ÇOK ÖLÇÜTLÜ OY DEĞERLERİ ÜZERİNDE EN İYİ-N ÖNERİ SİSTEMİ VE ŞİLİN ATAKLARIN ETKİSİ

Tuğba KAYA

Doktora Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Bilimleri Bilim Dalı

Danışman: Prof. Dr. Cihan KALELİ

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Mayıs 2022

Bu tez çalışması BAP Komisyonu tarafından kabul edilen 20DRP034 no.lu proje kapsamında desteklenmiştir.

13/05/2022

DANIŞMAN ONAYI

Danışmanlığını yürüttüğüm Doktora öğrencisi Tuğba KAYA, ÇOK ÖLÇÜTLÜ OY DEĞERLERİ ÜZERİNDE EN İYİ-N ÖNERİ SİSTEMİ VE ŞİLİN ATAKLARIN ETKİSİ başlıklı tez çalışmasını tamamlamıştır. Hazırlamış olduğu tez tarafımca incelenmiş ve öğrencinin tez savunma sınavına alınması bilimsel ve etik açıdan uygun görülmüştür.

Tez Danışmanı

Prof. Dr. Cihan KALELİ

ÖZET

ÇOK ÖLÇÜTLÜ OY DEĞERLERİ ÜZERİNDE EN İYİ-N ÖNERİ SİSTEMİ VE ŞİLİN ATAKLARIN ETKİSİ

Tuğba KAYA

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Bilimleri Bilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Mayıs 2022

Danışman: Prof. Dr. Cihan KALELİ

Öneri sistemleri, kullanıcıların bir ürün için tahminler oluşturmasına ve kullanıcıların geri bildirimlerine dayalı olarak ürün listeleri sunulmasına yardımcı olan popüler yöntemlere sahiptir. Bu tür sistemlerin çıktıları, doğru öneriler üreterek müşteri memnuniyetini artırabilir. Ancak, sistemlerin başarısı, kullanıcı tercih verilerinin kişiselleştirilme düzeyine bağlıdır. Dolayısıyla daha detaylı müşteri verilerinin toplanması ile bu kişiselleştirme seviyesini artırmak mümkündür. Çok ölçütlü öneri sistemleri olarak adlandırılan bu sistemler, artırılmış bu kişiselleştirme seviyeleri ile kullanıcıya daha doğru tavsiyeler sunulmasını sağlar. Ancak literatürdeki çalışmalara bakıldığında çok ölçütlü sistemler üzerine geliştirilmiş bir en iyi-*n* öneri sistemlerinin yer almadığı görülmektedir. Bu tez kapsamında ilk olarak, çok ölçütlü veri kümeleri için yeni bir komşuluk oluşturma süreci ile sezgisel bulanık küme tabanlı en iyi-*n* öneri sistemi yöntemi önerilmektedir. Önerilen yöntem iki önemli noktadan oluşmaktadır: *i)* Ürünler arasındaki ilişkisel yapının belirlenmesi; *ii)* Kullanıcı eğilimlerinin, ayırt edici yapılarının ve derecelendirme dağılımlarının araştırılmasıdır. Derecelendirme dağılımı ve ürünler arasındaki ilişkisel yapı, birliktelik kuralı ve Entropi ölçümü ile belirlenirken, değerlendirme sırasında kullanıcıların tutum ve eğilimleri sezgisel bulanık kümeler kullanılarak analiz edilmektedir. Önerilen yaklaşımların performans karşılaştırması, tek ölçütlü sistemlerde yer alan algoritmanın çok ölçütlü sisteme uyarlanması ile gerçekleştirilmiştir. Kullanıcı memnuniyetinin yanı sıra önerilen listenin şans eser bulma, çeşitlilik ve yenilik gibi farklı ölçümlerde gerçekleştirilmiştir. Önerilen yöntemlerin kullanıcıya oldukça başarılı/doğru tavsiyeler sunduğu ve kullanıcı memnuniyetinin sağlandığı görülmüştür.

Kullanıcıya başarılı tavsiyeler sunmanın yanı sıra, şilin saldırılarına karşı güvenlik açığı, öneri sistemlerinin performansını kötü etkilemektedir. Bu nedenle çalışmanın ikinci aşamasında, ürün tabanlı çok ölçütlü en iyi-*n* öneri sistemlerinin şilin saldırılara karşı ne derece gürbüz olduğunun analizi gerçekleştirilmiştir. Çalışmada ürün tabanlı sistemler de oldukça başarılı yeni bir atak modeli olan güçlü ürün atak tipinin dört farklı varyantı kullanılmıştır. Saldırıların etkinliği için yeni ve mevcut değerlendirme kriterleri, çok ölçütlü sistemlere uyarlanmış bir saldırı modeli, hedef ve güçlü ürün seçimini içeren deneysel bir tasarım kullanılmıştır. Gerçek dünya veri setine dayanan deneysel sonuçlara göre, çok ölçütlü en iyi-*n* algoritmalarının manipülasyonlara açık olduğu görülmüştür.

Manipülasyonlara açık bu sistemleri bu tür saldırılara karşı savunmak için çeşitli yöntemler mevcut olsa da şilin profillerini yakalayarak sağlam sistemler elde etmek, çok ölçütlü derecelendirmeye dayalı olanlar için zor olmaya devam etmektedir. Dolayısıyla son fazda, kullanıcı özelliklerine dayalı olarak sahte profilleri tespit eden bir sınıflandırma yöntemi önerilmiştir. Bu kullanıcı özelliklerine, literatürde var olan genel ve model tabanlı yöntemlere ek olarak ürün popülerliğine ve kullanıcı karakteristik yapısına dayalı yeni özellikler eklenmiştir. Deneysel sonuçlarda, önerilen yöntemin, seçilen boyutunun ve saldırı boyutunun küçük olması durumunda bile gerçek kullanıcılardan saldırı profillerini başarılı bir şekilde ayırt ettiği sonucuna varılmıştır. Ampirik sonuçlar ayrıca, derecelendirme profillerine dayalı ürün popülerliği ve kullanıcı özelliklerinin, şilin saldırı profillerini yakalamada oldukça faydalı özellikler olduğunu göstermiştir.

Anahtar Sözcükler: En iyi-n öneri sistemleri, Çok ölçütlü işbirlikçi filtreleme, Gürbüzlük analizi, Şilin atak, Şilin atak tespiti.

ABSTRACT

TOP-N RECOMMENDER SYSTEM AND EFFECT OF SHILLING ATTACK ON MULTI-CRITERIA RATING VALUES

Tuğba KAYA

Department of Computer Engineering

Programme in Computer Science

Eskişehir Technical University, Institute of Graduate Programs, May 2022

Supervisor: Prof. Dr. Cihan KALELİ

Recommendation systems have popular ways of helping users generate estimates for a product and create product lists based on users' feedback. The outputs of such systems can increase customer satisfaction by generating accurate recommendations. However, the success of the systems depends on the level of customization of the user preference data. Therefore, it is possible to increase this level of personalization by collecting more detailed customer data. These systems, called multi-criteria recommendation systems, provide more accurate recommendations to the user with these increased levels of customization. However, when we look at the studies in the literature, it is seen that there are no top-n recommendation systems developed on multi-criteria systems. In this thesis, firstly, a new neighborhood creation process and an intuitive fuzzy set-based top-n recommendation system method are proposed for multi-criteria datasets. The proposed method consists of two important points: i) determining the relational structure between products; ii) the investigation of user trends, distinctive structures, and rating distributions. While the relational structure between the rating distribution and products is determined by the association rule and Entropy measurement, the attitudes and tendencies of the users are analyzed using intuitionistic fuzzy sets during the evaluation. The performance comparison of the proposed approaches is carried out by adapting the algorithm in the single-criteria systems to the multi-criteria system. In addition to user satisfaction, the proposed list is based on different measures such as serendipity, diversity, and novelty. It has been seen that the suggested methods offer very successful/correct recommendations to the user, and user satisfaction is achieved.

In addition to providing successful recommendations to the user, the vulnerability to shilling attacks adversely affects the performance of recommendation systems.

Therefore, in the second stage of the study, an analysis of how robust the item-based multi-criteria top-n recommendation systems are against shilling attacks is carried out. In the study, four different variants of the power item attack type, which is a very successful new attack model in item-based systems, are used. An experimental design including new and existing evaluation criteria for the effectiveness of attacks, an attack model adapted to multi-criteria systems, a target, and strong product selection is used. According to the experimental results based on the real-world dataset, it has been seen that the multi-criteria top-n algorithms are open to manipulations.

While various methods exist to defend these manipulated systems against such attacks, obtaining robust systems by capturing shilling profiles remains difficult for those based on multi-criteria ratings. Therefore, in the last phase, a classification method is proposed that detects fake profiles based on user characteristics. In addition to the general and model-based methods available in the literature, new features based on product popularity and user characteristics have been added to these user features. From the experimental results, it is concluded that the proposed method successfully distinguishes attack profiles from real users even when the selected size and attack size are small. Empirical results also show that product popularity and user attributes based on rating profiles are highly useful features in capturing shilling attack profiles.

Keywords: Top-n recommender systems, Multi-criteria collaborative filtering, Robustness analysis, Shilling attack, Shilling attack detection.

TEŞEKKÜR

Öncelikle danışmanlığımı üstlenen ve tez çalışmam süresi boyunca konu seçimi ve araştırmanın yürütülmesine dek her türlü konuda desteğini ve bilgi birikimini eksik etmeyen, insani tutumundan kaynaklı kısa sürede güzel işler başarmamı sağlayan danışmanım Sayın Prof. Dr. Cihan KALELİ hocama teşekkürü bir borç bilirim.

Tez izleme komitemde yer alarak değerli görüşleri ile araştırmanın şekillenmesine yardımcı olan hocalarım Sayın Doç. Dr. Zehra KAMIŞLI ÖZTÜRK' e ve Sayın Doç. Dr. Alper Kürşat UYSAL' a çok teşekkür ederim.

Donanımsal desteklerinden ötürü AVKAR YAZILIM' a teşekkürlerimi sunarım.

Her daim güzel yürekleri ile benim bugünlere gelmemdeki süreçte hiçbir fedakarlıktan kaçınmayan yoklarını var eden başta canım babam Hayrettin TÜRKOĞLU ve canım annem Rahime TÜRKOĞLU olmak üzere TÜRKOĞLU ailesine sonsuz minnettarlığımı sunarım. Çalışmanın her adımında yol arkadaşım olan canım eşim Ahmet KAYA' ya sabrı ve yardımlarından ötürü sevgilerimi sunarım. Allah' ın bize bir lütfu olan, bu hayattaki en büyük şükür sebebim olan ve hayatımdaki en olumsuz durumları bile gülümsemesiyle yok eden boncuk gözlü yavrum Yiğit KAYA' ya en derin sevgilerimi ve varlığından ötürü Rabbime şükrümü sunuyorum.

Tuğba KAYA

13/05/2022

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan "bilimsel intihal tespit programı"yla tarandığını ve hiçbir şekilde "intihal içermediğini" beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Tuğba KAYA

İÇİNDEKİLER

Sayfa

TABLOLAR DİZİNİ

Sayfa

ŞEKİLLER DİZİNİ

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

İF	:	İşbirlikçi Filtreleme (Collaborative Filtering)
ÇÖİF	:	Çok Ölçütlü İşbirlikçi Filtreleme (Multi-criteria Collaborative Filtering)
DU	:	Belirsizlik Derecesi (Degree of Uncertainity)
PIA	:	Güçlü Ürün Saldırısı (Power Item Attack)
PIA-MT	:	Güçlü Ürün Saldırısı-Çoklu Hedefler (Power Item Attack- Multiple targets)
SBD	:	Sezgisel Bulanık Değerler (Intuitionistic Fuzzy Set)
YM	:	Yahoo! Movies
ARM	:	Birliktelik Kuralı Madenciliği (Association Rule Mining)
BASEMCCF	:	Temel Çok Ölçütlü İF (Baseline Multi-criteria Collaborative Filtering)
IFMC-EntARMCF	:	Sezgisel Bulanık Çok Ölçütlü Entropi ve ARM-tabanlı İF (Intuitionistic Fuzzy Multi-Criteria Entropy and ARM-based Collaborative Filtering)
CrispMC-EntARMCF	:	Keskin Değerli Çok Ölçütlü Entropi ve ARM-tabanlı İF (Crisp Multi-Criteria Entropy and ARM-based Collaborative Filtering)
PMI	:	Nokta Bazında Karşılıklı Bilgi (Point-wise Mutual Information)
M	:	Eğitim Seti (Train Set)
P	:	Prob Seti (Probe Set)
T	:	Test Seti (Test Set)
β	:	Beta
α	:	Alfa
I_S	:	Seçilen Ürün Kümesi (Selected Items)
I_F	:	Dolgu Ürün Kümesi (Filler Items)
I_T	:	Hedef Ürün Kümesi (Target Items)
I_U	:	Derecelendirilmemiş Ürün Kümesi (Unrated Items)
ID	:	Derece (Indegree)
AS	:	Toplu Benzerlik (Aggregate Similarity)

NR	:	Derecelendirme Sayısı (Number of Rating)
SR	:	Benzerlik Oranı (Sim Ratio)
MUP-G	:	En Popüler Olmayan-Genel (Most Unpopular- General)
MUP-C	:	En Popüler Olmayan-Kriter (Most Unpopular- Criteria)
MUP-U	:	En Popüler Olmayan-Kullanıcı (Most Unpopular- User)
TOD	:	Toplam Öncelik Derecesi (Total Priority Degree)
AvgHR	:	Ortalama İsabet Oranı (Average Hit Ratio)
WRV	:	Ağırlıklandırılmış Sıra Değeri (Weighted Rank Value)
kNN	:	k-en yakın komşular (k-nearest neighbor)
Base	:	Temel (Baseline)
Mix	:	Karma (Mix)

1. GİRİŞ

Son yıllarda bilim ve teknolojinin hızla ilerlemesi kullanıcılarla ilgili büyük miktarda verinin ortaya çıkmasına neden olmuştur. Kullanıcıların online sistemlerde değerlendirmeye aldıkları ürünler/hizmetler için yazdıkları yorumlar, verdikleri oy değerleri, geri bildirimler bu büyük miktarda veriyi oluşturmaktadır. Ancak, kişi internet üzerinden bir ürün/hizmet almak istediğinde, karar verme sürecinde bu aşırı bilgi yığını kişinin çok fazla zaman harcamasına neden olacaktır (Isinkaye, Folajimi, & Ojokoh, 2015). Öneri sistemleri, aşırı bilgi sorunun üstesinden gelebilen ve farklı kullanıcılara ait heterojen kaynaklardan gelen bu büyük verileri analiz ederek kullanıcıların ürün/hizmet seçme sürecinde çok vakit harcamasını engelleyerek doğru seçim yapmasına yardımcı olan etkili bir mekanizmadır.

1.1. Öneri Sistemleri

Öneri sistemlerinin amacı birçok alternatif arasından kullanıcılara ilgilerini çekebilecek anlamlı tavsiyeler üretmektir. Bu sistemler hem kullanıcılar hem de internet satıcısı için oldukça faydalıdır. Amazon' daki kitap önerileri veya Netflix' teki filmler, bu öneri sistemlerinin çevrimiçi hizmet sağlayıcıların nasıl bir işleyişe sahip olduğunun gerçek dünyadaki örnekleridir. Dolayısıyla bu sistemlerin varlığı, kullanıcılara zaman kaybı olmadan doğru hizmet alabilmelerine imkân tanımaktadır.

Genel olarak, öneri sistemleri kullanıcılara, iki şekilde yardımcı olmaktadır (Sarwar, Karypis, Konstan, & Riedl, 2001). İlk olarak, kullanıcılar belirli bir ürün veya ürün grubu için bir tahmini değer (*prediction*) isteyebilir. Örneğin kullanıcı belli bir filmi izleme konusunda kararsız kaldığında öneri sisteminden yardım alabilir ve ardından izleyip izlememe konusunda bilinçli bir karar verebilir. Diğer yönden kullanıcının ilgisini çekebilecek bir sıralı ürün listesi (*recommendation*) sunulmasıdır. Film tabanlı örneğe göre, kullanıcının bir film izlemek istediği ancak hangisini izleyeceğinden emin olmadığı bir senaryoda sistem kullanıcının daha önce gördüğü ve beğendiği filmlere dayanarak ilgisini çekebilecek bir film listesi oluşturabilir. Özetle öneri sistemleri, kullanıcıların ilgi alanlarını tahmin etmeyi ve onlar için oldukça ilginç olabilecek ürün öğelerini sunmayı amaçlar.

Öneri sistemleri, karakteristik bilgilerin kullanıldığı içerik tabanlı filtreleme, kullanıcı-ürün etkileşimine dayanan işbirlikçi filtreleme ve bu iki yaklaşımın olumlu

yönlerinin alındığı hibrit sistemler olarak üçe ayrılır (Adomavicius & Tuzhilin, 2005) (Breese, Heckerman, & Kadie, 2013).

1.1.1. İşbirlikçi filtreleme teknikleri

Bu sistemler, bugüne kadar kullanılan en popüler ve başarılı filtreleme tekniklerinden biridir. İşbirlikçi filtreleme (İF) teknikleri, kullanıcının sistemdeki diğer kullanıcıların görüşlerine dayanarak seçimler yapmasına yardımcı olur. Bu teknikte benimsenen temel yaklaşım, geçmişte öğeler üzerinde hemfikir olan ya da aynı fikirde olmayan kullanıcıların gelecekte de aynı düşünceye sahip olmalarıdır. Dolayısıyla, İF teknikleri, bir kullanıcı için tahmin üretme aşamasında, sistemde kullanıcı ile benzer düşüncelere sahip kullanıcıları bulur ve daha sonra bu kullanıcıların tercihlerini birleştirerek öneri üretme sürecini gerçekleştirir.

Bu bilgi işleme tekniği, insanlar tarafından günlük yaşamda yaygın olarak kullanılır. İnsanlar sıklıkla başkalarından tavsiye isteyebilir ve doğal olarak en çok bildikleri ve güvendikleri kişilerin tavsiyelerine inanırlar. Örneğin, bir kişi belirli bir müzik CD' sini satın alma konusunda emin değilse, fikrine güvendiği kişilere sorabilir. Kişi, elindeki tüm bilgilere dayanarak, CD' yi satın alıp almayacağına dair güvenle karar verebilir. Bu nedenle, İF yaklaşımı insanların sözel olarak birbirlerine ifade ettiği bu süreci otomatik bir yapıya dönüştürmektedir.

Günümüzde artık İF algoritmaları internet uygulamalarında yaygın olarak kullanılmaktadır ve dahası önemli başarılar da elde etmektedir. Bu başarıya örnek olarak, web uygulamalarında, en büyük çevrimiçi kitap ve müzik mağazaları olan Amazon.com ve CDNow.com, kullanıcılar için kişiselleştirilmiş bilgi filtrelemesi sağlamak için bu teknikleri kullanmaktadırlar.

İF sistemleri hafıza tabanlı, model tabanlı ve hibrit olarak üçe ayrılır. Hafıza tabanlı sistemler, en yaygın tekniktir ve tahminleri/önerileri hesaplamak için sistem veri tabanındaki kullanıcı-ürün matrisleri üzerinde işlemleri yapar (Breese, Heckerman, & Kadie, 2013). Model tabanlı algoritmalar makine öğrenmesi ve veri madenciliği teknikleri oluşturulan modeli öğrenmek için önceki oy değerlerinden yararlanarak süreci ilerletir (Ricci, Rokach, & Shapira, 2015).

1.1.2. İçerik tabanlı filtreleme teknikleri

Bu sistemler, öneri üretmek için ürünlerin özelliklerin analizinin daha önemli olduğunu vurgular. Kullanıcıya web sayfaları, yayınlar, haberler gibi dokümanlar önerildiğinde, içerik tabanlı filtreleme tekniği en başarılı olanıdır. İçerik tabanlı filtreleme tekniğinde, kullanıcının geçmişte değerlendirdiği öğelerin içeriklerinden çıkarılan özellikler kullanılarak kullanıcı profillerine dayalı öneriler yapılır (Isinkaye, Folajimi, & Ojokoh, 2015).

1.2. Öneri Sistemleri Özellikleri

Öneri sistemlerinde, farklı uygulamaların farklı ihtiyaçların olması, sistem tasarımcısının hangi özelliğin ön planda olması gerektiğini belirlemesi gerekmektedir. Ancak özelliklerin bazılarında başarı sağlanırken diğer bir özellikte tersi durum yaşanabilir. En bariz örnek ise çeşitlilik (*diversity*) gibi özellikler geliştirilirken, doğruluktaki (*accuracy*) düşmedir. Dolayısıyla bu değiş-tokuş (*trade-off*) durumunu ve bunların genel performans üzerindeki etkilerini anlamak ve değerlendirmek önemlidir.

1.2.1. Yenilik

Öneri sisteminde üretilen önerilerin yeni olması, kullanıcının bilmediği ürünlerin sunulması demektir. Eğer yeni bir tavsiye üretilmesi gerekirse, kullanıcının önceden değerlendirmeye aldığı veya kullandığı ürünlerin basitçe filtrelenmesi ile gerçekleşir. Ancak buradaki problem çoğu kullanıcının geçmişte deneyimlediği her ürünü belirtmemesidir. Dolayısıyla, bir ürünün yeni olduğunun belirtilebilmesi için sadece filtreleme işleminin yapılması yetersiz kalacaktır.

Yapılan çalışmalarda (Adomavicius & Tuzhilin, 2005), yenilik (*novelty*) kelimesinin tanımına göre “Yeni ürün (Novel item)” aşağıda belirtilen üç özelliklere sahip olması gerektiği gösterilmiştir (Zhang L. , 2013):

1) Bilinmezlik (Unknown): Ürünün kullanıcı tarafından bilinmemesi

2) Memnun edici (Satisfactory): Ürünün kullanıcıyı memnun etmesi

3) Benzersizlik (Dissimilarity): Ürünün kullanıcının profilindeki ürünlerden farklı olmasıdır.

Sonuç olarak, kullanıcıya sunulan bir ürünün yeni olması, belirtilen üç kriterin sağlanması durumunda gerçekleşmektedir.

1.2.2. Çeşitlilik

Öneri sistemlerinde çeşitlilik (*diversity*) metriği genellikle benzerliğin karşıtı olarak tanımlanır. Bazı durumlarda, ürün aralığının keşfedilmesi daha uzun sürebileceğinden, benzer ürün önermek kullanıcı için o kadar yararlı olmayabilir (Ricci, Rokach, & Shapira, 2015). Örneğin, tatil paketlerinin yer aldığı bir tatil öneri sistemi olsun. Kullanıcı beş tatil öneri listesi istediğinde, lokasyonu aynı olan ancak sadece otel veya eğlence programı seçimleri değişen bir öneri listesi hazırlamak, beş farklı lokasyon önermek kadar faydalı olmayabilir. Dolayısıyla üretilen önerilerde çeşitliliğin olması, kullanıcı memnuniyetinin artmasını sağlayabilir (Bradley & Smyth, 2001).

1.2.3. Şans eseri bulma

Şans eseri bulma (*serendipity*), başarılı tavsiyelerin kullanıcı için ne kadar şaşırtıcı olduğunu ölçen bir metriktir. Örneğin, kullanıcı bir oyuncunun oynadığı birçok filmi olumlu olarak değerlendirdiyse, o oyuncunun yeni filmini kullanıcıya önermek yeni olabilir. Çünkü kullanıcı o oyuncunun diğer filmlerini biliyor fakat bu filmi bilmiyor. Ancak yeni olan bu film kullanıcı için şaşırtıcı değildir. Dolayısıyla kullanıcıya rastgele tavsiyeler üretilmesi oldukça şaşırtıcı olabilir (Kaminskas & Bridge, 2016).

Sonuç olarak yenilik, genel olarak şimdiki ve geçmiş deneyimler arasındaki fark olarak ifade edilirken; çeşitlilik bir deneyimin parçaları içindeki içsel farklılıklarla ilgilidir. Şans eseri bulma metriğinde ise, her tesadüfi ürün (serendipity item) yeni ürün olabilirken, her yeni ürün (novel item) tesadüfi ürün olarak etiketlenemez (Kotkov, Wang, & Veijalainen, 2016). Dolayısıyla tesadüfi ürün hem yeni hem de çeşitli olarak ifade edilebilir.

1.3. Çok Ölçütlü İF

Geleneksel öneri sistemlerinde, öneri üretilmesi ve benzerlik hesaplaması için ürünlere verilen değerlendirmelerin yer aldığı iki boyutlu kullanıcı ve ürün matrisini kullanılır. Bu sistemlerin çoğunda kullanıcı ürün değerlendirmesi tek bir kriter üzerinden gerçekleştirilir ve kullanıcı ürün matrisi arasındaki ilişki $R: Kullanıcı \times Ürün \rightarrow R_0$ olarak ifade edilir. Örneğin film öneri sisteminde, Şekil 1.1' de görüldüğü gibi kullanıcıların her biri sistemdeki filmleri tek bir ölçüt üzerinden değerlendirmeye almışlardır ve sistemde yer alan üç kullanıcıdan Alice, Fargo isimli film için tahmini bir değer üretilmesini istemektedir. Bu örnek için, en popüler öneri sistemi tekniklerinden biri olan İF teknikleri kullanılarak en yakın komşular belirlenir ve tahmini değer üretilir.

Alice ve John isimli kullanıcıların geçmişte deneyimledikleri dört film üzerinden benzer davranışlar sergilemesi, en yakın komşunun John olmasını sağlamıştır. Her ne kadar gerçek sistemlerde birden fazla komşuluğun daha yaygın kullanıldığı görülse de bu örnek için tek bir komşuluk seçilmiştir. Sonuç olarak Alice kullanıcısının Fargo isimli filminin tahmini değeri, John kullanıcısının bu filme verdiği oy değeri olacaktır.

	Wanted	WALL-E	Star Wars	Seven	Fargo
Alice	5	7	5	7	?
John	5	7	5	7	9
Mason	6	6	6	6	5
:	:	:	:	:	:

Şekil 1.1. *Tek ölçütlü film öneri sistemi*

Gerçek dünya uygulamalarının hızla büyümesi, öneri sistemlerinin çok ölçütlü değerlendirmeleri içerecek şekilde genişletilmesi, yeni nesil öneri sistemleri için önemli konulardan biri olmuştur. Dahası ürünlere verilen değerlendirmelerin çok ölçütlü olması, ürünlerin veya kullanıcıların karakteristik yapılarını daha iyi ortaya çıkarabilmektedir. Bu sistemlere örnek olarak, restoran değerlendirmeleri için kullanıcısına yemek, dekor ve hizmet olarak üç farklı ölçüt sunan Zagat's Guide, müşterilerinin elektronik tüketimleri için çok ölçütlü değerlendirmeler (örneğin, ekran boyutu, performans, pil ömrü ve maliyet) sağlayan Buy.com ve son olarak Yahoo! Movies platformu her bir kullanıcısının filmleri dört farklı ölçüte göre değerlendirmesine imkân vermiştir.

Bazı çok ölçütlü sistemlerde, kullanıcı ürün arasındaki ilişkiyi belirlemek için $R: Kullanıcı \times Ürün \rightarrow R_0 \times R_1 \times \ldots \times R_k$ fonksiyonu kullanılabilir. Burada R_0 kullanıcının genel değerlendirmesini, $i \in 1, 2, \ldots, k$ olmak üzere, R_k kullanıcının i. kritere vermiş olduğu oy değerini ifade etmektedir. Ancak bazı sistemler, genel derecelendirmeyi kullanmamayı ve yalnızca bireysel kriter değerlendirmelerine odaklanmayı da seçebilir.

Tek ölçütlü film öneri sisteminde Alice ve John kullanıcıları benzer davranışlar sergilediğini Şekil 1.1' e bakarak belirlemiştik. Bu sistemin Şekil 1.2' deki gibi çok ölçütlü olarak değiştirilmesi durumunda, kullanıcıların aynı genel değerlendirmelere sahip olmalarına rağmen, alt ölçütlerin değerlendirmelerinde farklı tercih gösterdiklerini görebiliriz. Bu çok ölçütlü sistem üzerine aynı İF teknikleri uygulandığında, bir önceki örneğin aksine, Alice kullanıcısının Mason kullanıcısına daha çok benzediği yapılan

hesaplamalarda görülmüştür. Dolayısıyla, Fargo isimli filmin tahmini değerinin 5 olacağı çıkarımı yapılabilir.

	Wanted	WALL-E	Star Wars	Seven	Fargo
Alice	$5_{2,2,8,8}$	$7_{5,5,9,9}$	$5_{2,2,8,8}$	$7_{5,5,9,9}$	$?_{?,?,?,?}$
John	$5_{8,8,2,2}$	$7_{9,9,5,5}$	$5_{8,8,2,2}$	$7_{9,9,5,5}$	$9_{8,8,10,10}$
Mason	$6_{3,3,9,9}$	$6_{4,4,8,8}$	$6_{3,3,9,9}$	$6_{4,4,8,8}$	$5_{2,2,8,8}$
:	:	:	:	:	:

Şekil 1.2. *Çok ölçütlü film öneri sistemi*

Bu örneklerde, tek bir genel değerlendirmenin, belirli bir ürünün farklı yönleri için kullanıcı tercihlerinin altında yatan heterojenliği gizleyebileceğini ve çok ölçütlü derecelendirmelerin her kullanıcının tercihlerini daha iyi anlamaya yardımcı olabileceğini ve bunun sonucunda kullanıcılara daha doğru öneriler sunmayı mümkün kılabileceğini ifade eder.

Çok ölçütlü işbirlikçi filtreleme (ÇÖİF) sistemleri benzerlik tabanlı ve birleştirme fonksiyonu tabanlı olmak üzere iki gruba ayrılır. Benzerlik tabanlı yaklaşımda, tek ölçütte olduğu gibi doğrudan korelasyon tabanlı ve kosinüs tabanlı metotlar kullanılamaz. Çok ölçütlü sistemde, tek derecelendirme yerine kullanıcının $k+1$ oy değeri yer almaktadır. Dolayısıyla ÇÖİF sistemlerinde kullanıcılar arasındaki benzerlikleri hesaplamak için iki farklı yaklaşım kullanılır.

İlk yaklaşım, iki kullanıcı arasındaki benzerlik ilk olarak her bir kriter açısından geleneksel benzerlik hesaplamaları yöntemleri kullanılarak hesaplanır. Daha sonra her bir kriter için elde edilen bu benzerlik değerleri birleştirilerek, son benzerlik değeri bulunur. Adomavicius ve arkadaşları (Adomavicius, Manouselis, & Kwon, 2011) *ortalama* ve *en kötü benzerlik* olarak iki farklı birleştirme yaklaşımı önermişlerdir. *Ortalama benzerlik* hesaplamasında, her bir kriterin benzerlik değerlerinin ortalaması alınırken; *en kötü benzerlik* senaryosunda ise, kriterler açısından en düşük benzerlik değerine sahip olan değer kullanılır.

- *Ortalama benzerlik hesaplaması*:

$$sim_o(u, u') = \frac{1}{k+1} \sum_{c=0}^{k} sim_c(u, u') \tag{1.1}$$

- *En kötü benzerlik hesaplaması*:

$$sim_o(u,u') = \min_{c=0,\dots,k} sim_c(u,u') \quad (1.2)$$

ÇÖİF sistemlerde, kullanıcılar arasındaki benzerlikleri hesaplamasında önerilen bir diğer yaklaşım, Öklid, Manhattan ve Minkowski gibi mesafe ölçütlerinin kullanılmasıdır.

Birleştirme tabanlı yaklaşımda, genel bir varsayım genel oy değerleri ile kriter bazında oy değerleri arasında ilişkisel bir yapı olduğudur. Tang ve McCalla (Tang & McCalla, 2009) her bir kriter için elde edilen benzerlikleri ağırlıklandırarak birleştirme işlemi gerçekleştirmiştir. Yaklaşımlarından her bir kriterin ağırlık değeri w_c ile ifade edilir ve bu değer kriterin öneri için ne kadar önemli ve faydalı olduğunu yansıtacak şekilde seçilir.

- *Toplama tabanlı benzerlik hesaplaması (Aggregate Similarity)*:

$$sim_o(u,u') = \sum_{c=0}^{k} w_c sim_c(u,u') \quad (1.3)$$

1.4. ÇÖİF Sistemlerinde Karşılaşılan Problemler

Geleneksel tek ölçütlü öneri sistemlerde kullanıcıya üretilen öneri listelerinin doğruluğunu ve kullanıcı memnuniyet durumunu arttırmak için literatürde oldukça geniş çalışmalar yer almaktadır. Bu çalışmalara örnek olarak, Pich vd. (Pichl, Zangerle, & Specht, 2014) kullanıcılara müzik tavsiye listeleri sunarak kullanıcı memnuniyetini arttırmayı amaçlamışlardır. Ren vd. (Ren, Li, & Zhou, 2012) çalışmalarında küçük veri seti üzerinde en iyi-*n* (top-*n*) öneri sisteminin doğruluğunu geliştirmişlerdir. Cremonesi vd. (Cremonesi, Koren, & Turrin, 2010) öneri listesi oluşturmak için ürün tabanlı algoritmaları kullanmışlardır. En iyi-n öneri sistemi için yeni bir tercih modelinin geliştirildiği ve öneri doğruluğunun arttırıldığı çalışma, Lee ve arkadaşları (Lee, Lee, Lee, Hwang, & Kim, 2016) tarafından yapılmıştır. Bu alanda yapılan diğer çalışmalarda, kullanıcı ürün matrisinin yapısal özelliklerine dayanarak (Kang, Peng, Yang, & Cheng, 2016), ürünler arasındaki ilişkisel yapının Derin Öğrenme' yle belirlenip ürün tabanlı algoritmalara (Xue, ve diğerleri, 2019) ve kullanıcı ürün topluluğu tespitine dayanarak kullanıcılara sunulan önerinin kalitesini iyileştirilmek istenmiştir.

Literatür incelemesinde, öneri sistemlerinin çoğu, kullanıcıların ürünler hakkındaki genel görüşlerini açıkça belirten tek bir kritere dayanmaktadır. Öte yandan, nispeten yeni olan ÇÖİF yöntemlerinde birkaç alt kriterin kullanıcıların tercihleri hakkında daha fazla ayrıntı içerdiğinden, daha doğru ve kişiselleştirilmiş tahminler üretebilirler. Örneğin, konaklamak için bir otele karar vermek için insanlar farklı düşüncelere sahip olabilir. Bir kişi yalnızca temizliğini düşünürken, diğer bir kişi otelin internet hızına odaklanırken, üçüncü bir kişi yalnızca iyi bir konum isteyebilir. Kullanıcılar hakkında bu tür çıkarımlarda bulanabilmek için kullanıcılara deneyimledikleri ürünlerle ilgili birden fazla özelliğini değerlendirmelerine izin verilmelidir. Bu nedenle, alt kriterlere verilen değerlendirmeler otelin genel skorunu etkileyecektir. Sonuç olarak, bir ürün veya hizmet için birden fazla kriteri değerlendirmek, müşteri ile ilgili kişisel çıkarımların yapılmasında ve müşteri memnuniyetinin arttırılmasında fayda sağlayabilir. Dolayısıyla bu geniş bilgi yelpazesini kullanılıp, kullanıcılara memnun kalabileceği bir tavsiye listelerinin oluşturulması öneri sistemleri için oldukça önemlidir. Ancak, literatürde ÇÖİF yöntemleri için sadece belirli bir ürün için üretilen tahminin doğruluğunu geliştirmek üzerine çalışmaların yer aldığı, kullanıcıya sevebileceği bir ürün listesi oluşturmak ile ilgili çalışmaların yer almadığı görülmektedir. Dolayısıyla, ÇÖİF üzerinde çalışan bir en iyi-n algoritmalarının literatüre tanıtılması gerekmektedir. Çalışmalardaki bu eksiklik, kullanıcı memnuniyetinin azalmasına ve hatta sisteme olan güvenin kaybolmasına neden olmaktadır. Dolayısıyla üretilen tavsiyelerde kullanıcı memnuniyetinin artması, çok yönlü kullanıcı bilgilerinden kişisel karakteristik analizinin iyi bir şekilde yapılması ile sağlanacaktır.

ÇÖİF yöntemlerini kullanan şirketlerin, müşterilerini hayal kırıklığına uğratmamak için kullanıcılarına yüksek kaliteli öneriler (tavsiye listeleri) sunmaları gerekmektedir. Kullanıcılara sunulan öneri listesine yanlış ürünleri dahil etmek, müşterilerin başka e-ticaret sitelerine yönelmesine neden olabilir. Bu durum firmaların müşteri kaybetmesine ve rekabet ortamında alt sıralara yerleşmesine neden olabilmektedir. Böyle bir senaryonun gerçekleşmesini isteyen kötü niyetli kişiler veya işletmeler, sunulan bu hizmetleri İF yöntemleri doğrultusunda kendi lehlerine yönlendirmeye çalışabilirler. Literatür incelemesinde, Kaur ve Goel (Kaur & Goel, 2016) sisteme yapılan saldırı sonucunda oy değerlerinde nasıl bir manipülasyon olduğunu ölçmüşlerdir. Kumari ve Bedi (Kumari & Bedi, 2017) tarafından yapılan bir başka çalışmada, veri seti olarak kullanılan haber öğelerine uygulanan atak sonuçlarını incelemişlerdir. Gerçek

kullanıcılarla benzer özelliklere sahip kötü niyetli sahte profiller üreten ve güçlü saldırı yeteneklerine sahip bir saldırıda, Chen vd. (Chen, ve diğerleri, 2018) tarafından yapılmıştır. Deldjoo ve arkadaşları (Deldjoo, Di Noia, Di Sciascio, & Merra, 2021) saldırganın öneri sistemi üzerine nasıl bir etkisini olduğunun analizini yapmışlardır. Kumar ve Bedi (Kumari & Bedi, 2017) çeşitli profil enjeksiyon saldırılarını tanımlamışlar ve bu saldırıların performansını ölçmek için saldırı senaryolarını genişletmişlerdir. Ancak yapılan kapsamlı literatür incelemesinde ÇÖİF en iyi-n öneri sistemlerinin, şilin saldırılarına karşı ne kadar sağlam olduğu ve bu sistemlerde şilin saldırısının nasıl gerçekleştirileceği gibi durumların araştırmaya dahil edilmediği görülmüştür. Dahası bu sistemlere gerçekleştirilen atakların tespiti de başka bir eksik olarak karşımıza çıkmaktadır. Bu sebeple, ÇÖİF sistemlerinde, belirtilen ihtiyaçların ve problemlerin giderilmesi için yeni yaklaşımlara ihtiyaç duyulmaktadır.

1.5. Tezin Amacı ve Katkıları

ÇÖİF yöntemleri nispeten yenidir ve tek ölçütlü sistemlere göre bu sistemler, kullanıcıların tercihleri hakkında daha fazla ayrıntı içerdiğinden, daha doğru ve kişiselleştirilmiş tahminler/tavsiyeler üretebilirler. Dahası son yıllarda yapılan çalışmalar incelendiğinde, kullanıcılara doğru tahminler sunmanın yanı sıra üretilen önerilerin çeşitliliği, yeniliği ve beklenmedikliğin kullanıcı memnuniyeti açısından oldukça önemli olduğu sonucuna varılabilir. Ancak, bu çok ölçütlü sistemlerde başarılı tahmini değerler üretilirken, kullanıcıya öneri listesi sunulmasının eksik olması bir problem olarak karşımıza çıkmaktadır.

Bu sistemleri kullanarak sunulan öneri listelerinin, kötü niyetli kullanıcı veya firmalar tarafından manipüle edilmesi mümkündür. Dolayısıyla, kullanıcıya başarılı tavsiyeler sunulurken bu sistemlerin, bu tür saldırılara karşı ne derece gürbüz olduğunun analizinin yapılması gerekmektedir. Ayrıca, başarılı tavsiyeler üreten bu sistemlerin güvenli bir şekilde kullanılabilmesi için, sistem içerisinde yer alabilecek sahte kullanıcı profillerin tespit edilmesi ihtiyacı ortaya çıkmaktadır.

Dolayısıyla bu doktora tezi kapsamında, çalışmalar üç farklı yöntem üç ana başlıkta incelenmiştir. Çalışmanın ilk aşamasında, ÇÖİF yöntemleri kullanılarak kullanıcı tutum ve eğilimlerinin dikkate alındığı kullanıcıyı memnun edecek bir öneri listesi hazırlanması amaçlanmıştır. İkinci bölümde, ilk bölümde oluşturulan sistemin ataklara karşı gürbüzlük

analizi yapılmıştır. Son bölümde ise bu çok ölçütlü sistemleri manipüle eden kötü niyetli kullanıcıların atak tespiti gerçekleştirilmiştir.

Tez kapsamında yapılan çalışmaların her biri literatür için ilk özelliğini korumaktadır ve bu çalışmaların her birinin literatüre katkıları aşağıda listelenmiştir:

- Çok ölçütlü öneri sistemlerinde, kullanıcıların karakteristik yapısının ve değerlendirme esnasındaki tutumlarının/eğilimlerinin dikkate alındığı ayrıca ürünler arasındaki ilişkisel yapının da farklı bir boyutta incelendiği yeni bir en iyi-n öneri sistemi geliştirilmiştir. Dahası önerilen bu yöntemleri, karşılaştırabilmek için belirlenen kıyaslama algoritmasının bu sisteme uyarlanması yapılmıştır. Yapılan deneysel çalışmalarla elde edilen sonuçlar incelendiğinde, kullanıcıya ürün listesi hazırlanması aşamasında önerilen bu yaklaşımların kullanılması, kullanıcı memnuniyetini arttırdığı ve dahası liste içerisinde yer alan ürünlerde çeşitlilik ve yenilik sağladığı da gözlenmiştir.
- Çok ölçütlü sistemlerde kullanıcı memnuniyetini arttıran ürünler sunulsa da bu sistemlerin kötü niyetli kullanıcılar veya firmalar tarafından manipüle edilmesi de olası bir durumdur. Dolayısıyla önerilen yaklaşımların ataklara karşı ne derece gürbüz olduğunun analizinin yapılması gerekmektedir. Ancak önerilen modellerin ürün tabanlı olması ve literatürde yer alan çoğu atak modelinin bu sistemlerde gürbüz olması nedeniyle bu sistemler de başarılı olan bir atak modeli kullanılarak yeni bir şilin atak stratejisi geliştirilmiştir. Atak profillerin oluşturulması ve atak süreci ile ilgili adımların her biri, çok ölçütlü sisteme uyarlanarak sürdürülmüş ve yeni yöntemler önerilmiştir. Ayrıca sistemin performansını ölçmek için literatürde var olan metriklere ek olarak, manipüle edilen ürünlerin sırasını da dikkate alan yeni bir metrik geliştirilmiştir. Deneysel sonuçlarda önerilen yaklaşımların bu ataklara karşı savunmasız olduğunu göstermiştir.
- Çok ölçütlü sistemler üzerine gerçekleştirilen atakta, sistemin savunmasız olduğu ancak bu sistemleri bu yaklaşımlarla kullanmak isteyen kullanıcıların doğru güvenilir tavsiyeler alması gerekmektedir. Dolayısıyla bu durum, bu sistemler üzerinde bir atak tespit işleminin yapılmasını zorunlu kılmıştır. Çalışmada sınıflandırma tabanlı yeni bir şilin atak tespit

yöntemi önerilmiş ve bu yaklaşım için gerekli olan özellik çıkarımlarında ürün popülerliği ve kullanıcı karakteristik yapısı dikkate alınarak yeni metotlar sunulmuştur. Deneysel sonuçlarda önerilen bu yeni yaklaşımın atak tespitinde oldukça başarılı olduğu görülmüş ve bu sistemi kullanmak isteyen kullanıcıların bu tespit işleminin ardından sistemleri güvenle kullanabilecekleri çıkarımı yapılmıştır.

1.6. Tez Organizasyonu

Bu tez, organizasyon açısından toplam yedi bölümden oluşmaktadır. İkinci bölümde önerilen çok ölçütlü en iyi-n öneri sistemleri üretilmesinde kullanılan metotlarla ilgili genel bir bilgilendirme yer almaktadır. Üçüncü bölümde şilin ataklardan ve bu atakların tespitinden bahsedilmiştir. Tez kapsamında önerilen çok ölçütlü en iyi-n öneri sistemleri yöntemlerine Bölüm 4' te yer verilirken, önerilen bu yöntemler üzerine gerçekleştirilen şilin ataklar Bölüm 5' te sunulmuştur. Çok ölçütlü en iyi-n öneri sistemi üzerine gerçekleştirilen saldırıların tespiti Bölüm 6' da ifade edilmiştir. Son olarak, tez kapsamında yapılan çalışmalarla elde edilen kazanımlara ve sonrasında gerçekleştirilebilecek çalışmalara son bölümde yer verilmiştir.

2. GENEL BİLGİLER

Bu bölümde tez kapsamında yapılan çalışmalarda kullanılan yöntemlerle ilgili bir bilgilendirme yapılmıştır. İlk adımda kullanıcıya hazırlanan en iyi-n öneri sistemlerinde, kullanıcıların karakteristik yapısının belirlenmesinde kullanılan *sezgisel bulanık mantık* yönteminden bahsedilmiştir. Sonrasında, bu sistemlerde kullanılan yeni komşuluk seçme yöntemi olan *Entropi tabanlı komşuluk seçme* metodu ve derecelendirme değerlerinin nitel sıralamasının dikkate alındığı *tercih model* açıklanmıştır. Çok ölçütlü sistem üzerinde gerçekleştirilen çalışmalarda kriterleri ağırlıklandırmak için kullanılan *çoklu lineer regresyon* metodu ve ürünler arası ilişkisel yapının belirlendiği *birliktelik kuralı* açıklamalarına yer verilmiştir. Sonrasında sınıflandırma yöntemi olarak kullanılan *k en yakın komşular algoritması* tanıtılmıştır. Son olarak, tez kapsamında geliştirilen yaklaşımların performans analizinin gerçekleştirildiği *veri seti* ile ilgili bilgilendirmeler yapılmıştır.

2.1. Sezgisel Bulanık Mantık

Bulanık küme teorisinde bir elemanın üyelik durumu 0 ile 1 arasında değişen değerlerle ifade edilebilir. Ancak gerçek hayatta, üye olmama derecesinin üyelik derecesinin 1 eksiği şeklinde ifade edilmesi her zaman doğru olmayabilir. Çünkü bu aşama da bir tereddüt durumunun yaşanması da mümkündür. Dolayısıyla bu durumun ifade edildiği en popüler yöntemlerden biri Sezgisel Bulanık Mantık (Intuitionistic Fuzzy Set, IFS)' tır (Atanassov, 1999). Bu yöntemin tanımlaması aşağıdaki gibidir:

$$A = \langle\, x, \mu_A(x), v_A(x) \mid x \epsilon I \,\rangle \tag{2.1}$$

burada

$$\mu_A : I \rightarrow [0,1] \; ve \; v_A : I \rightarrow [0,1] \tag{2.2}$$

μ_A ve v_A, x elementinin üye olma ve olmama durumudur. Tereddüt indeksi (hesitation index), x elementinin A kümesine ait olmasındaki tereddüt seviyesini işaret eder. Bu kavram aşağıdaki gibi π_A ile ifade edilir.

$$\pi_A(x) = 1 - \mu_A(x) - v_A(x) \tag{2.3}$$

Eğer π_A değeri çok küçük ise tereddüt durumunun az olduğu ve x elementi hakkında bilgini oldukça kesin olduğu söylenebilir. Aksi durumda ise çok fazla tereddüt ve bu bilgi de bir belirsizlik olduğunu ifade edilebilir. Eğer π_A değeri sıfıra eşitse bu durumda klasik bulanık mantık yapısına dönüşür.

2.2. Entropi Tabanlı Komşuluk Seçme

Kaleli (Kaleli, 2014) tarafından önerilen bu modelde, tahmini oy değerlerin bulunması sırasında bir işletmenin derecelendirme vektöründen elde edilen tüm mevcut bilgilerin kullanılması hayati önem taşır. Kullanıcıların derecelendirme alanlarıyla ilgili farklı düşünceleri nedeniyle, kullanıcı/ürün vektörlerinin kullandıkları değerlendirme skorları bir *belirsizlik derecesi* (*Degree of Uncertainity, DU*) durumu ortaya çıkarabilir (Kaleli, 2014). Ayrıca, bir derecelendirme alanının tüm üyeleri, ifade edilen tercihlerde yakın olasılıklarla kullanıldığından, varlıkların *DU* değerleri de buna göre artar. Araştırmacı, *DU* değerlerini Shannon Entropi' si ile ölçmenin mümkün olduğunu ve aktif kullanıcı veya ürünlerin en yakın komşuluklarını oluşturmak için benzerliklerle birlikte bu *DU* değerlerinin kullanılması gerektiğini göstermişlerdir. Dolayısıyla Entropi' nin dahil edildiği yeni komşuluk seçme süreci aşağıdaki gibi özetlenmiştir:

İlk olarak sistemde yer alan kullanıcı değerlendirmelerinin nasıl bir dağılım gösterdiği bulunmuştur.

$$P_{i,r} = \frac{i\ vektöründeki\ r\ oy\ değerlerinin\ sayısı}{i\ vektöründeki\ toplam\ oy\ değerleri\ sayısı} \tag{2.4}$$

Burada $P_{i,r}$, her bir oy değerinin meydana gelme olasılığıdır.

Daha sonra, kullanıcıların bu oyları nasıl kullandıkları, nasıl bir dağılıma sahip olduklarını belirleme için bir Entropi hesaplaması yapılmıştır. Bu sayede her bir kullanıcı veya ürünün *DU* bulunmuştur.

$$DU_i = \frac{\sum_{i \in DO} P_{i,r} \log_2(P_{i,r})}{i\ vektöründeki\ toplam\ oy\ değerleri\ sayısı} \tag{2.5}$$

DO, derecelendirme ölçeğini, *i* kullanıcı ya da ürünü ifade eder.

Öneri sisteminde, var olan bütün bilgilerin kullanılması gerekmektedir. Dolayısıyla kullanıcı veya ürünler arasındaki benzerlik değerleri ile bu bulunan *DU* değerlerinin optimal bir şekilde birleştirilmesi gerekmektedir. Bu amaç doğrultusunda, bu model için klasik optimizasyon problemlerinden biri olan 0-1 sırt çantası algoritması kullanılmıştır.

$$\sum_{i=1}^{N} P_i X_i \tag{2.6}$$

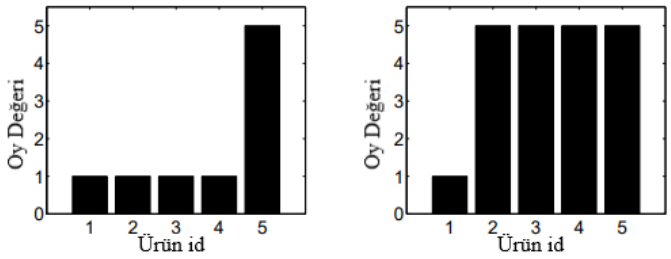
$$\sum_{i=1}^{N} W_i X_i \leq C \qquad X_i = 0 \; veya \; 1 \tag{2.7}$$

Bu optimizasyon probleminde, Denklem 2.6' da elde edilecek değerin oldukça büyük olmasıdır. Dolayısıyla önerilen yaklaşımda sabit bir *C* katsayısı bulmak için bu farkların ortalaması en yakın komşuluk sayısı ile çarpılır.

2.3. Tercih Model

Lee vd. (Lee, Lee, Lee, Hwang, & Kim, 2016) tarafından önerilen bu modelde, kullanıcıların davranış kalıpları arasındaki benzerliği kullanan İF, her kullanıcı için en çekici ilk *n* öğeleri önerir. Bu süreçte, mevcut İF algoritmaları, derecelendirme tahmini açısından en yüksek derecelendirmeye sahip ilk *n* öğeleri döndürür ve ilk *n* önerisi için öğelerin nitel sırasını yok sayar. Son zamanlarda yapılan bazı çalışmalar ilk *n* tavsiyesini optimize etmek için İF algoritmaları geliştirmiş olsa da yine de nitel kullanıcı tercihlerini, yani açık derecelendirmelere sahip öğeler arasındaki göreceli tercihleri belirlemede sınırlamalara sahiptirler.

Araştırmacılar, *tercih model* (*Preference model*) için belirledikleri senaryoda, iki kullanıcı *u* ve *v*' nin aynı beş üründen oluşan $\{i_1, \ldots, i_5\}$ kümeyi değerlendirmeye aldığı varsayılsın ve bu değerlendirmeler Şekil 2.1' de gösterilmektedir. İki kullanıcı i_5 ürününe aynı 5 puanını vermiş olsa bile, puan dağılımlarına bağlı olarak puanın anlamı farklı yorumlanabilir. Örneğin hem *u* hem de *v*' nin i_5 ürününe 5 puan verdiğini Şekil 2.1' de görmekteyiz. Ancak *v* kullanıcısının çoğu ürüne 5 puanı vermesi bu kullanıcının 5 puan verme eğiliminde olduğunu göstermektedir. Dahası *v* kullanıcısının i_5 ürünü ile ilgili ne kadar memnun olduğu veya ne kadar tatmin edici bir ürün olduğunu belirlemek zordur. Aksine, *u* kullanıcısı nadiren 5 puan verdiği için, *u* kullanıcısının bu i_5 ürününden gerçekten memnun olduğu ile ilgili bir çıkarım yapmak mantıklı olacaktır. Dolayısıyla, *u* kullanıcısının i_5 ürününü, *v* kullanıcısına göre daha fazla tercih ettiği bu sonuçlardan çıkartılabilir.



(a)Kullanıcı *u* (b) Kullanıcı *v*

Şekil 2.1. *u ve v kullanıcılarının oy dağılımları*

Borda count metodundan ilham alınan bu tercih modelinde, ürünler artan derecelendirme sırasına göre sıralandığında, hedef kullanıcı *u* için *i* ürünün tercih puanın (Preference score) $pref(r_{u,i})$ hesaplanmasında, i ürününde daha düşük veya ona eşit sıralanan ürünlerin sayısı toplanarak hesaplanır. Spesifik olarak, tercih puanını hesaplamak için iki durum vardır.

1. Kullanıcı derecelendirmesi verildiğinde $r_{u,i}$, i' den daha düşük sıralanan ürünlerin sayısı sayılır. $pref_{>}(r_{u,i})$, tercih puanıdır ki burada $r_{u,i}$ düşük sıralamaya sahip ürün sayısıdır, $pref_{>}(r_{u,i}) = |\{r_{u,i} \in {R_u}^{+} \mid r_{u,i} > r_{u,j}\}|$.
2. $pref_{=}(r_{u,i})$, tercih puanıdır ki burada $r_{u,i}$ eşit sıralamaya sahip ürün sayısıdır, $pref_{=}(r_{u,i}) = |\{r_{u,i} \in {R_u}^{+} \mid r_{u,i} = r_{u,j}\}|$.

İki durumu birleştirerek, $r_{u,i}$'nin tercih puanı, lineer kombinasyonla ağırlıklandırarak hesaplanır. Burada ${R_u}^{+}$ normalizasyon parametresi olarak kullanılır.

$$pref(r_{u,i}) = \alpha \frac{pref_{>}(r_{u,i})}{|{R_u}^{+}|} + \beta \frac{pref_{=}(r_{u,i})}{|{R_u}^{+}|} \tag{2.8}$$

Burada α ve β, sırasıyla $pref_{>}(r_{u,i})$ ve $pref_{=}(r_{u,i})$ için ağırlık parametreleridir. Literatürde yer alan çalışmalara göre bu değerler sırasıyla 1 ve 0,5 olarak belirlenir.

Bu denklem kullanarak, farklı oy dağılımlarına bağlı olarak kullanıcı oylarının nitel kullanıcı tercih skorlarına dönüştürülür. Tercih skorları hesaplandıktan sonra kullanıcı oy değerleri karşılık gelen tercih puanlarına dönüştürülür ve komşuluk sürecinin ardından tahmini değer üretme aşamasından bu skor kullanılır.

2.4. Çoklu Lineer Regresyon

Çoklu doğrusal regresyon analizi (multiple linear regression analysis), bir değişkeni etkileyen iki ve daha fazla bağımsız değişken arasındaki neden-sonuç ilişkilerini doğrusal

bir modelle açıklayan ve bu bağımsız değişkenlerin etki düzeylerini belirlemek için kullanılan bir yöntemdir. Çoklu doğrusal regresyon aşağıdaki gibi ifade edilebilir;

- Bağımsız değişkenlerin bağımlı değişken üzerindeki etki değerlerini bulma
- Bağımlı değişkeni etkilediği belirlenen değişkenler yardımıyla bağımlı değişkenin değerini tahmin edebilmek

Bu yöntemin matematiksel ifadesi aşağıda verilmiştir:

$$y_i = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n + \varepsilon_i \tag{2.9}$$

Burada y_i bağımlı değişken, X_n bağımsız değişkenler, ε_i hata değeri ve β_n katsayı değeridir.

2.5. Birliktelik Kuralı

Birliktelik kuralları analizi problemi, birbiriyle ilişkili özellikleri ortaya çıkarmak ve aralarındaki büyüklük ilişkisini belirlemek için kullanılan bir kavramdır. Bir birliktelik kuralı $X \rightarrow Y$ biçiminde gösterilir. Bu ifade, işlemde '*X* ürünün ortaya çıkma olasılığı *Y* ürünün ortaya çıkmasına bağlıdır.' anlamına gelir.

Birliktelik kuralının üç önemli unsuru vardır: *destek* (*support*), *güven* (*confidence*) ve *kaldırma* (*lift*).

Bir kuralın desteği, *X* ve *Y* içeren işlemlerin sayısıdır ve Denklem 2.10' da gösterilmiştir.

$$destek = \frac{frq(X,Y)}{N} \tag{2.10}$$

Bir kuralın güveni, *X* ve *Y* içeren işlemlerin sayısının, *X* içeren işlemlerin sayısına bölümüdür.

$$g\ddot{u}ven = \frac{frq(X,Y)}{frq(X)} \tag{2.11}$$

lift parametresi, *X* ve *Y* meydana gelme olasılığının sadece *Y* olma olasılığından kaç kat daha fazla olduğunu söyler. Bu değer 0 ve ∞ arasında değişmektedir.

$$lift(X,Y) = \frac{destek(X,Y)}{destek(X)destek(Y)} \tag{2.12}$$

Bu birliktelik kuralı $X \rightarrow Y$ için;

- *lift* değeri 1' e eşitse, bu, *X* ve *Y*' nin bağımsız olduğu anlamına gelir, başka bir deyişle, *X*' in varlığının, *Y*' nin varlığı üzerinde neredeyse hiçbir etkisi yoktur.
- *lift* değeri 1' den büyükse, bu, *X* ve *Y* arasında pozitif korelasyon olduğu anlamına gelir, *X*' in varlığı, *Y*' nin varlığı üzerinde olumlu bir etkiye sahiptir.
- *lift* değeri 1' den küçükse, bu, *X* ve *Y*' nin negatif korelasyona sahip olduğu anlamına gelir, *X*' in varlığının, *Y*' nin varlığı üzerinde olumsuz bir etkisi vardır.

2.6. k En Yakın Sınıflandırma

k-en yakın komşular (*k*NN) algoritması hem sınıflandırma hem de regresyon tahmin problemleri için kullanılabilen bir tür denetimli makine öğrenmesi algoritmasıdır. Bu algoritma basit öneri sistemlerinde, görüntü tanıma teknolojisinde ve karar verme modellerinde kullanıldığı gibi günümüzde birçok problem bir şekilde sınıflandırma problemi olarak tasarlanıp çözülebilmektedir. *k*NN, en basit anlamı ile içerisinde tahmin edilecek değerin bağımsız değişkenlerinin oluşturduğu vektörün en yakın komşularının hangi sınıfta yoğun olduğu bilgisi üzerinden sınıfını tahmin etmeye dayanmaktadır. Dolayısıyla bu algoritmayı aşağıdaki özellikler en iyi şekilde tanımlamaktadır:

Tembel öğrenme algoritması, *k*NN, özel bir eğitim aşamasına sahip olmadığı ve sınıflandırma sırasında tüm verileri eğitim olarak kullandığı için tembel bir öğrenme algoritmasıdır.

Parametrik olmayan öğrenme algoritması, *k*NN, temel alınan veriler hakkında hiçbir şey varsaymadığı için parametrik olmayan bir öğrenme algoritmasıdır.

Bu algoritmanın çalışma adımları aşağıda yer almaktadır:

1. Eğitim ve test veri setini yükle,
2. *k* değerini seç, bu değer pozitif ve bir tamsayı olsun,
3. $\boldsymbol{d(x, x_i)}$ hesapla, $\boldsymbol{i = 1, 2, \dots, n}$; *d* noktalar arasındaki Öklid mesafesini gösterir,
4. Hesaplanan *n* Öklid mesafelerini azalan bir sıraya göre düzenle,
5. Bu sıralanmış listeden ilk *k* mesafeyi alın,
6. Bu *k*-mesafelerine karşılık gelen *k*-noktalarını bulun,

7. Eğer $\boldsymbol{k_i > k_j \ \forall \ i \neq j}$ ise *x*, *i* sınıfına aittir, burada $\boldsymbol{k_i}$, *k*-noktaları arasındaki *i*. sınıfa ait noktaların sayısını gösterir, yani $k \geq 0$.

2.7. Veri Seti

Yapılan çalışmalarda öneri sistemleri alanında çok iyi bilinen çok ölçütlü bir veri seti olan Yahoo! Movies kullanılmaktadır. Bu veri seti 6078 kullanıcı ve 976 filmden oluşmaktadır (Lakiotaki, Matsatsinis, & Tsoukias, 2011). Genel derecelendirmeye ek olarak, kullanıcılar senaryo (story), oyunculuk (acting), yönetmenlik (direction) ve görsel efekt (visual) olmak üzere dört alt kriterde çok ölçütlü derecelendirmeler yer almaktadır. Orijinal veri setinde, eldeki tüm oy değerleri 13-fold derecelendirme değerlerinden oluşur, F en kötü değerlendirme derecesini ve A+ en çok tercih edilen değeri belirtir [A+, A, A-, B+, B, B-, C+, C, C–, D+, D, D-, F].

Çalışmada verilen bu derecelendirme aralığı 1-5 ölçeğine dönüştürülmüştür (Turk & Bilge, 2019). Bu dönüşüm, üçerli gruplar şeklinde yapılarak indirgenmiştir. Örneğin, ölçekte [13, 12, 11], [10, 9, 8] sırasıyla 5 ve 4 yıldızlarına karşılık gelirken, üçlü grup dışında kalan [1], 1 yıldızı ifade etmektedir.

Tablo 2.1. *Veri seti ile ilgili bilgiler*

	Kullanıcı Sayısı	**Ürün Sayısı**
YM5	5,678	3,079
YM10	1,827	1,471
YM20	429	491

YM veri setinin YM5, YM10 ve YM20 olarak adlandırılan üç farklı alt setleri oluşturulmuştur. Bu veri setlerinde yer alan rakamsal değerler, veri setindeki kullanıcı ve ürünlerin minimum derecelendirme sayılarını ifade etmektedir. Örneğin YM20 veri setinde, her kullanıcı en az 20 ürünü derecelendirmiştir, her üründe en az 20 kullanıcı tarafından değerlendirilmeye alınmıştır. Benzer durum diğer iki veri seti içinde geçerlidir.

Tez kapsamında yapılan deneylerde, ilk çalışmada üç alt veri seti de kullanılırken, diğer iki çalışmada veri setinin çok seyrek olmasından kaynaklı sadece iki alt veri seti (YM10 ve YM20) kullanılmıştır.

2.8. Pareto Prensibi

‘80/20 kuralı’, ‘asgari çaba ilkesi’ veya ‘dengesizlik ilkesi’ olarak da bilinen Pareto prensibi, İtalyan ekonomist Vilfredo Pareto tarafından bulunmuştur (Sanders, 1987).

19.yy ’daki İngiltere ekonomisinin servet ve gelir dağılımını incelerken, servetin %80’ inin, nüfusun %20’ sine ait olduğunu fark etmiştir. Kendi ülkesi ve diğer ülkelerle de bu verilerin incelemesini yaptığında benzer sonuçlar olduğunu görmüştür. Ancak o dönemde bu durumu açıklamak yetersiz olmuştur. Daha sonraki zamanlarda George K. Zipf ve Joseph Moses Juran tarafından bu kuram önem kazanmıştır ve Vilfredo Pareto’ ya ithafen de Pareto prensibi olarak literatüre geçmiştir.

Pareto prensibi, ekonomi ile ilgili olarak ortaya atılsa da günlük yaşamın büyük bir bölümüyle de uyum sağladığı görülmüştür. Bu kurama göre hayattaki sonuçların %80’ i, o işi yapmak için sarf edilen çabanın %20’ sinden kaynaklanır. Yani bir işin, en önemli ya da daha öncelikli olan %20’ lik kısmını çok iyi yaparsanız ve işinizin çoğunu halletmiş olursunuz. Pareto prensibi size bilinçli bir seçim yaparak, daha az çaba ile daha çok işi, verimli bir şekilde yapma fırsatı verir. Bundan dolayı da şirket politikalarının hazırlanmasından, kişisel gelişime kadar birçok alanda kullanılmaktadır. Örneğin;

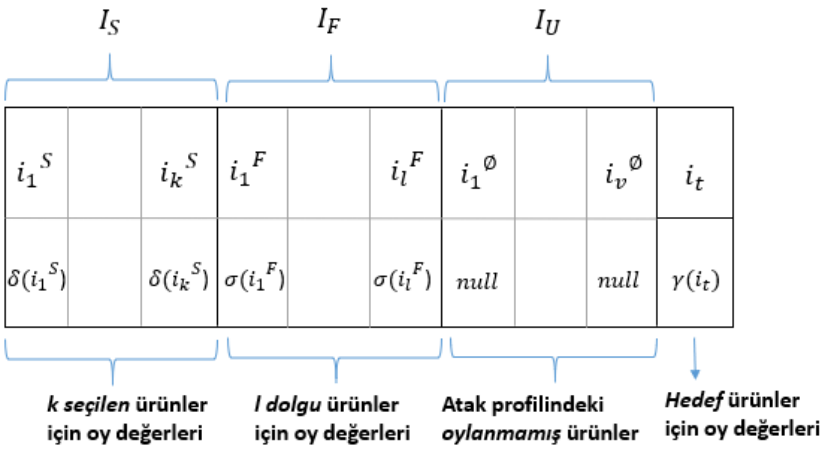
- Bağışların %80’ i toplumun %20’ si tarafından yapılır.
- Trafik kazalarının %80’ i sürücülerin %20’ sinden kaynaklanır.
- Satışın %80’ ini satış yapanların %20’ si tarafından gerçekleştirilir.

Bu oran kesin olarak 80/20 ya da toplamda 100’ e tamamlanmak zorunda değildir. 70/30, 85/15, 80/16’ da olabilir. Bu prensipteki asıl olay neden ve sonuçlar arasındaki dengesiz orantıdır.

3. ŞİLİN ATAKLAR

Kötü niyetli kullanıcılar/şirketler, üretilen tahminleri etkilemek veya hedeflerinde sistemin güvenilirliğini azaltmak için öneri sistemlerini manipüle etmek isteyebilir. Bu amaçla, şilin saldırıları olarak da adlandırılan sahte profilleri sisteme enjekte etmeye çalışırlar ve ne yazık ki, öneri sistemlerinin bu tür saldırılara karşı oldukça savunmasız olduğu bilinen bir olgudur (Lam & Riedl, 2004). Mekanizmalarına bağlı olarak, şilin saldırıları genellikle iki farklı kategoriye ayrılır; *itme* (*push*) ve *çekme* (*nuke*). Daha spesifik olarak, *itme* saldırıları, hedef öğelerin kullanılan derecelendirme ölçeğinde mümkün olan en yüksek oy değeriyle derecelendirildiği atak profilleri ekleyerek, hedef öğeler olarak adlandırılan belirli öğelerin popülerliğini artırmayı amaçlar (Mobasher, Burke, Bhaumik, & Williams, 2007). Öte yandan, *çekme* olanlar genellikle en düşük derecelendirme değerini vererek hedef öğelerin popülaritesini azaltmaya odaklanır.

Son yıllarda, öneri sistemleri ilgili literatürde, İF olanlar da dahil olmak üzere, öneri algoritmalarının popüler öğelere karşı güçlü bir şekilde önyargılı olduğunu doğrulanmıştır. Bu önyargıda katalogda çok fazla gişe rekorları kıran öğeyi çok sık önerirken, popüler olmayan pek çok öğeye yeterli şans vermemesidir ki bu ürünler bireyleri memnun edebilecek ürünler olabilmektedir (Jannach, Lerche, Kamehkhosh, & Jugovac, 2015). Bu nedenle, algoritmaların bu tür popülerlik yanlılığından yararlanarak, itme olanları kullanan saldırganlar, sağlanan önerilerde hedef öğelerinin görünürlüğünü artırabilir ve böylece daha yüksek satış oranları elde edebilir. Öte yandan, çekme sağlayıcılarını kullanan manipülatif kişiler, ürünlerinin ortalama derecelendirmesine göre rakiplerine etkili bir şekilde zarar verebilir. Bu nedenle, genellikle şilin saldırılarının öneri listelerinde yer alan öğelerin adil olmayan bir şekilde temsil edilmesine yol açabileceği ve çeşitlilik ve yenilik gibi doğruluğun ötesinde perspektifler açısından öneri kalitesini olumsuz yönde etkileyebileceği varsayılmaktadır (Kaminskas & Bridge, 2016).



Şekil 3.1. *Atak profilin genel yapısı*

Şekil 3.1' de açıklandığı gibi, tipik olarak dört ana bölümden oluşan bir şilin saldırı profilinin genel biçimi sunulmuştur. Hedef ürün (Target Items, I_T) saldırganın niyetini belirtirken, derecelendirme değeri saldırının amacına göre maksimum veya minimum oy değerini alır. Seçilen ürün seti (Selected Items, I_S) saldırgan tarafından belirlenen belirli özelliklere sahip ürünlerin koleksiyonudur ve dolgu ürün seti (Filler Items, I_F) saldırı tespitini önler. Son olarak, kalan oylanmamış ürünler derecelendirilmemiş öğeler (Unrated Items, I_U) kümesini oluşturur. Şekil 3.1' deki δ, σ ve γ sembollerinin I_S, I_F ve I_T kümelerinin içindeki ürünlere, derecelendirme değerlerinin nasıl atanması gerektiğini gösteren fonksiyonlardır.

3.1. Şilin Atak Modelleri

Saldırganın bakış açısına göre, bir saldırı gerçekleştirmek için saldırı girişiminde bulunulan öneri sistemi hakkında algoritma, kullanıcılar, ürünler ve derecelendirmeler ile ilgili bir miktar veriye ihtiyaç duyabilir (Lam & Riedl, 2004). Bu bilgi miktarına göre saldırılar ikiye ayrılmaktadır: *yüksek bilgili saldırı* (*high-knowledge attack*), sistemin veri tabanındaki derecelendirme dağılımı hakkında ayrıntılı bir bilgi gerektirirken, *düşük bilgili saldırı* (*low-knowledge attack*), sistemden bağımsız bilgi gerektirir (Mobasher, Burke, Bhaumik, & Sandvig, 2007). Birçok araştırmaya konu olan en popüler ve iyi bilinen şilin atak modelleri kısaca şu şekilde açıklanabilir (Bhaumik, Williams, Mobasher, & Burke, 2006) (Mobasher, Burke, Bhaumik, & Williams, 2007) (Bilge & Polat, 2013) (Burke, Mobasher, Williams, & Bhaumik, 2006b) (Aggarwal, 2016):

Rastgele saldırı (Random attack): Rastgele saldırı modelinde, dolgu öğeleri rastgele seçilir ve her bir oy değeri sistem genel ortalaması etrafında dağılan

derecelendirmelerdir. Bu sistem genel ortalaması, kullanıcı ürün matrisindeki tüm ürünler arasındaki tüm oy değerlerinin ortalamasıdır. Bu saldırıda, seçilen ürünler kümesi boştur ve hedef ürün, saldırganın amacı sırasıyla itme ise maksimum veya çekme ise minimum derecelendirmeye ayarlanır. Sistemlerin çoğunda, bir yabancı tarafından sistemin genel ortalamasını belirlemek çok zor değildir, bu nedenle bu saldırıyı gerçekleştirmek için gereken bilgi oldukça azdır. Ancak, bu saldırı genellikle çok etkili değildir.

Ortalama saldırı (Average attack): Ortalama saldırı ile rastgele saldırı arasındaki tek fark, dolgu ürünlerine atanan derecelendirmelerdir. Ortalama saldırı modelinde, dolgu ürünlerine, veri tabanında o ürünü derecelendiren tüm kullanıcılar arasında, o belirli ürün için ortalama derecelendirmeye tam olarak veya yaklaşık olarak karşılık gelen derecelendirmeler atanır. Rastgele saldırıda olduğu gibi, seçilen ürünler kümesi boştur ve hedef ürün, amaca bağlı olarak, ya mümkün olan maksimum reyting değeri (r_{max}), ya da minimum olası reyting değeri (r_{min}) atanır. Rastgele saldırıya benzer şekilde, dolgu ürünleri rastgele seçilir; ancak rastgele saldırıdan farklı olarak, ortalama saldırı, sistemin genel ortalamasından ziyade her bir ürünün ortalamasını kullanır. Her bir dolgu ürünün ortalamasının bilinmesi gerektiğinden, ortalama saldırı rastgele saldırıdan daha fazla bilgi gerektirir, bu nedenle uygulanması daha zordur.

Sürü saldırısı (Bandwagon attack): Enjekte edilen sahte profillerin mevcut kullanıcı profillerine benzer olma şansını artırmak için, sürü saldırı modelinde seçilen ürünler kümesi olarak az sayıda popüler öğe kullanılır. Popüler öğeler terimi aslında çok satan kitaplar, gişe rekorları kıran filmler veya yaygın olarak kullanılan ders kitapları gibi çok sayıda kullanıcı tarafından olumlu bir şekilde derecelendirilmesi muhtemel öğeleri ifade eder. Bir saldırgan sahte kullanıcı profilindeki bu popüler öğeleri her zaman derecelendirirse, hedef kullanıcının tahmin edilen derecelendirmelerinin saldırı tarafından önyargılı olması daha olasıdır. Bu nedenle, seçilen ürünlere ve hedef ürüne, olası maksimum derecelendirme değeri olan r_{max} atanır. Seçilen ürünlere ek olarak, rastgele seçilen bir dizi dolgu ürünü de kullanılır ve sistem geneline dağıtılan derecelendirme değerleri, rastgele saldırıda olduğu gibi ortalama anlamına gelir. Bir sürü saldırısı düzenlemek için eşyaların popülaritesi hakkında biraz bilgi sahibi olunmasına rağmen, bu bilgi sisteme bağlı değildir; kamuya açık bir bilgidir ve genel olarak herhangi bir ürün uzayının en popüler ürünlerini belirlemek çok zor değildir. Bu nedenle, bu saldırı

modelini uygulamak kolaydır. Ayrıca, daha az bilgi gereksinimine rağmen, sürü saldırısı ortalama saldırı kadar iyi performans gösterebilir.

Segment saldırısı (Segment attack): Bu saldırı modeli belirli bir ürünü satın alma olasılığı olan belirli bir kullanıcı grubunu hedef almak için tasarlanmıştır (Mobasher, Burke, Bhaumik, & Williams, Effective attack models for shilling item-based collaborative filtering systems, 2005) (Mobasher, Burke, Bhaumik, & Sandvig, Attacks and remedies in collaborative recommendation, 2007). Başka bir deyişle, belirli bir ürünü tanıtmak isteyen bir saldırganın, ürünün tüm kullanıcılara ne sıklıkla önerildiğiyle değil, olası alıcılara ne sıklıkla önerildiğiyle ilgilenmesi muhtemeldir. Örneğin, bir korku filminin yapımcısı, başka korku filmlerini izlemiş ve beğenmiş film izleyicilerine, filmin tavsiye edilmesini isteyebilir. Saldırı profillerinde, saldırganlar, hedef ürünlerle birlikte, segmentteki kullanıcıların muhtemelen beğeneceği öğeler için maksimum derecelendirme değeri r_{max}' ı ve dolgu öğeleri için minimum derecelendirme değeri r_{min}' i ekler, böylece öğe benzerliklerinin varyasyonlarını en üst düzeye çıkarır.

Sevme/nefret saldırısı (Love/Hate attack): Sevme/nefret saldırısı, bilgi gereksinimi olmayan, ancak yine de hem kullanıcı tabanlı hem de öğe tabanlı öneri algoritmasına karşı etkili bir saldırı olan çok basit bir saldırıdır (Williams C. A., 2012). Saldırı profillerinde, rastgele seçilen dolgu ürünleri minimum derecelendirme değeri r_{min} ile derecelendirilirken, hedef ürünlere maksimum derecelendirme değeri r_{max} verilir.

Ters sürü saldırısı (Reverse Bandwagon attack): Ters sürü saldırısı (Mobasher, Burke, Bhaumik, & Sandvig, Attacks and remedies in collaborative recommendation, 2007), sürü atak modelinin bir varyasyonudur. Bu saldırıda, hedef ürünler kümesi ile en az popüler olan öğelere minimum derecelendirme değeri verilerek profiller oluşturulur. Dolgu ürünlerin derecelendirmeleri, rastgele saldırıda olduğu gibi rastgele belirlenir. Benzer şekilde, ters sürü saldırısının uygulanması nispeten kolaydır.

Yoklayıcı saldırısı (Probe attack): Yoklayıcı saldırısı (Mobasher, Burke, Bhaumik, & Williams, 2007) (O'Mahony, Hurley, & Silvestre, 2005), öneri sahibinin kendisine yanıt vererek profiller oluşturur. Saldırganlar bu stratejiyi gerçekleştirmek için bir çekirdek profil oluşturmak için özgün derecelendirmeler atar ve ardından bunu sistemden öneriler oluşturmak için kullanır. Bu öneriler komşu kullanıcılar tarafından oluşturulmuştur (O'Mahony, Hurley, & Silvestre, 2005). Bu şekilde, bir hedef ürün,

komşu tabanlı bir öneri sisteminde kolayca önyargılı olabilir. Bir anlamda, yoklayıcı saldırısı, saldırganların sistemin derecelendirme dağılımını aşamalı olarak öğrenmesi için bir yol sağlar. Bu saldırının basit olmasının yanında bir diğer avantajı da daha az alan bilgisi gerektirmesidir.

Yukarıda belirtilen atak modellerine ek olarak, *mükemmel bilgi saldırısı* (*perfect knowledge attack*), *örnekleme saldırısı* (*sampling attack*), *favori ürün saldırısı* (*favorite item attack*), *gürültü enjeksiyonu* (*noise injection*), *güçlü kullanıcı saldırısı* (*power user attack*), *güçlü ürün saldırısı* (*power item attack*) vb. şeklinde devam edilebilir.

3.2. Şilin Atakların Tespiti

Çeşitli İF öneri sistemlerinin (İFÖS) şilin saldırılarına karşı savunmasız olduğu çalışmalarla kanıtlanmıştır (Bilge, Gunes, & Polat, 2014) (Burke, Mobasher, Williams, & Bhaumik, 2006a) (Chirita, Nejdl, & Zamfir, 2005) (Gunes, Bilge, Kaleli, & Polat, 2013). İFÖS' lerinde şilin saldırılarının etkisini azaltmanın iki yaygın yolu vardır. İlk yol, şilin saldırı tespiti gerçekleştirmek ve böylece İF algoritmalarını çalıştırmadan önce saldırı profillerini veri tabanından kaldırmaktır. Benimsenen bu yol için bir dizi şema önerilmiştir. Diğer bir alternatif yol, saldırıya dayanıklı İF algoritmaları, yani sağlam öneri algoritmaları geliştirmektir. Bu bölümün alt başlıklarında temel olarak *denetimli sınıflandırma teknikleri* (*supervised classification*), *denetimsiz kümeleme teknikleri* (*unsupervised clustering*), *yarı denetimli teknikler* (*semi-supervised method)* ve *diğer teknikler* olarak şilin saldırı tespit algoritmaları tanıtılmıştır.

3.2.1. Denetimli sınıflandırma teknikleri

Saldırı tespit şemalarında *denetimli sınıflandırma* (*supervised classification*) teknikleri kullanılmaktadır. (Burke, Mobasher, Williams, & Bhaumik, 2006a) (Burke, Mobasher, Williams, & Bhaumik, 2006b) Her bir profilden türetilen niteliklere dayalı olarak kötü niyetli kullanıcıları tespit etmek için bir sınıflandırma yaklaşımı kullanır. Bunlara literatürde *genel nitelikler* (*generic attributes*) olarak adlandırılır. Bu tür türetilmiş özelliklerden ikisi, *ortalama anlaşmadan sapma derecesi* (*rating deviation from mean agreement, RDMA*) ve *üst komşularla benzerlik derecesi* (*degree of similarity with top neighbors, DegSim*), Chirita ve arkadaşları (Chirita, Nejdl, & Zamfir, 2005) tarafından önerilen ve kötü niyetli profillerin tespit edilmesi için bir ölçü olarak kullanılan özelliklerdir. RDMA, ilgili öğe için derecelendirme sayısının tersiyle ağırlıklandırılan her öğe için profilin ortalama sapmasını inceler. DegSim, bir kullanıcının ve en yakın

komşularının benzerliklerinin ortalaması olarak tahmin edilir. (Burke, Mobasher, Williams, & Bhaumik, 2006a) (Burke, Mobasher, Williams, & Bhaumik, 2006b), alternatif olarak, bu metrikleri bir sınıflandırma niteliği olarak değerlendirmiş ve RDMA' ya dayanan iki yeni nitelik türetmiştir. Bu yeniden üretilen nitelikler, *ortalama anlaşmadan ağırlıklı sapma* (*weighted deviation from mean agreement, WDMA*) ve *ağırlıklı anlaşma derecesi* (*weighted degree of agreement, WDA*)' dir. WDMA, güçlü bir şekilde RDMA' ya dayanmaktadır; ancak, derecelendirme çevresinin karesini alarak seyrek öğeler için daha yüksek bir ağırlık verir. Öte yandan WDA, öğe başına derecelendirme sayısını tamamen yok sayar ve kullanıcının derecelendirmelerinin öğe ortalamasından farklarının toplamını kullanır. RDMA tabanlı özniteliklere ek olarak, Burke vd. (Burke, Mobasher, Williams, & Bhaumik, 2006a) (Burke, Mobasher, Williams, & Bhaumik, 2006b) bir genel öznitelik daha önerilmiştir. Belirli bir kullanıcı profilinin uzunluğunun, veri tabanındaki ortalama uzunluktan ne kadar farklı olduğunu ölçen *uzunluk varyansı* (*length variance, lengthVar*) olarak adlandırılır ki burada uzunluk, bir kullanıcının derecelendirme sayısıdır. Ayrıca Burke ve arkadaşları (Burke, Mobasher, Williams, & Bhaumik, 2006a), DegSim' i hesaplarken ortak oran faktörünü dikkate alan ve şilin saldırılarını tespit etmek için bazı özellikler önermişlerdir (Chirita, Nejdl, & Zamfir, 2005). Bunların bazıları, DegSim özniteliğinin başka bir varyasyonu olan DegSim', *Dolgu Ortalama Hedef Farkı* (*Filler Mean Target Difference, FMTD*), *Profil Varyansı* (*Profile Variance*) ve *Hedef Model Odak* özniteliği (*Target Model Focus, TMF*)'dir.

Birçok çalışmada, söz konusu *ortalama anlaşmadan derecelendirme sapması* (*RDMA*), *ağırlıklı anlaşma derecesi* (*WDA*), *dolgu ortalama hedef farkı* (*FMTD*), ve *hedef model odağı* (*TMF*) gibi sınıflandırma niteliklerini kullanan saldırganları tespit etmek için denetimli sınıflandırmada kNN, C4.5 ve SVM sınıflandırıcıları kullanılmıştır (Burke, Mobasher, Williams, & Bhaumik, 2006a) (Mobasher, Burke, Bhaumik, & Williams, 2007) (Williams, Mobasher, & Burke, 2007). Daha sonra, daha fazla saldırı türünü tespit etmek için faydalı nitelikler, *genel nitelikler*, *model nitelikler* (*model attributes*) ve *profil içi nitelikler* (*intra-profile attributes*) dahil olmak üzere üç türe genişletilir (Gunes, Kaleli, Bilge, & Polat, 2014) (Mobasher, Burke, Bhaumik, & Williams, 2007). Alternatif olarak Mobasher ve arkadaşları (Mobasher, Burke, Bhaumik, & Williams, 2007), modele özgü öznitelikleri kullanarak hem istatistiksel teknikleri hem de sınıflandırmayı kullanan bir hibrit saldırı tespit modeli önermektedir. Bryan vd.

(Bryan, O'Mahony, & Cunningham, 2008), amacı sahte profillerin daha yüksek bir H_v puanına sahip olmasını sağlamak olan bir öğe alt kümesi arasında ilişkilendirilen sahte profilleri bulmak için varyansa göre düzeltilmiş bir H_v puanı kullanmıştır. Zhou vd. (Zhou, Koh, Wen, Alam, & Dobbie, 2014), RDMA kullanan grup saldırı profillerini tanımlamak için yeni bir teknik önermişler ve gerçek profiller ile saldırı profilleri arasındaki benzerlik farkını daha iyi ölçen DegSim'e dayalı geliştirilmiş bir metrik DegSim' belirlemişlerdir. Zhang ve Zhou (Zhang & Zhou, 2014), toplu modda çalışmadan, ana fikri özellik çıkarma yöntemi olan Hilbert-Huang dönüşümü (HHT) ve destek vektör makinesini (SVM) birleştirerek profil enjeksiyon saldırılarını tespit etmek için HHT-SVM adlı çevrimiçi bir yöntem sunar ki bu çalışma kullanıcı profillerine dayalıdır ve aşamalı olarak çalışabilir. İlk olarak, öğelerin yeniliği ve popülerliğine dayalı olarak her kullanıcı profili için derecelendirme serileri oluştururlar. Daha sonra her bir derecelendirme serisini ayrıştırmak için ampirik mod ayrıştırma (empirical mode decomposition, EMD) yaklaşımını kullanırlar ve HHT'yi tanıtarak profil enjeksiyon saldırılarını karakterize etmek için Hilbert spektrumu tabanlı özellikleri çıkarırlar. Son olarak, önerilen özelliklere dayalı olarak profil enjeksiyon saldırılarını tespit etmek için SVM'den yararlanırlar.

Dengesiz sınıflandırma problemiyle (imbalance classification problem) başa çıkmak için Yang vd. (Yang, Xu, Cai, & Xu, 2016), saldırı profillerini tüm kullanıcı profillerinden ayırmak için dolgu ürünü veya seçilen ürünler üzerindeki belirli derecelendirmelerin (maksimum puan, minimum puan veya ortalama puan gibi) sayısına odaklanan üç yeni özellik önermektedir. Özellikler *Kendinde Maksimum Derecelendirmeli Dolgu Boyutu* (*Filler Size with Maximum Rating in Itself, FSMAXRI*), *Kendinde Minimum Derecelendirmeli Dolgu Boyutu* (*Filler Size with Minimum Rating in Itself, FSMINRI*) ve *Kendinde Ortalama Derecelendirmeli Dolgu Boyutu* (*Filler Size with Average Rating in Itself, FSARI*). Dahası, Yang ve arkadaşları (Yang, Xu, Cai, & Xu, 2016) çeşitli saldırı modellerinin istatistiksel özelliklerine dayalı olarak kullanıcı profillerinden iyi tasarlanmış özellikler çıkarır ve dengesiz sınıflandırma ile başa çıkabilen şilin saldırılarını tespit etmek için çıkarılan özelliklere dayanan yeniden ölçeklendirme AdaBoost (RAdaBoost) olarak adlandırılan bir AdaBoost varyantı önermişlerdir. Yu vd. (Yu, Li, & Zhang, 2014), söz konusu denetimli yaklaşımların kullanıcı profillerinin artmasıyla aşamalı olarak güncellenemediğini göz önünde bulundurarak, kaba küme teorisini tanıtarak bu bilmeceyi çözmek için aşamalı öğrenmeye

dayalı bir saldırı tespit algoritması önermektedir. İlk olarak, en iyi etiket örneklerini seçerek karar kurallarını eğitmek için en bilgilendirici eğitim setlerini oluşturmaya yönelik bir algoritma kullanılır. İkinci olarak, saldırı profillerini kapsamayı daha makul hale getirmek için karar kurallarını verimli bir şekilde yeniden eğitmek için artımlı bir güncelleme şeması kullanılır. Son olarak, saldırı profillerini gerçek kullanıcı profillerinden ayırt etmek için istatistiksel özelliklere dayalı saldırı tespit algoritması kullanılır.

3.2.2. Denetimsiz kümeleme teknikleri

Saldırı tespitinde kümeleme yaklaşımı ilk olarak O'Mahony vd. (O'Mahony, Hurley, & Silvestre, 2003) tarafından kullanılmıştır. Bu yaklaşımda şüpheli kullanıcıları ortadan kaldırmak için komşuluk seçimi, kümeleme yoluyla filtrelenmiştir. Hedeflenen öğelerin popülaritesini azaltmayı amaçlayan kötü niyetli profilleri tespit etmek için bazı değişikliklerle itibar raporlama sistemlerinde kullanılan kümeleme yaklaşımını kullanırlar. Bu yaklaşım, veri tabanını periyodik olarak kümeler ve küme merkezlerinin önemli ölçüde değişip değişmediğini kontrol eder. Eğer öyleyse, küme merkezlerini etkileyen ekstrem profiller kötü niyetli olarak kabul edilir. Mehta (Mehta, 2007) ve Mehta ve Nejdl (Mehta & Nejdl, 2008), öneri üretme sürecinde kullanılacak kullanıcıları belirlemek için geleneksel en yakın komşu yöntemi yerine kullanıcıları kümelemek için PLSA tabanlı bir kümeleme yöntemini tanıtmaktadır. Bhaumik vd. (Bhaumik, Mobasher, & Burke, 2011), veri setinin istatistiksel özelliklerine dayanan saldırı tespiti için çeşitli sınıflandırma özelliklerine dayalı denetimsiz bir kümeleme algoritması uygulamakta ve buna göre bu özelliklere dayanarak kullanıcı profilleri üretmektedir. Nispeten küçük kümeleri şüpheli kullanıcı grupları olarak bulmak için üretilen profillere k-ortalamalar kümeleme uygularlar.

3.2.3. Yarı denetimli teknikler

Öneri sistemlerinin çoğunda genellikle az sayıda etiketli kullanıcı ve çok sayıda etiketlenmemiş kullanıcı olmasına rağmen hem etiketli hem de etiketsiz kullanıcı profillerinin modellenmesine çok az dikkat edilmiştir. Wu vd. (Wu, Wu, Cao, & Tao, 2012) daha karmaşık şilin saldırı türlerini tespit etmek için bir Hibrit Şilin Saldırı Detektörü (Hybrid Shilling Attack Detector) veya kısaca HySAD sunar. İlk olarak, HySAD algoritması, MC-Relief adlı bir sarmalayıcı aracılığıyla özellik seçimi amacıyla popüler şilin saldırısı algılama ölçümlerini toplar. Daha sonra HySAD algoritması hem

etiketli hem de etiketsiz kullanıcı profillerinin sınıflandırılması için yarı denetimli naive Bayes (SNB_λ) sınıflandırıcısını kullanır. Wu vd. (Wu, ve diğerleri, 2015), şilin tespiti için hem kullanıcı özelliklerini hem de kullanıcı-ürün ilişkilerini birleştirmek ve yüksek tespit oranları elde etmek için hPSD adlı yarı denetimli bir hibrit öğrenme modeli önermektedir. Cao vd. (Cao, Wu, Mao, & Zhang, 2013) ve Wu vd. (Wu, Cao, Mao, & Wang, 2011), hem etiketli hem de etiketsiz kullanıcı profillerinden yararlanmak için yeni bir yarı denetimli öğrenme tabanlı şilin saldırı tespiti veya kısaca Semi-SAD önermektedir. Semi-SAD algoritması, naive Bayes sınıflandırıcısını ve EM-λ' yı, artırılmış bir Beklenti Maksimizasyonu (Expectation Maximization, EM) birleştirir (Nigam, McCallum, Thrun, & Mitchell, 2000). Etiketlenmiş veriler değerli ve etiketlenmemiş verilere kıyasla oldukça nadir olduğundan, algoritma önce küçük bir etiketlenmiş kullanıcı profili kümesi üzerinde naive Bayes sınıflandırıcısını eğitir ve ardından ilk naive Bayes sınıflandırıcısını geliştirmek için EM' yi kullanır.

3.2.4. Diğer teknikler

Bahsedilen yaklaşımlara ek olarak, araştırmacılar, öneri sistemlerindeki saldırıları tespit etmek için başka teknikler de önermektedir. O'Mahony vd. (O'Mahony, Hurley, & Silvestre, 2006), kullanıcı profillerinin olasılık dağılımlarını tahmin ederek ve yorumlayarak bir öneri sistemi veri tabanında doğal ve kötü niyetli gürültü kalıplarını keşfetmek için bir sinyal algılama yaklaşımı önermektedir. Su vd. (Su, Zeng, & Chen, 2005), bir kullanıcının derecelendirmelerinin tutarlılığının ürün-ürün benzerliklerine göre test edildiği bir benzerlik yayma algoritması kullanarak bireysel saldırganlardan ziyade grup şilin kullanıcılarını tespit etmeye odaklanır. Tang ve Tang (Tang & Tang, 2011), tavsiye sistemlerinde ilk *n* listelerini saptırmak için şüpheli davranışları tespit etmek için kullanıcıların derecelendirme zaman aralıklarını analiz eder. Kullanıcıların zaman davranışını ölçmek için yayılma, frekans ve bağlama özelliklerinden oluşan bir öznitelik oluştururlar. Benzer bir yaklaşım olarak, Zhang ve arkadaşları (Zhang, Chakrabarti, Ford, & Makedon, 2006), bir öğe için bir zaman serisi derecelendirmesi oluşturmayı önermektedir. Öğe için ardışık derecelendirmeleri gruplamak için bir pencere kullanırlar, her pencerede örnek ortalamasını ve Entropi 'yi hesaplarlar ve şüpheli davranışı saptamak için sonuçları yorumlarlar. Bryan vd. (Bryan, O'Mahony, & Cunningham, 2008) saldırı profillerinin alınmasını en üst düzeye çıkarmak için ek adımlar içeren H_v-*skor* metriğinin seyrek bir matris varyasyonunu kullanır.

4. ÇÖİF İÇİN YENİ EN İYİ-N ÖNERİ METODU

Bu bölümde çok ölçütlü sistemler için kullanıcılara en iyi-n öneri sistemlerinin geliştirildiği yaklaşımlara yer verilmiştir.

4.1. Giriş

Çok ölçütlü değerlendirmelerin olduğu sistemlerde, kullanıcılar hakkında daha fazla bilgiye sahip olmak, etkin müşteri analizi yapılmasını sağlar. Ancak burada cevaplanması gereken soru, ÇÖİF sistemlerinde kullanıcı bilgilerinin nasıl etkin bir şekilde analiz edileceği ve kişiselleştirilmiş tavsiyelerin nasıl sağlanacağıdır. Öneri sistemlerinde kullanıcı analizi, müşterilerin derecelendirme ölçeklerini nasıl değerlendirdiğine ve bu değerlendirme sırasındaki tutumlarına/eğilimlerine bağlıdır. Kullanıcılar tercihlerini ifade etmek için aynı derecelendirme ölçeğini kullansalar da verdikleri oy değerleri değerlendirmelerine göre değişebilir. Bazı kullanıcılar bir öğeyi kolayca beğenebilirken, bir başkası için durum böyle olmayabilir. Örneğin, 1-5 derecelendirme ölçeğinde, bir kullanıcı beğenisini 5 yıldız ile ifade edebilirken, aynı kullanıcı beğenmediği ürünü 1 skoru ile değerlendirebilir. Ancak başka bir kullanıcı bu beğeni 4 veya 5 puan iken, beğenmeme 1, 2 derecelendirme olabilir. Bu nedenle ilk kullanıcı tüm oy ölçeğini kullanmadan ikili bir tutum (beğenme veya beğenmeme) sergilerken, aynı tutum ikinci kullanıcı için geçerli değildir. Dolayısıyla eğer kullanıcı oy ölçeğindeki tüm derecelendirmeleri kullanırsa, bu durum kullanıcının veya ürünün *DU'* sunu arttıracaktır (Kaleli, 2014). Bu *DU*, ÇÖİF' deki kişiselleştirilmiş tavsiye oluşturma aşamasının kritik noktası olan komşuluk seçme sürecini etkileyebilir. Kullanıcılar arasındaki *DU* değerinin yakın olması, iki kullanıcının derecelendirme alanını benzer şekilde değerlendirdiğini gösterebilir. Dahası, aralarındaki benzerlik ve korelasyon da benzer şekilde yüksekse, bu iki kullanıcı arasında sağlıklı bir komşuluk olduğu sonucuna varılabilir.

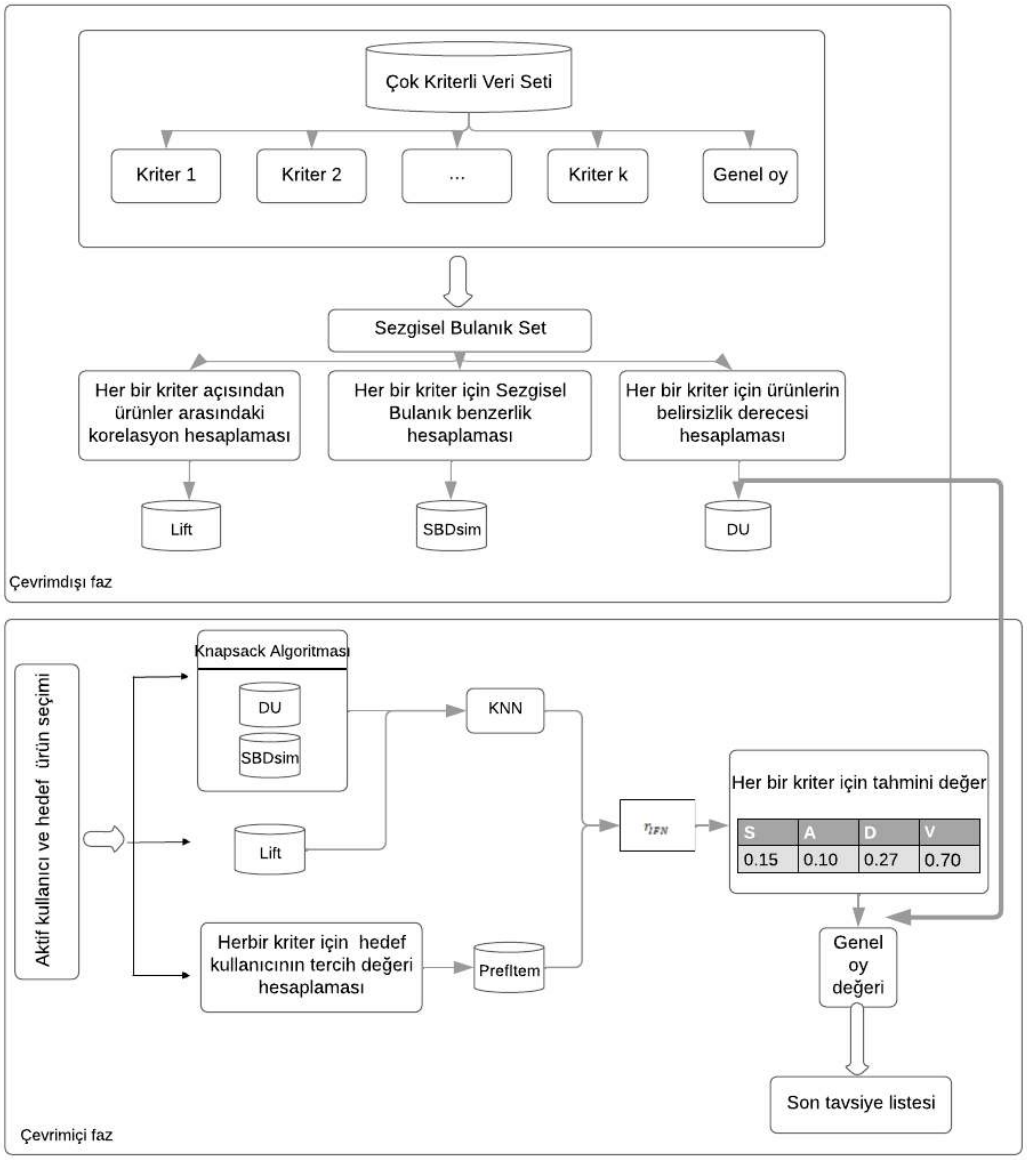
Öneri sistemleri için bir diğer önemli nokta ise değerlendirme sırasında yaşanan, kararsızlık ve tereddüt durumunu gösteren kullanıcı tutumudur. Müşteriler bir ürünü deneyimlediğinde ve değerlendirdiğinde, ürün hakkındaki düşüncelerini dilsel olarak ifade etmeleri daha kolaydır. Kesin sayıların kullanılması, ürünün o kümeye ait olup olmadığı konusunda kesin bir çıkarım sağlamaktadır. Örneğin bir kullanıcı, deneyimlerine göre bir ürünü değerlendirdikten sonra 5 yıldız üzerinden 4 puan verebilir. Bu ölçekte dil açısından dört iyi ifade edilebilir ve bu puana göre kullanıcının bu ürünü

uygun bulduğu söylenebilir. Ancak gerçek hayatta bazen böyle kesin değerlendirmeler olmayabilir. Kullanıcının bu ürün beğenisiyle ilgili iyi ve orta arasında kalabilir. Dolayısıyla, kullanıcı bu ürünü %75 oranında iyi, %20 oranında iyi olmadığı şeklinde bir düşünceye sahip olabilirken, değerlendirme sırasında yaşayabileceği tereddüt ise %5 olabilir. Bu nedenle, kişiselleştirilmiş öneriler üretme sürecinde, kullanıcı değerlendirmesindeki belirsizlik ve tereddüt durumu, sezgisel bulanık set (SBS) kullanılarak iyi bir şekilde ifade edilecektir. SBS, tereddüt ve belirsizlikle başa çıkma açısından geleneksel bulanık kümeden daha esnek/pratiktir ve bu yaklaşım öneri sürecindeki başarı ve kullanıcı memnuniyetini arttıracaktır. Sonuç olarak, iyi bir kullanıcı analizi komşuluk seçme aşamasının başarılı olmasını sağlayacak ve bu başarı ÇÖİF için kullanıcı memnuniyetini artıracaktır. Bu nedenle, bu başlıkta ÇÖİF sistemleri için çok ölçütlü değerlendirmeler kullanılarak yeni bir en iyi-n öneri yöntemi geliştirilmiştir. Bu çalışma, kullanıcılar hakkında çıkarımların ve sağlıklı komşuluk oluşturmanın, kullanıcı memnuniyetini artırıp artırmadığı gösterilecektir.

Son yıllarda yapılan çalışmalar incelendiğinde, kullanıcılara doğru tahminler sunmanın yanı sıra üretilen önerilerin çeşitliliği, yeniliği ve tesadüfi olması kullanıcı memnuniyeti açısından oldukça önemli olduğu sonucuna varılabilir. Böylece oluşturulan listenin kalite değerlendirmesine ek olarak, önerilen modelin şans eseri bulma, yenilik ve çeşitlilik ölçümü sunulacaktır.

4.2. Önerilen Yaklaşım

Önerilen modelimizde, ÇÖİF sistemlerinde kullanıcıların ilgisini çekebilecek ürünlerin bir listesinin oluşturulması amaçlanmaktadır. Sezgisel Bulanık Çok Ölçütlü Entropi ve ARM-tabanlı İşbirlikçi Filtreleme (Intuitionistic Fuzzy Multi-Criteria Entropy and ARM-based Collaborative Filtering, IFMC-EntARMCF) modeli ve bu modelin detayları Şekil 4.1’ de gösterilmektedir.



Şekil 4.1. *Önerilen yaklaşımın şeması*

Her bir ürün hakkındaki olası bilgileri kapsayan gerekli hesaplamalar/parametreler, önerilen yöntemin ürün tabanlı algoritmasına dayalı olarak çevrimdışı fazda belirlenir. Tüm derecelendirme değerlerinin kullanılması, herhangi bir ürün hakkında belirsizliğe yol açabildiğini önceden belirtmiştik. Bu nedenle, her bir kriter açısından ürünlerin *DU* değeri belirlenir ve kullanıcıların oy değerleri incelendikten sonra ürünler arasındaki ilişkisel yapı ortaya çıkarılır. Ürünler arasındaki benzerliklerin yanı sıra, bu ürünlerin davranışını ve korelasyonunu belirlemek için *lift* hesaplamaları yapılır. Sonraki aşamada,

hedef ürünün en yakın komşuları çevrimiçi fazda bulunur ve aktif kullanıcı hedef ürün için tahmin üretme süreci başlar. Hedef ürüne en yakın komşular çevrimiçi olarak bulunduktan sonra, aktif kullanıcı ve hedef ürün için tahminler üretilir.

Klasik komşuluk seçim yönteminde ürünler veya kullanıcılar arasındaki benzerlikler göz önünde bulundurularak en yakın olanlar bulunur. Ancak her bir kullanıcının farklı karakteristik yapılara sahip olması kullanıcıların değerlendirme esnasındaki tutumunu etkilemektedir. Dolayısıyla, klasik komşuluk seçme yöntemlerinde bu durum, kullanılan oy değerlerinin dağılımı ve karakteristik yapısı göz ardı edilmektedir. Bu nedenle, Kaleli (Kaleli, 2014) tarafından önerilen yaklaşım, komşuluk sürecini oluşturmanın ilk adımı olarak dikkate alınmaktadır. Ürünler arasındaki korelasyon da belirlendikten sonra komşuluk seçim sürecinin son adımı tamamlanmış olur. Bu aşamada, veri setlerindeki ürünler istatistiksel olarak bağımsız kabul edilir ve *lift* hesabı dikkate alınarak benzer kullanıcılar ortaya çıkarılır. Dolayısıyla, komşuluk seçme sürecine, araştırmacı tarafından önerilen yaklaşıma *lift* hesaplaması da dahil edilerek gerçekleştirilmiştir.

Komşu ürünler belirlendikten sonra, hedef ürün için bu komşuluklar kullanılarak tahmini oy değeri belirleme süreci başlar. İlk aşamada, kullanıcıların belirlenen oy dağılımları dikkate alınarak tercih skor hesaplaması yapılır. Daha sonra her bir kriter için elde edilen bu tahmini oy değerleri ağırlıklandırarak genel bir skor bulunur. En yüksek genel skora sahip ürünler öneri listesine eklenir ve bu liste kullanıcıya sunulur. Önerilen metodoloji (IFMC-EntARMCF) detaylı bir şekilde aşağıda açıklanmıştır.

Bulanıklaştırma: Bu aşamada, kullanıcılar tarafından verilen kesin derecelendirme değerleri ilk olarak Tablo 4.1' de belirtildiği gibi sezgisel bulanık değerlere (*SBD*) dönüştürülür.

Tablo 4.1. *Dilsel terimler ve sezgisel bulanık değer karşılıkları*

Kesin Değer	Dilsel Terimler	SBD
5	Güçlü olarak ilgili	(0,90, 0,10)
4	Oldukça ilgili	(0,75, 0,20)
3	İlgili	(0,50, 0,45)
2	Az ilgili	(0,35, 0,60)
1	İlgisiz	(0,10, 0,90)
-	-	N/A

Benzerlik hesaplaması: Önerilen yaklaşımın ikinci adımında, ürünler arasındaki ilişkisel yapıyı belirlemek için Ke ve arkadaşları (Ke, Song, & Quan, 2018) tarafından önerilen benzerlik hesaplaması kullanılır.

$$sim(A,B) = 1 - \sum_{u \in U_{A,B}}^{N} \sqrt{\left(\frac{\mu_A(x_u) - v_A(x_u)}{2} - \frac{\mu_B(x_u) - v_B(x_u)}{2}\right)^2 + \frac{1}{3}\left(\frac{\mu_A(x_u) + v_A(x_u)}{2} - \frac{\mu_B(x_u) + v_B(x_u)}{2}\right)^2} \quad (4.1)$$

Burada, $sim(A,B)$, A ve B ürünleri arasındaki benzerlik değerini ifade eder. $U_{A,B}$, A ve B ürünlerini değerlendirmeye alan kullanıcı kümeleri iken, $\mu_A(x_u)$ ve $v_A(x_u)$, A ürününe u kullanıcısının verdiği *SBD*' dir.

A ve B ürünleri arasındaki benzerlik hesaplamasında, her iki ürünü de değerlendirmeye alan kullanıcılar dikkate alınır. Bununla birlikte, bu tür benzerlik hesaplamasının bazı sıkıntıları vardır. Örneğin, iki ürün arasındaki benzerliğin 0,65 olduğunu ve her iki ürünü de değerlendirmeye alan kullanıcı sayısının 4 olduğunu varsayalım. Başka bir durumda ise, ürünler arasındaki benzerliğin 0,55 olduğunu ve bu ürünleri ortak değerlendiren kullanıcı sayısının 50 olduğunu varsayalım. Her iki durumda karşılaştırıldığında, ilk durumda benzerlik yüksek olmasına rağmen güvenirlik az, ikinci durumda ise güvenilirlik düzeyi daha yüksektir. Bu nedenle, *Jaccard* metriği bu sorunu çözmek için bir çözüm olarak kullanılır.

$$Jaccard(A,B) = \frac{|S_A \cap S_B|}{|S_A \cup S_B|} \quad (4.2)$$

Her iki ürünü de değerlendirmeye alan kullanıcı sayısı $|S_A \cap S_B|$ ifade edilir. Bu metrik ortak oy veren kullanıcı oranını bulur.

Denklem 4.1 ve 4.2 birleştirilerek, önerdiğimiz modelin benzerlik hesaplaması elde edilir.

$$Csim(A,B) = sim(A,B) \times Jaccard(A,B) \quad (4.3)$$

Komşuluk seçimi: Önerilen yaklaşımın üçüncü adımı komşuluk seçimi yöntemidir. Komşuluk seçimi aşamasında ilk olarak, çevrimdışı fazda elde edilen parametreler kullanılarak Kaleli (Kaleli, 2014) tarafından önerilen yöntem uygulanır. Her bir kriter açısından her bir ürünün *DU* değeri Denklem 2.5 kullanılarak elde edilir. Ardından, hedef ürünün komşuluğunu bulmak için, hedef ürün *i* ile diğer ürünler arasındaki *DU* değer farklarının mutlak değeri hesaplanır. Sabit bir *C* katsayısı bulmak için bu farkların

ortalaması en yakın komşuluk sayısı (k=300) ile çarpılır (Denklem 2.7). Komşuluk seçim sürecinin sonraki aşamasında kriterler açısından ürünler arasındaki korelasyon ilişkisini bulmak için Denklem 2.12 kullanılarak *lift* hesaplaması yapılır.

Bu hesaplama sonucunda hedef ürün *i* ile en yüksek korelasyona sahip olan ve sırt çantası (Knapsack) algoritmasına göre koşulu sağlayan ürünler belirlenir. Bu ürünlerden en büyük değere sahip olanlar en yakın komşu olarak seçilir.

Her bir kriter için oy değeri tahmini: Denklem 2.8 kullanılarak komşu ürünler için bulunan tercih skorundan sonra, Denklem 4.4 kullanılarak her bir kriter için *u* kullanıcısının hedef ürünü *i* için tahmini değer hesaplanır.

$$r_{u,i}{}^{kriter_t} = \sum_{j \in NN} pref(r_{u,j}) \times Csim_{i,j} \tag{4.4}$$

Her bir kriter için ürün tabanlı tahmini değerlerin birleştirilmesi: Her bir kriterin tahmini değeri hesaplandıktan sonra, genel oy değeri Denklem 4.5 kullanılarak bulunur. Denklemde gerekli olan ağırlık değerleri Entropi yardımıyla hesaplanmıştır. Entropi kavramı incelendiğinde, değişkenin olası sonuçlarının doğasında var olan ortalama bilgi, sürpriz veya belirsizlik düzeyi olarak tanımlanabilir (Shannon, 1948). Tanımdan da anlaşılacağı gibi Entropi, bilgi düzeyini ölçen ve belirsizlik değerini belirleyen bir metriktir. Bu nedenle, ürünlerin ağırlık katsayı değerleri için normalleştirilmiş Entropi değeri kullanılır.

$$r_{u,i}{}^{overall} = \frac{\sum_{t=1}^{c} w_i{}^{kriter_t} r_{(u,i)}{}^{kriter_t}}{\sum_{t=1}^{c} w_i{}^{kriter_t}} \tag{4.5}$$

w_i , *i* ürününün *DU* değerini ifade eder.

Son tavsiye listesi: Son adım, aktif kullanıcının memnun kalabileceği *n* adet ürün listelerini hazırlamaktır. Burada, genel oy değeri hesaplanan tüm ürünler arasından en yüksek değerlere sahip ürünler, kullanıcının öneri listesine dahil edilir.

Önerilen yöntemin tüm adımları Şekil 4.2’ de verilmiştir.

Girdi: kullanıcı-ürün matrisi ($D_{n\times m}$), aktif kullanıcı vektörü ($U_{1\times m}$), hedef ürünler (q), oy ölçeği (RD), komşuluk sayısı (k), kriter (t)

Çıktı: Üst-n öneri listesi

for each C_t kriterleri **do**
  for D matrisindeki tüm i ürünleri **do**
    $IFRatings \leftarrow Fuzzification_t$
    $totalRatings \leftarrow FindTotald_t \triangleleft i$ ürünün oy vektörü d_t toplamı bul
      for D matrisindeki tüm r oy değerleri **do**
        $numof_r \leftarrow Findd_t, r \triangleleft i$ ürünün oy vektöründe d_t oy sayısını bul
        $prob_t[r] = numof_r/totalRatings$
      end for
      $DU_i^t \leftarrow prob_t, totalRatings$
  end for
  for D matrisindeki tüm i ürünleri **do**
    for D matrisindeki tüm j ürünleri **do**
      $sim_t[j]^t \leftarrow IFNSimd_t, d_j$
      $dif_t[j]^t \leftarrow |DU_i - DU_j|$
      $numLike(i,j) \leftarrow Findd_t, d_j \triangleleft i$ ve j ürünlerine r_{max} veren kullanıcı sayısını bul
      $support(i,j)^t = numLike(i,j)/size(usersrated_{i,j})$
      $numLike(i) \leftarrow Findd_t \triangleleft i$ ürününe r_{max} veren kullanıcı sayısını bul
      $support(i)t = numLike(i)/n$
      $numLike(j) \leftarrow Findd_j \triangleleft j$ ürününe r_{max} veren kullanıcı sayısını bul
      $support(j)^t = numLike(j)/n$
      $lift_t[j]^t \leftarrow support(i,j)^t/(support(i)^t * support(j)^t)$
    end for
    $C = round(mean(dif_t) \times k)$
    $NNt \leftarrow ((Knapsackdif_t, IFNSim_t, C) \cap lift_t)$
  end for
  $Pref(r_u)^t \leftarrow \alpha = 1$ and $\beta = 0.5$
  $P_{u,q}^t \leftarrow Pref(r_u)^t, NN_qt, q, IFNSim_qt \triangleleft$ her bir kriter t için hesapla
end for
$P_{u,q}^{overall} \leftarrow DU, P_{u,q}^t \triangleleft$ aktif kullanıcının u hedef ürünü q için genel oy değeri hesapla
Üst-$n_u \leftarrow$seç n(sırala($P_{u,allitems}$))

Şekil 4.2. *IFMC-EntARMCF algoritmasının sözde kodu*

Önerilen bu yöntemde, sezgisel bulanık sayılar yerine kesin değerler kullanıldığında nasıl bir sonuç elde edildiği incelenmek istenmiştir. Bu nedenle CrispMC-EntARMCF olarak önerilen yaklaşımda Şekil 4.2' deki tüm hesaplamalar gerçek derecelendirme değerlerine göre yapılır ve sadece benzerlik hesaplamasında kullanılan formül farklılaşmaktadır. Bu yaklaşımda, ürünler arasındaki benzerlikleri hesaplamak için ayarlanmış kosinüs benzerlik (adjusted cosine similarity) metriği kullanılır. Denklem 4.6 kullanılarak yapılan benzerlik hesaplamasında, her iki ürünü de değerlendirmeye alan kullanıcılar dikkate alınır.

$$sim(A,B) = \frac{\sum_{u \in U}(r_{u,A} - \overline{r_u})(r_{u,B} - \overline{r_u})}{\sqrt{\sum_{u \in U}(r_{u,A} - \overline{r_u})^2}\sqrt{\sum_{u \in U}(r_{u,B} - \overline{r_u})^2}} \quad (4.6)$$

u kullanıcısının ortalama oy değeri $\overline{r_u}$ ile ifade edilirken, $r_{u,A}$ ve $r_{u,B}$, *u* kullanıcısının *A* ve *B* ürünlerine verdiği oy değerleridir.

4.3. Deneysel Değerlendirmeler

Bu bölümde, önerilen yöntemlerini değerlendirmek için yapılan deneysel çalışmalar yer almaktadır. İlk olarak önerilen yöntemleri karşılaştırmak için seçilen kıyaslama algoritması açıklanmıştır. Daha sonra deneyde izlenen metodolojiden ve performans değerlendirme ölçütlerinden ayrıntılı bir şekilde bahsedilmiştir. Son olarak deneysel sonuçlara yer verilmiştir.

4.3.1. Kıyaslama algoritması: BaseMCCF

Bu bölümde, önerilen yöntemin sonuçlarını karşılaştırmak için kıyaslama algoritması olarak tek kriterli veri seti üzerinde uygulanan en iyi-n öneri sistemlerinin çok ölçütlü sisteme uyarlanması yapılmıştır. Kıyaslama algoritmasının, Temel Çok Ölçütlü İF (Baseline Multi-criteria Collaborative Filtering, BaseMCCF) olarak adlandırılan çok ölçütlü veri seti üzerindeki uygulaması alt bölümlerde adım adım gösterilmektedir.

Bu yöntemde, ilk olarak Denklem 4.6 kullanılarak kriterler açısından ürünler arasındaki benzerlik hesaplaması yapılır. Diğer yöntemlerde olduğu gibi, bu yöntemin benzerlik hesaplamasında da ürünleri ortak değerlendiren kullanıcılar dikkate alınır.

Benzerlik hesaplamasından sonraki adım, tahmin üretilmesidir. Bu aşamada, tahmini değer, hedef ürün *i* ile benzer davranış sergileyen diğer ürünler dikkate alınarak hesaplanır. Ancak buradaki kritik nokta, seyrek bir veri kümesi olması durumunda benzerlik hesaplamasında güvenilir sonuçlar üretilemeyebilir (Cremonesi, Koren, &

Turrin, 2010). Dolayısıyla, bu algoritmada tahmin üretme aşamasında bir katsayı ($d_{A,B}$) kullanılmıştır. Ayrıca en iyi-n öneri sisteminin amacı, yüksek tahmini değerlere sahip ürünlerden bir liste oluşturmaktadır. Dolayısıyla tahmin üretme aşamasında, normalizasyon işleminin yapılmaması, sistem doğruluğunu geliştirdiği gözlenmiştir.

Her bir kriter için aktif kullanıcı *u*, hedef ürünü *i* için tahmini değer aşağıdaki denklem kullanılarak hesaplanır.

$$d_{i,j} = \frac{n_{i,j}}{n_{i,j} + \lambda_1} sim_{i,j} \tag{4.7}$$

$n_{i,j}$, *i* ve *j* ürünlerini ortak değerlendiren kullanıcı sayısını, λ_1 küçülme faktörünü, $sim_{i,j}$, *i* ve *j* ürünlerinin benzerlik değerini ifade eder. Bu çalışmada λ_1 değeri 100 olarak seçilmiştir.

$$r_{u,i}{}^{kriter} = b_{u,i} + \sum_{j \in D_{(u,i)}{}^{k}} d_{i,j}(r_{u,j} - b_{u,j}) \tag{4.8}$$

$b_{u,i}$, kullanıcı *u* hedef ürünü *i* ile ilişkilendirilen taraflılık değeridir (Koren, 2008).

Her bir kriter açısından *u* kullanıcısının *i* ürünü için tahmini değerleri hesaplandıktan sonra, Denklem 4.5 kullanılarak genel oy değeri hesaplanır. Bu hesaplama için gerekli olan ağırlık katsayı değerleri Denklem 2.9 kullanılarak elde edilir. Bu lineer kombinasyon da her bir kriter bağımsız değişken iken genel oy değeri bağımlı değişkeni ifade eder. Bu regresyon sonucunda elde edilen beta değerleri $\beta_1, \beta_2, \ldots, \beta_n$ ağırlık katsayıları olarak kullanılır ve genel sonuç hesaplanır. Bulunan değerler sonucunda kullanıcı için bir öneri listesi hazırlanır.

4.3.2. Değerlendirme kriterleri

Öneri sistemlerinin performansı farklı değerlendirme metrikleri kullanılarak değerlendirilir. Bu bölümde çalışmada dört farklı metrik kullanılmıştır. Kullanılan metriklerle ilgili bilgiler aşağıda yer almaktadır:

Geri Çağırma (Recall): Sistemin genel performansını ölçmek için bu metrik kullanılmıştır. Eğer test ürün *i*, kullanıcıya sunulan öneri listesi içerisinde ($p \leq N$) ise *isabet* (*hit*), değilse *isabet edememe* (*miss*)' dir. Bu metriğin nasıl hesaplandığı aşağıdaki denklemde gösterilmiştir.

$$geri\ çağırma\ (N) = \frac{\#isabet}{|T|} \tag{4.9}$$

T test oylarının toplam sayısını ifade eder.

Çeşitlilik (Diversity): Çeşitlilik, tavsiye listesindeki ürünler arasındaki farklılıkların ortalaması olarak ifade edildiğini önceki bölümlerde yer vermiştik (Bradley & Smyth, 2001). Matematiksel hesaplaması aşağıda yer almaktadır.

$$çeşitlilik = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(1 - benzerlik(c_i, c_j))}{\frac{n}{2} \times (n-1)} \tag{4.10}$$

Yenilik (Novelty): Yenilik, kullanıcının bilmediği ama şaşırdığı ürünlerin sunulması olarak tanımlamıştık (Zhang L. , 2013). Bu metrik Denklem 4.11 kullanılarak hesaplanır.

$$yenilik = \frac{1}{|R|}\sum_{i \in R} -\log_2(p(i)) \tag{4.11}$$

R, kullanıcıya sunulan öneri listesinin boyutunu; *p(i), i* ürününün herhangi bir kullanıcı tarafından derecelendirilme olasılığını temsil eder.

Şans Eseri Bulma (Serendipity): Şans eseri bulma, önceki bölümlerde tanımlamasına yer vermiştik. Bu metriğin hesaplama adımları aşağıda gösterilmektedir.

İlk aşamada, *nokta bazında karşılıklı bilgi* (*point-wise mutual information, PMI*) hesaplaması yapılır. Bu hesaplamada, her iki ürünü ve her bir ürünü ayrı ayrı değerlendirerek kullanıcıların sayısına göre iki ürünün ne kadar benzer olduğunu gösterir.

$$PMI(i,j) = \log_2 \frac{\frac{p(i,j)}{p(i)p(j)}}{-p(i,j)} \tag{4.12}$$

Burada *p(i)* herhangi bir kullanıcının *i* ürününe oy vermiş olma olasılığı iken, *p(i, j)* ise *i* ve *j* ürünlerinin birlikte derecelendirilme olasılığıdır.

Denklem 4.12' ye göre, *PMI* değerinin -1 olması iki ürünün hiç değerlendirmediğini, 0 ise, iki ürünün birbirinden bağımsız olmasını ve 1 sonucu ise iki ürünün birlikte ortaya çıktığını ifade eder.

Kullanıcıya sunulan ürünlerin, şans eseri bulma değerini ölçmek için öneri listesindeki her bir öğe için *PMI* hesaplanır. Yüksek *PMI* değeri, *i* ve *j* ürünlerinin temel olarak birlikte meydana geldiğini ve bu durum iki ürünün birlikte olmasının kullanıcı için

şaşırtıcı olmadığını gösterir. *PMI'* nın tamamlayıcısı [0,1]' e normalize edilerek şans eseri bulma metriği bulunur.

$$şans\ eseri\ bulma = \frac{1}{|R|}\sum_{i \in R}\frac{1 - PMI(i,j)}{2} \tag{4.13}$$

R, kullanıcıya sunulan öneri listesinin boyutunu temsil eder.

4.3.3. Deney metodolojisi

Bu yaklaşımda, Koren (Koren, 2008) tarafından önerilen *bir artı rastgele* (*one plus random*) metodolojisi uygulanır. Deneylerin ilk aşamasında, veri seti, *k*-fold kullanılarak prob kümesi *P* ve eğitim kümesi *M* olmak üzere ikiye ayrılır. *P* veri setinde yüksek derecelendirmeye sahip ürünler, test veri setini *T* oluşturur. Daha sonra, *T* test kümesindeki *u* kullanıcısı tarafından yüksek puan alan her bir *i* ürünü için:

- Kullanıcının değerlendirmediği 1000 ürün rastgele seçilir.
- 1000+*i* ürünlerin tahmini oy değerleri bulunur.
- 1001 ürün tahmini değerlerine göre sıralanır. Bu listedeki test ürün *i* sıralaması *p* ile gösterilir.
- Bu sıralı liste kullanılarak *u* kullanıcı için en iyi-n öneri listesi hazırlanır.

4.3.4. Deney sonuçları

Bu çalışmada, ÇÖİF sistemleri için kullanıcı ve ürün özelliklerini, eğilimini ve derecelendirme dağılımını dikkate alarak en iyi-n öneri listesi oluşturmaya yönelik bir yaklaşımı tanıtmayı amaçladık. Üç farklı veri setine (YM5, YM10 ve YM20) üç farklı yöntem (BaseMCCF, IFMC-EntARMCF ve CrispMC-EntARMCF) uygulanmış ve performans değerlendirmesi yapılmıştır.

Deneylerin ilk aşamasında, veri seti, test metodolojisine dayalı olarak 10-fold kullanılarak bir eğitim *M* ve bir prob seti *P* olarak ayrılmıştır. *P* setinde, bir kullanıcı için beş yıldıza sahip her ürün, test veri setini *T* oluşturmuştur. Her kullanıcının beş yıldızlı ürününe ek olarak, *T* için rastgele seçilen 1000 adet derecelendirilmemiş ürün seçilmiştir. Daha sonra, aktif kullanıcının 1001 ürünü için tahmin üretme ve komşuluk seçim aşamasında üç farklı yöntem uygulanmıştır.

BaseMCCF- Bu yaklaşımda ilk olarak her bir kriter açısından ürünler arası benzerlik bulunur. Daha sonra en yakın komşu sayısı 300 seçilerek hedef ürünün komşuluğu belirlenir ve 1001 ürünün her biri için tahmini bir değer üretilir.

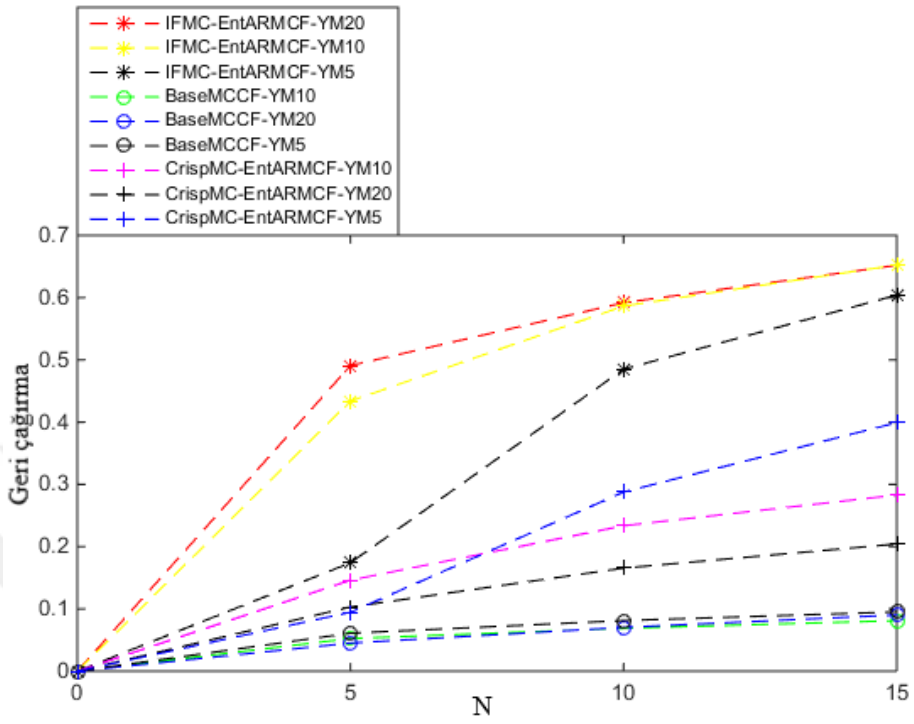
IFMC-EntARMCF- Önerilen bu yaklaşımda, kullanıcıların derecelendirme değerleri *SBD* dönüştürülür. Daha sonra ürünler arasında benzerlik hesabı yapılır ve en yakın 300 komşu seçilir. Komşuluk ürünler kullanılarak 1001 ürün için tahmini değerler belirlenir.

CrispMC-EntARMCF- Önerilen yaklaşımın bir diğer adımı da, IFMC-EntARMCF yöntemde belirlenen adımların kesin değerlere uygulanmasıdır.

Her üç durumda da son adım olarak, kullanıcıya 5, 10 ve 15 adet üründen oluşan bir öneri listesi hazırlanır.

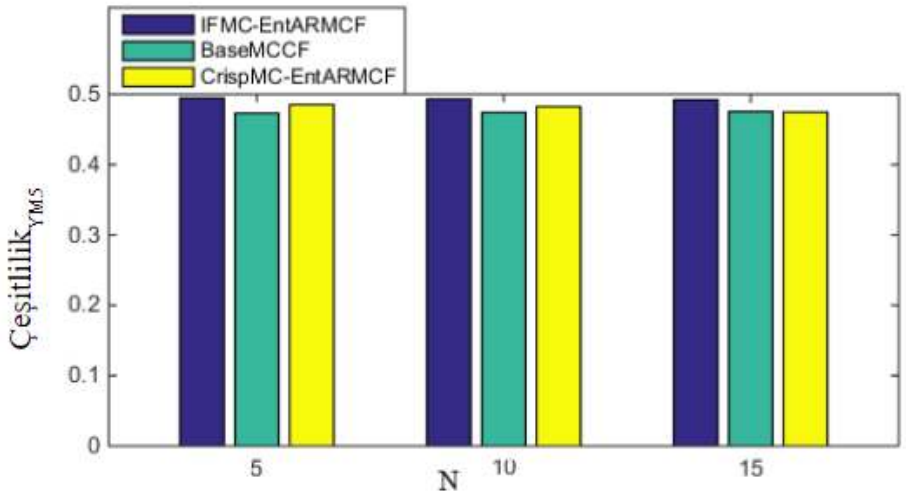
Önerilen yöntemlerin doğruluğunu çok ölçütlü veri setine uygulanan temel yöntemle karşılaştırmak için önce geri çağırma hesaplaması yapılmıştır. Şekil 4.3 incelendiğinde kullanıcı için 5, 10 ve 15 filmlik öneri listeleri hazırlanır. YM10 veri seti için, beş filmden oluşan öneri listesinin sonuçlarına bakıldığında, kullanıcı memnuniyet düzeyi BaseMCCF yönteminde %5, CrispMC-EntARMCF yönteminde %14,6 iken IFMC-EntARMCF' de %43' a kadar çıkmaktadır. Aynı şekilde 15 filmlik bir listede IFMC-EntARMCF' de kullanıcının memnuniyet seviyesi %65,2 gibi yüksek rakamlara ulaşmaktadır. YM10 veri seti için en iyi-15 liste sonuçlarına göre, kullanıcının beğendiği film sayısı IFMC-EntARMCF, BaseMCCF ve CrispMC-EntARMCF yöntemleri için sırasıyla 10, 1 ve 4' tür. Bir diğer veri seti olan YM20 için sonuçlar incelendiğinde, önerilen yöntem (IFMC-EntARMCF) kullanıldığında, kullanıcıların on filmin yaklaşık yarısından fazlasını beğendiği görülmektedir. Veri seti için en kötü sonuç BaseMCCF yöntemi kullanıldığında elde edilir. Son veri seti olan YM5' de en iyi-5 için BaseMCCF yöntemine göre kullanıcı memnuniyeti %6,1; CrispMC-EntARMCF ve IFMC-EntARMCF yöntemlerinde bu oranın sırasıyla %9,4 ve %17,4 gibi oranlara ulaşmaktadır. Kullanıcıya sunulan listedeki ürünlerin sayısında bir artış olması durumunda CrispMC-EntARMCF ve IFMC-EntARMCF yöntemleri için memnuniyet oranının oldukça arttığı görülmektedir. Dolayısıyla kullanıcıya sunulan film listesi genişletilirse oranlarda artış meydana gelmektedir. Her üç veri seti içinde, tüm yöntemler incelendiğinde en iyi

sonuçların IFMC-EntARMCF yöntemiyle, en kötü sonuçların ise BaseMCCF yöntemiyle elde edildiği görülmektedir.



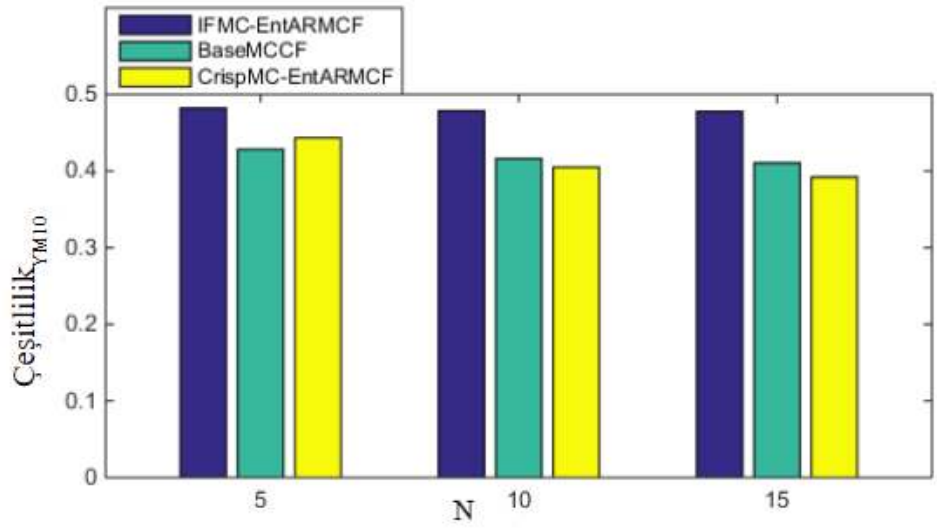
Şekil 4.3. *Üç yöntemin üç veri seti üzerinde geri çağırma sonuçları*

YM5 veri seti için hazırlanan listelerdeki ürün çeşitlilik oranları incelendiğinde beş filmden oluşan bir listedeki ürün çeşitlilik oranı IFMC-EntARMCF, CrispMC-EntARMCF ve BaseMCCF yöntemlerinde sırasıyla %49,4, %48,5 ve %47,2 şeklindedir. Bu sonuca göre "Beş ürün arasından, kullanıcıya değişik gelen ürün sayısı tüm yöntemler için yaklaşık olarak üçtür." sonucuna varılabilir. Şekil 4.4' te görüldüğü gibi kullanıcılara önerilen listede yer alan ürünlerin sayısında meydana gelen artış, IFMC-EntARMCF ve CrispMC-EntARMCF yöntemlerinin çeşitlilik performansında düşüşe neden olmaktadır. Ancak tersi durum BASEMCCF yöntemi için geçerlidir. Sonuç olarak bu veri seti için ürün çeşitliliği açısından en iyi performans sonucunun IFMC-EntARMCF yöntemi ile elde edilebileceği söylenebilir.



Şekil 4.4. *YM5 veri seti için üç yöntemin ürün çeşitlilik sonucu*

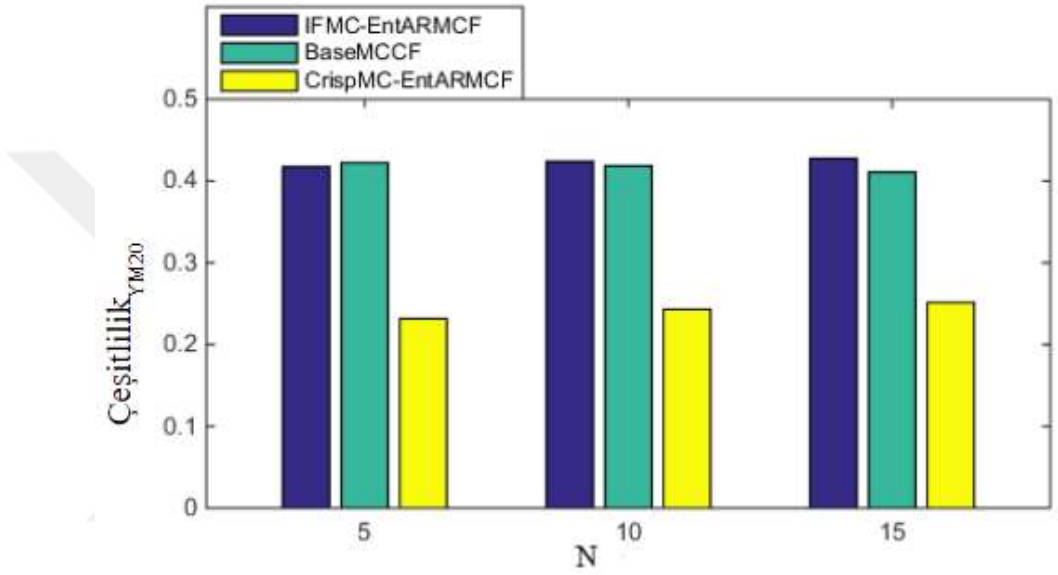
Şekil 4.5' teki sonuçlara göre, YM10 veri seti için CrispMC-EntARMCF ve BaseMCCF yöntemleri kullanılarak kullanıcıya sunulan 5 ürünlük listedeki çeşitlilik oranı sırasıyla %44,3 ve %42,8' dir. Ancak aynı listedeki filmlerin çeşitliliği IFMC-EntARMCF yönteminde %48,2 gibi yüksek rakamlara ulaşmaktadır. Genel sonuçlara bakıldığında, bu performans kriterinin her üç yöntem için küçük dalgalanmalarla değiştiği görülmektedir. Ancak, uygulanan yaklaşımların sonuçlarına bakıldığında IFMC-EntARMCF yönteminin yüksek çeşitlilik oranına sahip olduğu ve kullanıcının beğenebileceği filmler sunduğu sonucuna varılabilir.



Şekil 4.5. *YM10 veri seti için üç yöntemin ürün çeşitlilik sonucu*

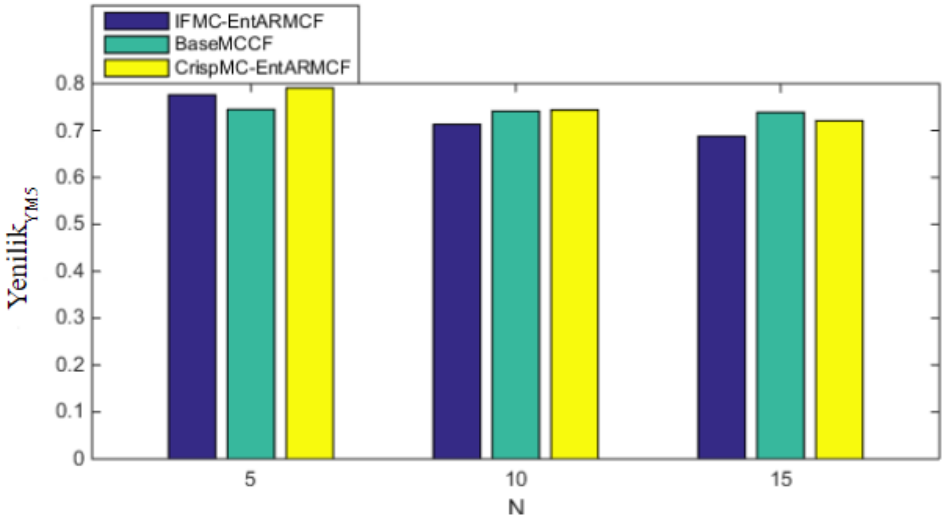
Bu performans kriteri için YM20 veri seti üzerindeki sonuçlar Şekil 4.6' da verilmiştir. Bu sonuçlara göre 5 ürünlük bir liste hazırlanması durumunda BASEMCCF

yöntemi ile kullanıcıya yüksek çeşitliliğe sahip ürünler önerilirken, ürün sayısının artması durumunda bu birinci sırayı IFMC-EntARMCF yöntemi almaktadır. IFMC-EntARMCF yöntemi kullanılarak kullanıcı için on filmlik bir liste hazırlanırsa, bu listedeki filmlerin yüzde 42,3 oranında bir çeşitliliğine sahip olduğu sonucuna varılabilir. Çeşitlilik kriterinde en kötü sonuca sahip yöntem CrispMC-EntARMCF' dir ve bu yaklaşımda performans oranı oldukça düşüktür. Örneğin, bu yönteme göre kullanıcıya on film sunulduğunda yaklaşık iki film kullanıcı için çeşitlilik göstermektedir.



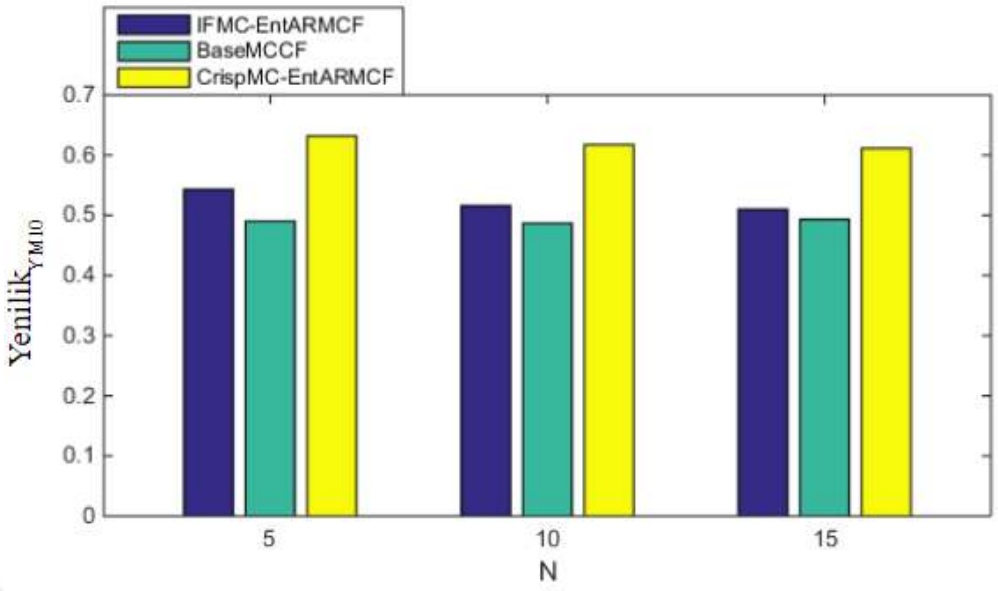
Şekil 4.6. *YM20 veri seti için üç yöntemin ürün çeşitlilik sonucu*

Farklı *N* değerleri ve üç farklı veri kümesi için ortalama yenilik performans karşılaştırması Şekil 4.7 - 4.9' da gösterilmektedir. Sonuçlara göre, tüm yöntemler için kullanıcıya on ürünlük bir liste hazırlandığında bu listedeki 7 ürün kullanıcı için yenidir ve bu oran %70' in üzerindedir.



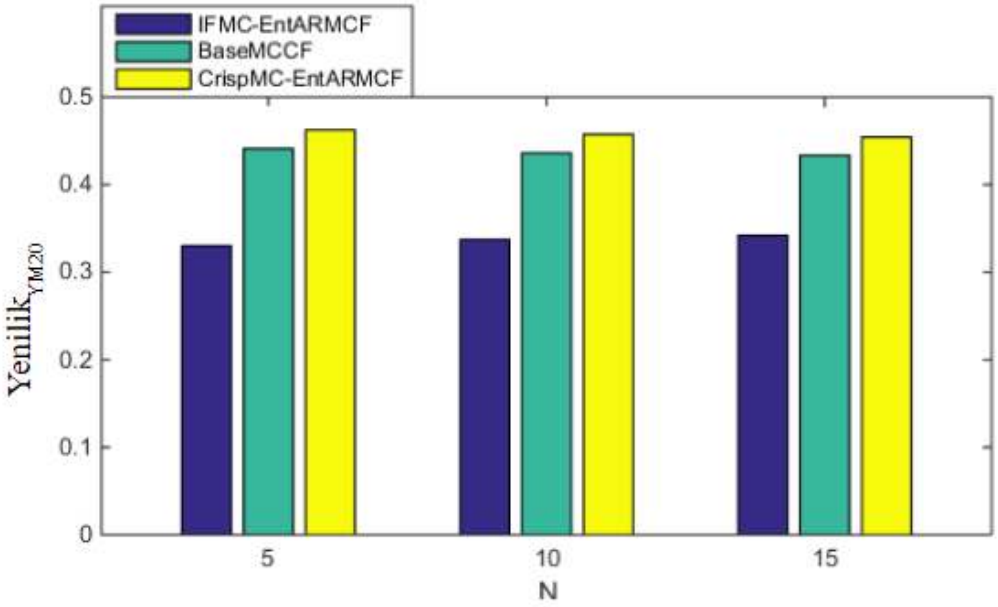
Şekil 4.7. *YM5 veri seti için üç yöntemin ürün yenilik sonucu*

Şekil 4.8' de YM10 veri seti için farklı *N* değerleri için sonuçlar gösterilmiştir. Şekle göre BaseMCCF ile IFMC-EntARMCF yaklaşımları arasında ortalama %5 oranında fark var iken CrispMC-EntARMCF yönteminde bu fark daha büyüktür. Kullanıcılar için 15 filmlik bir öneri listesi üretildiğinde, bu listede kullanıcının izlemediği ve yeni gelen filmlerin oranı CrispMC-EntARMCF yönteminde %63,1 gibi yüksek bir rakam iken, BaseMCCF ve IFMC-EntARMCF yöntemlerinde sırasıyla bu oran %49,3 ve %51' dir. Şekil 4.8' e göre her ne kadar BaseMCCF yöntemi, film yeniliği açısından 3. sırada olmasına rağmen, %45 üzerinde bir orana sahip olduğu sonucuna varılmıştır.



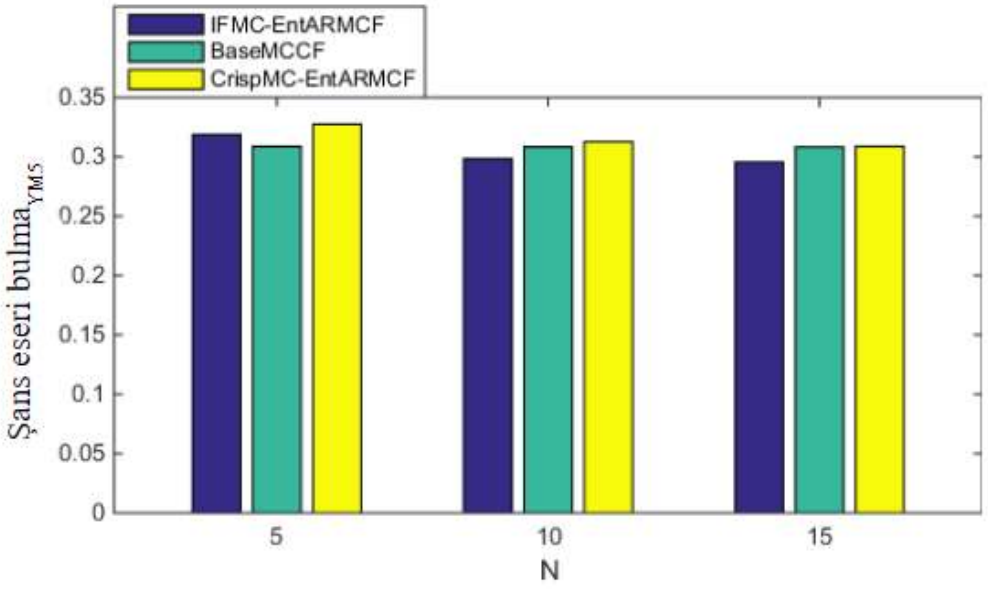
Şekil 4.8. *YM10 veri seti için üç yöntemin ürün yenilik sonucu*

Şekil 4.9' da YM20 veri seti için yenilik performans kriteri değerlendirildiğinde, IFMC-EntARMCF' de önerilen film listesinin yenilik oranının yaklaşık %34 olduğu görülmektedir. Ancak CrispMC-EntARMCF ve BaseMCCF yöntemleri kullanılırsa bu oranın sırasıyla %45-46, %43,44' a kadar çıktığı görülmekte olup, bu oran ile on ürünlük listenin yaklaşık yarısının kullanıcı için yeni olduğu çıkarımı yapılabilir. Bu performans kriterine göre üç farklı veri seti için üç farklı yöntemin analizinde en iyi sonuç CrispMC-EntARMCF yöntemi ile elde edilirken, en kötü sonuç IFMC-EntARMCF yöntemi ile elde edilmektedir.



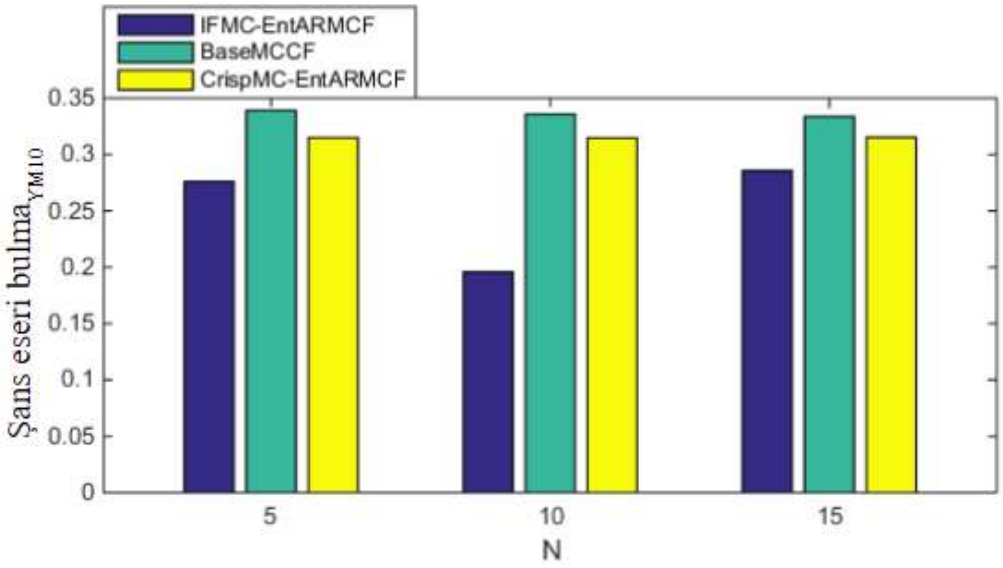
Şekil 4.9. *YM20 veri seti için üç yöntemin ürün yenilik sonucu*

Son performans kriteri olan şans eseri bulma metriğinin üç farklı yaklaşıma göre elde edilen sonuçları aşağıdaki şekillerde gösterilmiştir. İlk veri seti olan, YM5 veri setinin sonuçlarına göre kullanıcılara beş ürün sunulduğunda, kullanıcının henüz keşfetmediği ancak şaşırdığı ürünlerin oranı IFMC-EntARMCF, CrispMC-EntARMCF ve BaseMCCF yöntemleri için sırasıyla %31,8, %32,7, ve %30,8' dir. Kullanıcılara sunulan liste içeriği genişletilirse bu oranın IFMC-EntARMCF ve CrispMC-EntARMCF yöntemleri için %29 ve %30 gibi rakamlara düşmektedir. Sonuç olarak, üç yönteme göre kullanıcılar için önerilen on film arasından yaklaşık üçü, kullanıcının daha önce keşfetmediği ancak memnun olduğu ürünlerdir.



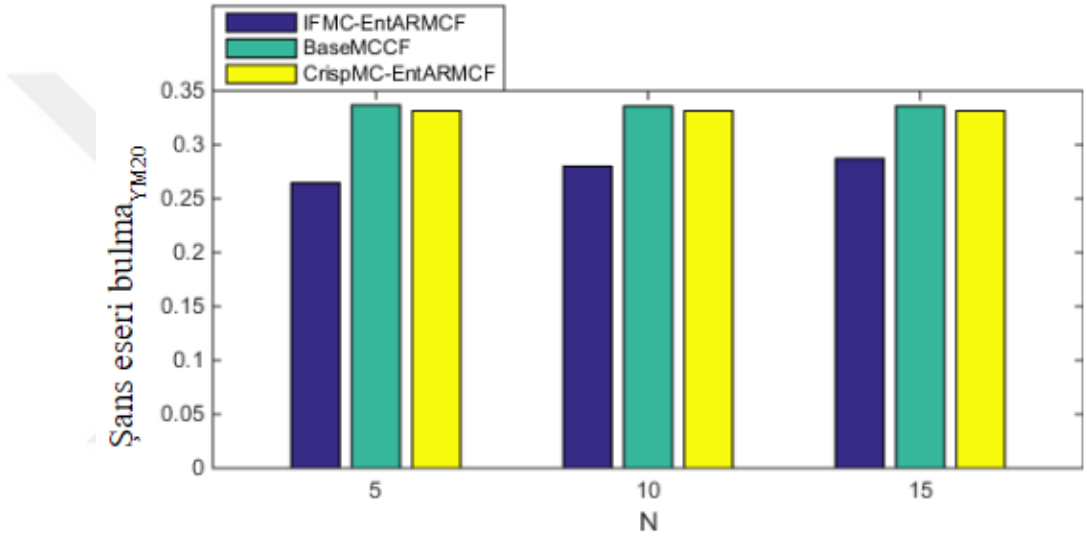
Şekil 4.10. *YM5 veri seti için üç yöntemin ürün şans eseri bulma sonucu*

YM10 veri seti üzerinde nasıl bir sonucun olduğunu görmek için Şekil 4.11 incelenebilir. Şekil 4.11' e göre BaseMCCF yöntemi ile kullanıcılara bir liste hazırlanırsa bu listedeki ürünlerin kullanıcı şaşırtma oranının yüksek olduğu görülmektedir. BaseMCCF yöntemine göre kullanıcıya sunulan on filmden üç tanesi oldukça şaşırtıcı ürün iken; IFMC-EntARMCF ve CrispMC-EntARMCF yöntemlerinde bu sayı sırasıyla 2 ve 3' tür.



Şekil 4.11. *YM10 veri seti için üç yöntemin ürün şans eseri bulma sonucu*

Şekil 4.12 incelendiğinde, YM20 veri seti için üç yöntemin performans sıralamasının Şekil 4.11 ile aynı olduğu görülmektedir. Kullanıcıya sunulan ilk 10 ürünlük listede, ilk yönteme (IFMC-EntARMCF) göre, kullanıcının beklemediği ancak önerilerden memnun kalabileceği yaklaşık iki film olduğu sonucuna varılabilir. BaseMCCF yöntemine bakıldığında aynı boyuttaki liste için heyecan verici ve şaşırtıcı film sayısı yaklaşık üçtür. Ancak buradaki kritik nokta, BaseMCCF ve CrispMC-EntARMCF yöntemleriyle sunulan film listelerindeki ürünlerin kullanıcılar için şaşırtıcı, tesadüfi olma durumları yüksek iken, memnuniyet durumu daha azdır.



Şekil 4.12. *YM20 veri seti için üç yöntemin ürün şans eseri bulma sonucu*

Son performans ölçütü olan, şans eseri bulma metriği için tüm sonuçlar incelendiğinde en iyi sonucun BaseMCCF yönteminde olduğu, en kötü sonucun ise IFMC-EntARMCF yöntemi ile alındığı görülmektedir.

4.4. Değerlendirmeler

Literatürde her ne kadar tek kritere dayalı öneri sistemleri incelenmiş ve sonuçları umut verici olsa da günümüzde çok ölçütlü sistemlerin birçok endüstriyel alan kullanıldığı görülmektedir. Tek kriterli sistemlerde kullanıcı ürünü sadece genel bir bakış açısıyla değerlendirmesi ve buna karşılık, çok ölçütlü değerlendirme sistemleri kullanıcılara çok yönlü değerlendirme imkânı vermektedir. Ancak kişiselleştirilmiş uygulamalarda kullanıcının çok ölçütlü görüşlerinin başarılı bir şekilde analiz edilebilmesi için yeni yönlendirme tekniklerine ihtiyaç duyulmaktadır. Bu nedenle, bu

çalışmada iki yeni yaklaşıma ek olarak, çok ölçütlü sistemlerin uygulandığı ve yöntemlerin karşılaştırıldığı bir kıyaslama algoritma önerilmiştir.

Bu çalışmanın amacı, çok ölçütlü sistemler üzerinden kullanıcı için bir en iyi-n öneri listesi hazırlanmasıdır. Dolayısıyla, bu çalışmada Yahoo! Movies veri setinin alt veri setleri kullanılarak üç farklı yaklaşımın sonuçları incelenmiş ve önerilen sistemlerin kalitesini ölçmek için farklı performans karşılaştırmaları yapılmıştır. Değerlendirme sırasında, kullanıcı özellikleri ve eğilimlerinin yanı sıra ürünler arasındaki ilişkisel yapı ve kullanıcı tereddüt durumları dikkate alınmıştır.

IFMC-EntARMCF ve CrispMC-EntARMCF yöntemleri, komşuluk seçim sürecinde Entropi ve birliktelik kurallarını birleştirerek kullanıcı memnuniyetini en üst düzeye çıkarmayı amaçlamıştır. IFMC-EntARMCF yönteminde, kullanıcıların derecelendirme değerleri SBD' e dönüştürülürken, ikinci yaklaşımda (CrispMC-EntARMCF) bu dönüşüm gerçekleştirilmeden adımlar devam ettirilmiştir. Sonrasında derecelendirme değerlerinin niteliksel sıralarının dikkate alındığı tercih modeli kullanılarak tahmin üretme süreci gerçekleştirilmiştir. Önerilen yaklaşımlar çok ölçütlü veri kümesine uygulandığından, Adomavicius (Adomavicius, Manouselis, & Kwon, Multi-criteria recommender systems, 2011) tarafından benimsenen adımlar takip edilmiş ve kriterleri ağırlıklandırmak için *DU* kullanılmıştır.

Üç yöntemin sonuçları kullanıcı memnuniyeti açısından karşılaştırıldığında, üç veri seti için IFMC-EntARMCF yöntemi diğer yöntemlere kıyasla kullanıcı memnuniyetinde %60 başarı oranıyla ilk sıraya yerleşmektedir. CrispMC-EntARMCF yönteminde bu başarının %40 olduğu görülürken, BaseMCCF yönteminde bu başarının nispeten düşük olduğu görülmektedir. Kullanıcı memnuniyetine ek olarak, listedeki ürünlerin çeşitli, yeni ve tesadüfi olduğu dikkate alınarak çeşitlilik, yenilik ve şans eser bulma hesaplamaları yapılmıştır. Yöntemlerin sonuçları çeşitlilik açısından analiz edildiğinde, IFMC-EntARMCF yöntemi ortalama %45 değişkenlik gösterirken; öte yandan CrispMC-EntARMCF ve BaseMCCF yöntemleri daha iyi sonuçlara sahiptir. Bir diğer performans kriteri olan yenilik, kullanıcının bilmediği ancak kullanıcıya sunulan listeden memnun olduğu ürün oranıdır; IFMC-EntARMCF ve BaseMCCF yöntemlerinde sırasıyla %60 ve %50 başarı oranı elde edilirken, CrispMC-EntARMCF yönteminde bu değer en yüksek seviyeye maksimum %70' e ulaşmaktadır. Son performans kriteri olan şans eseri bulmada BaseMCCF yönteminin başarısı diğer yöntemlere göre daha yüksektir. Kullanıcı için

hazırlanan liste, BaseMCCF yöntemine göre kullanıcının aşina olmadığı ürün oranı %30' dur. Yukarıdaki sonuçlar incelendiğinde, üç yaklaşım farklı durumlarda yüksek başarı elde etmiştir. Sonuç olarak, kullanıcı özelliklerinin, eğilimlerinin ve yapısal özelliklerinin iyi bir analizinin sonuçları iyileştirdiği, önerilen yaklaşımlarla gösterilmiştir. Bu iyileşme ile Booking.com ve TripAdvisor gibi büyük firmalar bu yaklaşımlarla müşteri memnuniyetini arttıracak ve pazarda üst sıralara çıkmalarını ve sektörde rekabet avantajı elde etmelerini sağlayacaktır.

5. ÇOK ÖLÇÜTLÜ EN İYİ-N İŞBİRLİKÇİ ÖNERİ SİSTEMLERİNİN GÜRBÜZLÜK ANALİZİ

Bu bölümde, bir önceki bölümde önerilen çok ölçütlü en iyi-n öneri sistemlerinin (Kaya & Kaleli, 2022), kötü niyetli kullanıcılar tarafından gerçekleştirilecek saldırılara karşı ne derece gürbüz olduğunun analizi yapılmıştır.

5.1. Giriş

Son yıllarda bilim ve teknolojinin hızla ilerlemesi, kullanıcılar hakkında detaylı verilerin toplanmasına yol açmıştır. Kullanıcılar tarafından ürünler hakkında yapılan yorumlar, oylar ve kullanıcıların çevrim içi sistemlerde verdiği geri bildirimler bu büyük miktardaki veriyi oluşturmaktadır. Öneri sistemleri, farklı kullanıcılara ait heterojen kaynaklardan gelen bu verileri analiz eder ve kullanıcıların ürün seçim sürecinde fazla zaman harcamasını engelleyerek doğru seçimi yapmalarına yardımcı olur. En popüler öneri sistemi tekniklerinden biri olan İF yönteminde, hedef kullanıcıya benzer davranışlar sergileyen diğer kullanıcıların tercihleri analiz edilir ve kullanıcının en çok beğeneceği bir en iyi-n öneri listesi hazırlanır. Geleneksel İF yöntemlerinde müşteriler, satın alınan ürünler için belirli bir alandan genel bir puan verir. Ancak tavsiyelerin başarısı için müşterilerden alınan geri bildirimlerin değerli olduğu düşünüldüğünde, daha detaylı tercihlerin sistemlerin doğruluğunu artıracağı sonucuna varılabilir. Bu nedenle, bazı bölgelerdeki müşterilerin artık daha ayrıntılı geri bildirim sağlamalarına, ürün ve hizmetleri çeşitli boyutlarda derecelendirmelerine izin verilmektedir. Örneğin, turizm alanındaki çoğu büyük otel rezervasyon platformunda müşteriler, temizlik, kahvaltı servisi, paranın karşılığı veya personelin güler yüzlülüğü gibi ölçütlerle otelleri derecelendirebilir. Son zamanlarda, Amazon.com video oyunları için bir derecelendirme boyutu getirmiş ve eBay platformu da artık müşterilerin, satıcıları çeşitli boyutlarda derecelendirmesine olanak tanıyor. Başka bir örnek ise, Yahoo! Movies platformu da kullanıcılarına filmlerin oyunculuk, senaryo, yönetmenlik ve görsellik gibi alt özelliklerini ve genel değerlendirmeyi dikkate alan çok yönlü bir değerlendirme sunmaktadır. Bu çok yönlü değerlendirmeyi kullanan İF yöntemlerine ÇÖİF adlandırıldığını önceki bölümlerde ifade etmiştik.

İF yöntemlerini (ÇÖİF dahil) kullanan şirketler, müşterilerini hayal kırıklığına uğratmamak için kullanıcılarına yüksek kaliteli öneriler (tavsiye listeleri) sunmak zorundadırlar. Kullanıcılara sunulan öneri listesine yanlış ürünleri dahil etmek,

müşterilerin başka e-ticaret sitelerine yönelmesine neden olabilir. Bu durum firmaların müşteri kaybetmesine ve rekabet ortamında alt sıralara yerleşmesine neden olabilmektedir. Böyle bir senaryonun gerçekleşmesini isteyen kötü niyetli kişi veya işletmeler, sunulan bu hizmetleri İF yöntemlerine göre kendi lehlerine yönlendirmeye çalışabilirler. Bazı ürünleri hedef ürün olarak seçip popülerliğini arttırmak veya azaltmak isteyebilirler. Dolayısıyla bu amaçları gerçekleştirmek için sahte kullanıcıları sisteme ekleyebilirler. Bu tür saldırılara "şilin saldırıları" veya "profil enjeksiyon saldırıları" denir (Lam & Riedl, 2004).

İF yöntemleri, açık yapıları nedeniyle şilin saldırılarına karşı savunmasızdır. Kullanıcı tabanlı İF algoritmaları, birçok farklı bireyin tercihlerini temsil eden kullanıcı profillerini toplar, kullanıcı komşuluklarını oluşturur ve öneriler üretir. Doğrudan kullanıcılar tarafından sağlanan veriler üzerinde çalıştıkları için bu durum, kötü niyetli kullanıcılara karşı güvenlik açığına neden olur. İF' nin bu güvenlik açığı, bu öneri algoritmalarının zayıflıklarını daha iyi anlamak için bazı son çalışmalar yapılmıştır. Bu çalışmalara göre, kullanıcı bazlı İF yönteminde yüksek bilgili saldırı modelleri oldukça başarılı olabilirken, ürün tabanlı olanlarında oldukça sağlam, gürbüz olduğu görülmüştür.

Ürün tabanlı İF sistemlerine karşı başarılı saldırılar gerçekleştirmek için, Seminario ve Wilson (Seminario, 2013) tarafından güçlü ürün atak (PIA) modeli adı verilen yeni bir model önerilmiştir. PIA' da benimsenen yaklaşım, güçlü ürünlerin İF' de bulunabileceği ve bunların geniş bir ürün grubu üzerinde çok yüksek bir etkiye sahip olabileceğidir. Bu etki, güçlü ürün *i*' nin, başka bir *j* ürünü için tahmin sürecini olumlu veya olumsuz yönde değiştirebilir veya komşuluk sürecine dahil olarak en iyi-n listesini manipüle edebilir. Örneğin, Amazon Vine (https://www.amazon.com/gp/vine/help), satın alan müşterilerinin bilinçli bir karar vermesine yardımcı olmak için Amazon' daki en güvenilir yorumcuları yeni ve yayın öncesi ürünler hakkında görüşlerini sunmaya davet etmektedir. Ancak bu çalışmalar geleneksel İF' ye uygulanırken, çok ölçütlü en iyi-n öneri sistemi yöntemlerinin sağlamlığını analiz eden bir çalışmanın henüz olmadığı görülmektedir.

Üstelik bu saldırı modelinin çok ölçütlü en iyi-n algoritması üzerinde nasıl tasarlanacağı da cevaplanması gereken bir sorudur. Bu nedenle, bu bölümde, ürün tabanlı çok ölçütlü en iyi-n öneri sistemi algoritmalarının saldırılara karşı ne kadar sağlam olduğunun analizi PIA modeli kullanılarak yapılmıştır. Gerçekleştirilen saldırıların birincil amacı, hedef ürünlerin en iyi-n içinde yer almasını sağlamak ve bunları

kullanıcıya bir öneri olarak sunmaktır. Bu niyete (*itme)* sahip bir saldırgan, hedef ürünlerin tahmini değerlerini arttıracak şekilde manipüle edecek ve sonuç olarak değerleri değişen bu ürünler kullanıcıya sunulan öneri listesi içerisinde yer alacaktır.

5.2. Önerilen Yaklaşım: Çok Ölçütlü En İyi-n İşbirlikçi Öneri Sistemlerine Şilin Saldırıların Tasarlanması

Kullanıcıya sunulan öneri listesine yapılan saldırıda, saldırgan, öneri listesindeki ürünleri manipüle etmeyi ve gerçek ürünlerin yerine sahte ürünlerin listeye dahil edilmesini sağlamayı amaçlar. Dolayısıyla bu amaçla oluşturulacak saldırı metodolojisinde, saldırganın sisteme *m* sahte kullanıcı profilleri eklemek istediğini ve her saldırı kullanıcısının, kullanıcının beğenmeyeceği veya popüler olmayan hedef ürünü değerlendireceğini varsayıyoruz. Böylece yanlış ürünlerin listeye alınması hedeflendiğinden saldırı niyeti itmedir (pushing). Ancak buradaki kritik noktalardan biri, saldırıya uğrayacak en iyi-n öneri sistemlerinin ürün tabanlı olması ve bu sistemlerin, kullanıcı tabanlı sistemlere ve bilinen diğer saldırı türlerini kullanan sistemlere kıyasla itme saldırılarına karşı oldukça sağlam olmasıdır. Bir diğer kritik nokta ise çok ölçütlü en iyi-n İF sistemleri için saldırı tasarımının nasıl yapılması gerektiğinin ve birden çok ölçütten oluşan sistemde saldırıya özel parametrik değerlerinin nasıl belirlenmesini gerektiğinin, karar verilmesi gerekmektedir.

Bu başlıkta ilk olarak kullanılan saldırı modelini, saldırı tasarımını ve çok ölçütlü bir sistemde saldırıya özgü parametrelerin nasıl belirleneceğini tartışacağız.

5.2.1. PIA atak modeli

Seminario ve Wilson (Seminario & Wilson, 2014) tarafından önerilen PIA, güçlü ürünler içerdiği ve bu ürünler komşuluk sürecinde yüksek etkiye sahip olduğu için literatürde bilinen diğer saldırı türlerine kıyasla ürün bazlı öneri sistemlerine yapılan saldırılarda oldukça başarılı olduğu gözlemlenmiştir. Ayrıca modelin tespit edilmesi zor olsa da, birden fazla hedef ürün kullanıldığında PIA modelinin, tek hedefli PIA' ya göre daha etkili bir itme saldırısı olduğu ve modelin PIA-çoklu hedefler (PIA-multiple targets, PIA-MT) olarak adlandırıldığı görülmüştür.

PIA-MT modeline uygun olarak sisteme eklenecek saldırgan kullanıcı profilinin içeriği Şekil 5.1' de gösterilmektedir.

I_S	I_F	I_U	I_T
Güçlü Ürünler: ürün ortalaması etrafında normal dağılımlı oy değerleri	Boş	Boş	Çoklu Hedef Ürünler: itme atak için r_{max}

Şekil 5.1. *PIA-MT atak kullanıcı profilinin içeriği*

Verilen şekle göre PIA-MT atak modelindeki sahte kullanıcı profillerini oluşturan elementler aşağıda açıklanmıştır:

- **Seçilen Ürün Kümesi (Is):** Saldırgan tarafından belirlenen özel karakteristikti ürünleri içerir. Bu küme güçlü ürünlerden oluşur ve bu güçlü ürünlere, ortalama oy değerleri atanır. PIA atak modelinde, güçlü ürünlerin seçimi için literatürde üç farklı yöntem yer almaktadır. Bu bölümde yeni bir seçim modeli de önerilmiştir. Toplamda güçlü ürünler dört farklı kritere göre seçilecektir. Tüm seçim kriterleri aşağıda ayrıntılı olarak açıklanmıştır:
  - *Derece (InDegree, ID):* Güçlü ürünler, en fazla sayıda benzer komşuluklara katılanlardır. Her *i* ürünü için, *j* ürünün her biriyle olan benzerliği hesaplanır ve ardından her bir *i* ürünü için en iyi-n komşulukları dışındakiler göz ardı edilir. Her j ürünü için benzerlik puanlarının sayısı bulunur ve ilk *n* ürün *j*' ler seçilir. (Seminario & Wilson, 2014) (Seminario & Wilson, 2016).
  - *Toplu Benzerlik (Aggregate Similarity, AS):* En yüksek toplam benzerlik puanlarına sahip en iyi *n* ürünler, güçlü öğeler kümesi olarak seçilir (Seminario & Wilson, 2014) (Seminario & Wilson, 2016).
  - *Derecelendirme Sayısı (Number of Rating, NR):* Güçlü ürünler, popüler ürünler gibi en yüksek sayıda kullanıcı derecelendirmesine sahip öğelerdir. En iyi *n* ürünleri, profillerinde sahip oldukları toplam kullanıcı derecelendirmesine göre seçilir (Seminario & Wilson, 2014) (Seminario & Wilson, 2016).
  - *Benzerlik Oranı (Sim Ratio, SR):* Pozitif benzerliklerin toplamının negatif mutlak benzerlik değerlerinin toplamına bölümüdür (Turkoglu, 2016). En yüksek benzerlik oranı değerine (simRatio) sahip *n* ürün, güçlü ürün kümesini oluşturur. Önerilen bu yeni metrik aşağıdaki denklem kullanılarak hesaplanır.

$$simRatio_i = \frac{\sum_{j \in sim^+} sim_{i,j}}{\sum_{j \in sim^-} |sim_{i,j}|} \quad (5.1)$$

Burada, sim^+, pozitif benzerliklere sahip ürünleri, sim^- negatif benzerliklere sahip ürünleri temsil eder.

- **Dolgu Ürün Kümesi (Filler Items, I_F):** I_S ile hedef ürün arasında güçlü bir korelasyon istendiğinden PIA-MT modeli için bu küme boştur.
- **Derecelendirilmemiş Ürün Kümesi (Unrated Items, I_U):** Bu küme, I_S, I_F ve hedef ürünler dışındaki ürünlerdir ve boş değer atanır.
- **Hedef Ürün Kümesi (Target Items, I_T):** Hedef ürün derecelendirme değeri, saldırının amacına bağlı olarak maksimum veya minimum derecelendirme değerini alır. Seminario ve Wilson (Seminario & Wilson, 2014) tarafından yapılan çalışmalarda, saldırının başarısı için hedef ürün seçiminin önemli olduğu görülmüştür. Bu nedenle hedef ürün seçimi üç farklı yöntemle yapılmaktadır.
  - *En Popüler Olmayan- Genel (MUP-G):* Sistemde derecelendirme sayısı ve ortalama derecelendirmesi en düşük olan ürünler seçilir.
  - *En Popüler Olmayan- Kullanıcı (MUP-U):* Kullanıcının ortalama oy değeri bulunur ve bu ortalama değerden küçük oy değerine sahip kullanıcı ürünleri seçilir.
  - *En Popüler Olmayan- Kriter (MUP-C):* Kriterler açısından popüler olmayan ürünleri belirlemek için yeni bir yöntem (Toplam Öncelik Derecesi, TOD) önerilmiştir. Öncelikle aktif kullanıcının ve ürünlerin kriter öncelikleri belirlenir. Daha sonra aktif kullanıcının kriter önceliğine aykırı ürünler aşağıdaki eşitlik kullanılarak seçilir.

$$TOD_u = \sum_{c \in Kriter} \frac{KriterDeğeri_{(aktif\ kullanıcı,c)}}{Sıra_{(u,c)}} \quad (5.2)$$

$$TOD_i = \sum_{c \in Kriter} \frac{KriterDeğeri_{(u_i,c)}}{Sıra_{(i,c)}} \quad (5.3)$$

$$Hedef\ \text{Ürünler}_u = seç_{j \in \text{ürünler}}^{\max\ (TOD_u - TOD_j)} \quad (5.4)$$

Belirtilen hedef ürün seçim kriterini bir örnekle açıklayalım:

Kullanıcı veri kümesinin 4 alt kriterden (*S, A, D, V*) ve bir genel derecelendirmeden oluştuğu varsayılsın. Aktif kullanıcı *u* için matematiksel yöntemlerle elde edilen kriterlerin öncelik sırası *A, D, V, S* iken, i_1 ve i_2 ürünleri için bu öncelik sırası *V, S, A, D* ve *D, A, S, V* olsun. Burada, aktif kullanıcının kriter değeri *S=1, A=4, D=3 ve V=2* olarak bulunur. Aktif kullanıcı için toplam öncelik değeri;

$TOD_u = (\frac{4}{1} + \frac{3}{2} + \frac{2}{3} + \frac{1}{4}) \approx 6.35$ olarak bulunur.

Sistemde yer alan iki ürün için toplam öncelik değeri ise;

$TOD_{i_1} = (\frac{2}{1} + \frac{1}{2} + \frac{4}{3} + \frac{3}{4}) \approx 4.5$

$TOD_{i_2} = (\frac{3}{1} + \frac{4}{2} + \frac{1}{3} + \frac{2}{4}) \approx 5.8$ şeklinde bulunur.

Aktif kullanıcı ve ürünlerin belirlenen toplam öncelik değerlerinin farkları bulunur. En fazla farka sahip olan ürünler, hedef ürün olarak seçilir.

$$Hedef\ \text{Ürün}ler_u = se\text{ç}_{j \in \text{ürün}ler}^{\max\,(TOD_u - TOD_{i_1}, TOD_u - TOD_{i_2})}$$

Bu durumda aktif kullanıcının i_1 ve i_2 ürünleri arasındaki fark sırasıyla 1,85 ve 0,55'tir. Belirlenen seçim kriter doğrultusunda, aktif kullanıcı ile ters düşen ürün i_1' dir ve bir hedef ürün seçilmesi gerektiğinde bu ürün i_1 olacaktır.

Saldırı kullanıcı profillerinin parçası olan güçlü ürünler, Şekil 5.1'de gösterildiği gibi profilde sunulan hedef ürünlerle ilişkilendirilir. Daha sonra PIA-MT atak modeli, kullanıcıya sunulan öneri listesini etkilemek ve veri kümesindeki diğer ürünlerle ilişki kurmak için saldırı kullanıcı profillerinde sunulan hedef ve güçlü ürün kombinasyonlarını kullanır.

5.2.2. Atak özel-parametrelerin dizaynı

Tek kriterli sistemlerde, tüm derecelendirme değerlerinden yararlanılarak saldırı profili için gerekli istatistiksel parametreler üretilir. Bununla birlikte, çok ölçütlü sistemlerde, birden fazla kriterin dahil edilmesi, bu saldırıya özgü özelliklerin birden fazla şekilde tahmin edilmesine izin verebilir. Dolayısıyla, çok ölçütlü sistemde bu hesaplama için aşağıdaki teknik kullanılır:

Her kritere özel saldırı parametreleri: Saldırı profili için gerekli istatistiksel parametreler her bir kriter için ayrı ayrı hesaplanır ve bu değerler saldırı profilinin ilgili

bölümlerinde bağımsız olarak kullanılır. Dolayısıyla parametre değerleri kriter sayısına eşit olarak üretilmektedir. Böyle bir yaklaşımı benimsemek, mevcut tüm bilgilerin sezgisel olarak daha iyi kullanılması, veri kaybı olmaması ve her bir alt kriter için oluşturulan sahte profillerin gerçek kullanıcılara benzeme olasılığının daha yüksek olmasıdır.

İstatistiksel parametrelere ek olarak, PIA saldırı modeli için gerekli olan güçlü ve hedef ürünlerin seçiminin yapılması gerekmektedir. Bu ürünlerin seçim yöntemleri önceki bölümlerde ifade edilmiştir. Bu bölümde çoklu sistemler üzerinden bu seçim adımlarının nasıl yapılması gerektiği açıklanmıştır:

Her kriter üzerinden seçim: Güçlü ve hedef ürünlerin seçimi, her bir kriter için ayrı ayrı belirlenir. Sistemin bu şekilde oluşturulması, güçlü olduğu belirlenen ve kullanıcıları etkileyebilecek ürünlerin her bir kriteri için farklılaşma durumunun olmasıdır. Örneğin gerçek kullanıcılar bir ürünü birden fazla sistemde değerlendirmek istediğinde kritik kriterler olabilir ve bu kriterlerin başarısı o ürünü o kullanıcılar için güçlü kılabilir. Bunun tersi de söz konusu olabilir. Dolayısıyla böyle bir durum, bu seçim işlemlerini ayrı ayrı gerçekleştirerek saldırının başarısını olumlu yönde etkileyebilir.

5.2.3. Çok ölçütlü en iyi-n öneri sistemleri için atak metodolojisi

Bu bölümde, en iyi-n manipülasyonunu gerçekleştirmek için kullanılan metodolojinin adımları verilmiştir.

Şekil 5.2' deki algoritmanın adımları incelendiğinde, mevcut tüm bilgileri kapsüllenmiş bir şekilde kullanmak sonuç üretmenin sezgisel olarak en iyi yoludur. Her bir alt kriter için ayrı ayrı değerler, sahte profillerin gerçek kullanıcılara benzemesine yardımcı olabileceğinden (Turk & Bilge, 2019), saldırı profilleri ekleme, saldırı sürecini başlatma ve tahmin oluşturma gibi Adım 11' e kadar olan tüm işlemler her bir kriter için ayrı ayrı gerçekleştirilir. Daha sonra, her bir kriter için üretilen tahmini değerler, çok ölçütlü en iyi-n öneri sistemi algoritmalarına göre toplanır. Bu tek sonuçla kullanıcı için bir öneri listesi hazırlanır ve saldırının başarısı değerlendirilir.

Girdi: Çok ölçütlü veri seti ($D_{n\times m}$)

Çıktı: Üst-n öneri listesi, Performans sonuçları

1 **for each** C_t kriterleri **do**

2 k-fold kullanarak veri seti prob P ve eğitim setine M böl

3 M veri setindeki güçlü ve hedef ürünleri belirle

4 Güçlü ve hedef ürünleri kullanarak sahte güçlü kullanıcı profillerini oluştur ve bunları M veri setine ekle

5 P veri setinden u kullanıcısının oy vermediği 1000 ürün seç

6 n adet çoklu hedef ürün seç; n, u kullanıcısına sunulan listedeki ürün sayısıdır

7 T test seti oluşturmak için oy verilmemiş ürünlere n çoklu hedef ürünler ekle

8 Atak sürecini başlat

9 1000+n ürün için tahmini değer üret

10 **end for**

11 Üst-n öneri sistemi algoritmalarına dayanarak 1000+n ürün için genel tahmini değeri üret

12 Sıralı listeyi kullanarak u kullanıcı için üst-n öneri listesi hazırla

13 Hedef ürünlerin sayısını ve bu listeye dahil edilme sırasını belirle

Şekil 5.2. *Çok ölçütlü en iyi-n öneri sistemlerinin atak metodolojisi*

Sentetik kullanıcı profilleri, yapılacak saldırının amacı (bu durumda itme) doğrultusunda runtime içerisinde aynı anda birden fazla hedef ürüne uygulanırken, Şekil 5.1' de gösterildiği gibi güçlü (ID, AS, NR, SR) ve hedef ürünlerden (MUP-G, MUP-U, MUP-C) oluşmaktadır.

5.3. Deneysel Değerlendirmeler

Bu bölümde, BaseMCCF, IFMC-EntARMCF ve CrispMC-EntARMCF en iyi-n öneri algoritmalarının sağlamlığını analiz etmek için bir dizi deney gerçekleştirilir. Öncelikle çalışmanın deneysel tasarımı hakkında bilgi verilmiştir. Daha sonra performans değerlendirme kriterlerinden ve deney sonuçlarından bahsedilmiştir.

5.3.1. Değerlendirme kriterleri

Gürbüzlük metrikleri, öneri sistemlerine gerçekleştirilen saldırıları değerlendirmek için geliştirilmiştir (Mobasher, Burke, Bhaumik, & Williams, 2007) (Burke, O'Mahony, & Hurley, 2015). Örneğin Ortalama İsabet Oranı (Average Hit Ratio, *AvgHR*), en iyi-n listesinde hedef ürünleri yer alan kullanıcı sayıların yüzdesini hesaplarken, önerilen yeni bir performans metriği olan Ağırlıklı Sıra Değeri (Weighted Rank Value, *WRV*), hedef ürünlerin en iyi-n listesine dahil edilme sırasını gösterir.

Ortalama İsabet Oranı (AvgHR): Kullanıcı için oluşturulan en iyi-n öneri listesinde yer alan birden çok hedef ürünün yüzdesidir. Bu metrikte çoklu hedef ürünlerin liste içerisinde yer alma sayıları, liste boyutuna bölünerek, kullanıcının isabet oranı bulunur. Her bir kullanıcı için bulunan bu oranın ortalaması *AvgHR* verir. Denklem 5.6 kullanılarak hesaplanan *AvgHR* metriğinde yüksek değer elde edilmesi saldırının oldukça etkili olduğunu gösterir.

$$\text{İ}sabet\ Oranı_u = \frac{\sum_{i \in IT_T} H_{u,i}}{n} \tag{5.5}$$

$$AvgHR = \frac{\sum_{i \in U_T} \text{İ}sabet\ Oranı_u}{|U_T|} \tag{5.6}$$

Burada IT_T ve U_T sırasıyla hedef ürünler ve kullanıcılar kümesidir. *n* kullanıcıya sunulan öneri listesi boyutudur.

Ağırlıklandırılmış Sıra Değeri (*WRV*): Bu çalışmada yeni bir gürbüzlük metriği önerilmektedir. Kullanıcı için öneri listesi hazırlanırken ilk sırada yer alan ürünler, kullanıcı için yüksek önceliğe sahip ürünler olabilir. Bu nedenle, saldırının başarısı, hedef ürünlerin en iyi-n listesinde hangi sırada dahil edildiği ile de ifade edilebilir. Bu sebeple, Denklem 5.7 kullanılarak hesaplanan bu yeni metrikte, yüksek bir *WRV* elde etmek, saldırının oldukça etkili olduğunu gösterir.

$$Derece_u = \sum_{i \in IT_T} \frac{n}{Sıra_{u,i}}$$

$$WRV = \frac{\sum_{i \in U_T} Derece_u}{|U_T|} \tag{5.7}$$

Önerilen ölçüm metriğini bir örnekle açıklamak gerekirse;

Sistemde saldırıya uğrayacak hedef ürünlerin i_{12}, i_{45}, i_{23}, i_{65}, i_{102} olduğunu ve saldırıdan sonra *u* kullanıcısına sunulan beş ürünün listesinin i_{12}, i_{56}, i_{78}, i_{102}, i_{65} sırada

olduğunu varsayalım. Saldırı modelinin WRV başarısını ölçmek istersek, hedef ürünlerden hangilerinin ilk 5' te yer aldığı incelenir. Liste incelendiğinde ilk beşte i_{12}, i_{102} ve i_{65} ürünlerinin olduğu ve bu ürünlerin 1., 4. ve 5. sırada yer aldığı görülmektedir. Dolayısıyla, u kullanıcısının bu listeye göre, $Derece_u = \frac{5}{1} + \frac{5}{4} + \frac{5}{5} = 7.25$ olur. Diğer tüm kullanıcılar için bu hesaplamaların ortalaması ile bulunan sonuç WRV değeri olacaktır.

Bu yeni metrikle elde edilen değerlerin aralığı, sırasıyla en iyi-5, 10 ve 15 için 0-11.4167, 0-29.2897, 0-49.7734 arasındadır.

5.3.2. Deney dizaynı

Deneylerimiz, önceki bölümlerde verilen deney metodolojisine dayalı olarak, veri kümesi 10-fold kullanarak test ve eğitim olarak ayrılır. Eğitim veri seti kullanılarak elde edilen güçlü ve hedef ürünleri ile PIA-ID, PIA-AS, PIA-NR ve PIA-SR olmak üzere dört farklı saldırı modeli kullanılarak saldırı profilleri oluşturulur. Eğitim veri seti, gerçek kullanıcı profillerine saldırı profillerinin de dahil edilmesiyle yeniden şekillendirilir. Ardından, kullanıcının test veri kümesinden derecelendirilmediği 1000 ürüne, kullanıcıya sunulacak listedeki ürün sayısı kadar 5, 10 ve 15 gibi hedef ürünler eklenerek tahmini değerler üretilir. 1000+*n* ürün listesinden *u* kullanıcısına en çok beğenebilecekleri *n* üründen oluşan bir liste sunulur ve performans değerlendirmesi yapılır.

Bu şema için gerekli parametre seçimleri aşağıda verilmiştir:

Sahte Kullanıcı Profilleri: Sisteme eklenecek sentetik kullanıcı profilleri, PIA-MT modeline göre I_S ve I_T' den oluşur. Bu ürünlerin nasıl seçileceği önceki bölümlerde bahsedilmiştir. Saldırının amacı (itme saldırısı), hedef ürünleri en iyi-n içerisine dahil etmek olduğundan, hedef ürünlerin derecelendirme değeri r_{max} (=5) olarak belirlenir. Sisteme ne kadar sentetik profil ekleneceği ise her veri seti için değişir. Önceki çalışmalar, bu oranın %5 ile %10 arasında seçilmesinin çok etkili olduğunu göstermiştir (Mobasher, Burke, Bhaumik, & Williams, 2007) (Burke, O'Mahony, & Hurley, 2015). Ancak bu çalışmada, saldırı kullanıcı profili 0,1 olarak belirlenmiştir. Çünkü saldırı boyutundaki değişiklik, saldırının maliyetini/faydasını etkilediği ve %10'dan büyük bir değerin, gerçek dünya uygulamalarında uygun görülmediği çalışmalarla ifade edilmiştir (Batmaz, Yilmazel, & Kaleli, 2020). Dolayısıyla bu çalışmada, sahte kullanıcı sayısı YM10 ve YM20 için sırasıyla 183, 43 (%10) seçilmiştir.

Güçlü Ürün Seçimi: Güçlü ürünlerinin seçimi için kullanılan dört farklı yöntem önceki bölümde verilmiştir (Bölüm 5.2.1). Sentetik kullanıcı profiline dahil edilen bu güçlü ürünlerin sayısı, her veri kümesi için farklıdır. YM10 veri kümesi için sentetik kullanıcı profilindeki güçlü ürün sayısı 15 (veri kümesindeki ürün sayısının %1), 74 (%5), 147 (%10) iken, bu değerler YM20 veri kümesi için sırasıyla 4, 21 ve 43' tür.

Hedef Ürünlerin Seçimi: Hedef ürünlerin üç farklı yöntemle (MUP-G, MUP-U ve MUP-C) seçildiği önceki bölümlerde belirtilmiştir. Bu seçim kriterlerinin ilki olan ve az bilgi gerektiren MUP-G yönteminde tüm ürünlerin ortalaması ve reyting sayıları bulunmaktadır. Ortalaması dörtten az olan ürünler içerisinden en düşük oy sayısına sahip ürünler seçilir. Diğer yöntemde ise fazla bilgi gerektiren MUP-U yöntemi, kullanıcının ortalama reyting değerinin altında kalan ürünler seçilmektedir. Ancak burada kritik nokta, seçim sürecinin kullanıcı tarafından derecelendirilen ürünler dikkate alınarak yapılmasıdır. Son yöntemde ise yüksek bilgi olan MUP-C, aktif kullanıcının ve ürünlerin kriter öncelikleri belirlenir ve bu önceliğe ters düşen ürünler kullanıcı tarafından popüler olmayan ürünler olur.

Bu üç yönteme göre belirlenen ilk 50 üründen seçilecek hedef ürün sayısı, kullanıcıya sunulacak en iyi-n öneri listesindeki *n* ürün sayısı kadar olacaktır. Bu nedenle, bu çalışmada 50 üründen rastgele 5, 10 ve 15 adet çoklu hedef ürün seçilmiştir.

Test Varyasyonları: Bu çalışmada, 2 veri setinde (YM10, YM20), 4 güçlü ürün seçim yöntemi, 3 güçlü ürün boyutu, 3 hedef ürün seçim türü, 3 hedef ürün boyutu, 1 saldırı boyutu ve 3 farklı algoritmalar (IFMC-EntARMCF, BaseMCCF, CrispMC-EntARMCF) kullanılmıştır. Çok ölçütlü veri setindeki her bir kriter için seçim yöntemleri ve oluşturulan saldırı kullanıcı profilleri, Adomavicius vd. (Adomavicius, Manouselis, & Kwon, 2011) tarafından benimsenen yaklaşım adımları takip edilerek gerçekleştirilir.

5.3.3. Deney sonuçları

Bu bölümde, önceki bölümde önerilen çok ölçütlü en iyi-n öneri algoritmalarının gürbüzlük analiz sonuçlarına yer verilmiştir.

İlk olarak çalışmada deneyin ilk adımında veri seti, 10-fold yöntem kullanılarak test ve eğitim setlerine ayrılmıştır. Burada test seti, rastgele seçilmiş 1000 derecelendirilmemiş ürün ve 5, 10, 15 çoklu hedef üründen oluşur. Bir sonraki aşama olan öneri oluşturma sürecinde, 1000+*n* ürün kullanılarak sisteme dahil edilen sahte profiller

ile kullanıcı için bir liste hazırlanır. Sentetik kullanıcı profilleri oluşturmak için 4 farklı güçlü (ID, AS, NR, SR) ve üç farklı hedef (MUP-G, MUP-U, MUP-C) ürün seçim yöntemine göre belirlenen güçlü ve hedef ürünler kullanılmaktadır. Daha sonra bu sahte profiller iki veri setinin (YM10, YM20) her birine eklenir ve saldırı süreci başlatılır. Sistemdeki tüm kullanıcılar için çalıştırılan bu süreçlerde, önerilen sistemlerin saldırı başarısını ve gürbüzlük analizini *AvgHR*, *WRV*, *çeşitlilik*, *yenilik* ve *şans eseri bulma* kriterleri kullanılarak hesaplanır.

Tablo 5.1. *BASEMCCF algoritmasının ID seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM10		**ID 0,01**			**ID 0,05**			**ID 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	**0,0577**	**0,11**	**0,1459**	**0,0607**	**0,1149**	**0,1463**	**0,0671**	**0,1139**	**0,1454**
	WRV	**0,6239**	**2,8129**	**6,5228**	**0,6334**	**3,053**	**6,6135**	**0,7278**	**2,9676**	**6,2575**
	Çeşitlilik	**0,4393**	**0,4311**	**0,427**	0,4403	**0,4301**	**0,4271**	**0,4395**	**0,4308**	**0,4273**
	Yenilik	0,6496	0,6413	0,6357	0,6504	0,641	0,6355	0,6484	0,641	0,6368
	Şans eseri bulma	0,3254	0,2817	0,2604	0,3237	0,2809	0,2609	0,3231	0,2816	0,2613
MUP-U	AvgHR	0,0394	0,0741	0,0989	0,051	0,084	0,1137	0,0485	0,0814	0,11
	WRV	0,3746	1,9519	4,6394	0,5043	2,1908	5,1993	0,4968	2,1813	5,1833
	Çeşitlilik	0,4386	0,428	0,4265	**0,4313**	0,426	0,4261	0,4316	0,4279	0,426
	Yenilik	**0,6552**	**0,6449**	**0,6403**	**0,6541**	**0,6458**	**0,6413**	**0,6544**	**0,6465**	**0,6414**
	Şans eseri bulma	0,3967	0,3841	0,3823	**0,3929**	0,3824	0,3813	**0,3941**	**0,3832**	**0,3817**
MUP-C	AvgHR	0,0431	0,0664	0,1104	0,0175	0,0361	0,0792	0,0091	0,0318	0,075
	WRV	0,4622	1,5361	4,5087	0,1301	0,6625	2,5732	0,0753	0,5873	2,1027
	Çeşitlilik	0,4328	0,4278	0,414	0,4	0,4065	0,4014	0,2892	0,326	0,3367
	Yenilik	0,6473	0,6377	0,6261	0,6069	0,6144	0,6091	0,5054	0,5397	0,5501
	Şans eseri bulma	**0,4508**	**0,4309**	**0,4129**	0,3923	**0,3968**	**0,3888**	0,269	0,3072	0,3198

Tablo 5.2. *BASEMCCF algoritmasının ID seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM20		**ID 0,01**			**ID 0,05**			**ID 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0243	0,0876	0,1351	0,0339	0,0935	0,1343	0,0305	0,0908	0,1314
	WRV	0,2151	2,2346	5,7225	0,331	2,3217	5,7161	0,2853	2,1435	5,3636
	Çeşitlilik	**0,4413**	**0,4389**	**0,435**	0,4416	**0,4396**	**0,4373**	0,4424	**0,4413**	**0,4387**
	Yenilik	**0,2774**	**0,2797**	**0,283**	**0,2823**	**0,2791**	**0,2801**	**0,2845**	**0,2786**	**0,2805**
	Şans eseri bulma	0,3848	0,3377	0,309	0,3825	0,3343	0,3062	0,3805	0,3322	0,304
MUP-U	AvgHR	0,0873	0,1	0,1116	0,0772	0,0866	0,1119	0,0769	0,1004	0,1151
	WRV	0,9687	3,1269	5,5713	0,8577	2,6918	5,7524	0,8622	2,6682	5,4262
	Çeşitlilik	0,4116	0,4066	0,402	0,4126	0,4075	0,4036	0,4074	0,4066	0,4022
	Yenilik	0,265	0,2604	0,2635	0,2664	0,2627	0,2643	0,2708	0,2643	0,2664
	Şans eseri bulma	0,3796	0,3617	**0,356**	0,375	0,3568	0,348	0,368	0,3469	0,3393
MUP-C	AvgHR	**0,1002**	**0,1742**	**0,2183**	**0,1005**	**0,1624**	**0,2033**	**0,0918**	**0,1534**	**0,1889**
	WRV	**1,1404**	**4,5718**	**9,2972**	**1,0945**	**4,3324**	**8,9674**	**1,0382**	**3,978**	**8,2193**
	Çeşitlilik	0,4388	0,4378	0,4321	**0,4461**	0,4375	0,4329	**0,4474**	0,4397	0,4341
	Yenilik	0,2738	0,2674	0,2676	0,276	0,2697	0,2686	0,2762	0,2716	0,2714
	Şans eseri bulma	**0,4422**	**0,3826**	0,3539	**0,4363**	**0,3824**	**0,3548**	**0,4294**	**0,3789**	**0,3539**

Kullanıcı BASEMCCF yöntemini kullanarak YM10 veri seti ile 5, 10 ve 15 ürün listesi hazırlamak istediğinde, bu sistemin PIA-MT saldırı sonrası performans ölçümleri Tablo 5.1' de verilmektedir. Tabloda güçlü ürün seçim yöntemi olarak ID seçildiğinde ve bu seçimin boyutu arttığında, kullanıcı için hazırlanan listedeki hedef ürün sayısında yani sahte ürünlerin yer almasında artış olduğu görülmektedir. Ayrıca hedef ürünlerin seçilmesinde MUP-G yönteminin kullanılması durumunda kötü niyetli kullanıcının yaptığı saldırının diğer seçim yöntemlerine göre başarılı olduğu görülmektedir. Örneğin, sistem kullanıcısı on üründen oluşan bir liste istediğinde, kötü niyetli kullanıcı tarafından manipüle edilen bir yanlış ürün bu listede yer alacaktır. Bu yanlış ürün yer aldığı sıra ise, on üründen oluşan listede ağırlıklı olarak dördüncü veya beşinci sırada olduğu sonucuna varılabilir. Listede yer alan ürün çeşitliliğine bakıldığında MUP-G' nin seçim yönteminde maksimuma ulaştığı ve saldırı öncesi duruma göre önemli bir farklılık olmadığı görülmektedir. Aynı zamanda yenilik ve şans eseri bulma oranlarında özellikle yenilik değerinde artış olduğu söylenebilir. MUP-U seçim kriterlerinde saldırı öncesi duruma göre %65 değerine kadar ulaşmaktadır. Saldırının başarılı olduğu durumlarda, on ürünlük bir liste hazırlandığında, bu performans kriterlerinin oranı göz önüne alındığında,

kullanıcının beklemediği ürün sayısı iki iken, altı yeni ürün ve dört farklı ürün listede yer almaktadır.

Tablo 5.2, YM20 veri seti kullanıldığında elde edilen sonuçları göstermektedir. PIA-MT saldırısı ile MUP-U yöntemi seçildiğinde birden fazla hedef ürünün listeye alınma oranı diğer veri setine göre daha yüksektir. Saldırı profili için gereken güçlü ürün yüzdesi 0,01 olarak seçildiğinde kullanıcıya sunulan on ürünün iki tanesi hedef ürünler olduğu ancak bu yüzde artması durumunda *AvgHR* ölçüsünde bir düşüş meydana geldiği söylenebilir. Ancak burada kritik noktalardan biri bu hedef ürünlerin kullanıcıya sunulan listede nasıl yer aldığıdır. Dolayısıyla bir sonraki ürün yüzdesi 0,05 olarak seçildiğinde, Tablo 5.2' de kullanıcıya sunulan on üründen yaklaşık 2 tanesinin hedef ürün olduğu ve bu hedef ürünlerden en az birinin ilk beşte kullanıcıya sunulduğu görülmektedir. Diğer kriterlerdeki değişikliklere bakıldığında ise saldırının başarılı olduğu noktalarda listede yenilik, çeşitlilik ve şans eseri bulma metriklerinin düşük bir oranı yer alıyor. Saldırı sonrası çeşitlilik ve yenilik ölçülerinde bir düşüş olduğu söylenebilir. MUP-G seçim kriterinde çeşitlilik ve yenilik metrikleri maksimum değerlerine ulaşırken şans eseri bulma kriteri MUP-C' de en yüksek değerine ulaşır.

Güçlü ürün seçim kriteri olarak AS belirlendiğinde, YM10 ve YM20 veri setleri kullanılarak BASEMCCF sistemine uygulanan saldırı sonrası sistemin gürbüzlüğü Tablo 5.3 ve Tablo 5.4' te gösterilmektedir.

Tablo 5.3. *BASEMCCF algoritmasının AS seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM10		**AS 0,01**			**AS 0,05**			**AS 0,1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	**0,0483**	**0,0876**	**0,1181**	**0,0243**	**0,0427**	**0,082**	**0,0142**	0,0309	0,0709
	WRV	**0,4962**	**2,2449**	**4,9889**	**0,1653**	**1,0239**	2,802	**0,2464**	0,8781	2,4701
	Çeşitlilik	0,4248	0,4242	0,4226	0,2832	0,3379	0,3601	0,1361	0,2084	0,2517
	Yenilik	0,6404	0,6372	0,6333	0,5034	0,5523	0,572	0,3939	0,4492	0,4833
	Şans eseri bulma	0,3363	0,2929	0,2705	**0,3627**	0,2919	0,2561	**0,3234**	**0,2303**	0,1944
MUP-U	AvgHR	0,0397	0,0741	0,101	0,012	0,0418	0,0673	0,0094	**0,0348**	0,0652
	WRV	0,4044	1,9467	4,7122	0,1318	0,98	**2,9377**	0,0998	**0,9937**	**2,7695**
	Çeşitlilik	**0,4283**	**0,4254**	**0,4259**	**0,3119**	**0,3577**	**0,3732**	**0,1889**	**0,2497**	**0,2858**
	Yenilik	**0,6452**	**0,6411**	**0,6366**	**0,5325**	**0,5724**	**0,5858**	**0,4364**	**0,4851**	**0,5131**
	Şans eseri bulma	0,3883	0,3811	0,3792	0,2779	0,3126	0,3284	0,1828	0,225	**0,2555**
MUP-C	AvgHR	0,0414	0,0669	0,1096	0,0064	0,0293	0,0747	0,0033	0,0228	**0,0736**
	WRV	0,4099	1,4828	3,9927	0,0499	0,4913	2,053	0,0213	0,3799	1,8333
	Çeşitlilik	0,4203	0,4211	0,4101	0,2772	0,334	0,354	0,134	0,2061	0,2498
	Yenilik	0,6328	0,6295	0,6204	0,5006	0,5505	0,5674	0,3921	0,447	0,4821
	Şans eseri bulma	**0,4344**	**0,4208**	**0,4058**	0,2652	**0,3199**	**0,3394**	0,1364	0,1973	0,2381

Tablo 5.4. *BASEMCCF algoritmasının AS seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM20		**AS 0,01**			**AS 0,05**			**AS 0,1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0232	0,0794	0,1333	0,0322	0,0895	0,1291	0,0347	0,0757	0,1145
	WRV	0,2191	2,0127	5,7317	0,2736	2,3566	5,6225	0,369	1,9102	4,766
	Çeşitlilik	0,4411	**0,439**	**0,4355**	**0,439**	**0,4378**	**0,4338**	0,4407	0,4352	**0,4299**
	Yenilik	**0,2801**	**0,2784**	**0,2829**	**0,2817**	**0,2791**	**0,2825**	**0,284**	**0,2782**	**0,2783**
	Şans eseri bulma	0,3855	0,3409	0,31	0,3832	0,3372	0,3106	0,3827	0,3419	0,312
MUP-U	AvgHR	0,0533	0,0762	0,1038	0,0615	0,0758	0,1049	0,0552	0,088	0,1027
	WRV	0,5912	2,1902	5,0351	0,6827	1,8487	4,5102	0,5277	2,2482	4,8846
	Çeşitlilik	0,4088	0,4076	0,4011	0,4053	0,4058	0,3998	0,4006	0,4023	0,3963
	Yenilik	0,2685	0,2602	0,2629	0,2683	0,2597	0,2642	0,2684	0,2591	0,2618
	Şans eseri bulma	0,372	0,3587	0,3532	0,3717	0,3578	0,3503	0,374	0,3535	0,3464
MUP-C	AvgHR	**0,1055**	**0,1745**	**0,217**	**0,0995**	**0,1607**	**0,2023**	**0,1**	**0,1515**	**0,192**
	WRV	**1,2149**	**4,638**	**9,5578**	**1,1149**	**4,2609**	**9,1254**	**1,1422**	**4,2266**	**8,7436**
	Çeşitlilik	**0,4422**	0,4376	0,4322	0,4386	0,437	0,4303	**0,4432**	**0,4357**	0,4282
	Yenilik	0,27	0,2669	0,2671	0,2764	0,2694	0,2695	0,2752	0,2687	0,2683
	Şans eseri bulma	**0,4425**	**0,384**	**0,3562**	**0,4438**	**0,3881**	**0,3578**	**0,4406**	**0,386**	**0,3568**

Sonuçlardan YM10 veri setinde saldırının en başarılı olduğu yöntemin MUP-G yöntemi olduğu görülürken, YM20 veri setinde MUP-U seçildiğinde bu başarının sağlandığı söylenebilir. İki veri kümesinin sonuçları karşılaştırıldığında, YM10 veri kümesinin PIA-MT saldırısına karşı daha sağlam olduğunu ve maksimum %11 oranına ulaştığını, YM20 veri kümesinde ise bu sonucu %21 ulaştığını söyleyebiliriz. Güçlü ürün oranı 0.01 seçildiğinde ve kullanıcı için on adet ürün listesi hazırlandığında, listede manipüle edilen bir ürün olduğu ve YM10 veri seti kullanılıyorsa bu ürünün ilk beşte olduğu çıkarımı yapılabilir. YM20 için yanlış ürün sayısı yaklaşık olarak ikidir ve bu ürünlerin sıralaması hakkında bir çıkarım yapılmak istendiğinde ilk olasılığın hedef ürünlerden birinin üçüncü sırada yer alabileceği söylenebilirken, bir diğer ihtimal ise en az iki yanlış ürünün ilk beşte yer almasıdır. Listedeki ürün çeşitliliği MUP-U yönteminde YM10 için maksimum %40 noktasına ulaşırken, YM20 için MUP-G yöntemi ile aynı oran elde edilir. Saldırının başarılı olduğu durumlarda ise bu kriterin oranında anlamlı bir farklılık olmadığı görülmektedir. Öte yandan yenilik açısından YM10' da yeni ürünlerin listede oldukça fazla yer aldığı, YM20 için ise bu oranın %28 civarında olduğu söylenebilir.

Tablo 5.5. *BASEMCCF algoritmasının NR seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM10		**NR 0,01**			**NR 0,05**			**NR 0,1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	**0,0227**	**0,0526**	**0,0875**	0,0175	0,0388	0,0726	0,0134	0,0369	0,0755
	WRV	**0,2532**	**1,3263**	**3,5087**	**0,2413**	1,0774	3,0466	0,1527	0,9482	3,1724
	Çeşitlilik	**0,4471**	**0,4392**	**0,4352**	0,4384	**0,4274**	**0,4184**	**0,4204**	**0,4001**	**0,3865**
	Yenilik	0,6635	**0,6549**	**0,6491**	0,6596	**0,6487**	**0,6396**	**0,6431**	**0,6267**	**0,6147**
	Şans eseri bulma	0,3399	0,3023	0,2829	0,3377	0,3018	0,2838	0,3353	0,2969	0,2754
MUP-U	AvgHR	0,0202	0,0462	0,0728	**0,0182**	**0,0483**	0,0759	**0,0295**	**0,0548**	0,0827
	WRV	0,2097	1,1503	3,1438	0,1822	**1,2578**	**3,2067**	**0,2962**	**1,4305**	**3,5786**
	Çeşitlilik	0,4446	0,4378	0,433	0,4375	0,4246	0,4126	0,416	0,3931	0,3802
	Yenilik	**0,6648**	0,6547	0,6485	**0,6602**	0,6463	0,6346	0,6421	0,6212	0,6085
	Şans eseri bulma	0,4065	0,3939	0,3905	0,4075	0,3931	0,386	0,3996	0,3822	0,3733
MUP-C	AvgHR	0,0208	0,0392	0,0833	0,0114	0,0373	**0,077**	0,0186	0,0478	**0,838**
	WRV	0,167	0,7261	2,8596	0,0827	0,6782	2,7696	0,1705	1,0649	3,4254
	Çeşitlilik	0,4456	0,4385	0,4311	**0,4407**	0,4259	0,4153	0,4182	0,3996	0,3831
	Yenilik	0,6624	0,6533	0,6437	0,6595	0,6465	0,6355	0,6428	0,6261	0,6107
	Şans eseri bulma	**0,4609**	**0,4437**	**0,4306**	**0,4595**	**0,4396**	**0,4266**	**0,4479**	**0,4239**	**0,4076**

Tablo 5.6. *BASEMCCF algoritmasının NR seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM20		**NR 0,01**			**NR 0,05**			**NR 0,1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0279	0,077	0,1297	0,0256	0,0831	0,1285	0,0249	0,087	0,1304
	WRV	0,2582	1,8606	5,5457	0,2154	1,7936	4,9462	0,2608	2,1149	5,4113
	Çeşitlilik	**0,4422**	**0,4413**	**0,4369**	0,4388	**0,4408**	**0,4355**	0,4377	0,4383	**0,4357**
	Yenilik	**0,2749**	**0,2781**	**0,2841**	**0,2732**	**0,2773**	**0,2824**	**0,2736**	**0,2777**	**0,2831**
	Şans eseri bulma	0,3892	0,3418	0,311	0,3874	0,3403	0,3119	0,3899	0,3386	0,3108
MUP-U	AvgHR	0,0599	0,0773	0,102	0,0569	0,0831	0,1073	0,0576	0,0729	0,0984
	WRV	0,5813	2,1398	4,7909	0,5819	2,2777	4,9207	0,6223	1,759	4,7141
	Çeşitlilik	0,4052	0,4075	0,4011	0,4056	0,4078	0,402	0,4059	0,4073	0,4009
	Yenilik	0,2625	0,2584	0,2618	0,2619	0,2585	0,261	0,2595	0,2586	0,262
	Şans eseri bulma	0,3686	0,3572	**0,3516**	0,3679	0,3563	**0,3528**	0,3671	0,3573	**0,3533**
MUP-C	AvgHR	**0,1031**	**0,174**	**0,2177**	**0,1029**	**0,1742**	**0,2172**	**0,101**	**0,174**	**0,2165**
	WRV	**1,0813**	**4,3213**	**9,0989**	**1,0473**	**4,3725**	**9,0424**	**1,0559**	**4,325**	**9,0888**
	Çeşitlilik	0,4403	0,4399	0,4321	**0,4414**	0,4382	0,4318	**0,4414**	**0,4392**	0,4327
	Yenilik	0,2679	0,2655	0,2666	0,2695	0,265	0,2671	0,27	0,2658	0,267
	Şans eseri bulma	**0,4229**	**0,3755**	0,35	**0,4228**	**0,3755**	0,3503	**0,4232**	**0,3753**	0,3504

Tablo 5.7. *BASEMCCF algoritmasının SR seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM10		**SR 0,01**			**SR 0,05**			**SR 0,1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	**0,0494**	**0,0953**	**0,125**	**0,0188**	**0,0482**	**0,0889**	0,0079	0,0304	**0,0784**
	WRV	**0,5279**	**2,3323**	**5,2492**	**0,2022**	**1,1238**	3,016	0,0988	0,7764	2,5345
	Çeşitlilik	0,4263	0,4219	0,4213	0,2295	0,3126	0,3455	0,0896	0,1583	0,2228
	Yenilik	0,6261	0,6266	0,625	0,4496	0,511	0,5381	0,354	0,3988	0,4429
	Şans eseri bulma	0,3359	0,2891	0,2649	**0,3554**	0,2771	0,2375	**0,298**	**0,194**	0,1609
MUP-U	AvgHR	0,048	0,0794	0,1102	0,0167	0,044	0,0748	**0,0111**	**0,0363**	0,0648
	WRV	0,4586	2,2574	5,152	0,1768	1,0761	**3,2557**	**0,1177**	**0,9592**	**3,0464**
	Çeşitlilik	**0,4295**	**0,4242**	**0,4233**	**0,2794**	**0,346**	**0,3727**	**0,1583**	**0,2169**	**0,2679**
	Yenilik	**0,6307**	**0,6304**	**0,6288**	**0,4962**	**0,5459**	**0,5672**	**0,4098**	**0,4476**	**0,4825**
	Şans eseri bulma	0,3744	0,3702	0,3713	0,242	**0,285**	**0,3086**	0,1534	0,1856	**0,2224**
MUP-C	AvgHR	0,0453	0,0735	0,1092	0,0122	0,0361	0,0791	0,0036	0,0213	0,0742
	WRV	0,4276	1,5729	3,955	0,0784	0,5822	2,1634	0,0192	0,2976	1,689
	Çeşitlilik	0,4196	0,4147	0,4073	0,2293	0,3079	0,3442	0,091	0,1566	0,2193
	Yenilik	0,6165	0,6155	0,6113	0,4497	0,5074	0,5366	0,3553	0,3975	0,4399
	Şans eseri bulma	**0,4161**	**0,406**	**0,3945**	0,2012	0,2673	0,3005	0,0879	0,1359	0,1853

Tablo 5.8. *BASEMCCF algoritmasının SR seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

BASEMCCF-YM20		SR 0,01			SR 0,05			SR 0.1		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0294	0,0832	0,1276	0,0353	0,0851	0,1248	0,0294	0,0832	0,1245
	WRV	0,3249	1,9994	5,66	0,3262	2,1462	5,5085	0,2713	2,2358	5,7277
	Çeşitlilik	0,4363	0,4356	**0,4318**	0,4362	**0,4356**	**0,4284**	**0,4367**	**0,434**	0,4272
	Yenilik	**0,2744**	**0,2774**	**0,2804**	**0,275**	**0,2723**	**0,2732**	**0,2701**	**0,2668**	**0,2671**
	Şans eseri bulma	0,3854	0,3407	0,3113	0,3851	0,3365	0,3056	0,3832	0,3316	0,3012
MUP-U	AvgHR	0,043	0,0788	0,0988	0,0558	0,0785	0,0893	0,043	0,0769	0,0914
	WRV	0,4531	2,228	4,5904	0,5501	2,0026	4,3018	0,5309	1,8058	4,4946
	Çeşitlilik	0,4017	0,4062	0,4001	0,4057	0,4011	0,3969	0,4056	0,3989	0,3941
	Yenilik	0,2674	0,2593	0,2616	0,2602	0,2539	0,2572	0,2558	0,2506	0,2511
	Şans eseri bulma	0,3736	0,3556	0,3512	0,3653	0,3459	0,3405	0,364	0,34	0,3316
MUP-C	AvgHR	**0,0946**	**0,1544**	**0,1956**	**0,0727**	**0,1157**	**0,1586**	**0,0701**	**0,1078**	**0,1472**
	WRV	**1,1398**	**4,1396**	**8,5467**	**0,9198**	**3,4637**	**7,2157**	**0,9738**	**3,3067**	**6,836**
	Çeşitlilik	**0,4379**	**0,4367**	0,4296	**0,4381**	0,4335	0,4282	0,4347	0,4333	**0,4274**
	Yenilik	0,2655	0,2649	0,2663	0,2696	0,2614	0,2628	0,2667	0,2566	0,2582
	Şans eseri bulma	**0,4346**	**0,3818**	**0,355**	**0,4194**	**0,3744**	**0,3475**	**0,4115**	**0,367**	**0,3409**

BASEMCCF yönteminin diğer tüm sonuçları incelendiğinde, her iki veri setinde de sistemin gürbüzlük durumlarının benzer olduğu Tablo 5.5 – Tablo 5.8' de görülmektedir. Bu yöntem YM10 veri seti için PIA-MT saldırısına karşı oldukça gürbüz iken YM20 veri seti için de benzer durumlar söz konusudur. Ayrıca yanlış ürünlerin ilk etapta kullanıcıya sunulma ihtimalinin yüksek olduğu sonucuna varılabilir. Tablolarda dikkat edilmesi gereken noktalardan biri, kullanıcıya verilen listedeki yanlış ürün sayısı az olsa da bu ürünlerin sunum aşamasında ilk sıralarda yer almasıdır. Örneğin Tablo 5.6' da en yüksek *AvgHR* başarısı, MUP-C yönteminde görülürken, *WRV* değerinde en yüksek başarının aynı metrikte olduğu söylenebilir. Listedeki hedef ürün sayısı az olsa da bu ürünlerin ilk sıralarda yer alabileceği çıkarımı yapılabilir. Tablo 5.8' de 0.01 güçlü ürün seçimi durumunda, kullanıcıya sunulan 5 üründen oluşan listede, bir ürün kullanıcının memnun olmayacağı ürün iken, bu ürünler kullanıcıya son sıralarda tavsiye edilir. Saldırı ile diğer kriterlerin nasıl değiştiğine bakıldığında, genel değerlendirmede YM10 veri setinde listede yer alan yeni, çeşit ve şans eseri bulunan ürünlerinin sayısı sırasıyla 6, 4, 3 iken, YM20 için 3, 4, 4' tür.

BASEMCCF için tüm sonuçlar YM10 üzerinde incelendiğinde, PIA-MT saldırısının güçlü ürün seçimi ID ve hedef ürün seçimi MUP-G olduğunda en başarılı

olduğu, en gürbüz durumların ise AS ve MUP-C' de olduğu görülmektedir. YM20 veri setinde, kriter bazlı hedef ürün seçilirse ve güçlü ürün seçimi olarak SR belirlenirse saldırının bu algoritma üzerinde başarılı olmadığı ve başarılı olduğu durumun NR seçimi ile gerçekleştiği sonucuna varılabilir.

Kullanıcıya beğeneceği ürünleri önerme konusunda oldukça başarılı bir algoritma olan IFMC-EntARMCF yönteminin PIA-MT saldırısına karşı gürbüzlük analizi Tablo 5.9- 5.16' da gösterilmektedir.

Tablo 5.9. *IFMC-EntARMCF algoritmasının ID seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM10		**ID 0,01**			**ID 0,05**			**ID 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,1199	**0,1899**	**0,225**	**0,1168**	**0,1825**	**0,2104**	**0,1161**	**0,1781**	**0,2112**
	WRV	0,9526	**4,2496**	**9,132**	**0,9237**	**4,0397**	**8,5526**	**0,914**	**3,919**	**8,3741**
	Çeşitlilik	**0,474**	**0,4682**	**0,4678**	**0,4742**	**0,4677**	**0,468**	**0,473**	**0,4681**	**0,4682**
	Yenilik	0,5101	**0,4748**	**0,4676**	**0,5028**	**0,4645**	**0,4558**	**0,4993**	**0,4611**	**0,451**
	Şans eseri bulma	0,3206	0,2735	0,2514	0,3233	0,2732	0,2526	0,3227	0,2711	0,249
MUP-U	AvgHR	**0,1536**	0,1887	0,1915	0,0812	0,121	0,1302	0,083	0,1206	0,132
	WRV	**1,2121**	4,2206	7,86	0,605	2,5452	5,0209	0,6288	2,5651	5,2132
	Çeşitlilik	0,4686	0,4613	0,4615	0,4705	0,4622	0,4623	0,4689	0,4617	0,4617
	Yenilik	**0,5111**	0,4703	0,4613	0,4928	0,4563	0,4496	0,4925	0,4545	0,4462
	Şans eseri bulma	0,3448	0,3293	0,3303	0,3384	0,3302	0,3331	0,338	0,3275	0,3295
MUP-C	AvgHR	0,0509	0,1059	0,1246	0,0572	0,1015	0,1256	0,0511	0,0964	0,1181
	WRV	0,3896	2,3659	4,9474	0,4632	2,2165	4,5137	0,4233	2,1175	4,3654
	Çeşitlilik	0,4697	0,4628	0,4631	0,4697	0,463	0,4632	0,4697	0,4631	0,4631
	Yenilik	0,5046	0,459	0,4474	0,497	0,4537	0,4425	0,4943	0,4509	0,4378
	Şans eseri bulma	**0,4638**	**0,434**	**0,424**	**0,4485**	**0,4235**	**0,4144**	**0,4424**	**0,4167**	**0,4064**

Tablo 5.10. *IFMC-EntARMCF algoritmasının ID seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM20		**ID 0,01**			**ID 0,05**			**ID 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0754	0,179	0,2623	0,1433	0,2519	0,3019	0,2115	0,2242	0,1892
	WRV	0,9454	6,8702	17,830	1,8281	9,3322	20,375	2,7757	10,218	21,945
	Çeşitlilik	**0,416**	**0,4192**	**0,42**	**0,4002**	**0,3996**	**0,398**	**0,3771**	**0,3774**	**0,3691**
	Yenilik	**0,3357**	**0,3438**	**0,3509**	**0,3396**	**0,3446**	**0,349**	**0,3361**	**0,3423**	**0,3473**
	Şans eseri bulma	0,3352	0,2825	0,2478	0,31	0,2495	0,217	0,2857	0,2242	0,1892
MUP-U	AvgHR	**0,3861**	**0,5246**	**0,5211**	**0,3165**	**0,4007**	**0,4027**	**0,2857**	**0,3385**	0,3382
	WRV	**5,3872**	**18,682**	**32,201**	**4,3867**	**14,590**	**25,384**	**3,9724**	**13,078**	**22,503**
	Çeşitlilik	0,3498	0,3338	0,3365	0,308	0,293	0,2945	0,2839	0,2728	0,2734
	Yenilik	0,2883	0,2931	0,2966	0,2902	0,2927	0,2937	0,2903	0,2916	0,2914
	Şans eseri bulma	0,1978	0,1486	0,1525	0,1822	0,1437	0,1452	0,165	0,1357	0,1374
MUP-C	AvgHR	0,1031	0,2515	0,4053	0,1237	0,2586	0,3933	0,1292	0,2547	**0,3766**
	WRV	1,375	9,0265	23,884	1,6854	9,216	23,239	1,7566	9,2436	22,355
	Çeşitlilik	0,403	0,408	0,4035	0,3796	0,3851	0,3818	0,3596	0,3615	0,3572
	Yenilik	0,3248	0,3258	0,3237	0,3179	0,3204	0,3196	0,3167	0,3165	0,6142
	Şans eseri bulma	**0,4147**	**0,3444**	**0,2893**	**0,3681**	**0,3062**	**0,2582**	**0,322**	**0,2648**	**0,2205**

Saldırının bu sisteme etkisinin gösterildiği ilk iki tabloda (Tablo 5.9 ve Tablo 5.10) YM10 ve YM20 veri setleri üzerinde, güçlü ürün seçimi ID olarak belirlendiğinde sistemin performansının nasıl değiştiği verilmektedir. YM10 veri seti üzerine saldırı yapıldığını ve bunun sonuçlarının ne olduğunu inceleyelim. Böyle bir durumda Tablo 5.9' da on üründen oluşan listede yaklaşık iki yanlış ürünün yer aldığı ve hedeflenen ürün MUP-G seçilerek bu başarılı saldırının gerçekleştiği görülmektedir. Ayrıca bu yanlış ürünlerden en az birinin ilk beşte olduğu çıkarımı yapılabilir. Listede yer alan ürünlerin çeşitlilik ve yeniliklerinde ufak tefek dalgalanmalar olurken, şans eseri bulma metriğinde artış olduğu görülüyor. Sonuç olarak bu kriterlerde elde edilen başarı %40' ın üzerindedir.

Veri seti YM20 olarak değiştirildiğinde saldırının oldukça başarılı olduğunu ve sistemin bu saldırıya karşı gürbüz olmadığını söyleyebiliriz. En başarılı *AvgHR* değerinin hedef ürün MUP-U seçilerek elde edildiği Tablo 5.10' da gösterilmektedir ve kullanıcıya sunulan listedeki ürünlerin yaklaşık yarısının hedef yanlış ürünlerden oluştuğu görülmektedir. Hatta bu hedef ürünlerin sıralaması hakkında bir çıkarımda bulunulduğunda, ID 0,01 oranında seçildiğinde 10 ürün listesindeki hedef ürünlerden en az iki tanesinin ilk beşe girebileceği söylenebilir. Dahası bu ürünler, sıralama olarak kullanıcıya ilk iki sırada sunulabilir. Diğer performans ölçütlerinin en iyi sonucu verdiği

noktalar çeşitlilik gösterirken, saldırı başarılı olduğunda bu ölçülerin oranında bir azalma olmaktadır.

Tablo 5.11. *IFMC-EntARMCF algoritmasının AS seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM10		**AS 0,01**			**AS 0,05**			**AS 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,1063	0,1859	0,2251	0,1147	0,1895	0,2331	0,056	0,1158	0,1487
	WRV	0,8541	4,172	9,0066	0,8889	4,2464	9,4343	0,4316	2,468	5,5649
	Çeşitlilik	**0,4739**	**0,4683**	**0,4678**	**0,4742**	**0,468**	**0,4678**	0,4698	0,4634	**0,4639**
	Yenilik	**0,5078**	**0,4751**	**0,4663**	**0,5045**	**0,4681**	**0,4598**	**0,4971**	**0,4533**	**0,4421**
	Şans eseri bulma	0,3246	0,2753	0,2532	0,325	0,2731	0,2472	0,3485	0,3008	0,2774
MUP-U	AvgHR	**0,4434**	**0,4671**	**0,4195**	**0,4525**	**0,4717**	**0,424**	**0,4472**	**0,4671**	**0,4217**
	WRV	**3,4871**	**10,445**	**17,877**	**3,51**	**10,628**	**18,38**	**3,4982**	**10,519**	**18,061**
	Çeşitlilik	0,4543	0,4567	0,4615	0,4547	0,4563	0,4613	0,4538	0,4563	0,4613
	Yenilik	0,4815	0,4401	0,4337	0,4776	0,4332	0,4249	0,4707	0,4246	0,417
	Şans eseri bulma	0,2329	0,2259	0,2488	0,2301	0,2189	0,241	0,224	0,2116	0,2314
MUP-C	AvgHR	0,0574	0,1168	0,1372	0,0574	0,1168	0,1372	0,0575	0,1131	0,1408
	WRV	0,4419	2,6272	5,6486	0,4419	2,6272	5,6486	0,4665	2,5642	5,6873
	Çeşitlilik	0,4697	0,4633	0,4631	0,4697	0,4633	0,4631	**0,4699**	**0,4636**	0,4638
	Yenilik	0,5019	0,4571	0,4433	0,5019	0,4571	0,4433	0,4901	0,444	0,43
	Şans eseri bulma	**0,4613**	**0,4296**	**0,4183**	**0,4613**	**0,4296**	**0,4183**	**0,4405**	**0,4037**	**0,3889**

Tablo 5.12. *IFMC-EntARMCF algoritmasının AS seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM20		**AS 0,01**			**AS 0,05**			**AS 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0908	0,1796	0,2563	0,1811	0,2685	0,3316	0,2222	0,3139	0,3677
	WRV	1,2403	7,0872	17,871	2,4528	9,9759	21,476	2,9502	11,658	23,517
	Çeşitlilik	**0,4194**	**0,424**	**0,4234**	**0,3934**	**0,392**	**0,3862**	**0,3464**	**0,349**	**0,3456**
	Yenilik	**0,3387**	**0,3456**	**0,3492**	**0,32**	**0,3302**	**0,335**	**0,3052**	**0,3183**	**0,3241**
	Şans eseri bulma	0,3312	0,2802	0,2501	0,3028	0,239	0,2028	**0,275**	0,2026	0,1694
MUP-U	AvgHR	**0,3111**	**0,3649**	**0,3654**	**0,3417**	**0,4347**	**0,4308**	**0,3075**	**0,3598**	0,3614
	WRV	**4,355**	**13,930**	**24,407**	**4,7096**	**16,059**	**27,491**	**4,2924**	**13,686**	**23,779**
	Çeşitlilik	0,2802	0,277	0,2784	0,2939	0,2885	0,2907	0,2797	0,2769	0,2785
	Yenilik	0,2588	0,2618	0,2637	0,261	0,2665	0,2668	0,2599	0,262	0,2638
	Şans eseri bulma	0,1579	0,1371	0,1373	0,1645	0,1338	0,1364	0,1572	0,1368	0,137
MUP-C	AvgHR	0,0961	0,2429	0,398	0,1286	0,2746	0,4123	0,1411	0,2768	**0,3884**
	WRV	1,4047	8,6918	23,742	1,7876	9,3729	23,334	1,9212	9,8351	22,48
	Çeşitlilik	0,4105	0,4111	0,4058	0,3712	0,3722	0,3687	0,3326	0,336	0,3349
	Yenilik	0,3237	0,3249	0,323	0,298	0,3008	0,3013	0,2837	0,2844	0,2884
	Şans eseri bulma	**0,413**	**0,3417**	**0,2851**	**0,3158**	**0,2578**	**0,2143**	0,2543	**0,2056**	**0,1732**

Tablo 5.13. *IFMC-EntARMCF algoritmasının NR seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM10		**NR 0,01**			**NR 0,05**			**NR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,1077	0,1858	0,228	0,1133	0,1926	0,2345	0,0617	0,1184	0,1632
	WRV	0,844	4,1401	9,205	0,8804	4,2189	9,4339	0,4693	2,5197	6,2645
	Çeşitlilik	**0,4742**	**0,4678**	**0,4678**	**0,474**	**0,4675**	**0,4675**	**0,4696**	**0,463**	**0,4633**
	Yenilik	**0,5014**	**0,4662**	**0,4611**	**0,4835**	**0,4439**	**0,4367**	**0,4657**	**0,4138**	**0,401**
	Şans eseri bulma	0,3278	0,2725	0,2446	0,3266	0,2588	0,2199	0,3282	0,2526	0,2131
MUP-U	AvgHR	**0,4026**	**0,3881**	**0,3379**	**0,4037**	**0,3982**	**0,342**	**0,4043**	**0,3044**	**0,3413**
	WRV	**3,1871**	**9,1161**	**15,255**	**3,1881**	**9,2151**	**15,704**	**3,2073**	**9,1713**	**15,451**
	Çeşitlilik	0,4583	0,4569	0,4597	0,4589	0,4569	0,4596	0,4586	0,4571	0,4597
	Yenilik	0,4817	0,4368	0,429	0,4732	0,4222	0,4114	0,465	0,4104	0,3969
	Şans eseri bulma	0,241	0,2333	0,2517	0,2281	0,2049	0,2171	0,2189	0,1901	0,194
MUP-C	AvgHR	0,0589	0,1139	0,1391	0,0582	0,1148	0,1391	0,0607	0,1162	0,1407
	WRV	0,466	2,5535	5,705	0,4626	2,6033	5,6878	0,4755	2,627	5,7852
	Çeşitlilik	0,4694	0,4631	0,4632	0,469	0,4633	0,4632	0,469	0,4626	0,4632
	Yenilik	0,4911	0,4487	0,4351	0,4728	0,4255	0,4093	0,4598	0,4055	0,3899
	Şans eseri bulma	**0,4147**	**0,386**	**0,3784**	**0,3581**	**0,3151**	**0,3004**	**0,322**	**0,27**	**0,2535**

Tablo 5.14. *IFMC-EntARMCF algoritmasının NR seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM20		**NR 0,01**			**NR 0,05**			**NR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,3343	0,4277	0,4546	0,2675	0,3854	0,4437	0,126	0,2211	0,3085
	WRV	4,3398	15,399	28,743	3,4735	13,480	27,100	1,7524	8,318	20,900
	Çeşitlilik	0,346	0,3419	0,3423	**0,3797**	**0,3785**	**0,3743**	**0,4178**	**0,4216**	**0,4184**
	Yenilik	0,2956	0,3113	0,3146	**0,3095**	**0,3273**	**0,3328**	**0,3401**	**0,3501**	**0,354**
	Şans eseri bulma	0,2434	0,1813	0,16	0,2708	0,2048	**0,1723**	**0,3191**	**0,2662**	**0,231**
MUP-U	AvgHR	**0,4395**	**0,571**	**0,5625**	**0,4005**	**0,5057**	0,4965	**0,3733**	**0,435**	0,433
	WRV	**6,2137**	**20,39**	**35,185**	**5,6133**	**17,938**	**31,414**	**5,2562**	**16,214**	28,248
	Çeşitlilik	0,3537	0,3403	0,343	0,3088	0,3049	0,3086	0,2949	0,2939	0,296
	Yenilik	0,2753	0,2817	0,2851	0,2338	0,244	0,2463	0,231	0,2373	0,2387
	Şans eseri bulma	0,192	0,1457	0,1521	0,1673	0,1353	0,1408	0,159	0,1395	0,141
MUP-C	AvgHR	0,1297	0,3042	0,4621	0,215	0,404	**0,5285**	0,2269	0,3768	**0,4877**
	WRV	1,8964	10,983	27,14	2,8383	13,325	29,858	3,0228	12,994	**28,438**
	Çeşitlilik	**0,4089**	**0,4093**	**0,4017**	0,3698	0,3689	0,3611	0,3375	0,3399	0,3373
	Yenilik	**0,3257**	**0,3236**	**0,3203**	0,271	0,2808	0,286	0,2494	0,2595	0,2666
	Şans eseri bulma	**0,4**	**0,3207**	**0,2596**	**0,2825**	**0,2144**	0,1722	0,2324	0,1826	0,1501

Tablo 5.11 güçlü ürün seçimi AS olarak değiştirildiğinde ve saldırı uygulandığında, PIA-MT saldırısının, önceki tabloya kıyasla YM10 veri kümesinde oldukça başarılı olduğunu ve %45 oranında bir *AvgHR* başarısına ulaştığını göstermektedir. Ancak diğer performans ölçütleri olan çeşitlilikte, yenilikte ve şansta bir azalma var. Tablo 5.13' e göre, güçlü ürün NR ve hedef ürün MUP-U olarak seçildiğinde de aynı durumun gerçekleştiğini söyleyebiliriz. Her iki tabloda da (Tablo 5.11 ve Tablo 5.13), listedeki hedef ürünlerin seçimi MUP-C ise, şans eseri bulma oranında artış vardır.

YM20 verileri üzerinde farklı güçlü ürün seçim yöntemleri ile sistemin atağa karşı başarı analizi yapıldığında AS olarak yapılan seçimde saldırının başarısında azalma olduğu görülmektedir. Ancak seçim kriteri NR olarak değiştirildiğinde saldırı başarısında bir artış olmaktadır (bkz. Tablo 5.12 ve Tablo 5.14). Çeşitlilik, yenilik ve şans eser bulma performans kriterleri incelendiğinde farklı hedef ürün seçimlerinde maksimum noktalara ulaştığı görülmektedir. Ancak saldırının başarılı olduğu durumlarda *çeşitlilik*, *yenilik* ve *şans eseri bulma* kriterlerinde azalma meydana gelmektedir.

Tablo 5.15. *IFMC-EntARMCF algoritmasının SR seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM10		**SR 0,01**			**SR 0,05**			**SR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0684	0,1221	0,1617	0,0593	0,1142	0,1509	0,0578	0,1183	0,1597
	WRV	0,5254	2,6716	6,2587	0,4558	2,4273	5,8926	0,4307	2,4831	6,0479
	Çeşitlilik	0,4693	0,4627	**0,4637**	**0,4695**	0,4632	**0,4636**	**0,4701**	**0,4635**	0,4636
	Yenilik	0,4978	**0,4578**	**0,4499**	**0,5016**	**0,4611**	**0,4508**	**0,4962**	**0,456**	**0,4433**
	Şans eseri bulma	0,3484	0,3049	0,2788	0,3475	0,3067	0,2852	0,3481	0,3022	0,2788
MUP-U	AvgHR	**0,1484**	**0,1834**	**0,189**	**0,1397**	**0,1828**	**0,1847**	**0,1464**	**0,1862**	**0,1864**
	WRV	**1,1181**	**4,1464**	**7,7732**	**1,0321**	**3,9344**	**7,4385**	**1,1155**	**4,0188**	**7,5825**
	Çeşitlilik	0,4004	0,3993	0,3995	0,3984	0,398	0,3988	0,3973	0,3972	0,3982
	Yenilik	0,4392	0,4138	0,4106	0,4324	0,4045	0,4013	0,4251	0,398	0,3943
	Şans eseri bulma	0,291	0,2801	0,2825	0,282	0,2678	0,2708	0,2762	0,261	0,2625
MUP-C	AvgHR	0,0599	0,1144	0,1431	0,0567	0,1114	0,136	0,0605	0,1142	0,1421
	WRV	0,483	2,6019	5,7772	0,4636	2,5473	5,6808	0,4876	2,5688	5,7354
	Çeşitlilik	**0,4698**	**0,4631**	0,4635	0,4694	**0,4633**	0,4635	0,4699	**0,4635**	**0,4638**
	Yenilik	**0,4997**	0,4569	0,4441	0,4967	0,4532	0,4386	0,4903	0,4444	0,4315
	Şans eseri bulma	**0,4621**	**0,4306**	**0,4177**	**0,4558**	**0,4231**	**0,4107**	**0,4421**	**0,4078**	**0,3941**

Tablo 5.16. *IFMC-EntARMCF algoritmasının SR seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

IFMC-EntARMCF-YM20		**SR 0,01**			**SR 0,05**			**SR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,1815	0,3066	0,3741	0,3005	**0,4243**	**0,4758**	**0,2906**	**0,3847**	**0,4234**
	WRV	2,4656	11,407	23,024	0,9491	14,477	**28,234**	**3,8603**	**13,344**	**25,42**
	Çeşitlilik	**0,4115**	**0,4085**	**0,4057**	**0,3329**	**0,3231**	**0,3199**	**0,2899**	**0,2895**	**0,2901**
	Yenilik	**0,3405**	**0,3518**	**0,3567**	**0,3322**	**0,3431**	**0,3471**	**0,3266**	**0,3379**	**0,3376**
	Şans eseri bulma	0,3019	0,2426	0,2129	**0,2341**	0,156	0,1264	**0,1747**	0,1261	0,1111
MUP-U	AvgHR	**0,4647**	**0,5813**	**0,568**	**0,3392**	0,41	0,4114	0,2155	0,3145	0,3274
	WRV	**6,3717**	**20,412**	**35,038**	**4,579**	**14,928**	26,435	2,9817	12,406	22,783
	Çeşitlilik	0,3425	0,3269	0,3318	0,2796	0,2758	0,2789	0,2616	0,2641	0,2667
	Yenilik	0,2825	0,2878	0,2905	0,2765	0,2807	0,281	0,2788	0,2803	0,2804
	Şans eseri bulma	0,17	0,1276	0,1365	0,135	0,1132	0,1161	0,1449	0,1234	0,1226
MUP-C	AvgHR	0,1775	0,3562	0,5067	0,1828	0,3223	0,4611	0,1545	0,2362	317
	WRV	2,3876	12,416	28,866	2,3064	10,684	26,498	2,1787	8,7979	19,487
	Çeşitlilik	0,4011	0,3984	0,3898	0,3247	0,3208	0,3149	0,287	0,2893	0,2895
	Yenilik	0,3187	0,3178	0,3165	0,299	0,3011	0,3007	0,3017	0,3024	0,3034
	Şans eseri bulma	**0,3759**	**0,294**	**0,2366**	0,2164	**0,169**	**0,1308**	0,1705	**0,144**	**0,126**

Tablo 5.15’ te son güçlü ürün seçim yöntemi olan SR belirlendiğinde, YM10 veri kümesindeki saldırı başarısının %15 civarında olduğu, ancak güçlü ürün artışının başarıda önemli bir fark oluşturmadığı gösterilmiştir. Saldırı, hedef ürün MUP-U olarak seçildiğinde en başarılı olurken, aynı başarı listede yenilik, çeşitlilik ve şans eser bulma kriterlerinin performansında azalması ile gerçekleşiyor. Veri seti YM20 olarak değiştirildiğinde sonuçlara bakıldığında hedef ürün MUP-U olarak seçildiğinde başarı oranının %58,5’ lere ulaştığı ancak güçlü ürün oranı arttıkça *AvgHR* oranının azaldığı, tersi durum hedef ürün MUP-G seçildiğinde görülmektedir. Tablo 5.16’ ya göre, saldırının başarılı olduğu noktalarda diğer kriterler olan çeşitlilik, yenilik ve tesadüfün en yüksek başarıyı sağlamadığı söylenebilir.

PIA-MT saldırısı sonrasında kullanıcı memnuniyeti açısından başarılı öneriler üreten IFMC-EntARMCF algoritmasının tüm sonuçları incelendiğinde, hedef ürün seçimi MUP-U olarak belirlendiğinde genel olarak atağın çok başarılı olduğu ve bu algoritmanın manipülasyona açık olduğu söylenebilir.

PIA-MT saldırısının gerçekleştirildiği başka bir başarılı en iyi-n öneri algoritması olan CrispMC-EntARMCF’ nin performans değerlendirmesi aşağıdaki tablolarda gösterilmektedir.

Tablo 5.17. *CrispMC-EntARMCF algoritmasının ID seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM10		**ID 0,01**			**ID 0,05**			**ID 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,1935	0,2748	0,3153	0,0356	0,0769	0,1052	0,2113	0,2952	0,337
	WRV	1,5539	6,3455	13,711	0,2728	1,7545	4,5123	1,7019	6,976	14,584
	Çeşitlilik	**0,4154**	**0,3782**	**0,3709**	0,3287	0,2625	0,2413	**0,3797**	**0,3357**	**0,3272**
	Yenilik	**0,6063**	**0,5938**	**0,5913**	**0,5275**	**0,5011**	**0,4934**	**0,5649**	**0,5468**	**0,5438**
	Şans eseri bulma	0,2842	0,2289	0,2049	**0,331**	0,2757	0,255	0,2728	0,2047	0,1779
MUP-U	AvgHR	0,2726	0,2965	0,2893	0,2752	0,2986	0,2976	0,2724	0,3021	0,3042
	WRV	2,2305	7,1772	13,059	2,2344	7,2022	13,652	2,1847	7,1097	13,682
	Çeşitlilik	0,3451	0,3135	0,3105	**0,3364**	**0,3047**	**0,299**	0,3295	0,2935	0,2869
	Yenilik	0,5135	0,4738	0,4665	0,5065	0,4671	0,458	0,5017	0,4603	0,4516
	Şans eseri bulma	**0,3165**	**0,3152**	**0,3265**	0,3091	**0,3068**	**0,3153**	**0,3018**	**0,2958**	**0,3049**
MUP-C	AvgHR	**0,5093**	**0,6204**	**0,6554**	**0,5078**	**0,617**	**0,6495**	**0,5006**	**0,6121**	**0,6445**
	WRV	**3,9481**	**13,201**	**24,803**	**3,9554**	**13,151**	**24,635**	**3,8798**	**13,04**	**24,629**
	Çeşitlilik	0,2324	0,1693	0,1491	0,232	0,1671	0,1492	0,2303	0,163	0,1438
	Yenilik	0,4592	0,412	0,3991	0,4564	0,4087	0,3976	0,4548	0,4033	0,3916
	Şans eseri bulma	0,2289	0,1712	0,1526	0,2243	0,1658	0,1489	0,2223	0,1595	0,1418

Tablo 5.18. *CrispMC-EntARMCF algoritmasının ID seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM20		**ID 0,01**			**ID 0,05**			**ID 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,0746	0,1084	0,1405	**0,0905**	0,1193	0,1347	0,0809	0,1205	0,142
	WRV	0,907	3,6149	8,3385	**1,2158**	4,4825	8,7018	1,031	3,9517	8,9971
	Çeşitlilik	**0,2402**	**0,2508**	**0,2581**	**0,2419**	**0,2517**	0,2579	**0,2396**	**0,2518**	**0,2605**
	Yenilik	**0,4628**	**0,4593**	**0,4535**	**0,462**	**0,4574**	**0,4533**	0,4584	**0,4547**	**0,4508**
	Şans eseri bulma	0,3508	0,3164	0,2951	0,3451	0,3127	0,2957	0,3479	0,3114	0,2935
MUP-U	AvgHR	**0,0775**	**0,1415**	**0,1718**	0,0848	**0,1247**	**0,1721**	**0,0832**	**0,1362**	**0,1731**
	WRV	**0,9781**	**4,6872**	**10,530**	1,0924	**4,5566**	**10,737**	**1,0536**	**4,5427**	**10,634**
	Çeşitlilik	0,217	0,222	0,2276	0,2106	0,2201	0,2309	0,2167	0,2251	0,2308
	Yenilik	0,4248	0,4163	0,413	0,4223	0,4146	0,4117	0,4194	0,4143	0,4098
	Şans eseri bulma	0,3303	0,2979	0,2887	0,3247	0,3012	0,2869	0,3256	0,2953	0,2851
MUP-C	AvgHR	0,0721	0,1205	0,1504	0,0714	0,1146	0,1446	0,061	0,1128	0,1408
	WRV	0,9743	4,4596	9,9027	0,9312	4,3635	9,5073	0,8715	4,1618	8,9324
	Çeşitlilik	0,2328	0,2501	0,2564	0,2371	**0,2517**	**0,2587**	0,232	0,2476	0,2598
	Yenilik	0,462	0,4548	0,4506	0,4607	0,4538	0,449	**0,4598**	0,4524	0,4482
	Şans eseri bulma	**0,367**	**0,3247**	**0,3055**	**0,364**	**0,3241**	**0,3047**	**0,3617**	**0,3216**	**0,3031**

Saldırının ilk veri seti olan YM10 üzerindeki etkisi incelendiğinde, Tablo 5.17' de görüldüğü gibi, güçlü ürün seçimi ID olduğunda, hedef ürün MUP-C olarak seçildiğinde, saldırının başarılı olduğunu söylenebilir. Belirlenen seçim kriterleri doğrultusunda %65' e varan başarı gösterilirken, sistemin ataklara karşı gürbüz olmadığı görülüyor. Güçlü ürün seçimi oranı 0,05 olarak ayarlandığında ve on ürün listesi hazırlandığında, saldırgan bu listeye altı yanlış ürün ekleyebilir. Eklenen bu ürünlerden en az ikisinin ilk beşte yer aldığı çıkarımı yapılabilirken, bu ürünlerin ikinci ve üçüncü sıralarda yer alma ihtimali de vardır. Listede yer alan ürünlerin çeşitlilik, yenilik ve şans eser bulma kriterlerindeki değişimlerde azalma olduğu ve farklı seçim durumlarında en yüksek noktaya ulaştığı görülmektedir.

YM20 için sonuçlar incelendiğinde (bkz. Tablo 5.18), YM10 veri setinde olduğu gibi yüksek saldırı başarısının olmadığı, hedef ürünlerin MUP-U olarak seçilmesiyle en başarılı saldırının gerçekleştirildiği görülmektedir. Saldırı başarılı olursa, saldırgan ürün listesine en fazla iki ürün ekler ve bu yanlış ürünleri kullanıcıya öneri olarak sunar. Yanlış ürünlerin yer aldığı bu listedeki diğer performans kriterlerinin değerlendirilmesinde çeşitlilik, yenilik ve tesadüf metriklerinde saldırı öncesi durumla yaklaşık sonuçlar alındığı görülmektedir. Hedef ürün MUP-G olarak seçilirse, çeşitlilik ve yenilik kriterlerinde en başarılı sonuçlar sırasıyla %25, %46 iken MUP-U kriterinde ise şans eseri bulma konusunda en yüksek sonucu vermektedir.

Tablo 5.19. *CrispMC-EntARMCF algoritmasının AS seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM10		**AS 0,01**			**AS 0,05**			**AS 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,1982	0,2717	0,3164	0,0324	0,0695	0,0991	0,0404	0,079	0,1008
	WRV	1,5877	4,4569	13,717	0,2546	1,5176	3,9465	0,312	1,8211	4,1972
	Çeşitlilik	**0,4096**	**0,3737**	**0,3702**	0,2414	0,164	0,1464	0,2046	0,1313	0,1066
	Yenilik	**0,5958**	**0,584**	**0,584**	0,4479	0,4028	0,3977	0,4291	0,3719	0,3561
	Şans eseri bulma	0,2858	0,2311	0,2045	**0,3651**	0,2744	0,2162	**0,3157**	0,2134	0,1515
MUP-U	AvgHR	0,2639	0,2936	0,289	0,264	0,2903	0,2872	0,2524	0,2912	0,2941
	WRV	2,1424	7,0334	13,131	2,1451	6,9553	13,065	2,0339	6,9511	13,175
	Çeşitlilik	0,3396	0,3067	0,3025	**0,3154**	**0,2769**	**0,2705**	**0,2964**	**0,2532**	**0,2413**
	Yenilik	0,5077	0,469	0,4606	**0,4934**	**0,4504**	**0,441**	**0,4802**	**0,437**	**0,4253**
	Şans eseri bulma	**0,3104**	**0,3084**	**0,3187**	0,2887	**0,2817**	**0,2907**	0,2747	**0,262**	**0,2665**
MUP-C	AvgHR	**0,5075**	**0,6193**	**0,6549**	**0,4959**	**0,6112**	**0,6483**	**0,475**	**0,6011**	**0,6403**
	WRV	**3,9141**	**13,171**	**24,786**	**3,8273**	**12,975**	**24,695**	**3,7129**	**12,886**	**24,106**
	Çeşitlilik	0,2325	0,1671	0,1473	0,2292	0,1623	0,1403	0,2227	0,156	0,1355
	Yenilik	0,4571	0,409	0,3961	0,4509	0,4009	0,3878	0,4459	0,3938	0,3798
	Şans eseri bulma	0,2257	0,1673	0,1493	0,2183	0,1569	0,1388	0,2137	0,1479	0,1287

Tablo 5.20. *CrispMC-EntARMCF algoritmasının AS seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM20		**AS 0,01**			**AS 0,05**			**AS 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	**0,0937**	0,1243	0,1395	**0,0906**	0,1221	0,1387	**0,0987**	0,1199	0,1402
	WRV	**1,2559**	4,5179	8,7105	**1,2287**	4,3786	8,6461	**1,3185**	4,5063	8,8214
	Çeşitlilik	0,2393	**0,2519**	0,2591	**0,2422**	0,2517	0,2621	**0,2449**	**0,2574**	**0,2639**
	Yenilik	**0,4621**	**0,4551**	**0,4523**	**0,4575**	**0,4513**	**0,4482**	**0,4521**	**0,447**	**0,4458**
	Şans eseri bulma	0,3451	0,3108	0,2947	0,3453	0,3096	**0,2923**	**0,3418**	**0,3083**	**0,2896**
MUP-U	AvgHR	0,0838	**0,1378**	**0,1783**	0,0875	**0,1428**	**0,1778**	0,0835	**0,1365**	**0,185**
	WRV	1,1375	**5,1342**	**11,348**	1,1456	**5,2792**	**11,476**	1,1213	**5,1299**	**11,752**
	Çeşitlilik	0,2129	0,2269	0,2328	0,2175	0,2341	0,241	0,2201	0,2347	0,2422
	Yenilik	0,4197	0,4131	0,4087	0,4147	0,4086	0,4053	0,4134	0,4059	0,4018
	Şans eseri bulma	0,3244	0,2956	0,2842	0,3182	0,2887	0,2801	0,3153	0,2855	0,2722
MUP-C	AvgHR	0,07	0,1239	0,1472	0,0746	0,1219	0,1494	0,0654	0,1208	0,145
	WRV	0,9651	4,4961	9,7207	0,9678	4,593	10,454	0,832	4,5492	9,4739
	Çeşitlilik	**0,2398**	0,2514	**0,2607**	0,2377	**0,2554**	**0,2641**	0,2434	0,2538	0,2636
	Yenilik	0,4584	0,4515	0,4482	0,4528	0,4476	0,4447	0,4515	0,4452	0,4416
	Şans eseri bulma	**0,3589**	**0,317**	**0,3013**	**0,3461**	**0,3088**	0,2916	0,3381	0,3004	0,2857

Tablo 5.21. *CrispMC-EntARMCF algoritmasının NR seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM10		**NR 0,01**			**NR 0,05**			**NR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,206	0,2988	0,346	0,0566	0,0918	0,1123	0,2629	0,3593	0,4116
	WRV	1,6531	6,8331	14,900	0,4562	2,1949	4,9898	2,1238	8,3513	17,634
	Çeşitlilik	**0,4027**	**0,3742**	**0,3671**	0,1672	0,1198	0,1067	**0,3327**	**0,305**	**0,3067**
	Yenilik	**0,5853**	**0,5745**	**0,5741**	0,4005	0,3391	0,3227	**0,5434**	**0,5336**	**0,5387**
	Şans eseri bulma	0,2811	0,2218	0,196	**0,3604**	**0,2873**	0,2334	0,2572	0,1917	0,1647
MUP-U	AvgHR	0,262	0,3012	0,2945	0,2621	0,3127	0,3154	0,2698	0,3247	0,3341
	WRV	2,1135	7,1588	13,364	2,1014	7,4265	14,111	2,1344	7,483	14,830
	Çeşitlilik	0,3164	0,2763	0,2708	**0,2768**	**0,2212**	**0,2096**	0,2531	0,1907	0,1792
	Yenilik	0,4976	0,4582	0,4509	**0,4832**	**0,4391**	**0,4315**	0,4734	0,4257	0,4188
	Şans eseri bulma	**0,3066**	**0,2963**	**0,3057**	0,2833	0,2583	**0,2626**	**0,2652**	**0,2308**	**0,2332**
MUP-C	AvgHR	**0,4776**	**0,5881**	**0,6217**	**0,4329**	**0,5523**	**0,5829**	**0,4086**	**0,5363**	**0,5699**
	WRV	**3,7105**	**12,581**	**23,823**	**3,3327**	**12,006**	**22,843**	**3,1959**	**11,752**	**22,015**
	Çeşitlilik	0,2349	0,1729	0,1515	0,234	0,17	0,15	0,2267	0,1653	0,15
	Yenilik	0,4575	0,4126	0,4002	0,4591	0,4095	0,3985	0,4552	0,4085	0,4006
	Şans eseri bulma	0,2388	0,1825	0,1659	0,2462	0,1869	0,1714	0,2422	0,1824	0,169

Tablo 5.22. *CrispMC-EntARMCF algoritmasının NR seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM20		**NR 0,01**			**NR 0,05**			**NR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	**0,0939**	0,1218	0,1438	**0,0939**	0,1215	0,143	**0,0976**	0,1276	0,1421
	WRV	**1,1517**	4,1464	9,0382	**1,3261**	4,4467	9,0402	**1,2507**	4,5004	9,0121
	Çeşitlilik	**0,2342**	**0,2504**	**0,2583**	0,2298	**0,2522**	0,2559	**0,2385**	0,2506	**0,2567**
	Yenilik	**0,4634**	**0,4578**	**0,4546**	0,4626	**0,458**	**0,4545**	0,4625	**0,4567**	**0,454**
	Şans eseri bulma	0,3445	0,3126	0,2946	0,3437	0,3137	0,2966	0,3447	0,311	0,2947
MUP-U	AvgHR	0,0895	**0,1458**	**0,1777**	0,0797	**0,1393**	**0,179**	0,091	**0,15**	**0,1757**
	WRV	1,0797	**5,5345**	**10,929**	1,0154	**4,8853**	**11,697**	1,1552	**5,1606**	**11,033**
	Çeşitlilik	0,2083	0,2226	0,2298	0,213	0,2237	0,2298	0,2168	0,2227	0,2286
	Yenilik	0,4238	0,417	0,4121	0,4246	0,4169	0,4125	0,4227	0,4173	0,4133
	Şans eseri bulma	0,3266	0,2971	0,2878	0,329	0,2896	0,2865	0,3255	0,2964	0,2883
MUP-C	AvgHR	0,0721	0,1205	0,1504	0,0672	0,1204	0,1508	0,0755	0,1216	0,15
	WRV	0,9743	4,4596	9,9027	0,9039	4,6047	10,109	0,9913	4,6289	9,8917
	Çeşitlilik	0,2328	0,2501	0,2564	**0,2392**	0,2498	**0,2571**	0,2311	**0,251**	0,2565
	Yenilik	0,462	0,4548	0,4506	**0,4628**	0,4559	0,4509	**0,4628**	0,4543	0,4503
	Şans eseri bulma	**0,367**	**0,3247**	**0,3055**	**0,3687**	**0,3265**	**0,3057**	**0,3656**	**0,3252**	**0,3055**

Tablo 5.23. *CrispMC-EntARMCF algoritmasının SR seçim kriterine göre YM10 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM10		**SR 0,01**			**SR 0,05**			**SR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,185	0,2688	0,3143	0,0442	0,0749	0,1029	0,2062	0,2904	0,3396
	WRV	1,4929	6,3461	13,592	0,3448	1,7118	4,4863	1,6483	6,8047	14,543
	Çeşitlilik	**0,4164**	**0,3804**	**0,3753**	0,2802	0,2042	0,1865	**0,3713**	**0,3325**	**0,334**
	Yenilik	**0,5978**	**0,5873**	**0,5868**	0,4704	0,4366	0,4325	**0,5472**	**0,5306**	**0,5337**
	Şans eseri bulma	0,289	0,2314	0,205	**0,3529**	0,2737	0,2273	0,2773	0,2043	0,1711
MUP-U	AvgHR	0,2699	0,2927	0,289	0,2628	0,2969	0,2923	0,267	0,2975	0,3014
	WRV	2,2034	7,1112	13,161	2,1565	7,124	12,946	2,1654	7,0643	13,368
	Çeşitlilik	0,3451	0,3104	0,3092	**0,329**	**0,2904**	**0,2876**	0,3084	0,2636	0,2594
	Yenilik	0,51	0,4714	0,4631	**0,4994**	**0,4574**	**0,4501**	0,4871	0,4442	0,4381
	Şans eseri bulma	**0,3128**	**0,3106**	**0,3211**	0,298	**0,289**	**0,3**	**0,2825**	**0,2682**	**0,2779**
MUP-C	AvgHR	**0,5101**	**0,6192**	**0,6552**	**0,5037**	**0,6156**	**0,6534**	**0,4892**	**0,612**	**0,6484**
	WRV	**3,9698**	**13,131**	**25,079**	**3,9253**	**13,075**	**24,983**	**3,774**	**13,087**	**24,653**
	Çeşitlilik	0,2323	0,1695	0,1468	0,2297	0,1646	0,1415	0,2287	0,1586	0,137
	Yenilik	0,4568	0,4097	0,3968	0,4517	0,4024	0,3885	0,4492	0,3952	0,382
	Şans eseri bulma	0,2266	0,1686	0,1503	0,2191	0,1586	0,1391	0,2154	0,1483	0,1307

Tablo 5.24. *CrispMC-EntARMCF algoritmasının SR seçim kriterine göre YM20 veri kümesindeki saldırı sonrası performansı*

CrispMC-EntARMCF-YM20		**SR 0,01**			**SR 0,05**			**SR 0.1**		
		En iyi-5	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*	*En iyi-5*	*En iyi-10*	*En iyi-15*
MUP-G	AvgHR	0,082	0,1193	0,1418	0,0914	0,1155	0,1458	**0,0842**	0,1205	0,1392
	WRV	**1,1603**	4,1535	9,4988	**1,1881**	4,053	8,906	**1,0923**	4,5437	9,4098
	Çeşitlilik	**0,2399**	0,2512	**0,2593**	0,2367	**0,2531**	0,2599	**0,2458**	0,2496	0,2614
	Yenilik	**0,4631**	**0,4579**	**0,4542**	0,4592	**0,4556**	**0,4518**	**0,4561**	**0,4539**	**0,4494**
	Şans eseri bulma	0,348	0,3136	0,2951	0,3457	0,3138	0,2927	0,3488	0,3129	0,2953
MUP-U	AvgHR	**0,0855**	**0,1392**	**0,1712**	**0,0919**	**0,1338**	**0,167**	0,0822	**0,1356**	**0,1754**
	WRV	1,0528	**4,9854**	**11,024**	1,1047	**4,6403**	**10,65**	1,0865	**4,9839**	**11,471**
	Çeşitlilik	0,2173	0,224	0,2288	0,2154	0,2223	0,2298	0,2207	0,2261	0,2378
	Yenilik	0,4219	0,4172	0,4122	0,4223	0,4167	0,4114	0,4202	0,4131	0,4098
	Şans eseri bulma	0,3271	0,2978	0,2883	0,324	0,2984	0,2862	0,3232	0,2932	0,2803
MUP-C	AvgHR	0,0721	0,1197	0,1461	0,0699	0,1176	0,1442	0,0628	0,1161	0,1381
	WRV	0,9811	4,7719	10,014	0,9102	4,369	9,3287	0,821	4,3732	9,2508
	Çeşitlilik	**0,2399**	**0,2522**	0,2572	**0,2391**	0,2509	**0,2613**	0,2316	**0,2512**	**0,2617**
	Yenilik	0,4609	0,4542	0,4511	**0,4602**	0,452	0,4489	0,4557	0,4494	0,4468
	Şans eseri bulma	**0,3677**	**0,3255**	**0,3067**	**0,3602**	**0,3208**	**0,3022**	**0,3533**	**0,3136**	**0,2974**

Tablo 5.19, 5.21, 5.23, CrispMC-EntARMCF yönteminin farklı hedef ürün ve güçlü ürün seçim kriterlerine göre PIA-MT saldırı gürbüzlük analizinin sonuçlarını göstermektedir. YM10 veri kümesinde güçlü ürün seçimi AS olarak belirlendiğinde sistemin manipüle edilmesinin oldukça başarılı olduğu görülmektedir (bkz. Tablo 5.19). Kriter tabanlı hedef ürün seçimi yapıldığında, kullanıcıya sunulan on üründen oluşan listedeki altı ürünün saldırganın eklemek istediği öğeler olduğu söylenebilir. Ayrıca en az iki yanlış ürün ilk beşte yer alabilirken bu ürünler 2., 3. ve 4. sıralarda kullanıcıya sunulabilmektedir. Saldırı başarılı olduğu durumlarda, listedeki ürün çeşitliliğinde, yeniliğinde ve kullanıcıyı şaşırtacak ürünlerin varlığında saldırı öncesi duruma göre azalma vardır. Çeşitlilik ve tesadüf oranları oldukça düşükken, yenilik oranı yüzde 45' tir. Bu metriklerin farklı hedef ürün seçim yöntemlerinde yüksek başarı oranları sağladığı görülmektedir.

Tablo 5.21' de, güçlü ürün seçimi NR olarak belirlendiğinde ve Tablo 5.9' daki gibi MUP-C kriteri seçildiğinde, saldırı başarısının %60' lara ulaştığı gösterilmiştir. Listedeki yanlış ürün sayısı, 15 ürünlük bir liste için hazırlandığında dokuz ürün olduğu şeklinde yorumlanabilir. *WRV* değerine göre bu yanlış ürünlerin de yüksek bir öncelikle kullanıcıya sunulmasının muhtemel olduğu söylenebilir. Ayrıca saldırının başarılı olduğu durumlarda çeşitlilik ve şans eseri bulma ölçütlerinde %20 başarının olduğu ve kullanıcıya sunulan on adet manipüle edilmiş ürün listesindeki iki ürünün farklı ve kullanıcıyı şaşırttığı görülmektedir. Yenilik kriterinde bu oran %45 iken yeni ürün sayısı dörttür. Bu üç kriterin farklı hedef ürün seçimlerinde en yüksek başarıyı sağladığı durumlar, 10 ürünlük listede kullanıcı için 4 ürün çeşit, kullanıcının bilmediği ama memnun kaldığı ürün sayısı üç ve kullanıcı için yeni ürün altı adettir. Tablo 5.23' te, güçlü ürün seçimi SR yapıldığında, aynı benzer sonuçların elde edildiğini gösterilmiştir. Saldırı başarısının %65' lere ulaştığı noktalarda çeşitlilik ve şans eseri bulma ölçütlerinde bir azalma, yenilik kriterinde ise %45 oranında artma söz konusudur.

YM20' den alınan sonuçlar incelendiğinde sistemin saldırılara karşı oldukça gürbüz olduğu görülüyor. Listede yer alan hedef ürünlerin oranının, maksimum %20' lere ulaştığı tablolarda görülmektedir. Tablo 5.18 incelendiğinde, yanlış ürünler listeye dahil edilmesi, en çok hedef ürün MUP-U olarak seçildiğinde ortaya çıkar. Güçlü ürün seçim boyutu 0,01 olarak ayarlandığında ve beş ürün listesi hazırlandığında, hedef ürünlerin kullanıcıya en son sırada sunulduğu sonucuna varılabilir. Benzer sonuçlar Tablo 5.20,

5.22 ve 5.24 gösterilmekte ve her seçim kriterinde saldırının sistemde en başarılı olduğu zaman yakın sonuçların olduğu görülmektedir. Her biri için benzer yorumların yapılabileceği söylenebilir.

PIA-MT saldırısında YM10 veri setinde CrispMC-EntARMCF en iyi-n algoritmasının, kriter bazlı hedef ürün seçimlerinin genel olarak başarılı olduğu görülmektedir. Özellikle güçlü ürün seçimi ID ve SR olduğunda başarı oranı %66' ya ulaşmaktadır. Hatta ilk etapta kullanıcıya yanlış ürünlerin sunulma ihtimalinin yüksek olduğu çıkarımı bile yapılabilir. Ancak aynı başarı YM20 veri seti için mevcut değildir ve veri seti üzerindeki saldırı başarısı maksimum %20' ye kadar çıkabilmektedir. Dolayısıyla bu veri setinin YM10' a göre daha gürbüz olduğu söylenebilir.

5.4. Değerlendirmeler

Öneri sistemi tekniklerinin çoğu, öneriler oluşturmak için genel bir değerlendirme kullanır. Ancak bu değerlendirme bir sınırlılık olarak kabul edilebilir. Öneri oluşturma/ürün listesi hazırlama aşamasında kullanıcılar tercihlerini ayrıntılı olarak ifade edemeyebilir ve doğru sonuçlar vermeyebilir. Bu nedenle, bu sorunu hafifletmek için, öneriler üretirken çok ölçütlü değerlendirmeler kullanarak öneri sisteminin performansının doğruluğunu artırmak için çok ölçütlü öneri sistemleri (multi-criteria recommender systems, ÇKÖS) geliştirilmiştir. Kullanıcının beğenebileceği ürünler listesi hazırlamayı amaçlayan çok ölçütlü en iyi-n öneri sisteminde, ürünün çok yönlü olarak tercihlerini ifade ederek daha doğru ve kişiye özel öneriler üretilir.

Ancak, bu çok ölçütlü en iyi-n öneri sistemleri, kötü niyetli kullanıcılar tarafından manipüle edilebilir. Bu nedenle yapılan çalışmalarda bu sistemlerin şilin saldırılarına karşı gürbüzlük analizinin yapılmadığı görülmektedir. Bu bölümde, ürün tabanlı çok ölçütlü en iyi-n öneri sistemlerinin şilin saldırılarına karşı gürbüzlük analizi yapılmıştır. Literatürdeki çalışmalar, ürün tabanlı öneri sistemlerinin saldırılara karşı oldukça dirençli olduğunu belirtse de güçlü ürün ve çoklu hedef (PIA-MT) modelinin ürün tabanlı algoritmalarının, itme saldırılarına karşı oldukça savunmasız olduğu gösterilmiştir. Ayrıca bu güçlü ürünler, komşuluk seçimi ve tahmin sürecinde diğer ürünleri önemli ölçüde etkilemektedir.

Bu bölümde, kullanıcıya sunulan en iyi-n öneri listesini manipüle ederek, kullanıcının memnun olmadığı ürünleri sunmayı amaçlanmaktadır. Bu nedenle, önceki

bölümde önerdiğimiz üç farklı çok ölçütlü en iyi-n öneri algoritması (BASEMCCF, IFMC-EntARMCF, CrispMC-EntARMCF), Yahoo!Movies veri seti, PIA-MT saldırısına karşı gürbüzlük analizi ve farklı performans ölçümleri yapılmıştır (Kaya & Kaleli, 2022) .

Üç farklı öneri algoritması üzerinde gerçekleştirilen PIA-MT saldırısının sonuçları incelendiğinde BASEMCCF yönteminin diğer iki yönteme göre daha sağlam olduğu ve listedeki yanlış madde sayısının en fazla iki ürün olduğu görülmektedir. PIA-MT, birden fazla hedef ürün MUP-C kriterine göre ve güçlü ürün seçimi de ID olarak yapıldığında atak oldukça başarılı iken bu seçimler sırasıyla MUP-C ve AS olduğunda sistem oldukça gürbüz olmaktadır. Ancak listedeki ürünlerin çeşitliliğine, yeniliğine ve tesadüf durumuna bakıldığında yüksek başarı elde edilmekte ve listedeki yanlış ürünün ilk sırada olduğu çıkarımı yapılabilmektedir. Kullanıcının beğenebileceği ürünleri başarıyla öneren IFMC-EntARMCF algoritması bu saldırıya karşı savunmasız görünüyor. Özellikle hedef ürün seçimi olarak MUP-U belirlendiğinde saldırının başarısı %60' a ulaşmaktadır. Saldırının başarılı olduğu noktalarda ilk sırada yanlış ürünlerin kullanıcıya sunulma olasılığı bulunurken, diğer performans kriterlerinin başarı oranının düşük olduğu görülmektedir. CrispMC-EntARMCF algoritmasında hedef ürün MUP-C olarak seçildiğinde de benzer bir başarı elde edilmektedir. Aynı zamanda listede ilk sıralarda yer alabilecek yanlış ürünlerin çeşitliliğinin, yeniliğinin ve şans eseri bulma durumlarının azaldığı görülmektedir. Sonuç olarak, bir önceki bölümde önerilen algoritmalar oldukça başarılı öneriler üretirken, saldırılara karşı savunmasızdır. Bu sistemleri kullanmak isteyen firmaların/kullanıcıların uygun bir tespit yöntemi uygulaması gerekmektedir.

6. ÇOK ÖLÇÜTLÜ ÖNERİ SİSTEMLERİ İÇİN YENİ SINIFLANDIRMA TABANLI ŞİLİN ATAK TESPİT YÖNTEMİ

Bir önceki bölümde, çok ölçütlü en iyi-n öneri sistemlerinin (Kaya & Kaleli, 2022) PIA atak modeline karşı savunmasız olduğu görülmüştür. Dolayısıyla bu bölümde, çok ölçütlü sistemler üzerinde gerçekleştirilen bu PIA atağın tespiti yapılmıştır.

6.1. Giriş

Son yıllarda dijital platformların yaygınlaşması, internette mevcut bilgilerin çarpıcı bir şekilde artmasıyla sonuçlanmış, bu da bu kadar büyük veri koleksiyonda ilgili içeriği kolayca bulamadıkları için bireylerin karar verme sürecini daha karmaşık hale getirmektedir. Tavsiye sistemleri, bilgi erişim sistemlerinin öne çıkan bir varyasyonu olarak, geçmişte ilgi alanlarına göre alakasız olanları filtrelerken, kullanıcılara uygun ürünler/hizmetler önererek bu tür aşırı bilgi yükleme sorununun üstesinden gelmek için oldukça faydalı yapay zekâ çözümleridir (Isinkaye, Folajimi, & Ojokoh, 2015).

İF algoritmaları, tahmine dayalı doğruluk ve çevrimiçi performansta önemli başarıları nedeniyle öneri sistemlerinde bireysel öneriler sağlamak için en çok kullanılan yaklaşımlardır (Sharma, Gopalani, & Meena, 2017). Geçmişte benzer zevklere sahip bireylerin gelecekte de benzer tercihlere sahip olacağı ilkesine dayanarak, kullanıcılar tarafından deneyimlenmemiş öğeler için bir derecelendirme tahmininde bulunurlar veya en çok tercih edilen öğelerin sıralı bir listesini önerirler. İF algoritmaları genellikle bu tür öneri görevlerini kullanıcılar/ürünler (Aggarwal, 2016) arasındaki benzerliklere dayalı olarak konumlanmış komşulukları kullanarak veya yapılandırılmış bir tercih verisi modeli (Bokde, Girase, & Mukhopadhyay, 2015) kullanarak gerçekleştirir. Bunu yaparken, tipik olarak, belirli bir derecelendirme ölçeğindeki ürünler üzerinde kullanıcıların geçmiş seçimlerini içeren bir kullanıcı ürün derecelendirme matrisi üzerinde çalışırlar.

İF algoritmalarının başarısı genellikle kullanıcılar/ürünler arasındaki korelasyonun, geçmiş tercihleri (Liu, Hu, Mian, Tian, & Zhu, 2014) göz önünde bulundurularak ne kadar doğru belirlendiğine bağlıdır. Bu nedenle, ayrıntılı geri bildirim alarak sistemin kişiselleştirme yeteneğini geliştirmesine yardımcı olacaktır. ÇKÖS olarak da adlandırılan bu tür sistemler, bireylerin bir ürünü genel bir tercihle birlikte olası çoklu alt perspektifler üzerinde değerlendirmesine olanak tanır (Adomavicius & Kwon, 2007) (Yalcin & Bilge, 2020) (Manouselis & Costopoulou, 2007). Bu nedenle, ÇKÖS' leri, kullanıcı profilleri

arasındaki korelasyonları doğru bir şekilde yakalama ve böylece daha iyi kişiselleştirilmiş öneriler üretme konusunda daha yeteneklidir, çünkü iki kullanıcı, genel puanlarına göre temel olarak birbirine oldukça benzer görünseler bile, öğelerin alt yönleriyle ilgili görüşleri dikkate alındığında oldukça farklı olabilir. Bu sistemler genellikle, tavsiye oluşturma aşamasında fonksiyon-tabanlı (*function-based*) veya benzerlik-tabanlı (*similarity-based*) yaklaşımları izleyerek bu tür çok ölçütlü derecelendirme bilgisinden yararlanır; ilki, alt kriter tercihleri ve genel derecelendirmeler arasındaki ilişkilere dayalı olarak tanımlanan bir toplama işlevi kullanırken, ikincisi, kullanıcılar/ürünler arasında genel benzerlikler elde etmek için her bir kriter için tahmini benzerlik değerlerini toplamayı amaçlar (Adomavicius, Manouselis, & Kwon, 2011) .

Geleneksel öneri sistemleri için, toplanan verilerin kalitesi, doğruluk açısından nitelikli tavsiyelere ulaşmak için hayati önem taşır. Ancak, bu sistemlerin, bazı kötü niyetli kullanıcıların veya rakip hizmet sağlayıcıların, tahmin edilen oyları etkilemek veya amaçları adına sistemin performansını azaltmak için derecelendirme matrislerine sahte profiller eklemek için kullandığı bazı saldırılara karşı oldukça savunmasız olduğu bilinen bir olgudur (Gunes, Kaleli, Bilge, & Polat, 2014) (Si & Li, 2020). Bu tür şilin saldırıları, hedeflenen ürünler üzerindeki saldırı mekanizmalarına göre genellikle iki türe ayrılır (Mobasher, Burke, Bhaumik, & Williams, 2007). İtme atak olarak da adlandırılan saldırı türü, hedeflenen öğelerin popülaritesini artırmayı amaçlar, böylece öneri sistemleri onları daha fazla tavsiye edilebilir olarak değerlendirirken, diğeri ise çekme olarak adlandırılır ve önerilme olasılığını azaltmak için popülerliklerini azaltmayı amaçlamaktadır. Böyle bir itme saldırısına maruz kalan herhangi bir sistem, öneri algoritmalarının iyi bilinen popülerlik yanlılığı nedeniyle, kaçınılmaz olarak, birkaç seçilmiş hedef ürün baskın olduğu öneri listeleri sağlar (Jannach, Lerche, Kamehkhosh, & Jugovac, 2015). Öte yandan, bir çekme saldırısı, hedef ürünler son zamanlarda ilgi çekseler bile, bu ürünler hak ettiği ilgiyi göremeyeceği öneriler üretmeye yol açar.

Sistemlerin bu tür manipülasyonlardan korunması ve böylece daha sağlam ve adil bir sistem elde etmek için, son zamanlarda yapılan birçok çalışma, sınıflandırma-tabanlı (Samaiya, Raghuwanshi, & Pateriya, 2019) (Bansal & Baliyan, 2021) (Batmaz, Yilmazel, & Kaleli, 2020), kümeleme-tabanlı (Cai & Zhang, 2019) (Li, ve diğerleri, 2021) veya istatistiksel yaklaşımları (Gao, Yuan, Ling, & Xiong, 2014) (Xia, ve diğerleri, 2015) kullanarak pratik saldırı profili tespit prosedürleri geliştirmeyi amaçlamıştır. Ancak, bu

algılama yaklaşımları tipik olarak tek kritere dayalı öneri sistemleri için tasarlanmıştır ve dahili mekanizmaları nedeniyle çok ölçütlü ortama doğrudan uygulanamazlar. Daha karmaşık tavsiye stratejileri ve tercih verilerinin yapısı göz önüne alındığında, ÇKÖS' leri, özellikle çok sayıda kriterin varlığında, saldırganlar tarafından hangi alt kriter derecelendirmelerinin manipüle edildiğini tespit etmek daha zor hale geldiğinden, tespit edilmesi ve kaçınılması daha zor olan daha manipülatif şilin girişimlerine maruz kalabilir (Turk & Bilge, 2019). Bununla birlikte, ÇKÖS' leri için bu saldırılara karşı herhangi bir savunma mekanizması henüz oluşturulmamıştır.

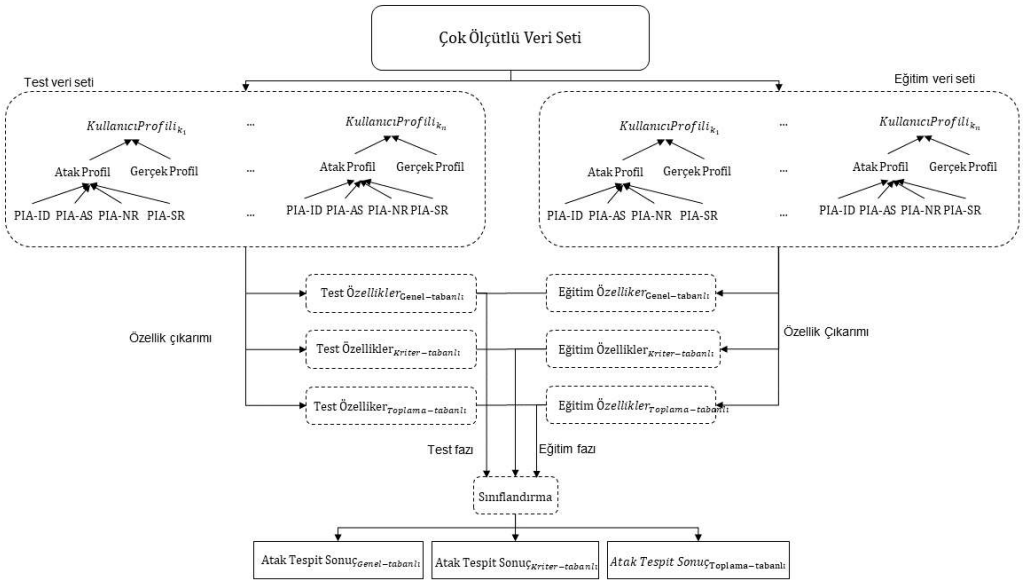
Bu nedenle, bu çalışmada, ilk olarak, PIA (Seminario, 2013) olarak adlandırılan, bir önceki bölümde çok ölçütlü sistemlere göre önerilen ve komşuluk seçim süreci üzerinde önemli bir etkisi olan ürünleri hedef alan, oldukça etkili bir saldırı türünü ele alıyoruz. Ardından, çeşitli genel ve model tabanlı ürün niteliklerine dayalı olarak atak profillerini filtreleyen bir sınıflandırma yaklaşımı öneriyoruz. Bu özniteliklerden bazıları literatürde mevcut yöntemlerle üretilirken, bazıları da genellikle ürün popülerliği ve kullanıcı karakteristiğine dayalı özelliklerdir.

6.2. Çok Ölçütlü Oy Tabanlı Sistemler İçin Şilin Atak Tespit Yöntemi

Bu bölümde, ÇKÖS' lerine yönelik şilin saldırısı tespit yaklaşımımızı tanıtıyoruz. Önce çerçevemiz hakkında genel bilgiler veriyoruz ve ardından kullanıcı profillerine dayalı olarak gerçekleştirilen özellik çıkarma işlemini kapsamlı bir şekilde açıklıyoruz.

6.2.1. Önerilen yaklaşımın şeması

Çok ölçütlü derecelendirme tabanlı sistemler için önerilen şilin saldırısı algılama yöntemimiz, Şekil 6.1'de gösterildiği gibi aşağıdaki dört ana aşamadan oluşur: *(i)* çok ölçütlü veri setinin test ve eğitim olarak ayrılması, *(ii)* kullanıcı profillerinden özniteliklerin çıkarılması, *(iii)* eğitim setinin uygun sınıflandırıcı yöntemlerle sınıflandırılması ve son olarak *(iv)* elde edilen tespit sonuçlarının test edilmesidir.



Şekil 6.1. *ÇKÖS için önerilen şilin saldırısı tespit mekanizmasının şeması*

Ayrıntılı olarak açıklandığında ilk olarak, çok ölçütlü verileri eğitim ve test setlerine ayrılır. Bu veri setleri farklı atak modellerinden üretilen sahte kullanıcıların, veri setinde yer alan gerçek kullanıcılarla birleştirilmesi oluşturulur. Ancak burada kritik nokta, eğitim veri setini oluşturma aşamasında atak profilleri, uygulanan saldırı modelinin tüm varyantları (PIA-IS, PIA-AS, PIA-NR, PIA-SR) kullanılarak karma bir saldırı profilleri yapılır. Ancak bu durum test veri seti için geçerli değildir. Test veri setinde her bir atak modeli için tek tek sahte profiller oluşturulur. Sonraki süreçte, seçilen farklı boyutlardaki bu saldırı profilleri, gerçek kullanıcılara eklenerek hem test hem de eğitim setleri oluşturulur. Ancak sisteme enjekte edilecek saldırı profilleri, (Adomavicius, Manouselis, & Kwon, 2011)' de tanıtılan yaklaşım benimsenerek her bir kriter için ayrı ayrı gerçekleştirilmektedir.

Bir sonraki aşama olan özellik çıkarımında hem eğitim hem de test setlerinde kullanıcılara sunulan özellikleri karakterize etmek için üç farklı senaryo benimsenerek, kullanıcı profillerinden 15 özellik çıkartılır. İlk senaryo, sistemi tek bir kriter olarak ele alıp genel değerlendirmeyi dikkate alarak ilerlerken, ikincisi ise her bir kriteri ayrı ayrı ele alarak özellik çıkarma işlemini gerçekleştirmektedir. Öte yandan, son senaryoda, regresyon analizi kullanarak mevcut tüm kriterler için elde edilen özellikler toplanır. Sonraki aşama olan eğitim fazında, elde edilen eğitim özelliklerine dayalı olarak güçlü bir sınıflandırıcı olan *k*NN sınıflandırma algoritması kullanılır (McRoberts, Tomppo,

Finley, & Heikkinen, 2007). Son olarak, test veri setlerinden alınan öznitelikler, elde edilen eğitilmiş sınıflandırıcıya girdi olarak eklenir ve test aşamasında üç farklı atak tespit sonucu üretilir.

Öznitelik çıkarma ve algılama sonucu oluşturmada benimsenen üç yaklaşımdan ilki olan *genel-tabanlı* (*overall-based*), çok ölçütlü alandaki en güvenilir bilgiyi yansıtır. Böyle bir durum, saldırganın sistemle ilgili bilgilerin çoğunu yalnızca kriteri kullanarak elde etmesine neden olabilir. İkinci durumda, *kriter-tabanlı* (*criteria-based*), her bir alt kriter için ayrı değerler sahte profillerin gerçek kullanıcılara benzemesine yardımcı olabileceğinden, sistemle mevcut tüm bilgileri kapsüllenmiş bir şekilde kullanarak sonuç üretmenin sezgisel olarak en iyi yoludur (Turk & Bilge, 2019). Son durumda, *toplama-tabanlı* (*aggregate-based*), her bir kriter için elde edilen değerler tek bir değer elde etmek için toplanır. Böyle bir yaklaşımın saldırı sürecinde sistem üzerinde çok az etkisi olmasına rağmen, pervasızca derecelendirilmiş alt kriterlerin neden olduğu müdahaleci etkilerin azaltılmasına yardımcı olacaktır (Turk & Bilge, 2019). Sonuç olarak bu üç yöntem kullanılarak elde edilen atak tespit sonuçlarının performans analizi yapılmıştır.

6.2.2. Özellik çıkarımı

Literatürdeki çalışmalarda, kullanıcı profillerinden çıkarılan özellikleri karakterize etmek için farklı metrikler benimsenmiştir (Williams, Mobasher, & Burke, 2007) (Morid, Shajari, & Hashemi, 2014) (Burke, Mobasher, Williams, & Bhaumik, 2006a). Bu özellikler genel olarak iki kategoriye ayrılır: *genel* (*general*) ve *tipe özgü* (*type-specific*) özelliklerdir. Genel nitelikler, saldırı profillerinin istatistiksel imzalarına dayanırken, tipe özgü özellikler, saldırı kalıplarının karakteristik yapıları kullanılarak türetilir. Bununla birlikte, son çalışmalarda, dolgu boyutuna (çalışmamız için seçilen boyut) ve özellik çıkarımı için belirli derecelendirme değerlerine dayalı yeni metrikler önerilmiştir. Bu nedenle, bu çalışmada kullanılan 15 özniteliklerden, sekizi genel, dolgu boyutu ve belirli derecelendirme değerlerine dayalı iken geri kalanı çalışmada önerilen yeni öznitelikleri oluşturmaktadır. Aşağıdaki bölümlerde her bir öznitelik detaylı bir şekilde açıklanmıştır.

6.2.2.1. Genel nitelikler

En iyi-n Komşularla Benzerlik Derecesi (DegSim): Bir profilin en yakın komşularının *k* ortalama benzerlik değeridir. Araştırmacıların varsaydığı gibi, saldırı profillerinin, en yakın 25 komşuyla gerçek kullanıcılar' dan daha fazla benzerliğe sahip olması muhtemeldir (Chirita, Nejdl, & Zamfir, 2005) (Resnick, Iacovou, Suchak,

Bergstrom, & Riedl, 1994). Bu nedenle en yakın komşu sayısı 25 olarak seçilir ve Denklem 6.1 kullanılarak metrik hesaplama yapılır

$$DegSim_u = \frac{\sum_{v \in komşuluklar(u)} W_{u,v}}{k} \tag{6.1}$$

$W_{u,v}$, *u* ve *v* kullanıcıları arasındaki benzerlik değeridir.

En iyi-n Komşularla Ağırlıklı Benzerlik Derecesi (WDegSim): Bu nitelik DegSim' e benzer; ancak, bir kullanıcı için ortak değerlendirmeye aldıkları ürün sayısı düşük olan kullanıcıların komşuluk üzerindeki etkisini azaltmak için, kullanıcılar arasındaki benzerlik hesaplamasına ek olarak, kullanıcılar tarafından ortak puanlanan ürün sayısı dikkate alınarak yapılır (Batmaz, Yilmazel, & Kaleli, 2020). Komşuları bulmak için bu özniteliğin hesaplanmasında bir eşik değeri (*d*) kullanılır. Bu özellik, Denklem 6.2' de verilen formül kullanılarak hesaplanır.

$$\begin{gathered} W_{u,v} = W_{u,v} \times \frac{I_{u,v}}{d}, \;\; if \; |I_{u,v}| < d \\ W'_{u,v} = W_{u,v}, \;\; otherwise \\ WDegSim_u = \frac{\sum_{v \in komşuluklar(u)} W'_{u,v}}{k} \end{gathered} \tag{6.2}$$

d eşik değeri, $I_{u,v}$ *u* ve *v* kullanıcıların ortak değerlendirdiği ürün sayısını temsil eder.

Entropi: Kullanıcıların farklı özellikleri, değerlendirme sırasında tutum ve davranışlarını etkileyebilir. Bu durum, kullanıcılar aynı derecelendirme ölçeğini kullansa bile farklı sonuçlar üretilmesine neden olabilir. Bu nedenle, bu metrik, kullanıcıların değerlendirmelerinde oylama ölçeğindeki derecelendirme değerlerinin dağılımını inceler.

$$Entropi_u = -\sum_{r \in RD} P_{u,r} \log_2(P_{u,r}) \tag{6.3}$$

burada *RD*, tanımlanan derecelendirme alanıdır ve $P_{u,r}$, Denklem 2.4' teki her derecelendirmenin olasılığını temsil eder.

6.2.2.2. Seçilen boyuta dayalı özellikler

Sistem üzerinde kullanıcı profilleri tarafından değerlendirilen ürün sayısı farklı olduğu için farklı özellik çıkarımları yapmak mümkündür. Saldırı profili ürünlerinin sayısının gerçek kullanıcılara göre fazla olması ve özellikle dolgu boyutunun (saldırı

modelimiz için seçilen boyut) oldukça geniş olması, atak profillerinin kolayca tespit edilmesine neden olacaktır. Bu nedenle, sistemdeki ürün sayısının toplam ürün sayısına yakın olması, saldırı ve gerçek kullanıcılar arasındaki farkın kısmi bir karakterizasyonunu sağlar (Yang, Xu, Cai, & Xu, 2016). Benzer şekilde, farklı ürün türleri için değerlendirme sayısı, çeşitli özelliklerin üretilmesine neden olabilir. Örneğin, PIA saldırıları için, güçlü ürün kümeleri, saldırganlar tarafından seçilen ürün seti (I_S) olarak seçilebilir, bu da güçlü ürünlerin oy sayıları gerçek kullanıcı profillerinden çok uzaklaşmasına neden olur. Bu nedenle, araştırmacılar tarafından önerilen güçlü ürünlü seçilen boyut (selected size with power items, SSPI), toplam ürünlerle seçilen boyut (selected size with total items, SSTI), ve kendi içinde güç öğeleri ile seçilen boyut (selected size with power items in itself, SSPII) olarak adlandırılan üç yeni özellik çıkarımı bu atak modeli için kullanılmıştır (Yang, Xu, Cai, & Xu, 2016).

SSTI: Bu metrik, *u* kullanıcısı tarafından derecelendirilen ürün sayısının sistemdeki toplam ürün sayısına oranı olarak ifade edilir.

$$SSTI_u = \frac{\sum_{i=1}^{|I|} O_{r_{u,i}}}{|I|} \tag{6.4}$$

$|I|$ sistemdeki toplam ürün sayısını ifade ederken, *u* kullanıcı *i* ürününe değerlendirmeye aldıysa $O_{r_{u,i}}$ 1 aksi takdirde 0 olur.

SSPI: Kullanıcı *u* tarafından derecelendirilen güçlü ürünlerin sayısının sistemdeki toplam güçlü ürün sayısına oranıdır. Sistemdeki güçlü ürünleri belirlemek için, önceki bölümlerde (Bölüm 5.2.1) açıklanan *ID, AS, NR, SR* gibi dört farklı yaklaşımdan yararlanılır (Yang, Xu, Cai, & Xu, 2016).

$$SSPI_u = \frac{\sum_{i=1}^{|K|} O_{r_{u,i}}}{|K|} \tag{6.5}$$

$|K|$ sistemdeki toplam güçlü ürün sayısını ifade eder.

SSPII: Bu metrik, *u* kullanıcısı tarafından derecelendirilen güçlü ürün sayısının kullanıcı *u* tarafından değerlendirilen ürün sayısına oranı olarak ifade edilir (Yang, Xu, Cai, & Xu, 2016).

$$SSPII_u = \frac{\sum_{i=1}^{|K|} O_{r_{u,i}}}{NR_u} \tag{6.6}$$

NR_u u kullanıcı tarafından değerlendirilmeye alınan ürün sayısıdır.

6.2.2.3. Belli oylara dayalı özellikler

Saldırganın öneri sistemlerindeki farklı saldırı niyetleri olmasından dolayı, saldırı profillerinde seçilen/dolgu ürün setleri maksimum, minimum veya ortalama gibi belirli derecelendirme değerleriyle doldurulabilir. Örneğin PIA-ID, PIA-AS ve PIA-SR saldırı modellerinde seçilen ürün setinin güçlü ürünlerden oluşması ve bu ürünlerin sistem üzerindeki etkisinin yüksek olacağı için maksimum oy değerini alabilir. Benzer şekilde, PIA-NR, güçlü ürünlere verilen oy sayısı ile belirlenirken, bu ürünlerin oy değerleri oldukça düşük ve minimum olabilir. Bu nedenle, belirli oy değerleri gerçek kullanıcılarla saldırı kullanıcıları arasındaki farkı ortaya çıkarma durumundan dolayı araştırmacılar tarafından *FSMAXRI*, *FSMINRI* ve *FSARI* olarak adlandırılan üç yeni metrik önerilmiştir (Yang, Xu, Cai, & Xu, 2016). Ancak bu bölümde atak modelimize uygun olan 2 yeni nitelik kullanılmıştır ve bu özellikler atak modeli doğrultusunda Kendinde Maksimum Derecelendirmeye Sahip Seçilmiş Boyut (Selected Size with Maximum Rating in Itself, SSMAXRI) ve Kendinde Minimum Derecelendirmeye Sahip Seçilmiş Boyut (Selected Size with Minimum Rating in Itself, SSMINRI) olarak güncellenmiştir.

SSMAXRI: Kullanıcı *u* tarafından en yüksek derecelendirme değerine sahip ürünlerin kullanıcı *u* tarafından oylanan ürün sayısına oranını temsil eder (Yang, Xu, Cai, & Xu, 2016).

$$SSMAXRI_u = \frac{\sum_{i=1}^{|I|}(O_{r_{u,i}} = r_{max})}{NR_u} \tag{6.7}$$

burada r_{max}, sistemdeki maksimum derecelendirme değeridir ve $(O_{r_{u,i}} = r_{max})$ kullanıcı *i* ürününü maksimum derecelendirme değeri (r_{max}) ile değerlendirdiyse 1' e, aksi takdirde 0' a karşılık gelir.

SSMINRI: Kullanıcı *u* tarafından en düşük derecelendirme değerine sahip ürünlerin kullanıcı *u* tarafından oylanan ürün sayısına oranını temsil eder (Yang, Xu, Cai, & Xu, 2016).

$$SSMINRI_u = \frac{\sum_{i=1}^{|I|}(O_{r_{u,i}} = r_{min})}{NR_u} \tag{6.8}$$

burada r_{min}, sistemdeki en düşük derecelendirme değeridir ve ($O_{r_{u,i}} = r_{min}$) kullanıcı i ürününü maksimum derecelendirme değeri (r_{min}) ile değerlendirdiyse 1' e, aksi takdirde 0' a karşılık gelir.

6.2.2.4. Önerilen özellikler

İF tabanlı öneri sistemleri, e-ticaret ve bilgi tabanlı sistemlerde önemli bir role sahiptir ve böyle bir sistemin yokluğunda bireyleri görülmesi zor veya başka şekilde zaman alıcı olan ilgili ürünleri bulmaya yönlendirir (Abdollahpouri, Mansoury, Burke, & Mobasher, 2019). Tavsiye verenlerin etkinliğinin önündeki bir engel, popülerlik yanlılığı sorunudur, yani popüler ürünler şiddetle tavsiye edilirken diğer ürünlerin çoğu hiç ilgi görmeyebilir. Ancak bu popüler ürünlerin kötü niyetli kullanıcılar tarafından bilinmesi ve saldırı profillerinde yer alması sistemin başarılı bir şekilde manipüle edilmesini sağlayabilir. Örneğin PIA saldırı modelinde saldırı profilleri sistemde oldukça popüler olan güçlü ürünlerden oluşmaktadır. Bu nedenle, ürünün popülerliği, gerçek ve saldırı profilleri arasındaki farkları değerlendirmek için kullanılabilir. Ancak burada asıl soru sistemde hangi ürünlerin popüler sayılması gerektiğidir. Bu amaçla, iyi bilinen Pareto ilkesini takip ediyoruz (Sanders, 1987), yani veri kümesinde sağlanan tüm derecelendirmelerin %20' sine sahip olan öğeler popüler olarak sınıflandırılır ve katalogdaki geri kalan öğeler popüler olmayanlar olarak etiketlenir.

Ayrıca, güçlü ürünlerden oluşan bu saldırı profillerinde derecelendirme değerlerinin dağılımı, profillerin birbirlerine benzer davranışlar sergilemeleri ve genel kanıdan farklı davranabilmeleri nedeniyle bu farkı belirleyen bir diğer faktördür. Bu nedenle, aşağıda ayrıntılı olarak açıklanan hem ürün popülerliğine hem de derecelendirme değerlerinin dağılımına dayalı yedi yeni özellik öneriyoruz.

Popüler Ürün Sayısı (Number of Popular Items, NrPopItem): u kullanıcısının profilindeki Denklem 6.9 kullanılarak hesaplanabilen popüler öğelerin sayısıdır.

$$NrPopItem_u = \sum_{i \in I_u} \mathbb{1}(i \in P) \tag{6.9}$$

burada, I_u, u kullanıcısının profilindeki derecelendirilmiş ürünler kümesidir ve P, yukarıda açıklanan Pareto ilkesiyle oluşturulmuş popüler ürünler kümesidir.

Popüler Ürünlerin Oranı (Ratio of Popular Items, RatioPopItem): u kullanıcısı tarafından derecelendirilen popüler ürünlerin sayısının, kullanıcı u' profilindeki tüm ürünlerin sayısına oranıdır. Denklem 6.10' da verilen formül kullanılarak hesaplanabilir.

$$RatioPopItem_u = \frac{\sum_{i \in I_u} \mathbb{1}(i \in P)}{|I_u|} \tag{6.10}$$

Derecelendirilen Öğelerin Ortalama Popülerliği (Average Popularity of Rated Items, AvgPopItem): u kullanıcısının profilindeki derecelendirilen ürünlerin ortalama popülerliğidir. Burada ilk önce her bir *i* ürünün aldığı derecelendirme sayısının sistemdeki mevcut tüm derecelendirmelerin sayısına oranı yani Popülarite Derecesini (Popularity Degree, PD_i) hesaplıyoruz. Ardından, Denklem 6.11' de olduğu gibi profilindeki derecelendirilmiş ürünlerin tahmini *PD* değerlerinin ortalamasını alarak her kullanıcı *u* için bir *Ortalama Popülerlik Skoru* hesaplanır.

$$AvgPopItem_u = \frac{\sum_{i \in I_u} PD_i}{|I_u|} \tag{6.11}$$

Gişe Rekortmeni Sayısı (Number of BlockBuster, NrBB): Yakın tarihli bir ilgili çalışma, ürünlerin popülerliğinin her zaman bireyler tarafından çok sevildiği anlamına gelmediği varsayımına dayalı olarak hem popüler hem de çok beğenilen ürünler için gişe rekorları kıran (BlockBuster) terimi tanıtılmıştır (Yalcin & Bilge, 2022). Bu stratejiyi benimseyerek, reyting sayısını ve sağlanan reytinglerin değerini göz önünde bulundurarak her bir *i* ürünü için bir gişe rekortmeni puanı (BB_i) hesapladık ve ardından bunları tekrar Pareto ilkesine göre gişe rekorları kıran veya kırmayan olarak sınıflandırıldı.

İlk olarak, *i* ürünün ortalama oy değeri Denklem 6.12 kullanılarak bulunur. Bir sonraki adımda, BB_i değeri için gereken p_i değeri, popüler olarak değerlendirilen i ürününü değerlendirmeye alan kullanıcı sayısıdır. Ancak önerilen metriği hesaplama adımında μ_i ve p_i değerleri farklı ölçeklerde olduğundan minimum-maksimum normalizasyon yapılır. Ürünler için BB_i değerleri Denklem 6.13 kullanılarak hesaplanır.

$$\mu_i = \frac{\sum_{u \in U_i} r_{u,i}}{|U_i|} \tag{6.12}$$

burada $r_{u,i}$ *i* ürünü için *u* kullanıcısının oy değeridir ve U_i, *i* ürününü değerlendiren kullanıcılar grubudur.

$$BB_i = (\overline{\mu_i} + \overline{p_i})/2 \tag{6.13}$$

$\overline{\mu_i}$ ve $\overline{p_i}$ normalize edilmiş μ_i ve p değerleridir.

Son adımda, *u* kullanıcısı tarafından değerlendirilen gişe rekorları kıran ürünlerin sayısı aşağıdaki denklem kullanılarak bulunur.

$$NrBB_u = \sum_{i \in I_u} \mathbb{1}(i \in BB) \tag{6.14}$$

burada BB, tahmini BB_i puanları üzerinde Pareto (Sanders, 1987) ilkesi uygulanarak oluşturulan gişe rekorları kıran ürünler kümesidir ve I_u, *u* kullanıcısı tarafından değerlendirilen ürünler kümesidir.

Standart Sapma (Standard Deviation, StdDev): u kullanıcısı tarafından verilen oy değerlerinin, kullanıcı ortalamasından sapma oranı hesaplanır. Bu metrik Denklem 6.15 kullanılarak hesaplanır.

$$StdDev_u = \sqrt{\frac{\sum_{i \in I_u}(r_{u,i} - \overline{r_u})}{|I_u|}} \tag{6.15}$$

$\overline{r_u}$, *u* kullanıcısının ortalamasını ifade eder.

Belirsizlik Derecesi (Degree of Uncertainity, DU): Bu değer normalize edilmiş Entropi değeridir. Hesaplaması için Bölüm 2.2' deki Denklem 2.5 kullanılır.

Profil Anomalisi (Profile Anomaly, PA): öneri sistemlerinde yer alan kullanıcılar farklı karakteristik yapılara sahip olmaları değerlendirme esnasındaki tutumları farklı olmaktadır. Kullanıcıların bazıları diğer kullanıcılar ile benzer davranışlar sergileyebilirken, kimi kullanıcılar genel kanattan farklı davranabilirler. Somut olarak, belirli bir kullanıcı için PA_u değeri *u* Denklem 6.16' da verilen formül kullanılarak hesaplanır.

$$PA_u = \frac{\sum_{i \in I_u}(r_{u,i} - \overline{r_i})}{|I_u|} \tag{6.16}$$

$\overline{r_i}$, *i* ürünün ortalama oy değeridir.

6.3. Deneysel Değerlendirmeler

Bu bölümde, önerilen tespit yönteminin etkinliğini ölçmek için çok ölçütlü veri seti üzerinde çeşitli deneyler yapılmıştır. Bu deneyler üç gruba ayrılır. Araştırmanın birinci grubunda, kullanıcı profillerinden çıkarılan özniteliklerin bilgi kazanç değeri

hesaplanmıştır. Deneyin ikinci grubunda, seçilen boyutun tespit yöntemine etkisi incelenirken, önerilen özelliklerin son grupta performans üzerindeki etkisi karşılaştırmalı olarak analiz edilmiştir.

6.3.1. Değerlendirme kriterleri

Önerilen atak tespit yöntemlerinin etkinliğini ölçmek için, literatürde yaygın olarak kullanılan *Sınıflandırma Doğruluğu (Classification Accuracy)*, *Tespit Oranı (Detection Rate)* ve *Yanlış Alarm Oranı (False Alarm Rate)* olmak üzere üç değerlendirme metriği kullanıldı (Yang, Xu, Cai, & Xu, 2016) (Chung, Hsu, & Huang, 2013). Aşağıda, bu metrikleri ayrıntılı olarak açıklandı.

Sınıflandırma Doğruluğu: Doğru bir şekilde sınıflandırılan test kullanıcılarının, tüm test kullanıcılara oranıdır. Denklem 6.17 kullanılarak hesaplama yapılır.

$$Sınıflandırma\ Doğruluğu\ = \frac{\#Doğru\ Sınıflandırma}{\#\ Test\ Kullanıcıları} \tag{6.17}$$

Tespit Oranı: Tespit edilen atak profilleri sayısının, toplam atak profil sayısına bölünmesidir. Denklem 6.18 kullanılarak hesaplama yapılır.

$$Tespit\ Oranı\ = \frac{\#Tespit\ Edilen\ Atak\ Profilleri}{\#\ Atak\ Profilleri} \tag{6.18}$$

Yanlış Alarm Oranı: Atak profili olarak tespit edilen gerçek kullanıcıların, sistemdeki tüm gerçek kullanıcılara oranıdır. Denklem 6.19 kullanılarak hesaplama yapılır.

$$Yanlış\ Alarm\ Oranı\ = \frac{\#Yanlış\ Tespit\ Edilen\ Gerçek\ Profilleri}{\#\ Gerçek\ Kullanıcı\ Profilleri} \tag{6.19}$$

Bu değerlendirme ölçütlerine ek olarak, kullanıcı profillerinden çıkarılan özelliklerin etkisini ölçülmüştür. Bu amaçla, bir özelliğin ne kadar bilgilendirici olduğunu değerlendirmek için en iyi bilinen ve yaygın olarak kullanılan metriklerden biri olan *bilgi kazancı (information gain)* kullanılmıştır (Williams C. A., 2012). Entropi'nin tersi olan *bilgi kazancı*, 0 ile 1 arasındadır ve belirli bir özellik için sınıflandırma sonuçlarından ne kadar değer elde edilebileceğini gösterir. Kazanılan bilgi 1 ise bunun anlamı, bilgi ile sınıf arasında bire bir bağlantının olduğunu ifade ederken; öte yandan, özellikler sınıflardan ne kadar bağımsız olursa bilgi kazancı o kadar düşük olur.

İki sınıflı bir durumda, rastgele bir örneğin *P* ve *N* sınıflarının bilgi kazancı şu şekilde hesaplanır (Williams C. A., 2012):

$$I(p,n) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n} \tag{6.20}$$

p ve *n*, *P* ve *N* sınıflarındaki elementlerin sayısını ifade eder.

S kümesine bir *A* özniteliği uygulamanın, kümeyi (S_1, S_2, ..., S_m) kümelerine bölebileceği düşünülebilir. Denklem 6.21' de verilen formülü kullanarak tüm alt kümeler için beklenen bilgiler hesaplanır:

$$E(A) = \sum_{i=1}^{m} \frac{p_i + n_i}{p+n} I(p_i, n_i) \tag{6.21}$$

p_i , *P* sınıfındaki element iken n_i *N* sınıfındaki elementi temsil eder.

Son olarak, uygulanan öznitelik için bilgi kazancı Denklem 6.22 kullanılarak hesaplanır.

$$Bilgi\ Kazancı(A) = I(p,n) - E(A) \tag{6.22}$$

A değerlendirilmeye alınan niteliktir.

6.3.2. Deney dizaynı

Deneylerimizde, 10-fold kullanılarak deney metodolojisini izleyerek veri setini test ve eğitim seti olarak ayırdık. Eğitim setinde, saldırı profilleri, PIA saldırı modelinin farklı varyantlarına göre yani PIA-ID, PIA-AS, PIA-NR, PIA-SR üretilir ve burada saldırgan hedef ürünün en yüksek veya en düşük değeri almasını isteyebilir. Bu nedenle, tek bir atak profili boyutu ile farklı seçilen boyutlara (selected size) sahip PIA saldırı modellerine göre her saldırı modeli için itme saldırı profilleri oluşturuyoruz.

Deneylerde, saldırı boyutundaki değişiklik saldırının maliyetini/faydasını etkilediği için saldırı boyutu 0,1 olarak ayarlanmıştır; %10' dan büyük bir değer, gerçek dünya uygulamalarında uygun görülmez (Batmaz, Yilmazel, & Kaleli, 2020). Sonuçların rasyonelliğini sağlamak için, saldırı profilinin diğer kısmı olan hedef ürün, her saldırı profili için rastgele seçilir (Yang, Xu, Cai, & Xu, 2016). Sonuç olarak, her saldırı türü için, eğitim seti için sırasıyla YM10 ve YM20 için %3 ila %100 arasında değişen sırasıyla çeşitli seçilmiş boyutlarda 165 ve 39 saldırgan oluşturuyoruz. Ardından, eğitim veri setini, dört atak tipine göre oluşturulmuş kullanıcı profillerini YM10 ve YM20 veri

setlerini ekleyerek, güncelliyoruz. Böylece YM10 ve YM20 eğitim veri setleri 1645, 386 gerçek kullanıcı ve 660 (165x4), 156 (39x4) karma saldırgan profilinden oluşmaktadır. %3, %5, %10, %20, %30, %40, %50, %60, %70 ve %80 seçilen boyutları ve %10 atak boyutu ile dört farklı saldırı tipinden yararlanarak test veri seti için saldırı profilleri oluşturuyoruz. Ancak buradaki kritik nokta, eğitim verisi oluşturmanın aksine, saldırgan profillerinin veri seti içerisine ekleme işleminde her saldırı modeli için ayrı ayrı yapılmasıdır. Ardından, test veri setimizi oluşturacak şekilde oluşturulan saldırı profillerini gerçek kullanıcı profillerine eklenir. Bu nedenle, dört saldırı türü, bir saldırı boyutu (attack size) ve on seçilen boyut dahil olmak üzere 40 (4x1x10) test veri seti vardır.

Test ve eğitim veri seti oluşturulduktan sonra, gerçek ve saldırı kullanıcı profilleri için atak tespit özellikleri belirlenir. Aşağıda özetlenen eğitim sınıflandırıcısı için 15 algılama özniteliği kullanıyoruz.

- Üç genel özellikler: *DegSim (k=25), WDegSim (k=25, d=20), Entropi*
- Üç seçilen boyut tabanlı özellikler: *SSTI, SSPI, SSPII*
- İki belli oylara dayalı özellikler: *SSMAXRI, SSMINRI*
- Yedi önerdiğimiz özellikler: *StdDev, PA, NrPopItem, AvgPopItem, RatioPopItem, DU, NrBB*

Önceki bölümlerde belirttiğimiz gibi, atak tespit özellikleri, çoklu kriterli bilgilere göre *toplama-tabanlı*, *kriter-tabanlı*, *genel-tabanlı* olmak üzere üç farklı şekilde belirlenebilir. Eğitim seti için atak tespit özellikleri oluşturulduktan sonra, sınıflandırmada kullanılan girdiler ‘gerçek’ veya ‘saldırgan’ olarak etiketlenir. Son adım olarak, eğitim özelliklerine dayalı ünlü *k*NN algoritmasını kullanarak eğitim aşamasını, yani sınıflandırmayı gerçekleştiriyoruz. Burada Öklid uzaklığını kullanarak en yakın dokuz komşuyu ele alıyoruz.

Özellikler, önerilen özelliklerin çalışmada performans üzerindeki başarısını göstermek için *Temel* (*Base*) ve *Karma* (*Mix*) olarak ayrılmıştır. *Mix*, çalışmadaki tüm özellikleri içerir. Öte yandan, *Base*, önerilen öznitelikler dışında literatürde var olan özelliklerdir, yani, *DegSim*, *WDegSim*, *Entropi*, *SSTI*, *SSPI*, *SSPII*, *SSMAXRI*, *SSMINRI*. Bu nedenle, *Base*, *Mix* çerçevemizi değerlendirmek için bir kıyaslama yöntemi olarak düşünülebilir.

6.3.3. Deney sonuçları ve analizi

Bu bölüm, deneylerin sonuçlarını ve onlardan elde edilen bilgileri sunar. Bu amaçla, öncelikle bilgi kazancı kullanılarak çıkarılan özniteliklerin etkisini ölçmek için yapılan deneylerin sonuçlarını sunuyoruz. Ardından, şilin saldırısı algılama çerçevemizin kapsamlı bir değerlendirmesini sunuyoruz.

6.3.3.1. Çıkartılan özelliklerin etkisi

Hem YM10 hem de YM20 veri kümeleri için karşılık gelen bilgi kazancı değerlerini ve önerilen öznitelikleri aşağıdaki Tablo 6.1' de gösterilirken, önerilen özniteliklerimizin '*' sembolü ile ifade ediyoruz.

Elde edilen sonuçlara göre, YM20 veri seti için üç farklı senaryonun genel değerlendirmesinde en yüksek bilgi kazancı değerinin *WDegSim'* e ait olduğu, onu *SSMAXRI* ve *DegSim* niteliklerinin izlediği söylenebilir. Bu gözlemin açıklamasını, sisteme uygulanan saldırı modelinde güçlü ürünlerin olması ve bu da komşuluk sürecinde, özellikle PIA-ID, PIA-SR ve PIA-AS modellerinde etkinliğini arttırmasından kaynaklıdır şeklinde yapılabilir. Ayrıca saldırı profillerinde seçilen ürün kümelerinin güçlü ürünlerden oluşması ve bu ürünlere verilen maksimum oy değerlerinin sayısının gerçek kullanıcılara göre daha fazla olması da diğer bir ayırt edici noktadır.

Tablo 6.1. *Kullanıcılardan çıkartılan özelliklerin bilgi kazancı değerleri*

YM20				**YM10**			
Özellikler	*Genel*	*Toplama*	*Kriter*	*Özellikler*	*Genel*	*Toplama*	*Kriter*
WDegSim	0,3060	0,2862	0,3084	SSTI	0,3258	0,3492	0,3249
SSMAXRI	0,2648	0,2558	0,2638	SSPI	0,3209	0,3560	0,3972
DegSim	0,2532	0,3030	0,2690	NrPopItem*	0,3025	0,2911	0,3066
SSPI	0,2312	0,2081	0,2324	NrBB*	0,2884	0,2870	0,2922
SSTI	0,2137	0,1876	0,2133	WDegSim	0,2863	0,2860	0,2889
NrBB*	0,1766	0,1546	0,1733	DegSim	0,2809	0,2580	0,2835
NrPopItem*	0,1700	0,1497	0,1698	SSMAXRI	0,2521	0,2664	0,2628
DU*	0,1482	0,1131	0,1211	DU*	0,1945	0,1630	0,1663
PA*	0,1433	0,0783	0,1307	StdDev*	0,1889	0,1292	0,1757
StdDev*	0,1266	0,1189	0,1415	PA*	0,1328	0,0860	0,1588
Entropi	0,1203	0,1709	0,1378	Entropi	0,1040	0,0729	0,1467
SSMINRI	0,0728	0,0888	0,0847	SSMINRI	0,1006	0,0829	0,0905
RatioPopItem*	0,0448	0,0382	0,0428	RatioPopItem*	0,0379	0,0773	0,0364
SSPII	0,0429	0,0545	0,0458	SSPII	0,0375	0,0581	0,0348
AvgPopItem*	0,0338	0,0295	0,0349	AvgPopItem*	0,0182	0,0659	0,0174

YM10 veri kümesi için sonuçlara baktığımızda, *SSTI* ve *SSPI* en başarılı özellikler iken, önerilen özelliklerden *NrPopItem* ve *NrBB* bu sırayı takip etmektedir. *SSTI* özelliğinin sahte profilleri ayırt etmedeki etkinliğinin bir nedeni, gerçek dünya uygulamalarında bireylerin büyük miktardaki içerik içerisinden birkaç ürünü değerlendirmeye almasıdır. Dolayısıyla bu da bu niteliğin önemli bir şekilde ayırt edici özelliğini arttırmaktadır. Elde edilen bilgi kazancı değerlerine göre, *SSPI* ve *NrPopItem* metrikleri diğer baskın nitelikler gibi görünmektedir. Bu nitelikler ürün popülerliğine dayanırken, sahte profillerde bu ürün popülerliğini temsil eden güçlü ürünlerin yer alması ayırt edici olmalarını sağlamıştır. Son olarak, *NrBB*, ürünlerin hem popülaritesine hem de beğeni derecesine bağlı olduğu için bir sonraki bilgilendirici özellik olarak görünmektedir. Burada seçilen ürün kümeleri, PIA-NR için oy sayısına göre seçilir, bu nedenle yüksek oy değerlerine sahip olmaları beklenir.

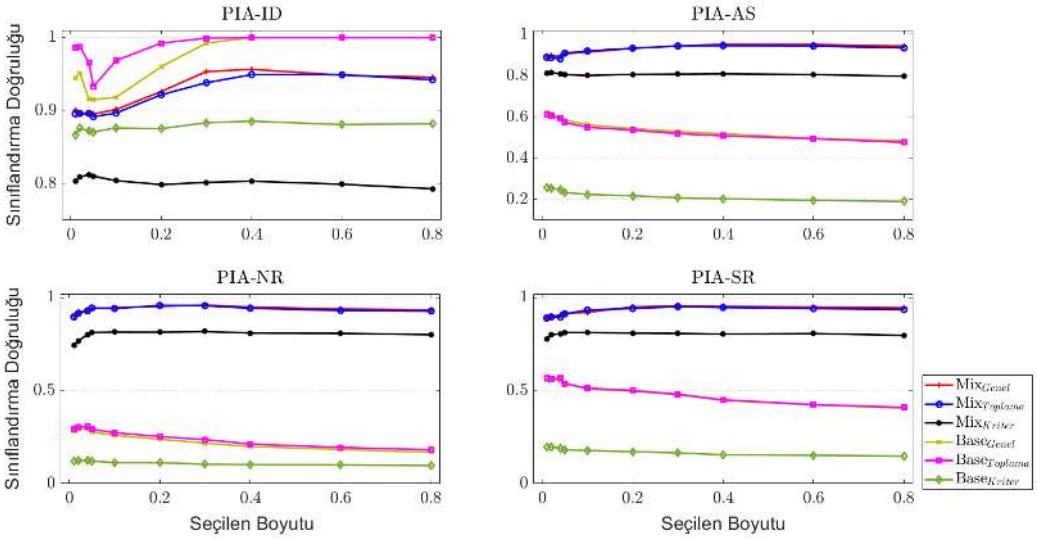
Genel olarak, *SSMINRI*, *RatioPopItem*, *SSPII* ve *AvgPopItem* niteliklerinin her iki veri kümesinde de en düşük bilgi kazancına ve benzer sıralamalara sahip olduğu sonucuna varılabilir. Böyle bir sıralama, geniş seçilen boyut ve saldırı profillerindeki maksimuma yakın oy değerlerine sahip olmasından kaynaklansa da *SSMAXRI*, *NrPopItem* ve *NrBB* niteliklerinin şilin profiller üzerindeki ayırt edici etkisini arttırdığını Tablo 6.1' e bakarak söyleyebiliriz. Ayrıca, oy değerlerinin dağılımına ve ürün popülerliğine dayalı diğer özelliklerin biraz düşük bilgi kazanım değerlerine sahip olduğunu ve daha az etkili olduğunu gözlemliyoruz. Bu gözlem, YM20 veri kümesine kıyasla YM10 için daha belirgindir, bu nedenle, gerçek dünya uygulama senaryolarında daha yaygın olan seyrek veri toplama için önerilen özelliklerimizin daha başarılı olduğu sonucuna varılabilir.

6.3.3.2. Önerilen şilin saldırı tespit yaklaşımın değerlendirilmesi

Hem *Base* senaryosunun hem de bizim önerdiğimiz mekanizmanın *Mix'* in şilin saldırılarını tespit etmedeki performansını incelemek için, iki farklı veri seti olan YM20 ve YM10 üzerinde birkaç deney yapıyoruz. Deneylerde saldırı boyutunu 0,1 olarak belirledik ve dört farklı saldırı türünü (PIA-ID, PIA-AS, PIA-NR ve PIA-SR), ve 0 ile 0,8 arasında değişen seçilen boyutu ile çok ölçütlü bilgi için benimsenen üç farklı senaryoyu (*genel-tabanlı, toplama-tabanlı, kriter-tabanlı*) ele aldık.

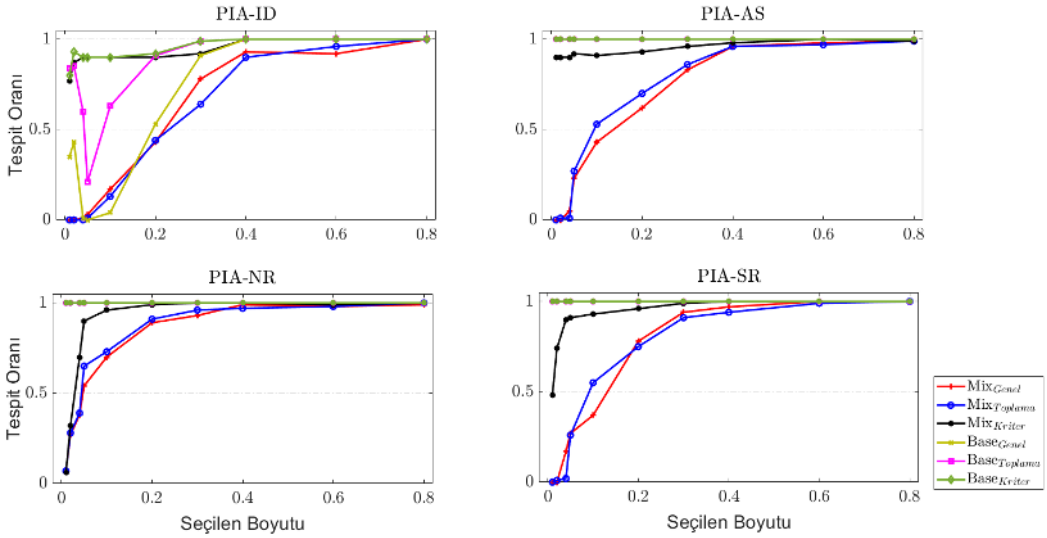
YM20 veri seti için, sınıflandırma doğruluğu açısından, çeşitli PIA atak tipleri için *Mix* ve *Base* özellikli *k*NN sınıflandırma sonuçları Şekil 6.2' de gösterilmiştir. PIA-ID modelde *k*NN sınıflandırma başarısının en yüksek olduğu ve genellikle *Base* yöntemlerin

en iyi sonucu verdiği görülmektedir. Özellikle $Base_{Genel}$ ve $Base_{Toplama}$ seçilen boyutu arttıkça en başarılı sonucun elde edildiği, seçilen boyutu 0,4 olarak belirlendiğinde sınıflandırmanın %100 doğruluğa ulaştığı söylenebilir. *Base* yöntemlerinin her birinde seçilen boyutunun artışı, doğruluk üzerinde pozitif etkisi olmaktadır. Kalan diğer atak tiplerinde bu yöntemin başarısı oldukça düşmektedir ve $Base_{Genel}$ ve $Base_{Toplama}$ özelliklerinde çakışma olurken, $Base_{Kriter}$ benzer ve en kötü sonuçlar alınmaktadır. Özellikle PIA-NR atak modelinde $Base_{Toplama}$ ve $Base_{Genel}$ artan seçilen boyut başarıyı %20' lere düşürmektedir. Diğer taraftan PIA-AS ve PIA-SR atak modellerinde, artan seçilen boyut ile $Base_{Toplama}$ ve $Base_{Genel}$ yöntemlerinde başarı %40, $Base_{Kriter}$ ile bu başarı yaklaşık %20'dir. Sonuç olarak *Base* yöntemin düşük seçilen boyutunda bile PIA-ID atak modelinde başarılı olmasının nedeni, bu atak modelinde ürün popülerliğinin değil komşuluk sayısının önemli olmasıdır. Dolayısıyla bu yöntem de bu ölçütün ön planda olması doğru bir sınıflandırma yapılmasını sağlamıştır. Ürün popülerliğinin de dikkate alındığı *Mix* yöntemin sonuçlarına bakıldığında %80 üzerinde başarı görülmektedir. Ancak PIA-ID harici diğer üç atak tipinde $Mix_{Toplama}$ ve Mix_{Genel} senaryolarında başarı %90 üzeri, Mix_{Kriter} yönteminde %80 başarı vardır. Dahası bu atak modellerinde atak profillerindeki ürün sayılarının artışı yöntemin doğruluğunu da arttırmaktadır. Aksine PIA-ID atak modelinde Mix_{Genel} ve $Mix_{Toplama}$ atak tipleri en yüksek başarıya seçilen boyutun 0,4 seçilmesi ile ulaşmaktadır. Mix_{Kriter} yöntemi için ise maksimum başarı düşük seçilen boyutu ile olmaktadır. Sonuç olarak diğer üç atak tipinin ürün popülerliğine ve baskınlığına dayanması bu yöntemin başarılı olmasını sağlamıştır.



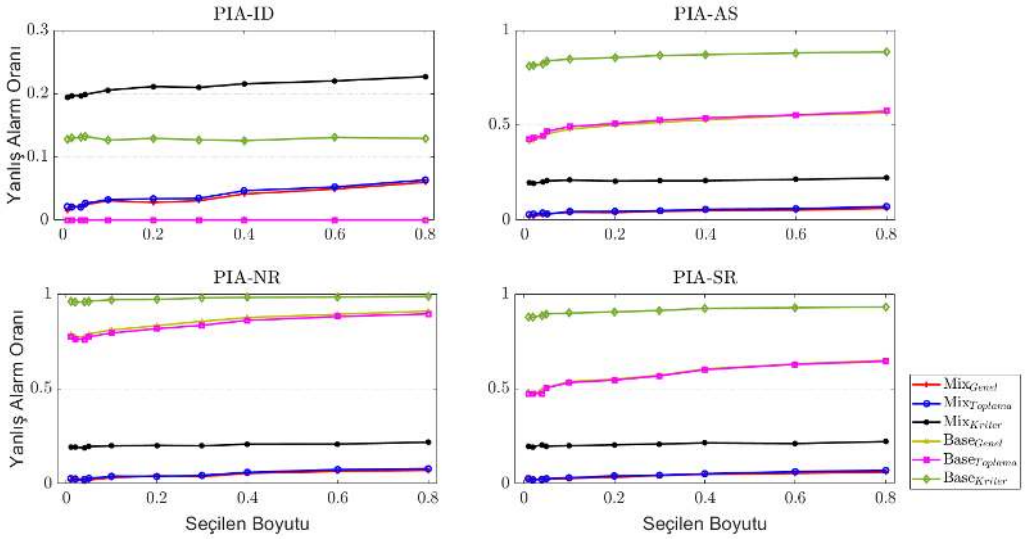
Şekil 6.2. *YM20 veri seti üzerinde bütün senaryolar için sınıflandırma doğruluğu üzerine seçilen boyutun etkisi*

Diğer performans ölçütü olan atak tespit oranı sonuçlarına göre, tüm atak tiplerinde *Base* yöntem oldukça başarılıdır (bknz. Şekil 6.3). Bu yöntemin tamamı, PIA-ID atak tipi hariç diğer tüm atak tipinde her bir seçilen boyutu için atak profilleri gerçek kullanıcılardan ayırt edilmektedir ve %100 oranında başarı sağlanmaktadır. Ancak PIA-ID modelinde $Base_{Toplama}$ ve $Base_{Kriter}$ senaryolarında, seçilen boyut 0,3 belirlendiğinde %100 başarı gösterirken, seçilen boyutu 0,4 olduğunda $Base_{Genel}$ bu başarıya ulaşır. Dolayısıyla artan seçilen boyutu, performansı da arttırmaktadır. Benzer durum tüm atak tipleri için *Mix* yöntemde de geçerlidir. PIA-ID ve PIA-AS atak tiplerinde Mix_{Kriter} düşük seçilen boyutunda ayırt edici iken, aksine $Mix_{Toplama}$ ve Mix_{Genel}' da seçilen boyutun 0,8 belirlenmesinde gerçekleşmektedir. Diğer iki atak modeline baktığımızda, seçilen boyut Mix_{Kriter} senaryosunda 0,3, $Mix_{Toplama}$ ve Mix_{Genel} yöntemlerinde 0,6 olarak belirlendiğinde maksimum başarı elde edilir. Sonuç olarak *Base* yöntemin üç atak tipinde sınıflandırma doğruluğunun düşük, atak tespit oranının yüksek olması sistemdeki kullanıcıların çoğunluğunun sahte profil olarak tespit edildiği çıkarımı yapılabilir. Dahası her iki yöntemde *kriter-tabanlı* senaryonun daha başarılı olması da aynı şekilde açıklanabilir.



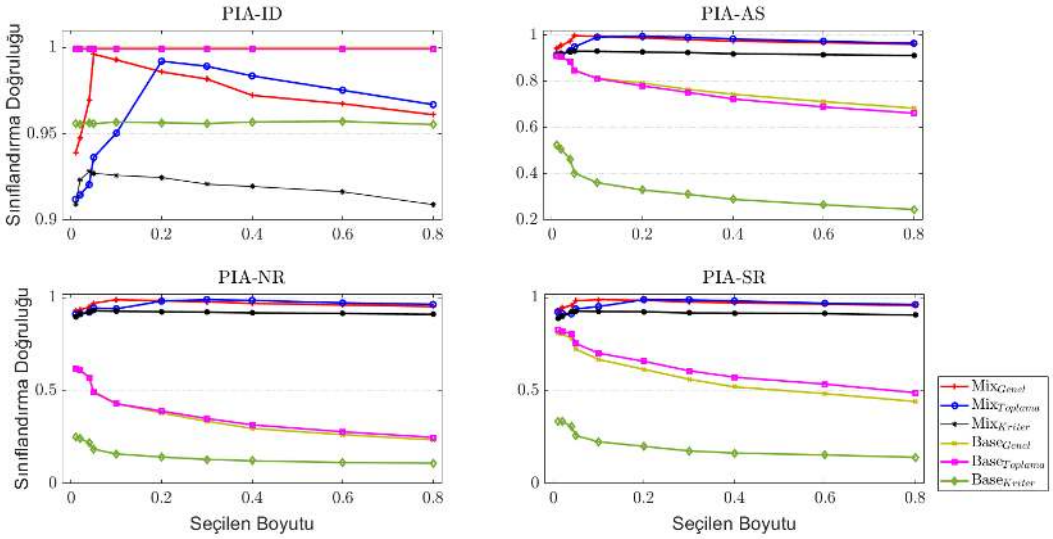
Şekil 6.3. *YM20 veri seti üzerinde bütün senaryolar için tespit oranı üzerine seçilen boyutun etkisi*

Atak tespit oranının yüksek çıktığı fakat sınıflandırma doğruluğunun düşük olduğu durumları yanlış alarm oranı sonuçlarına bakarak yorumlayabiliriz. Tüm atak tiplerinde $Base_{Toplama}$ ve $Base_{Genel}$ yöntemlerin bir çakışmanın olduğu ve gerçek kullanıcıların doğru tespitinin PIA-ID atak tipinde oldukça başarılı olduğunu Şekil 6.4' te verilmiştir. Ancak $Base_{Genel}$ ve $Base_{Toplama}$ yöntemleri diğer atak tiplerinde aynı başarıyı elde edemez. Özellikle PIA-NR atak tipinde seçilen boyutunda artış bu hatayı %100' e götürmektedir. Bu *Base* yöntemin *kriter-tabanlı* senaryosunda hata oranı PIA-ID atak modelinde %13' lere ulaşırken, diğer atak tiplerinde bu oran artan seçilen boyutu ile %100' e ulaşmaktadır. Önerilen özelliklerin dâhil edildiği *Mix* yöntemin *genel-* ve *toplama-tabanlı* senaryolarında tüm atak tipleri için bir çakışma meydana gelmektedir. Bu yöntemlerin en yüksek hata oranına sahip olduğu atak modeli PIA-ID iken kalan diğer atak tiplerinde bu hata oranı %10 düşmektedir. Aksine Mix_{Kriter} senaryoda PIA-ID atak tipinde hata oranı %23' tür. Diğer atak tiplerinde ise benzer sonuçlara sahiptir. Her iki yöntemde de seçilen boyutunun artması bu performansı ters etkilemektedir.



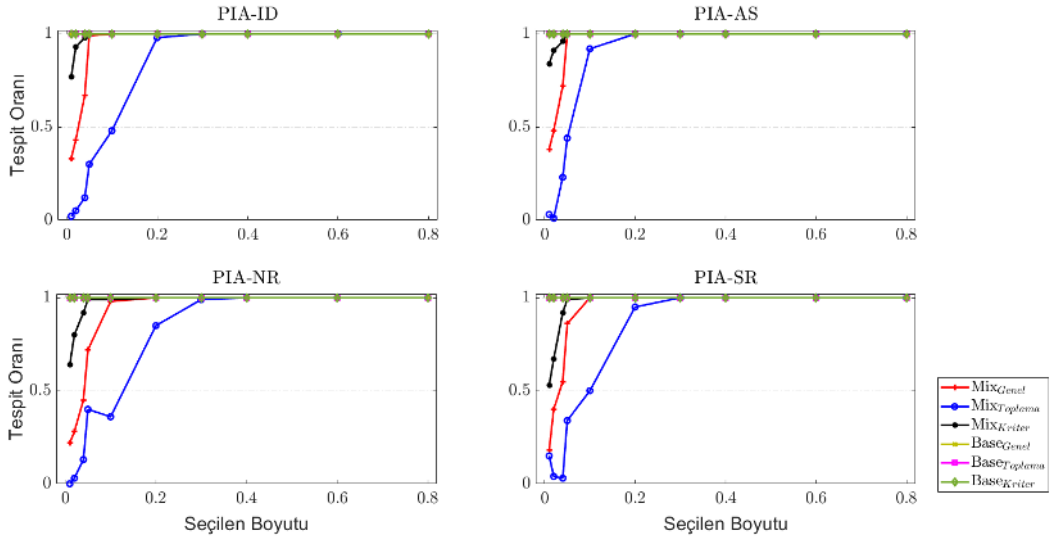
Şekil 6.4. *YM20 veri seti üzerinde bütün senaryolar için yanlış alarm oranı üzerine seçilen boyutun etkisi*

Bir diğer veri seti üzerinde yapılan deneylerin sınıflandırma doğruluğu Şekil 6.5' de verilen sonuçlara göre tüm yöntemlerde PIA-ID atak tipinde %90 üzeri başarı sağlanırken, kalan atak tipleri için aynı başarı söz konusu değildir. PIA-ID atak tipi için her bir seçilen boyutunda %100 başarı $Base_{Toplama}$ ve $Base_{Genel}$ yöntemlerinde elde edilir. Aksine $Base_{Kriter}$ yönteminde bu başarı %96 olmaktadır. Diğer atak modellerinden PIA-AS ve PIA-SR de $Base_{Toplama}$ ve $Base_{Genel}$ benzer sonuçlar vardır ve artan seçilen boyutu doğruluğu sırasıyla %70 ve %50' lere düşürmektedir. Kalan atak tipinde ise bu yöntemlerin başarısı oldukça düşmektedir. *Kriter-tabanlı* senaryoda diğer senaryolara göre daha çok sınıflandırma hatası vardır. Bütün özelliklerin yer aldığı *Mix* yöntem en düşük başarıyı PIA-ID ile elde eder. Bu atak tipi için seçilen boyutundaki artış Mix_{Genel} ve $Mix_{Toplama}$ senaryolarını belli bir noktaya kadar maksimuma ulaştırır ve daha sonra düşmeye başlar. Aynı durum *kriter-tabanlı* senaryo için de geçerlidir. Ancak geri kalan tüm atak tipleri için benzer sonuçların olduğu ve Base yönteme göre daha başarılı olduğu söylenebilir. Sonuç olarak PIA-ID hariç diğer atak tiplerinde *Mix* yöntemin başarılı olduğu söylenebilir. Dahası *kriter-tabanlı* senaryoların daha düşük performans gösterdiği ve genellikle artan seçilen boyutunun performansı düşürdüğü görülmektedir.



Şekil 6.5. *YM10 veri seti üzerinde bütün senaryolar için sınıflandırma doğruluğu üzerine seçilen boyutun etkisi*

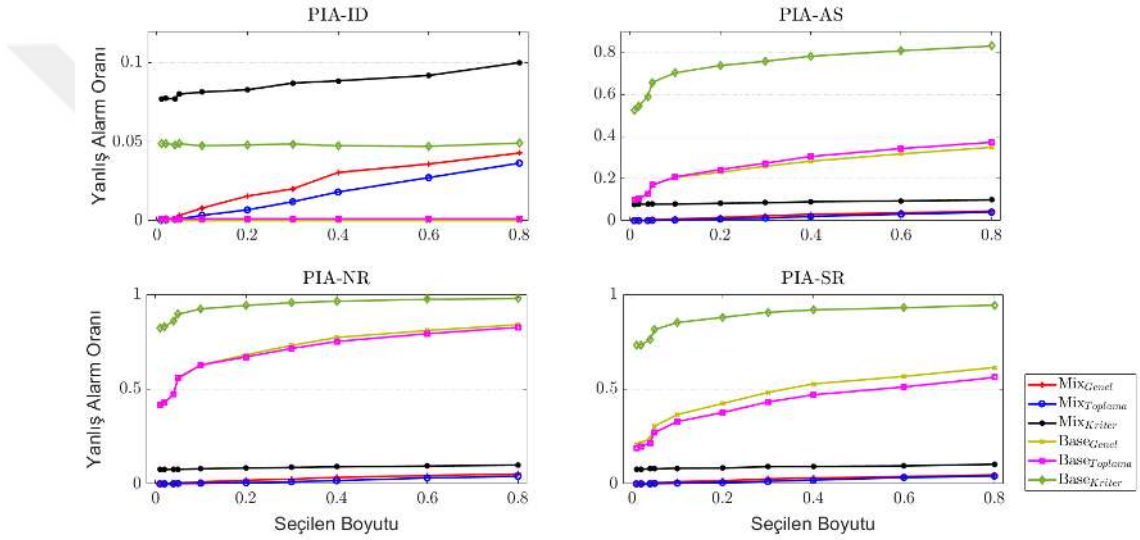
Bir diğer metrik olan sahte profillerin tespit sonuçlarına göre tüm atak tipleri için her bir seçilen boyutunda %100 başarı tüm *Base* yöntemde sağlanmıştır (bknz. Şekil 6.6). Aynı zamanda tüm atak tipleri için Mix_{Kriter} ve Mix_{Genel} düşük seçilen boyutta %100 performans göstermektedir. Ancak $Mix_{Toplama}$ senaryosu PIA-ID ve PIA-AS atak tiplerinde seçilen boyut 0,2 olarak belirlendiğinde başarılıdır. Kalan atak tiplerinde ise bu senaryo artan seçilen boyut ile doğru sahte profil tespiti yapabilmektedir.



Şekil 6.6. *YM10 veri seti üzerinde bütün senaryolar için tespit oranı üzerine seçilen boyutun etkisi*

Bu veri seti için son performans ölçütü olan yanlış alarm oranında en yüksek başarı *Base* yöntem ile PIA-ID atak modeli ile elde edildiği Şekil 6.7' de görülmektedir.

Özellikle $Base_{Toplama}$ ve $Base_{Genel}$ hata oranı sıfırdır ve artan seçilen boyutu bu oranı değiştirmemektedir. Ancak diğer atak tiplerinde $Base_{Genel}$ ve $Base_{Toplama}$ birebir sonuçların olduğu ve artan seçilen boyutu hata oranını arttırmaktadır. Özellikle PIA-NR atak tipinde bu hata %100 e yaklaşmaktadır. Dahası *kriter-tabanlı* senaryonun oldukça başarısız olduğu ve seçilen boyutu ile ters orantı olduğu söylenebilir. Son olarak *Mix* yöntemin sonuçlarına bakıldığında, PIA-ID atak tipinde Mix_{Genel} ve $Mix_{Toplama}$ diğer senaryoya göre oldukça başarılıdır. Diğer kalan atak tiplerinde en başarılı yöntem olan *Mix* yöntemi benzer sonuçlar verdiği ve $Mix_{Toplama}$ ve Mix_{Genel} sonuçlarında çakışma olduğu görülmektedir.



Şekil 6.7 *YM10 veri seti üzerinde bütün senaryolar için yanlış alarm oranı üzerine seçilen boyutun etkisi*

Çok ölçütlü sistemi üç farklı senaryoda (*genel-tabanlı, kriter-tabanlı, toplama-tabanlı*) değerlendirdiğimiz bu çalışmada, *kriter-tabanlı* senaryonun farklı atak tipleri için düşük seçilen boyutta sahte profillerin tespitinde başarılı olduğu görülmektedir. Bu ölçüt açısından bu senaryo diğerlerine göre daha başarılıdır. Aksine diğer performans ölçütlerinde başarı sıralamasında sonda yer almaktadır. Diğer iki senaryonun ise birbirleriyle benzer sonuçlar verdiği hatta bazı durumda çakışmanın olduğu çıkartılabilir.

6.4. Değerlendirmeler

Öneri sistemleri, kullanıcılardan elde edilen verileri analiz eder ve ürün seçim sürecinde çok fazla zaman harcamalarını engelleyerek doğru seçimi yapmalarına rehberlik eder. Şilin saldırıları, öneri sistemlerinin güvenilirliği ve sağlamlığı için ciddi bir tehdit oluşturuyor. Şilin saldırı tespiti olarak adlandırılan gürbüz öneri sistemleri,

saldırı profillerini tespit eden ve etkilerini en aza indirmeyi amaçlayan mekanizmalardır. Literatürde yapılan çalışmalarda çoğu araştırmacı tek kriterli veri üzerinde bir saldırı profili tespit etmiştir. Ancak çok ölçütlü bir yapıyı benimseyen şirketler veya sistemler, müşteri memnuniyeti ve adaletinin çok önemli olması, saldırı profili tespitinin farklı yapılması gerektiğini göstermiştir. Ayrıca, literatürde bulunan özellikler bazı durumlarda saldırı profilleri ile gerçek kullanıcılar arasında ayrım yapamamaktadır. Bu nedenle, bu çalışma, çok ölçütlü bilgilere dayanan daha sağlam sistemler elde etmek için sınıflandırma tabanlı yeni bir şilin saldırısı tespit yaklaşımı önermektedir. Özellikle, ürün popülerliğini ve kullanıcı özelliklerini dikkate alan *Mix* yaklaşım, mevcut özellikleri içeren *Base* yaklaşımıyla karşılaştırılır.

Bu çalışma, dört PIA saldırı modeli tipinin (PIA-ID, PIA-AS, PIA-NR, PIA-SR) istatistiksel özelliklerini göz önünde bulundurarak, önerilen yaklaşımın ve özelliklerin iki farklı çok ölçütlü veri seti üzerindeki etkinliğini üç farklı senaryoda göstermektedir. İyi bilinen bir sınıflandırıcı olan *k*NN ile yapılan deneyler, önerilen yaklaşımın, *Mix* yöntemin şilin profillerini tespit etmede genellikle kıyaslama yöntemi olan *Base* yönteme göre üstesinden gelmektedir. PIA-ID saldırı türünün özel yapısına göre, küçük seçilen boyutta bile *Base* yönteminin oldukça başarılı olduğunu gözlemliyoruz. Diğer saldırı türlerinde popülerlik ve derecelendirme değerleri dağılımının önemi, *Mix* yöntemini başarılı kılmaktadır. Ayrıca artan seçilen boyut, sınıflandırma performansında ve saldırı profili tespitinde %100 başarı sağlar. Başarısı, belirtilen saldırı modellerindeki saldırı profillerinin güçlü ve popüler ürünler olmasına dayanmaktadır. Başarı üç farklı senaryoda değerlendirildiğinde diğerlerine göre kriter bazlı senaryonun gerisinde kalmaktadır. Ancak durum böyle olsa bile, saldırganın kriterler hakkında tüm olası bilgilere sahip olmasına ve başarılı bir saldırı yapmasına rağmen önerilen yöntem ve özelliklerin oldukça başarılı olduğunu görebiliriz.

Ayrıca, *kriter-tabanlı* senaryoda, saldırganın olası tüm bilgileri elde etmesi daha başarılı bir saldırıya yol açabilir. Bu durumda saldırının tespitinin zor olabileceğini gösterebilir. Ancak bu senaryonun da saldırı profillerini gerçek kullanıcılardan ayırt etmede oldukça başarılı olduğunu gördük.

7. DEĞERLENDİRMELER VE GERÇEKLEŞTİRİLMESİ PLANLANAN DİĞER ÇALIŞMALAR

Tek ölçütlü öneri sistemleri üzerinde literatürde oldukça fazla çalışmalar olsa da günümüzde çok ölçütlü sistemlerin kullanımı gittikçe artmıştır. Bu sistemleri kullanan firmalar, kullanıcıların aldığı hizmetlerle ilgili çok yönlü değerlendirmelerine imkân tanımaktadır. Eğer kullanıcılarla ilgili bu çok yönlü değerlendirmeler iyi bir şekilde analiz edilebilirse, daha başarılı kişiselleştirilmiş tavsiyeler sunulabilir. Dolayısıyla literatürde çok ölçütlü en iyi-n öneri sistemleri alanında çalışmaların yer almadığı ve böyle bir durumun varlığı bu sistemleri kullanan firmalar açısından ileriki zamanlarda büyük problemlere yol açabilir.

Bu tez kapsamında, ilk bölümde, yeni birçok ölçütlü en iyi-n öneri sistemleri tasarlanmıştır. Önerilen yöntemlerde kullanıcıların karakteristik yapıları eğilimleri dikkate alınırken, ürün dağılımlarına ve ürünler arası ilişkisel yapıya dayanan yeni bir komşuluk seçme yöntemi belirlenmiştir. Çok ölçütlü sistemlere uyarlanan bir kıyaslama algoritması ile önerilen yöntemlerin performans karşılaştırması yapılmıştır. Üç farklı veri seti üzerinde gerçekleştirilen çalışmalarda, kullanıcıya ürün listesi sunmada önerilen yöntemlerin oldukça başarılı, kullanıcı memnun edici sonuçlar verdiği gözlenmiştir. Dahası önerilen bu listede, çeşit ve yeni ürünlerin de var olduğu ve bu durumun memnuniyet etkisi yarattığı söylenebilir.

Önerilen sistemlerin oldukça başarılı tavsiyeler üretmesi, kötü niyetli kişilerin bu sistemleri hedef noktası haline getirebileceği ihtimalini göstermiştir. Dolayısıyla ikinci aşamada, çok ölçütlü en iyi-n öneri sistemlerinin şilin ataklara karşı ne derece gürbüz olduğunun analizi yapılmış ve liste içerisindeki ürünlerin manipülasyonu amaçlanmıştır. Önerilen algoritmaların ürün tabanlı olması ve literatürde var olan atak modellerin, bu ürün tabanlı sistemlerde başarısız olması PIA atak modellerinin (PIA-ID, PIA-AS, PIA-NR, PIA-SR) kullanılmasını sağlamıştır. Bu atak modeli için gerekli olan güçlü ürünler seçiminde literatürde var olan yöntemlere ek olarak yeni bir seçim yöntemi belirlenmiş ve çoklu hedef ürün kullanılarak atak gerçekleştirilmiştir. Performans analizinde ise manipüle edilen ürünlerin sayısına ek olarak bu yanlış ürünlerin nasıl bir sırada kullanıcıya sunulduğu da incelenmiştir. İki veri seti üzerinde gerçekleştirilen deneylerde, önerilen yöntemlerin her ne kadar başarılı tavsiyeler sunsa da ataklara karşı savunmasız oldukları görülmüştür.

Son bölümde, PIA ataklara (PIA-ID, PIA-AS, PIA-NR, PIA-SR) karşı savunmasız olan bu çok ölçütlü sistemlerin, atak tespit işlemi yapılmıştır. *k*NN sınıflandırma tabanlı yeni bir şilin atak tespit yöntemi geliştirilmiştir. Üç farklı senaryo üzerinden gerçekleştirilen çalışmada, özellik çıkarımı yöntemlerinde ürün popülerliğine ve kullanıcı karakteristik yapısına dayanan yeni metrikler önerilmiştir. İki veri seti ve 15 özellik üzerine gerçekleştirilen deneyin sonuçlarında, sahte profillerin gerçek kullanıcılardan ayırt edilmesinin oldukça başarılı olduğu görülmüştür.

Bu tez kapsamında yapılan çalışmaların ürün tabanlı olması, ileride aynı adımların kullanıcı tabanlı veya kullanıcı ürün tabanlı şeklinde bir geliştirmeye açık olduğu görülmüştür. Kullanıcı tabanlı geliştirilen bu sistemlerin gürbüzlük analizinde güçlü kullanıcı atak modeli kullanılabilir. Dahası atakların tespitinde farklı bir sınıflandırma algoritması ve yeni özellikler kullanılabilir.

KAYNAKÇA

Abdollahpouri, H., Mansoury, M., Burke, R., & Mobasher, B. (2019). The unfairness of popularity bias in recommendation. *arXiv preprint arXiv:1907.13286*.

Adomavicius, G., & Kwon, Y. (2007). New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems, 22*, 48–55.

Adomavicius, G., & 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*, 734-749.

Adomavicius, G., Manouselis, N., & Kwon, Y. (2011). Multi-criteria recommender systems. *Recommender systems handbook* (s. 769-803). içinde

Aggarwal, C. C. (2016). Neighborhood-based collaborative filtering. *Recommender systems* (s. 29-70). içinde Springer.

Atanassov, K. T. (1999). Intuitionistic fuzzy sets. *In Intuitionistic fuzzy sets*, 1-137.

Bansal, S., & Baliyan, N. (2021). ShillDetector: A binary grey wolf optimization technique for detection of shilling profiles. *Journal of Ambient Intelligence and Humanized Computing*, 1-14.

Batmaz, Z., Yilmazel, B., & Kaleli, C. (2020). Shilling attack detection in binary data: a classification approach. *Journal of Ambient Intelligence and Humanized Computing, 11*, 2601-2611.

Bhaumik, R., Mobasher, B., & Burke, R. (2011). A clustering approach to unsupervised attack detection in collaborative recommender systems. *Proceedings of the International Conference on Data Science (ICDATA)* (s. 1). içinde Citeseer.

Bhaumik, R., Williams, C., Mobasher, B., & Burke, R. (2006). Securing collaborative filtering against malicious attacks through anomaly detection. *Proceedings of the 4th workshop on intelligent techniques for web personalization (ITWP'06), Boston*, *6*, s. 10.

Bilge, A., & Polat, H. (2013). A scalable privacy-preserving recommendation scheme via bisecting k-means clustering. *Information Processing & Management, 49*, 912-927.

Bilge, A., Gunes, I., & Polat, H. (2014). Robustness analysis of privacy-preserving model-based recommendation schemes. *Expert Systems with Applications*, 3671-3681.

Bokde, D., Girase, S., & Mukhopadhyay, D. (2015). Matrix factorization model in collaborative filtering algorithms: A survey. *Procedia Computer Science, 49*, 136-146.

Bradley, K., & Smyth, B. (2001). Improving recommendation diversity. *Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland*, (s. 85-94).

Breese, J. S., Heckerman, D., & Kadie, C. (2013). Empirical analysis of predictive algorithms for collaborative filtering. *arXiv preprint arXiv:1301.7363*.

Bryan, K., O'Mahony, M., & Cunningham, P. (2008). Unsupervised retrieval of attack profiles in collaborative recommender systems. *Proceedings of the 2008 ACM conference on Recommender systems* (s. 155-162). içinde

Burke, R., Mobasher, B., Williams, C., & Bhaumik, R. (2006a). Classification features for attack detection in collaborative recommender systems. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, (s. 542-547).

Burke, R., Mobasher, B., Williams, C., & Bhaumik, R. (2006b). Detecting profile injection attacks in collaborative recommender systems. *The 8th IEEE International Conference on E-Commerce Technology and The 3rd IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services (CEC/EEE'06)*, (s. 23-23).

Burke, R., O'Mahony, M. P., & Hurley, N. J. (2015). Robust collaborative recommendation. *Recommender systems handbook* (s. 961-995). içinde Springer.

Cai, H., & Zhang, F. (2019). An unsupervised method for detecting shilling attacks in recommender systems by mining item relationship and identifying target items. *The Computer Journal, 62*, 579-597.

Cao, J., Wu, Z., Mao, B., & Zhang, Y. (2013). Shilling attack detection utilizing semi-supervised learning method for collaborative recommender system. *World Wide Web*, 729-748.

Chen, R., Hua, Q., Chang, Y., Wang, B., Zhang, L., & 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. doi:https://doi.org/10.1109/ACCESS.2018.2877208

Chirita, P., Nejdl, W., & Zamfir, C. (2005). Preventing shilling attacks in online recommender systems. *Proceedings of the 7th annual ACM international workshop on Web information and data management*, (s. 67-74).

Chung, C. Y., Hsu, P. Y., & Huang, S. H. (2013). βP: A novel approach to filter out malicious rating profiles from recommender systems. *Decision Support Systems, 55*, 314-325.

Cremonesi, P., Koren, Y., & Turrin, R. (2010). Performance of recommender algorithms on top-n recommendation tasks. *Proceedings of the fourth ACM conference on Recommender systems*, (s. 39-46).

Deldjoo, Y., Di Noia, T., Di Sciascio, E., & Merra, F. A. (2021). A Regression Framework to Interpret the Robustness of Recommender Systems Against Shilling Attacks.

Gao, M., Yuan, Q., Ling, B., & Xiong, Q. (2014). Detection of abnormal item based on time intervals for recommender systems. *The Scientific World Journal, 2014*.

Gunes, I., Bilge, A., Kaleli, C., & Polat, H. (2013). Shilling attacks against privacy-preserving collaborative filtering. *Journal of Advanced Management Science*, 54-60.

Gunes, I., Kaleli, C., Bilge, A., & Polat, H. (2014). Shilling attacks against recommender systems: a comprehensive survey. *Artificial Intelligence Review, 42*, 767-799.

Isinkaye, F. O., Folajimi, Y., & Ojokoh, B. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 261-273.

Jannach, D., Lerche, L., Kamehkhosh, I., & Jugovac, M. (2015). What recommenders recommend: an analysis of recommendation biases and possible countermeasures. *User Modeling and User-Adapted Interaction, 25*, 427-491. doi:https://doi.org/10.1007/s11257-015-9165-3

Kaleli, C. (2014). An entropy-based neighbor selection approach for collaborative filtering. *Knowledge-Based Systems, 56*, 273-280.

Kaminskas, M., & Bridge, D. (2016). Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems (TiiS), 7*, 1-42.

Kang, Z., Peng, C., Yang, M., & Cheng, Q. (2016). Top-n recommendation on graphs. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, (s. 2101-2106).

Kaur, P., & Goel, S. (2016). Shilling attack models in recommender system. *2016 International conference on inventive computation technologies (ICICT)*, *2*, s. 1-5.

Kaya, T., & Kaleli, C. (2022). A novel top-n recommendation method for multi-criteria collaborative filtering. *Expert Systems with Applications*, 116695.

Ke, D., Song, Y., & Quan, W. (2018). New distance measure for Atanassov's intuitionistic fuzzy sets and its application in decision making. *Symmetry, 10*, 429-449.

Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, (s. 426-434).

Kotkov, D., Wang, S., & Veijalainen, J. (2016). A survey of serendipity in recommender systems. *Knowledge-Based Systems, 111*, 180-192.

Kumari, T., & Bedi, P. (2017). A comprehensive study of shilling attacks in recommender systems. *International Journal of Computer Science Issues (IJCSI), 14*, 44.

Lakiotaki, K., Matsatsinis, N. F., & Tsoukias, A. (2011). Multicriteria user modeling in recommender systems. *IEEE Intelligent Systems, 26*, 64-76.

Lam, S. K., & Riedl, J. (2004). Shilling recommender systems for fun and profit. *Proceedings of the 13th international conference on World Wide Web*, (s. 393-402).

Lee, J., Lee, D., Lee, Y., Hwang, W., & Kim, S. (2016). Improving the accuracy of top-n recommendation using a preference model. *Information Sciences, 348*, 290-304.

Li, H., Gao, M., Zhou, F., Wang, Y., Fan, Q., & Yang, L. (2021). Fusing hypergraph spectral features for shilling attack detection. *Journal of Information Security and Applications, 63*, 103051.

Liu, H., Hu, Z., Mian, A., Tian, H., & Zhu, X. (2014). A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-based systems, 56*, 156-166.

Manouselis, N., & Costopoulou, C. (2007). Analysis and classification of multi-criteria recommender systems. *World Wide Web, 10*, 415-441.

McRoberts, R. E., Tomppo, E. O., Finley, A. O., & Heikkinen, J. (2007). Estimating areal means and variances of forest attributes using the k-Nearest Neighbors technique and satellite imagery. *Remote Sensing of Environment, 111*, 466-480.

Mehta, B. (2007). Unsupervised shilling detection for collaborative filtering. *AAAI*, 1402-1407.

Mehta, B., & Nejdl, W. (2008). Attack resistant collaborative filtering. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (s. 75-82). içinde

Mobasher, B., Burke, R., Bhaumik, R., & Sandvig, J. J. (2007). Attacks and remedies in collaborative recommendation. *IEEE Intelligent Systems, 22*, 56-63.

Mobasher, B., Burke, R., Bhaumik, R., & Williams, C. (2005). Effective attack models for shilling item-based collaborative filtering systems. *Proceedings of the WebKDD Workshop*, (s. 13-23).

Mobasher, B., Burke, R., Bhaumik, R., & Williams, C. (2007). Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology (TOIT), 7*, 23–es.

Morid, M. A., Shajari, M., & Hashemi, A. R. (2014). Defending recommender systems by influence analysis. *Information Retrieval, 17*, 137-152.

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine learning*, 103-134.

O'Mahony, M. P., Hurley, N. J., & Silvestre, G. (2003). Collaborative filtering--safe and sound? *International Symposium on Methodologies for Intelligent Systems* (s. 506-510). içinde Springer.

O'Mahony, M. P., Hurley, N. J., & Silvestre, G. C. (2005). Recommender systems: Attack types and strategies. *AAAI*, (s. 334-339).

O'Mahony, M., Hurley, N. J., & Silvestre, G. C. (2006). Detecting noise in recommender system databases. *Proceedings of the 11th international conference on Intelligent user interfaces*, (s. 109-115).

Pichl, M., Zangerle, E., & Specht, G. (2014). Combining Spotify and Twitter Data for Generating a Recent and Public Dataset for Music Recommendation. *Proceedings of the 26th GI-Workshop Grundlagen von Datenbanken, Bozen-Bolzano*, (s. 35-40).

Ren, Y., Li, G., & Zhou, W. (2012). Learning user preference patterns for Top-N recommendations. *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, (s. 137-144).

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, (s. 175-186).

Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender systems handbook.* Springer.

Samaiya, N., Raghuwanshi, S. K., & Pateriya, R. K. (2019). Shilling attack detection in recommender system using PCA and SVM. *Emerging technologies in data mining and information security* (s. 629-637). içinde Springer.

Sanders, R. (1987). The Pareto principle: its use and abuse. *Journal of Services Marketing*.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th international conference on World Wide Web*, 285-295.

Seminario, C. E. (2013). Accuracy and robustness impacts of power user attacks on collaborative recommender systems. *Proceedings of the 7th ACM conference on Recommender systems*, (s. 447-450).

Seminario, C. E., & Wilson, D. C. (2014). Attacking item-based recommender systems with power items. *Proceedings of the 8th ACM Conference on Recommender systems*, (s. 57-64).

Seminario, C. E., & Wilson, D. C. (2016). Nuking item-based collaborative recommenders with power items and multiple targets. *The Twenty-Ninth International Flairs Conference.*

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal, 27*, 379-423.

Sharma, R., Gopalani, D., & Meena, Y. (2017). Collaborative filtering-based recommender system: Approaches and research challenges. *2017 3rd international conference on computational intelligence & communication technology (CICT)*, (s. 1-6).

Si, M., & Li, Q. (2020). Shilling attacks against collaborative recommender systems: a review. *Artificial Intelligence Review, 53*, 291-319.

Su, X., Zeng, H., & Chen, Z. (2005). Finding group shilling in recommendation system. *Special interest tracks and posters of the 14th international conference on World Wide Web* (s. 960-961). içinde

Tang, T., & McCalla, G. (2009). The pedagogical value of papers: a collaborative-filtering based paper. *Journal of Digital Information*.

Tang, T., & Tang, Y. (2011). An effective recommender attack detection method based on time SFM factors. *2011 IEEE 3rd International Conference on Communication Software and Networks* (s. 78-81). içinde IEEE.

Turk, A., & Bilge, A. (2019). Robustness analysis of multi-criteria collaborative filtering algorithms against shilling attacks. *Expert Systems with Applications, 115*, 386-402.

Turkoglu, T. (2016). *Çoklu ölçüt oy değerleri üzerinden veri madenciliği.* Master's thesis, Anadolu Üniversitesi.

Williams, C. A. (2012). Thesis: Profile injection attack detection for securing collaborative recommender systems.

Williams, C. A., Mobasher, B., & Burke, R. (2007). Defending recommender systems: detection of profile injection attacks. *Service Oriented Computing and Applications, 1*, 157-170.

Wu, Z., Cao, J., Mao, B., & Wang, Y. (2011). Semi-SAD: applying semi-supervised learning to shilling attack detection. *Proceedings of the fifth ACM conference on Recommender systems* (s. 289-292). içinde

Wu, Z., Wang, Y., Wang, Y., Wu, J., Cao, J., & Zhang, L. (2015). Spammers detection from product reviews: a hybrid model. *2015 IEEE International Conference on Data Mining* (s. 1039-1044). içinde IEEE.

Wu, Z., Wu, J., Cao, J., & Tao, D. (2012). HySAD: A semi-supervised hybrid shilling attack detector for trustworthy product recommendation. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (s. 985-993). içinde

Xia, H., Fang, B., Gao, M., Ma, H., Tang, Y., & Wen, J. (2015). A novel item anomaly detection approach against shilling attacks in collaborative recommendation systems using the dynamic time interval segmentation technique. *Information Sciences, 306*, 150-165.

Xue, F., He, X., Wang, X., Xu, J., Liu, K., & Hong, R. (2019). Deep item-based collaborative filtering for top-n recommendation. *ACM Transactions on Information Systems (TOIS), 37*, 1-25.

Yalcin, E., & Bilge, A. (2020). Binary multicriteria collaborative filtering. *Turkish Journal of Electrical Engineering & Computer Sciences, 28*, 3419-3437.

Yalcin, E., & Bilge, A. (2022). Treating adverse effects of blockbuster bias on beyond-accuracy quality of personalized recommendations. *Engineering Science and Technology, an International Journal, 33*, 101083.

Yang, Z., Xu, L., Cai, Z., & Xu, Z. (2016). Re-scale AdaBoost for attack detection in collaborative filtering recommender systems. *Knowledge-Based Systems, 100*, 74-88.

Yu, H., Li, P., & Zhang, F. (2014). An algorithm for detecting recommendation attack based on incremental learning. *Journal of Information & Computational Science*, 2365-2373.

Zhang, F., & Zhou, Q. (2014). HHT--SVM: An online method for detecting profile injection attacks in collaborative recommender systems. *Knowledge-Based Systems*, 96-105.

Zhang, L. (2013). The Definition of Novelty in Recommendation System. *Journal of Engineering Science & Technology Review, 6*, 141-145.

Zhang, S., Chakrabarti, A., Ford, J., & Makedon, F. (2006). Attack detection in time series for recommender systems. *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (s. 809-814). içinde

Zhou, W., Koh, Y. S., Wen, J., Alam, S., & Dobbie, G. (2014). Detection of abnormal profiles on group attacks in recommender systems. *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (s. 955-958). içinde

CLİCK OR TAP HERE TO ENTER TEXT.

ÖZGEÇMİŞ

ORCID NO:

Adı Soyadı : Tuğba KAYA

Yabancı Dil : İngilizce

Doğum Yeri ve Yılı :

E-posta :

Eğitim Geçmişi:

- Doktora (2018-) :Eskişehir Teknik Ünv., Fen Bil. Ens., Bilgisayar Mühendisliği A.B.D.
- Doktora (2016-2018) :Anadolu Üniversitesi, Fen Bil. Ens., Bilgisayar Mühendisliği A.B.D.
- Yüksek Lisans (2014-2016) :Anadolu Üniversitesi, Fen Bil. Ens., Bilgisayar Mühendisliği A.B.D.
- Lisans (2009-2013) :Süleyman Demirel Üniversitesi, Bilgisayar Mühendisliği Bölümü

Meslek Geçmişi:

- Araştırma Görevlisi (2018-) :Eskişehir Teknik Üniversitesi, Bilgisayar Mühendisliği
- Araştırma Görevlisi (2014-2018) :Anadolu Üniversitesi, Bilgisayar Mühendisliği
- Araştırma Görevlisi (2013-2014) :Ardahan Üniversitesi, Bilgisayar Mühendisliği

Tez Kapsamındaki Yayınlar ve Bilimsel Faaliyetler:

Kaya, T., & Kaleli, C. (2022). A novel top-n recommendation method for multi-criteria collaborative filtering. *Expert Systems with Applications*, 116695.

Kaya, T., & Kaleli, C. (2022). Robustness analysis of multi-criteria top-n collaborative filtering, A*rabian Journal for Science and Engineering*. (submitted)

Kaya, T., Yalcin, E. & Kaleli, C. (2022). A novel classification based shilling attack detection for multi-criteria recommender systems, *Computational Intelligence*. (submitted)

Diğer yayınlanan makaleler:

Aydemir, S. B., **Kaya, T**. (2021). TOPSIS method for multi-attribute group decision making based on neutrality aggregation operator under single valued neutrosophic environment: A case study of airline companies, *Neutrosophic Operational Research,* 471-492.

Kaya, T., Öztürk Kamisli, Z. (2020). Feature analysis for multi-criteria rating values of airline companies, *SDÜ Mühendislik Bilimleri ve Tasarım Dergisi.*

Kaya, T. (2017). Hybrid approach on airline companies, *8th international advanced technologies symposium*, Elazığ, Türkiye.

Yakut, I., **Turkoglu, T.**, Yakut, F. (2015). Understanding customers' evaluations through mining airline reviews, *Internation Journal of Data mining Knowledge Management Process*, 5(6), 1-10.

Yakut, I., **Turkoglu, T.** (2016). Evaluating multi-criteria flight reviews with similarity-based clustering, *6th World Conference on Soft Computing*, Berkeley, California, USA.