

**T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI**



İLGİ METRİĞİ İLE BULANIK KÜMELEME TABANLI VIDEO ÖNERİ SİSTEMİ

Yüksek Lisans Tezi

Naciye AYDIN

Danışman

Dr. Öğr. Üyesi Gökhan KAYHAN

**SAMSUN
2023**

TEZ KABUL VE ONAYI

Naciye AYDİN tarafından, Dr. Öğr. Üyesi Gökhan KAYHAN danışmanlığında hazırlanan “İLGİ METRİĞİ İLE BULANIK KÜMELEME TABANLI VIDEO ÖNERİ SİSTEMİ” başlıklı bu çalışma, jürimiz tarafından 15.3.2023 tarihinde yapılan sınav sonucunda oy birliği ile başarılı bulunarak Yüksek Lisans Tezi olarak kabul edilmiştir.

	Unvanı Adı Soyadı Üniversitesi Ana Bilim/Ana Sanat Dalı	Sonuç
Başkan	Dr. Öğr. Üyesi Nurettin ŞENYER Samsun Üniversitesi Yazılım Mühendisliği Bölümü	☒ Kabul ☐ Ret
Üye	Dr. Öğr. Üyesi Gökhan KAYHAN Ondokuz Mayıs Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalı	☒ Kabul ☐ Ret
Üye	Dr. Öğr. Üyesi Sercan DEMİRCİ Ondokuz Mayıs Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalı	☒ Kabul ☐ Ret

Bu tez, Enstitü Yönetim Kurulunca belirlenen ve yukarıda adları yazılı jüri üyeleri tarafından uygun görülmüştür.

Prof. Dr. Ahmet TABAK
Enstitü Müdürü

BİLİMSEL ETİĞE UYGUNLUK BEYANI

Hazırladığım Yüksek Lisans tezinin bütün aşamalarında bilimsel etiğe ve akademik kurallara riayet ettiğimi, çalışmada doğrudan veya dolaylı olarak kullandığım her alıntıya kaynak gösterdiğimi ve yararlandığım eserlerin Kaynaklar'da gösterilenlerden oluştuğunu, her unsurun enstitü yazım kılavuzuna uygun yazıldığını ve TÜBİTAK Araştırma ve Yayın Etiği Kurulu Yönetmeliği'nin 3. bölüm 9. maddesinde belirtilen durumlara aykırı davranılmadığını taahhüt ve beyan ederim.

Etik Kurul Gerekli mi ?

Evet ☐

Hayır ☒

12/04/2023
Naciye AYDİN

TEZ ÇALIŞMASI ÖZGÜNLÜK RAPORU BEYANI

Tez Başlığı : İLGİ METRİĞİ İLE BULANIK KÜMELEME TABANLI VIDEO ÖNERİ SİSTEMİ

Yukarıda başlığı belirtilen tez çalışması için şahsım tarafından 12.04.2023 tarihinde intihal tespit programından alınmış olan özgünlük raporu sonucunda;

Benzerlik oranı : % 6

Tek kaynak oranı : % 1 çıkmıştır.

12/04/2023
Dr. Öğr. Üyesi Gökhan KAYHAN

ÖZET

İLGİ METRİĞİ İLE BULANIK KÜMELEME TABANLI VİDEO ÖNERİ SİSTEMİ

Naciye AYDIN

Ondokuz Mayıs Üniversitesi

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Ana Bilim Dalı

Yüksek Lisans, Mart/2023

Danışman: Dr. Öğr. Üyesi Gökhan KAYHAN

Bu tez kapsamında küme sayısını tahmin eden bulanık ve ilgi tabanlı bir kümeleme yöntemi önerilmiştir. Önerilen Bulanık C Ortalamalar Jensen Shannon (BCOJS) yöntemi, 6 kümeden ve 1000 elemandan oluşan yapay bir veri seti ile çalıştırılmıştır. Çalışmanın sonuçları, bulanık tabanlı bir kümeleme algoritması olan Bulanık C Ortalamalar (BCO), ilgi tabanlı bir yöntem olan Jensen Shannon (JS), bulanık tabanlı olasılıksal kümeleme yöntemleri olan Olasılıksal C Ortalamalar (OCO) ve Olasılıksal Bulanık C Ortalamalar (OBCO) yöntemleri ile karşılaştırılmıştır. Karşılaştırma sonuçlarını değerlendirmek için, 7 farklı küme geçerlilik indeksi ve doğruluk metriği kullanılmıştır. BCOJS, OCO, OBCO, JS ile yapılan kümeleme sonuçları doğruluk metriği ile kıyaslandığında BCO ve BCOJS yöntemi, %81.7059, %81.6864 ile diğer üç yöntemle göre daha başarılı bulunmuştur. Küme geçerlilik indeksleri ile yöntemin kümeleme yeteneği test edildiğinde ise BCO ve BCOJS yöntemi yine OCO ve OBCO yönteminden daha iyi sonuç vermiştir. BCOJS algoritmasının eşik değerini belirlemedeki zorluğunu gidermek için eşik değerinin algoritma tarafından belirlendiği adaptif BCOJS yöntemi geliştirilmiştir. Adaptif BCOJS yöntemi farklı maksimum küme sayılarında küme sayısını doğru tahmin etmiştir. Küme geçerlilik indeksi ve doğruluk metriği ile yöntemin kümeleme yeteneği test edildiğinde adaptif BCOJS yöntemi başarılı bulunmuştur. Ayrıca önerilen iki yöntemin başarımı film öneri sisteminde kullanıcıya film önermek amacıyla oluşturulan veri seti üzerinde denenmiştir. Bu amaçla film verisi, aksiyon, macera, komedi, drama ve korku türleri için dirichlet fonksiyonu ile ağırlıklandırılmış ve seçilen bu 5 film türünün özelliklerini içeren bir veri seti oluşturulmuştur. Oluşturulan bu film veri seti, BCO, BCOJS, adaptif BCOJS, OCO, OBCO, JS ile kümelenecek doğruluk metrikleri açısından karşılaştırılmıştır. Bu karşılaştırma sonucunda BCOJS yöntemi %89.4628 ile adaptif BCOJS ise %89.0593 ile diğer yöntemlerden yüksek başarı elde ettiği görülmüştür. Ayrıca önerilen iki yöntemin başarımı küme geçerlilik indeksleri açısından da BCO, OCO, OBCO yöntemleri ile karşılaştırılmıştır. Elde edilen sonuçlara göre BCOJS ve adaptif BCOJS yöntemi, uygun küme sayısını tahmin ederek kullanıcının izleyebileceği benzer filmleri türlerine göre gruplama amacını başarılı bir şekilde gerçekleştirmiştir.

Anahtar Sözcükler: Bulanık Kümeleme, Bulanık C Ortalamalar, Jensen Shannon, İlgi Metriği, Öneri Sistemleri

ABSTRACT

FUZZY CLUSTERING-BASED VIDEO SUGGESTION SYSTEM WITH THE METRIC OF INTEREST

Naciye AYDIN

Ondokuz Mayıs University

Institute of Graduate Studies

Department of Computer Engineering

Master, March/2023

Supervisor: Assist. Prof. Dr. Gökhan KAYHAN

In this thesis, a fuzzy and interest-based clustering method that estimates the number of clusters is proposed. The proposed Fuzzy C Means Jensen Shannon (FCMJS) method was run with an artificial data set consisting of 6 clusters and 1000 elements. The results of the study were compared with Fuzzy C Means (FCM), a fuzzy based clustering algorithm, Jensen Shannon (JS) an interest based method, Probabilistic C Means (PCM) and Probabilistic Fuzzy C Means (PFCM) methods, which are fuzzy-based probabilistic clustering methods. To evaluate the comparison results, 7 different cluster validity indices and accuracy metric were used. When the clustering results of FCMJS, PCM, PFCM, and JS were compared with the accuracy metric, the FCM and FCMJS methods were found to be more successful with 81.7059% and 81.6864% accuracy, respectively, compared to the other three methods. When the clustering ability of the method was tested using cluster validity indices, the FCM and FCMJS methods gave better results than the PCM and PFCM methods. An adaptive FCMJS method has been developed to overcome the difficulty of determining the threshold value in the FCMJS algorithm. When the adaptive FCMJS method was tested with different maximum cluster numbers, it predicted the correct number of clusters. When the clustering ability of the method was tested using cluster validity indices and accuracy metrics, the adaptive FCMJS method was found to be successful. In addition, the performance of the proposed methods was tested on a dataset created to recommend movies to users. The movie data was weighted using the Dirichlet function for action, adventure, comedy, drama, and horror genres to create a dataset containing the characteristics of these 5 movie genres. This movie dataset was clustered using FCM, FCMJS, adaptive FCMJS, PCM, PFCM, and JS, and compared in terms of accuracy metrics. In this comparison, the FCMJS method achieved a high success rate of 89.4628% and the adaptive FCMJS method achieved 89.0593% compared to other methods. Additionally, the performance of the proposed methods was compared with FCM, PCM, and PFCM methods in terms of cluster validity indices. According to the results, the FCMJS and adaptive FCMJS methods successfully grouped similar movies based on their genres by predicting the appropriate number of clusters that a user may watch.

Keywords: Fuzzy Clustering, Fuzzy C Means, Jensen Shannon, Interest Metric, Recommendation System

ÖN SÖZ VE TEŞEKKÜR

Lisansüstü eğitim öğretim çalışmalarımın sürecinde bilgi ve tecrübeleri ile bana yol gösteren danışman hocam Dr. Öğr. Üyesi Gökhan KAYHAN'a, tez çalışmamda öneri ve görüşleri ile bana yardımcı olan hocam Dr. Öğr. Üyesi Sercan DEMİRCİ'ye ve tüm eğitim hayatım boyunca bana katkı sağlayan değerli hocalarıma teşekkürlerimi sunarım.

Tüm hayatım boyunca maddi ve manevi desteğini bana veren çok kıymetli aileme teşekkürlerimi sunarım.

Naciye AYDİN

İÇİNDEKİLER

TEZ KABUL VE ONAYI	i
BİLİMSEL ETİĞE UYGUNLUK BEYANI	ii
TEZ ÇALIŞMASI ÖZGÜNLÜK RAPORU BEYANI	ii
ÖZET	iii
ABSTRACT	iv
ÖNSÖZ VE TEŞEKKÜR	v
İÇİNDEKİLER	vi
SİMGELER VE KISALTMALAR	viii
ŞEKİLLER DİZİNİ	ix
TABLolar DİZİNİ	x
1. GİRİŞ	1
2. ÖNERİ SİSTEMLERİ	2
2.1. Literatürde Öneri Sistemi Çalışmaları	4
2.2. Öneri Sistemi Teknikleri	8
2.2.1. İçerik Tabanlı Filtreleme Yöntemi	8
2.2.2. İşbirlikçi Filtreleme Yöntemi	9
2.2.2.1. Bellek Tabanlı İşbirlikçi Filtreleme Yöntemleri	11
Kullanıcı Tabanlı İşbirlikçi Filtreleme Yöntemleri	12
Ürün Tabanlı İşbirlikçi Filtreleme Yöntemleri	13
2.2.2.2. Model Tabanlı İşbirlikçi Filtreleme Yöntemleri	13
Matris Çarpanlarına Dayalı Yöntemler	14
Kümeleme Tabanlı Algoritmalar	14
Diğer Model Tabanlı Yöntemler	14
2.2.3. Hibrit Filtreleme Yöntemi	15
2.2.4. Diğer Tavsiye Yöntemleri	16
2.2.4.1. Bilgi Tabanlı Öneri Sistemleri	16
2.2.4.2. Demografik Filtreleme Yöntemleri	16
2.2.4.3. Hesaplamalı Zeka Tekniklerine Dayalı Tavsiye Sistemleri	17
2.2.4.4. Grup Tabanlı Öneri Sistemleri	17
2.2.4.5. Sosyal Ağ Tabanlı Öneri Sistemleri	17
2.3. Öneri Sistemlerinde Karşılaşılan Sorunlar	18
2.3.1. Sınırlı İçerik Analizi ve Çeşitlilik	18
2.3.2. Aşırı Uzmanlaşma	18
2.3.3. Soğuk Başlatma Sorunu	18
2.3.4. Seyreklik	19
2.3.5. Ölçeklenebilirlik	20
3. BULANIK KÜMELEME YÖNTEMLERİ	21
3.1. Kümeleme Yöntemleri	21
3.1.1. Bulanık C Ortalamalar Algoritması (BCO)	21
3.1.2. Olasılıksal C Ortalamalar Algoritması (OCO)	24
3.1.3. Olasılıksal Bulanık C Ortalamalar (OBCO)	24
3.1.4. Jensen Shannon	25
3.2. Kümeleme Yöntemlerinde Performans Değerlendirmesi	26
3.2.1. Bölüm Entropisi	27
3.2.2. Bölüm Katsayısı	27
3.2.3. Uyarlanmış Bölüm Katsayısı	27
3.2.4. Uyarlanmış Bölüm Entropisi	28
3.2.5. Xie Beni İndeksi	28
3.2.6. Fukuyama Sugeno İndeksi	28
3.2.7. SC İndeksi	29
3.2.8. Doğruluk Metriği	29
4. YÖNTEM	31

4.1. Bulanık C Ortalamalar Jensen Shannon Algoritması.....	31
4.1.1. BCOJS Algoritmasının Bir Örnek Üzerinden Açıklanması.....	35
4.2. Adaptifleştirilmiş Bulanık C Ortalamalar Jensen Shannon Algoritması	39
5. BULGULAR.....	42
5.1. Önerilen BCOJS Yöntemin Oluşturulan Veri Seti ile Test Edilmesi.....	42
5.2. Önerilen Yöntemin Diğer Yöntemlerle Karşılaştırılması	44
5.3. Film Veri Setinin Düzenlenmesi.....	45
5.4. BCOJS yönteminin Farklı Eşik Değerleri İle Küme Tahmini	47
5.5. BCOJS Algoritmasının Kümeleme Sonuçlarının Gösterilmesi	48
5.6. BCOJS Algoritmasının Sonuçlarının Diğer Yöntemlerle Karşılaştırılması.....	50
5.7. BCOJS Yönteminde Eşik Değerinin Adaptifleştirilmesi ile Elde Edilen Sonuçlar	54
5.7.1. Yapay Veri Seti İçin Sonuçların Değerlendirilmesi.....	54
5.7.2. Film Veri Seti İçin Sonuçların Değerlendirilmesi	56
5.8. Geliştirilen Yöntemin Öneri Sistemi Üzerinden Gösterilmesi.....	57
6. SONUÇ	59
KAYNAKÇA	61
ÖZ GEÇMİŞ.....	65



SİMGELER VE KISALTMALAR

BCO	: Bulanık C Ortalamalar
BCOJS	: Bulanık C Ortalamalar Jensen Shannon
BE	: Bölüm Entropisi
BK	: Bölüm Katsayısı
FS	: Fukuyama Sugeno
JS	. Jensen Shannon
OBCO	: Olasılıksal Bulanık C Ortalamalar
OCO	: Olasılıksal C Ortalamalar
UBE	: Uyarlanmış Bölüm Entropisi
UBK	: Uyarlanmış Bölüm Katsayısı
XB	: Xie Beni

ŞEKİLLER DİZİNİ

Şekil 2.1. Öneri Sistemlerinin Genel Şeması	3
Şekil 2.2. İçerik Tabanlı Tavsiye Sistemlerinin Genel Şeması	9
Şekil 2.3. Kullanıcı – Ürün derecelendirme matrisi örneği	10
Şekil 2.4. Bellek tabanlı işbirlikçi filtreleme yöntemlerinin ana yapısı	11
Şekil 3.1. Bulanık C Ortalamalar Algoritması Akış Şeması	23
Şekil 4.1. Bulanık C Ortalamalar Jensen Shannon Algoritması Akış Diyagramı	34
Şekil 4.2. Yapay veri seti	35
Şekil 4.3. BCO ile elde edilen küme merkezleri	35
Şekil 4.4. Adaptifleştirilmiş Bulanık C Ortalamalar Jensen Shannon Algoritması Akış Diyagramı	41
Şekil 5.1. Yapay veri setinin genel görünümü	42
Şekil 5.2. Yapay veri setinin BCOJS ile kümelenmesi sonucu oluşan küme merkezleri	43
Şekil 5.3. Yapay veri setinin BCO ile kümelenmesi sonucu oluşan küme merkezleri	44
Şekil 5.4. Kullanılan film veri setinin gerçekteki görünümü	46
Şekil 5.5. Yeni film veri setinin öznitelikleri şeması	47
Şekil 5.6. Dirichlet dağılımı uygulanan veri setinin nihai görünümü	47
Şekil 5.7. BCOJS algoritmasının çalıştırılması sonucu film türlerinin kümelenmesi	49
Şekil 5.8. BCOJS algoritmasının çalıştırılması sonucu 24 numaralı küme grubunun film dağılımları	50
Şekil 5.9. BCOJS algoritması sonucu 24 numaralı kümedeki film türlerinin yüzde dağılımı pasta grafiği	50
Şekil 5.10. (a): Yapay veri setinin adaptifleştirilmiş BCOJS ile kümelenmesi, (b): Yapay veri setinin BCO ile kümelenmesi	54
Şekil 5.11. Nobody’s Fool filmi ile benzer filmlerin listesi	58

TABLÖLER DİZİNİ

Tablo 4.1. BCOJS algoritmasının sözde kodu	33
Tablo 4.2. Üyelik değerlerinin gösterilmesi.....	36
Tablo 4.3. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisinin gösterilmesi	37
Tablo 4.4. BCO ile oluşan küme merkezlerinin gösterilmesi	37
Tablo 4.5. BCO ile oluşan küme merkezlerinin ilk güncellenmesi	37
Tablo 4.6. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisi	38
Tablo 4.7. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisi	38
Tablo 4.8. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisi	38
Tablo 4.9. BCOJS kümesi ile oluşan küme merkezleri	39
Tablo 4.10. BCOJS algoritmasının sözde kodu	40
Tablo 5.1. BCOJS ve BCO algoritmalarının küme merkezlerinin karşılaştırılması.....	44
Tablo 5.2. Doğruluk metriği ile JS, BCO, BCOJS, OCO, OBCO yöntemlerinin karşılaştırılması	45
Tablo 5.3. BCO, BCOJS, OCO, OBCO yöntemlerinin küme geçerlilik indeksleri ile kıyaslanması.....	45
Tablo 5.4. Veri setinde bulunan film türleri.....	46
Tablo 5.5. Farklı eşik değerlerinin karşılaştırılması ile küme sayısı tahmini	48
Tablo 5.6. Farklı eşik değerleri ile küme geçerlilik indekslerinin karşılaştırılması	48
Tablo 5.7. JS, BCO, BCOJS, OCO, OBCO algoritmalarının doğruluk metriği sonucu elde edilen değerlerinin karşılaştırılması	51
Tablo 5.8. BCO, BCOJS, OCO, OBCO algoritmaları için küme geçerlilik indekslerinin karşılaştırılması	52
Tablo 5.9. BCO, BCOJS, OCO, OBCO algoritmaları için küme geçerlilik indekslerinin sonuç grafikleri	53
Tablo 5.10. Yapay veri seti için adaptifleştirilmiş BCOJS yöntemi ile tahmin edilen küme sayısı	54
Tablo 5.11. Yapay veri seti için adaptifleştirilmiş BCOJS ve BCO yöntemlerinin küme merkezlerinin karşılaştırılması.....	55
Tablo 5.12. Doğruluk metriği ile JS, BCO, BCOJS, adaptif BCOJS, OCO, OBCO yöntemlerinin karşılaştırılması.....	55
Tablo 5.13. BCO, BCOJS, adaptif BCOJS, OCO, OBCO yöntemlerinin küme geçerlilik indeksleri ile kıyaslanması	56
Tablo 5.14. Film veri seti için adaptifleştirilmiş BCOJS yöntemi ile tahmin edilen küme sayısı	56
Tablo 5.15. Doğruluk metriği ile JS, BCO, adaptif BCOJS, OCO, OBCO yöntemlerinin karşılaştırılması	57
Tablo 5.16. BCO, adaptif BCOJS, OCO, OBCO algoritmaları için küme geçerlilik indekslerinin karşılaştırılması	57
Tablo 5.17. No Body's Fool filmi ile benzer 5 filmin tür dağılımı.....	58

1. GİRİŞ

Teknolojinin gelişmesi ve ulaşılabilirliğinin artması ile birlikte günümüzde film, müzik, e-ticaret siteleri gibi çevrimiçi platformlarda bulunan ürün çeşitliliği geniş bir yelpazede yayılmaktadır. Bu çeşitlilik içerisinde, kullanıcının kendisi ile ilgili ürünü seçmesi problem haline gelmiştir. Öneri sistemleri, son yıllarda artan bu devasa veri hacmini doğru şekilde işleyerek tüketicinin ilgili olduğu ürünü bulmasına yardımcı olmaya çalışır.

Bu tez kapsamında hem küme sayısını tahmin eden bulanık tabanlı bir yöntem hem de önerilen yöntem ile kullanıcıya benzer film türlerinin önerisinin yapılabileceği bir çalışma gerçekleştirilmiştir. Önerilen bu yöntemde küme sayısını tahmin etmek için Jensen Shannon ilgi metriği kullanılmıştır. Bu ilgi metriği belirlenen bir eşik değerine göre kümeleri birleştirdiği için uygun küme sayısını bulabilmektedir. Önerilen yöntem modifiye edilerek eşik değerinin algoritma tarafından dinamik bir şekilde belirlenmesini sağlayan başka bir yöntem geliştirilmiştir. Geliştirilen iki yöntem de kullanıcının izleyebileceği türlerin gruplamasını yapar. Oluşan gruplara göre kullanıcı bir filme benzer diğer filmlerin önerisini görebilir.

Bu tez çalışmasının ikinci bölümünde literatürde yer alan öneri sistemi ile ilgili yapılan çeşitli çalışmalardan, öneri sistemi tekniklerinden ve öneri tekniklerinin karşılaştığı sorunlardan bahseder. Üçüncü bölümde literatürde yer alan diğer bulanık kümeleme algoritmalarından ve bu algoritmaları değerlendirmek için kullanılan doğruluk metriği ile küme geçerlilik indeksleri yöntemlerinden bahsedilmiştir. Dördüncü bölümde geliştirilen Bulanık C Ortalamalar Jensen Shannon (BCOJS) yöntemi ve BCOJS yönteminin eşik değeri belirleme problemini gideren adaptif BCOJS yöntemi açıklanmıştır. Beşinci bölümde ise önerilen iki yöntem veri setleri ile çalıştırılarak diğer yöntemlerle kıyaslanmıştır. Son bölümde ise tez çalışmasından elde edilen sonuçlar verilmiştir.

2. ÖNERİ SİSTEMLERİ

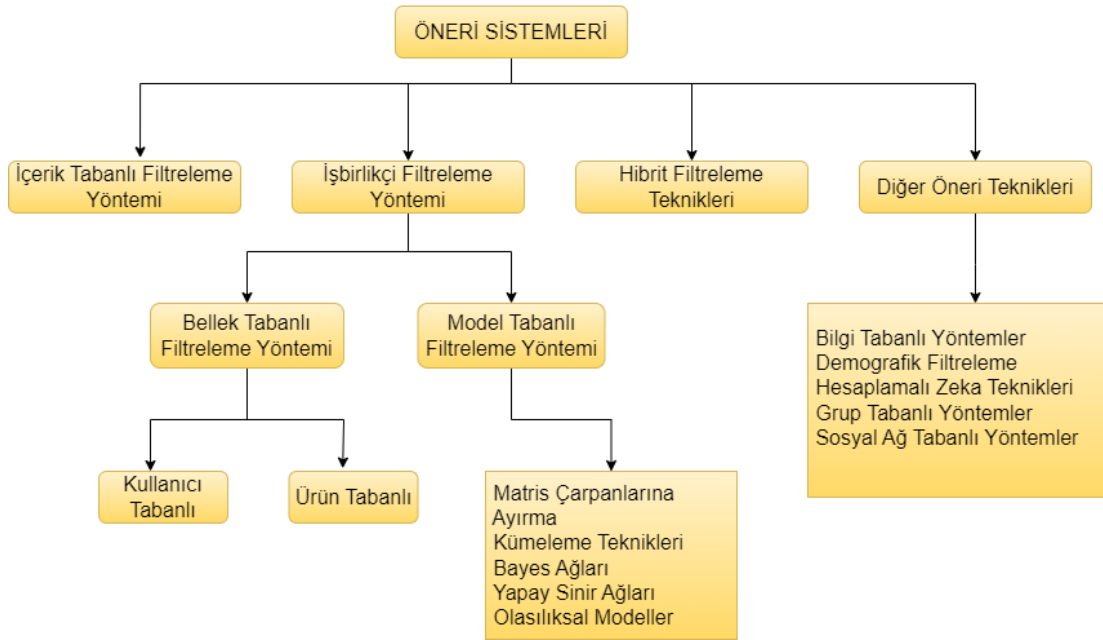
Öneri sistemleri 1990'ların ortalarında öngörülemeyen öğelerin derecelendirmelerini tahmin etmeye çalışan araştırmalar sonucunda bağımsız bir çalışma alanı olarak ortaya çıkmıştır (Adomavicius and Tuzhilin, 2005). Öneri sistemlerinin görevi kullanıcılara ilgili ürün hakkında tahmin veya tavsiyede bulunmaktır. Tahmine dayalı sistemler kullanıcının bir ürüne verdiği puanı tahmin ederek o ürünü isteyip istemeyeceği üzerine bir ihtimal belirtir. Kullanıcın muhtemelen seveceği ürünlerin bir listesinin önerilmesi ise tavsiye görevini belirtir (Jalili, et al., 2018) .

Öneri sistemleri gerçek hayatta film, müzik, belge, kitap, e-ticaret, e-öğrenme, haber, sağlık gibi birçok farklı alanda kullanılmaktadır. Günümüzde öneri sistemleri, kullanıcılara tahmin ve öneri sağlamak için ürün derecelendirmeleri, kişisel bilgiler, yorum, konum vb. gibi farklı kaynaklardan gelen birçok bilgiyi işler. İşlenen bu verilerle analizler yapar (Patel and Patel, 2020). Öneri sistemleri, teknolojinin gelişmesi ile gelişen çevrimiçi siteler ve e-ticaret ile hayatımıza daha çok girmeye başlamıştır. Bu sistemlerin günlük hayatta çeşitli katkıları bulunmaktadır. Öneri sistemlerinin hayatımıza kattığı avantajlı yönleri şu şekilde açıklanabilir:

- Ürün hakkında daha kolay ve doğru bilgi edinmesini sağlar: Bir ürün almaya karar verildiğinde; ürünün nerede hangi fiyata satıldığı, ürüne benzer diğer ürünlerin neler olduğu, ürünün pratikte kullanımı ile ilgili avantaj dezavantajlarının neler olduğu, ürünün kullanıcı memnuniyet durumu gibi bilgiler araştırılır. Öneri sistemlerinden ürünle ilgili bu tür bilgiler kolaylıkla öğrenilebilir.
- Satışın artmasına neden olur: E-ticaretin gelişmesi ile birlikte artık mağazadan satın alınan her ürün internetten de satın alınabilmektedir. Öneri sistemleri yaptıkları tavsiyeler ile satılan ürün sayısını artırır. Örneğin Amazon öneri sistemleri ile kâr durumunu yaklaşık %30 arttırmıştır (Walek and Fojtik, 2020).
- Zaman tasarrufu sağlar: Ürün almaya karar verilirken, özellikle de uzun süreli kullanımı olan ve yüksek ücret ödemesinde bulunulan ürünler için bilgi edinme sürecinde zaman harcanır. Önerilerin gerekli bilgileri doğru şekilde sunması bu süreçte vakit kazancı sağlar.

- Ürün çeşitliliği sağlar: Almak istenilen ürünün internette incelenmesi, öneri sistemlerinin ürünle ilgili çeşitli opsiyonları kullanıcı ile buluşturmasını sağlar. Bu durum kullanıcının farklı alternatiflere ulaşmasına yardımcı olur.
- Kişisel farklılığı dikkate alır: İnsanların zevk ve ilgilerinin farklı olması tercih ettikleri ürünün nitelik ve özelliklerini de farklılaştırır. Öneri sistemleri kullanıcı davranışlarından çıkarımlar yaparak müşteriye özel ürün tavsiyesinde bulunur.
- Müşterinin güvenini sağlar: Ürün hakkında yeterli ve doğru bilgiye ulaşabilen müşteri ürün çeşitliliği, zaman tasarrufu gibi çeşitli faydaları elde ettikçe öneri sistemine güvenir.

Tavsiye sistemleri ifade edilen bu faydaları sağlayarak doğru önerilerde bulunmak için büyük miktarda veri kullanır. Oluşan bu veriler ürün ve kullanıcı bilgilerinden gelir. Ürünle ilgili veriler ürünü tanımlayan spesifik bilgi içeriklerini barındırır. Kullanıcı ile ilgili elde edilen bilgiler ise genellikle etkileşimde bulunduğu ürünler ve ürünlerle ilgili yaptığı derecelendirmelerden elde edilir. Öneri sistemleri bazen ürün hakkında çeşitli bilgileri elde ederek, bazen kullanıcı ile ilgili bilgileri elde ederek, bazen de hem kullanıcı hem ürün bilgilerini harmanlayarak öneride bulunur.



Şekil 2.1. Öneri Sistemlerinin Genel Şeması

Şekil 2.1.'de öneri sistemlerinin türleri ile ilgili genel bir şema verilmiştir. Literatürde, öneri türlerine ilişkin tavsiye sistemi teknikleri farklı çalışmalarda farklı kategorilere ayrılır. Birçok çalışma öneri sistemlerini içerik tabanlı filtreleme, işbirlikçi filtreleme ve hibrit filtreleme olmak üzere üç teknik bölümde inceler (Duan et al., 2022 ; Kumar and Thakur, 2018). Bazı çalışmalar ise bu üç yönteme ek olarak demografik filtreleme yöntemleri (Bobadilla et al., 2013), bilgi tabanlı tavsiye sistemleri (Adomavicius and Tuzhilin, 2005) gibi farklı türde tavsiye sistemlerinden bahseder. Belirtilen bu tavsiye sistemlerine ilaveten, literatürde çok çeşitli öneri yöntemleri (Tarus et al., 2018 ; Lu et al., 2015) mevcuttur ve her geçen gün değişen ihtiyaçlara göre çeşitlendirilebilmektedir.

2.1. Literatürde Öneri Sistemi Çalışmaları

Literatürde öneri sistemleri ile ilgili akademik ve endüstriyel alanlarda yapılan çalışmalar, film tavsiye sistemleri (Gomez-Urbe and Hunt, 2015), müzik öneri sistemleri (Schedl et al., 2018), nesnelerin interneti (Felfernig et al., 2019), akıllı şehir (Quijano-Sánchez et al., 2020), e-turizm (Hamid et al., 2021), e-öğrenme (Tarus et al., 2018), e-ticaret (Alamdari et al., 2020), enerji verimliliği (Himeur et al., 2021), TV, kitaplar, belge vb. gibi birçok farklı alanda (Lu et al., 2015) yaygın olarak kullanılmaktadır. Günümüzde verinin çok fazla artması ile kullanıcının ihtiyaç duyabileceği her alanda öneri sistemi çalışmaları yapılmaktadır.

Geniş bir alanda yer edinen öneri sistemlerinin kullanımında ölçeklenebilirlik, seyreklik, soğuk başlatma, çeşitlilik gibi birçok farklı sorunla karşılaşmaktadır. Karşılaşılan bu sorunlar tavsiye sistemlerinin performansının ve doğruluğunun zayıflamasına neden olur. Literatürde yapılan öneri sistemi çalışmaları, yapılan tavsiyelerin geliştirilerek daha iyi bir performansa ulaşmasına ve daha doğru sonuçlar vermesine katkı sağlayarak karşılaşılan dezavantajları gidermeye çalışır.

Kullanıcıların ilgi alanlarını bulmak ve tavsiyelerin doğruluğunu arttırmak için güvene dayalı ilişkiler kullanılmıştır (Ahmadian et al., 2022 ; Ayub et al., 2020). Güvene dayalı ilişkiler, kullanıcıların tanımadıkları bir kullanıcı yerine güvenilir bir kullanıcıya ya da tanıdığı bir arkadaşına inanmaya meyilli olduğunu göstermektedir. Bu nedenle bazı e-ticaret siteleri kullanıcıların sosyal ağ profillerini kullanır. Örneğin Amazon ziyaretçilerinin Facebook profillerindeki arkadaş listelerini kullanarak kullanıcıların ilgilenebileceği ürünlerin önerisinde bulunur (Ayub et al., 2020). Ahmadian ve arkadaşları güvene dayalı ilişkilerden ve kullanıcı etiket matrislerinden

yararlanarak derin sinir ağı tabanlı bir öneri sistemi (TTDNN) ile işbirlikçi filtrelemedeki veri seyrekliği problemini ele alır (Ahmadian et al., 2022). Ahmadian ve arkadaşları önerdikleri TTDNN yönteminde yapay sinir ağı modeli olarak bilinen bir otomatik kodlayıcı kullanarak, etiket verilerinin ve güven ilişkilerinin gizli özelliklerini keşfetmeye çalışır. Elde edilen gizli özellikler ile eksik derecelendirmeler tahmin edilerek önerilerde bulunulur.

Yassine ve arkadaşları tarafından film tavsiyesi üzerine yapılan bir çalışmada işbirlikçi filtreleme K-means denetimsiz makine öğrenimi algoritmasıyla birleştirilerek hibrit bir film öneri sistemi geliştirilmiştir (Yassine, et al., 2021). Dört işlevsel modülden oluşan film öneri sisteminin ilk modülünde, önerilen yöntemin doğruluğunu arttırmak için film veri setinde temel bileşen analizi kullanılarak öznelik sayısını (film türlerini) azaltma yoluna gidilmiştir. İkinci modülde, K-Ortalamlar ile filmler belirli sayıda gruba ayrılmıştır. Sonrasında kullanıcının derecelendirmelerinin yer aldığı bir modül ve önerilerin yapıldığı son modül ile tavsiye sistemi oluşturulmuştur.

Farklı kullanıcıların aynı ürünü tercih etmesinin nedenleri farklı olabilir. Örneğin bir kullanıcı bir filmi türü ve oyuncularını sebebi ile tercih edebilir iken, başka bir kullanıcı aynı filmi beğendiği bir senarist tarafından yazıldığı için tercih edebilir. Shambour, çalışmasında kullanıcıların aynı ürünü farklı sebeplerle tercih etmelerini dikkate alarak bir öneri algoritması sunmuştur. Önerilen yöntemde kullanıcılar arasındaki doğrusal olmayan gizli ilişkileri kullanarak daha yüksek doğrulukta, çok kriterli ve derin öğrenme tabanlı bir yöntemden bahsetmiştir (Shambour, 2021).

İşbirlikçi filtreleme yönteminde kullanıcıların benzer davranışlarını anlamak ve doğru tavsiyelerde bulunmak için benzerlik metrikleri kullanılır. Kullanılan benzerlik metriğine göre önerinin doğruluğu, performansı değişkenlik gösterir. Gazdar ve Hidri çalışmasında, integral denklem, lineer diferansiyel denklemler sistemi ve lineer olmayan sistemler gibi matematiksel araçları kullanarak işbirlikçi filtreleme yönteminde kullanılan bir benzerlik ölçüsü geliştirmiş ve benzerlik ölçüsünü yaygın olarak kullanılan veri setlerinde test etmiştir. Yöntem klasik benzerlik ölçüleri ile aynı karmaşıklıkta olmasına karşın tavsiyelerin doğruluğu ile daha iyi sonuçlar elde etmiştir (Gazdar and Hidri, 2020).

Geleneksel işbirlikçi filtreleme yöntemleri öneri yapmak için kullanıcının geçmiş derecelendirmelerindeki ilişkilerden yararlanır. Geleneksel öneri sistemleri

kişilerin zaman içerisinde arkadaş, çevre, duygu durumu değişimi, trend gibi faktörlerin etkisiyle beğenilerinin değişebileceğini hesaba katmaz. Cui ve arkadaşları yaptıkları çalışma ile tavsiye sistemlerindeki zaman etkisine odaklanmıştır. Cui ve arkadaşları geliştirdikleri guguk kuşu tabanlı K-ortalamlar algoritması (TCCF) ile yeni bir işbirlikçi filtreleme yöntemi önermiştir. Çalışma, K-ortalamlar kümeleme algoritmasını kullanarak benzer kullanıcıları kümelemiş ve K-ortalamlar yönteminin başlatma sorununu optimize etmek için optimizasyon yöntemlerinde tercih edilen guguk kuşu arama algoritmasını kullanmıştır. Cui ve arkadaşları, kullanıcının zaman içindeki ilgi değişimini ölçmek için tasarladıkları zaman korelasyon katsayısını, geliştirdikleri algoritmaya uyarlayarak, kullanıcının ürünlere olan ilgisinin zaman içindeki değişimini dikkate alan işbirlikçi öneri algoritmasını önermiştir (Cui et al., 2020).

Walek ve Fojtik, işbirlikçi filtrelemeyi, içerik tabanlı yaklaşımı ve bulanık uzman sistemleri bir arada kullanarak hibrit bir film öneri çalışması yapmıştır (Walek and Fojtik, 2020).

Ahuja ve arkadaşları, K-Means ve KNN algoritmalarını kullanarak üç aşamalı bir film öneri sistemi oluşturmuştur (Ahuja, et al., 2019). Önerdikleri tavsiye sisteminin giriş bölümünde, kullanıcıların kendileri hakkında verdiği bilgiler bulunur. İşlem aşamasında ise hangi kullanıcının hangi filmi değerlendirdiği bilgisini tutan bir matris kullanır. WCSS ile uygun küme sayısını tahmin ederek K-Ortalamlar algoritması ile filmleri kümelerine ayırmıştır. Ardından kullanıcıların film kümelerine verdiği derecelendirmeler için ikinci bir matris oluşturarak, kullanıcılar ve kümelenecek matris arasındaki pearson korelasyon benzerliğini bulur ve KNN algoritması ile film derecelendirmelerini tahmin eder. Son aşamada ise kullanıcılara uygun filmler önerilir.

Başka bir çalışmada Reddy ve arkadaşları kullanıcıların izlemeyi tercih edebileceği film türlerini bulmayı amaçlamıştır (Reddy, et al., 2019). Çalışmada tür korelasyonunu kullanan içerik tabanlı filtreleme ile bir öneri sistemi tasarlamışlardır. Çalışmada farklı türde film türlerinin olduğu bir veri seti ve kullanıcıların filmleri derecelendirdiği başka bir veri seti bir arada kullanılmıştır. Bir kullanıcı belirli bir türdeki bir filme yüksek puan verdiğinde, kullanıcıya benzer türler içeren filmler önerilmiştir.

Model tabanlı yöntemlerin, bellek tabanlı yöntemlere göre daha yüksek doğruluk ve daha düşük hesaplama karmaşıklığı gibi avantajları olmasına karşın, bu yöntemlerde modelin anlaşılıp yorumlanması zordur. Bir çalışma, benzer kullanıcı ve ürün bilgilerini kullanarak, kullanıcı ve ürün tabanlı işbirlikçi filtreleme yaklaşımıyla bir bayes modeli önererek anlaşılması kolay, model tabanlı bir yöntem geliştirmeyi amaçlamıştır (Valdiviezo-Diaz et al., 2019).

Katarya ve Verma, yaptıkları çalışmada bulanık c ortalamalar (BCO) kümeleme tekniğini kullanarak, web sayfalarının kullanım örüntülerinde var olan sıralı bilgilere ve içeriğe dayalı bir web sayfası öneri sistemi sunmuştur. Çalışmada bir web kullanıcısının birden fazla kümeye ait olabileceği düşüncesiyle BCO yöntemi tercih edilmiştir. Kullanıcıların arama geçmişleri benzerse kullanıcılar aynı kümede yer alır. BCO yardımı ile hedef kullanıcı ile benzer kullanıcılar bulunur. Web sayfasında yer alan haber, sağlık, teknoloji gibi farklı kategori bilgileri ile kullanıcı bilgileri arasındaki etkileşime göre web sayfaları için ağırlık hesaplaması yapılır. Yapılan bu ağırlıklandırma sonucuna göre kullanıcıya ilgili web sayfası önerilir (Katarya and Verma, 2017).

Matris çarpanlarına ayırma tekniği hesaplama karmaşıklığı ve öneri doğruluğu ile sağladığı avantajlar nedeni ile model tabanlı tavsiye sistemlerinde tercih edilir bir yöntemdir. Hernando ve arkadaşları kullanıcı-ürün matrisinde yer alan kullanıcı derecelendirmelerini tahmin etmek için matris çarpanlarına ayırmaya dayalı yeni bir teknik önermiştir. Önerilen bu teknikte klasik matris çarpanlarına ayırma yönteminde olduğu gibi her bir kullanıcı ve ürün *k bileşeni vektörü* ile ilişkilendirilir. Bu yöntemde *k bileşen vektörleri* [0-1] aralığında olasılıksal bir yaklaşıma uyarlanmıştır. Önerilen bu olasılıksal yaklaşım ile klasik matris çarpanlarına ayırma yöntemlerinde *k* gizli faktörün yorumlanmasındaki zorluk giderilmeye çalışılmıştır (Hernando et al., 2016).

(Koohi and Kiani, 2016) yaptıkları araştırmada, Bulanık C Ortalamalar (BCO) algoritmasını ilk kez kullanıcı tabanlı işbirlikçi filtreleme yönteminde kullanarak bir film öneri sistemi çalışması yapmıştır. Çalışmada BCO algoritması, benzer kullanıcıları kümelemek için kullanılmıştır. Sonraki aşamada iki kullanıcı arasındaki benzerlik ölçülerek kullanıcıların komşuları bulur. Bulunan komşulara göre derecelendirme matrisi tahmin edilir ve sistem öneride bulunur.

Wang ve arkadaşları öneri sistemlerinde karşılaşılan veri seyrekliği ve hesaplama karmaşıklığı sorununa çözüm getiren model tabanlı hibrit bir film öneri

çalışması yapmıştır (Wang et al., 2014). Wang ve arkadaşları, hesaplamalı zeka tekniklerinden biri olan genetik algoritma ile K-ortalamlar kümeleme yöntemini optimize ederek geliştirdiği öneri yöntemini sunmuştur. Çalışma, veri seyrekliğini gidermek için temel bileşen analizi tekniğini kullanmıştır. Yapılan çalışma geleneksel işbirlikçi yöntemlerde karşılaşılan hesaplama karmaşıklığını gidermek için nesneleri kümeleme yöntemi ile gruplara ayırmıştır. K-ortalamlar yönteminin başlatma ve yerel optimuma düşmesi sorununu optimize etmek için genetik algoritma kullanmıştır.

Wang ve Wang, çalışmalarında geliştirdikleri yeni bir müzik öneri yönteminden bahsetmişlerdir. Literatürde içerik tabanlı müzik öneri sistemleri genellikle ses içerikleri ile ilgili özellikleri çıkarır ve kullanıcıya ilgilenebileceği müzikler üzerine tavsiyede bulunur. Wang ve Wang, derin inanç ağı ve olasılıksal grafik modelini kullanarak geliştirdikleri içerik tabanlı yöntem ile geleneksel sistemlerde iki aşamadan oluşan özellik çıkarımını ve öneride bulunma kısmını otomatik bir süreçle birleştirmiştir (X. Wang and Wang, 2014).

İşbirlikçi filtreleme yöntemlerinde, veri hacmi büyüdükçe hesaplama karmaşıklığı artar. Dolayısı ile bu yöntemlerde ölçeklenebilirlik bir sorundur. Bu nedenle işbirlikçi filtreleme yöntemlerinin büyük ölçekli sistemlerde kullanılması bir dezavantaj oluşturmaktadır. İşbirlikçi filtrelemenin ölçeklenebilirlik sorununu gidermek için Zhao ve Shang, kullanıcı tabanlı bir işbirlikçi filtreleme algoritmasını bir MapReduce modeli kullanarak paralelleştirmiş ve bulut bilişim platformu olan Hadoop üzerinde uygulamıştır (Zhao and Shang, 2010) .

2.2. Öneri Sistemi Teknikleri

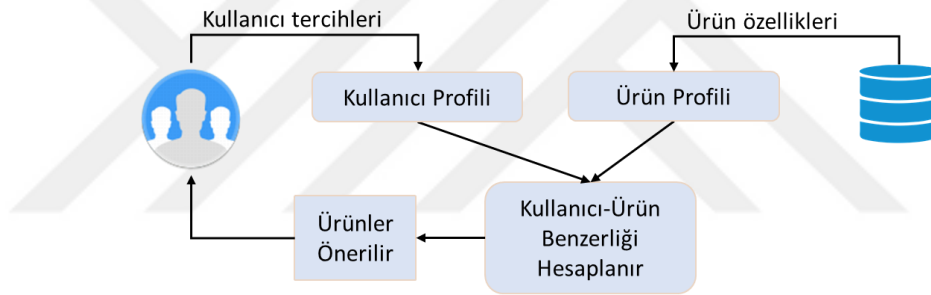
2.2.1. İçerik Tabanlı Filtreleme Yöntemi

İçerik tabanlı filtreleme yöntemleri, kullanıcıların geçmişte yaptığı seçimlerden ve öğelerin belirlenen özniteliklerinden yola çıkarak kullanıcıya ilgilenebileceği benzer ürünlerin tavsiyesini sunmayı hedefler (Adomavicius and Tuzhilin, 2005).

Geçmişten günümüze içerik tabanlı yöntemlerin gelişimi incelendiğinde, bu yöntemlerin bilgi erişimi alanına kadar dayandığı görülür. İçerik tabanlı tavsiye sistemleri, ilk başlarda belge analizlerine dayanırken, günümüzde teknolojinin gelişimi ile birlikte kullanım alanları genişlemiştir (Lops et al., 2019). Saf içerik tabanlı yöntemlerin bazı sınırlamalarının olması sebebi ile bir dönem, bu yöntemler kendi başına bir öneri sistemi geliştirmek yerine işbirlikçi filtreleme yöntemlerinin performansını arttırmak için kullanılmıştır (Lops et al., 2019; Melville et al., 2002).

Günümüzde ise bilgi kaynaklarının çok büyük bir oranda artması ve daha iyi tahminlerde bulunmak için bu bilgi kaynaklarından yararlanılan hibrit sistemlerin geliştirilmesi ile birlikte içerik tabanlı yöntemler tekrar önem kazanmıştır (Lops et al., 2019).

Şekil 2.2’de, gösterildiği üzere içerik tabanlı tavsiye sistemlerinde kullanıcı ve ürün profilleri kullanılır. Kullanıcının geçmişte tercih ettiği ürünlerden veya anketlerle alınan bilgilerinden yararlanılarak kullanıcının zevklerini, ilgi alanlarını, ihtiyaçlarını keşfetmeye yönelik bir kullanıcı profili oluşturulur. Ürün profilini oluşturmak için veri tabanından alınan ürün bilgileri ile öznitelik çalışması yapılarak ürünler hakkında bilgi edinilir. Kullanıcı ve ürün profilleri dikkate alındığında; u kullanıcısı için i ürününün $v(u,i)$ değeri ile bir benzerlik elde edilir. Kullanıcı profili ile ürünleri eşleştirerek hesaplanan kullanıcı ürün benzerliği sonucunda kullanıcıya ilgilenebileceği ürünler önerilir (Pérez-Almaguer et al., 2021).



Şekil 2.2. İçerik Tabanlı Tavsiye Sistemlerinin Genel Şeması

İçerik tabanlı yöntemler, metin, görüntü, ses gibi belirli içerik bilgilerini analiz ederler. Örneğin; içerik tabanlı tavsiye sistemleri, kullanılan bu içerik bilgileri ile kullanıcının satın aldığı, ziyaret ettiği, duyduğu, görüntülediği, beğendiği ürünlere benzer önerilerde bulunabilir (Bobadilla et al., 2013), ürün hakkındaki bir bilginin açıklanmasının kullanıcıya sistem tarafından yapıldığı yöntemlerde kullanılabilir (Gedikli vd., 2014).

2.2.2. İşbirlikçi Filtreleme Yöntemi

İşbirlikçi filtreleme sistemi insanların karar verme biçimine dayanır. Tarih boyunca insanlar bir konu üzerine karar kılmadan önce kendi deneyim ve bilgi birikiminin yanı sıra diğer insanların düşüncelerini de önemsediler (Zhao and Shang, 2010). Öneri sistemlerinin arkasındaki temel fikre göre insanlar geçmişte benzer zevklere sahip ise gelecekte de benzer zevklere ilgi duyacaktır (Kluver et al., 2018).

Bu düşünceye dayanarak işbirlikçi filtreleme yöntemleri, bir kullanıcının davranışını diğer kullanıcının davranışı ile karşılaştırarak benzerlik durumuna göre kullanıcının tercihlerini tahmin etmeye çalışır (Zhao and Shang, 2010).

İşbirlikçi filtreleme, Goldberg’ın içerik tabanlı filtrelemeye ek olarak işbirlikçi terimini ortaya attığı çalışması ile ilk kez ortaya çıkmıştır (Jalili et al., 2018). Goldberg çalışmasında, artan e-posta kullanımı ile okuyucuların çok fazla belgeye maruz kalması sorununa işbirlikçi yöntem geliştirerek çözüm bulmuştur. Çalışmada, bazı kullanıcılar belgeleri okuyarak ek açıklamalarda ve derecelendirmelerde bulunur. Yapılan açıklama ve derecelendirmelere göre e-postalar filtrelemeden geçirilir. Kullanıcılar tüm belgeleri okumak yerine kendisi ile benzer kullanıcıların okuduklarını okuyarak ilgisini çekebilecek e-postalar ile ilgilenir (Goldberg et al., 1992).

İşbirlikçi filtreleme algoritmasının ilk adımı, derecelendirme matrisinde tutulan bilgilerle kullanıcı profilinin ele edilmesidir. Derecelendirme matrisi, Şekil 2.3.’de gösterildiği gibi her satırında bir kullanıcıyı, her sütununda bir ürünü temsil eder. Matrisin satır ile sütunların kesiştiği yerde ise kullanıcı derecelendirmeleri bulunur (Zhao and Shang, 2010 ; Adomavicius and Tuzhilin, 2005). İşbirlikçi filtrelemenin ikinci adımında, bir benzerlik yöntemi kullanılarak kullanıcılar arasındaki benzerlik ve en yakın komşular bulunur. Bu şekilde kullanıcıların tercihleri, alışkanlıkları ve puanlamadıkları ürünlerin derecelendirmeleri tahmin edilmeye çalışılır (Zhao and Shang, 2010).

	Ürün 1	Ürün 2	Ürün 3	Ürün 4	Ürün 5	...
Kullanıcı 1	3		5			...
Kullanıcı 2			4		2	...
Kullanıcı 3						...
Kullanıcı 4			5		1	...
Kullanıcı 5	5			1	5	...
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.

Şekil 2.3. Kullanıcı – Ürün derecelendirme matrisi örneği

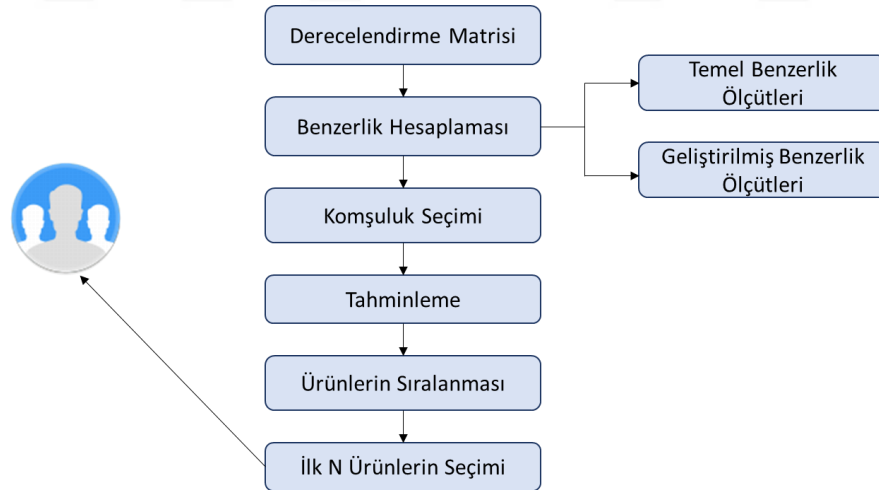
İşbirlikçi öneri sistemleri, model tabanlı öneri sistemleri ve bellek tabanlı öneri sistemleri olmak üzere iki temel yaklaşımdan oluşur (Adomavicius and Tuzhilin, 2005).

2.2.2.1. Bellek Tabanlı İşbirlikçi Filtreleme Yöntemleri

Bellek tabanlı işbirlikçi filtreleme yöntemleri kullanıcı-ürün derecelendirme matrisine göre kullanıcılar veya öğeler arasındaki benzer ilişkileri bulur. Bulunan benzerlik ilişkisine göre çevrimiçi kullanıcılara önerilerde bulunur (Yang et al., 2016).

Bellek tabanlı öneri sistemlerinde kullanıcıların ürünler üzerindeki derecelendirmelerinden yola çıkılarak derecelendirilmemiş ürünlerin puanları tahmin edilmeye çalışılır (Chen et al., 2018). Bu öneri sistemleri, kullanıcıları sınırlı içeriğe hapsedmemeleri ve tatmin edici bir doğrulukta performansa sahip olmaları nedeniyle avantajlı yöntemlerdir (Yang et al., 2016) .

Bellek tabanlı yöntemlerin kullanılmasında gerçekleşen adımlar (Yang et al., 2016) Şekil 2.4.'te verilmiştir. Şekil 2.4.'de bu yöntemlerin ilk adımında temel benzerlik ölçüsü veya geliştirilen diğer benzerlik ölçütleri ile kullanıcı ya da ürünler arasındaki benzerlik hesaplanır. Sonrasında komşuluk seçme algoritmaları ile kullanıcı veya ürünlerin komşuları seçilir. Seçilen komşuluk ve derecelendirmelere göre hesaplanan ağırlıklı ortalama sonucunda ürünlerin derecelendirmeleri tahmin edilir. Matris derecelendirmeleri tahmin edildikten sonra tavsiye edilecek ürünler için bir sıralama listesi oluşturulur. Bu listedeki belirlenen ilk N adet ürün kullanıcıya önerilir.



Şekil 2.4. Bellek tabanlı işbirlikçi filtreleme yöntemlerinin ana yapısı

Bellek tabanlı filtreleme yöntemleri kullanıcı tabanlı ve ürün tabanlı işbirlikçi filtreleme yöntemleri olarak iki kategoride incelenir. Kullanıcı-ürün derecelendirme matrisi üzerinden işlem yapılan bu yöntemlerde, kullanıcı tabanlı yöntemler kullanıcılar arasındaki benzerliği esas aldığından satırlar arasındaki ilişkiyi araştırır.

Ürün tabanlı yöntemler ise derecelendirme matrisinin sütunları arasındaki ilişkiye odaklanır (Yang et al., 2016).

Kullanıcı Tabanlı İşbirlikçi Filtreleme Yöntemleri

Geçmişte benzer beğenileri olan kullanıcıların gelecekte de benzer ürünleri tercih edebileceği fikrine dayanarak, kullanıcı tabanlı işbirlikçi filtreleme yöntemleri aktif kullanıcıya benzeyen kullanıcıların ürün derecelendirmelerini kullanarak, aktif kullanıcıların eksik derecelendirmelerini tahmin eder. Bu tahmin değerlerine göre sistem önerilerde bulunur (Chen et al., 2018).

Kullanıcı tabanlı yöntemler öneri oluşturmak için kullanıcılar arasındaki benzerliğin hesaplanması, en yakın komşuların bulunması, puanların tahmin edilmesi ve son olarak tavsiyelerin yapılması aşamalarından oluşur. Kullanıcılar arasındaki benzerliği bulmak için kullanıcıların derecelendirdiği ürünlere göre derecelendirme vektörleri bulunur. İki kullanıcının arasındaki derecelendirme vektörleri karşılaştırılarak iki kullanıcı arasındaki benzerlik bulunur. Benzerlik bulma adımında klasik benzerlik metrikleri (kosinüs, pearson katsayısı korelasyonu) kullanılır (Chen et al., 2018; Kluver et al., 2018).

Sonraki adımda eksik derecelendirilmiş ürünleri tahmin etmek için aktif kullanıcılar ile diğer kullanıcıların benzerliklerinden yola çıkılarak aktif kullanıcının komşuları bulunur. Komşuları bulmak için genellikle K En Yakın Komşu algoritması (KNN) ve eşik belirleme yöntemi kullanılır (Chen et al., 2018 ; Adomavicius and Tuzhilin, 2005). KNN algoritması aktif kullanıcıya en benzer k adet kullanıcıyı komşusu olarak seçer. Eşik belirleme yönteminde ise bir eşik değeri belirlenir. Bu eşik şartını sağlayan değerler kullanıcının komşuları olarak seçilir (Chen et al., 2018). Bulunan komşuların ürün derecelendirmeleri kullanılarak derecelendirilmemiş ürünlerin puanları tahmin edilir. Puanları tahmin edilen ürünler derecelerine göre sıralanır. Son aşamada ise aktif kullanıcıya, ilk N ögenin önerisi yapılır (Chen et al., 2018 ; Yang et al., 2016).

Kullanıcı tabanlı yöntemler öneri doğruluğu açısından tatmin edici bir sonuç elde etse de, günümüzde çevrimiçi kullanıcı sayısının çok fazla olması sonucu bu yöntemlerde kullanıcıların komşularını hesaplama maliyetinin artması sistemin karmaşıklığını arttırmıştır. Karmaşıklığın artması tahmin süresinin yavaşlamasına ve

ölçeklenebilirlik soruna yol açtığı için günümüzde kullanıcı tabanlı yöntemlerin popülerliği azalmıştır (Kluver et al., 2018).

Ürün Tabanlı İşbirlikçi Filtreleme Yöntemleri

Ürün tabanlı işbirlikçi filtreleme yöntemleri, kullanıcılar arasındaki benzerlikleri hesaplamak yerine ürünler arasındaki benzerlikleri hesaplar. Ürünlerin komşulukları üzerinden ortalama bir derecelendirme yaparak, öğelerin puanlarını tahmin etmeye çalışır (Kluver et al., 2018).

Kullanıcı derecelendirmeleri benzer olan ürünlerin, benzer türlerde olma ihtimalinin yüksek olduğu varsayımına dayanır (Yang et et al., 2016). Ürün tabanlı tavsiye yöntemleri kullanıcı tabanlı yöntemlerde olduğu gibi benzerliğin hesaplanması (ürünler arasında), en yakın komşuların bulunması ve puanların tahmin edilmesi olmak üzere üç adımda gerçekleşir. Kullanıcı tabanlı yöntemlerde kullanılan klasik benzerlik yöntemleri ile ürünler arasındaki benzerlikler hesaplanır. Sonrasında bu benzerliklere göre KNN ve ya belirlenen bir eşik değerine göre en yakın komşular bulunur. Bulunan komşuluklara göre derecelendirilmemiş ürünlerin puanları tahmin edilir. Tahmin edilen ürünler, derecelerine göre sıralanarak aktif kullanıcıya, ilk N ürünün önerisi yapılır (Chen et al., 2018).

Ürün tabanlı işbirlikçi yöntemler işlem adımları ve karmaşıklık yönü ile kullanıcı tabanlı yöntemlere benzerlik gösterirler. Ancak sistemdeki ürün sayısı kullanıcı sayısından çok daha az olduğunda, ürün tabanlı tavsiye sistemleri kullanıcı tabanlı sistemlere göre daha tercih edilir bir yöntemdir (Jalili et al., 2018).

2.2.2.2. Model Tabanlı İşbirlikçi Filtreleme Yöntemleri

En yakın komşu algoritmaları olarak da bilinen bellek tabanlı işbirlikçi filtreleme yöntemleri (Kluver et al., 2018), uygulaması kolay ve anlaşılır olmasına karşın büyük miktarda kullanıcı ve ürün verisi için ölçeklenebilir değildir. Bu dezavantajı gidermek için model tabanlı yöntemler ortaya çıkmıştır (Hernando et al., 2016). Model tabanlı yöntemler derin öğrenme, makine öğrenmesi ve veri madenciliği yaklaşımlarını kullanarak bir öğrenme aşaması ile çevrimdışı sistemde tahmin modelleri oluşturur (Yang et al., 2016). Öğrenme aşamasının tamamlanması ile ölçeklenebilir bir model çerçevesinde, eksik kullanıcı derecelendirmeleri tahmin edilmeye çalışılır.

Matris çarpanlarına dayalı algoritmalar, kümeleme tabanlı algoritmalar, çeşitli makine öğrenimi teknikleri ve derin öğrenme yöntemleri model tabanlı işbirlikçi filtrelemenin model oluşturma aşamasında kullanılan yöntemlerdir (Yang et al., 2016).

Matris Çarpanlarına Dayalı Yöntemler

Bellek tabanlı yöntemlerde kullanılan en yakın komşu algoritmaları, kullanıcılar veya ürünler arasındaki ikili ilişkileri tahmin etmede başarılıdır. Ancak birkaç farklı ürünün benzerliğinin ortak bir özelliğe göre belirlendiği veya kullanıcıların derecelendirmelerini belirli bir özelliğe olan ilgiye göre yaptıkları düşünüldüğünde en yakın komşu algoritmaları ilgili bu özelliği bilemez. Matris çarpanlarına ayırma algoritmaları ile gizli özellik modelleri kullanılarak ilgili özellik bulunur (Kluver et al., 2018).

Matris çarpanlarına ayırma algoritmaları, modelin öğrenme aşamasında bir optimizasyon problemini çözerek, kullanıcının derecelendirmelerini tahmin etmek için ‘k gizli faktörünü’ bulma fikrine dayanır. Bu algoritmalar, bellek tabanlı yöntemlere kıyasla tavsiyelerin doğruluğu konusunda daha başarılı bulunmaktadır. Ancak bu yöntemlerde çarpanlara ayırma algoritmalarında elde edilen gizli faktörün yorumlanmasının zorluğu dezavantajlı bir durum oluşturmaktadır (Hernando et al., 2016).

Kümeleme Tabanlı Algoritmalar

Kümeleme algoritmaları, işbirlikçi filtreleme yöntemlerinde kullanılarak daha düşük hesaplama karmaşıklığında ve daha yüksek doğrulukta ürün tavsiyelerinde bulunabilir (Yang et al., 2016).

Kümeleme tabanlı yöntemlerin ilk aşamasında neyin kümelenmesi gerektiği belirlenir. Bu belirlemenin ardından kümelenmesi gereken nesneler (kullanıcılar veya ürünler) matematiksel modellerle temsil edilir. Kümelenecek nesnelerin temsilinden sonra modele uygun benzerlik yöntemleri seçilir. Veri seti üzerinde belirlenen kümeleme algoritması çalıştırılarak kümeleme sonuçları elde edilir (Yang et al., 2016).

Diğer Model Tabanlı Yöntemler

Matris çarpanlarına dayalı geliştirilen algoritmalar ve kümeleme algoritmaları dışında, bayes ağları ve gizli faktör modeli (LFM) gibi olasılık modelleri, doğrusal regresyon yaklaşımları, grafiğe dayalı yaklaşımlar gibi matematiksel modellemeler de

model tabanlı öneri sistemlerinde kullanılmaktadır (Kluver et al., 2018 ; Yang et al., 2016).

2.2.3. Hibrit Filtreleme Yöntemi

İşbirlikçi filtreleme, çok sayıda kullanıcının tercihlerinden oluşan bir veri havuzundan kalıplar çıkararak öneride bulunmaya dayalı bir yöntemdir. İçerik tabanlı filtreleme yöntemi ise bir kullanıcının ürünlerle ilgili tercihlerini, ürünlerin belirlenen özniteliklerini de dikkate alarak bir tavsiye modeli oluşturmaya çalışır (Lops, et al., 2019). Her iki yöntem de tek başına kullanıldığında bazı dezavantajları barındırır.

İçerik tabanlı yöntemlerin kullanıcıların bireysel tercihlerini dikkate alarak önerilerde bulunması yeni önerilerin de eski önerilere benzemesine yol açar. İçerik tabanlı yöntemler kullanıcı tercihlerini dikkate aldığından kullanıcı davranışlarından kalıp yakalamaya çalışır. Ancak kullanıcı davranışları ürünleri değerlendirirken farklılık gösterir. Bazı kullanıcılar sadece beğendikleri ürünleri derecelendirebilir, bazı kullanıcılar ise sadece düşük puanlama yapacağı ürünleri değerlendirme eğiliminde olabilir. Kullanıcı davranışındaki farklı eğilimler, tavsiye sistemini belirli kalıplara hapsedebilir. Bu durum içerik tabanlı yöntemlerde çeşitliliği ortadan kaldıracaktır. Örneğin kullanıcı sadece beğendiği bir aktörün başrol oynadığı savaş filmlerini derecelendirirse, içerik tabanlı filtrelemenin, önerilerin çoğunda kullanıcıya o aktörün diğer filmleri ya da savaş filmleri ile ilgili tavsiyede bulunması muhtemeldir. Ancak bu durum kullanıcının sevdiği ürünlere ulaşmasını kolaylaştırırken yeni ürünler keşfetmesini engelleyerek kullanıcıyı sınırlandırır (Salter and Antonopoulos, 2006).

Yalnızca işbirlikçi filtrelemenin kullanıldığı yöntemlerde ise soğuk başlatma, veri seyrekliği, yüksek hesaplama karmaşıklığı gibi farklı dezavantajlar mevcuttur. Bu yöntemlerde çok sayıda kullanıcı veri tabanında bulunan ürünleri derecelendirene kadar sistem istenen doğrulukta öneride bulunamaz (Bobadilla et al., 2013; Salter and Antonopoulos, 2006).

Hibrit öneri sistemleri, herhangi bir öneri sisteminde karşılaşılan dezavantajları azaltarak ve avantajlardan yararlanarak daha iyi bir performans elde etmek için, içerik tabanlı filtreleme ve işbirlikçi filtreleme sistemleri gibi iki veya daha fazla öneri tekniğinin melezleştirilmesi ile kullanılan yöntemlerdir (Tarus et al., 2018).

2.2.4. Diğer Tavsiye Yöntemleri

Öneri sistemlerinin güncel çalışmaların yapıldığı popüler bir araştırma alanı olmasından dolayı, karşılaşılan problemlere bulunan çözümler de sürekli iyileştirilerek geliştirilmeye çalışılmıştır. Bu gelişmelerle birlikte öneri sistemlerinin temel tekniklerin dışında diğer yöntem ve fikirlerle de kullanılmasıyla literatürdeki çözüm teknikleri geniş bir alana yayılmıştır.

Üç temel öneri tekniği dışında, bilgi tabanlı öneri sistemleri, demografik filtreleme, güven farkındalığına dayalı, ontoloji tabanlı, hesaplama zeka tekniklerine dayalı, sosyal ağ tabanlı, yapay sinir ağlarına dayalı, evrimsel hesaplama dayalı tavsiye sistemleri gibi çeşitli tavsiye sistemleri de (Lu et al., 2015) bulunmaktadır.

2.2.4.1. Bilgi Tabanlı Öneri Sistemleri

Bilgi tabanlı yöntemler, bir ürünün kullanıcının ihtiyacını nasıl karşıladığı üzerine bilgi sahibi olan sistemlerdir. Bu sistemler, kullanıcın ürün ile ihtiyaç bilgisi arasındaki ilişki düzeyine göre öneride bulunur (Burke, 2002). Bilgiye dayalı sistemler öneride bulunurken kullanıcı hakkında bilgi, ürün hakkında bilgi ve ürün ile kullanıcının eşleşme bilgisi olmak üzere üç tür bilgiyi kullanır (Tarus et al., 2018).

Bilgi tabanlı öneri teknikleri sağlık hizmetlerinde karar desteğinde, finans uygulamalarında ve ev satışları gibi alanlarda tercih edilmektedir. Bu yöntemin kullanıldığı alanlarda spesifik bir alan bilgisi gereklidir (Zhang et al., 2021).

Bilgi tabanlı sistemler, öneride bulunmak için var olan bir bilgiyi kullandıklarından, işbirlikçi filtreleme ve içerik tabanlı filtrelemede görülen soğuk başlatma problemi ile karşılaşmazlar. Bu tavsiye sistemleri alan bilgisini kullanarak öneri yaptıkları için işbirlikçi yöntemlerde görülen veri seyrekliği ve derecelendirme sorunuyla karşılaşmaması yönüyle avantajlı yöntemlerdir. Ancak bu yöntemler bilginin doğru şekilde elde edilip kullanılabilir şekilde getirilmesindeki zorlukları yönüyle dezavantajlıdır (Tarus et al., 2018).

2.2.4.2. Demografik Filtreleme Yöntemleri

Demografik filtreleme ülke, yaş, cinsiyet, hobi vb. bakımdan ortak nitelikler taşıyan kullanıcıların benzer tercihlere ilgi duyacağı ilkesinden yola çıkılarak geliştirilmiş öneri yöntemleridir (Bobadilla et al., 2013).

2.2.4.3. Hesaplamalı Zeka Tekniklerine Dayalı Tavsiye Sistemleri

Klasik öneri yöntemlerinde karşılaşılan veri seyrekliği, soğuk başlatma, ölçeklenebilirlik gibi sorunları iyileştirmek için öneri sistemlerinde daha modern bir yaklaşım olan hesaplamalı zeka teknikleri geliştirilmiştir. Hesaplamalı zeka teknikleri tavsiye modelleri oluşturmak için; yapay sinir ağları, bulanık kümeleme ve diğer kümeleme yöntemleri, sürü zekası teknikleri, evrimsel hesaplama teknikleri, yapay bağışıklık sistemleri, bayes ağları gibi yöntemleri kullanır (Abbas et al., 2015).

Bulanık Tabanlı Tavsiye Sistemleri: Bulanık tabanlı tavsiye sistemleri bulanık kümeleme teorisine dayanır. Bu tavsiye yöntemi, bulanık kümeleme mantığında olduğu gibi bilginin eksik veya belirsiz olması durumunda belirsizlikle başa çıkabilmek için tercih edilir (Abbas et al., 2015). Bulanık teknikler, bir kullanıcının birden fazla gruba dahil olabileceği (Katarya and Verma, 2017) durumlar için kullanılabilir. Örneğin, kullanıcıların ilgilendikleri filmlerin türüne göre gruplandırıldığı bir tavsiye sisteminde bir kullanıcı hem romantik film hem de bilim kurgu filmi grubuna veya başka film gruplarına dahil olabilir. Bu örnek gibi kesin olmayan kullanıcı tercihlerinde bulanık tabanlı tavsiye sistemleri tercih edilir.

2.2.4.4. Grup Tabanlı Öneri Sistemleri

Grup tabanlı tavsiye sistemleri ürün tavsiyesini kişi yerine kullanıcı gruplarına yapar. Ortak bir kitlenin seyahat, festival, tiyatro gibi aynı etkinliklere katıldığı sosyal olaylarda bu öneri sistemleri tercih edilir (Tarus et al., 2018). Grup tavsiye sistemleri, ortak bir kitle oluşturmak için sosyal seçim teorisi ve karar verme süreçlerinden ilham alır (Lu et al., 2015).

2.2.4.5. Sosyal Ağ Tabanlı Öneri Sistemleri

Sosyal ağ tabanlı tavsiye sistemleri, web tabanlı sistemlerin aşırı büyümesi sonucu ortaya çıkan öneri yöntemleri arasındadır. Bu yöntemler kullanıcı deneyimlerini iyileştirmek için, kullanıcıların sosyal medya profillerinde yer alan kullanıcı etkileşimleri, yorumları, diğer kullanıcılar ile olan ağ ilişkisi bilgilerini analiz ederek öneride bulunan tavsiye sistemleridir (Tarus et al., 2018 ; Lu et al., 2015).

Güvene dayalı yöntemlerde, kullanıcıların tanımadıkları bir kullanıcı yerine tanıdığı veya bir şekilde bağlantısı olan kullanıcıya inanma olasılığı daha yüksektir (Ayub et al., 2020). Bu sebeple bir ürünün satın alınma olasılığını arttırmak için, sosyal

ağ tabanlı yöntemlerde güvene dayalı yöntemler kullanılır (Lu et al., 2015). İşbirlikçi yöntemlerde veri seyrekliği sorunu istenen komşuların bulunamaması sorununa yol açar. Sosyal ağ tabanlı yöntemlerde kullanıcıların sosyal ilişkileri üzerine elde edilen bilgiler yardımı ile işbirlikçi yöntemlerin komşu bulma sorunu iyileştirilmeye çalışılır (Lu et al., 2015) .

2.3. Öneri Sistemlerinde Karşılaşılan Sorunlar

2.3.1. Sınırlı İçerik Analizi ve Çeşitlilik

İçerik tabanlı yöntemler, içerik hakkında bilgi edinmek için belirli öznitelikleri kullandığından bu yöntemlerin yaptığı öneriler bu özniteliklerle sınırlı olur. İki farklı kümenin aynı öznitelikler ile temsil edilmesi öneri yapılacak ürünlerin ayırt edilememesi gibi bir soruna yol açar. Örneğin bir döküman önerisinde, içerik tabanlı yöntem kullanıldığında ağırlıklı olarak öznitelik değerleri anahtar kelimelerden oluşur. Bu durum iyi bir makale ile kötü bir makalenin ayırt edilememesi gibi bir soruna yol açar (Adomavicius and Tuzhilin, 2005). Bu nedenle doğru ve yeterli sayıda öznitelikler bulundurmak için iyi bir öznitelik çıkarma çalışması yapılmalıdır.

Benzer sorunla işbirlikçi filtreleme yöntemlerinde de karşılaşılır. Kullanıcıya yalnızca yüksek puanlı popüler ürünlerin tavsiyesinde bulunan işbirlikçi yöntemlerde önerilerin çeşitliliği kısıtlanır. Kullanıcı popüler ürünleri farklı kaynakları kullanarak da elde edebileceğinden, öneri sistemleri bulunması güç ve kullanıcının ilgi alanı olan ürünleri keşfetmesine yardımcı olma amacını yerine getiremez (Chen et al., 2018).

2.3.2. Aşırı Uzmanlaşma

İçerik tabanlı sistemler kullanıcının profilinde bulunan ürünlerle benzer ürünlere odaklandığından dolayı yapılan tavsiyeler de kullanıcının değerlendirdiği ürünlere benzer içerikte önerilerle sınırlı kalır. Kullanıcının sürekli benzer içerikte ürünlere hapsolması onu çeşitlilikten yoksun bırakır. Örneğin aşırı uzmanlaşan bir sistemde kullanıcıya aynı olayı anlatan farklı bir haber makalesinin önerilmesi gibi bir sorunla karşılaşılabilir. Kullanıcıya sunulan öneriler homojen bir alternatif kümesinden ziyade belirli çeşitliliğin sağlandığı seçeneklerle sunulmalıdır. (Adomavicius and Tuzhilin, 2005).

2.3.3. Soğuk Başlatma Sorunu

Öneri sistemlerinde sisteme yeni bir kullanıcı veya yeni bir ürün dahil olduğunda ya da sistem ilk kez başlatıldığında, ürün ve kullanıcı bilgilerinin eksikliği nedeni ile

sistem güvenilir tavsiyelerde bulunamaz. Bu durum öneri sisteminin tavsiyede bulunmasını engeller veya önerilerin doğruluğunu negatif yönde etkiler.

Yeni kullanıcı sorunu: Öneri sistemlerinde, sisteme yeni bir kullanıcı gelmesi durumuyla ya da sistemdeki bir kullanıcının az miktarda ürünü değerlendirmesi gibi bir durumla sıklıkla karşılaşılır. İçerik tabanlı ve işbirlikçi filtreleme yöntemleri, kullanıcının geçmiş deneyimlerinden yola çıkarak önerilerde bulunduğundan, yeterli değerlendirmelerin eksikliğinde bu yöntemler kullanıcıyı tanımadığı için doğru önerilerde bulunamaz (Adomavicius and Tuzhilin, 2005).

Yeni ürün sorunu: İçerik tabanlı yöntemlerde ürün profillerinden çıkarılan ürün bilgisi ile ürünler öneride kullanılabilir hale gelir. Ancak işbirlikçi filtreleme yöntemleri, kullanıcı derecelendirmelerini analiz ederek ürün önerisinde bulunur. Sisteme yeni bir ürün eklendiğinde kullanıcılar ürüne yeterli derecelendirmeyi yapana kadar işbirlikçi filtreleme ürün tavsiyesinde bulunamaz (Adomavicius and Tuzhilin, 2005).

2.3.4. Seyreklik

Öneri sistemlerinde kullanılan veri setlerinin yüz binlerce hatta milyonlarca ürünün verisini içerebildiği bilinmektedir. Büyük miktarda veriye karşın kullanıcının derecelendirdiği ürün sayısı düşünüldüğünde bu oran çok küçük kalmaktadır. Derecelendirme verileri, kullanıcı-ürün derecelendirme matrislerinde tutulduğundan, matrisin büyük bir kısmı derecelendirilmemiş verilerden oluşur. Bu durum literatürde seyreklik sorunu olarak geçmektedir (Adomavicius and Tuzhilin, 2005).

Seyreklik sorunu, ürün sayısının çok fazla olduğu e-ticaret sistemlerinde sıklıkla görülmektedir. Örneğin Netflix'in 2006 yılında düzenlediği öneri sistemi yarışmasında 17.000 film ve 500.000 kullanıcıdan oluşan gerçek veri setinin içerisinde 100 milyon derecelendirme yapılmıştır. Bu veri setinin derecelendirme matrisindeki derecelendirilmemiş verilere bakıldığında veri seyrekliğinin %98.82 olduğu hesaplanmıştır (Duan et al., 2022). Yani derecelendirme matrisinin çok büyük bir kısmı eksik verilerden oluşmaktadır.

İşbirlikçi filtreleme yöntemlerinde kullanıcı derecelendirmeleri, etkili bir tahmin için önemlidir. Ancak veri seyrekliği bu tahminin doğruluğuna negatif yönde bir etki yapar. Örneğin film veri setlerinde sadece birkaç kişinin derecelendirdiği filmler bulunmaktadır. Öneri sistemleri az sayıda derecelendirmeye sahip filmleri yüksek

puan alsa bile önermeyebilir. Ayrıca seyreklik durumunda farklı kullanıcıların ortak derecelendirdiği ürün sayısı çok azdır (Duan et al., 2022). Ancak bazı kullanıcı tercihleri sıra dışıdır ve az sayıda kişinin seçebileceği popüler olmayan, hatta çok az sayıda değerlendirilen ürünleri beğenebilir. Seyreklik sorunu sistemin bu tür kullanıcılara doğru önerilerde bulunmamasına sebep olur (Adomavicius & Tuzhilin, 2005).

2.3.5. Ölçeklenebilirlik

Çevrimiçi sistemlerde veri miktarı geometrik olarak artarken aynı zamanda kullanıcılara ürün tavsiyesinde bulunulur (Chen et al., 2018). Bellek tabanlı yöntemler olarak bilinen çevrimiçi öneri sistemleri, en yakın komşu algoritmalarının uygulanması ile oluşturulan yöntemlerdir (Kluver et al., 2018). En yakın komşu algoritmaları her bir kullanıcının veya ürünün komşuluklarını hesapladığı için büyük verilerde bu yöntemler yüksek hesaplama karmaşıklığı oluşturur. Bu durum bellek tabanlı yöntemlerde ölçeklenebilirlik sorununa yol açar.

3. BULANIK KÜMELEME YÖNTEMLERİ

3.1. Kümeleme Yöntemleri

3.1.1. Bulanık C Ortalamalar Algoritması (BCO)

Bulanık C Ortalamalar algoritması, bulanık mantık ilkesi esas alınarak geliştirilmiş amaç fonksiyonu temelli bir kümeleme algoritmasıdır. Algoritma ilk kez (Dunn, 1974) tarafından geliştirilmiş, sonrasında ise algoritmanın amaç fonksiyonunun genelleştirilmiş hali (Bezdek, 1984) tarafından geliştirilmiştir (Ruspini, et al., 2019).

BCO algoritmasında her bir veri örneğinin birden fazla kümeye ait olabilmesini sağlayan bir üyelik değeri bulunur. Veri elemanları her grupta belirli bir üyelik derecesi bulundurarak birden fazla kümeye ait olur. Farklı gruplara ait üyelik değerlerinin toplamı 1 olmalıdır. Eşitlik (3.1)'de gösterilen amaç fonksiyonu en küçük kareler yönteminin genelleştirilmesi ile oluşturulmuştur. Amaç fonksiyonu temelli bir bulanık kümeleme yöntemi olan BCO algoritması, amaç fonksiyonunu olabilecek en küçük değerde tutmaya çalışır. Bu değer küçüldükçe, küme içi benzerlik ve kümeler arası benzersizliğin yükseldiği bir kümeleme gerçekleştirilmiş olur.

$$J_{BCO}(U, v) = \sum_{i=1}^N \sum_{j=1}^k u_{ij}^m d_{ij}^2 \quad (3.1)$$

Denklem (3.1)'de BCO algoritmasının amaç fonksiyonu verilmiştir. Bu denklemde; U değeri üyelik matrisini, v parametresi küme merkezlerini, N parametresi veri sayısını, k parametresi $2 \leq k < n$ aralığındaki küme sayısını, u_{ij} değeri (i, j) verisinin k . kümeye ait olma olasılığını, d_{ij} parametresi k . kümenin merkezleri ile veri noktası arasındaki öklid mesafesini, m değeri ise $(1 \leq m < \infty)$ aralığında değer alan bulanıklık parametresini göstermektedir.

Bulanıklık parametresi kümelemenin belirsizliğini gösterir. Bulanıklık parametresinin belirlenmesi ile ilgili (Pal and Bezdek, 1995) tarafından yapılan çalışmada optimum m değerinin $[1.5 - 2.5]$ aralığında olması gerektiği ve deneysel verilerde optimum m değerinin 2 olduğu belirtilmiştir. Bu parametre değeri 1'e yakın bir değerde seçildiğinde kümeleme sert kümelemeye benzerken, parametre 2 veya 2'ye

yaklaşan bir değerde seçildiğinde ise daha yumuşak bir bölümlleme gerçekleşir. Bir çok çalışmaya göre uygun m değeri 2 olarak belirlenmiştir (Zhou, et al., 2014).

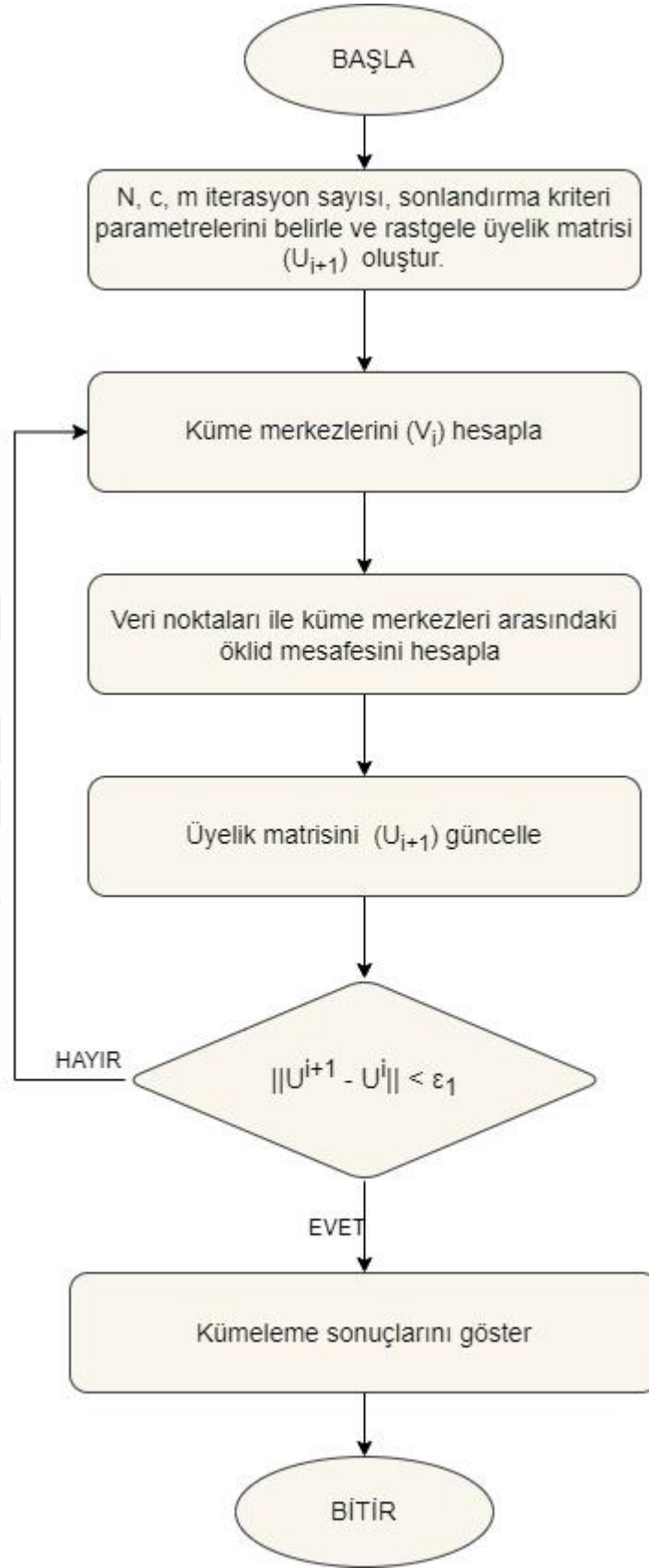
$$V_{ij} = \frac{\sum_{i=1}^N u_{ij}^m x_{ij}}{\sum_{i=1}^N u_{ij}^m} \quad (3.2)$$

Eşitlik (3.2)'de küme merkezlerinin hesaplanması verilmiştir. Bu denklemde u_{ik} i. veri örneğinin k kümesine olan üyelik değeridir. i. veri örneğinin tüm kümelere olan üyeliklerinin toplamı ise 1 'e eşittir. Burada x_{ij} veri noktasını temsil etmektedir.

$$U_{ij} = \frac{1}{\sum_{j=1}^k \left(\frac{d_{ij}}{d_{ji}} \right)^{\frac{2}{m-1}}} \quad (3.3)$$

Eşitlik (3.3)'te N sayıda veri örneği ve k sayıda kümeden oluşan değerler için üyelik matrisinin güncellenmesi verilmiştir. Şekil 3.1.'de BCO algoritmasının akış diyagramı gösterilmiştir. BCO algoritması küme sayısı, bulanıklık parametresi, iterasyon sayısı, sonlandırma kriteri ve üyelik matrisinin oluşturulması ile başlar. Ardından küme merkezleri hesaplanır. Sonrasında her bir veri elemanının her kümeye üyelik derecesi atanır. Her bir iterasyonda küme merkezleri, üyelik dereceleri güncellenir. Bu yineleme, ardışık iki iterasyon arasındaki üyelik matrisi farkının sonlandırma kriterinden küçük olduğunda veya yineleme sayısı maksimum iterasyona ulaştığında son bulur. Her bir iterasyonda sağlanan güncellemeler ile amaç fonksiyonu minimize edilir (Zhou, et al., 2014).

BCO algoritmasının, amaç fonksiyonuna bağlı çalışan en iyi yöntemlerden biri olması en popüler algoritmalarından biri olma avantajını sağlamıştır (Nayak, et al., 2015). Buna ilaveten algoritmanın kolay anlaşılır ve uygulanabilir olması, verileri doğru şekilde kümelemesi, tutarlı üyelik değerleri ile kolay yorumlanabilir olması gibi çeşitli faydaları kullanım avantajı sağlamıştır. Sağladığı bu avantajların dışında BCO algoritması, bazı eksiklikleri bulundurmaktadır. BCO algoritması başlangıç parametrelerine bağlı bir algoritmadır. Algoritmanın başlangıç parametrelerinin yanlış seçilmesi kümeleme sonucunu değiştirebilir. Ayrıca doğru kümeleme için küme sayısının bilinmesi gereklidir.



Şekil 3.1. Bulanık C Ortalamalar Algoritması Akış Şeması

3.1.2. Olasılıksal C Ortalamalar Algoritması (OCO)

Olasılıksal c ortalamalar (OCO) algoritması bulanık c ortalamalar algoritmasında karşılaşılan aykırı değerlerin hesaba katılmasını önlemek için geliştirilmiş bir kümeleme tekniğidir (Krishnapuram and Keller, 1993). BCO yönteminde var olan kümeler arasındaki üyelik değerlerinin toplamının 1 olma kısıtı OCO yönteminde yoktur. OCO algoritmasında üyelik değerleri *uyumluluk* ve ya *tipiklik* olarak yorumlanır. Üyelik değerleri, sınıflara ait noktaların olasılık dereceleri, yani noktaların uyumlulukları olarak düşünülür. OCO algoritmasının amaç fonksiyonu eşitlik (3.4)'te verilmiştir.

$$J_{oco}(U, V) = \sum_{i=1}^k \sum_{j=1}^N u_{ij}^m d_{ij}^2 + \sum_{i=1}^k n_i \sum_{j=1}^N (1 - u_{ij})^m \quad (3.4)$$

Eşitlik (3.4)'te U değeri üyelik matrisini, V parametresi küme merkezlerini, N parametresi veri sayısını, k parametresi küme sayısını, u_{ij} değeri $0 \leq u_{ij} \leq 1$ aralığındaki (i, j) verisinin k . küme ile olasılık derecesini, d_{ij} parametresi k . kümenin merkezleri ile veri noktası arasındaki öklid mesafesini, m değeri bulanıklık parametresini, n değeri tipiklik parametresini göstermektedir. n değerinin hesaplanması eşitlik (3.5)'te, U_{ij} değerinin hesaplanması ise eşitlik (3.6)'da verilmiştir.

$$n_i = \frac{\sum_{j=1}^N u_{ij}^m d_{ij}}{\sum_{j=1}^N u_{ij}^m} \quad (3.5)$$

$$U_{ij} = \frac{1}{1 + \frac{(d_{ij})^{\frac{2}{m-1}}}{n_i}} \quad (3.6)$$

3.1.3. Olasılıksal Bulanık C Ortalamalar (OBCO)

Olasılıksal bulanık c ortalamalar algoritması (OBCO), Pal ve arkadaşları tarafından 2005 yılında geliştirilmiştir (Pal et al., 2005). OBCO yöntemi BCO ve OCO algoritmalarının hibritleştirilmesi ile geliştirilen bir yöntemdir. OBCO algoritması BCO algoritmasında karşılaşılan gürültü duyarlılığı problemini ve OCO algoritmasında karşılaşılan çakışan kümeler sorununu iyileştirmeye odaklanır. Eşitlik

(3.7)'de OCO ve BCO yöntemlerinin melezleştirilmesi ile elde edilen OBCO algoritmasının amaç fonksiyonu verilmiştir.

$$J_m(U, T, V) = \sum_{i=1}^k \sum_{j=1}^N au_{ij}^m + bt_{ij}^n d_{ij}^2 + \sum_{i=1}^k \&_i \sum_{j=1}^N (1 - t_{ij})^n \quad (3.7)$$

Eşitlik (3.7) $a > 0, b > 0, m > 1, n > 1$ ve $0 \leq u_{ij}, t_{ij} \leq 1$ kısıtlarını sağlamalıdır. Eşitlikte yer alan a ve b değerleri bulanık üyelik ve tipiklik değerlerinin göreceli önemini tanımlayan sabitlerdir. Üyelik değerleri u_{ij} , küme merkezi ile veri noktası arasındaki öklid uzaklığı d_{ij} , tipiklik üssü n , küme varyansı ise $\&$ ile temsil edilir. OBCO algoritmasının amaç fonksiyonunu minimize etmek için U üyeliği, T tipikliği, V küme merkezleri eşitlik (3.3), (3.8), (3.9) 'da verilen ifadeler ile güncellenir.

$$T_{ij} = \frac{1}{1 + \left(\frac{b d_{ij}^2}{\&_i}\right)^{1/(n-1)}} \quad (3.8)$$

$$V_i = \frac{\sum_{j=1}^N (au_{ij}^m + bt_{ij}^n) x_j}{\sum_{j=1}^N au_{ij}^m + bt_{ij}^n} \quad (3.9)$$

3.1.4. Jensen Shannon

Jensen Shannon ıraksaması (JSD), (Lin, 1991) tarafından önerilmiştir. Bu metrik iki olasılık arasındaki farkı ölçmek için kullanılır. Jensen Shannon mesafesi, eşitlik (3.10)'da gösterilen Kullback Leibler (KL) sapmasını kullanarak simetrik bir hesaplama yapar.

$$KL(X||Y) = \sum_{i=1}^N X_i \log \frac{X_i}{Y_i} \quad (3.10)$$

$$JSD(P||Q) = \frac{1}{2} KL(P||M) + \frac{1}{2} KL(Q||M) \quad (3.11)$$

$$M = \frac{1}{2} (Q + P) \quad (3.12)$$

$$JSD(P||Q) = JSD(Q||P) \quad (3.13)$$

Eşitlik (3.11) 'de P ve Q JSD hesaplamasında kullanılan iki olasılık dağılımıdır. Jensen Shannon ıraksaması, P ve M olasılığı ile Q ile M olasılığı arasındaki Kullback Leibler sapmasının ortalaması alınarak hesaplanır. Burada logaritma değeri 2 tabanında kullanılır. M ise eşitlik (3.12)' de gösterilen P ve Q değerlerinin ortalamasını ifade eder. Jensen Shannon ıraksamasının önemli bir özelliği P ve Q iki olasılık arasındaki ilişkinin Q ve P olasılığı arasındaki ilişki değerine eşit olmasıdır. Eşitlik (3.13) P ve Q olasılığı ile Q ve P olasılığının JSD ile hesaplandığında aynı sonucu verdiğini gösteren simetrik bir benzerlik ölçüsünü ifade etmektedir. P ve Q değerlerinin olasılıkları toplamı 1 olmalıdır. Eşitlik (3.11)'de gösterilen JSD değerinin karekökünün hesaplanması ile Jensen Shannon mesafesi hesaplanır . Jensen Shannon mesafesi 0 ile 1 arasında değer alır. Değerin 0'a yakın olması iki olasılık arasındaki (P ve Q) yakınlığın yani ilginin yüksek olduğunu, 1'e yakın olması ise iki olasılık arasındaki ilginin düşük olduğunu gösterir (Aydın ve Kayhan, 2021).

3.2. Kümeleme Yöntemlerinde Performans Değerlendirmesi

Küme geçerlilik indeksleri ve doğrulama metrikleri bir kümeleme yönteminin küme kalitesini değerlendirmek için geliştirilen matematiksel fonksiyonlardır. Kümeleri doğrulamak için geliştirilen bu yöntemler kümeler arası uyum ve kümeleri ayırmanın performansı üzerine odaklanmıştır. Kümeler arası uyum, bir küme içindeki veri noktalarının kompaktlığı ile ilgilidir. Kompakt kümeler oluşturularak küme içi benzerlik maksimum seviyede tutulmaya çalışılır. Ayırma ise herhangi iki farklı kümenin birbirine minimum seviyede benzer olması anlamına gelir. Bunlar dışında küme geçerlilik indeksleri bir kümeleme yönteminin optimal küme sayısını değerlendirmek için de kullanılır (Sengupta et al., 2011).

Tek bir performans metriği kullanmak bir yöntemin tutarlılığından bahsetmek için yeterli olmayabilir. Bu sebeple bu tez kapsamında bulanık kümelemede kullanılabilecek çeşitli performans yöntemlerinden bahsedilmiştir.

3.2.1. Bölüm Entropisi

Bölüm entropisi (BE), kümeleme sonucunda elde edilen bölümlene matrisindeki bulanıklık değerini skaler bir ölçü ile gösterir. BE, $[0, \log k]$ aralığında değer alır. BE değeri 0'a yakın değerlerde kümelemenin iyi yapıldığını gösterirken, üst sınıra yakın değerlerde kümeleme yapısının iyi olmadığını veya yöntemin küme oluşturmadaki başarısının düşük olduğunu gösterir (Ozer, 2005). Eşitlik (3.14)'te BE değerinin hesaplanması verilmiştir (Liu et al., 2019). (3.14)'te verilen eşitlikte N veri sayısını, u üyelik matrisini, k küme sayısını belirtir.

$$V_{BE} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^k u_{ij} \log u_{ij} \quad (3.14)$$

3.2.2. Bölüm Katsayısı

Bölüm katsayısı (BK), kümeleme sonucunda oluşan bölümlene matrisindeki ikili bulanık kesişimi en aza indirmeye dayalı bir performans ölçüsüdür. Bu performans ölçüsü ile bölümlene matrisindeki üyelik paylaşımı sayısal bir değerle ifade edilir. BE değeri k küme sayısını göstermek üzere $[1/k, 1]$ aralığında yer alır. BK değeri, 1'e yaklaştıkça kümeleme performansı iyileşir (Ozer, 2005). Bölüm katsayısı, eşitlik (3.15)'te verilmiştir. (3.15)'te verilen eşitlikte N veri sayısını, u üyelik matrisini temsil eder (Liu et al., 2019).

$$V_{BK} = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^k (u_{ij})^2 \quad (3.15)$$

3.2.3. Uyarlanmış Bölüm Katsayısı

Hem BE, BK kümelemedeki bulanıklık miktarını ölçer. Ancak araştırmalar BE, BK değerlerinin küme sayılarındaki değişikliklerde monoton eğilim gösterdiğini söylemektedir. Bu monotonluğu gidermek için BK yöntemi modifiye edilerek uyarlanmış bölüm katsayısı yöntemi (UBK) geliştirilmiştir (Ozer, 2005). UBK metriğinin hesaplanması eşitlik (3.16)'da verilmiştir (Liu et al., 2019). Eşitlikte yer alan V_{BK} bölüm katsayısını, k ise küme sayısını belirtmektedir.

$$V_{UBK} = 1 - \frac{k}{k-1}(1 - V_{BK}) \quad (3.16)$$

V_{UBK} değeri $[0,1]$ aralığında yer alır. Bu değer 0'a yaklaştıkça maksimum belirsizliği, 1'e yaklaştıkça iyi bir bölümlenimin olduğunu gösterir.

3.2.4. Uyarlanmış Bölüm Entropisi

Uyarlanmış bölüm entropisi (UBE), monoton eğilim gösteren BE değerinin modifiye edilmiş halidir (Ozer, 2005). UBE belirli sayıda küme ile oluşturulmuş düzensizlik değerini tahmin eder. UBE 1'e yakın değerler için kümelemede düzensizliğin hakim olduğunu, 0'a yaklaşan değerler için iyi bir kümeleme yapıldığını gösterir (Schenatto et al., 2017). UBE değerinin hesaplanması, eşitlik (3.17)'de verilmiştir. (3.17)'de verilen eşitlikte; k küme sayısını, BE bölüm entropisini temsil eder.

$$UBE = 1 - \frac{V_{BE}}{\log k} \quad (3.17)$$

3.2.5. Xie Beni İndeksi

Xie Beni (XB) indeksi, küme kompaktlığı ve küme ayrıklığı olmak üzere iki farklı özelliğe odaklanan bir geçerlilik indeksidir. XB indeksinde pay kısmı kümelemenin kompaktlığını gösterirken, payda kısmı kümeler arası ayrımın değerini gösterir. XB indeksi küçük değerler için kümelemenin daha iyi gerçekleştiğini gösterir. Bu yöntemin dezavantajı büyüyen küme sayılarında monoton bir azalma göstermesidir (Cho and Yoo, 2006). Eşitlik (3.18)'de, XB indeksinin hesaplanması verilmiştir.

$$XB(U, V, X) = \frac{\sum_{i=1}^k \sum_{j=1}^N (u_{ij})^2 ||V_i - X_j||^2}{nd_{min}^2}, \quad d_{min} = \min ||v_i - v_j|| \quad (3.18)$$

Eşitlik (3.18)'de, N veri sayısını, u_{ij} bölümeleme matrisini, v_i küme merkezlerini, X_j veri noktasını, k küme sayısını göstermektedir.

3.2.6. Fukuyama Sugeno İndeksi

Fukuyama Sugeno (FS) indeksi, küme içi kompaktlığın ve kümeler arası ayrımın ölçülmesi için geliştirilen bir yöntemdir. FS değeri ne kadar küçükse kümelemenin kompaktlığı ve kümeler arası ayrım o kadar iyidir (Ozdemir ve Kaya,

2019). Küme sayısı büyüdükçe FS değerinde monoton azalan bir eğilim görülmektedir (Cho and Yoo, 2006).

$$FS = J_{BCO} - K_M \quad (3.19)$$

$$K_M = \sum_{i=1}^k \sum_{j=1}^N (u_{ij})^m (v_i - v_{ort}) \quad (3.20)$$

FS geçerlilik indeksinin hesaplanması eşitlik (3.19) 'da gösterildiği gibidir. Eşitlik (3.19) 'da J_{BCO} BCO yönteminde eşitlik (3.1)'de verilen amaç fonksiyonunu temsil eder. Burada J_{BCO} kompaktlığın bir ölçüsüdür. Eşitlik (3.20)'de verilen K_M ise küme temsilcisinin mesafesini ölçer. Eşitlik (3.20)'de k küme sayısını, N veri sayısını, u_{ij} bölümeleme matrisini, v_i küme merkezlerini, v_{ort} küme merkezlerinin ortalamasını temsil eder.

3.2.7. SC İndeksi

SC indeksi, küme kompaktlığını ve ayrışmasını birleştiren bir geçerlilik indeksidir. SC değeri ne kadar yüksekse kümeleme o kadar iyidir. SC indeksi kümeler arası ayrımın küme kompaktlığına oranı hesaplanarak bulunur (Bouguessa et al., 2006).

$$SC(U, V, X) = \frac{iz(S_B)}{\sum_{i=1}^k iz(\sum i)}, iz(A) = \sum_i A_{ii} \quad (3.21)$$

$$S_B = - \sum_{i=1}^k \sum_{j=1}^N (u_{ji})^m (v_i - v') (v_i - v')^T \quad (3.22)$$

$$\sum_i i = \frac{\sum_{j=1}^N (u_{ji})^m (X_j - V_i)(X_j - V_i)^T}{\sum_{j=1}^N (u_{ji})^m} \quad (3.23)$$

Eşitlik (3.21)'de SC indeksinin matematiksel ifadesi verilmiştir. Eşitlik (3.22)'de verilen S_B , kümeler arası bulanık dağılım matrisidir. Eşitlik (3.23)'te verilen $\sum_i$, bulanık kovaryans matrisini veren bir fonksiyondur.

3.2.8. Doğruluk Metriği

Grup içi ve gruplar arasındaki ölçümün doğruluğunu hesaplamak için oluşturulmuş bir yöntemdir. Bu doğruluk metriğinde, gruplar arası mesafe ölçümü için dinamik olarak bir grup lideri oluşturulur. Doğruluk metriği [0-1] arasında değer alır.

1'e yakın değerler için grup içi ve gruplar arası dağılımın yani kümelemenin daha iyi olduğu sonucuna varılır (X. Zhang et al., 2002).

$$N' = \sum_{i,j \in [1.k]} |G_i||G_j| \quad (3.24)$$

$$1 - \frac{1}{N'} \sum_{\substack{i,j \in [1.k] \\ i < j}} \sum_{\substack{x \in G_i \\ y \in G_j}} \frac{|(x, y) - (L_i, L_j)|}{x_1 y} \quad (3.25)$$

Eşitlik (3.25) 'te doğruluk metriğinin hesaplanması verilmiştir. Eşitlik (3.25)'e göre, G_i ve G_j grupları, k ise grup sayısını temsil eder. Eşitlikte L_i ve L_j değerleri G_i ve G_j gruplarının liderlerini ifade eder. (x, y) iki veri noktası arasındaki mesafeyi temsil eder. Eşitlik (3.24)'te N' değeri, G_i ve G_j gruplarının elemanlarının çarpımının toplamını verir.

4. YÖNTEM

Tezin bu bölümünün ilk kısmında küme sayısını tahmin etmek için geliştirilen bulanık ve ilgi tabanlı bir kümeleme yöntemi olan Bulanık C Ortalamalar Jensen Shannon (BCOJS) algoritmasının sözde kodu, akış şeması ve açıklaması yer almaktadır. Geliştirilen yöntemi daha iyi açıklamak için BCOJS algoritmasının uygulama adımları bir örnek üzerinden gösterilmiştir. Bu bölümün sonraki kısmında BCOJS algoritmasının JS eşik değeri parametresinin belirlenmesi problemini gidermek için geliştirilen adaptifleştirilmiş BCOJS yönteminden bahsedilmiştir.

4.1. Bulanık C Ortalamalar Jensen Shannon Algoritması

Bulanık C Ortalamalar Jensen Shannon (BCOJS) algoritması, BCO algoritmasının uygun küme sayısının seçilebilmesi sorununu iyileştirmek için geliştirilmiş bir algoritmadır. BCOJS algoritması, klasik BCO yöntemi ve Jensen Shannon mesafe matrisi olmak üzere iki kısımdan oluşmaktadır. Şekil 4.1.'de BCOJS algoritmasının akış şeması, Tablo 4.1.'de ise algoritmanın sözde kodu verilmiştir. BCOJS algoritmasının sözde kodunun açıklaması şu şekildedir:

Adım 1 - 11: Geliştirilen BCOJS algoritmasının ilk adımında algoritmanın gerekli girdi parametreleri (m : bulanıklık parametresi, k : maksimum küme sayısı, l : maksimum iterasyon sayısı, ε : Jensen Shannon mesafesi için gerekli eşik değeri parametresi) ve algoritma sonlandığında üretilcek çıktılar (k_T : tahmin edilen küme sayısı, V' : küme merkezleri ve U' : güncellenen üyelik matrisi yani bölümler matrisi) yer alır. İlgili parametrelerin belirlenmesinin ardından rastgele üyelik matrisi oluşturularak 4. adım çalıştırılır. Ardından BCO algoritmasının 5 ile 11 arasındaki diğer adımlarının çalıştırılması ile küme merkezleri ve üyelik matrisi güncellenir. Güncel üyelik matrisi hesaplandığında BCOJS algoritmasının ilk kısmı olan BCO yöntemi çalıştırılmış olur.

Adım 12: Bilindiği üzere BCO yönteminde oluşan üyelik matrisindeki sütun sayısı, algoritmanın başlangıcında belirtilen küme sayısı (k) kadardır. BCOJS algoritması uygun küme sayısını tahmin etmek için, BCO algoritmasının çalışması ile elde edilen güncel üyelik matrisini kullanır. Üyelik matrisinin her bir sütunu arasında, eşitlik (4.1) 'de gösterilen Jensen Shannon (JS) mesafesi hesaplanarak ($k \times k$) boyutunda JS mesafe matrisi oluşturulur.

$$JS = \sqrt{\frac{1}{2} \sum P(\log P - \log M) + \frac{1}{2} \sum Q(\log Q - \log M)} \quad (4.1)$$

Adım 13 - 14: Oluşan matris içerisindeki en küçük JS değeri bulunur. Matris içerisindeki en küçük JS değeri, belirlenen JS eşik değeri ile kıyaslanır. JS eşik değerinin, minimum değeri sağlayan küme çiftinden büyük ya da küçük olmasına bağlı olarak algoritma iki farklı duruma göre ilerler. Küme çiftleri arasındaki minimum JS değeri eşik değerinden küçük değilse, mevcut kümeler ile algoritma sonlandırılır. Hesaplanan minimum JS değeri eşik değerinden küçük olduğu sürece döngü bir sonraki adıma geçerek devam eder.

Adım 15 - 16: Matristeki minimum JS değerini sağlayan iki kümenin eleman sayıları kıyaslanır. Küme çiftleri arasından eleman sayısı büyük olan küme lider seçilir. İki küme arasındaki ağırlıklı ortalamaya göre (Eşitlik (4.2)) eleman sayısı büyük olan kümenin küme merkezi güncellenir.

Eşitlik (4.2)'de, C_i minimum küme merkezlerini, W_i ise küme merkezlerine ait eleman sayılarını temsil etmektedir.

$$V_C^I = \frac{\sum_{i=1}^N W_i C_i}{W_i} \quad (4.2)$$

Adım 17: Küme çiftlerinden eleman sayısı küçük olan kümenin elemanları büyük kümenin elemanlarına dahil edilir.

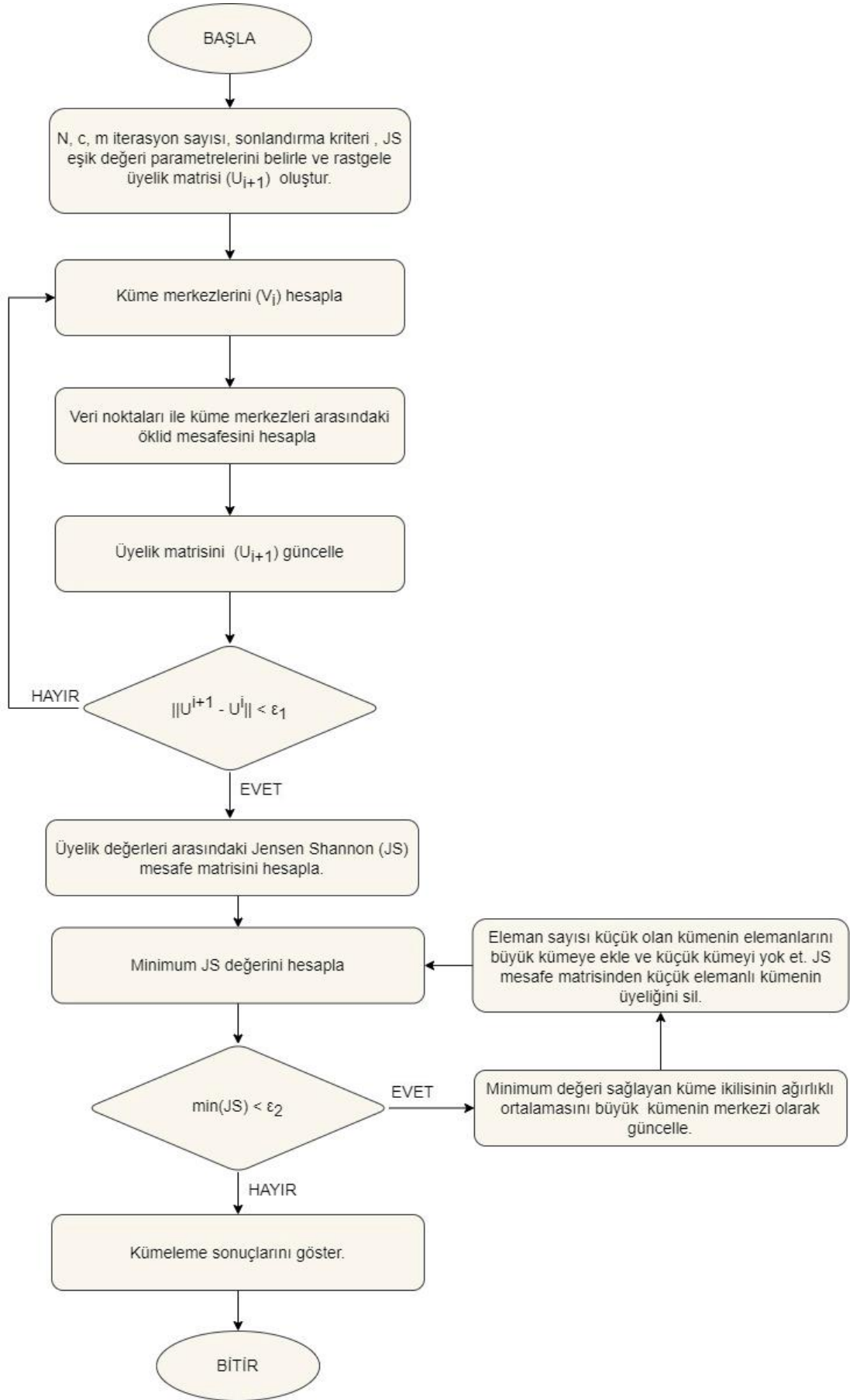
Adım 18 - 20: Eleman sayısı küçük olan küme, elemanlarını büyük kümeye verdiği için küçük küme yok edilir. Küme sayısı döngünün her bir iterasyonunda bir azaltılarak güncellenir. En küçük kümenin yok edilmesiyle JS mesafe matrisinden bu kümenin olduğu üyelik kaldırılır. Matristen küçük kümenin üyeliğinin silinmesi ile tekrardan minimum JS değeri hesaplanır.

Adım 21 – 23: Adım 14'te başlayan döngü sonlandığında, tahmin edilen küme sayısı bulunmuş olur. Küme sayısının bulunması ile son olarak üyelik matrisi güncellenir ve algoritma sonlandırılır.

Tablo 4.1. BCOJS algoritmasının sözde kodu

Algoritma: Bulanık C Ortalamalar Jensen Shannon

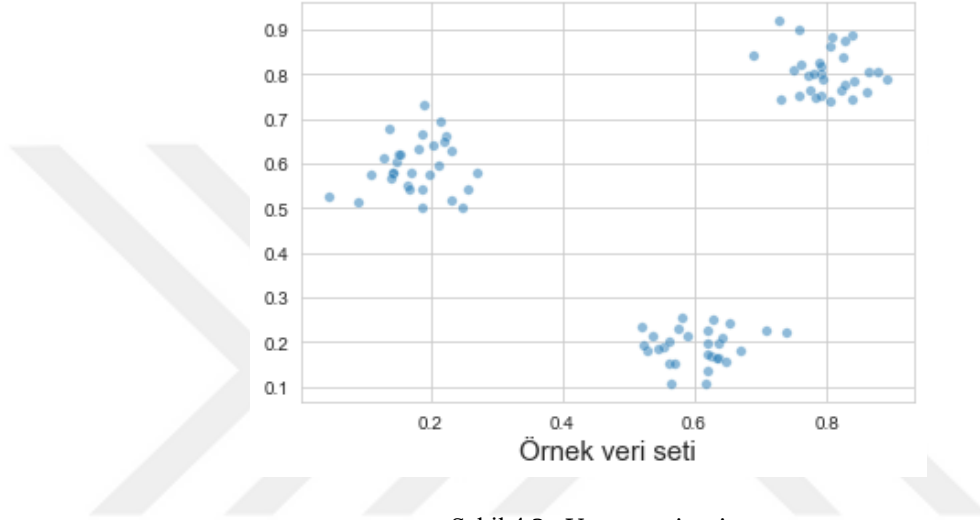
1. **Başla**
2. **Girdi:**
Maksimum Küme Sayısı : k ,
Veri Sayısı: N ,
Bulanıklık Parametresi: m ,
Eşik Değeri: ε_1 ,
Jensen Shannon Eşik Değeri: ε_2 ,
Maksimum İterasyon Sayısı: l
3. **Çıktı:**
Tahmin Edilen Küme Sayısı : k_T ,
Küme Merkezleri : V' ,
Güncel Üyelik matrisi : U'
4. Başlangıç Üyelik matrisini ($U^{(0)}$) hesapla
5. $i = 1$ 'den N 'ye kadar **tekrarla**
6. $j = 1$ 'den k 'ye kadar **tekrarla**
7. $l = 1, 2, 3, \dots$ için **tekrarla**
8. Küme merkezlerini hesapla.
9. Öklid mesafesini hesapla.
10. Üyelik matrisini (U^{i+1}) güncelle.
11. $\|U^{i+1} - U^i\| < \varepsilon_1$ oluncaya kadar
12. JS mesafe matrisini hesapla.
13. JS mesafe matrisindeki minimum değeri (min JS (x,y)) bul.
14. $\min JS(x,y) < \varepsilon_2$ **olduğu sürece**
15. JS mesafe matrisindeki minimum değeri sağlayan küme ikilisinin eleman sayılarını kıyasla.
16. Büyük olan kümenin merkezini minimum değerli küme ikilisinin ağırlıklı ortalaması ile güncelle.
17. Küçük eleman sayılı kümenin elemanlarını büyük kümeye ata.
18. $k = k - 1$
19. JS mesafe matrisinden küçük elemanlı kümenin üyeliğini sil.
20. JS mesafe matrisindeki minimum değeri hesapla.
21. $k_T = k$
22. Üyelik matrisini (U^{i+1}) güncelle.
23. **Bitir**



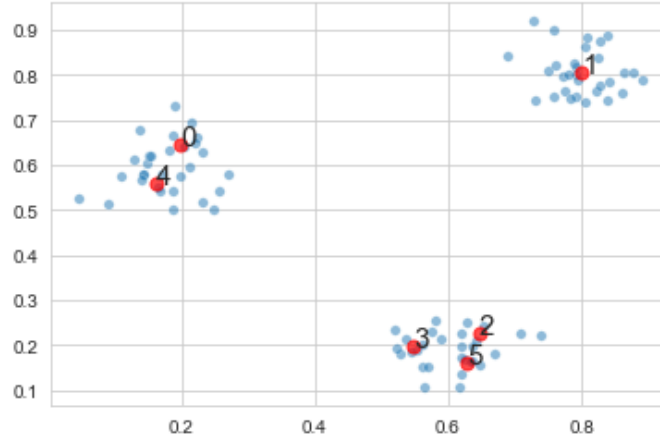
Şekil 4.1. Bulanık C Ortalamalar Jensen Shannon Algoritması Akış Diyagramı

4.1.1. BCOJS Algoritmasının Bir Örnek Üzerinden Açıklanması

Bu bölümde tanıtılan BCOJS algoritmasını daha iyi açıklamak için bu yöntem bir örnek üzerinden açıklanmıştır. Bu örneğin açıklanması için Şekil 4.2.'de gösterilen, 3 kümeden oluşan 90 adet veri noktası bulunan 2 boyutlu yapay veri seti oluşturulmuştur. Bu veri seti maksimum küme sayısı 6 verilerek BCOJS algoritmasında çalıştırılmıştır. Şekil 4.3.'te BCOJS yönteminin ilk bölümü olan BCO ile çalışması sonucu elde edilen küme merkezlerinin veri üzerindeki dağılımı gösterilmiştir.



Şekil 4.2. Yapay veri seti



Şekil 4.3. BCO ile elde edilen küme merkezleri

Kümelerin birbirinden uzak olması aralarındaki ilginin düşük olmasına yol açar. Açıklanan bu örnekte, JS eşik değeri 0.55 olarak verilmiştir. BCOJS yönteminin BCO kısmının çalıştırılması ile elimizde, her bir elemanın 6 kümeden bir kümeye ait olma olasılığı bulunmaktadır. 90 elemanlı veriden oluşan veri örneğinin üyelik değerleri Tablo 4.2.'de verilmiştir. Tablo 4.2.'de her bir veri elemanın 6 kümeden herhangi

birine ait olma olasılığının örnek bir kesiti verilerek üyelik matrisinin görünümü gösterilmiştir. Bu üyelik matrisindeki herhangi bir veri elemanının tüm kümelere ait üyelik değerlerinin toplamı 1'dir.

Tablo 4.2. Üyelik değerlerinin gösterilmesi

Veri Sayısı	U0	U1	U2	U3	U4	U5
1	0.1829	0.1160	0.3600	0.0659	0.0724	0.2024
2	0.0396	0.1216	0.2029	0.1477	0.1112	0.3767
3	0.1322	0.0634	0.0329	0.1903	0.5357	0.0453
·	·	·	·	·	·	·
·	·	·	·	·	·	·
·	·	·	·	·	·	·
90	0.2768	0.0180	0.1372	0.1968	0.3481	0.0228

Üyelik matrisindeki her bir üyelik değeri arasında Tablo 4.3'te verildiği gibi JS mesafe matrisi oluşturulur. Tablo 4.3 incelendiğinde, örneğin U5 kolonu ile U0, U1, U2, U3, U4 kolonları arasındaki JS değerleri hesaplanarak JS mesafe matrisi oluşturulur. Jensen Shannon mesafesi simetrik bir metrik olduğundan her küme çiftinden biri arasındaki mesafenin hesaplanması yeterlidir. Örneğin Tablo 4.3'te yer alan 0 ve 1 numaralı küme çifti ile 1 ve 0 numaralı küme çiftinin JS mesafeleri eşit olduğu için bu örnek matriste yalnızca 1 ve 0 numaralı küme grupları arasındaki mesafenin alınması yeterlidir.

Tablo 4.3'te en küçük JS değerinin (4,0) koordinatlı küme ikilisinin olduğu görülmektedir. 4 numaralı kümenin 17 elemanı var iken 0 numaralı kümenin eleman sayısı 13'tür. Küme ağırlığı 4 numaralı kümede olduğu için 4 ve 0 numaralı kümenin ağırlıklı ortalaması sonucu yeni küme merkezi eşitlik (4.2)'ye göre eşitlik (4.3)'te verildiği gibi hesaplanır.

$$\frac{17(0.1617, 0.5591) + 13(0.1978, 0.6438)}{17+13} = 0.1773, 0.5958 \quad (4.3)$$

Tablo 4.3. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisinin gösterilmesi

JS	0	1	2	3	4	5
0	-	-	-	-	-	-
1	0.79	-	-	-	-	-
2	0.78	0.77	-	-	-	-
3	0.78	0.78	0.58	-	-	-
4	0.51	0.79	0.78	0.78	-	-
5	0.79	0.79	0.51	0.52	0.79	-

Hesaplanan bu yeni merkez için Tablo 4.4'te BCO ile bulunan küme merkezleri kullanılmıştır. 4. ve 0. küme merkezlerinin ağırlıklı ortalaması alınarak 4. küme merkezi (0.1773, 0.5958) olarak güncellenmiştir. Artık 0 numaralı küme elemanlarını 4 numaralı kümeye bırakacağından 0 numaralı kümeye ihtiyaç kalmamıştır. 0 numaralı küme merkezi silinir ve küme sayısı bir azaltılır.

Tablo 4.4. BCO ile oluşan küme merkezlerinin gösterilmesi

Küme Grup Numarası	Merkez-1	Merkez-2
0	0.1978	0.6438
1	0.7998	0.8053
2	0.6461	0.2249
3	0.5472	0.1961
4	0.1617	0.5591
5	0.6273	0.1613

0 numaralı kümenin silinmesi ve 4. kümenin merkezinin güncellenmesi ile küme merkezlerinin yeni durumu Tablo 4.5'te gösterilmiştir.

Tablo 4.5. BCO ile oluşan küme merkezlerinin ilk güncellenmesi

Küme Grup Numarası	Merkez-1	Merkez-2
0	-	-
1	0.7998	0.8053
2	0.6461	0.2249
3	0.5472	0.1961
4	0.1773	0.5958
5	0.6273	0.1613

0 numaralı küme silindiğinden o kümenin üyelik değerleri kullanılmaz. JS mesafe matrisinden 0 numaralı küme silinir. Tablo 4.6'da algoritmanın son durumdaki üyelik matrisleri arasındaki minimum JS değeri 0.51'dir. Bu değer halen belirlenen 0.55 eşik değerinden küçük olduğu için küme merkezleri güncellenmeye devam eder. Tablo 4.6 incelendiğinde minimum JS değerli küme çiftinin 5. ve 2. küme olduğu

görülür. Eşitlik (4.2) 'ye göre ağırlıklı ortalama hesaplanarak küme merkezleri tekrar güncellenir. 5. kümenin eleman sayısı 2. kümenin eleman sayısından fazladır. Bu yüzden 2. küme kaldırılır. Küme merkezleri güncellenerek işleme devam edilir.

Tablo 4.6. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisi

JS	0	1	2	3	4	5
0	-	-	-	-	-	-
1	-	-	-	-	-	-
2	-	0.77	-	-	-	-
3	-	0.78	0.58	-	-	-
4	-	0.79	0.78	0.78	-	-
5	-	0.79	0.51	0.52	0.79	-

Tablo 4.7. incelendiğinde 5. ve 3. küme arasındaki JS değerinin 0.52 ile minimum olduğu ve eşik değerinden küçük olduğu görülür. 5. kümenin eleman sayısı 3. kümenin eleman sayısından fazladır. Bu yüzden 3. küme kaldırılır. Küme merkezleri güncellenerek işleme devam edilir.

Tablo 4.7. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisi

JS	0	1	2	3	4	5
0	-	-	-	-	-	-
1	-	-	-	-	-	-
2	-	0.77	-	-	-	-
3	-	0.78	-	-	-	-
4	-	0.79	-	0.78	-	-
5	-	0.79	-	0.52	0.79	-

Son durumda Tablo 4.8. incelendiğinde en küçük JS değerinin 0.77 olduğu ve eşik değerinden büyük olduğu görülür. Minimum JS değeri eşik değerinde küçük olmadığı için algoritma tahmin ettiği küme sayısı çıktısını vererek sonlanır.

Tablo 4.8. Üyelik değerleri arasındaki Jensen Shannon mesafe matrisi

JS	0	1	2	3	4	5
0	-	-	-	-	-	-
1	-	-	-	-	-	-
2	-	0.77	-	-	-	-
3	-	0.78	-	-	-	-
4	-	0.79	-	-	-	-
5	-	0.79	-	-	0.79	-

Tablo 4.9. incelendiğinde son durumda 6 kümeden 3 kümeye inen yeni küme merkezlerinin değişimi verilmiştir.

Tablo 4.9. BCOJS kümesi ile oluşan küme merkezleri

Küme Grup Numarası	Merkez-1	Merkez-2
1	0.7998	0.8053
4	0.1773	0.5958
5	0.5987	0.1911

4.2. Adaptifleştirilmiş Bulanık C Ortalamalar Jensen Shannon Algoritması

Küme sayısını tahmin etmek için geliştirilen BCOJS yönteminde JS eşik değeri belirlenmelidir. Kullanılan JS eşik değeri parametresi, iki küme arasındaki ilginin ne olacağını yani kümelerin birbiri ile ne kadar alaka içerdiğini belirtir. Bu eşik değeri rastgele veya veri setinin farklı eşik değerlerinde çalıştırılıp sonuçların küme geçerlilik indekslerinde karşılaştırılarak uygun değerin seçilmesi ile belirlenir. JS eşik değerinin belirlenmesinin zahmetli oluşu algoritma için dezavantaj oluşturmaktadır. Bu sebeple önerilen BCOJS algoritması geliştirilerek, algoritma tarafından belirlenen dinamik bir eşik değeri ile eşik değeri parametresinin belirlenmesi sorunu giderilmiştir.

Adaptif BCOJS yönteminde eşik değerinin algoritma tarafından belirlenmesi için çeyrekler açıklığı yöntemi kullanılmıştır. Çeyrekler açıklığı, sıralanmış bir veri setinin ilk çeyreği olan %25 ve son çeyreği olan %75 arasındaki farktır. İlk çeyrek ve son çeyrek $Q1$ ve $Q3$ olarak gösterilirse çeyrekler açıklığı eşitlik 4.4'te iki çeyrek arasındaki farka göre hesaplanır. Çeyrekler açıklığı ile eşikleme stratejisi için eşitlik 4.5 kullanılır. Eşitlik 4.5'te yer alan T değeri veri seti için kullanılabilecek minimum eşik değerini, c ise kullanıcı tarafından karar verilen ve genellikle 1.5 olarak kullanılan bir sabiti ifade eder (Yang et al., 2019).

$$IQR = Q3 - Q1 \quad (4.4)$$

$$T = Q1 - c * IQR \quad (4.5)$$

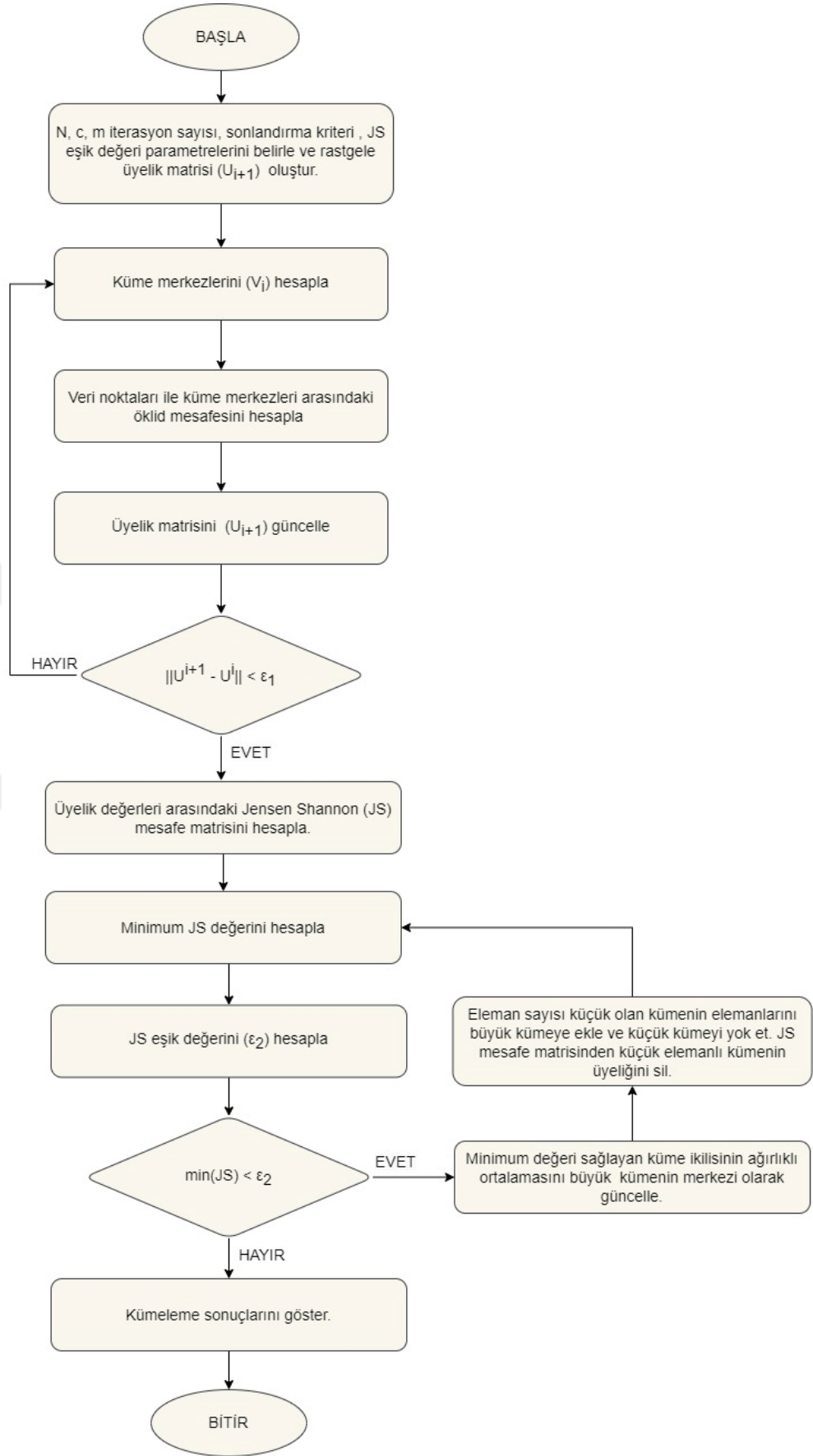
Tablo 4.10.'da adaptifleştirilmiş BCOJS algoritmasının sözde kodu verilmiştir. Adaptif BCOJS algoritması sadece eşik değerinin belirlenmesi yönüyle BCOJS algoritmasından farklıdır. Adım 14'e kadar adaptif BCOJS yönteminde BCOJS algoritması ile aynı adımlar izlenir. Eşitlik 4.5'te çeyrekler açıklığı ile eşikleme stratejisi kullanılarak adım 14'te yer alan JS eşik değeri hesaplanır. Eşik değeri

hesaplanan adaptif BCOJS algoritması adım 15'e geçerek BCOJS ile aynı adımları izler. Adım 20'de JS mesafe matrisindeki en küçük kümenin üyeliğinin silinmesi ile matris değişir. Bu sebeple adım 21'de tekrar JS eşik değeri hesaplanarak BCOJS algoritmasının adımlarına devam edilir. Şekil 4.4'te JS eşik değeri hesaplama adımının eklenmesi ile oluşturulan adaptif BCOJS algoritmasının akış diyagramı verilmiştir.

Tablo 4.10. BCOJS algoritmasının sözde kodu

Algoritma: Adaptifleştirilmiş Bulanık C Ortalamalar Jensen Shannon

1. **Başla**
2. **Girdi:**
Maksimum Küme Sayısı : k ,
Veri Sayısı: N ,
Bulanıklık Parametresi: m ,
Eşik Değeri: ε_1 ,
Jensen Shannon Eşik Değeri: ε_2 ,
Maksimum İterasyon Sayısı: l
3. **Çıktı:**
Tahmin Edilen Küme Sayısı : k_T ,
Küme Merkezleri : V ,
Güncel Üyelik matrisi : U
4. Başlangıç Üyelik matrisini ($U^{(0)}$) hesapla
5. $i = 1$ 'den N 'ye kadar **tekrarla**
6. $j = 1$ 'den k 'ye kadar **tekrarla**
7. $l = 1, 2, 3, \dots$ için **tekrarla**
8. Küme merkezlerini hesapla.
9. Öklid mesafesini hesapla.
10. Üyelik matrisini (U^{i+1}) güncelle.
11. $\|U^{i+1} - U^i\| < \varepsilon_1$ oluncaya kadar
12. JS mesafe matrisini hesapla.
13. JS mesafe matrisindeki minimum değeri (min JS (x,y)) bul.
14. Jensen Shannon eşik değerini (ε_2) hesapla.
15. $\min JS(x,y) < \varepsilon_2$ **olduğu sürece tekrarla**
16. JS mesafe matrisindeki minimum değeri sağlayan küme ikilisinin eleman sayılarını kıyasla.
17. Büyük olan kümenin merkezini minimum değerli küme ikilisinin ağırlıklı ortalaması ile güncelle.
18. Küçük eleman sayılı kümenin elemanlarını büyük kümeye ata.
19. $k = k - 1$
20. JS mesafe matrisinden küçük elemanlı kümenin üyeliğini sil.
21. Jensen Shannon eşik değerini (ε_2) hesapla.
22. JS mesafe matrisindeki minimum değeri hesapla.
23. $k_T = k$
24. Üyelik matrisini (U^{i+1}) güncelle.
25. **Bitir**



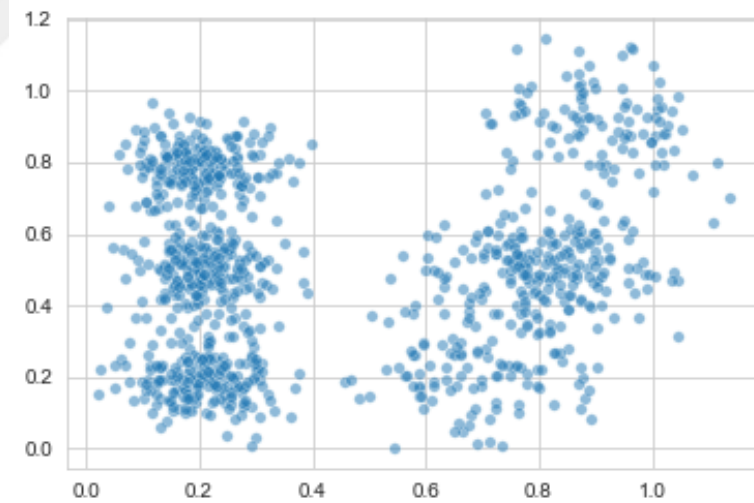
Şekil 4.4. Adaptifleştirilmiş Bulanık C Ortalamalar Jensen Shannon Algoritması Akış Diyagramı

5. BULGULAR

Bu bölümde önerilen BCOJS yönteminin oluşturulan yapay veri seti ile test edilmesi ve öneri sistemlerinde kullanılan film veri seti üzerine uygulanması aşaması verilmiştir. Film veri seti ön işleme adımlarından geçirilerek düzenlenmiştir. BCOJS ile elde edilen bulgular diğer yöntemlerle test edilerek karşılaştırılmıştır. BCOJS yönteminin eşik değerinin algoritma tarafından belirlendiği ikinci önerilen yöntem olan adaptifleştirilmiş BCOJS yöntemi yine film veri seti üzerine uygulanmış ve elde edilen bulgular verilmiştir.

5.1. Önerilen BCOJS Yöntemin Oluşturulan Veri Seti ile Test Edilmesi

Önerilen BCOJS algoritmasının sonuçlarını değerlendirmek için 1000 elemanlı, 6 farklı kümeden oluşan, 2 farklı öznitelik barındıran yapay bir veri seti oluşturulmuştur. Yöntemin sonuçlarını değerlendirmek ve görselleştirebilmek için 2 boyutlu yapay veri seti tercih edilmiştir. Şekil 5.1.'de oluşturulan yapay veri setinin veri dağılımının genel görüntüsü verilmektedir.

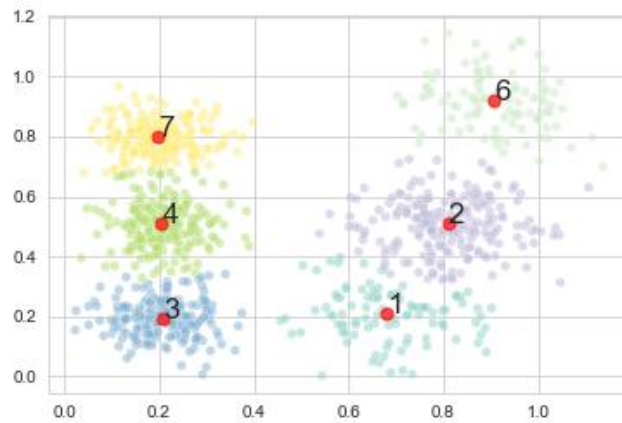


Şekil 5.1. Yapay veri setinin genel görünümü

BCOJS yöntemi maksimum küme sayısı 12, JS eşik değeri 0.47 bulanıklık parametresi ise 2 verildiğinde bu yöntemin tahmin ettiği küme sayısı 6 olarak bulunmuştur. BCOJS algoritmasının 6 kümeye ulaşma aşamasında ilk olarak BCO algoritması ile 12 kümenin üyeliğini içeren bölümlleme matrisi elde edilmiştir. Bölümlleme matrisindeki her bir sütun her bir kümeye ait üyelik değerini göstermektedir. Oluşan 12 kümenin üyelikleri arasında JS metriği ile mesafe matrisi oluşturulur. 11. ve 3. kümeye ait üyelik değerleri arasındaki 0.43 JS değeri mesafe

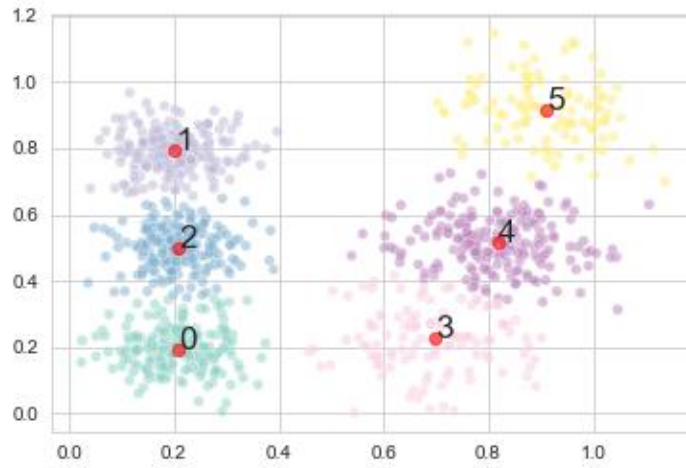
matrisindeki minimum değer olarak bulunmuştur. 3. küme daha fazla veri içerdiğinden 11. kümenin elemanlarını kendine alarak küme merkezini 3. ve 11. kümenin ağırlıklı ortalamasına göre günceller. Nihayetinde 11. küme merkezi silinir ve elemanları 3. kümeye dahil olmuş olur. Küme sayısı ve elemanlarının güncellenmesi ile tekrar mesafe matrisi oluşturulur. Bu döngü eşik değerinden küçük olana kadar devam eder. Döngü bittiğinde 0. küme elemanlarını 4. kümeye, 9. küme elemanlarını 1. kümeye, 5. küme elemanlarını 2. kümeye 10. küme elemanlarını 6. kümeye, 8. küme elemanlarını 2. kümeye bırakır.

BCOJS algoritmasının sonucunda 0., 5., 8., 9., 10., 11. kümeler elemanlarını kendisine en yakın eşik değeri olan kümeye teslim eder ve bu küme merkezleri silinir. 1., 2., 4. ve 6. kümelerin merkezleri ve elemanları güncellenmiş olur. Şekil 5.2.'de BCOJS algoritmasının verilen 12 küme sayısını tahmin ettiği 6 küme sayısına indirmesi sonucu her bir kümeye ait küme elemanları ve küme merkezleri renklendirilerek gösterilmiştir.



Şekil 5.2. Yapay veri setinin BCOJS ile kümeleneşmesi sonucu oluşun küme merkezleri

BCOJS yöntemi BCO yönteminden türetilmiş ilgi tabanlı bir bulanık kümeleme algoritmasıdır. BCOJS yönteminde tahmin edilen küme sayısı ile BCO yöntemi çalıştırıldığında iki kümeleme sonucunun benzerlik göstermesi beklenir. Şekil 5.3.'te BCO algoritmasının 6 küme sayısı ile yaptığı kümelemenin sonucunda oluşun kümeler ve bu kümelerin merkezleri gösterilmiştir. Şekil 5.2. ve 5.3.'te kırmızı noktalar ile gösterilen küme merkezlerinin değeri Tablo 5.1.'de verilmiştir.



Şekil 5.3. Yapay veri setinin BCO ile kümelenmesi sonucu oluşan küme merkezleri

Tablo 5.1.'de iki yöntemin çalışması sonucu oluşan 6 kümenin merkezinin BCO ve BCOJS algoritmalarında benzer olduğu görülmektedir. Veriler yalın ve iki boyutlu olduğu için yöntemlerin başarımları küçük duyarlılıklarda farklılık göstermektedir.

Tablo 5.1. BCOJS ve BCO algoritmalarının küme merkezlerinin karşılaştırılması

Merkezler	BCOJS algoritmasının küme merkezleri		BCO algoritmasının küme merkezleri	
	Boyut - 1	Boyut - 2	Boyut - 1	Boyut - 2
Merkez-0	0.2075	0.1928	0.2072	0.1936
Merkez-1	0.1964	0.7994	0.1996	0.7913
Merkez-2	0.2056	0.5100	0.2090	0.5005
Merkez-3	0.6796	0.2080	0.6987	0.2279
Merkez-4	0.8109	0.5097	0.8183	0.5149
Merkez-5	0.9077	0.9215	0.9089	0.9129

5.2. Önerilen Yöntemin Diğer Yöntemlerle Karşılaştırılması

BCOJS yönteminin başarısını değerlendirmek için yapay veri seti kullanılarak BCOJS yöntemi ile JS, BCO, OCO, OBCO yöntemlerinin doğruluk metriği sonucu karşılaştırılmıştır. Tablo 5.2.'de BCOJS yönteminin diğer yöntemler ile karşılaştırılmasının sonuçları verilmiştir. Bu karşılaştırmada 30 çalışmanın ortalaması sonucu elde edilen değerler verilmiştir. Tablo 5.2.'de verilen sonuçlara göre BCO %81.7059, BCOJS %81.6864 değeri ile birbirine çok yakın sonuç elde etmiştir. BCO ve BCOJS yöntemi diğer yöntemlerden çok daha iyi sonuç vermiştir. Ancak BCOJS yöntemi küme sayısını bilmeden kullandığı ilgi metriğine göre bu başarıyı elde etmiştir.

Tablo 5.2. Doğruluk metriği ile JS, BCO, BCOJS, OCO, OBCO yöntemlerinin karşılaştırılması

	Algoritma	Doğruluk
1	JS	% 52.5603
2	BCO	% 81.7059
3	BCOJS	% 81.6864
4	OCO	% 42.9304
5	OBCO	% 77.0532

Tablo 5.3.'te BCOJS yöntemi BCO, OCO, OBCO yöntemleri ile 7 farklı küme geçerlilik indeksi, 30 çalışma sonucunda elde edilen ortalamanın değerine yer verilerek kıyaslanmıştır. Bu karşılaştırma sonucunda artan değerlerde (BK, UBK, SC) ve azalan değerlerde (BE, UBE, XB, FS) daha iyi sonuç veren küme geçerlilik indeksleri ile karşılaştırma yapıldığında BCOJS yönteminin BCO, OCO ve OBCO yöntemine kıyasla başarılı olduğu görülmektedir. BCOJS tüm kümeleme indeksleri için OCO ve OBCO yönteminden başarılı bulunmuştur.

Tablo 5.3. BCO, BCOJS, OCO, OBCO yöntemlerinin küme geçerlilik indeksleri ile kıyaslanması

Metrikler	Başarı Değişimi	BCO	BCOJS	OCO	OBCO
BE	(-)	0.2745	0.2740	0.741422	0.6138
BK	(+)	0.7112	0.7117	0.23674	0.3227
UBE	(-)	0.2762	0.2760	0.744590	0.6101
UBK	(+)	0.6534	0.6541	0.19240	0.2025
XB	(-)	0.09561	0.09478	1903827.4098	0.5757
SC	(+)	1394.5061	1379.3676	12.9017	102.7029
FS	(-)	-103.46	-103.28	13.4861	-19.0400

5.3. Film Veri Setinin Düzenlenmesi

Bu çalışmada önerilen tavsiye yöntemini test etmek için, Minnesota Üniversitesi'nden bir araştırma grubunun MovieLens web sitesi (grouplens.org) üzerinden yayınladığı film veri seti kullanılmıştır. Kullanılan film veri seti toplamda 9742 adet filmi ve bu filmlerin türlerini içermektedir. Şekil 5.4.'te film veri setinde bulunan filmlerin isimleri ve ilgili filmin hangi türü içerdiği bilgisi ile veri setinin genel bir şeması gösterilmiştir.

Film Id		Film Adı	Film Türleri
0	1	Toy Story (1995)	[Adventure, Animation, Children, Comedy, Fantasy]
1	2	Jumanji (1995)	[Adventure, Children, Fantasy]
2	3	Grumpier Old Men (1995)	[Comedy, Romance]
3	4	Waiting to Exhale (1995)	[Comedy, Drama, Romance]
4	5	Father of the Bride Part II (1995)	[Comedy]

Şekil 5.4. Kullanılan film veri setinin gerçekteki görünümü

Tablo 5.4.’te gösterildiği üzere bu veri setinin tür sütununda 20 farklı film türü bulunmaktadır.

Tablo 5.4. Veri setinde bulunan film türleri

FİLM TÜRLERİ
(no genres listed)
Action
Adventure
Animation
Children
Comedy
Crime
Documentary
Drama
Fantasy
Film-Noir
Horror
IMAX
Musical
Mystery
Romance
Sci-Fi
Thriller
War
Western

Bu tez çalışmasında kullanılan yöntemin performansını ve doğruluğunu arttırmak için veri setinde 5 farklı film türü kullanılmıştır. Bu 5 farklı film türünün seçimine karar verilirken, yaygın olarak kullanılan film türleri tercih edilerek modifiye edilen veri setinde azami sayıda filme yer vermeye çalışılmıştır. Az sayıda belirleyici özellikle filmlerin çoğu veri setine dahil edilmiştir. Şekil 5.4. ‘te *Toy Story* filmi macera, animasyon, çocuk, komedi ve fantastik film türlerini içerirken, Şekil 5.5’te aynı film macera, animasyon ve komedi türleri ile sınırlandırılmıştır. Düzenlenen yeni film veri setinde toplamda 7650 adet film yer almaktadır.

	Film Id	Film Adı	Macera	Animasyon	Komedi	Drama	Korku
0	1.0	Toy Story (1995)	1	1	1	0	0
1	2.0	Jumanji (1995)	1	0	0	0	0
2	3.0	Grumpier Old Men (1995)	0	0	1	0	0
3	4.0	Waiting to Exhale (1995)	0	0	1	1	0
4	5.0	Father of the Bride Part II (1995)	0	0	1	0	0

Şekil 5.5. Yeni film veri setinin öznitelikleri şeması

Veri setinde bulunan her bir filmin öznitelik değerini toplamları 1 olacak şekilde, [0-1] aralığındaki bir olasılık dağılımında tutmak için dirichlet dağılımı kullanılmıştır. Dirichlet dağılımı beta dağılımının genelleştirilmiş halidir. Bu dağılım verileri özniteliklerin ağırlıklarına göre esnek bir şekilde eşleştirerek verimli bir model oluşturmak için kullanılır (Nedaie and Najafi, 2018).

Şekil 5.5.'te 0 ve 1'lerden oluşan film veri setinin dirichlet dağılımı ile ağırlıklandırılması sonucu, Şekil 5.6.'da veri setinin nihai şekli gösterilmiştir. Bu ağırlıklandırmada örneğin; *Toy Story* filmi incelendiğinde içerdiği animasyon, macera ve komedi türlerinin toplamı 1'e yakın değerler alırken, filmin türüne dahil olmayan drama ve korku türleri ise 0'a yakın bir değer ile ağırlıklandırılmıştır. Bir filmin tüm dağıtılmış ağırlığının toplamı ise 1'e eşittir.

	Film Id	Film Adı	Macera	Animasyon	Komedi	Drama	Korku
0	1.0	Toy Story (1995)	0.334296	0.297872	0.352717	0.009530	0.005585
1	2.0	Jumanji (1995)	0.922934	0.014360	0.030199	0.030076	0.002431
2	3.0	Grumpier Old Men (1995)	0.009522	0.048395	0.793318	0.001186	0.147579
3	4.0	Waiting to Exhale (1995)	0.045668	0.060441	0.401685	0.428189	0.064016
4	5.0	Father of the Bride Part II (1995)	0.077516	0.003087	0.891859	0.012927	0.014610

Şekil 5.6. Dirichlet dağılımı uygulanan veri setinin nihai görünümü

5.4. BCOJS yönteminin Farklı Eşik Değerleri İle Küme Tahmini

Önerilen yönteminin doğruluğunu göstermek için, bu bölümde BCOJS yöntemi farklı küme sayısı ve eşik değerleri ile test edilmiştir. Bu test için maksimum küme sayısı 25 verildiğinde eşik değerleri 0.35, 0.40, 0.45 iken sonuçların karşılaştırmaları yapılmıştır. Tablo 5.5.'te üç farklı eşik değeri için BCOJS yönteminde tahmin edilen küme sayıları verilmiştir. Eşik değerinin küçültülmesi istenen ilginin çok yüksek olduğu anlamına gelir. Tablo 5.5.'te eşik değeri 0.35 verildiğinde bu eşik değerindeki

ilgiye sahip küme sayısı 20 olarak bulunmuştur. Eşik değeri 0.40 verildiğinde 18, 0.45 verildiğinde ise 15 bulunmuştur.

Tablo 5.5. Farklı eşik değerlerinin karşılaştırılması ile küme sayısı tahmini

JS Eşik Değeri	Verilen Maksimum Küme Sayısı	Tahmin Edilen Küme Sayısı
0.35	25	20
0.40	25	18
0.45	25	15

BCOJS yönteminde eşik değerine karar vermek için, Tablo 5.5.'te verilen farklı eşik değerlerinin bulduğu sonuçlar Tablo 5.6.'da verilen küme geçerlilik indeksleri ile karşılaştırılmıştır. Bu karşılaştırma sonucu bu veri setinde eşik değerinin 0.45 olarak alınmasına karar verilmiştir.

Tablo 5.6. Farklı eşik değerleri ile küme geçerlilik indekslerinin karşılaştırılması

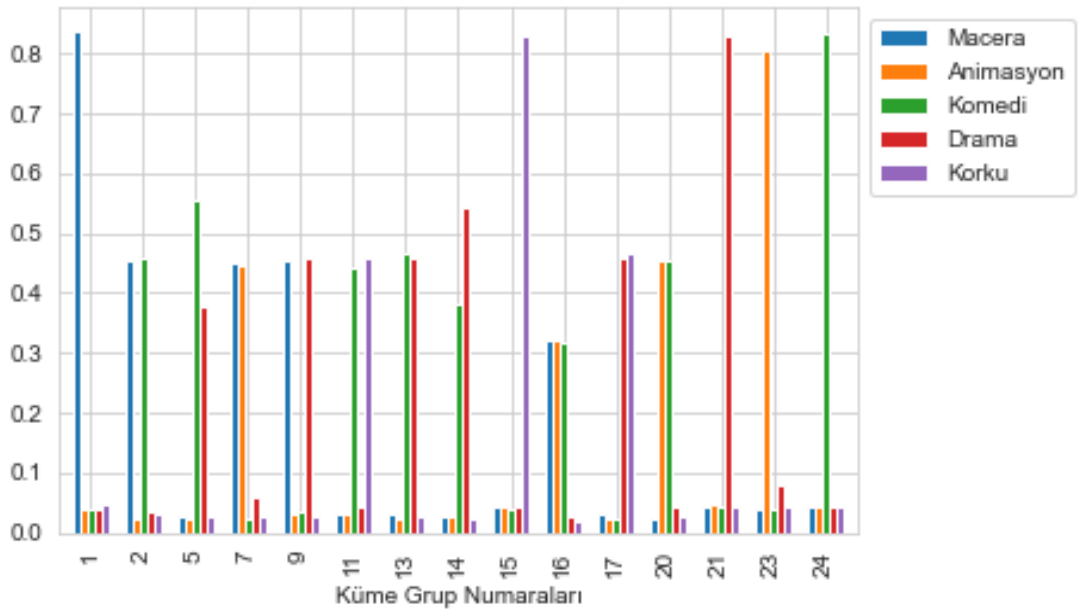
Küme Geçerlilik İndeksleri	Başarı Değişimi	Küme Sayısı: 15	Küme Sayısı: 18	Küme Sayısı: 20
		Eşik Değeri : 0.45	Eşik Değeri : 0.40	Eşik Değeri : 0.35
BE	(-)	0.4146	0.4719	0.5146
BC	(+)	0.6453	0.5575	0.5048
UBE	(-)	0.4154	0.4730	0.5159
UBK	(+)	0.6200	0.5315	0.4788
XB	(-)	0.0332	0.4228	0.5826
SC	(+)	6810.2000	7024.5659	6835.3835
FS	(-)	-1933.2115	-1660.1408	-1509.3835

5.5. BCOJS Algoritmasının Kümeleme Sonuçlarının Gösterilmesi

Öneri yönteminin sonuçlarını göstermek için düzenlenen MovieLens film verisi ile BCOJS algoritması farklı küme sayısı ve eşik değerleri ile çalıştırılmıştır. BCOJS yöntemi, verilen maksimum küme sayısını girdi parametresinde alıp BCO kümeleme algoritmasını çalıştırır. Ardından her bir küme arasındaki JSD değerinin hesaplandığı bir mesafe matrisinde minimum JS değeri eşik değerinden küçük olmadığında algoritma uygun küme sayısını hesaplamış olur.

BCOJS algoritması maksimum küme sayısı 25, JS eşik değeri 0.45 verildiğinde algoritma sonucu hesaplanan uygun küme sayısı ağırlıklı olarak 15 bulunmuştur. 25 farklı kümenin uygun küme sayısına indirilme sürecinde minimum JS değerli küme çiftinden ağırlığı fazla olan yani eleman sayısı yüksek olan küme, ağırlığı küçük olan kümenin elemanlarını kendisine alarak yeni küme merkezini günceller. Böylece

ağırlığı küçük olan küme elemanlarını büyük kümeye bıraktığından silinir. BCO algoritması sonucu oluşan 25 farklı kümenin BCOJS ile 1., 2., 5., 7., 9., 11., 13., 14., 15., 16., 17., 20., 21., 23., 24. küme grup numaraları olarak 15 farklı kümeye indirgenmesi sonucu küme dağılımı Şekil 5.7.'de verilmiştir. Şekil 5.7.'de ortaya çıkan kümeler, filmlerin türlerine göre çeşitli dağılımlar göstermiştir. Bu dağılımlar macera, komedi gibi sadece tek bir türü içeren kümelerden oluştuğu gibi komedi-dram gibi iki farklı türü içeren gruplardan, macera-animasyon-komedi gibi üç farklı film türünü içeren gruplardan oluşabilmektedir. Bu grup oluşumları dışında örneğin 5. kümede oluşan komedi-dram dağılımları komedi ağırlıklı iken, 14. küme grubunda oluşan komedi-dram dağılımının ise dram ağırlıklı olduğu görülmektedir. Şekil 5.7.'de 15 kümenin içinde, aynı tür grubunu içeren komedi-dram dağılımının farklı yoğunlukta farklı film grupları olarak dağıldığı görülmektedir.



Şekil 5.7. BCOJS algoritmasının çalıştırılması sonucu film türlerinin kümelenmesi

Şekil 5.7.'de, veri setinde kullanılan 5 farklı film türünün küme içindeki dağılımı renklendirilerek gösterilmiştir. Örneğin 24 numaralı küme incelendiğinde bu kümenin ortalama % 80 gibi bir ağırlıkla komedi türü, bir başka örnek olarak 13 numaralı kümenin ise komedi ve drama ağırlıklı türler içerdiği görülür.

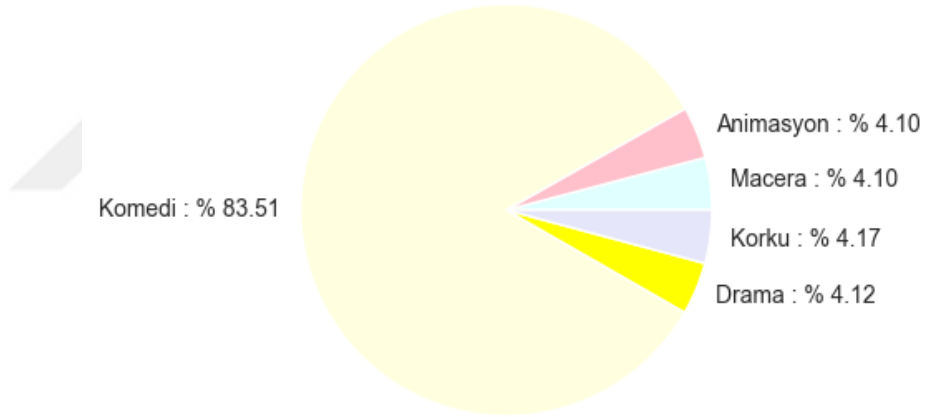
Şekil 5.8.'de, Şekil 5.7.'de gösterilen 24 numaralı kümenin 5 farklı türe göre dağılımının ağırlıklandırılmasından örnek bir kesit gösterilmiştir. Bu şekilde 24 numaralı kümenin bölümlene matrisinin ağırlıklı olduğu alanların komedi türünü

içerdiği görülür. Örneğin Grumpier Old Men (1995) filminin komedi türündeki ağırlığı 0.7933 ile komedi kümesindedir.

Film Id	Film Adı	Macera	Animasyon	Komedi	Drama	Korku	Küme Grup Numarası
2	3.0 Grumpier Old Men (1995)	0.009522	0.048395	0.793318	0.001186	0.147579	24
4	5.0 Father of the Bride Part II (1995)	0.077516	0.003087	0.891859	0.012927	0.014610	24
5	7.0 Sabrina (1995)	0.003648	0.075347	0.859792	0.037238	0.023975	24
15	18.0 Four Rooms (1995)	0.003434	0.005901	0.828743	0.156321	0.005601	24
16	19.0 Ace Ventura: When Nature Calls (1995)	0.053615	0.168157	0.682881	0.069712	0.025635	24

Şekil 5.8. BCOJS algoritmasının çalıştırılması sonucu 24 numaralı küme grubunun film dağılımları

Şekil 5.9.'da ise 24 numaralı kümenin tüm türler içindeki dağılımının pasta grafiği ile gösterilmesi verilmiştir. Şekil 5.9.'da verilen pasta grafiği incelendiğinde 24 numaralı küme grubunun %83.51 oranında ağırlıklı olarak komedi türü içerdiği görülmektedir.



Şekil 5.9. BCOJS algoritması sonucu 24 numaralı kümedeki film türlerinin yüzde dağılımı pasta grafiği

5.6. BCOJS Algoritmasının Sonuçlarının Diğer Yöntemlerle Karşılaştırılması

Bu bölümde BCOJS yönteminin doğruluğu diğer kümeleme yöntemlerinin doğruluk sonuçları ile kıyaslanarak sonuçların tartışması yapılmıştır. Bu karşılaştırmayı yapmak için bu bölümde verilen tabloların tümünde, küme sayısı 15 verilerek tüm algoritmalar için aynı küme sayısı kullanılmıştır. Bu karşılaştırma tablolarında yer alan değerler için tüm algoritmaların 30 çalışma sonucunda elde edilen değerlerin ortalaması kullanılmıştır.

Tablo 5.7.'de JS, BCO, BCOJS, OCO, OBCO algoritmalarının doğruluk metriğine göre değerlerinin karşılaştırması verilmiştir. Tablo 5.7. incelendiğinde doğruluk değeri en yüksek %89.46 ile BCOJS yönteminin bulunduğu görülür. Bu veri seti için doğruluk metriği sonucuna göre bir karşılaştırma yapıldığında grup içi ve gruplar arasındaki ölçümün BCOJS yönteminde en iyi olduğu, sonrasında ise BCO yönteminin takip ettiği görülmektedir. Tablo 5.7.'de yöntemlerin karşılaştırılması yapılırken JS metriği, veriden seçilen rastgele 15 küme merkezi ile yalnızca ilgilerine göre kümelenmiştir. JS ile kümeleme sonucu oluşan doğruluk %76.08 olarak hesaplanmıştır. BCO ve JS hibritleştirilmesi ile geliştirilen BCOJS yönteminin hem JS ile hem BCO ile kümeleme sonucundan daha yüksek doğruluk değeri elde ederek her iki yöntemin avantajlarını birleştirdiği görülmektedir.

Tablo 5.7. JS, BCO, BCOJS, OCO, OBCO algoritmalarının doğruluk metriği sonucu elde edilen değerlerinin karşılaştırılması

	Algoritma	Doğruluk
1	JS	% 76.0812
2	BCO	% 86.6970
3	BCOJS	% 89.4628
4	OCO	% 39.2000
5	OBCO	% 70.1224

Küme geçerlilik indeksinde kullanılan bazı metrikler küme sayısı arttıkça monoton azalma eğiliminde olabilmektedir. Ayrıca küme geçerlilik indeksleri ayrıklık, kompaktlık ve ya her ikisi birden olmak üzere küme oluşumlarını farklı yönlerden test etmektedir. Bu sebeple, tek bir performans ölçütü kullanmak bir yöntemin doğruluğunu açıklamada yetersiz kalmaktadır. Bu tez kapsamında önerilen yöntemin doğruluğunu göstermek için bulanık kümeleme yöntemlerinde sıklıkla tercih edilen BE, BK, UBE, UBK, FS, SCG, XB olmak üzere 7 farklı performans indeksi kullanılmıştır. Farklı küme geçerlilik indeksleri için Tablo 5.8. incelendiğinde BCOJS yönteminin sonuçlarının 7 farklı küme geçerlilik indeksinden 6'sında en başarılı olduğu, hesaplanan indeksler için BCO ve diğer iki yöntemden daha iyi olduğu görülmektedir.

BK, UBK, SC indeksleri artan değerlerde daha başarılı kümeleme olduğunu gösterir. BCOJS, BCO, OBCO, OCO yöntemleri BK, UBK, SC indekslerine göre karşılaştırıldığında, BCOJS yöntemine ait küme geçerlilik indeks değerlerinin Tablo

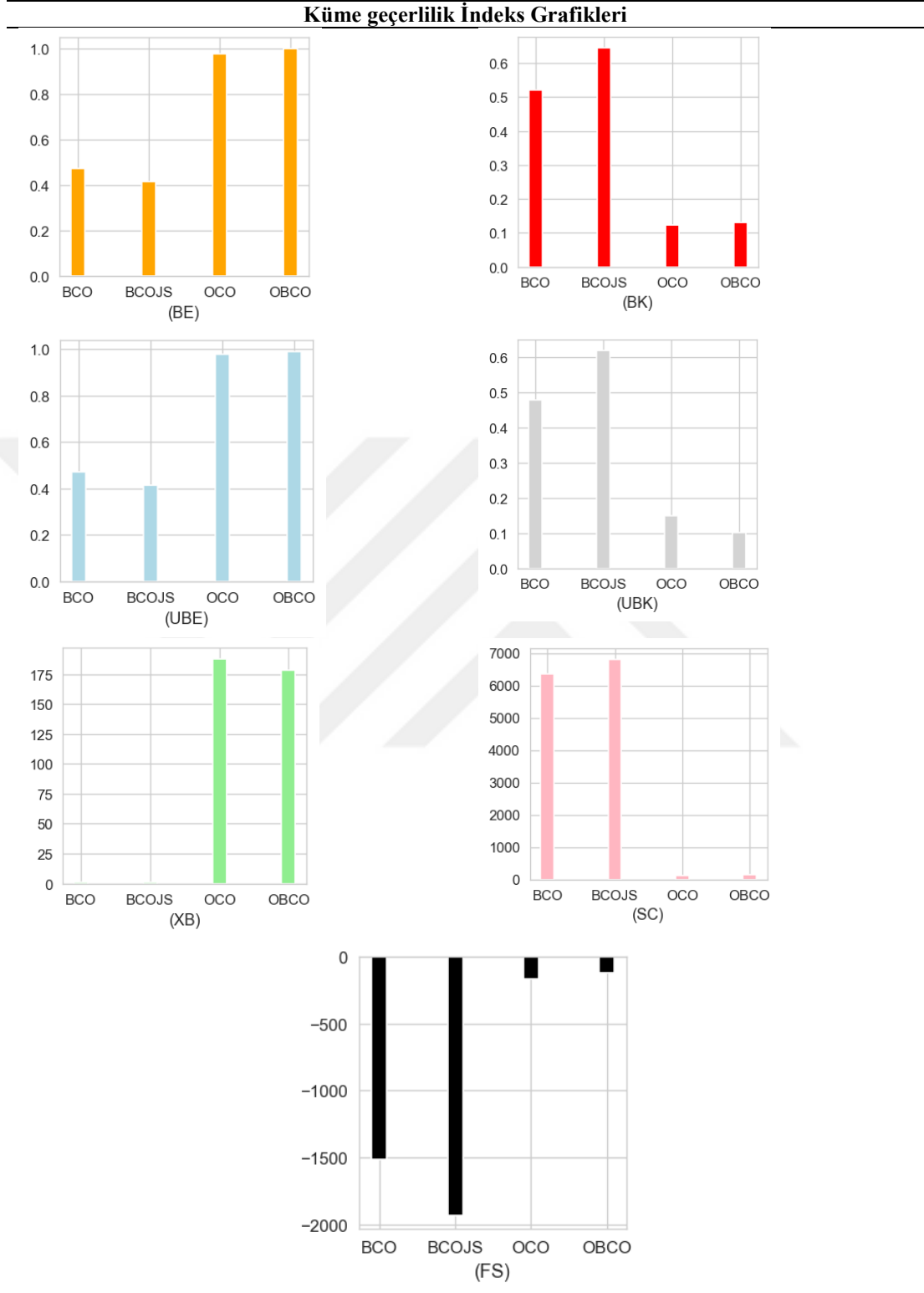
5.8.'de verilen çizelge ve Tablo 5.9.'da verilen sonuç grafiklerine göre daha yüksek çıktığı görülmektedir.

BE, UBE, XB, FS küme geçerlilik indeksleri azalan değerlerde daha başarılı kümeleme yapıldığını gösterir. Tablo 5.8.'de verilen çizelge ve Tablo 5.9.'da verilen sonuç grafikleri incelendiğinde BE, UBE, FS indekslerinin BCOJS yöntemi için daha başarılı sonuçlar verdiği çıkarımı yapılır. XB indeksinin BCO ve BCOJS'de 0.0320 ve 0.0332 değerleri ile 0 ' ya yakın ve neredeyse aynı sonucu verdiği görülmektedir. OCO ve OBCO yöntemlerinin küme geçerlilik indekslerine göre sonuçları bu veri seti için yorumlandığında ayrıklık ve kompaktlık yönünden başarısız olduğu sonucuna varılır.

Tablo 5.8. BCO, BCOJS, OCO, OBCO algoritmaları için küme geçerlilik indekslerinin karşılaştırılması

Küme Geçerlilik İndeksleri	Başarı Değişimi	BCO	BCOJS	OCO	OBCO
BE	(-)	0.4704	0.4146	0.9762	0.9982
BC	(+)	0.5200	0.6453	0.1228	0.1300
UBE	(-)	0.4713	0.4154	0.9781	0.9896
UBK	(+)	0.4800	0.6200	0.1495	0.1011
XB	(-)	0.0320	0.0332	187.6666	178.1601
SC	(+)	6344.8800	6810.2000	29.7841	61.4416
FS	(-)	-1517.1017	-1933.2115	166.3219	-121.7111

Tablo 5.9. BCO, BCOJS, OCO, OBCO algoritmaları için küme geçerlilik indekslerinin sonuç grafikleri



5.7. BCOJS Yönteminde Eşik Değerinin Adaptifleştirilmesi ile Elde Edilen Sonuçlar

Yapay veri seti ve film veri seti, eşik değerinin algoritma tarafından belirlendiği adaptif BCOJS yöntemi ile test edilerek sonuçları diğer yöntemlerle karşılaştırılmıştır.

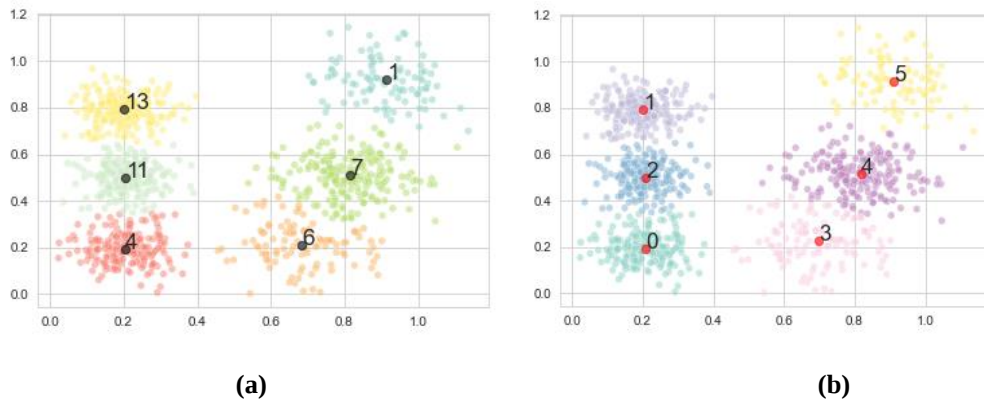
5.7.1. Yapay Veri Seti İçin Sonuçların Değerlendirilmesi

Şekil 5.1.'de gösterilen 6 kümeli yapay veri seti, adaptif BCOJS yöntemi ile çalıştırıldığında tahmin edilen küme sayıları Tablo 5.10.'da verilmiştir. Tablo 5.10.'da gerçek küme sayısı 6 olan veri setinin, maksimum küme sayıları 10, 12, 20, 50, 100 verildiğinde tahmin edilen küme sayısının 6 olduğu görülmektedir. Maksimum küme sayısı 25 verildiğinde ise küme sayısı bazen 5, bazen 6 olarak tahmin edilmiştir.

Tablo 5.10. Yapay veri seti için adaptifleştirilmiş BCOJS yöntemi ile tahmin edilen küme sayısı

Maksimum Küme Sayısı	Tahmin Edilen Küme Sayıları
10	6
12	6
20	6
25	5, 6
50	6
100	6

Şekil 5.10.'da adaptifleştirilmiş BCOJS yöntemi ile BCO yönteminin çalıştırılması sonucu elde edilen kümeleme dağılımı gösterilmiştir. Şekil 5.10. incelendiğinde iki kümeleme sonucunun benzerlik gösterdiği görülmektedir. Adaptifleştirilmiş BCOJS yöntemi algoritma içinde belirlenen bir eşikle aralarındaki ilgiye göre gerçek küme sayısını bilmeden kümeleme yapmıştır.



Şekil 5.10. (a): Yapay veri setinin adaptifleştirilmiş BCOJS ile kümelenmesi, (b): Yapay veri setinin BCO ile kümelenmesi

Tablo 5.11. incelendiğinde adaptifleştirilmiş BCOJS ve BCO yönteminin küme merkezleri karşılaştırılmıştır. Her iki yöntemin küme merkezlerinin birbirine yakın olduğu görülmektedir.

Tablo 5.11. Yapay veri seti için adaptifleştirilmiş BCOJS ve BCO yöntemlerinin küme merkezlerinin karşılaştırılması

Merkezler	Adaptifleştirilmiş BCOJS algoritmasının küme merkezleri		BCO algoritmasının küme merkezleri	
	Boyut - 1	Boyut - 2	Boyut - 1	Boyut - 2
Merkez-0	0.2048	0.1939	0.2072	0.1936
Merkez-1	0.2000	0.7929	0.1996	0.7913
Merkez-2	0.2062	0.4988	0.2090	0.5005
Merkez-3	0.6839	0.2106	0.6987	0.2279
Merkez-4	0.8140	0.5086	0.8183	0.5149
Merkez-5	0.9126	0.9195	0.9089	0.9129

Tablo 5.12.'de adaptifleştirilmiş BCOJS yönteminin diğer yöntemler ile karşılaştırılmasının sonuçları verilmiştir. Bu karşılaştırmada 30 çalışmanın ortalaması sonucu elde edilen değerler verilmiştir. Tablo 5.12.'de verilen sonuçlara göre adaptif BCOJS %81.9366 ile en iyi doğruluk yüzdesini elde etmiştir. BCO %81.7059, BCOJS %81.6864 oranında doğruluk elde etmiştir. Adaptif BCOJS, küme sayısını bilmeden ve eşik değerinin algoritma tarafından belirlendiği bu yöntemle diğer yöntemlerden daha iyi bir sonuç elde etmiştir.

Tablo 5.12. Doğruluk metriği ile JS, BCO, BCOJS, adaptif BCOJS, OCO, OBCO yöntemlerinin karşılaştırılması

	Algoritma	Doğruluk
1	JS	% 52.5603
2	BCO	% 81.7059
3	BCOJS	% 81.6864
4	Adaptif BCOJS	%81.9366
5	OCO	% 42.9304
6	OBCO	% 77.0532

Tablo 5.13.'te adaptif BCOJS yöntemi BCO, BCOJS, OCO, OBCO yöntemleri ile 7 farklı küme geçerlilik indeksi, 30 çalışma sonucunda elde edilen ortalamanın değerine yer verilerek kıyaslanmıştır. Bu tabloya göre artan değerlerde daha iyi sonuç veren BK, UBK, SC küme geçerlilik indeksleri ve azalan değerlerde daha iyi sonuç veren BE, UBE, XB, FS küme geçerlilik indeksleri ile karşılaştırma yapıldığında adaptif BCOJS yönteminin BCO, OCO ve OBCO yöntemine kıyasla başarılı olduğu görülmektedir.

Tablo 5.13. BCO, BCOJS, adaptif BCOJS, OCO, OBCO yöntemlerinin küme geçerlilik indeksleri ile kıyaslanması

Metrikler	Başarı Değişimi	Adaptif BCOJS	BCO	BCOJS	OCO	OBCO
BE	(-)	0.2734	0.2745	0.2740	0.741422	0.6138
BK	(+)	0.7129	0.7112	0.7117	0.23674	0.3227
UBE	(-)	0.2751	0.2762	0.2760	0.744590	0.6101
UBK	(+)	0.6555	0.6534	0.6541	0.19240	0.2025
XB	(-)	0.09379	0.09561	0.09478	1903827.4098	0.5757
SC	(+)	1409.5734	1394.5061	1379.3676	12.9017	102.7029
FS	(-)	-104.2678	-103.46	-103.28	13.4861	-19.0400

BCOJS ve adaptif BCOJS sadece eşik değerini belirleme yönüyle farklı olan yöntemlerdir. BCOJS yönteminde eşik değeri girdi parametrelerinde belirlenirken, adaptif BCOJS yönteminde eşik değeri algoritma tarafından belirlenir. Tablo 5.12. ve Tablo 5.13. karşılaştırıldığında adaptif BCOJS yönteminin BCOJS yöntemi kadar başarılı sonuçlar gösterdiği görülmektedir.

5.7.2. Film Veri Seti İçin Sonuçların Değerlendirilmesi

Bölüm 5.3.'te düzenlenen film veri seti maksimum küme sayısı 25, 31, 40 ve 50 verildiğinde tahmin edilen küme sayısı Tablo 5.14.'te gösterilmiştir. Tablo 5.10 ve Tablo 5.14. incelendiğinde hem yapay veri seti hem de film veri seti için adaptif BCOJS yönteminin farklı maksimum küme sayılarındaki küme tahminlerinde kararlı davrandığı görülmektedir. Tablo 5.10'da algoritma yapay veri setine uyarlandığında maksimum küme sayısı 10, 20, 50, 100 gibi çok farklı aralıklarda kullanılmıştır. Buna karşın tahmin edilen küme sayısı, çoğu durumda gerçek küme sayısı olan 6 değerini bazen de gerçek küme sayısına çok yakın olan 5 değerini almıştır. Benzer durum gerçek küme sayısı kesin olmayan film veri seti için de geçerlidir. Film veri seti için adaptif BCOJS algoritması aynı başlangıç şartlarında farklı maksimum küme sayıları ile denendiğinde tahmin edilen küme sayıları aynıdır. Algoritmanın farklı maksimum küme sayıları için tahmin ettiği küme sayıları her iki veri seti için de kararlılık göstermektedir.

Tablo 5.14. Film veri seti için adaptifleştirilmiş BCOJS yöntemi ile tahmin edilen küme sayısı

Maksimum Küme Sayısı	Tahmin Edilen Küme Sayıları
25	14
31	14
40	14
50	14

Tablo 5.15.'te adaptifleştirilmiş BCOJS yönteminin film veri seti için diğer yöntemler ile karşılaştırılmasının sonuçları verilmiştir. Tablo 5.15.'te verilen sonuçlara göre adaptif BCOJS %89.0593 ile en yüksek doğruluk yüzdesine ulaşan yöntem olmuştur. Adaptif BCOJS yönteminin hem JS hem BCO ile kümeleme sonucundan daha yüksek doğruluk değeri elde ederek her iki yöntemin avantajlarını birleştirdiği görülmektedir.

Tablo 5.15. Doğruluk metriği ile JS, BCO, adaptif BCOJS, OCO, OBCO yöntemlerinin karşılaştırılması

	Algoritma	Doğruluk
1	JS	% 72.6527
2	BCO	% 87.7012
3	Adaptif BCOJS	% 89.0593
4	OCO	% 37.7300
5	OBCO	% 69.6524

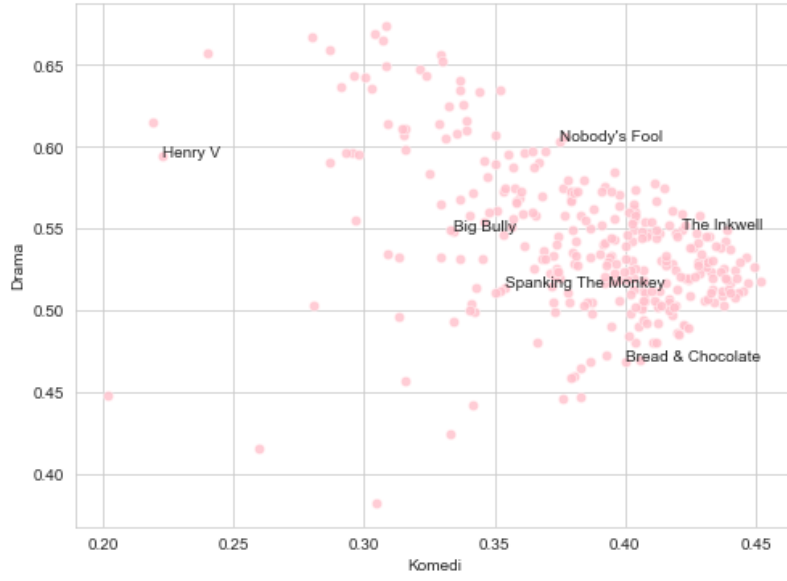
Tablo 5.16. incelendiğinde adaptif BCOJS yönteminin sonuçlarının 7 farklı küme geçerlilik indeksinin tümünde diğer yöntemlerden daha başarılı olduğu görülmektedir.

Tablo 5.16. BCO, adaptif BCOJS, OCO, OBCO algoritmaları için küme geçerlilik indekslerinin karşılaştırılması

Küme Geçerlilik İndeksleri	Başarı Değişimi	BCO	BCOJS	OCO	OBCO
BE	(-)	0.4542	0.3782	0.9658	0.9522
BC	(+)	0.5405	0.6845	0.1400	0.1501
UBE	(-)	0.4550	0.3789	0.9662	0.9585
UBK	(+)	0.5052	0.6602	0.1377	0.1455
XB	(-)	0.03519	0.0109	181.6600	169.0242
SC	(+)	7149.8883	7150.0802	35.9880	66.6402
FS	(-)	-1568.9578	-2158.4487	161.6901	-127.2987

5.8. Geliştirilen Yöntemin Öneri Sistemi Üzerinden Gösterilmesi

Bu bölümde geliştirilen bulanık kümeleme yönteminin, kullanıcıya izlediği film üzerinden yaptığı öneri bir örnekle anlatılmaktadır. Örneğin, 1994 yapımı Nobody's Fool filmini izleyen bir kullanıcı bu filme benzer türde başka bir film izlemek istediğinde, öneri yöntemi ilk olarak kullanıcının izlediği filmin atandığı kümeye gider. Nobody's Fool filmi Şekil 5.11.'de gösterildiği üzere BCOJS algoritması ile 14 numaralı küme grubuna atanmıştır. 14 numaralı küme grubunda, yer alan filmler kullanıcıya önerilir.



Şekil 5.11. Nobody's Fool filmi ile benzer filmlerin listesi

Şekil 5.11.'de kullanıcının izlediği Nobody's Fool filmi için kullanıcıya önerilen benzer filmler gösterilmiştir. Nobody's Fool, komedi ve dram türlerini içeren bir filmidir. Bu kategoride yer alan filmlerden Nobody's Fool filmi ile ilgisi bulunan komedi-dram filmlerinin BCOJS ile tür dağılımı Tablo 5.17.'de verilmiştir. Nobody's Fool filmi ile benzer bir film izlemek isteyen kullanıcı Tablo 5.17.'de verilen diğer 5 filmi izleyebilir.

Tablo 5.17. No Body's Fool filmi ile benzer 5 filmin tür dağılımı

Film Adı	Macera	Aksiyon	Komedi	Drama	Korku
No Body's Fool	0.0032	0.0023	0.3740	0.6000	0.0115
Henry V	0.0380	0.1000	0.2310	0.5945	0.0355
Big Bully	0.2591	0.0691	0.3341	0.5488	0.0218
The Inkwell	0.0031	0.0119	0.4217	0.5498	0.0133
Bread & Chocolate	0.0050	0.0081	0.336	0.634	0.014
Spanking The Monkey	0.1134	0.0120	0.3541	0.5143	0.0060

6. SONUÇ

Bu tez kapsamında, bulanık ve ilgi tabanlı bir kümeleme yöntemi önerilmiştir. Geliştirilen yöntemde BCO yöntemi ile JS metriği melezleştirilmiş ve veriye ait küme sayısının bilinmediği durumlarda uygun küme sayısının tahmin edilmesi hedeflenmiştir. Çeyrekler açıklığı ile eşik belirleme stratejisi kullanılarak önerilen BCOJS yönteminin JS eşik değeri parametresinin algoritma tarafından belirlenmesini sağlayan adaptif bir BCOJS yöntemi geliştirilmiştir. BCOJS ve adaptif BCOJS yönteminin kümeleme başarımını değerlendirmek amacıyla 6 kümeden oluşan yapay bir veri seti oluşturulmuştur. BCOJS ve adaptif BCOJS yönteminin, küme sayısını doğru tahmin ettiği gözlenmiştir. Önerilen yöntemlerin BCO yöntemi ile küme merkezlerinin denk olması yöntemlerin başarısını doğrulamaktadır. BCOJS ve adaptif BCOJS yönteminin BCO, OBCO, OCO, JS ile kümeleme sonuçları kıyaslanarak karşılaştırılmıştır. Karşılaştırma için doğruluk metriği ve 7 farklı küme geçerlilik indeksleri sonucu elde edilen değerler incelendiğinde BCOJS ve adaptif BCOJS yönteminin başarılı olduğu sonucuna varılmıştır.

Başarımı doğrulanan BCOJS yöntemi, film veri setine uygulanarak benzer filmleri önermek için kullanılmıştır. BCOJS algoritması hazırlanan film veri seti ile maksimum küme sayısı 25 tutularak 0.35, 0.40, 0.45 olarak 3 farklı eşik değeri için kümelendirilmiştir. Kümeleme sonuçları küme geçerlilik indekslerine göre karşılaştırılmış ve uygun eşik değeri 0.45 olarak belirlenmiştir. Eşik değeri ile başlangıçta 25 olan maksimum küme sayısı kümeleme sonucunda 15 olarak tahmin edilmiştir. Yapay veri setinde kullanılan doğrulama metriği ve küme geçerlilik indeksleri, film veri setinde de kullanılmıştır. Böylece BCOJS yönteminin 4 farklı yöntemle performans karşılaştırması yapılmıştır. Elde edilen sonuçlara göre BCOJS algoritmasının doğruluk metriği değeri %89.4628 ile JS, OCO, OBCO, BCO yöntemlerinden daha yüksek bulunmuştur. BCO yöntemi doğruluk metriği sonucu %86.6970 ile en iyi ikinci yöntem olarak bulunmuştur. BCOJS, OCO, OBCO, BCO yöntemlerinin, küme geçerlilik indekslerindeki kümeleme sonuçları karşılaştırıldığında BCOJS yöntemi başarılı bulunmuştur. Performans indeksleri ve doğruluk metriği BCOJS yöntemi ile elde edilen 15 küme grubunu başarılı bulmuştur. Adaptif BCOJS ile yapılan kümeleme sonuçlarına göre algoritma tarafından belirlenen eşik değeri ile küme sayılarının doğru tahmin edildiği sonucuna varılmıştır. Adaptif BCOJS yöntemi aynı başlatma koşullarında farklı sayıda maksimum küme sayıları ile test edildiğinde algoritmanın

küme sayılarını benzer değerlerde tahmin ettiği gözlenmiştir. Herhangi bir film için bu küme gruplarındaki tür dağılımlarında aldığı üyelikler sayısal değerler açısından incelendiğinde, yöntemin başarımı gözlemlenmektedir.

BCO algoritması veriler arasındaki ilgiyi dikkate almaz ve verilen küme sayısı kadar küme bulmaya odaklıdır. JS algoritması ise sadece ilgi tabanlı ve küme merkezinin hesaplanmadığı bir yöntemdir. BCO ile sadece istenen sayıda kümeye göre kümeleme yapılabilmesi, JS ile yapılan kümelemede ise küme merkezlerini belirleyecek bir mekanizmanın eksikliği BCOJS yöntemi ile giderilmiştir. Adaptif BCOJS yöntemi de BCOJS yönteminin eşik değeri belirleme zorluğunu gidererek küme sayısını kararlı bir şekilde doğru tahmin etmiştir. Bu açıdan BCOJS ve adaptif BCOJS yöntemlerinin, ilgi ilişkisinin önemli olduğu film öneri sisteminde küme sayısı bilinmeden optimum küme sayısının bulunmasında etkili bir yöntem olduğu sonucuna varılmıştır.

KAYNAKLAR

- Abbas, A., Zhang, L., and Khan, S. U. (2015). A survey on context-aware recommender systems based on computational intelligence techniques. *Computing*, 97, 667-690.
- 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.
- Ahmadian, S., Ahmadian, M., and Jalili, M. (2022). A deep learning based trust-and tag-aware recommender system. *Neurocomputing*, 488, 557-571.
- Ahuja, R., Solanki, A., and Nayyar, A. (2019, January). Movie recommender system using k-means clustering and k-nearest neighbor. In *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (263-268). IEEE.
- Alamdari, P. M., Navimipour, N. J., Hosseinzadeh, M., Safaei, A. A., & Darwesh, A. (2020). A systematic study on the recommender systems in the E-commerce. *Ieee Access*, 8, 115694-115716.
- Aydin, N., ve Kayhan, G. (2021). Jensen Shannon Mesafesi Temelli Uyarlanmış Bulanık C Ortalamalar Kümeleme Yöntemi. *Avrupa Bilim ve Teknoloji Dergisi*, (29), 58-64.
- Ayub, M., Ghazanfar, M. A., Mehmood, Z., Alyoubi, K. H., & Alfakeeh, A. S. (2020). Unifying user similarity and social trust to generate powerful recommendations for smart cities using collaborating filtering-based recommender systems. *Soft Computing*, 24, 11071-11094.
- Bezdek, J. C., Ehrlich, R., and Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & geosciences*, 10(2-3), 191-203.
- Bobadilla, J., Ortega, F., Hernando, A., and Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-based systems*, 46, 109-132.
- Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12, 331-370.
- Chen, R., Hua, Q., Chang, Y. S., Wang, B., Zhang, L., and Kong, X. (2018). A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access*, 6, 64301-64320.
- Cho, S. B., and Yoo, S. H. (2006). Fuzzy Bayesian validation for cluster analysis of yeast cell-cycle data. *Pattern recognition*, 39(12), 2405-2414.
- Cui, Z., Xu, X., Fei, X. U. E., Cai, X., Cao, Y., Zhang, W., and Chen, J. (2020). Personalized recommendation system based on collaborative filtering for IoT scenarios. *IEEE Transactions on Services Computing*, 13(4), 685-695.
- Duan, R., Jiang, C., and Jain, H. K. (2022). Combining review-based collaborative filtering and matrix factorization: A solution to rating's sparsity problem. *Decision Support Systems*, 156, 113748.
- Dunn, J. C. (1974). Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics*, 4(1), 95-104.
- Felfernig, A., Polat-Erdeniz, S., Uran, C., Reiterer, S., Atas, M., Tran, T. N. T., ... and Dolui, K. (2019). An overview of recommender systems in the internet of things. *Journal of Intelligent Information Systems*, 52(2), 285-309.
- Gazdar, A., and Hidri, L. (2020). A new similarity measure for collaborative filtering based recommender systems. *Knowledge-Based Systems*, 188, 105058.

- Gedikli, F., Jannach, D., and Ge, M. (2014). How should I explain? A comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, 72(4), 367-382.
- 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.
- Gomez-Urbe, C. A., and Hunt, N. (2015). The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS)*, 6(4), 1-19.
- Hamid, R. A., Albahri, A. S., Alwan, J. K., Al-Qaysi, Z. T., Albahri, O. S., Zaidan, A. A., ... and Zaidan, B. B. (2021). How smart is e-tourism? A systematic review of smart tourism recommendation system applying data management. *Computer Science Review*, 39, 100337.
- Hernando, A., Bobadilla, J., and Ortega, F. (2016). A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model. *Knowledge-Based Systems*, 97, 188-202.
- Himeur, Y., Alsalemi, A., Al-Kababji, A., Bensaali, F., Amira, A., Sardianos, C., ... and Varlamis, I. (2021). A survey of recommender systems for energy efficiency in buildings: Principles, challenges and prospects. *Information Fusion*, 72, 1-21.
- Jalili, M., Ahmadian, S., Izadi, M., Moradi, P., and Salehi, M. (2018). Evaluating collaborative filtering recommender algorithms: a survey. *IEEE access*, 6, 74003-74024.
- Katarya, R., and Verma, O. P. (2017). An effective web page recommender system with fuzzy c-mean clustering. *Multimedia Tools and Applications*, 76, 21481-21496.
- Kluver et al., J. A. (2018). Rating-based collaborative filtering: algorithms and evaluation. *Social information access: Systems and technologies*, 344-390. Springer.
- Koohi, H., and Kiani, K. (2016). User based Collaborative Filtering using fuzzy C-means. *Measurement: Journal of the International Measurement Confederation*, 91, 134-139.
- Krishnapuram, R., and Keller, J. M. (1993). A possibilistic approach to clustering. *IEEE transactions on fuzzy systems*, 1(2), 98-110.
- Kumar, P., and Thakur, R. S. (2018). Recommendation system techniques and related issues: a survey. *International Journal of Information Technology*, 10, 495-501.
- Lin, J. (1991). Divergence measures based on the Shannon entropy. *IEEE Transactions on Information theory*, 37(1), 145-151.
- Liu, Y., Zhang, X., Chen, J., and Chao, H. (2019). A validity index for fuzzy clustering based on bipartite modularity. *Journal of Electrical and Computer Engineering*, 2019.
- Lops, P., Jannach, D., Musto, C., Bogers, T., and Koolen, M. (2019). Trends in content-based recommendation: Preface to the special issue on Recommender systems based on rich item descriptions. *User Modeling and User-Adapted Interaction*, 29, 239-249.
- Lu, J., Wu, D., Mao, M., Wang, W., and Zhang, G. (2015). Recommender system application developments: a survey. *Decision support systems*, 74, 12-32.
- Melville, P., Mooney, R. J., and Nagarajan, R. (2002). Content-boosted collaborative filtering for improved recommendations. *Aaai/iaai*, 23, 187-192.
- Nayak, J., Naik, B., and Behera, H. (2014). Fuzzy C-means (FCM) clustering algorithm: a decade review from 2000 to 2014. *Smart Innovation, Systems and Technologies*, (s.133-149) Springer New Delhi

- Nedaie, A., and Najafi, A. A. (2018). Support vector machine with Dirichlet feature mapping. *Neural Networks*, 98, 87-101.
- Özdemir, Ö., and Kaya, A. A. (2019). Comparison of FCM, PCM, FPCM and PFCM algorithms in clustering methods. *Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi*. 92–102.
- Ozer, M. (2005). Fuzzy c-means clustering and Internet portals: A case study. *European Journal of Operational Research*, 164(3), 696-714.
- Pal, N. R., and Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model. *IEEE Transactions on Fuzzy systems*, 3(3), 370-379.
- Pal, N. R., Pal, K., Keller, J. M., and Bezdek, J. C. (2005). A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems*, 13(4), 517–530.
- Patel, K., and Patel, H. B. (2020). A state-of-the-art survey on recommendation system and prospective extensions. *Computers and Electronics in Agriculture*, 178, 105779.
- Pérez-Almaguer, Y., Yera, R., Alzahrani, A. A., and Martínez, L. (2021). Content-based group recommender systems: A general taxonomy and further improvements. *Expert Systems with Applications*, 184, 115444.
- Quijano-Sánchez, L., Cantador, I., Cortés-Cediel, M. E., and Gil, O. (2020). Recommender systems for smart cities. *Information systems*, 92, 101545.
- Reddy, S., Nalluri, S., Kunisetti, S., Ashok, S., and Venkatesh, B. (2019). Content-based movie recommendation system using genre correlation. *Smart Innovation, Systems and Technologies*, 105, 391–397.
- Ruspini, E. H., Bezdek, J. C., and Keller, J. M. (2019). Fuzzy clustering: A historical perspective. *IEEE Computational Intelligence Magazine*, 14(1), 45–55.
- Salter, J., and Antonopoulos, N. (2006). CinemaScreen recommender agent: combining collaborative and content-based filtering. *IEEE Intelligent Systems*, 21(1), 35-41.
- Schedl, M., Zamani, H., Chen, C. W., Deldjoo, Y., and Elahi, M. (2018). Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval*, 7, 95-116.
- Schenatto, K., De Souza, E. G., Bazzi, C. L., Gavioli, A., Betzek, N. M., and Beneduzzi, H. M. (2017). Normalization of data for delineating management zones. *Computers and Electronics in Agriculture*, 143, 238-248.
- Sengupta, S., De, S., Konar, A., and Janarthanan, R. (2011, December). An improved fuzzy clustering method using modified Fukuyama-Sugeno cluster validity index. In *2011 International Conference on Recent Trends in Information Systems* (pp. 269-274).
- Shambour, Q. (2021). A deep learning based algorithm for multi-criteria recommender systems. *Knowledge-based systems*, 211, 106545.
- Tarus, J. K., Niu, Z., and Mustafa, G. (2018). Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. *Artificial intelligence review*, 50, 21-48.
- Valdiviezo-Diaz, P., Ortega, F., Cobos, E., and Lara-Cabrera, R. (2019). A Collaborative Filtering Approach Based on Naïve Bayes Classifier. *IEEE Access*, 7, 108581–108592.
- Walek, B., and Fojtik, V. (2020). A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, 158, 113452.
- Wang, X., and Wang, Y. (2014, November). Improving content-based and hybrid music recommendation using deep learning. In *Proceedings of the 22nd ACM international conference on Multimedia*, Association for Computing Machinery, New York, USA.

- Wang, Z., Yu, X., Feng, N., and Wang, Z. (2014). An improved collaborative movie recommendation system using computational intelligence. *Journal of Visual Languages & Computing*, 25(6), 667-675.
- Yang, Z., Wu, B., Zheng, K., Wang, X., and Lei, L. (2016). A survey of collaborative filtering-based recommender systems for mobile internet applications. *IEEE Access*, 4, 3273-3287.
- Yassine, A. F. O. U. D. I., Mohamed, L. A. Z. A. A. R., and Al Achhab, M. (2021). Intelligent recommender system based on unsupervised machine learning and demographic attributes. *Simulation Modelling Practice and Theory*, 107, 102198.
- Zhang, Q., Lu, J., and Jin, Y. (2021). Artificial intelligence in recommender systems. *Complex & Intelligent Systems*, 7, 439-457.
- Zhang, X., Liu, J., Zhang, Q., and Zhu, W. (2002, November). gMeasure: A group-based network performance measurement service for peer-to-peer applications. In *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE, Taipei, Taiwan*
- Zhao, Z. D., and Shang, M. S. (2010). User-based collaborative-filtering recommendation algorithms on hadoop. *2010 Third International Conference on Knowledge Discovery and Data Mining, Phuket, Thailand*.
- Zhou, K. le, Fu, C., and Yang, S. L. (2014). Fuzziness parameter selection in fuzzy c-means: The perspective of cluster validation. *Science China Information Sciences*, 57(11), 1–8.

ÖZ GEÇMİŞ

Samsun Anadolu İmam Hatip Lisesi'nden mezun olduktan sonra, 2014 yılında Samsun Ondokuz Mayıs Üniversitesi Mühendislik Fakültesi, Bilgisayar Mühendisliği bölümünde lisans eğitimine başladı. 2018 yılında buradan mezun oldu. 2019 yılında OMÜ Bilgisayar Mühendisliği Ana Bilim Dalı Yüksek Lisans programına girdi. 2021 senesinden bu yana özel sektörde yazılım mühendisi olarak çalışmaktadır. Yabancı dil bilgisi olarak ingilizce bilmektedir.

İletişim Bilgileri

ORCID ID : <https://orcid.org/0000-0002-6261-6121>

Yayınlar:

1. Aydın, N., & Kayhan, G. (2021). Jensen Shannon Mesafesi Temelli Uyarlanmış Bulanık C Ortalamalar Kümeleme Yöntemi. *Avrupa Bilim ve Teknoloji Dergisi*, (29), 58-64.