorunlarla karşılaşılması muhtemeldir. Öneri üretilirken, veri kümesinin bütünü kullanılmaz. Bu yüzden, üretilen önerilerin doğruluğu hafıza-tabanlı sistemlerdeki kadar yüksek olmayabilir. Ancak, model-tabanlı sistemler, hafıza-tabanlı sistemlerde karşılaşılan bazı zorlukların (boşluklu-yapı, ölçeklenebilirlik vb.) üstesinden gelebilmektedir. Bu teknikte kullanılan modeller, veri işleme ve makine öğrenimi algoritmaları yardımıyla oluşturulur (regresyon-tabanlı *OF* algoritması, basit Bayes *OF* algoritması, kümeleme *OF* algoritmaları vb.) [15].

1.1.1.3. Karma teknikler

Karma teknikler, model ve hafıza-tabanlı tekniklerin sağladığı avantajları beraber kullanabilmek için geliştirilmiş tekniklerdir. Bu teknikler, *OF'*de karşılaşılan boşluklu-yapı, bilgi kaybı gibi problemlerin üstesinden gelinmesini sağlamaktadır. Ancak bu yaklaşımlar, uygulanması maliyetli ve karmaşıklığı arttıran tekniklerdir [16].

OF'nin uygulanmasında karşılaşılan bazı problemler vardır. Ticari amaçla kullanılan birçok öneri sistemi, çok büyük boyutlarda ürün kümeleri üzerinde çalışmaktadır. Dolayısıyla, *OF* amacıyla kullanılan kullanıcı-ürün matrisi son derece büyük ve boşluklu yapıdadır. Kullanıcı-ürün matrisinin boşluklu yapıda olması *OF* için öneri üretme açısından ciddi bir problemdir. Bu problem genellikle yeni bir ürün sisteme eklendiğinde ya da yeni bir kullanıcı sisteme dâhil olduğunda ortaya çıkmaktadır. Sisteme eklenen kullanıcı/ürün hakkında yeterli bilgi olmadığı için bu kullanıcıya/ürüne benzer kullanıcılar/ürünler belirlemek zordur. Bu duruma soğuk başlangıç problemi denir. İki türlü soğuk başlangıç problemi vardır. Bunlar yeni kullanıcı ve yeni ürün soğuk başlangıç problemleridir [17, 18]. Boşluklu-yapı probleminin üstesinden gelebilmek için tekil değer ayrışımı (*TDA*), gizli anlamsal indeksleme (*GAI*), temel bileşenler analizi (*TBA*) gibi boyut indirgeme teknikleri tabanlı *OF* yaklaşımları sunulmuştur [9, 19, 20].

Sistemdeki kullanıcı veya ürün sayısının çok fazla artması da *OF* amacıyla yapılan hesaplamaları zorlaştırabilir. Bu durum ölçeklenebilirlik problemi olarak adlandırılabilir. Bu problemi aşabilmek için *TDA* benzeri boyut indirgeme teknikleri kullanılmaktadır [21].

Bir öneri sisteminde, birbirinin aynısı ya da birbirine çok benzeyen, benzer özelliklere sahip ürünler farklı isimler altında bulunabilmektedir. Ürünlerin isimleri birbirinden farklı olduğu için, öneri sistemi bu ürünleri farklı ürünlermiş gibi kabul eder. Dolayısıyla kullanıcılara bu benzer ürünler tekrar tekrar önerilebilmektedir. Aynı ürünün defalarca kendisine önerilmesi, kullanıcılar açısından ciddi bir olumsuzluktur. Bu problemin üstesinden gelebilmek için birbirine çok benzeyen veya birbirinin aynısı olan ürünleri tespit edebilen algoritmalar (*GAI*, *TDA* vb.) kullanılmaktadır [22].

Öneri sisteminde bulunan bazı kullanıcıların yaptıkları tercihler, kimi zaman tutarlı olmayabilir. Bu *OF* için ciddi bir problemdir. Bu tip kullanıcılar sistemin istikrarını olumsuz yönde etkilemektedir. Bu tipte kullanıcılar, kendilerine benzer kullanıcılar için olumsuz bir örnek teşkil eder. Bu problemin üstesinden gelebilmek için içerik-tabanlı ve *OF* ile üretilen önerilerin, her bir kullanıcı için ayrı ayrı ağırlıklandırılarak beraber kullanımına olanak sağlayan karma teknikler kullanılabilir [23].

Öneri sisteminin verimliliğini olumsuz etkilemek isteyen kötü niyetli kullanıcılar olabilmektedir. Bu kullanıcılar, kötü niyetli olarak, sistemdeki ürünleri olması gerekenin tersi şekilde değerlendirebilir. Doğru olmayan bu değerlendirmeler sistemi olumsuz olarak etkilemektedir. Bu durum, diğer kullanıcılar üzerinde yanlış bir etki oluşmasına ve sistemin manipüle olmasına neden olmaktadır. Bu şekilde sistemin çalışma prensibinden faydalanarak çıkar elde etme girişimleri, genellikle ticari amaçlı sistemlerde görülmektedir. Bazı ürünler kötü niyetli olarak ön plana çıkartılabilmektedir. Bu durum, doğru öneriler üretme noktasında ciddi bir problem oluşturmaktadır. Bu saldırıların tespit edilmesi ve öneri sistemini bu saldırılardan koruyabilmek için alternatif karma *OF* sistemleri ve model-tabanlı *OF* sistemler geliştirilmiştir [24, 25].

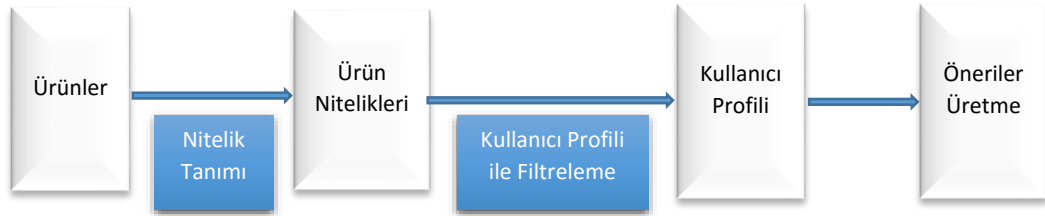
OF'de oluşturulan veri kümesi, kullanıcıların/tüketicilerin karakteristik özellikleri ve tüketim alışkanlıkları ile ilgili geniş yelpazede kişisel bilgi içermektedir [26]. Bu kişisel bilgiler mahremiyet kapsamında değerlendirildiğinde, bu bilgilerin, kötü niyetli kişiler tarafından kullanılma ihtimali ortaya çıkmaktadır. Kötü niyetli kişiler tarafından, bu bilgiler doğrultusunda arzu edilmeyen pazarlama stratejileri, şantaj veya fiyat ayrımcılığı gibi istenmeyen durumların oluşma ihtimali söz konusudur. *OF* sistemlerinin mahremiyet probleminin üstesinden gelebilmesi ve güvenilir olması kullanıcılar açısından hayati öneme sahiptir. Öneri sisteminin verimli bir şekilde çalışmasına engel olmayarak ve kullanıcıların mahremiyet konusundaki endişelerini giderebilmek için rasgele karıştırma tekniği benzeri teknikler kullanılmaktadır [27, 28].

1.1.2. İçerik-tabanlı filtreleme

Öneri sistemi tasarlanırken kullanılan bir diğer yaklaşım, içerik-tabanlı filtrelemedir. Bu yaklaşımda, kullanıcının sadece kendi geçmişi ve tercihleri referans alınarak yeni öneriler üretilmektedir. Daha geniş kapsamda, kullanıcıların başlangıçta oluşturduğu profiller ve zaman içerisinde yaptığı tercihler doğrultusunda oluşan dinamik profiller kullanılmaktadır. Kullanıcının profili incelenerek, geçmişte tercih ettiği ve beğendiği ürünler tespit edilir. Bu ürünlere benzer ürünler sistem tarafından ortaya çıkarılarak, kullanıcının bu ürünleri de

beğeneceği öngörüsü ile öneriler sunulmaktadır. İçerik-tabanlı filtrelemenin köklerinin bilgi erişimi ve bilgi filtreleme olduğu kabul edilmektedir [17, 29]. Şekil 1.1’de içerik-tabanlı filtreleme yaklaşımının çalışma mekanizması sunulmuştur. İçerik-tabanlı filtrelemenin çalışma mekanizmasının basamakları şu şekildedir:

1. Öneri üretme amacıyla, ürünlerin niteliklerinin tespit edilmesi.
2. Öneri bekleyen aktif kullanıcının geçmişteki tercihleri doğrultusunda ürünlerin niteliklerinin karşılaştırılması.
3. Aktif kullanıcının ilgisini çekeceği düşünülen ürünlerin önerilmesi.



Şekil 1.1. İçerik-Tabanlı Filtreleme Mekanizması

İçerik-tabanlı filtreleme yaklaşımında öneri üretebilmek amacıyla kullanılan karmaşık hesaplamalar gerçekleştirilmez. Ayrıca bu sistemler, benzer kullanıcıların tespit edilmesi noktasında, oluşabilecek yanlış sonuçlardan etkilenmez. Ancak, kullanıcılara sürekli benzer ürünlerin önerilmesi, kullanıcılar açısından bir olumsuzluk teşkil etmektedir. *OF*'de olduğu gibi sisteme sonradan dâhil olan yeni kullanıcıların yeterli seviyede ürün değerlendirmemiş olması ve kullanıcı geçmişi hakkında sistemin yeterli bilgiye sahip olmaması, öneri üretme aşamasında sorun teşkil etmektedir [17, 18].

1.1.3. Karma yaklaşımlar

Öneri sistemleri sınıflandırmaları ve uygulamaları genellikle *OF* ve içerik-tabanlı filtreleme yöntemleri kullanılarak uygulanmaktadır. Bu iki sistemin de kendilerine göre olumlu ve olumsuz yönleri bulunmaktadır. Daha performanslı ve verimli sistemler oluşturabilmek için bu iki yaklaşımın etkileşimi ve iyi yönleri birleştirilerek karma yaklaşımlar oluşturulmuştur [30]. Karma yaklaşımlar, *OF* ve

içerik-tabanlı filtrelemeyi birlikte kullanarak, her ikisinin olumlu yanlarının farklı algoritmalar yardımıyla birleştirilmesi ile elde edilmektedir. Netflix karma yaklaşımın kullanıldığı popüler öneri sistemlerinden biridir [17]. Netflix *OF* kullanarak kullanıcıların yaptıkları tercihlere ve beğenilere göre benzer kullanıcıların tercihleri doğrultusunda öneriler sunar. Buna ek olarak, içerik-tabanlı filtreleme yardımıyla, kullanıcıların beğendiği filmlere benzer filmler kendilerine tavsiye edilir.

1.1.4. Demografik filtreleme

Demografik filtreleme, *OF*'ye benzer şekilde, benzer eğilimleri olan kullanıcıları araştırmaktadır. *OF* kullanıcıların özellikleri yerine geçmişteki davranışları arasındaki benzerliği değerlendirirken, demografik filtreleme kullanan sistemler kullanıcıların kişisel özellikleri, cinsiyetleri, yaşları, eğitimleri, meslekleri ve yaşadıkları şehirler gibi sabit olan demografik bilgileri değerlendirir [3]. Demografik filtrelemede, kullanıcıların sistemle olan etkileşimleri, inceledikleri ürünler, gezindikleri Web siteleri gibi dinamik verileri de değerlendirebilir [5]. Bu teknikte, kullanıcılar, demografik özelliklerine göre sınıflandırılmaktadır. Bu sınıflandırmalara göre henüz deneyimlemedikleri veya kullanmadıkları ürünler sistem tarafından kullanıcılara tavsiye edilir.

Demografik filtreleme karmaşık bilgilere ihtiyaç duymamaktadır. Sınıflandırma yapıldıktan sonra kullanıcılara öneriler üretmek kolaydır. Ancak bu sınıflandırma çok genel olduğundan, yapılan önerilerin başarımı düşüktür. Sınıflandırma aşamasında kullanılan temel girdi verisi, kullanıcılardan doğrudan alındığı için eksik veya hatalı olma ihtimali mevcuttur. Ayrıca kullanıcı ihtiyaçlarının zamanla değişme ihtimali yüksek olduğu için güncelliğini yitirebilmektedir. Kişilerin demografik bilgilerini toplarken yaşanan zorluklar, kişilerin bilgilerinin sabit kalıp, zevk ve tercihlerinin zaman içerisinde değişmesi gibi etkenlerden dolayı demografik filtreleme tabanlı öneri sistemlerine olan ilgi son yıllarda azalmıştır [31].

1.2. Çoklu-Ölçütlü Öneri Sistemleri

Tipik bir tek-ölçüt tabanlı öneri sisteminde, kullanıcıların ürünler/hizmetler hakkındaki beğenileri toplanarak kullanıcı-ürün matrisi oluşturulur ve öneri üretme aşamasında bu matris kullanılır. Kullanıcıların bir ürün hakkındaki değerlendirmesi, o ürün hakkındaki genel görüşünü ifade etmektedir. Fakat bazı ürünlerde/hizmetlerde, tek bir genel beğeni yerine, bu ürünlerin/hizmetlerin daha önceden sistem sahipleri tarafından belirlenen çeşitli alt-ölçütlerle birlikte değerlendirilmesi hem sistem açısından hem de kullanıcılar açısından daha faydalı olabilmektedir. Örneğin bir kullanıcının, bir filmi değerlendirirken, film hakkındaki düşüncesini tek bir genel beğeni değeri ile belirtmek yerine, değerlendireceği filmi, görsel efektleri, oyunculuk performansı, yönetmeni vb. alt-ölçütlerle birlikte değerlendirebilmesi kullanıcının film hakkındaki duygu, düşünce ve hissiyatını daha farklı açılardan, daha doğru toplayabilme olanağı tanımaktadır. Bir başka örnek vermek gerekirse, bir otel öneri sisteminde, müşterilerin otel hakkındaki görüşlerini tek bir genel beğeni ölçütü ile değerlendirmesi yerine, otelin temizliği, konumu, fiyatı, yemek hizmetleri vb. alt-ölçütlerle birlikte değerlendirmesi tercih edilebilmektedir. Kullanıcıların değerlendirmelerini tek bir genel beğeni yerine, bu şekilde alt-ölçütlerle beraber yapabilmesi, kullanıcıların beğeni derecelerini daha doğru ve tutarlı ifade edebilmelerini sağlamaktadır. Bu doğrultuda Çoklu-ölçütlü öneri sistemleri (ÇÖÖS) yaygın bir biçimde kullanılmaya başlanmıştır.

ÇÖÖS'de kullanıcıların ürünler hakkındaki hisleri ve değerlendirmeleri, daha kişiselleştirilmiş ve doğru olarak toplanabilmektedir. Kullanıcıların ürünler hakkındaki değerlendirmelerini, genel beğenilerine ek olarak farklı ölçütler için de yapabilmesi, sistem tarafından üretilen önerilerin başarımını arttırabilmektedir. Kullanıcıların ürünler hakkındaki değerlendirmelerini farklı açılardan elde edebilmek, öneri sistemi için en önemli noktalardan birisi olan kullanıcı profili oluşturma ve benzer kullanıcıların tespiti aşamasında yararlı olabilmektedir. Tek-ölçüt tabanlı sistemlere kıyasla, ÇÖÖS daha başarılı kişiselleştirme sağlayabilmektedir. Alt-ölçütlerin değerlendirmesi ile elde edilen veriler, kullanıcılar arasındaki benzerliklerin daha doğru belirlenmesini sağlayabilmektedir.

Tek-ölçüt tabanlı sistemlere kıyasla daha verimli olabileceği düşünülen ÇÖÖS'nin günümüzdeki kullanımını her geçen gün artarak devam etmektedir. Yahoo! Movies (YM) platformu çok-ölçüt kullanan popüler film öneri sistemlerinden bir tanesidir. Bu platformda kullanıcıların ürünler hakkındaki değerlendirmelerini, genel bir değerlendirmeye ek olarak dört farklı açıdan (görsellik, oyunculuk, yönetmen ve senaryo) ve [A+, A, A-, B+, ..., F] değerlendirme yelpazesinde yapabilmektedirler. ÇÖÖS'ne bir diğer örnek ise popüler bir otel rezervasyon sistemi olan Hotel Reservation Service (HRS)'dir. Bu sistemde kullanıcılar oteller hakkındaki değerlendirmelerini, [1-10] değerlendirme yelpazesinde bir düzineden fazla alt-ölçüt için (fiyat/performans oranı, otel atmosferi/ortamı, personelin yaklaşımı, hizmet kapasitesi, kahvaltı servisi, restoran servisi, oda büyüklüğü, oda donanımı, uyku konforu vb.) yapabilmektedirler. Ek olarak, [1-3] değerlendirme yelpazesinde genel beğenilerini ifade eden genel beğeni ölçütü de bulunmaktadır.

Tek-ölçüt tabanlı *OF* sistemlerinde kullanılan veri kümeleri, kullanıcı ve ürün bilgilerinin saklandığı iki boyutlu bir matristir. Bu matris, *kullanıcı × ürün* formatındadır. Çoklu-ölçütlü *OF* sistemlerinde kullanılan veri kümesi üç boyutlu bir matris olup *kullanıcı × ürün × ölçüt* formatındadır. Kullanıcıların, ürünler için genel beğenilerine ek olarak *c* adet alt-ölçüt için değerlendirme yaptığı bir sistemde, bir kullanıcı bir ürün hakkında toplamda *c+1* değerlendirme yapmış olur. Sistemde değerlendirilmesi istenen alt-ölçütler sistem sahipleri tarafından belirlendiği için, kullanıcıların ürünler hakkındaki farklı düşünceleri kısmen kısıtlanmış olmaktadır. Yani, bir kullanıcı bir ürünü sistemde olmayan bir alt-ölçüt için değerlendirmek istiyor olabilir. Sistemde bulunan genel beğeni ölçütü bu eksikliği gidermek için faydalıdır. Genel beğeni ölçütü, bir kullanıcının bir ürün hakkındaki genel değerlendirmesini temsil etmektedir.

ÇÖÖS'nin daha iyi anlaşılabilmesi için örnek bir çoklu-ölçütlü film öneri sistemine ait veri kümesi örneği Tablo 1.2'de sunulmuştur. Tablo 1.1'de bulunan tek-ölçüt tabanlı geleneksel *OF* sistemindeki veri kümesine benzer bir veri kümesidir. Ancak bu sistemde kullanıcılar, filmleri dört alt-ölçüt ve genel bir beğeni ölçütü ile değerlendirmektedir. Yahoo! Movies film öneri sisteminin kullandığı ölçütlere benzer şekilde görsellik, oyunculuk, yönetmen ve senaryo alt ölçütleri ve film hakkında genel bir değerlendirme ölçütü kullanılmaktadır. Bu sistemde,

kullanıcılar her bir filme, beş ölçüt için [1-5] değerlendirme yelpazesinde, toplam beş değerlendirme yapmışlardır. Ayşe, sistemden “Başlangıç” filmi için bir öneri beklemektedir. ÇÖÖS’de, kullanıcılara üretilen öneriler, genel beğeni ölçütüne göre tahmin edilmektedir. Geleneksel tek-ölçüt tabanlı *OF* yaklaşımı düşünüldüğünde, Ayşe’ye benzer komşuların başarıyla tespit edilmesi gerekmektedir. Yapılan değerlendirmeler incelendiğinde, sadece genel beğeni ölçütüne göre Ayşe’ye en benzer iki kullanıcının Mehmet ve Hasan olduğu söylenebilir. Çünkü Mehmet’in ve Hasan’ın diğer dört film için yaptıkları değerlendirmeler Ayşe’nin yaptığı değerlendirmeler ile birebir aynıdır. Ancak kullanıcıların alt-ölçütlere yaptıkları değerlendirmeler dikkate alınarak incelendiğinde, Ayşe’nin, Mehmet ve Hasan ile benzer değerlendirmeler yapmadıkları görülmektedir. Ayşe’nin yaptığı değerlendirmelerin alt-ölçütlerle beraber incelenmesi, Ayşe’ye en benzer iki kullanıcının Ali ve Veli olduğunun tespit edilmesini sağlamaktadır. Dolayısıyla öneri sistemi, Ayşe’ye “Başlangıç” filmi için yapacağı öneride Ali ve Veli’yi komşu olarak belirleyip, bu kişilerin “Başlangıç” filmi için yaptığı değerlendirmelerden faydalanacaktır.

Tablo 1.2. Örnek Çoklu-Ölçütlü Öneri Sistemi Matrisi

	Sihirbaz	Adalet	Kaçış	Düzenbaz	Başlangıç
Ali	4 _{1,1,5,5}	2 _{1,1,5,5}	2 _{1,1,5,5}	1 _{1,1,5,5}	5
Veli	4 _{2,2,4,4}	2 _{2,2,4,4}	2 _{2,2,4,4}	1 _{2,2,4,4}	5
Mehmet	5 _{4,4,2,2}	1 _{4,4,2,2}	1 _{4,4,2,2}	2 _{4,4,2,2}	1
Hasan	5 _{5,5,1,1}	1 _{5,5,1,1}	1 _{5,5,1,1}	2 _{5,5,1,1}	1
Ayşe	5 _{1,1,5,5}	1 _{1,1,5,5}	1 _{1,1,5,5}	2 _{1,1,5,5}	?

Tablo 1.2’de görülen örnekten de anlaşılacağı üzere, öneri sistemlerinin ürünler için kullanıcılardan tek bir ölçüt yerine, birden fazla ölçüt ve farklı açılardan değerlendirmelerini istemeleri, sistemin daha kişiselleştirilmiş hale gelmesini ve kullanıcıların ürünler hakkındaki düşüncelerinin daha doğru yansıtılmasını sağlamaktadır. Dolayısıyla çoklu-ölçüt kullanılan sistemlerde kullanıcılar arasındaki benzerlikler daha doğru tespit edilebilmektedir.

ÇÖÖS’de kullanıcılar ürünleri tercih ederken, sistem tarafından kendilerine üretilen ve genel beğeni ölçütü için yapılan önerileri dikkate almaktadır. Tek-ölçüt tabanlı sistemlere kıyasla daha başarılı komşuluklar belirlenmesini sağlayabilen

ÇÖÖS'nde *OF* algoritmaları daha başarılı şekilde çalışabilmekte ve doğruluğu daha yüksek öneriler üretebilmektedir.

1.3. Öneri Sistemlerinde İkili-Veri Kullanımı

Günümüzde kullanılan öneri sistemlerinde, kullanıcıların ürünler hakkındaki düşünceleri farklı değerlendirme yelpazelerinde olabilmektedir. Sistemlerin belirlediği bu değerlendirme yelpazeleri genellikle sayısal bir değer aralığında olmaktadır. Örneğin, bir film öneri sistemi olan Netflix, kullanıcılardan, filmler hakkındaki düşüncelerini [1, 5] değerlendirme yelpazesinde, bir fıkra öneri sistemi olan Jester, kullanıcılardan fıkralar hakkındaki düşüncelerini [-10, +10] değerlendirme yelpazesinde yapmalarını istemektedirler. Bazı sistemlerde ise kullanıcıların değerlendirmeleri farklı anlamlar ifade eden harfler yardımıyla toplanabilmektedir. Çoklu-ölçütlü öneri sistemlerinden olan Yahoo! Movies, kullanıcıların ürünler hakkındaki düşüncelerini 13 farklı harf yardımı ile toplamaktadır. Bu sistemde, A+ en yüksek değerlendirmeyi ifade ederken, F en kötü değerlendirmeyi ifade etmektedir.

Öneri sistemlerinde, kullanıcılar, ürünler hakkında değerlendirme yaparken çok fazla vakit harcamama eğilimindedirler. Geniş değerlendirme yelpazelerinin kullanılması, kullanıcıların ürünler hakkındaki düşüncelerinin beklenen doğrulukta toplanamamasına sebep olabilmektedir. Özellikle *ÇÖÖS* düşünüldüğünde, bu olumsuzluk daha da ön plana çıkabilmektedir. Örneğin *ÇÖÖS*'den olan otel rezervasyon öneri sistemi *HRS*'de, kullanıcılar, otellerin 10'dan fazla farklı ölçütü için [1, 10] değerlendirme yelpazesinde değerlendirme yapmaktadır. Bu durum, kullanıcıların değerlendirme aşamasında sıkılmasına, dolayısıyla değerlendirmeyi bir an önce bitirebilmek için doğru olmayan değerlendirmeler yapmasına neden olabilir. Kullanıcıların, ürünler hakkındaki değerlendirmelerini sayısal bir değer olarak değil, ikili-veri formatında (1-beğendim, 0-beğenmedim) değerlendirebilmesi bazı avantajlar sağlayabilmektedir. Böyle bir değerlendirme yelpazesi, kullanıcıların ürünler hakkındaki düşüncelerini daha net ve doğru olarak ifade edebilmelerini sağlayabilmektedir. Aynı zamanda

sistemin performansı açısından da daha olumlu olabilmektedir. İkili-veri değerlendirme yelpazesinin kullanımındaki bir diğer amaç ise kullanıcının bir ölçüt için o ürünü beğenip beğenmemesi, sayısal olarak bir değerlendirme yapmasından daha anlamlı olabilecek olmasıdır. Örneğin bir kullanıcın, bir otelin erişilebilirliğini değerlendirirken sayısal bir değerlendirme yapması yerine ikili-veri şeklinde bir değerlendirme yapması daha tercih edilebilir bir durumdur. Bu doğrultuda, sistemin kullanıcılara ürünler için sayısal olarak bir öneri üretmesi yerine, beğenip beğenmeyeceği yönünde bir öneri üretmesi gerekmektedir.

1.4. Basit Bayes Sınıflandırıcı Tabanlı Ortak Filtreleme

Ürün değerlendirme yelpazesinin sayısal olarak yapıldığı sistemlerde, kullanıcılara öneriler üretmek amacıyla birçok *OF* algoritması başarıyla kullanılmaktadır [14, 16, 29, 30, 32]. Benzer şekilde, kullanıcıların ürünleri beğenip-beğenmemesi ile ilgilenen, değerlendirme yelpazesinin ikili veriye dayandığı sistemler üzerinde de *OF* amacıyla bazı algoritmalar kullanılmaktadır. *BBS OF* amacıyla etkin bir şekilde kullanılan sınıflandırma algoritmalarından biridir [33].

BBS olasılık ilkelerine göre tanımlanmış bir dizi hesaplama tekniği kullanarak, veri kümesinde bulunan verilerin sınıfını/kategorisini belirlemeyi amaçlayan bir algoritmadır. İkili verilerin kullanıldığı bir öneri sistemi düşünüldüğünde, kullanıcıların ürünler için yaptıkları değerlendirmeler iki farklı şekilde olabilmektedir. Dolayısıyla ürünler hakkında yapılan değerlendirmeler iki sınıf/kategori olarak değerlendirilebilir. Öneri sisteminin sahip olduğu ikili verilerden oluşan kullanıcı-ürün matrisi ve *BBS* algoritması kullanılarak kullanıcılar sınıflandırılıp, bu kullanıcılara ürünler hakkında sınıflandırmalar yapılarak, beğenip beğenmeyeceği yönünde tavsiyelerde bulunulur. *BBS* ile ilgili ayrıntılı bilgi Bölüm 3.3'de sunulmaktadır.

Bu kapsamda, tek-ölçüt tabanlı öneri sistemlerinde *BBS* algoritmasının *OF* amacıyla kullanıldığı çalışmalar mevcuttur [19, 33]. Çoklu-ölçütlü sistemlerin, tek-ölçütlü sistemlere kıyasla daha kişiselleştirilmiş öneriler sunduğu ve kullanıcılar arasındaki benzerliklerin daha başarılı şekilde belirlenebileceği göz önünde

bulundurulduğunda, tek-ölçüt tabanlı sistemlerde olduğu gibi, *BBS* algoritması, çoklu-ölçütlü tabanlı öneri sistemlerinde de etkin bir biçimde uygulanabilecektir. Bu tez kapsamında, ÇÖÖS'nin sağladığı avantajlar doğrultusunda *BBS* algoritmasının, bu tip sistemlerde uygulanabilirliği üzerine gerçek veri setleri kullanılarak çeşitli yaklaşımlarda bulunulmuştur.

1.5. Önerilen Yaklaşımlar

OF yaklaşımlarında, benzer kullanıcıların ve ürünlerin başarılı bir şekilde tespit edilebilmesi, kullanıcılara doğru öneriler üretebilmek için önemli bir aşamadır. Sayısal değerlendirme yelpazesi kullanılan öneri sistemlerinde, kullanıcılar ve ürünler arasındaki benzerlikleri hesaplayabilmek için çeşitli benzerlik metrikleri etkin ve başarılı bir biçimde kullanılabilir. İkili-verilerin kullanıldığı sistemlerde, kullanıcılar ve ürünler arasındaki benzerlikler, ikili-verilerden oluşan vektörler arasındaki benzerliklerin tespit edilmesini sağlayan ikili-veri benzerlik metrikleri ile hesaplanabilir. Öneri sistemlerinde kullanıcıların ürünler için yaptığı değerlendirmeleri içeren kullanıcı-ürün matrisi düşünüldüğünde, bir kullanıcının sistemdeki ürünlere yaptığı değerlendirmeler ikili verilerden oluşan yatay bir vektör olarak düşünülebilir. Benzer şekilde bir ürün için, kullanıcıların yaptıkları değerlendirmeler ikili-verilerden oluşan dikey bir vektör olarak düşünülebilir. Bu vektörler arasındaki benzerlikler, aynı zamanda kullanıcılar ve ürünler arasındaki benzerlikleri temsil eder.

Sayısal değerlendirme yelpazelerinin kullanıldığı ÇÖÖS'de, kullanıcılar/ürünler arasındaki benzerlikler, alt-ölçütlerle yapılan değerlendirmeler yardımı ve farklı benzerlik hesaplama teknikleri ile daha başarılı şekilde hesaplanabilmektedir. Benzerlik hesaplamalarının daha doğru yapılabilmesi, birbirine benzer kullanıcıların/ürünlerin daha başarılı tespit edilmesini sağlayabilmektedir. Komşulukların daha doğru tespit edilmesi, sistem tarafından, genel beğeni ölçütü için kullanıcılara yapılan önerilerin daha doğru olmasını sağlayabilmektedir.

Bu tez kapsamında, ikili-verinin kullanıldığı $\mathcal{C}\ddot{O}\ddot{O}S$ için çeşitli komşuluk belirleme modelleri önerilmiştir. Bu modellerde, sayısal değerlendirme yelpazesinin kullanıldığı $\mathcal{C}\ddot{O}\ddot{O}S$ 'deki hafıza-tabanlı OF yaklaşımları, ikili-veri kullanımına göre uyarlanmıştır. Bu yaklaşımlar kullanılarak, kullanıcılara, genel beğeni ölçütü için BBS algoritması ile yapılan önerilerin doğruluğunun arttığı, gerçek veri setleri kullanılarak gerçekleştirilen deneyler yardımıyla gözlemlenmiştir.

Tezin sonraki bölümleri şu şekilde organize edilmiştir: Tez kapsamında yapılan çalışmalarla ilgili olan ve daha önce yapılmış çalışmalar Bölüm 2'de sunulmuştur. Geliştirilen modeller temelinde, öncesinde bilinmesi gereken bilgiler Bölüm 3'te sunulmaktadır. Yapılan çalışmaların dayandığı problem ve önerilen modeller Bölüm 4'te sunulmaktadır. Geliştirilen modellerin üretilen önerilerin doğruluğu üzerindeki etkisini gözlemlemek için yapılan deneysel çalışmalar Bölüm 5'te sunulmuştur. Son olarak, Bölüm 6'da deney sonuçları değerlendirilerek yorumlanmıştır.

2. İLGİLİ ÇALIŞMALAR

OF alanında yapılan ilk çalışmalar Goldberg ve ark. [4] tarafından geliştirilen “Tapestry” algoritması ile başlamıştır. Bu sistem, kullanıcılara bilgi sistemi içerisinde bulunan değişik ürünler için yapılan sorgular doğrultusunda öneriler sunmaktadır. Tapestry otomatik olmayan bir sistem olarak tasarlanmıştır. Yapılan bu çalışmanın ardından, otomatik *OF* sistemleri yaygınlaşmaya başlamıştır. Resnick ve ark. [3] tarafından önerilen sistemde, yapılan yorumların birbirleriyle ilişkilerini otomatik olarak belirleyip, bu ilişkiler kullanılarak öneriler üretilmesi sağlanmıştır. Bu sistem kullanıcıların ilgilerini çekebilecek Usenet makalelerinin belirlenmesi amacıyla kullanılmıştır. *OF* sistemleri 1990’lı yılların ortalarından itibaren değişik alanlarda kullanılmaya başlanmıştır. Shardanand ve Maes [34] kullanıcılara ilgilerini çekebilecek müzik CD’leri önerilmesi amacıyla Ringo müzik öneri sistemini geliştirmiştir. Hill ve ark. [35] tarafından ise kullanıcılara film videoları önerebilmek amacıyla bir *OF* algoritması geliştirilmiştir. 2001 yılında ise Goldberg ve ark. [9] tarafından geliştirilen Jester *OF* sisteminde, kullanıcılara ilgilerini çekebilecek fıkraların önerilmesi amaçlanmıştır.

ÇÖÖS yaklaşımı ilk olarak, Plantie ve ark. [36] tarafından geliştirilen, bir film öneri sisteminin bilgi ve değerlendirme otomasyonunun uygulaması çalışması ile başlamıştır. Naak ve ark. [37] geliştirdiği araştırma makalesi yönetim sisteminde benzer şekilde çoklu-ölçütlü öneri yaklaşımını kullanmıştır. Bu sistemde kullanılan çoklu-ölçüt yaklaşımı sayesinde, araştırmacılar belirli ölçütlerde ilgilerini çekecek makalelere erişebilmektedirler. Film ve araştırma makaleleri üzerine yapılan çalışmalara ek olarak, turizm alanında da bazı çalışmalar yapılmıştır. Di Bitonto ve ark. [38] çoklu-ölçütlü bir mobil kültürel miras öneri sistemi geliştirmiştir. Ayrıca, Fuchs ve Zanker [39] popüler turizm platformlarından olan TripAdvisor’a yaptıkları çoklu-ölçüt tabanlı çalışma ile katkıda bulunmuştur.

Yüksek başarılı bir restoran öneri sisteminin uygulanabilmesi için Sanches Vilas ve ark. [40] çoklu-ölçüt tabanlı bir algoritma sunmuşlardır. Yapılan bir diğer çalışmada, sistemde bulunan taksonomileri ve kategorileri kullanılarak ÇÖÖS’nin analizi ve sınıflandırılması yapılmıştır [41]. Adomavicius ve Kwon [42] yaptıkları çalışmada, hafıza-tabanlı *OF* algoritmalarının, çoklu-ölçütlü sistemlere uygulanabilirliğini derin bir şekilde analiz etmişlerdir. Kullanıcıların detaylı

verileri kullanılarak, benzerlik hesaplamaları ile ilgili çeşitli yöntemler önermişlerdir. Kullanıcıların değerlendirmeleri k -boyutlu vektörler olarak düşünülebildiği için Manhattan, Öklid ve Chebyshev gibi uzaklık metriklerinin kullanılabilceğini ileri sürmüşlerdir. Her ölçüt için bağımsız olarak kullanıcılar arasındaki Pearson ilişki katsayı değerlerini hesaplandıktan sonra bunların en küçüğü veya ortalaması alınarak genel beğeni ölçütü için benzerlik metrikleri üretmişlerdir. Li ve ark. [43] tarafından geliştirilen yeni bir yaklaşım ile *OF* algoritmalarının başarımı, kullanıcıların çoklu-ölçütlü bağlamsal bilgilerini kullanarak arttırılmıştır.

Öneri sistemlerinde üretilen önerilerin doğruluğu kullanıcılar açısından büyük bir öneme sahiptir. Bu doğrultuda, üretilen önerilerin doğruluğunu artırma amacıyla, Lakiotaki ve ark. [44] çoklu-ölçütlü bir öneri sistemi geliştirmiştir. Geleneksel tek-ölçütlü *OF* yaklaşımlarına kıyasla doğruluğu yüksek önerilerin üretilmesini sağlamışlardır. Zhang ve ark. [45] yaptıkları çalışmada, iki adet çoklu-ölçütlü olasılıklı gizli anlamsal analiz algoritması önermişlerdir. Tek-ölçütlü yaklaşımlar çoklu-ölçütlü sistemlere göre uyarlanarak doğruluğu yüksek öneriler üretilmesi sağlanmıştır. Çoklu-ölçütlü sistemlerde, her ölçütün kullanıcılar üzerindeki etkisinin aynı olmayacağı hipotezine dayanan çalışmada, Liu ve ark. [46] her kullanıcı için değerlendirme aşamasında, baskın olan ölçütleri doğru bir şekilde belirleyebilme doğrultusunda çalışma yapmıştır. Lakiotaki ve ark. [47] önerdikleri öneri sistemi yaklaşımı, kullanıcı profillerini gruplandırıp, *OF* algoritmalarının uygulanmasından önce çoklu-ölçütlü karar analiz yöntemlerini kullanmak üzerine olmuştur. Tek-ölçüt tabanlı karma *OF* yaklaşımları, çoklu-ölçütlü sistemlere doğrudan uyarlanamamaktadır. Bu amaçla, Shambour ve Lu [48] önerilerin doğruluğunu arttıran çoklu-ölçütlü karma *OF* yaklaşımı önermiştir. Jannach ve ark. [49] tarafından yapılan çalışmada, çoklu-ölçütlü değerlendirmeler ve vektör regresyon teknikleri kullanılarak, her bir ölçütün kullanıcılar üzerindeki etkisi çalışılarak daha doğru öneriler üretilmesi sağlanmıştır. Hafıza-tabanlı çalışmalara ek olarak *ÇÖÖS* için model-tabanlı yaklaşımlar üzerine de bazı çalışmalar yapılmıştır [50, 51].

İkili-verilerin kullanıldığı öneri sistemlerinde, *OF* amacıyla *BBS* algoritması kullanılabilir. Miyahara ve Pazzani [33] tek-ölçütlü öneri sistemlerinde *BBS*-

tabanlı bir *OF* yaklaşımı önermişlerdir. Bu çalışmada, yapılan değerlendirmelerin pozitif ve negatif sınıflarının birbirinden bağımsız olarak kabul edilip ele alındığı dönüştürülmüş veri modeli ve yalnızca yapılan pozitif veya negatif değerlendirmelerin sınıflandırma için kullanılabileceği boşluklu yapıdaki veri modeli olmak üzere iki adet model geliştirmişlerdir. Geliştirilen modellerle üretilen önerilerin doğruluğu, geleneksel ilişki-tabanlı *OF* algoritmasıyla kıyaslanarak geliştirilen boşluklu yapıdaki veri modelinin, dönüştürülmüş veri modeli ve ilişki-tabanlı yaklaşımından daha başarılı öneriler ürettiği gözlemlenmiştir.

Miyahara ve Pazzani [33], sayısal yelpazede olan verileri ikili formata dönüştürerek, ikili-veri formatında sınıflandırma yapmışlardır. Ancak, Su ve Khoshgoftaar [52], sayısal değerlendirme yelpazesinde bulunan her bir değeri ayrı bir sınıf olarak değerlendirip, *BBS* algoritması ile her bir sınıf için istatistiksel sınıflandırma yaparak öneriler üretmişlerdir.

3. KURAMSAL BİLGİLER

3.1. Çoklu-Ölçütlü Ortak Filtreleme Yaklaşımları

Günümüzde kullanılan çoğu öneri sistemi, kullanıcıların ürünler hakkındaki düşüncelerini sayısal bir yelpazede tek bir değerlendirme ölçütü ile toplamaktadır. Tek-ölçütlü sistemlerin kullandığı veri setleri, kullanıcıların ve ürünlerin bulunduğu iki boyutlu bir matristir. Kullanıcıların yaptıkları değerlendirmelerin fonksiyonu Denklem 3.1’de sunulmaktadır:

$$R : \text{Kullanıcılar} \times \text{Ürünler} \rightarrow R_0 \quad (3.1)$$

Bu fonksiyondaki R_0 değeri, kullanıcıların ürünler hakkındaki fikir ve değerlendirmelerini temsil etmektedir. Bu değer genellikle sayısal bir yelpazede negatif olmayan bir değer şeklindedir. Bu fonksiyon, kullanıcıların ürünler için yapmış oldukları değerlendirmeler kullanılarak belirlenir. Örneğin Tablo 1.1’de sunulan örnekte, Ali’nin “Başlangıç” filmi için yapmış olduğu değerlendirme, $R(\text{Ali}, \text{Başlangıç}) = 5$, şeklinde ifade edilir. Öneri sistemi, belli bir kullanıcının belli bir ürünü ne derecede beğeneceği konusunda öneri üretmek üzerine kurulmuştur. Yani X kullanıcıya Y ürünü için bir öneri üretir. Fonksiyonel olarak bu $R(X, Y)$ şeklinde ifade edilebilir.

Öneri sistemleri kapsamında, gelecekte yoğun bir şekilde kullanılma potansiyeline sahip olan ÇÖÖS’de ise, genellikle kullanıcıların ürünler için yapmış oldukları değerlendirmeler, R_0 ’a ek olarak birbirinden bağımsız farklı m adet alt ölçüt ($c = 1, 2, \dots, m$) için toplanabilmektedir. Ayrıca bazı çoklu-ölçütlü sistemler kullanıcıların ürünler hakkındaki genel beğenisini ifade eden R_0 ölçütünü kullanmayabilir. Daha teknik olarak çoklu-ölçütlü sistemlerde kullanıcıların ürünler için yapmış oldukları değerlendirmelerin fonksiyonu iki şekilde olabilir:

$$R : \text{Kullanıcılar} \times \text{Ürünler} \rightarrow R_0 \times R_1 \times R_2 \times \dots \times R_m \quad (3.2)$$

veya R_0 ölçütü kullanılmadığında:

$$R : \text{Kullanıcılar} \times \text{Ürünler} \rightarrow R_1 \times R_2 \times \dots \times R_m \quad (3.3)$$

Çoklu-ölçüt tabanlı sistemlerde genel beğeni ölçütü ya da her bir alt-ölçüt için öneri üretilmesi amacıyla hem hafıza hem de model-tabanlı *OF* yaklaşımları ile çeşitli teknikler geliştirilmiştir.

3.1.1. Hafıza-tabanlı yaklaşımlar

Geleneksel *OF*'nin hafıza-tabanlı yaklaşımları incelendiğinde, öneri üretebilmek için, kullanıcılar/ürünler arasındaki benzerlikleri doğru bir şekilde hesaplayarak komşulukların belirlenmesinin önemli bir aşama olduğu söylenebilir. Tek-ölçüt kullanan ve sayısal bir değerlendirme yelpazesine sahip sistemlerde çeşitli benzerlik hesaplama teknikleri geliştirilmiştir. Bunlardan en popüler olanları Pearson ilgileşim katsayısı ve kosinüs benzerliğidir. $R(k, \bar{u})$, k kullanıcısının, \bar{u} ürünü için yapmış olduğu değerlendirmeyi ve $I(k, k')$, k ve k' kullanıcılarının ortak olarak değerlendirdiği ürünler kümesi olarak ifade edilirse, Pearson ilgileşim katsayısı benzerlik ölçütünün ve kosinüs benzerlik ölçütünün matematiksel formülleri sırasıyla Denklem 3.4 ve Denklem 3.5'de sunulmaktadır:

- Pearson ilgileşim katsayısı benzerlik ölçütü:

$$pearson(k, k') = \frac{\sum_{\bar{u} \in I(k, k')} (R(k, \bar{u}) - \overline{R(k)}) (R(k', \bar{u}) - \overline{R(k')})}{\sqrt{\sum_{\bar{u} \in I(k, k')} (R(k, \bar{u}) - \overline{R(k)})^2} \sqrt{\sum_{\bar{u} \in I(k, k')} (R(k', \bar{u}) - \overline{R(k')})^2}} \quad (3.4)$$

- Kosinüs-tabanlı benzerlik ölçütü:

$$kosinüs(k, k') = \frac{\sum_{\bar{u} \in I(k, k')} R(k, \bar{u}) R(k', \bar{u})}{\sqrt{\sum_{\bar{u} \in I(k, k')} R(k, \bar{u})^2} \sqrt{\sum_{\bar{u} \in I(k, k')} R(k', \bar{u})^2}} \quad (3.5)$$

Çoklu-ölçütlü sistemlerde bu matematiksel hesaplamalar doğrudan uygulanamaz. Bunun sebebi $R(k, \bar{u})$ fonksiyonunun, genel beğeniye ifade eden R_0 'a ek olarak m adet alt-ölçüt için yapılmış olan R_1, R_2, \dots, R_m değerlendirmelerini içermesidir. Dolayısıyla, $R(u, i) = (R_0, R_1, R_2, \dots, R_m)$ şeklinde olur. Toplamda $m+1$ ölçüt olduğu kabul edilirse, her (k, \bar{u}) için $m+1$ adet değerlendirme bulunur.

Kullanıcılar arasındaki nihai benzerlik değerlerini hesaplama aşamasında, yapılan $m+1$ değerlendirmeyi beraber kullanabilen iki farklı yaklaşım vardır.

İlk yaklaşımda, iki kullanıcı arasında, her bir ölçüt için ayrı ayrı Pearson ilişki katsayısı veya kosinüs-tabanlı benzerlik ölçütleri kullanarak $m+1$ adet benzerlik değeri hesaplanır. Daha sonra, hesaplanan $m+1$ adet benzerlik değeri bir araya getirilerek, iki kullanıcı arasındaki nihai benzerlik değeri hesaplanmış olur. Hesaplanan $m+1$ adet benzerlik değerini bir araya getirebilmek için Denklem 3.6 ve Denklem 3.7'de formülleri sunulan iki farklı teknik kullanılabilir [42];

- Ortalama benzerlik:

$$benz_{ort}(k, k') = \frac{1}{m+1} \sum_{c=0}^m benz_c(k, k') \quad (3.6)$$

- En küçük benzerlik:

$$benz_{min}(k, k') = \min_{c=0,1,\dots,m} benz_c(k, k') \quad (3.7)$$

İkinci yaklaşım ise, kullanıcılar arasındaki benzerlikleri, çok-boyutlu uzaklık metrikleri kullanarak hesaplamaktır. Bu yaklaşımda, Manhattan, Chebyshev ve Öklid gibi uzaklık metrikleri kullanılır [42]. Bir ürün (\ddot{u}) için, iki kullanıcının (k ve k') yaptığı, her bir c ölçütü için toplamda m adet değerlendirmesi, $R(k, \ddot{u}) = (R_0, R_1, R_2, \dots, R_m)$ ve $R(k', \ddot{u}) = (R'_0, R'_1, R'_2, \dots, R'_m)$ şeklinde ifade edilirse, bu iki kullanıcı arasındaki uzaklık $d(R(k, \ddot{u}), R(k', \ddot{u}))$, Denklem 3.8, Denklem 3.9 ve Denklem 3.10'da formülleri sunulan uzaklık metrikleri yardımıyla hesaplanabilir:

- Manhattan uzaklığı:

$$\sum_{c=0}^m |R_c(k, \ddot{u}) - R_c(k', \ddot{u})| \quad (3.8)$$

- Öklid uzaklığı:

$$\sqrt{\sum_{c=0}^m |R_c(k, \ddot{u}) - R_c(k', \ddot{u})|^2} \quad (3.9)$$

- Chebyshev uzaklığı:

$$\max_{c=0, \dots, m} |R_c(k, \ddot{u}) - R_c(k', \ddot{u})| \quad (3.10)$$

İki kullanıcı arasındaki nihai uzaklık ise, iki kullanıcının ortak olarak değerlendirdiği bütün ürünler kullanılarak aşağıdaki gibi hesaplanır:

$$uzaklık(k, k') = \frac{1}{|I(k, k')|} \sum_{\ddot{u} \in I(k, k')} d(R(k, \ddot{u}), R(k', \ddot{u})) \quad (3.11)$$

Kullanıcılar arasındaki uzaklık değerleri ne kadar yüksek olursa, birbirlerine olan benzerlikleri o kadar düşük olur. Yani kullanıcıların birbirlerine olan benzerliği ile uzaklıkları ters orantılıdır. *OF* teknikleri kullanıcılar arasındaki uzaklık metriğini değil benzerlik metriğini kullandığı için, hesaplanan uzaklık değerlerinin, benzerlik değerlerine dönüştürülmesi gerekmektedir. Bu dönüşüm Denklem 3.12'de sunulan şekilde sağlanır:

$$benz(k, k') = \frac{1}{1 + uzaklık(k, k')} \quad (3.12)$$

3.1.2. Model-tabanlı yaklaşımlar

Geleneksel *OF*'nin model-tabanlı yaklaşımları incelendiğinde, öneri üretebilmek amacıyla sistemde bulunan verilerden bir model oluşturulması gerektiği söylenebilir. Çoklu-ölçütlü sistemler için önerilen çoğu yaklaşım bu kategoride yer almaktadır. Toplama fonksiyonu tabanlı yaklaşım, olasılıklı modelleme yaklaşım ve çoklu-doğrusal *TDA* popüler olanlarındandır.

Çoklu-ölçütlü *OF* teknikleri incelendiğinde, genel beğenilerin içerisinde saklı olan ve kullanıcıların karakteristik özelliklerinin belirlenmesini sağlayabilen gizli örüntüler bulunmaktadır. Bu örüntüleri belirlemek amacıyla, kullanıcıların ürünler

hakkındaki genel beğenisini ifade eden genel değerlendirmeleri ile her bir alt-ölçüt için yapmış oldukları değerlendirmeler arasında ilişki bulunduğu varsayılarak, bir ilişki fonksiyonu tanımlanabilmektedir. Örneğin bir film tavsiye sistemi düşünüldüğünde, bazı kullanıcıların genel beğenisi ile senaryo alt-ölçütü için yapmış olduğu değerlendirme, diğer alt-ölçütlere kıyasla daha fazla benzerlik gösterebilmektedir. Dolayısıyla bu kullanıcılara senaryo alt-ölçütü için yapılan öneri ile genel beğeni için yapılan öneri benzerlik göstermelidir. Daha teknik bir ifadeyle bu ilişki fonksiyonu Denklem 3.13'de tanımlanmıştır [42]:

$$R_0 = f(R_1, R_2, \dots, R_m) \quad (3.13)$$

Bu fonksiyonda, R_0 , kullanıcının bir ürünün genel beğeni ölçütü için yapmış olduğu değerlendirmeyi, R_i ($i = 1, 2, \dots, m$) ise kullanıcının sırasıyla m adet alt-ölçüt için yapmış olduğu değerlendirmeyi ifade etmektedir. Bu fonksiyon, alan uzmanından yardım alınarak belirlenebilir. Bu en ilkel yöntemdir. Ek olarak, ölçütlerin ortalaması alınarak belirlenebilir. Son yıllarda bu konuda yapılan çalışmalarla, istatistiksel yöntemler ve makine öğrenmesi gibi veri madenciliği teknikleriyle bu fonksiyon daha doğru bir şekilde belirlenebilmektedir. Doğrusal regresyon yöntemi ile ölçütlerin ağırlıkları tespit edilerek ve bu ağırlıklardan faydalanarak genel beğeni değerini aşağıdaki gibi hesaplayabilmek mümkündür:

$$Rr_0 = w_1R_1 + w_2R_2 + \dots + w_mR_m + s \quad (3.14)$$

Denklem 3.14'de sunulan formülde, w_i ($i = 1, 2, \dots, m$), bir alt-ölçütün, genel beğeni ölçütü üzerindeki ağırlığını, s değeri ise veri kümesinden elde edilen sabit bir değeri ifade etmektedir. Dolayısıyla, her bir alt-ölçütün genel beğeni ölçütü üzerindeki etkisi belirlenip matematiksel olarak bir ilişki fonksiyonu ile ifade edilir. Daha sonra tek-ölçüt tabanlı sistemlerde kullanılan herhangi bir öneri üretme tekniği ile her bir alt-ölçüt için öneriler üretilir. Üretilen bu öneriler ilişki fonksiyonu ile ağırlıklandırılıp, genel beğeni ölçütü için bir öneri üretilmesi sağlanır.

3.2. İkili-Veri İçin Kullanılan Benzerlik Metrikleri

Bazı öneri sistemlerinde, kullanıcıların, ürünler için yapmış oldukları değerlendirmeler ikili-veri formatında olabilmektedir. Bu şekilde bir değerlendirme

yelpazesine sahip olan sistemin kullanıcı-ürün matrisi düşünüldüğünde, bir kullanıcının ürünler için yapmış olduğu değerlendirmeler, ikili-verilerden oluşan yatay bir vektör olarak düşünülebilir. Benzer şekilde bir ürün için, bütün kullanıcıların yapmış olduğu değerlendirmeler, ikili verilerden oluşan dikey bir vektör olarak düşünülebilir. Kullanıcılar ve ürünler arasındaki benzerlikler bu vektörler ve bazı benzerlik metrikleri ile hesaplanabilir. Tablo 3.1’de ikili verilerden oluşan örnek kullanıcı-ürün matrisi sunulmaktadır.

Tablo 3.1. İkili Verilerden Oluşan Örnek Kullanıcı-Ürün Matrisi

	Sihirbaz	Adalet	Kaçış	Düzenbaz	Başlangıç
Ali	1	0	0	0	1
Veli	0	-	0	-	1
Mehmet	1	0	-	0	1
Hasan	0	-	1	-	0
Ayşe	1	0	0	0	-

Tablo 3.1’de sunulan örnek kullanıcı-ürün matrisinde, beş kullanıcının, beş film için yapmış olduğu ikili verilerden oluşan değerlendirmeler sunulmuştur. Ali’nin, beş film için yapmış olduğu değerlendirmeler Denklem 3.15’de gösterildiği gibi yatay bir vektör olarak, Sihirbaz filmi için beş kullanıcının yapmış olduğu değerlendirmeler ise Denklem 3.16’da gösterildiği gibi dikey bir vektör olarak ifade edilebilir.

$$R_{Ali} = [1 \quad 1 \quad - \quad 0 \quad - \quad 0] \quad (3.15)$$

$$R_{Sihirbaz} = \begin{bmatrix} 1 \\ 0 \\ - \\ 1 \\ 0 \\ - \end{bmatrix} \quad (3.16)$$

İkili verilerden oluşan vektörler arasındaki benzerliğin belirlenmesini sağlayan metrikler, sınıflandırma, kümeleme gibi birçok örüntü analizi problemlerinin çözümünde kullanılabilir. Biyoloji, etnoloji, taksonomi, jeoloji, kimya ve son yıllarda parmak izi, el yazısından karakter tanıma gibi birçok alanda kullanılan çok sayıda benzerlik veya uzaklık metriği bulunmaktadır [53].

Tablo 3.2’de iki örüntü veya nesne olan ve ikili verilerden oluşan i ve j vektörleri için, operasyonel taksonomi biriminin (OTB) tanımladığı benzerlik ve uzaklık değerleri sunulmuştur [54]. Bu tabloda, a değeri, i ve j vektörlerindeki pozitif eşleşmelerin, yani ikisinin de değerinin 1 olduğu eşleşmelerin sayısını (1, 1), b değeri, i vektöründe 1, j vektöründe 0 olan eşleşmelerin sayısını (1, 0), c değeri, i vektöründe 0, j vektöründe 1 olan eşleşmelerin sayısını (0, 1) ifade etmektedir. Son olarak, d değeri, i ve j vektörlerindeki negatif eşleşmeleri, yani ikisinin de değerinin 0 olduğu eşleşmelerin sayısını (0, 0) temsil etmektedir. Çapraz toplam olan $a + d$, i ve j vektörlerindeki uyumlu eşleşmelerin sayısını, $b + c$ ise i ve j vektörlerindeki uyumsuz eşleşmelerin sayısını temsil etmektedir.

Tablo 3.2. İkili-Veriden Oluşan i ve j Örnekleri İçin OTB Değerleri

$j \backslash i$	1 (Varlık)	0 (Yokluk)	Toplam
1 (Varlık)	$a = i \cdot j$	$b = \bar{i} \cdot j$	$a + b$
0 (Yokluk)	$c = i \cdot \bar{j}$	$d = \bar{i} \cdot \bar{j}$	$c + d$
Toplam	$a + c$	$b + d$	$n = a + b + c + d$

Bu tez kapsamında önerilen modellerde, kullanıcı ve ürün vektörleri arasındaki benzerlikleri tespit etme aşamasında kullanılan ikili-veri benzerlik ölçütlerinin OTB tarafından tanımlanan değerlere göre formülleri Tablo 3.3’de sunulmaktadır [53].

Tablo 3.3. Kullanılan Benzerlik Metrikleri ve Formülleri

Metrik	Formül
Jaccard	$\frac{a}{a + b + c}$
Czekanowski	$\frac{2 \times a}{2 \times a + b + c}$
Simpson	$\frac{a}{\min(a + b, a + c)}$
Kulczynski	$\frac{a}{b + c}$
Johnson	$\frac{a}{a + b} + \frac{a}{a + c}$

3.3. Basit Bayes Sınıflandırıcı

İsmi İngiliz matematikçi Thomas Bayes'ten alan *BBS*, örüntü tanıma problemine kısıtlayıcı görülen bir önerme ile kullanılabilen olasılıkçı bir yaklaşımdır. Örüntü tanımada kullanılacak bütün nitelik ya da parametrelerin istatistiksel açıdan birbirinden bağımsız olma gerekliliği vardır. Veri madenciliği, örüntü tanıma ve makine öğrenmesi gibi disiplinler içinde yer alan başarılı bir sınıflandırma algoritmasıdır. Mevcut sınıflandırılmış durumdaki örnek veriler kullanılarak sisteme sonradan dâhil olan yeni bir verinin mevcut sınıflardan herhangi birine ait olma olasılığını hesaplar. *BBS*, Bayes teoreminin bağımsızlık önermesi ile basitleştirilmiş şeklidir. Bayes teoreminin denklemi, Denklem 3.17'de sunulmuştur:

$$P(A | B) = \frac{P(B | A)P(A)}{P(B)} \quad (3.17)$$

$P(A | B)$; B olayının gerçekleştiği durumda A olayının meydana gelme olasılığıdır.

$P(B | A)$; A olayının gerçekleştiği durumda B olayının meydana gelme olasılığıdır.

$P(A), P(B)$; Sırasıyla A ve B olaylarının önsel olasılıklarıdır.

Her ürünün/örneğin n adet özniteliğe sahip olduğu bir veri kümesi üzerinde çalışıldığı düşünüldüğünde ve hangi sınıfa ait olduğu bilinmeyen yeni bir ürün/örnek sınıflandırılmak istendiğinde, Bayes denklemi kullanılarak, örneğin/ürünün verilen m sınıfına ait olma olasılığı Denklem 3.18 yardımıyla hesaplanabilir.

$$p(\text{sınıf}_j) \prod_i^n p(f_i | \text{sınıf}_j) \quad (3.18)$$

$p(\text{sınıf}_j)$ ve $p(f_i | \text{sınıf}_j)$, veri kümesi kullanılarak hesaplanabilir. Bir örneğin/ürünün hangi sınıfa ait olduğunu tespit edebilmek için sistemde tanımlı olan her bir sınıf için bu değer hesaplanır. Daha sonra, en yüksek olasılığa sahip olan sınıf, o ürünün/örneğin ait olduğu sınıftır çıkarımında bulunulur.

Miyahara ve Pazzani [33] yaptıkları çalışmada, *OF* yaklaşımı amacı ile boşluklu-yapı veri modeli şeklinde adlandırdıkları bir *BBS* modeli geliştirmişlerdir. Bu modelde aktif kullanıcı dışında kalan kullanıcılar özniteliklerle, bu kullanıcıların ürünlere yaptıkları değerlendirmeler öznitelik verileriyle ilişkilendirilerek, aktif kullanıcıya hedef ürün için *BBS* algoritması ile öneri üretilmesi sağlanmıştır. Sayısal bir değer formatında bir öneri üretmek yerine, kullanıcı-ürün matrisindeki değerlendirmeleri ikili-veri formatında kullanarak bu iki sınıf için olasılık değerlerini hesaplayıp aktif kullanıcının öneri beklediği hedef ürün için sınıflandırma yapmışlardır. Bunu yaparken kullandıkları değerlendirmeler, aktif kullanıcı ve diğer kullanıcıların ortak olarak değerlendirdikleri ürünler olmuştur. En yüksek olasılığa sahip sınıfın hesaplanması işlemi Denklem 3.19’da sunulmuştur:

$$\begin{aligned}
Sınıf &= \operatorname{argmax}_{sınıf_j \in \{1,0\}} p(sınıf_j \mid K_1 = 1, K_3 = 0, \dots, K_n = 1) \\
&= \operatorname{argmax}_{sınıf_j \in \{1,0\}} p(K_1 = 1, K_3 = 1, \dots, K_n) \\
&= 1 \mid sınıf_j) p(sınıf_j) \\
&= \operatorname{argmax}_{sınıf_j \in \{1,0\}} p(sınıf_j) \prod_i p(K_i = sınıf_k \mid sınıf_j)
\end{aligned} \tag{3.19}$$

Örnek olarak, ikili-verilerden oluşan bir kullanıcı-ürün matrisi Tablo 3.4’de sunulmuştur. Bu matris gözlemlendiğinde, $p(Kullanıcı_1 = 1 \mid 1) = 1/3$, koşullu olasılığı 0.33 olarak hesaplanmaktadır. Boşluklu yapıda olabilen kullanıcı-ürün matrislerinde, ortak değerlendirilen ürün sayısı çok az miktarda olabilir veya hiç olmayabilir. Bu durum, koşullu olasılık değerlerinin “0” değerini almasına yol açabilir. Bu tip olumsuzlukların üstesinden gelebilmek için Laplace değerlendiricisi kullanılmaktadır. İkili-veriye dayanan sistemlerde, iki adet birbirinden farklı sınıf bulunduğu için paya 1, paydaya 2 eklenerek bu olumsuzluk giderilmiş olur. Bu örnekte $p(Kullanıcı_1 = 1 \mid 1) = (1+1)/(3+2) = 0.4$ olarak hesaplanır.

Tablo 3.4. İkili-Verilerden Oluşan Örnek Kullanıcı-Ürün Matrisi

	Ü ₁	Ü ₂	Ü ₃	Ü ₄	Ü ₅
K ₁	1	0	0		1
K ₂	0			1	0
K ₃		1	0		1
Sınıf Etiketleri	1	1	1	0	?

Tez kapsamında yapılan çalışmalarda, başarılı bir sınıflandırma algoritması olan *BBS*, *OF* yaklaşımı amacıyla çoklu-ölçütlü sistemlerde uygulanmıştır. Sayısal

değerlendirme yelpazesine sahip çoklu-ölçütlü sistemlerin sağladığı bazı avantajlar, ikili-veriye göre uyarlanıp, daha başarılı komşular seçilmesini sağlayan yeni modeller önerilerek, genel beğeni ölçütü için *BBS* algoritması ile üretilen önerilerin doğruluğu arttırılmıştır.

3.4. Uyumluluk Tekniği

Çoklu-ölçüt tabanlı sistemlerde, her bir alt-ölçüt, kullanıcıların ürünler hakkındaki genel beğenisi üzerinde farklı bir etkiye sahiptir. Örneğin, bir film öneri sistemi düşünülürse, bir kullanıcı için bir film hakkındaki genel beğenisi üzerinde en çok etkisi olan alt-ölçüt senaryo olurken, bir başka kullanıcı için görsel efekt olabilmektedir. Alt-ölçütlerin, her bir kullanıcının genel beğenisi üzerindeki etkisini belirlemek gerekir. Daha açık bir ifadeyle, bir kullanıcının her bir alt-ölçüt için yapmış olduğu değerlendirmeler ile genel ölçüt için yapmış olduğu değerlendirmeler arasındaki uyumluluğun tespit edilmesi gerekmektedir. Bu tespit, kullanıcılar açısından daha kişiselleştirilmiş bir sistem tasarlanmasını sağlayarak *OF* yaklaşımında çok önemli bir aşama olan kullanıcılar arasındaki benzerliklerin belirlenmesi noktasında sisteme olumlu katkı yapacaktır.

Her bir kullanıcı için, alt-ölçütlerin genel ölçüt üzerindeki etkisi, her bir alt-ölçüt ile genel ölçüt için yapmış olduğu uyumlu değerlendirme çiftleri gözlemlenerek belirlenebilir [55]. Daha teknik ifade edilecek olursa sayısal bir değerlendirme yelpazesinde, (*a*) kullanıcısının bir ürün (*ü*) için yapmış olduğu değerlendirme ($R_{a,ü}$) ile bu kullanıcının yapmış olduğu değerlendirmelerin ortalamasının farkı $f_{a,ü} = R_{a,ü} - \overline{R}_a$ şeklinde ifade edilebilir. Aynı ürün (*ü*) için iki farklı kullanıcının (*a* ve *b*) yapmış olduğu değerlendirmeler ($R_{a,ü}$ ve $R_{b,ü}$) şeklinde ifade edilirse, yapılan bu değerlendirmeler;

Eğer $f_{a,ü} > 0$ ve $f_{b,ü} > 0$ veya $f_{a,ü} < 0$ ve $f_{b,ü} < 0$ sağlanırsa uyumlu,

Eğer $f_{a,ü} > 0$ ve $f_{b,ü} < 0$ veya $f_{a,ü} < 0$ ve $f_{b,ü} > 0$ sağlanırsa uyumsuz,

Eğer $f_{a,ü} = 0$ ve $f_{b,ü} = 0$ veya $R_{a,ü}$ değeri boş (ürün değerlendirilmemiş) veya $R_{b,i}$ değeri boş ise bu değerlendirmeler bağlı olarak tanımlanabilir.

İki kullanıcı arasındaki benzerlik, uyumluluk tekniklerini kullanan Somers' benzerlik metriği ile hesaplanabilmektedir [56]. İki kullanıcının (a, b) değerlendirdiği toplam ürün çifti sayısı (L) , uyumlu çiftlerin sayısı (C) , uyumsuz çiftlerin sayısı (D) ve bağlı çiftlerinin sayısı (T) olarak tanımlandığında, Somers' metriğinin (d) formülü Denklem 3.20'de sunulmaktadır:

$$d_{a,b} = \frac{C - D}{L - T} \quad (3.20)$$

Uyumluluk tekniği, ikili verilerden oluşan çoklu-ölçütlü tabanlı sistemlerde, kullanıcıların ürünler hakkındaki genel beğenisi üzerinde hangi alt-ölçütün ne kadar öneme sahip olduğunu belirlemek amacıyla kullanılabilir. Bu yaklaşım daha kişiselleştirilmiş bir sistem oluşturulmasını sağlar. Bir kullanıcının, bir ürünün farklı iki ölçütü için (ö1, ö2) yapmış olduğu değerlendirmeler, $R_{\text{ö1}}$ ve $R_{\text{ö2}}$ ile temsil edildiğinde; Eğer, $R_{\text{ö1}} = 1$ ve $R_{\text{ö2}} = 1$ veya $R_{\text{ö1}} = 0$ ve $R_{\text{ö2}} = 0$ şartı sağlanır ise, bu değerlendirmeler uyumludur denilebilir. Kullanıcının bütün ürünler için yapmış olduğu değerlendirmeler gözlemlenerek, her bir alt ölçüt ile genel beğeni ölçütü arasındaki uyumlu değerlendirme çiftlerinin sayıları belirlenir. Örnek ikili-verilerden oluşan çoklu-ölçütlü film öneri sistemi Tablo 3.4'de sunulmuştur. Bu tabloda, bir kullanıcının beş adet filmin dört alt-ölçütü için ve filmin geneli için yapmış olduğu beğeni değerlendirmeleri bulunmaktadır.

Tablo 3.5. Bir Kullanıcının Beş Film için Beğeni Değerleri

	Senaryo	Oyunculuk	Yönetmen	Görsellik	Genel
Film 1	1	0	0	0	1
Film 2	1	0	0	0	1
Film 3	0	1	1	1	0
Film 4	0	1	1	1	0
Film 5	1	1	1	0	1

Bu örnekteki kullanıcının, her bir alt-ölçüt ve genel ölçüt için yaptığı değerlendirmelerin uyumluluğu gözlemlendiğinde, her bir filmin "senaryo" alt-ölçütü için yapmış olduğu değerlendirme ile genel değerlendirmenin uyumlu olduğu ve bu değerlendirme çifti sayısının toplamda beş olduğu görülmektedir. Benzer şekilde, "oyunculuk" ve "yönetmen" alt-ölçütleri için birer uyumlu çift vardır ve "görsellik" alt-ölçütü için hiç uyumlu çift bulunmamaktadır. Dolayısıyla, bu kullanıcı için filmleri değerlendirirken, o ürün hakkındaki genel düşüncesi üzerinde en çok

etkisi olan alt-ölçütün “senaryo” olduğu, en az etkisi olan alt-ölçütün ise “görsellik” olduğu sonucu çıkarılabilir. Bu dağılım her kullanıcı için farklılık gösterebilir. Daha başarılı kişiselleştirilmiş sistemler oluşturabilmek için kullanıcılar arasındaki benzerlikler belirlenirken, her kullanıcı için bu analizin yapılması ve kullanıcıların eğilimlerinin dikkate alınması gerekir.

Aktif kullanıcının, her bir ölçütün genel ölçütle uyumlu değerlendirme çiftleri belirlendikten sonra normalleştirilmesi gerekmektedir. Genel ölçüte ek olarak c adet alt-ölçütün bulunduğu bir sistem düşünüldüğünde, birinci alt-ölçüt ile genel ölçüt arasındaki uyumlu çiftlerin sayısının n_1 , ikinci alt-ölçüt ile genel ölçüt arasındaki uyumlu çiftlerin sayısı n_2 ile temsil edildiğinde, t 'inci ölçüt için normalleştirilmiş uyumluluk değeri (N_t) Denklem 3.21'de sunulan formül ile hesaplanabilir.

$$N_t = \frac{n_t}{\sum_{i=1}^c n_i}, t = 1, 2, \dots, c \quad (3.21)$$

Bu tez kapsamında geliştirilen benzerlik hesaplama modellerinde, çoklu-ölçüt tabanlı sistemlerdeki geleneksel benzerlik hesaplama yöntemleri olan ortalama ve en küçük benzerlik tekniklerinde, aktif kullanıcı ile diğer kullanıcıların benzerlikleri, aktif kullanıcının ölçütler arasındaki uyumluluğu ile ağırlıklandırılarak daha etkili ve başarılı komşulukların seçilmesi sağlanmıştır.

4. BASİT BAYES SINIFLANDIRICI TABANLI İKİLİ-VERİ ÇOKLU-ÖLÇÜTLÜ ÖNERİ SİSTEMLERİ

OF yaklaşımlarında, kullanıcılara ürünler hakkında öneriler üretme aşamasında benzer kullanıcıların/ürünlerin doğru olarak belirlenmesi kritik bir öneme sahiptir. Kullanıcıların ürünler hakkındaki değerlendirmelerini, genel bir beğeniye ek olarak birden fazla ölçüt için yapabilmesini sağlayan *ÇÖÖS*'de, kullanıcıların düşünceleri daha kişiselleştirilmiş bir şekilde toplanarak, kullanıcılar/ürünler arasındaki benzerliklerin daha etkili biçimde belirlenmesi sağlanabilmektedir. Bu durum, *OF* algoritmalarının bu sistemler üzerinde daha etkin bir biçimde uygulanabilmesini sağlayabilmektedir.

OF algoritmaları gözlemlendiğinde, bir aktif kullanıcı için hedef ürüne öneri üretme aşamasında, bu algoritmaları bütün veri seti üzerinde uygulamak yerine, belirli tekniklerle eğitilmiş olan aktif kullanıcıya en benzer kullanıcıların yapmış oldukları değerlendirmeleri içeren eğitilmiş bir veri setini kullanmak başarıyı yüksek öneriler üretilmesini sağlamaktadır. Benzer şekilde, eğitilmiş veri seti, ürünler arasındaki benzerlikler doğrultusunda da oluşturulabilir.

Bu çalışmada, ikili verilerden oluşan çoklu-ölçütlü bir *OF* sisteminde, genel beğeni ölçütü için *BBS* algoritması ile öneri üretme aşamasında kullanıcılar arasındaki benzerliklerin daha etkili hesaplanabilmesini sağlayan çeşitli modeller geliştirilmiştir. Bu benzerlik modelleri ile kullanıcılara genel beğeni ölçütü için *BBS* ile bir öneri üretilirken, bütün veri setini kullanmak yerine, her bir kullanıcı/ürün için farklı ve eğitilmiş bir veri setinin kullanılması sağlanmıştır.

İlk olarak, sayısal bir değerlendirme yelpazesine sahip çoklu-ölçüt tabanlı sistemlerdeki kullanıcılar/ürünler arasındaki benzerliklerin daha doğru belirlenmesini sağlayan geleneksel metotlar, ikili-veri kullanımına göre uyarlanmıştır. Eğitilmiş veri setinin, kullanıcı-tabanlı benzerlik (*KB*) olarak adlandırılan kullanıcılar arasındaki benzerliklere göre ve ürün-tabanlı benzerlik (*ÜB*) olarak adlandırılan ürünler arasındaki benzerliklere göre seçilmesini sağlayan iki model geliştirilmiştir.

Ayrıca, daha başarılı kişiselleştirme sağlamak amacıyla, her bir kullanıcı için hangi alt-ölçütün kişilerin ürünlerin genel beğenisi üzerinde ne kadar etkili ve önemli olduğu uyumluluk tekniği ile belirlenmiştir. Bu uyumluluk tekniği, önerilen *KB* ve *ÜB* modelleriyle beraber kullanılarak, daha başarılı komşulukların seçilmesini hedefleyen “Uyumluluk Tekniği ile Kullanıcı-Tabanlı Benzerlik (*U-KB*)” ve “Uyumluluk Tekniği ile Ürün-Tabanlı Benzerlik (*U-ÜB*)” adıyla iki model daha sunulmuştur.

Son olarak, kullanıcılara hedef ürünler için *BBS* ile öneri üretme aşamasında kullanıcı-tabanlı benzerlik yaklaşımları ile ürün-tabanlı benzerlik yaklaşımlarının beraber kullanıldığı karma yaklaşımlar sunulmuştur. Bu yaklaşımda, komşuluk seçme aşamasında hem kullanıcı hem de ürün-tabanlı benzerlik modelleri beraber kullanılarak üretilen önerilerin doğruluğunun iyileştirilmesi amaçlanmıştır.

4.1. Kullanıcı-Tabanlı Benzerlik

İkili verilerden oluşan ve genel beğeni ölçütüne ek olarak *c* adet alt-ölçütün bulunduğu çoklu-ölçüt tabanlı bir *OF* sisteminde, bir kullanıcının (*k*), bir ölçüt bazında, sistemde bulunan bütün ürünler ($\{u_1, u_2, \dots, u_n\}$) için yapmış olduğu değerlendirmeler ($\{d_1, d_2, \dots, d_n\}$) ile temsil edildiği düşünüldüğünde, bir kullanıcının bütün ürünlerin tek bir ölçütü için yapmış olduğu değerlendirmeleri içeren vektör Denklem 4.1’de sunulmuştur.

$$R(k) = [d_1, d_2, \dots, d_n] \quad (4.1)$$

Sistemde genel beğeni değerlendirme ölçütüne ek olarak *c* adet alt-ölçüt bulunduğu için, kullanıcı-ürün matrisinde, her bir kullanıcının bütün ürünler için toplamda *c*+1 ölçüt bazında yapmış olduğu değerlendirme bulunmaktadır. Dolayısıyla her bir kullanıcı toplamda *c*+1 değerlendirme vektörüne sahiptir ($R(k)_0, R(k)_1, R(k)_2, \dots, R(k)_c$).

Kullanıcı-ürün matrisinde bulunan bütün değerlendirmeler ikili formatta (1-0) olduğundan dolayı, bu vektörler ikili verilerden oluşan yatay bir vektör olarak düşünülebilir. Bu vektörler arasındaki benzerlik aynı zamanda kullanıcılar

arasındaki benzerlikleri de ifade etmektedir. Dolayısıyla, bir kullanıcı ile diğer kullanıcı arasındaki benzerlik bu vektörler arasında ikili benzerlik ölçütleri kullanılarak belirlenebilir.

Genel beğeni değerlendirme ölçütüne ek olarak c adet alt-ölçüt bulunduğu için toplamda $c+1$ adet benzerlik değeri hesaplanabilir. Genel beğeni değerlendirme ölçütü için aktif kullanıcı ile diğer kullanıcı arasındaki benzerlik değeri $benzerlik_0(k, k')$, birinci alt-ölçüt için olan benzerlik değeri $benzerlik_1(k, k')$, benzer şekilde c 'inci alt-ölçüt için olan benzerlik değeri $benzerlik_c(k, k')$ ile ifade edilirse toplamda $c+1$ adet benzerlik değeri elde edilmiş olur. İki kullanıcı arasındaki nihai benzerlik ise, her bir ölçüt için ayrı ayrı hesaplanmış olan benzerlik değerlerini bir araya getirerek belirlenebilir. Bu işlem çoklu-ölçütlü OF sistemlerinde kullanılan iki farklı teknik ile belirlenebilir. Bunlardan ilki, her bir ölçüt için ayrı ayrı hesaplanan benzerlik değerlerinin ortalamasının alınmasıdır. Bu yaklaşım Denklem 4.2'de sunulan formül yardımıyla gerçekleştirilebilir.

$$benz_{ort}(k, k') = \frac{\sum_{i=0}^c benzerlik_i(k, k')}{c + 1} \quad (4.2)$$

Diğer bir teknik ise, her bir alt-ölçüt için hesaplanan benzerlik değerlerinden en küçüğünü, kullanıcılar arasındaki nihai benzerlik değeri olarak kabul etmektir. Bu yaklaşım Denklem 4.3'de sunulan formül yardımıyla gerçekleştirilebilir.

$$benz_{min}(k, k') = \min_{i=0,1,2,\dots,c} benzerlik_i(k, k') \quad (4.3)$$

4.2. Ürün-Tabanlı Benzerlik

OF yaklaşımlarındaki komşuluk seçimi, kullanıcılar arasındaki benzerliklere göre olduğu gibi, hedef ürün ile diğer ürünler arasındaki benzerliklere göre de belirlenebilir. İkili-verilerden oluşan ve genel beğeni ölçütüne ek olarak c adet alt-ölçütün bulunduğu çoklu-ölçüt tabanlı bir OF sisteminde, bir ürünün (\bar{u}), bir ölçüt için, sistemi kullanan bütün kullanıcılar (k_1, k_2, \dots, k_n) tarafından yapılmış olan değerlendirmeler (d_1, d_2, \dots, d_n) ile temsil edildiği düşünüldüğünde, bir kullanıcının bütün ürünlerin tek bir ölçütü için yapmış olduğu değerlendirmeleri içeren vektör Denklem 4.4'de sunulmaktadır.

$$R(\ddot{u}) = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} \quad (4.4)$$

Sistemde, genel beğeni değerlendirme ölçütüne ek olarak toplamda $c+1$ ölçüt bulunduğu için, her bir ürün için bütün kullanıcılar $c+1$ ölçüt için değerlendirme yaparlar. Dolayısıyla her bir ürün toplamda $c+1$ değerlendirme vektörüne sahiptir ($R(\ddot{u})_0, R(\ddot{u})_1, R(\ddot{u})_2, \dots, R(\ddot{u})_c$).

Kullanıcıların ürünler için yapmış oldukları değerlendirmeler ikili-formatta olduğu için, kullanıcıların hedef ürün için yapmış oldukları değerlendirmeleri ifade eden bu vektörler dikey formatta ikili-verilerden oluşan bir vektör olarak düşünülebilir. Bu vektörler arasındaki benzerlikler ikili benzerlik ölçütleri yardımıyla hesaplanabilir. Hesaplanan bu benzerlik aynı zamanda iki ürünün birbirlerine olan benzerliğini ifade etmektedir. Genel beğeniye ek olarak toplamda $c+1$ adet ölçüt bulunduğu için her bir ölçüt için bu vektörler arasındaki benzerlikler hesaplanarak toplamda $c+1$ adet benzerlik değeri hesaplanabilir. Genel beğeni değerlendirme ölçütü için hedef ürün ile diğer ürün arasındaki benzerlik değeri $benzerlik_0(\ddot{u}, \ddot{u}')$, birinci alt-ölçüt için olan benzerlik değeri $benzerlik_1(\ddot{u}, \ddot{u}')$, benzer şekilde c 'inci alt-ölçüt için olan benzerlik değeri $benzerlik_c(\ddot{u}, \ddot{u}')$ ile ifade edilirse toplamda $c+1$ adet benzerlik değeri elde edilmiş olur.

Hedef ürün ile diğer ürün arasındaki nihai benzerlik değeri ise, her bir alt-ölçüt için ayrı ayrı hesaplanan bu benzerlik değerlerinin bir araya getirilmesi ile hesaplanır. Bu birleştirme iki şekilde olabilir. İlki, ayrı ayrı hesaplanan benzerlik değerinin ortalaması hesaplanarak yapılır. Bu yaklaşım Denklem 4.5'de sunulan formül yardımıyla gerçekleştirilebilir.

$$benz_{ort}(\ddot{u}, \ddot{u}') = \frac{\sum_{i=0}^c benzerlik_i(\ddot{u}, \ddot{u}')}{c + 1} \quad (4.5)$$

Diğer bir teknik ise, her bir alt-ölçüt için ayrı ayrı hesaplanan benzerlik değerlerinden en küçük olan benzerliği, ürünler arasındaki nihai benzerlik değeri olarak belirlemektir. Bu yaklaşım Denklem 4.6'da sunulan formül yardımıyla gerçekleştirilebilir.

$$benz_{min}(ü, ü') = \min_{i=0,1,2,...,c} benzerlik_i(ü, ü') \quad (4.6)$$

Daha anlaşılır olması açısından, ikili veriye dayanan çoklu-ölçüt tabanlı *OF* sistemleri için geliştirilen bu komşuluk modellerinin adımları ayrıntılı bir şekilde Tablo 4.1'de sunulmuştur.

Tablo 4.1. *KB ve ÜB Modelleri için İşlem Adımları*

Adım No	KB	ÜB
1	Aktif kullanıcı ile diğer kullanıcılar arasındaki her bir ölçüt için benzerlik değerleri ikili benzerlik metrikleri yardımı ile hesaplanır.	Hedef ürün ile diğer ürünler arasındaki her bir ölçüt için benzerlik değerleri ikili benzerlik metrikleri yardımı ile hesaplanır.
2	Her bir ölçüt için ayrı ayrı hesaplanan benzerlik değerleri bir araya getirilerek kullanıcıların, aktif kullanıcı ile olan nihai benzerlik değerleri hesaplanır ($benz_{ort}$ veya $benz_{min}$).	Her bir ölçüt için ayrı ayrı hesaplanan benzerlik değerleri bir araya getirilerek ürünlerin, aktif ürün ile olan nihai benzerlik değerleri hesaplanır ($benz_{ort}$ veya $benz_{min}$).
3	Bütün kullanıcılar, aktif kullanıcıyla olan nihai benzerlik değerlerine göre ağırlıklandırılır.	Bütün ürünler, hedef ürünle olan nihai benzerlik değerlerine göre ağırlıklandırılır.
4	Aktif kullanıcıya en benzer belirli sayıdaki kullanıcı eğitilmiş veri seti (belirleyici) olarak seçilir.	Hedef ürüne en benzer belirli sayıdaki ürün eğitilmiş veri seti (belirleyici) olarak seçilir.

Önerilen bu modeller, çoklu-ölçütlü sistemlerin daha başarılı kişiselleştirme yetenekleri sayesinde, kullanıcılar arasındaki benzerliklerin daha doğru olarak hesaplanıp, daha etkili komşuluklar seçilebilmesine olanak tanımaktadır. BBS algoritması ile kullanıcılara genel beğeni ölçütü için öneri üretilirken aktif kullanıcı/hedef ürün dışında kalan bütün kullanıcıların/ürünlerin değerlendirmelerinin kullanılması yerine, *KB* veya *ÜB* ile belirli sayıda benzer kullanıcıları komşu olarak belirlenip bu kullanıcıların/ürünlerin değerlendirmelerini kullanmak üretilen önerilerin doğruluğunu arttırabilecektir.

4.3. Benzerlik Tabanlı Modellerin Uyumluluk Tekniği ile Kullanılması

Çoklu-ölçüt tabanlı sistemlerde kullanıcıların ürünleri farklı açılardan değerlendirebilmeleri, onlar hakkında daha kişiselleştirilmiş bilgilerin elde edilmesini sağlamaktadır. Kullanıcıların alt-ölçütler ile genel beğeni ölçütü için

yapmış oldukları değerlendirmeler arasındaki ilişkiler uyumluluk tekniği ile analiz edilerek belirlenebilir. Bu ilişki, kullanıcıların ürünleri değerlendirirken hangi alt-ölçütün onlar açısından ne kadar önemli olduğunun belirlenmesini sağlayarak daha kişiselleştirilmiş bir sistem oluşturulmasını sağlar. Her bir kullanıcının alt-ölçütler için olan farklı eğilimleri, komşuluk belirlenmesi aşamasında kullanılabilir.

İlk olarak, aktif kullanıcı için hangi alt-ölçütün o kullanıcının ürünler hakkındaki genel beğenisi üzerinde ne kadar etkili ve önemli olduğu uyumluluk tekniği ile her bir alt-ölçüt ve genel beğeni ölçütü arasındaki uyumlu çiftler belirlenerek normalleştirilir. Normalleştirilmiş bu uyumluluk değerleri hem *KB* yaklaşımında hem de *ÜB* yaklaşımında kullanıcılar/ürünler arasındaki benzerliklerin ağırlıklandırılmasında kullanılarak daha kişiselleştirilmiş benzerlik değerleri hesaplanabilmesini sağlar.

4.3.1. Uyumluluk tekniği ile kullanıcı-tabanlı benzerlik

Aktif kullanıcının her bir alt-ölçüt için normalleştirilmiş uyumluluk değerleri hesaplandıktan sonra, aktif kullanıcı ile diğer kullanıcıların ilgili alt-ölçüt için olan benzerlik değeri, aktif kullanıcının ilgili alt-ölçüt için olan normalleştirilmiş uyumluluk değeri ile ağırlıklandırılarak yeni bir benzerlik değeri hesaplanır. Bu işlem bütün alt-ölçütler için yapılır. Daha açık bir ifadeyle, aktif kullanıcı ile diğer kullanıcılar arasındaki her bir alt-ölçüt için olan benzerlik değerleri, aktif kullanıcının ilgili alt-ölçüt için olan normalleştirilmiş uyumluluk değeri ile ağırlıklandırılır. Bu durum, aktif kullanıcının alt-ölçütlere verdiği önemin, kullanıcılar arasındaki benzerliklerin belirlenmesinde kullanılmasını sağlayarak daha kişiselleştirilmiş bir sistem oluşturulmasını sağlar.

Daha teknik ifade edilecek olursa, aktif kullanıcının belirli bir t 'inci alt-ölçütü için normalize edilmiş uyumluluk değeri N_t olarak, aktif kullanıcı (k) ile diğer kullanıcı (k') arasındaki bu alt-ölçüt için olan benzerlik değeri $benz_t(k, k')$ ile ifade edilirse, kullanıcılar arasındaki bu alt-ölçüt için olan ağırlıklandırılmış benzerlik değeri Denklem 3.17'de sunulan formül ile hesaplanabilir.

$$benz(k, k') = benz_t(k, k') \times N_t \quad (4.7)$$

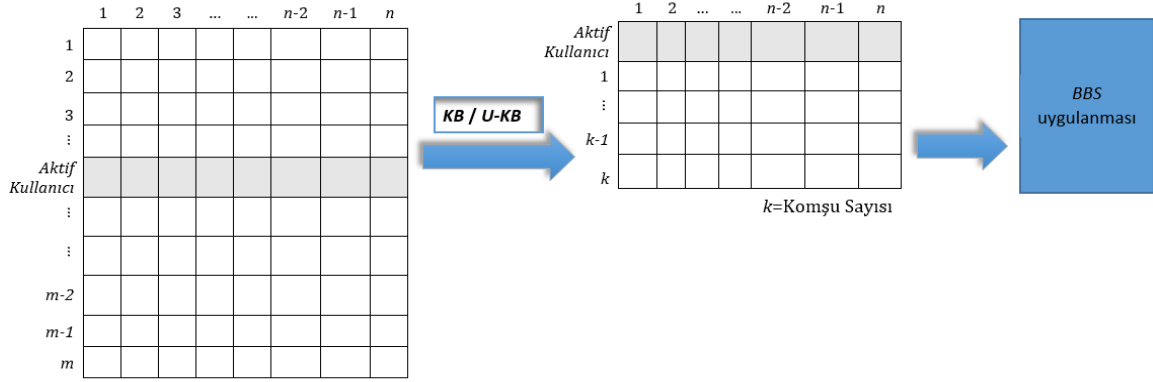
Aktif kullanıcı ile diğer kullanıcılar arasındaki her bir alt-ölçüt için hesaplanan benzerlik değerleri, aktif kullanıcının normalize edilmiş uyumluluk değerleri ile ağırlıklandırıldıktan sonra bu benzerlik değerlerini bir araya getirerek kullanıcılar arasındaki ağırlıklandırılmış nihai benzerlik değerlerini belirlemek gerekir. Aktif kullanıcı ile diğer kullanıcının genel beğeni ölçütüne göre hesaplanan benzerlik değeri $benz_0(k, k')$ ile ifade edilirse, kullanıcılar arasındaki ağırlıklandırılmış nihai benzerlik değeri iki şekilde hesaplanabilir. Bunlardan ilki, her bir alt-ölçüt için ayrı ayrı hesaplanan ağırlıklandırılmış benzerlik değerlerinin ortalaması alınarak yapılabilir;

$$benz_{ort}(k, k') = \frac{[\sum_{i=0}^c (benz_i(k, k'))] + benz_0(k, k')}{c + 1} \quad (4.8)$$

Kullanıcılar arasındaki ağırlıklandırılmış nihai benzerlik değeri, her bir alt-ölçüt için ayrı ayrı hesaplanan ağırlıklandırılmış benzerlik değerlerinden en küçük olanı seçilerek de belirlenebilir:

$$benz_{min}(k, k') = \min_{i=0,1,2,\dots,c} benz_i(k, k') \quad (4.9)$$

Kullanıcı-tabanlı benzerlik yaklaşımlarında (*KB* veya *U-KB*), aktif kullanıcı ile diğer kullanıcılar arasındaki benzerlikler, önerilen *KB* veya *U-KB* modelleri ile belirlendikten sonra belirli sayıda komşuluklar seçilerek kullanıcı-ürün matrisinin ($m \times n$) boyutu indirgenir. Daha sonra, aktif kullanıcıya en benzer kullanıcıların genel beğeni ölçütü için yapmış olduğu değerlendirmeler üzerinde *BBS* algoritması uygulanır. Kullanıcı-tabanlı benzerlik yaklaşımlarının (*KB* veya *U-KB*) çalışma mekanizması Şekil 4.1'de sunulmaktadır.



Şekil 4.1. Kullanıcı-Tabanlı Benzerlik Yaklaşımlarının Çalışma Mekanizması

4.3.2. Uyumluluk tekniği ile ürün-tabanlı benzerlik

Aktif kullanıcı için bir hedef ürüne öneri üretme aşamasında, aktif kullanıcının alt-ölçütler için olan normalleştirilmiş uyumluluk değerleri, $\bar{U}B$ yaklaşımında da kullanılabilir. Aktif kullanıcının her bir alt-ölçüt için olan normalleştirilmiş uyumluluk değerleri hesaplandıktan sonra, öneri üretilecek hedef ürünle diğer ürünler arasındaki benzerlikler ağırlıklandırılır. Bu işlem bütün alt-ölçütler için yapılır. Hedef ürün için yapılacak öneride, ürünler arasındaki benzerliklerinin aktif kullanıcının normalleştirilmiş uyumluluk değerleri ile ağırlıklandırmak, hedef ürüne benzer ürünlerin daha doğru olarak belirlenmesini sağlayacaktır.

Daha teknik ifade edilecek olursa, aktif kullanıcının belirli bir t 'inci alt-ölçütü için normalize edilmiş uyumluluk değeri N_t olarak, hedef ürün (\bar{u}) ile diğer ürün arasındaki bu alt-ölçüt için olan benzerlik değeri $benz_c(\bar{u}, \bar{u}')$ ile ifade edilirse, ürünler arasındaki bu alt-ölçüt için olan ağırlıklandırılmış benzerlik değeri Denklem 4.10'da sunulduğu gibi hesaplanabilir:

$$benz(\bar{u}, \bar{u}') = benz_t(\bar{u}, \bar{u}') \times N_t \quad (4.10)$$

Hedef ürün ile diğer ürünler arasındaki her bir alt-ölçüt için hesaplanan benzerlik değerleri, aktif kullanıcının normalize edilmiş uyumluluk değeri ile ağırlıklandırılır. Daha sonra, ağırlıklandırılan bu benzerlik değerleri bir araya getirilerek, ürünler arasındaki nihai benzerlik değeri belirlenir. Hedef ürün ile diğer ürün arasındaki genel beğeni ölçütüne göre olan benzerlik değeri $benz_0(\bar{u}, \bar{u}')$ ile

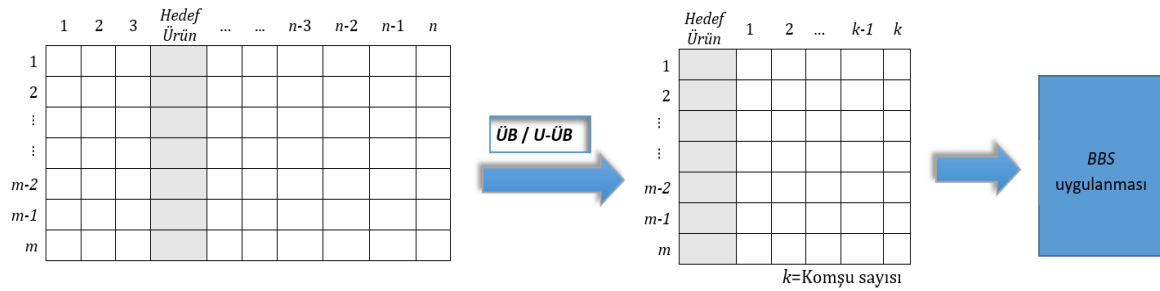
ifade edilirse, ürünler arasındaki nihai benzerlik değeri iki şekilde hesaplanabilir. İlk olarak, her bir alt-ölçüt için hesaplanan ürünler arasındaki ağırlıklandırılmış benzerlik değerlerinin ortalaması alınır;

$$benz_{ort}(\ddot{u}, \ddot{u}') = \frac{[\sum_{i=0}^c (benz_i(\ddot{u}, \ddot{u}'))] + benz_0(\ddot{u}, \ddot{u}')}{c + 1} \quad (4.11)$$

Bir diğer yöntem ise, ürünler arasındaki nihai benzerlik değerini, her bir alt-ölçüt için ayrı ayrı hesaplanan ağırlıklandırılmış benzerlik değerlerinden en küçük olanı olarak belirlemektir.

$$benz_{min}(\ddot{u}, \ddot{u}') = \min_{i=0,1,2,\dots,c} benz_i(\ddot{u}, \ddot{u}') \quad (4.12)$$

Ürün tabanlı benzerlik yaklaşımlarında ($\ddot{U}B$ veya $U-\ddot{U}B$), hedef ürün ile diğer ürünler arasındaki benzerlikler, önerilen $\ddot{U}B$ veya $U-\ddot{U}B$ modelleri ile belirlendikten sonra belirli sayıda komşuluklar seçilerek kullanıcı-ürün matrisinin ($m \times n$) boyutu indirgenir. Daha sonra, hedef ürüne en benzer ürünlerin genel beğeni ölçütü için yapılmış olan değerlendirmeler üzerinde BBS algoritması uygulanır. Ürün-tabanlı benzerlik yaklaşımlarının ($\ddot{U}B$ veya $U-\ddot{U}B$) çalışma mekanizması Şekil 4.2'de sunulmaktadır.



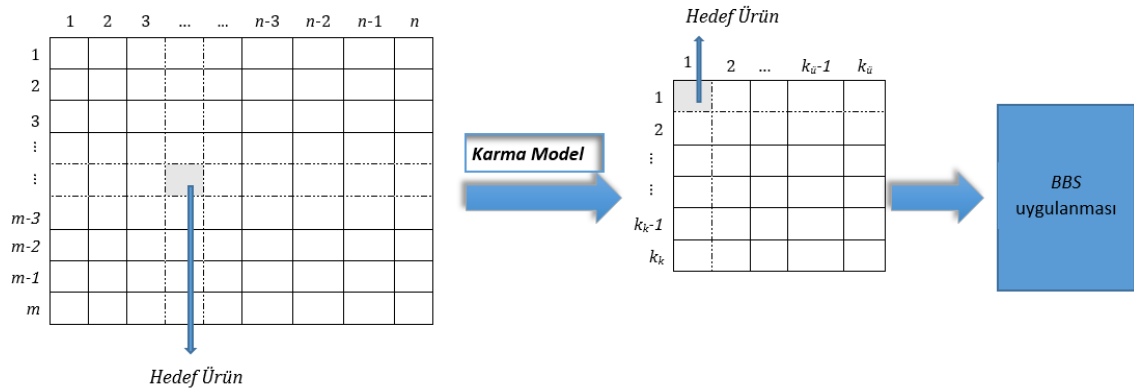
Şekil 4.2. Ürün-Tabanlı Benzerlik Yaklaşımlarının Çalışma Mekanizması

4.4. Karma Model

BBS algoritması ile OF yaklaşımında, bütün veri kümesini kullanmak yerine, önerilen kullanıcı/ürün tabanlı benzerlik modelleri yardımıyla, aktif kullanıcıya/hedef ürüne en benzer kullanıcıların/ürünlerin oluşturduğu daha küçük

veri kümesi kullanmak başarımı yüksek öneriler üretilmesini sağlayabilir. Bu aşamaya kadar önerilen modeller yardımıyla ve seçilecek komşuluk değerlerine göre, veri kümesi ya kullanıcılar arasındaki benzerliklere göre ya da ürünler arasındaki benzerliklere göre veri kümesinin daha düşük boyutlara indirgenmesi sağlanabilir. Kullanıcı-ürün matrisi düşünüldüğünde, bu indirgeme işlemi, kullanıcı-tabanlı benzerlik yaklaşımında (KB veya $U-KB$) sadece kullanıcılar arasında olur iken ürün-tabanlı benzerlik yaklaşımında ($ÜB$ veya $U-ÜB$) sadece ürünler arasında olur.

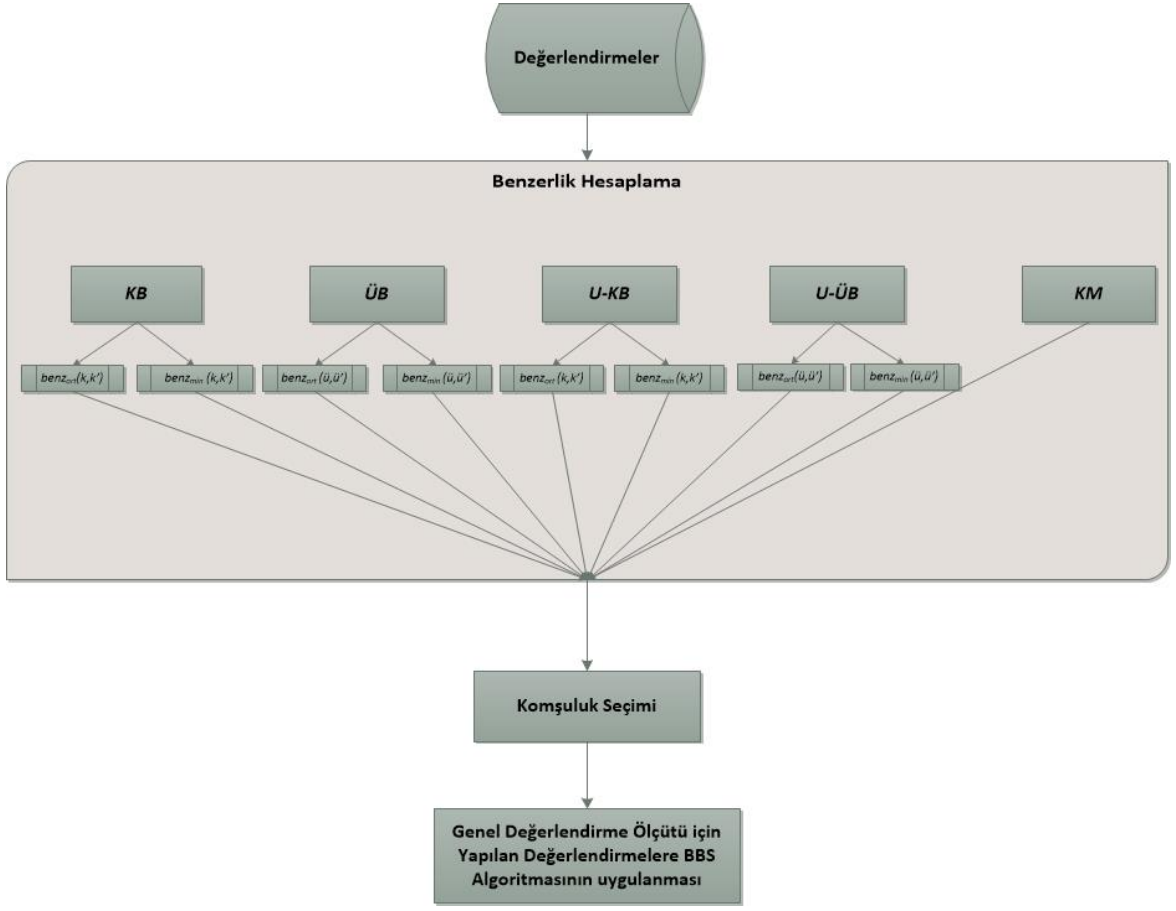
BBS algoritmasının başarımını arttırmak amacıyla, veri kümesi hem kullanıcılar hem de ürünler arasındaki benzerlikler doğrultusunda, benzer kullanıcıların ve ürünlerin beraber seçilmesi ile iki boyutta bir indirgenme yapılabilir. Hem kullanıcı hem de ürün-tabanlı benzerlik modellerinin beraber kullanılması ile iki boyutta indirgeme işlemi yapılabilir. Karma model yaklaşımının çalışma mekanizması Şekil 4.3'de sunulmaktadır.



Şekil 4.3. Karma Model Yaklaşımının Çalışma Mekanizması

Tez kapsamında yapılan ve Bölüm 5'de sunulan deneysel çalışmalar kısmında, önerilen ürün-tabanlı ve kullanıcı-tabanlı benzerlik modelleri, başarımı doğrultusunda beraber kullanılmıştır. Karma model (KM) olarak adlandırılan ve seçilen komşuluk değerlerine göre tek boyutta değil iki boyutta da veri kümesinin boyutunun indirgenmesini sağlayan deneylerin sonuçları ayrıntılı bir şekilde sunulmuştur.

Tez kapsamında önerilen modellerin daha iyi anlaşılabilmesi için bu modellere genel bir bakış Şekil 4.4'de sunulmuştur.



Şekil 4.4. Önerilen Modellere Genel Bakış

5. DENEYSEL ÇALIŞMALAR

Bu bölümde, önerilen modellerin başarımının analizi için yapılan deneysel çalışmalar ile ilgili ayrıntılı bilgiler sunulmaktadır. Yapılan deneysel çalışmalarda kullanılan veri kümesi ile ilgili ayrıntılı bilgi Bölüm 5.1 ve üretilen önerilerin doğruluğunun analizi için kullanılan değerlendirme ölçütleri ise Bölüm 5.2’de sunulmuştur. Deneysel çalışmalarda benimsenen metodoloji Bölüm 5.3’ ve deney sonuçları ve analizi Bölüm 5.4’de sunulmuştur.

5.1. Veri Kümesi

Önerilen modellerin komşuluk seçimi üzerindeki etkinliğinin ve başarımının analizi için yapılan deneysel çalışmalarda, Yahoo! şirketinin sahip olduğu çoklu-ölçütlü bir film öneri sistemi olan *YM* veri kümesi kullanılmıştır. Bu öneri sisteminde, genel beğeni değerlendirme ölçütüne ek olarak kullanıcılar filmleri dört farklı alt-ölçüt için değerlendirebilmektedir. Bu alt-ölçütler görsellik, senaryo, yönetmen ve oyunculuk ölçütleridir. Kullanıcılar bu sistemde, sistem tarafından belirlenen harfler yardımıyla muhtemel 13 adet değerlendirme yapabilmektedir (A+, A, A-, B+, B, B-, C+, C, C-, D+, D, D-, F). Bu harflerden A+ en yüksek değerlendirmeyi, F ise en düşük değerlendirmeyi temsil etmektedir. Orijinal *YM* veri kümesi, toplamda 127.729 kullanıcının, 8.272 film için yapmış olduğu 257.317 değerlendirmeyi içermektedir [57].

İkili-veriye dayanan çoklu-ölçütlü bir veri kümesi bulunmadığı için, önerilen modellerin analizini yapmak amacıyla, 13 farklı harften oluşan bir değerlendirme yelpazesine sahip olan *YM* veri kümesi üzerinde önışleme yapılmıştır. Öncelikle, harfler aracılığı ile yapılmış olan değerlendirmeler, sayısal verilere dönüştürülmüştür. En yüksek beğeni derecesini ifade eden A+, 13 olarak, en düşük beğeni derecesini ifade eden F ise 1 olarak dönüştürme işlemine tabi tutulmuştur. Benzer şekilde sistemdeki bütün değerlendirmelerin dönüşümü [1, 13] sayısal değerlendirme yelpazesinde olacak şekilde yapılmıştır. Bu tez kapsamında yapılan çalışmada, *BBS* algoritması ile kullanıcılara ürünleri beğenip beğenmeyeceği yönünde bir öneri üretileceği için, sayısal değerlere dönüştürülmüş olan

değerlendirmeler, ikili verilere dönüştürülmüştür. Bir kullanıcının yapmış olduğu değerlendirme, kendisinin daha önce yapmış olduğu değerlendirmelerin ortalamasından büyük ise 1 (Beğendim), küçük ise 0 (Beğenmedim) şeklinde bir dönüşüm yapılmıştır.

Çok boşluklu yapıda olan *YM* veri kümesinden, *YM10* ve *YM20* adında iki farklı test kümesi çıkarımı yapılarak deneysel çalışmalarda bu veri kümeleri kullanılmıştır. *YM10* en az 10 değerlendirme yapan kullanıcılar ve en az 10 kez değerlendirilmiş ürünlerin oluşturduğu veri kümesidir. *YM20* ise en az 20 değerlendirme yapan kullanıcılar ve en az 20 kez değerlendirilmiş ürünlerin oluşturduğu veri kümesidir. Bu veri kümelerinin içerdiği kullanıcı sayıları, ürün sayıları ve genel beğeni değerlendirme ölçütü için yapılmış olan değerlendirme sayıları Tablo 5.1’de sunulmaktadır.

Tablo 5.1. *YM10 ve YM20 Veri Kümeleri ile İlgili Ayrıntılı Bilgi*

Veri-Kümesi	Kullanıcı Sayısı	Ürün Sayısı	Genel Değerlendirme Sayısı
YM10	1.827	1.471	48.026
YM20	429	491	18.504

5.2. Değerlendirme Ölçütleri

Makine öğrenimi ve özellikle istatistiksel sınıflandırma alanlarında kullanılan algoritmaların performansını değerlendirmek amacıyla karışıklık matrisi kullanılmaktadır. Bu matriste, her bir sütun, önerilen sınıfı temsil eder. Her bir satır ise, sınıflandırılan ürünün, orijinal sınıfını temsil etmektedir. İkili verilerin sınıflandırıldığı bir sistem düşünüldüğünde iki ayrı sınıf için (1-pozitif, 0-negatif) karışıklık matrisi Tablo 5.2’de sunulmaktadır.

Tablo 5.2. *İkili-Veri için Karışıklık Matrisi*

		ÖNERİLEN DURUM	
		<i>Pozitif Durum</i>	<i>Negatif Durum</i>
GERÇEK DURUM	<i>Pozitif Durum</i>	Doğru Pozitif (<i>DP</i>)	Yanlış Negatif (<i>YN</i>)
	<i>Negatif Durum</i>	Yanlış Pozitif (<i>YP</i>)	Doğru Negatif (<i>DN</i>)

Tez kapsamında yapılan deneylerde, üretilen önerilerin başarımını değerlendirmek amacıyla sınıflandırma doğruluğu metriği (*SD*) ve F1 metriği (*F1*)

kullanılmıştır. *SD* ikili-veri için yapılan sınıflandırmanın ne kadar doğru olduğunu ölçmek için kullanılan bir metriktir. *SD* yapılan doğru sınıflandırmaların sayısının, yapılan sınıflandırma sayısına oranıdır. *SD* Denklem 5.1 ile hesaplanabilir:

$$SD = \frac{\text{Doğru sınıflandırma sayısı}}{\text{Bütün sınıflandırma sayısı}} = \frac{DP + DN}{DP + YN + YP + DN} \quad (5.1)$$

İkili verilerin sınıflandırmasının istatistiksel analizinde sıklıkla kullanılan bir diğer metrik ise *F1*'dir. *F1* test veri setinden hesaplanabilen duyarlık (*D*) ve çağrışım (*Ç*) değerlerinin harmonik ortalamasıdır. *D* yapılan doğru pozitif sınıflandırma sayısının, yapılan bütün pozitif sınıflandırma sayısına oranıdır. *Ç* ise yapılan doğru pozitif sınıflandırma sayısının, gerçek durumdaki pozitif sınıflandırma sayısına oranıdır. Bu metriklerin formülleri sırasıyla Denklem 5.2, Denklem 5.3 ve Denklem 5.4'de sunulmuştur.

$$D = \frac{\text{Doğru pozitif sayısı}}{\text{Önerilen pozitif durum sayısı}} = \frac{DP}{DP + YP} \quad (5.2)$$

$$\Ç = \frac{\text{Doğru pozitif sayısı}}{\text{Gerçek pozitif durum sayısı}} = \frac{DP}{DP + YN} \quad (5.3)$$

$$F1 = 2 \times \frac{D \times \Ç}{D + \Ç} \quad (5.4)$$

Bu iki metrik ne kadar yüksek ise, yapılan önerilerin doğruluğu o kadar yüksektir. Bu değerlendirme ölçütleri kullanılarak, önerilen modellerin, *BBS* algoritması ile üretilen önerilerin doğruluğunu ne kadar değiştirdiği gözlemlenmiştir.

5.3. Metodoloji

Tez kapsamında yapılan deneylerde, ilk olarak, ürünlerin genel değerlendirme ölçütü için, bir aktif kullanıcı ve hedef ürün seçilerek, yapılan değerlendirme değeri boş kabul edilmiştir. Geriye kalan bütün veriler, eğitilmiş veri kümesi olarak kabul edilip, bu veri kümesi üzerinde *BBS* algoritması ile aktif kullanıcının hedef ürünü

için bir öneri üretilmiştir. Bu işlem veri kümesindeki bütün değerlendirmeler için yapılmıştır. Herhangi bir komşuluk seçilmeden ve alt-ölçütlere yapılan değerlendirmeler kullanılmadan, sadece genel beğeni ölçütü için yapılmış olan değerlendirmeler üzerinde *BBS* algoritmasının uygulanması işlemi salt-Bayes olarak adlandırılırsa, YM10 ve YM20 veri kümeleri için salt-Bayes işleminin *SD* ve *F1* değeri Tablo 5.3'de sunulmaktadır. Ayrıca, salt-Bayes işlemi ile YM10 veri kümesinde toplamda 48.026 genel beğeni için öneri üretilirken, YM20 veri kümesinde 18.504 öneri üretilmiştir

Tablo 5.3. *Salt-Bayes için SD ve F1 Değerleri*

Veri Kümesi	<i>SD</i>	<i>F1</i>
YM10	70,5814	0,7790
YM20	71,7685	0,7822

Salt-Bayes işlemi tek bir ölçüt üzerinde gerçekleştirilmiştir. Tez kapsamında önerilen çoklu-ölçütlü modellerin başarımı, salt-Bayes işlemi baz alınarak değerlendirilmiştir. Önerilen modellerin başarımının analizi amacıyla, öneri üretilecek aktif kullanıcının hedef ürünü için önerilen komşuluk belirleme modelleri ile benzer kullanıcılar/ürünler belirlenerek belirli sayıda komşuluklar seçilmiştir. Bu komşuluklar, veri kümesinin daha küçük bir boyuta indirgenmesini sağlamıştır. Bu şekilde eğitilmiş/indirgenmiş veri kümesinin genel beğeni ölçütü üzerinde *BBS* algoritması uygulanarak öneriler üretilmiştir. Bu işlem veri kümesindeki bütün genel beğeni değerlendirmeleri için yapıлып, *SD* ve *F1* değerlendirme ölçütleri ile üretilen önerilerin başarımı analiz edilmiştir.

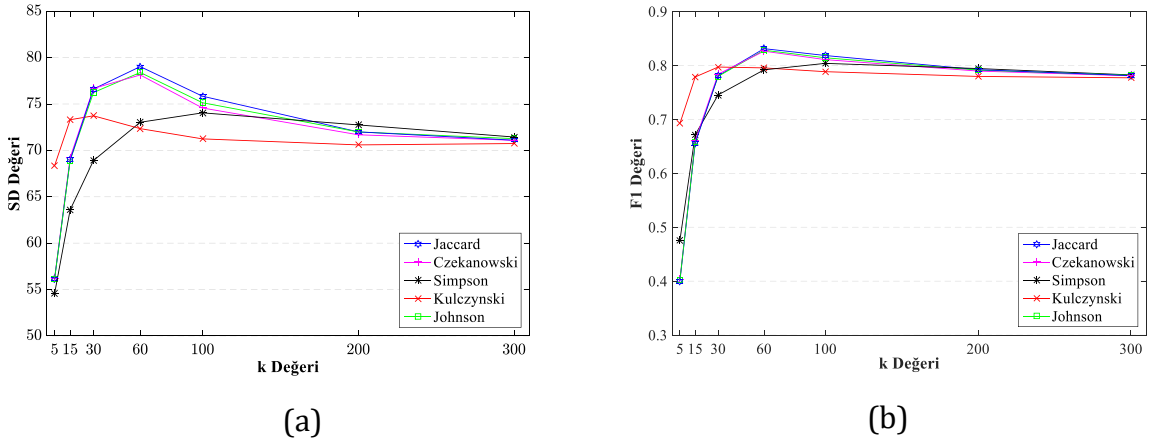
5.4. Deney Sonuçları ve Analizi

Önerilen modellerin, komşuluk seçimi üzerindeki etkisi ve genel beğeni ölçütü için yapılan önerilerin doğruluğuna etkisini gözlemlemek amacıyla çeşitli deneyler yapılmıştır. Bu deneylerde, önerilen *KB*, *ÜB*, *U-KB*, *U-ÜB* ve *KM* modelleri kullanılmıştır. Bu modeller kullanılarak, en benzer kullanıcıların/ürünlerin oluşturduğu bir alt-veri kümesi seçilerek genel beğeni ölçütü için yapılmış değerlendirmeler için *BBS* ile öneriler üretilmiştir. *SD* ve *F1* metrikleri yardımıyla üretilen önerilerin başarımı analiz edilerek, herhangi bir komşuluk seçilmeden

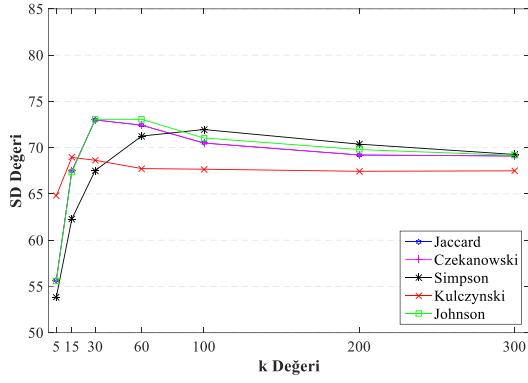
bütün genel beğeniler üzerinden öneri üreten salt-Bayes işleminin başarımının arttırılması hedeflenmiştir.

5.4.1. Kullanıcı-tabanlı benzerlik modeli

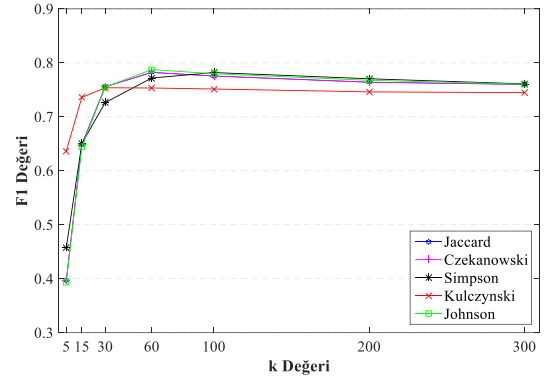
İlk olarak, *KB* modelinin iki farklı tekniği olan $benz_{ort}$ ve $benz_{min}$ ve beş farklı ikili-veri benzerlik metrikleri (Jaccard, Czekanowski, Simpson, Kulczynski, Johnson) kullanılarak, aktif kullanıcı için 5 ile 300 arasında değişen komşu kullanıcılar (k) seçilerek, bu komşuların genel beğeni ölçütü için yapmış oldukları değerlendirmelerin oluşturduğu veri kümesi üzerinde *BBS* algoritması uygulanmıştır. *YM20* veri kümesi üzerinde yapılan bu deneyler sonucunda genel beğeni ölçütü için üretilen önerilerin, değişkenlik gösteren *SD* ve *F1* değerleri sırasıyla Şekil 5.1 ve Şekil 5.2'de sunulmuştur.



Şekil 5.1. *KB* Modeli $benz_{ort}$ için *SD* ve *F1* Değerleri - *YM20*



(a)



(b)

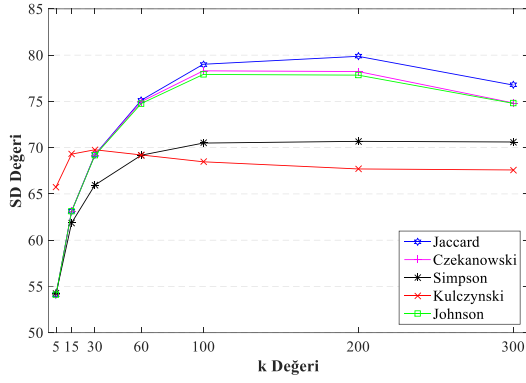
Şekil 5.2. KB Modeli $benz_{min}$ için SD ve F1 Değerleri - YM20

YM20 veri kümesinde genel beğeni ölçütü için yapılan deneylerin sonuçları gözlemlendiğinde, KB modeli için $benz_{ort}$ tekniği, bütün benzerlik ölçütleri için $benz_{min}$ tekniğinden daha başarılı sonuçlar elde edilmesini sağlamaktadır. Seçilen ikili benzerlik metriği ve seçilen komşuluk değerlerine göre elde edilen doğruluk değerleri değişkenlik göstermiştir. Genel olarak, $benz_{min}$ tekniğiyle yapılan önerilerin doğruluğu, salt-Bayes ile yapılan önerilerin doğruluğundan daha düşük olsa da, bazı komşuluk değerleri ve benzerlik metrikleri ile başarıyı daha yüksek öneriler üretilmesi sağlanmıştır. Örneğin Jaccard metriği ve $k=60$ için, $benz_{min}$ tekniğiyle, $SD=72,4$ ve $F1=0,78$ değerleri elde edilmiştir.

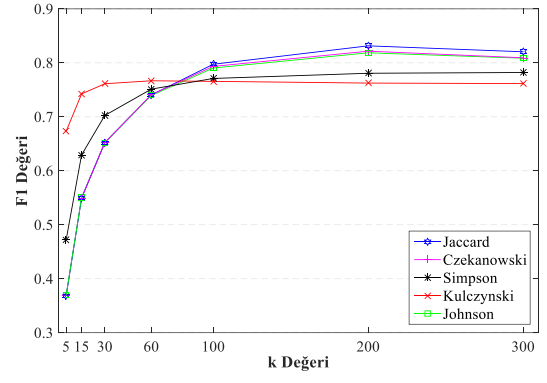
Şekil 5.1 incelendiğinde, $benz_{ort}$ için, en başarılı benzerlik metriğinin Jaccard olduğu ve en yüksek doğruluk değerinin $k=60$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=60$ seçildiğinde, SD değeri 79,0, F1 değeri ise 0,83 olmaktadır.

Şekil 5.2 incelendiğinde, $benz_{min}$ için en başarılı benzerlik metriğinin Jaccard ve Czekanowski olduğu ve en yüksek doğruluk değerinin $k=60$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=30$ seçildiğinde, SD değeri 72,9, F1 değeri ise 0,75 olmaktadır.

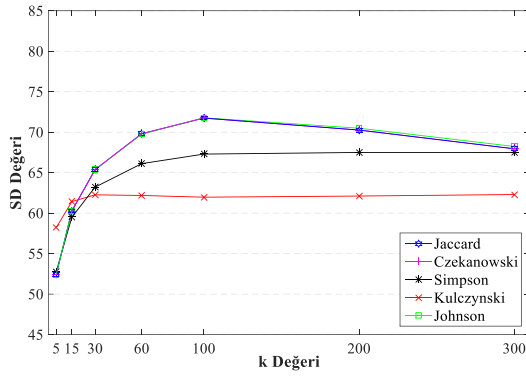
YM10 veri kümesi KB modeli için yapılan deneylerin SD ve F1 değerleri sırasıyla Şekil 5.3 ve Şekil 5.4'de sunulmuştur.



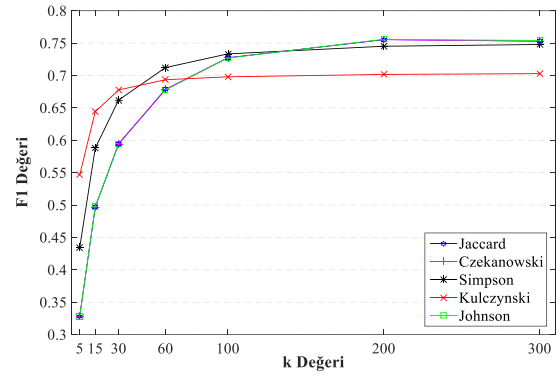
(a)



(b)

Şekil 5.3. KB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM10

(a)



(b)

Şekil 5.4. KB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10

YM10 veri kümesinde genel beğeni ölçütü için yapılan deneylerin sonuçları gözlemlendiğinde, KB modeli için $benz_{ort}$ tekniği, bütün benzerlik ölçütleri için $benz_{min}$ tekniğinden daha başarılı sonuçlar elde edilmesini sağlamaktadır. Seçilen ikili benzerlik metriği ve seçilen komşuluk değerlerine göre elde edilen doğruluk değerleri değişiklik göstermiştir. Genel olarak $benz_{min}$ tekniğiyle yapılan önerilerin doğruluğu, salt-Bayes ile yapılan önerilerin doğruluğundan daha düşük olsa da, bazı komşuluk değerleri ve benzerlik metrikleri ile başarıyı daha yüksek öneriler üretilmesi sağlanmıştır. Örneğin Jaccard metriği ve $k=100$ için, $benz_{min}$ tekniğiyle, $SD=71,7$ değeri elde edilmiştir.

Şekil 5.3 incelendiğinde, $benz_{ort}$ için en başarılı benzerlik metriğinin Jaccard olduğu ve en yüksek doğruluk değerinin $k=200$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=200$ seçildiğinde, SD değeri 79,8 iken $F1$ değeri ise 0,83 olarak elde edilmektedir.

Şekil 5.4 incelendiğinde, $benz_{min}$ için, en başarılı benzerlik metriklerinin Jaccard, Czekanowski ve Johnson olduğu ve en yüksek doğruluk değerinin $k=100$ ile bu benzerlik metrikleri ile sağlandığı görülmektedir. Üç benzerlik metriğinden birisi ve $k=100$ seçildiğinde, SD değeri 71,7, F1 değeri ise 0,72 olmaktadır.

KB modeli kullanılarak üretilen önerilerin doğruluğunun, salt-Bayes ile üretilen önerilerin doğruluğundan genellikle daha yüksek olduğu gözlemlenmektedir. Bu durum her iki veri kümesi için de geçerlidir. $benz_{ort}$ tekniği ve Jaccard metriği her iki veri kümesi için de başarıyı en yüksek sonuçların elde edilmesini sağlamıştır. YM20 veri kümesi için $k=60$ olarak seçildiğinde salt-Bayes ile üretilen önerilerin doğruluğunun %10 iyileştiği gözlemlenmektedir. YM10 veri kümesi için ise $k=200$ olarak seçildiğinde, bu iyileşmenin %13 olduğu gözlemlenmektedir. Üretilen önerilerin doğruluğundaki bu farkların istatistiksel açıdan anlamlı olması gerekmektedir. Bu anlamlılık bağımlı örneklem t -testi ile belirlenebilir. Bağımlı örneklem t -testi, bir değişkenin iki farklı durumda gözlemlenen değerlerinin ortalamalarını karşılaştırır. Bu iki durum genellikle uygulanacak yöntemin öncesi ve sonrası şeklinde olur. Dolayısıyla, her iki veri kümesi için de KB modeli ve salt-Bayes arasında bağımlı örneklem t -testi uygulanmıştır. t -testinin sonuçları Tablo 5.4’de sunulmaktadır.

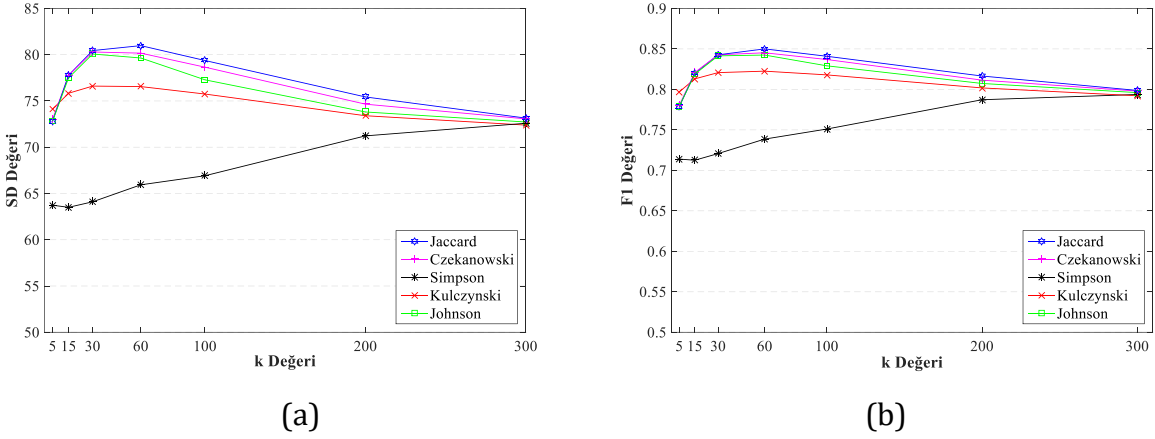
Tablo 5.4. *KB Modeli/Salt-Bayes için t-test Sonuçları*

Veri Kümesi	t değeri	p değeri
YM10	16,07	0,000
YM20	7,761	0,000

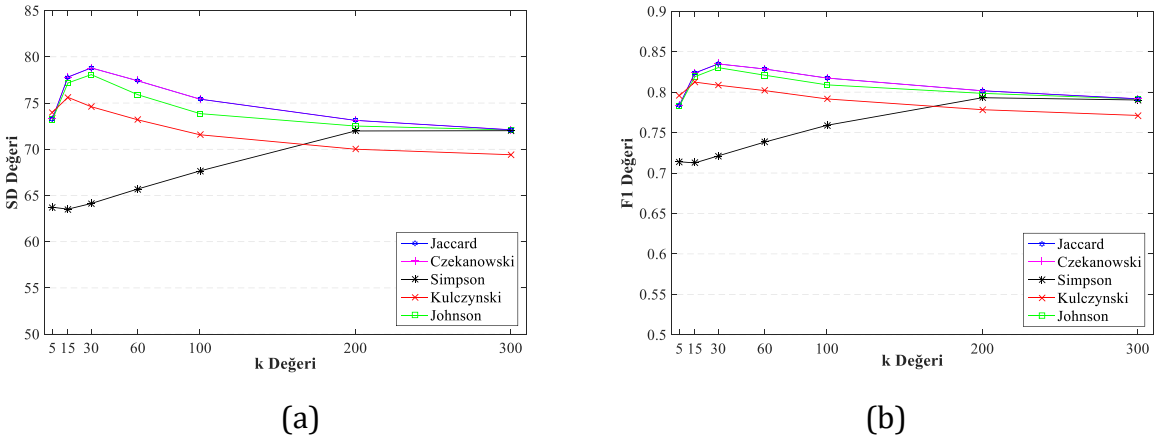
YM20 veri kümesi için üretilen 18.504 öneri ve YM10 veri kümesi için üretilen 48.026 öneri, 10 farklı örnekleme bölünerek, bu örneklemler üzerinden t -testi uygulanmıştır. Bu nedenle serbestlik derecesi dokuz olmuştur. Diğer modellerin sonuçlarının analizi için yapılan t -testlerde de bu yöntem kullanılmıştır. Yapılan t -testin sonucuna göre, $p<0,001$ olduğu için “%99,9 güvenle, KB modeli ile üretilen önerilerin ve salt-Bayes modeli ile üretilen önerilerin doğruluklarının arasında çok yüksek düzeyde istatistiksel olarak anlamlı bir farklılık vardır.” çıkarımı yapılabilir.

5.4.2. Ürün-tabanlı benzerlik modeli

ÜB modelinin iki farklı tekniği olan $benz_{ort}$ ve $benz_{min}$ ve beş farklı ikili veri benzerlik metrikleri kullanılarak, öneri üretilecek hedef ürün için 5 ile 300 arasında değişen komşu ürünler (k) seçilerek, bu komşuların genel beğeni ölçütü için yapılmış olan değerlendirmelerin oluşturduğu veri kümesi üzerinde *BBS* algoritması uygulanmıştır. YM20 veri kümesi üzerinde yapılan bu deneyler sonucunda genel beğeni ölçütü için üretilen önerilerin, değişkenlik gösteren *SD* ve *F1* değerleri sırasıyla Şekil 5.5 ve Şekil 5.6'da sunulmaktadır.



Şekil 5.5. ÜB Modeli $benz_{ort}$ için *SD* ve *F1* Değerleri - YM20



Şekil 5.6. ÜB Modeli $benz_{min}$ için *SD* ve *F1* Değerleri - YM20

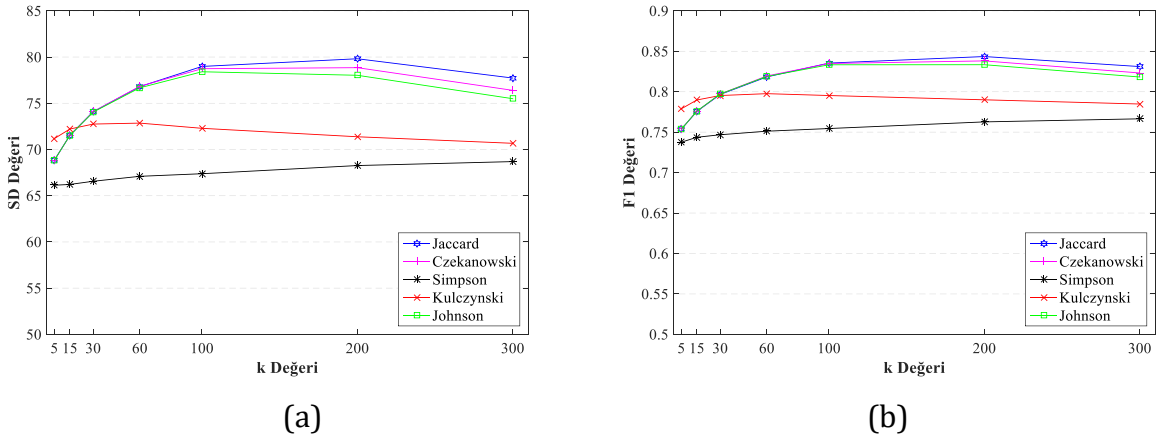
YM20 veri kümesinde genel beğeni ölçütü için yapılan deneylerin sonuçları gözlemlendiğinde, ÜB modeli için $benz_{ort}$ tekniği, bütün benzerlik ölçütleri için $benz_{min}$ tekniğinden daha başarılı sonuçlar elde edilmesini sağlamaktadır. Seçilen

ikili benzerlik metriği ve seçilen komşuluk değerlerine göre elde edilen doğruluk değerleri değişkenlik göstermiştir.

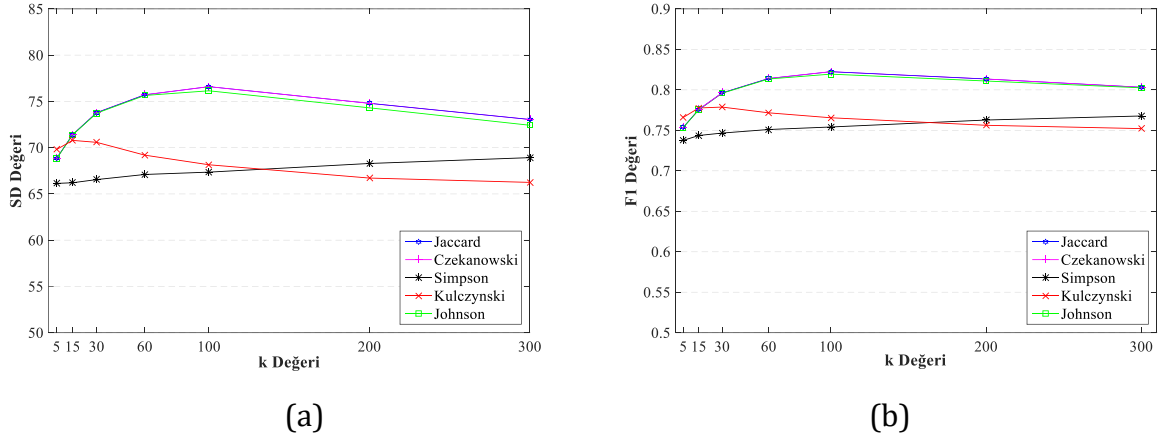
Şekil 5.5 incelendiğinde $benz_{ort}$ için, en başarılı benzerlik metriğinin Jaccard olduğu ve en yüksek doğruluk değerinin $k=60$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=60$ seçildiğinde, SD değeri 80,9, $F1$ değeri ise 0,85 olarak elde edilmektedir.

Şekil 5.6 incelendiğinde $benz_{min}$ için, en başarılı benzerlik metriğinin Jaccard ve Czekanowski olduğu ve en yüksek doğruluk değerinin $k=30$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=30$ seçildiğinde, SD değeri 78,7, $F1$ değeri ise 0,83 olmaktadır. Hem $benz_{min}$ hem de $benz_{ort}$ için yapılan deneylerin sonuçları gözlemlendiğinde, Simpson metriğinin diğer metriklerden farklı bir eğilime sahip olduğu söylenebilir. Düşük komşuluk sayılarında Simpson metriği kullanıldığında doğruluk değerlerinin diğer metriklerle oranla daha düşük olduğu görülmektedir. Komşuluk sayısının artırılması, Simpson metriğinin başarımının artmasını sağlamaktadır.

YM10 veri kümesi $\dot{U}B$ modeli için yapılan deneylerin SD ve $F1$ değerleri sırasıyla Şekil 5.7 ve Şekil 5.8'de sunulmaktadır.



Şekil 5.7. $\dot{U}B$ Modeli $benz_{ort}$ için SD ve $F1$ Değerleri – YM10



Şekil 5.8. ÜB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10

YM10 veri kümesinde genel beğeni ölçütü için yapılan deneylerin sonuçları gözlemlendiğinde, ÜB modeli için $benz_{ort}$ tekniği, bütün benzerlik ölçütleri için $benz_{min}$ tekniğinden daha başarılı sonuçlar elde edilmesini sağlamaktadır. Seçilen ikili benzerlik metriği ve seçilen komşuluk değerlerine göre elde edilen doğruluk değerleri değişkenlik göstermiştir.

Şekil 5.7 incelendiğinde, $benz_{ort}$ için en başarılı benzerlik metriğinin Jaccard olduğu ve en yüksek doğruluk değerinin $k=200$ ile sağlandığı görülmektedir. Jaccard ve $k=200$ seçildiğinde, SD değeri 79,8, F1 değeri ise 0,84 olarak elde edilmektedir.

Şekil 5.8 incelendiğinde, $benz_{min}$ için en başarılı benzerlik metriklerinin Jaccard ve Czekanowski ve en yüksek doğruluk değerinin $k=100$ ile bu benzerlik metrikleri ile sağlandığı görülmektedir. İki benzerlik metriğinden birisi ve $k=100$ seçildiğinde, SD değeri 76,5, F1 değeri 0,82 olarak elde edilmektedir.

ÜB modeli kullanılarak üretilen önerilerin doğruluğunun, salt-Bayes ile üretilen önerilerin doğruluğundan genellikle daha yüksek olduğu gözlemlenmektedir. Bu durum her iki veri kümesi için de geçerlidir. KB modeline benzer şekilde ÜB modelinde de, $benz_{ort}$ tekniği ve Jaccard metriği her iki veri kümesi için de en başarılı sonuçların elde edilmesini sağlamıştır. YM20 veri kümesi için $k=60$ olarak seçildiğinde salt-Bayes ile üretilen önerilerin doğruluğunun %12 arttığı gözlemlenmektedir. YM10 veri kümesi için ise $k=200$ olarak seçildiğinde, bu artışın %13 olduğu gözlemlenmektedir. Üretilen önerilerin doğruluğundaki bu farkların istatistiksel açıdan anlamlı olduğunu gözlemlemek için her iki veri kümesi

için uygulanan $\dot{U}B$ modeli ve salt-Bayes arasındaki bağımlı örneklem t -testi sonuçları Tablo 5.5’de sunulmaktadır.

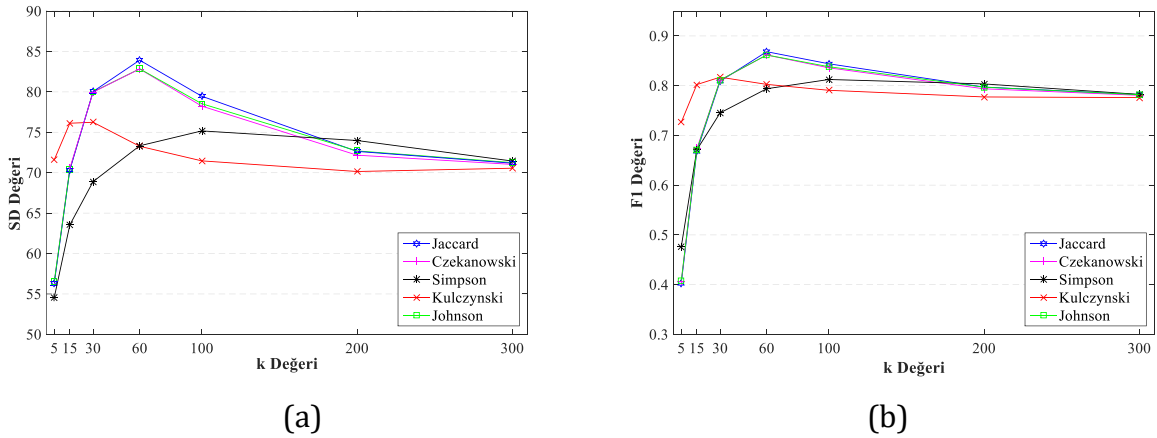
Tablo 5.5. $\dot{U}B$ Modeli/Salt-Bayes için t -test Sonuçları

Veri Kümesi	t değeri	p değeri
YM10	23,389	0,000
YM20	16,745	0,000

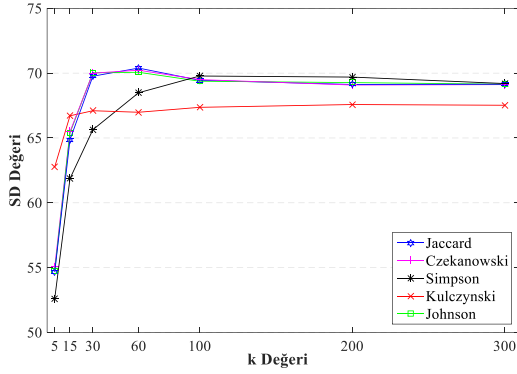
Bu sonuçlara göre $p < 0,001$ olduğu için, “%99,9 güvenle, salt-Bayes ile üretilen önerilerin ve $\dot{U}B$ modeli ile üretilen önerilerin doğruluklarının arasında çok yüksek düzeyde istatistiksel olarak anlamlı bir farklılık vardır.” çıkarımı yapılabilir.

5.4.3. Uyumluluk tekniği ile kullanıcı-tabanlı benzerlik modeli

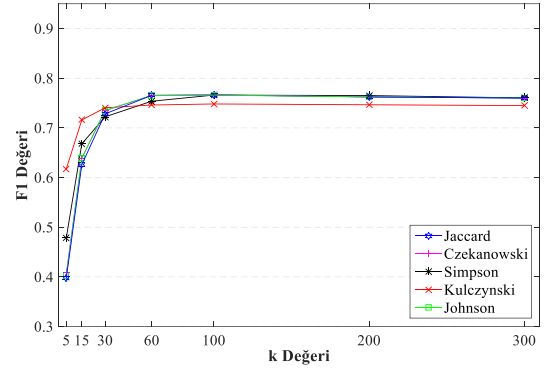
Uyumluluk tekniği ile kullanıcı-tabanlı benzerlik modelinde sunulan iki farklı teknik ($benz_{ort}$ ve $benz_{min}$) için 5 ile 300 arasında değişen komşuluklar seçilerek ve beş farklı benzerlik ölçütü kullanılarak öneriler üretilmiştir. YM20 veri kümesi üzerinde yapılan bu deneyler sonucunda genel beğeni ölçütü için üretilen önerilerin, değişkenlik gösteren SD ve $F1$ değerleri sırasıyla Şekil 5.9 ve Şekil 5.10’da sunulmaktadır.



Şekil 5.9. U -KB Modeli $benz_{ort}$ için SD ve $F1$ Değerleri - YM20



(a)



(b)

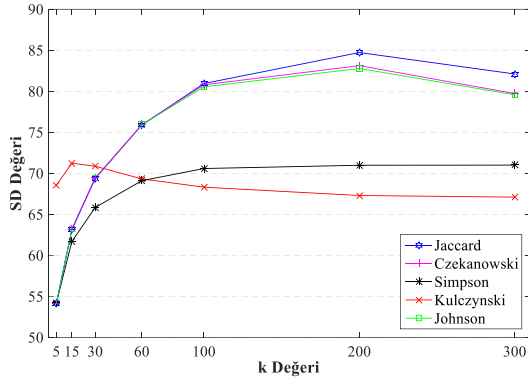
Şekil 5.10. *U-KB Modeli benz_{min} için SD ve F1 Değerleri - YM20*

YM20 veri kümesi için yapılan deneylerin sonuçları gözlemlendiğinde, uyumluluk tekniğinin kullanıcı-tabanlı komşuluk modeline olumlu katkı yaptığı üretilen önerilerin doğruluğu incelenerek söylenebilir. *U-KB* modelinde, genel olarak, *benz_{ort}* tekniği, *benz_{min}* tekniğinden başarıyı daha yüksek öneriler üretilmesini sağlamıştır. Uyumluluk tekniğinin *benz_{min}* tekniği için üretilen önerilerde olumlu bir katkısı olmamıştır. Her iki teknikte de en başarılı benzerlik metriği Jaccard olmuştur.

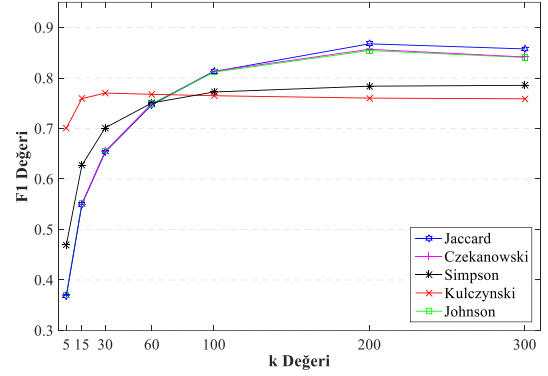
Şekil 5.9 incelendiğinde, *benz_{ort}* için en başarılı benzerlik metriğinin Jaccard, ikinci en iyi metriğin ise Johnson olduğu ve en yüksek doğruluk değerinin $k=60$ ile sağlandığı görülmektedir. Jaccard ve $k=60$ seçildiğinde, *SD* değeri 83,9, *F1* değeri ise 0,86 olmaktadır.

Şekil 5.10 incelendiğinde, *benz_{min}* için en başarılı benzerlik metriğinin Jaccard ve en yüksek doğruluk değerinin $k=60$ ile sağlandığı görülmektedir. Jaccard ve $k=60$ seçildiğinde, *SD* değeri 70,3, *F1* değeri ise 0,76 olmaktadır.

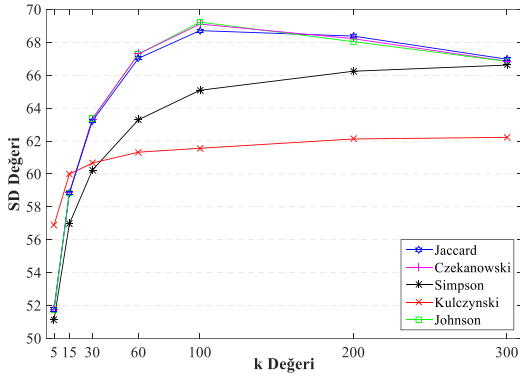
YM10 veri kümesi *U-KB* modeli için yapılan deneylerin *SD* ve *F1* değerleri sırasıyla Şekil 5.11 ve Şekil 5.12'de sunulmaktadır.



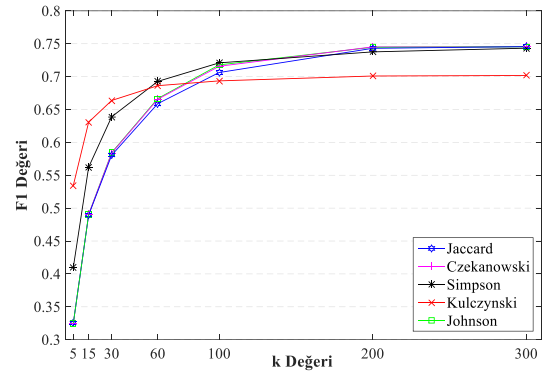
(a)



(b)

Şekil 5.11. U-KB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM10

(a)



(b)

Şekil 5.12. U-KB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM10

YM10 veri kümesi için yapılan deneyler incelendiğinde uyumluluk tekniğinin bu veri kümesi üzerinde de kullanıcı-tabanlı yaklaşım için olumlu katkı sağladığı söylenebilir. Bu veri kümesinde de $benz_{ort}$ tekniği $benz_{min}$ tekniğinden daha başarılı sonuçlar elde edilmesini sağlamıştır.

Şekil 5.11 incelendiğinde, $benz_{ort}$ için en başarılı benzerlik metriğinin Jaccard, ikinci en iyi metriğin ise Czekanowski olduğu ve en yüksek doğruluk değerinin $k=200$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=200$ seçildiğinde, SD değeri 84,7, F1 değeri ise 0,86 olmaktadır.

Şekil 5.12 incelendiğinde ise, $benz_{min}$ için en başarılı benzerlik metriklerinin Johnson ve Czekanowski ve en yüksek doğruluk değerinin $k=100$ ile sağlandığı görülmektedir. Johnson metriği ve $k=100$ seçildiğinde, SD değeri 69,2 olmaktadır. YM10 veri kümesi için $benz_{min}$ tekniği ile üretilen önerilerin doğruluk değeri, salt-Bayes ile üretilen önerilerin doğruluk değerinden daha düşük olmuştur. Benzer

şekilde uyumluluğun, *KB* modeli ve $benz_{min}$ tekniği ile üretilen önerilerin doğruluğu üzerinde olumlu katkısı olmamıştır.

Deney sonuçları gözlemlendiğinde, *U-KB* modeli ile üretilen önerilerin doğruluğu, *KB* modeli ile üretilen önerilerin doğruluğundan genel olarak daha yüksektir çıkarımı yapılabilir. Bu durum her iki veri kümesi için de geçerlidir. *KB* modeline benzer şekilde *U-KB* modelinde de $benz_{ort}$ ve Jaccard metriği en başarılı sonuçların elde edilmesini sağlamışlardır. Her iki veri kümesi için de *U-KB* modeli ile elde edilen en yüksek doğruluk değeri, *KB* modeli ile elde edilen en yüksek doğruluk değerinden %6 daha yüksektir. Uyumluluk tekniğinin, *KB* modeli üzerindeki olumlu katkısının istatistiksel açıdan anlamlı olduğunu gözlemlemek gerekmektedir. Bu amaçla *U-KB* ile *KB* modeli arasında bağımlı örneklem *t*-testi yapılmıştır. Yapılan *t*-testinin sonuçları Tablo 5.6'da sunulmaktadır.

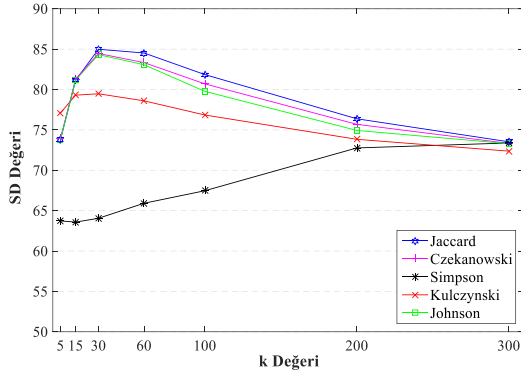
Tablo 5.6. *U-KB Modeli/KB Modeli için t-test Sonuçları*

Veri Kümesi	<i>t</i> değeri	<i>p</i> değeri
YM10	24,438	0,000
YM20	10,311	0,000

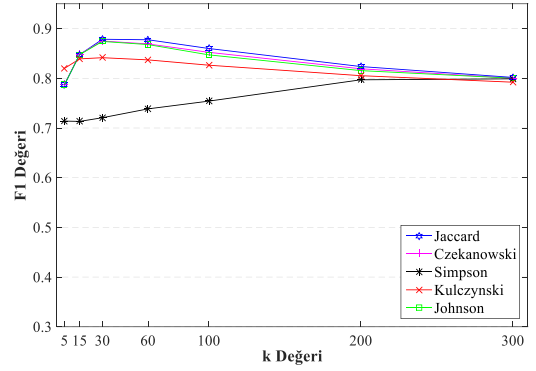
Bu sonuçlara göre $p < 0,001$ olduğu için, “%99,9 güvenle, *U-KB* modeli ile üretilen önerilerin ve *KB* modeli ile üretilen önerilerin doğruluklarının arasında çok yüksek düzeyde istatistiksel olarak anlamlı bir farklılık vardır.” çıkarımı yapılabilir.

5.4.4. Uyumluluk tekniği ile ürün-tabanlı benzerlik modeli

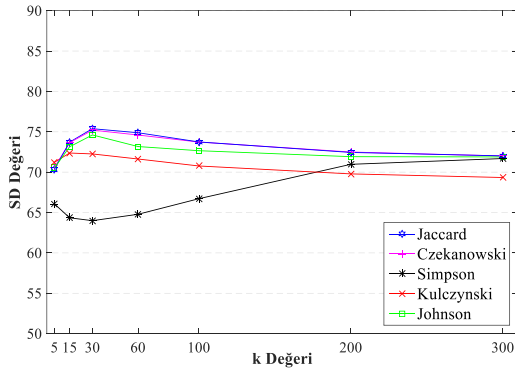
Uyumluluk tekniği ile ürün-tabanlı benzerlik modelinde sunulan iki farklı teknik ($benz_{ort}$ ve $benz_{min}$) için 5 ile 300 arasında değişen komşuluklar seçilerek ve beş farklı benzerlik ölçütü kullanılarak öneriler üretilmiştir. YM20 veri kümesi için yapılan deneyler neticesinde elde edilen doğruluk değerleri sırasıyla Şekil 5.13 ve Şekil 5.14'de sunulmaktadır.



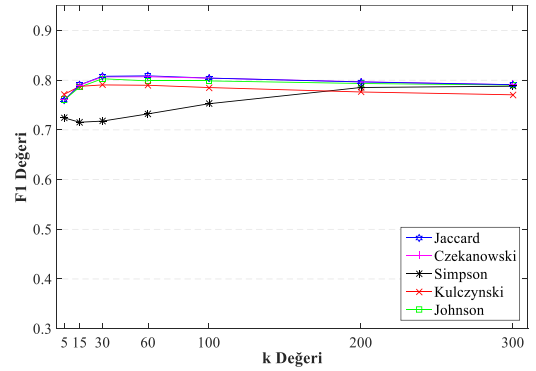
(a)



(b)

Şekil 5.13. U-ÜB Modeli $benz_{ort}$ için SD ve F1 Değerleri – YM20

(a)



(b)

Şekil 5.14. U-ÜB Modeli $benz_{min}$ için SD ve F1 Değerleri – YM20

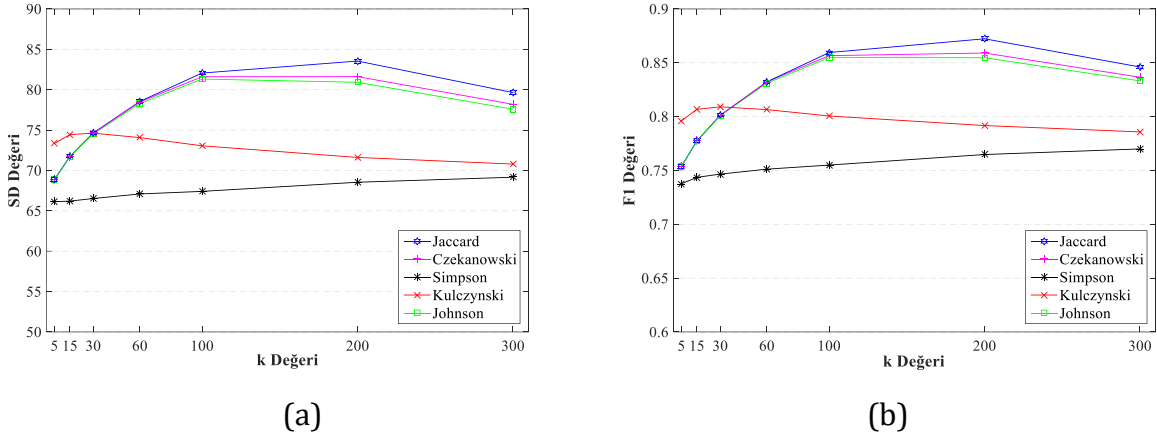
YM20 veri kümesi için yapılan deneylerin sonuçları gözlemlendiğinde, uyumluluk tekniğinin ürün-tabanlı komşuluk seçme modeline olumlu katkı yaptığı gözlemlenebilir. U-ÜB modelinde, genel olarak, $benz_{ort}$ tekniği, $benz_{min}$ tekniğinden başarıyı daha yüksek öneriler üretilmesini sağlamıştır. Uyumluluk tekniğinin $benz_{min}$ tekniği ile üretilen önerilerde olumlu katkısı olmamıştır. Her iki teknikte de en başarılı benzerlik metriği Jaccard olmuştur. Başarı sırasındaki ikinci metrik ise Czekanowski olmuştur.

Şekil 5.13 incelendiğinde, $benz_{ort}$ için, en başarılı benzerlik metriğinin Jaccard olduğu ve en yüksek doğruluk değerinin $k=30$ ile sağlandığı görülmektedir. Jaccard ve $k=30$ seçildiğinde, SD değeri 84,9 ve F1 değeri 0,87 olarak elde edilmektedir.

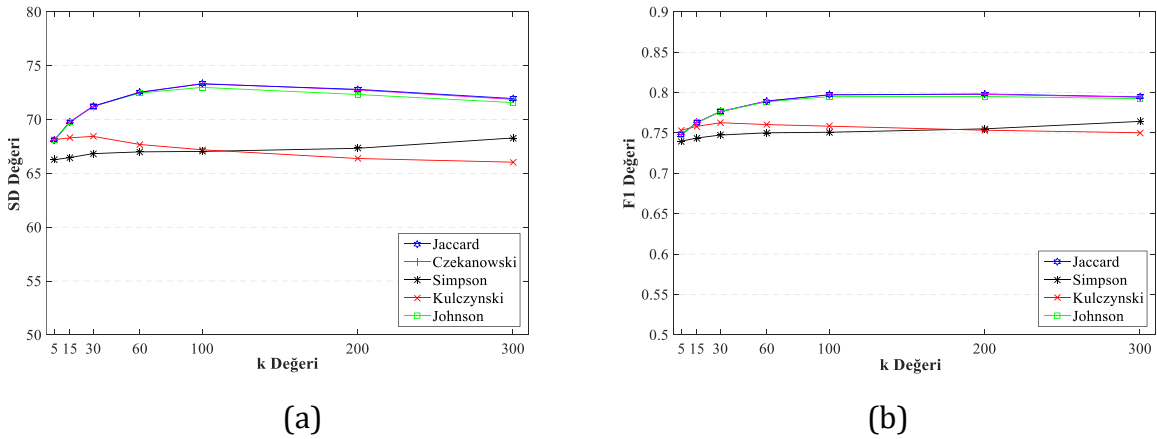
Şekil 5.14 incelendiğinde, $benz_{min}$ için, en başarılı benzerlik metriğinin yine Jaccard ve en yüksek doğruluk değerinin $k=30$ ve $k=60$ ile sağlandığı görülmektedir.

En yüksek SD değeri, Jaccard metriği ve $k=30$ seçildiğinde 75,3 olmaktadır. En yüksek $F1$ değeri ise Jaccard metriği ve $k=60$ seçildiğinde 0,80 olarak elde edilmiştir.

YM10 veri kümesi $U-\dot{U}B$ modeli için yapılan deneylerin SD ve $F1$ değerleri Şekil 5.15 ve Şekil 5.16'da sunulmaktadır.



Şekil 5.15. $U-\dot{U}B$ Modeli $benz_{ort}$ için SD ve $F1$ Değerleri – YM10



Şekil 5.16. $U-\dot{U}B$ Modeli $benz_{min}$ için SD ve $F1$ Değerleri – YM10

YM10 veri kümesi için yapılan deneylerin sonuçları gözlemlendiğinde, uyumluluk tekniğinin bu veri setinde de ürün-tabanlı model neticesinde üretilen önerilerin doğruluğunu arttırdığı gözlemlenebilir. Şu ana kadar verilen bütün deney sonuçlarına paralel şekilde bu deneyde de $benz_{ort}$ tekniğini $benz_{min}$ tekniğinden daha başarılı sonuçlar elde edilmesini sağlamıştır.

Şekil 5.15 incelendiğinde, $benz_{ort}$ için, en başarılı benzerlik metriğın Jaccard, en başarılı ikinci metriğın ise Czekanowski olduđu görölmektedir. En yüksek

doğruluk değerinin $k=200$ ile sağlandığı görülmektedir. Jaccard metriği ve $k=200$ seçildiğinde, SD değeri 83,5 ve $F1$ değeri 0,87 olmaktadır.

Şekil 5.16 incelendiğinde ise, $benz_{min}$ için, en başarılı benzerlik metriğininin Czekanowski, en başarılı ikinci metriğin ise Jaccard olduğu görülmektedir. En yüksek doğruluk değerinin $k=100$ ile sağlandığı görülmektedir. Czekanowski metriği ve $k=100$ seçildiğinde, SD değeri 73,3 ve $F1$ değeri 0,79 olarak elde edilmektedir.

Deney sonuçları gözlemlendiğinde, $U-ÜB$ modeli ile üretilen önerilerin doğruluğu, $ÜB$ modeli ile üretilen önerilerin doğruluğundan genel olarak daha yüksektir çıkarımı yapılabilir. Bu durum her iki veri kümesi için de geçerlidir. $ÜB$ modeline benzer şekilde $U-ÜB$ modelinde de $benz_{ort}$ ve Jaccard metriği en başarılı sonuçların elde edilmesini sağlamışlardır. Her iki veri kümesi için de $U-ÜB$ modeli ile elde edilen ek yüksek doğruluk değeri, $ÜB$ modeli ile elde edilen en yüksek doğruluk değerinden %4 daha yüksektir. Uyumluluk tekniğinin, $ÜB$ modeli üzerindeki olumlu katkısının istatistiksel açıdan anlamlı olduğunu gözlemlemek gerekmektedir. Bu amaçla $U-ÜB$ ile $ÜB$ modeli arasında bağımlı örneklem t -testi yapılmıştır. Yapılan t -testinin sonuçları Tablo 5.7’de sunulmaktadır.

Tablo 5.7. $U-ÜB$ Modeli/ $ÜB$ Modeli için t -test Sonuçları

Veri Kümesi	t değeri	p değeri
YM10	23,133	0,000
YM20	14,226	0,000

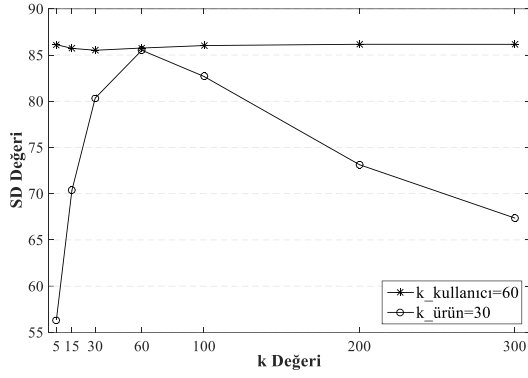
Yapılan t -test sonuçlarına göre, $p<0,001$ olduğu için, “%99,9 güvenle, $U-ÜB$ modeli ile üretilen önerilerin ve $ÜB$ modeli ile üretilen önerilerin doğruluklarının arasında çok yüksek düzeyde istatistiksel olarak anlamlı bir farklılık vardır.” çıkarımı yapılabilir.

5.4.5. Karma model

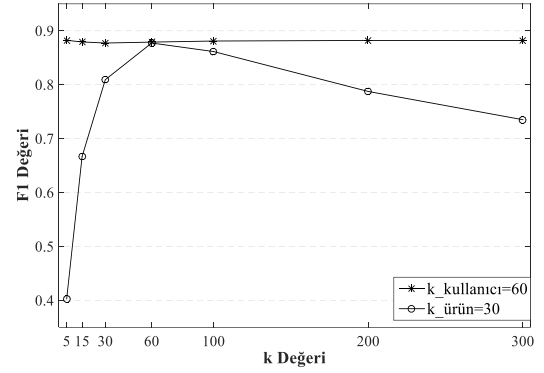
Bu bölümde, önerilen karma model tekniğinin, üretilen önerilerin doğruluğunu nasıl etkilediğini gözlemlemek amacıyla, hem kullanıcılar arasındaki hem de ürünler arasındaki benzerlikleri kullanarak yapılan deneylerin sonuçları

sunulmuştur. Bu noktaya kadar önerilen modeller için yapılan deneyler gözlemlendiğinde, kullanıcılar arasındaki benzerliği en iyi belirleyen modelin $U-KB$, ürünler arasındaki benzerliği en iyi belirleyen modelin ise $U-ÜB$ olduğu tespit edilmiştir. Bu modellerin $benz_{ort}$ tekniği yapılan bütün deneylerde $benz_{min}$ tekniğinden daha başarılı olduğu belirlenmiştir. Bu nedenle, karma model için yapılan deneylerde $U-KB$ ve $U-ÜB$ modellerinin $benz_{ort}$ teknikleri beraber kullanılarak, öneri üretilecek aktif kullanıcının hedef ürünü için, aynı anda hem komşu kullanıcılar hem de komşu ürünler seçilerek, seçilen bu veri kümesi üzerinde BBS algoritması ile öneriler üretilmiştir.

Şekil 5.9'da sunulan ve YM20 veri kümesi ve $U-KB$ için yapılan deneylerin sonuçları gözlemlendiğinde, başarıyı en yüksek öneriler, $k=60$ ve Jaccard metriği kullanılarak elde edilmiştir. Bu yüzden, karma model için yapılan ilk deneyde, kullanıcılar için seçilen komşuluk değerini $k_{kullanıcı}=60$ olarak sabitleyip, $U-ÜB$ modeli kullanılarak 5 ile 300 arasında değişen değerlerde hedef ürüne benzer ürünler belirlenerek, iki boyutta da daraltılmış veri kümesi üzerinde öneriler üretilmiştir. Benzer şekilde Şekil 5.13'de sunulan ve YM20 veri kümesi için $U-ÜB$ için yapılan deneylerin sonuçları gözlemlendiğinde, başarıyı en yüksek öneriler, $k=30$ ve Jaccard metriği kullanılarak elde edilmiştir. Bu yüzden, karma model için yapılan ikinci deneyde, hedef ürünle benzer ürünler için seçilen komşuluk değerini $k_{ürün}=30$ olarak sabitleyip, $U-KB$ modeli kullanılarak aktif kullanıcı için 5 ile 300 arasında değişen değerlerde komşular belirlenerek, iki boyutta daraltılmış veri kümesi kullanılarak öneriler üretilmiştir. YM20 veri kümesi için yapılan bu deneylerin sonuçları Şekil 5.17'de sunulmaktadır.



(a)



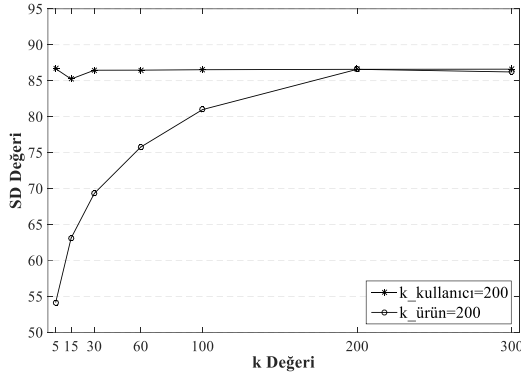
(b)

Şekil 5.17. Karma Modeli için SD ve F1 Değerleri – YM20

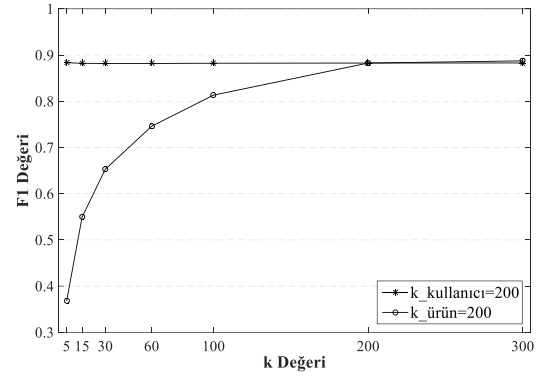
Şekil 5.17 incelendiğinde, YM20 veri kümesi için yapılan deneylerde en yüksek SD değeri 86,1 ve en yüksek F1 değeri 0,88 olarak elde edilmiştir. Bu değerler $k_{kullanıcı}=60$ ve $k_{ürün}=200$ olarak seçildiğinde elde edilmiştir.

Şekil 5.11' de sunulan ve YM10 veri kümesi ve *U-KB* için yapılan deneylerin sonuçları gözlemlendiğinde, başarıyı en yüksek öneriler, $k=200$ ve Jaccard metriği kullanılarak elde edilmiştir. Bu yüzden, karma model için yapılan ilk deneyde, kullanıcılar için seçilen komşuluk değerini $k_{kullanıcı}=200$ olarak sabitleyip, *U-ÜB* modeli kullanılarak 5 ile 300 arasında değişen değerlerde hedef ürüne benzer ürünler belirlenerek, iki boyutta da daraltılmış veri kümesi üzerinde öneriler üretilmiştir. Benzer şekilde Şekil 5.15' de sunulan ve YM10 veri kümesi için *U-ÜB* için yapılan deneylerin sonuçları gözlemlendiğinde, başarıyı en yüksek öneriler, $k=200$ ve Jaccard metriği kullanılarak elde edilmiştir. Bu yüzden, karma model için yapılan ikinci deneyde, hedef ürünle benzer ürünler için seçilen komşuluk değerini $k_{ürün}=200$ olarak sabitleyip, *U-KB* modeli kullanılarak aktif kullanıcı için 5 ile 300 arasında değişen değerlerde komşular belirlenerek, iki boyutta daraltılmış veri kümesi kullanılarak öneriler üretilmiştir.

YM10 veri kümesi için yapılan bu deneylerin sonuçları Şekil 5.18' de sunulmaktadır.



(a)



(b)

Şekil 5.18. Karma Modeli için SD ve F1 Değerleri – YM10

Şekil 5.18 incelendiğinde, YM10 veri kümesi için yapılan deneylerde en yüksek SD değeri 86,6 ve en yüksek F1 değeri 0,88 olarak elde edilmiştir. Bu değerler $k_{kullanıcı}=200$ ve $k_{ürün}=200$ olarak seçildiğinde elde edilmiştir.

Doğruluğu en yüksek önerilerin üretilmesini sağlayan komşuluk değerlerinin her iki veri kümesi için de farklılık gösterdiği gözlemlenmektedir. Bunun sebebi YM10 veri kümesinin daha boşluklu yapıda olmasıdır. Bu nedenle YM10 veri kümesinde YM20 veri kümesine kıyasla daha yüksek komşuluk değerleri seçildiğinde daha başarılı sonuçlar elde edilmesi sağlanmıştır.

Her iki veri kümesi için de, karma modelin, $U-KB$ ve $U-ÜB$ modelleri ile üretilen önerilere kıyasla doğruluğu daha yüksek önerilerin üretilmesini sağladığı deney sonuçları ile gözlemlenmiştir. YM20 veri kümesi için, karma modelde $k_{kullanıcı}=60$ ve $k_{ürün}=100$ olarak seçildiğinde elde edilen doğruluk değeri, $U-KB$ ile elde edilen en yüksek doğruluk değerinden %2, $U-ÜB$ ile elde edilen en yüksek doğruluk değerinden ise %1 daha yüksek olmuştur. YM10 veri kümesinde ise, $k_{kullanıcı}=200$ ve $k_{ürün}=200$ olarak seçildiğinde elde edilen doğruluk değeri, $U-KB$ ile elde edilen en yüksek doğruluk değerinden %2, $U-ÜB$ ile elde edilen en yüksek doğruluk değerinden ise %3 daha yüksek olmuştur. Karma modelin $U-KB$ ve $U-ÜB$ modellerine kıyasla doğruluğu daha yüksek öneriler ürettiğinin istatistiksel açıdan anlamlı olduğunu gözlemlenmek gerekmektedir. Bu amaçla, karma model ile $U-KB$ ve $U-ÜB$ modelleri arasında bağımlı örneklem t -testi yapılmıştır. Yapılan t -testinin sonuçları Tablo 5.8'de sunulmaktadır.

Tablo 5.8. Karma Model/U-KB ve U-ÜB için t-test Sonuçları

Veri Kümesi	Karma Model/U-KB		Güven Aralığı	Karma Model/U-ÜB		Güven Aralığı
	t değeri	p değeri		t değeri	p değeri	
YM10	3,138	0,012	%95	7,556	0,000	%99,9
YM20	2,109	0,064	%90	2,983	0,015	%95

Yapılan t-testlerde elde edilen p değerine göre güven aralıkları hesaplanarak Tablo 5.8'de sunulmuştur. Karma model ile üretilen önerilerin ve U-KB/U-ÜB modelleri ile üretilen önerilerin doğruluk değerleri arasında bu güven aralık değerleri doğrultusunda istatistiksel olarak anlamlı bir farklılık vardır çıkarımı yapılabilir.

6. SONUÇLAR VE DEĞERLENDİRME

Bu tez kapsamında yapılan çalışmalarda, ikili veriye dayanan çoklu-ölçütlü ortak filtreleme sistemlerinde basit Bayes sınıflandırıcı algoritmasının uygulanabilirliği ele alınmıştır. Tek-ölçüt tabanlı sistemlerde ortak filtreleme amacıyla kullanılabilen basit Bayes sınıflandırıcı algoritması, çoklu-ölçütlü sistemlerin kullanıcılara ve sistem sahiplerine sağladığı avantajlarla beraber kullanarak, genel beğeni ölçütü için kullanıcılara üretilen önerilerin doğruluğunu iyileştirmeyi amaçlayan çeşitli modeller geliştirilmiştir. Çoklu-ölçütlü sistemler, kullanıcıların ürünler/hizmetler hakkındaki düşüncelerini farklı açılardan değerlendirebildikleri sistemlerdir. Bu yetenek, kullanıcı veya ürün profillerinin daha kişiselleştirilmiş olarak oluşturulmasını sağlamaktadır. Bu yetenek doğrultusunda geliştirilen modeller, kullanıcılar/ürünler arasındaki benzerlik değerlerinin daha doğru olarak hesaplanmasını sağlamaktadır. Kullanıcılar/ürünler arasındaki benzerliklerin daha doğru belirlenmesi, daha etkili komşulukların seçilmesini sağlamıştır. Tez kapsamında geliştirilen modeller, ikili veriye dayanan çoklu-ölçütlü sistemlerde uygulanmıştır. İkili-verilerden oluşan sistemlerde, kullanıcı ve ürün değerlendirmeleri ikili-verilerden oluşan vektörler gibi düşünülerek, kullanıcılar/ürünler arasındaki benzerlikler, çeşitli ikili benzerlik metrikleri yardımıyla belirlenmiştir.

İlk olarak, daha etkili komşulukların belirlenmesi sağlanarak basit Bayes sınıflandırıcı algoritmasının genel beğeni ölçütü üzerindeki başarımını arttıran, kullanıcı ve ürün-tabanlı benzerlik modelleri geliştirilmiştir. Bu modeller yardımıyla, basit Bayes sınıflandırıcı algoritmasını bütün kullanıcıların/ürünlerin genel beğeni ölçütü için yapılan değerlendirmeler üzerinde uygulamak yerine, belirli komşuluklar seçilerek, seçilen komşuların genel beğeni ölçütü için yapmış oldukları değerlendirmeler üzerinde uygulanmıştır. Ardından, kullanıcıların her bir ölçüt için olan hassasiyetlerinin, komşuluk belirlenmesinde kullanılabilmesini sağlayabilen uyumluluk tekniği, bu modellerle beraber kullanılmıştır. Yapılan deneylerin sonuçları gözlemlendiğinde, geliştirilen iki yeni model (Uyumluluk tekniği ile kullanıcı-tabanlı ve ürün-tabanlı benzerlik) ile genel beğeni ölçütü için üretilen önerilerin başarımının daha da arttırıldığı gözlemlenmiştir. Veri kümesinin sadece tek bir boyutta değil, geliştirilen modeller yardımıyla iki boyutta da

indirgenmesini sađlayan karma model ile kullanıcı ve ürün-tabanlı komşuları beraber ve aynı anda seçerek üretilen önerilerin başarımının arttırılması sađlanmıştır.

Geliştirilen modellerin başarımının analizinde gerçek veri setleri kullanılarak deneyler yapılmıştır. Ortak filtreleme yaklaşımlarında popüler olarak kullanılan iki farklı değerlendirme ölçütü (sınıflandırma doğruluđu ve F1 metriđi) ile üretilen önerilerin doğrulukları gözlemlenmiştir. Yapılan deneysel çalışmalarda, kullanılan benzerlik metrikleri arasında Jaccard metriđinin kullanıcılar/ürünler arasındaki benzerlikleri en etkili belirleyen metrik olduđu gözlemlenmiştir. İkili-veriye dayanan çoklu-ölçüt tabanlı sistemlerde ortak filtreleme amacıyla bir çalışma yapılmaması, bu tez kapsamında yapılan çalışmalar ve önerilen yaklaşımların özgünlüğünü sađlamaktadır.

Tez kapsamında yapılan çalışmalar ve önerilen yaklaşımlar, bu alanda yapılacak sonraki çalışmalara ışık tutacağına inanılmaktadır. Tek ölçütlü ortak filtreleme yaklaşımlarında karşılaşılan problemlerin üstesinden gelmek için önerilen yaklaşımlar ikili-veriye dayanan çoklu-ölçütlü ortak filtreleme sistemleri için uyarlanabilir. Ayrıca, önerilen modeller geliştirilerek ortak filtreleme yaklaşımlarında ana hedeflerden birisi olan üretilen önerilerin doğruluđunun iyileştirilmesi sađlanabilir.

KAYNAKÇA

- [1] Mahmood, T. and Ricci, F. (2009). Improving recommender systems with adaptive conversational strategies. *Proceedings of the 20th ACM Conference on Hypertext and Hypermedia*, New York, USA: ACM, 73-82.
- [2] Burke, R. (2007). Hybrid Web recommender systems. *The adaptive web*. Springer Berlin Heidelberg. 4321(1), 377-408.
- [3] Resnick, P. and Varian, H. R. (1997). Recommender Systems. *Communications of the ACM*, 40(3), 56-58.
- [4] Goldberg, D., Nichols, D., Oki, B. M. and Terry, D. (1992). Using collaborative filtering to weave an information Tapestry. *Communications of the ACM*, 35(12), 61-70.
- [5] Ricci, F., Rokach, L. and Shapira, B. (2011). Introduction to Recommender Systems Handbook, Kantor P. B., *Recommender Systems Handbook* (1-35). Springer US.
- [6] Kazienko, P., and Musiał, K. (2006). Recommendation Framework for Online Social Networks, Last M., *Advances in Web Intelligence and Data Mining*. Springer Berlin Heidelberg. 23(1), 111-120.
- [7] Radev, D. R., Fan, W. and Zhang, Z. (2001). WebInEssence: A personalized Web-based multi-document summarization and recommendation system. *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics-01*, Ann Arbor, MI, USA, 79-88.
- [8] Barwise, P., Elberse, A. and Hammond, K. (2002). *Marketing and the Internet: A research review* (pp. 01-801). London: London Business School.
- [9] Goldberg, K., Roeder, T., Gupta, D. and Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2), 133-151.
- [10] Herlocker, J. L., Konstan, J. A., Terveen, L. G. and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5-53.
- [11] Sarwar, B., Karypis, G., Konstan, J. A. and Riedl, J. T. (2000). Analysis of recommendation algorithms for e-commerce. *Proceedings of the 2nd ACM Conference on Electronic Commerce*, New York, USA: ACM, 158-167.

- [12] Choi, K. and Suh, Y. (2013). A new similarity function for selecting neighbors for each target item in collaborative filtering. *Knowledge-Based Systems*, 37(1), 146-153.
- [13] Herlocker, J. L., Konstan, J. A., Borchers, A. and Riedl, J. T. (1999). An algorithmic framework for performing collaborative filtering. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, USA: ACM, 230-237.
- [14] Karypis, G. (2001). Evaluation of item-based top-*n* recommendation algorithms. *Proceedings of the 10th International Conference on Information and Knowledge Management*, New York, USA: ACM, 247-254.
- [15] Breese, J. S., Heckerman, D. and Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 43-52.
- [16] Ghazanfar, M. A., Prügel-Bennett, A. and Szedmak, S. (2012). Kernel-mapping recommender system algorithms. *Information Sciences*, 208, 81-104.
- [17] Adomavicius, G. and 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.
- [18] Yu, K., Schwaighofer, A., Tresp, V., Xu, X. and Kriegel, H. P. (2004). Probabilistic memory-based collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, 16(1), 56-69.
- [19] Billsus, D. and Pazzani, M. J. (1998). Learning collaborative information filters. *Proceedings of the 15th International Conference on Machine Learning*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 46-54.
- [20] Landauer, T. K. and Littman, M. L. (1994). Computerized cross-language document retrieval using latent semantic indexing, U.S. Patent, No: 5,301,109.
- [21] Sarwar, B., Karypis, G., Konstan, J. A. and Riedl, J. T. (2002). Incremental singular value decomposition algorithms for highly scalable recommender systems. *Proceedings of the 5th International Conference on Computer and Information Science*, 27-28.

- [22] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K. and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391-407.
- [23] Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D. and Sartin, M. (1999). Combining content-based and collaborative filters in an online newspaper. *Proceedings of ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation*, 60.
- [24] Lam, S. K. and Riedl, J. T. (2004). Shilling recommender systems for fun and profit. *Proceedings of the 13th International Conference on World Wide Web*, New York, USA: ACM, 393-402.
- [25] Mobasher, B., Burke, R., Bhaumik, R. and Williams, C. A. (2005). Effective attack models for shilling item-based collaborative filtering systems. *Proceedings of the 2005 Web KDD Workshop*, New York, USA: ACM
- [26] Jannach, D., Zanker, M., Felfernig, A. and Friedrich, G. (2010). *Recommender Systems: An Introduction*, New York: Cambridge University Press.
- [27] Hofmann, T. and Hartmann, D. (2005). Collaborative filtering with privacy via factor analysis. *Proceedings of the 2005 ACM Symposium on Applied Computing*, New York, USA: ACM, 791-795.
- [28] Miller, B. N., Konstan, J. A. and Riedl, J. T. (2004). PocketLens: Toward a personal recommender system. *ACM Transactions on Information Systems*, 22(3), 437-476.
- [29] Balabanović, M. and Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66-72.
- [30] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331-370.
- [31] Mahmood, T. and Ricci, F. (2007). Towards learning user-adaptive state Models in a Conversational Recommender System. *Proceedings of the LWA 2007 Workshops*, Martin-Luther-University, Germany, 373-378.
- [32] Su, X. and Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009(2009), 19.
- [33] Miyahara, K. and Pazzani, M. J. (2002). Improvement of collaborative filtering with the simple Bayesian classifier. *Proceedings of the 6th Pacific Rim*

International Conference on Artificial Intelligence, Heidelberg: Springer-Verlag Berlin, 679-689.

- [34] Shardanand, U. and Maes, P. (1995). Social information filtering: algorithms for automating “word of mouth”. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, USA: ACM Press/Addison-Wesley Publishing Co., 210-217.
- [35] Hill, W., Stead, L., Rosenstein, M. and Furnas, G. (1995). Recommending and evaluating choices in a virtual community of use. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, USA: ACM Press/Addison-Wesley Publishing Co., 194-201.
- [36] Plantié, M., Montmain, J. and Dray, G. (2005). Movies recommenders systems: Automation of the information and evaluation phases in a multi-criteria decision-making process. *Database and Expert Systems Applications*, 3588, 633-644.
- [37] Naak, A., Hage, H. and Aimeur, E. (2009). A multi-criteria collaborative filtering approach for research paper recommendation in papyres. *Proceedings of the E-Technologies: Innovation in an Open World*, Heidelberg: Springer-Verlag Berlin, 25-39.
- [38] Di Bitonto, P., Laterza, M., Roselli, T. and Rossano, V. (2010). Multi-criteria retrieval in cultural heritage recommendation systems. *Knowledge-Based and Intelligent Information and Engineering Systems*, 6277, 64-73.
- [39] Fuchs, M. and Zanker, M. (2012). Multi-criteria ratings for recommender systems: An empirical analysis in the tourism domain. *Proceedings of the 13th International Conference on E-commerce and Web Technologies*, Cham, Switzerland: Springer Berlin Heidelberg. 123, 100-111.
- [40] Sanchez-Vilas, F., Ismoilov, J., Lousame, F. P., Sanchez, E. and Lama, M. (2011). Applying multicriteria algorithms to restaurant recommendation. *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, Washington, USA: IEEE Computer Society, 87-91.
- [41] Manouselis, N. and Costopoulou, C. (2007). Analysis and classification of multi-criteria recommender systems. *World Wide Web*, 10(4), 415-441.

- [42] Adomavicius, G. and Kwon, Y. (2007). New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems*, 22(3), 48-55.
- [43] Li, Q., Wang, C. and Geng, G. (2008). Improving personalized services in mobile commerce by a novel multicriteria rating approach. *Proceedings of the 17th International Conference on World Wide Web*, New York, USA: ACM, 1235-1236.
- [44] Lakiotaki, K., Tsafarakis, S. and Matsatsinis, N. (2008). UTA-Rec: A recommender system based on multiple criteria analysis. *Proceedings of the 2008 ACM Conference on Recommender Systems*, New York, USA: ACM, 219-226.
- [45] Zhang, Y., Zhuang, Y., Wu, J. and Zhang, L. (2009). Applying probabilistic latent semantic analysis to multi-criteria recommender system. *AI Communications*, 22(2), 97-107.
- [46] Liu, L., Mehandjiev, N. and Xu, D. L. (2011). Multi-criteria service recommendation based on user criteria preferences. *Proceedings of the 5th ACM Conference on Recommender Systems*, New York, USA: ACM, 77-84.
- [47] Lakiotaki, K., Matsatsinis, N. F. and Tsoukias, A. (2011). Multicriteria user modeling in recommender systems. *IEEE Intelligent Systems*, 26(2), 64-76.
- [48] Shambour, Q. and Lu, J. (2011). A hybrid multi-criteria semantic-enhanced collaborative filtering approach for personalized recommendations. *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 0*, Washington, USA: IEEE Computer Society, 71-78.
- [49] Jannach, D., Karakaya, Z. and Gedikli, F. (2012). Accuracy improvements for multi-criteria recommender systems. *Proceedings of the 13th ACM Conference on Electronic Commerce*, New York, USA: ACM, 674-689.
- [50] Mikeli, A., Apostolou, D. and Despotis, D. (2013). A multi-criteria recommendation method for interval scaled ratings. *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, Washington, USA: IEEE, 9-12.
- [51] Nilashi, M., bin Ibrahim, O. and Ithnin, N. (2014). Multi-criteria collaborative filtering with high accuracy using higher order singular value decomposition and Neuro-Fuzzy system. *Knowledge-Based Systems*, 60, 82-101.

- [52] Su, X. and Khoshgoftaar, T. M. (2006). Collaborative filtering for multi-class data using belief nets algorithms. *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence*, Washington, USA: IEEE, 497-504.
- [53] Choi, S. S., Cha, S. H. and Tappert, C. C. (2010). A survey of binary similarity and distance measures. *Journal of Systemics, Cybernetics and Informatics*, 8(1), 43-48.
- [54] Dunn, G. and Everitt B. S. (2004). *An Introduction to Mathematical Taxonomy*. North Chelmsford, USA: Courier Corporation.
- [55] Lathia, N., Hailes, S. and Capra, L. (2007). Private distributed collaborative filtering using estimated concordance measures. *Proceedings of the 2007 ACM Conference on Recommender Systems*, New York, USA: ACM, 1-8.
- [56] Agresti, A. (2010). *Analysis of Ordinal Categorical Data* (Vol. 656). John Wiley & Sons.
- [57] Bokde, D., Girase, S. and Mukhopadhyay, D. (2015). An item-based collaborative filtering using dimensionality reduction techniques on Mahout framework. *CoRR*, vol. *abs/1503.06562*.