ROL MADENCİLİĞİNDE AHC KULLANARAK ROL TABANLI ERİŞİM KONTROL SİSTEMLERİNİN YENİDEN YAPILANDIRILMASI: BİR YAZILIM ENDÜSTRİSİ DENEYİMİ

Osman DURDAĞ

Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY

Yüksek Lisans Tezi
Bilgisayar Mühendisliği Ana Bilim Dalı

2023

T.C.
ATATÜRK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

ROL MADENCİLİĞİNDE AHC KULLANARAK ROL TABANLI ERİŞİM KONTROL SİSTEMLERİNİN YENİDEN YAPILANDIRILMASI: BİR YAZILIM ENDÜSTRİSİ DENEYİMİ

(Reconfiguring Role-Based Access Control Systems by Using AHC in Role Mining: A Software Industry Experience)

YÜKSEK LİSANS TEZİ

Osman DURDAĞ

Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY

Erzurum

Temmuz, 2023

KABUL VE ONAY TUTANAĞI

Osman DURDAĞ tarafından hazırlanan "ROL MADENCİLİĞİNDE AHC KULLANARAK ROL TABANLI ERİŞİM KONTROL SİSTEMLERİNİN YENİDEN YAPILANDIRILMASI: BİR YAZILIM ENDÜSTRİSİ DENEYİMİ" başlıklı çalışması 18/07/2023 tarihinde yapılan tez savunma sınavı sonucunda başarılı bulunarak jürimiz tarafından Bilgisayar Mühendisliği Ana Bilim Dalı, Bilgisayar Mühendisliği Bilim Dalında, yüksek lisans tezi olarak kabul edilmiştir.

Jüri Başkanı:	Doç. Dr. Ayça KOLUKISA *Hacettepe Üniversitesi*	Aslı Islak İmzalıdır
Danışman:	Dr. Öğr. Üyesi Ahmet COŞKUNÇAY *Atatürk Üniversitesi*	Aslı Islak İmzalıdır
Jüri Üyesi:	Doç. Dr. Barış ÖZYER *Atatürk Üniversitesi*	Aslı Islak İmzalıdır

Enstitü Yönetim Kurulunun
…./…/…. tarih ve ……
sayılı kararı.

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili maddelerinde belirtilen şartları yerine getirdiğini onaylarım.

Prof. Dr. Saltuk Buğrahan CEYHUN
Enstitü Müdürü
Aslı Islak İmzalıdır

ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU

Yüksek Lisans Tezi olarak Dr. Öğr. Üyesi Ahmet COŞKUNÇAY danışmanlığında sunulan "ROL MADENCİLİĞİNDE AHC KULLANARAK ROL TABANLI ERİŞİM KONTROL SİSTEMLERİNİN YENİDEN YAPILANDIRILMASI: BİR YAZILIM ENDÜSTRİSİ DENEYİMİ" başlıklı çalışmanın tarafımızdan bilimsel etik ilkelere uyularak yazıldığını, yararlanılan eserlerin kaynakçada gösterildiğini, Fen Bilimleri Enstitüsü tarafından belirlenmiş olan Turnitin Programı benzerlik oranlarının aşılmadığını ve aşağıdaki oranlarda olduğunu beyan ederiz.

Tez Bölümleri	**Tezin Benzerlik Oranı (%)**	**Maksimum Oran (%)**
Giriş	*2*	**30**
Kuramsal Temeller	*11*	**30**
Materyal ve Yöntem	*2*	**35**
Bulgular	*0*	**20**
Tartışma	*0*	**20**
Tezin Geneli	*2*	**25**

Not: *Yedi kelimeye kadar benzerlikler ile Başlık, Kaynakça, İçindekiler, Teşekkür, Dizin ve Ekler kısımları tarama dışı bırakılabilir. Yukarıdaki azami benzerlik oranları yanında tek bir kaynaktan olan benzerlik oranlarının* ***%5'den*** *büyük olmaması gerekir.*

Beyan edilen bilgilerin doğru olduğunu, aksi halde doğacak hukuki sorumlulukları kabul ve beyan ederiz.

Tez Yazarı (Öğrenci)	**Tez Danışmanı**
Osman DURDAĞ	Dr. Öğr. Üyesi Ahmet COŞKUNÇAY
18.7.2023	18.7.2023
İmza: Aslı Islak İmzalıdır	İmza: Aslı Islak İmzalıdır

***** Tez ile ilgili YÖKTEZ'de yayınlamasına ilişkin bir engelleme var ise aşağıdaki alanı doldurunuz.

☐ Tezle ilgili patent başvurusu yapılması / patent alma sürecinin devam etmesi sebebiyle Enstitü Yönetim Kurulunun …./…/…. tarih ve ……. sayılı kararı ile teze erişim 2 (iki) yıl süreyle engellenmiştir.

☐ Enstitü Yönetim Kurulunun …./…/…. tarih ve ……. sayılı kararı ile teze erişim 6 (altı) ay süreyle engellenmiştir.

TEŞEKKÜR

Tez konusu belirleme, araştırma, uygulama ve yazma sürecindeki destek ve katkılarından dolayı danışman hocam Sayın Dr. Öğr. Üyesi Ahmet COŞKUNÇAY'a, yönlendirme ve fikir verme yönündeki desteklerinden dolayı Sayın Doç. Dr. Barış ÖZYER hocama, çalışmamızda verilerini kullanmamıza izin veren Univera şirketine (www.univera.com.tr), tüm tez sürecimde yanımda olan ve bu süreçte katkılarını esirgemeyen arkadaşım Sefa ÖZDEMİR'e, kıymetli dönütler veren değerli mesai arkadaşlarım Aslıhan KILIÇ, Coşkun IRMAK ve tüm Atatürk Üniversitesi ailesine teşekkür ederim. Ayrıca, bana inanan, güvenen, desteklerini esirgemeyerek çalışma azmimi ve motivasyonumu yüksek tutan, her zaman yanımda olan başta annem Hacer DURDAĞ olmak üzere tüm aileme teşekkür ederim.

Osman DURDAĞ

ÖZET

YÜKSEK LİSANS TEZİ
ROL MADENCİLİĞİNDE AHC KULLANARAK ROL TABANLI ERİŞİM KONTROL SİSTEMLERİNİN YENİDEN YAPILANDIRILMASI: BİR YAZILIM ENDÜSTRİSİ DENEYİMİ

Osman DURDAĞ

Danışman: Dr. Öğr. Üyesi Ahmet COŞKUNÇAY

Amaç: Bu çalışmanın amacı, uzun süreli kullanımı sonucunda içerdiği kirli veri sayısı artmış, yönetimi ve sürdürülebilirliği zorlaşmış Rol Tabanlı Erişim Kontrol sistemlerinin Rol Madenciliğinde AHC kullanımı aracılığıyla iyileştirilmesinin ve elde edilen sonuçların yeni bir sistem oluşturma aşamasında referans olarak kullanılabilmesinin mümkün olup olmadığının incelenmesidir.

Yöntem: Bu çalışmada, yazılım endüstrisinde faaliyet gösteren bir yazılım şirketi ile iş birliği yapılmış, şirketin 10 müşterisinin hali hazırda kullandığı rol tabanlı erişim kontrol sistemi tablolarının iyileştirilmesi amacıyla var olan rollerin kümelenmesi doğrultusunda deneysel bir yaklaşım uygulanmıştır. Sunulan müşteri tabloları üzerinde öncelikle veri ön işleme ve öznitelik çıkarımı yapılmıştır. Kümeleme aşamasında, rolleri benzerliklerine göre kümeler halinde gruplandırmak için, yaygın olarak kullanılan ve denetimsiz bir algoritma olan AHC tercih edilmiştir. Oluşacak küme sayısını tahmin etmek amacıyla öncelikle her bir tablo için dendrogram oluşturulmuş ve dendrogramda yapılan farklı kestirim değerleri için küme sayısı belirlenmiştir. Aday kümelerin ikili karşılaştırma metrikleri aracılığıyla doğrulukları hesaplanmıştır.

Bulgular: Elde edilen rol kümelerinin, dendrogram kestirim değeri azaldıkça sayılarının arttığı fakat aynı kümelerdeki rollerin birbirlerine daha çok benzediği tespit edilmiştir. Öte yandan, dendrogram kestirim değeri arttıkça, oluşan rol kümelerinin sayılarının azaldığı ancak aynı kümelerdeki rollerin birbirlerine daha az benzer olduğu görülmüştür. Özellikle kullandıkları sistemde az sayıda roller tanımlanmış olan bazı müşteriler için bazı kestirim değerlerinde uzmanların belirlediği ideal çıktılar elde edilmiştir.

Sonuç: Bulgular, uygulanan yaklaşım sonucunda elde edilen kümelerin, hali hazırda kullanılan erişim kontrol sistemlerinin iyileştirilmesinde referans olarak kullanılabileceğini göstermiştir. Ayrıca dendrogram kestirim noktasının sezgisel olarak belirlenmesinin ve tabloların kullanıcıların inisiyatiflerine bağlı olarak oluşturulmasının mükemmel sonuçları elde etme konusunda sınırlılık meydana getirdiği gözlenmiştir.

Anahtar Kelimeler: Rol tabanlı erişim kontrolü, Rol madenciliği, İkili değerlendirme, Toplayıcı hiyerarşik kümeleme, Veri madenciliği, Hiyerarşik kümeleme, Erişim kontrolü, Performans değerlendirme

Temmuz 2023, 66 sayfa

ABSTRACT

MASTER DISSERTATION

RECONFIGURING ROLE-BASED ACCESS CONTROL SYSTEMS BY USING AHC IN ROLE MINING: A SOFTWARE INDUSTRY EXPERIENCE

Osman DURDAĞ

Supervisor: Assistant Professor Ahmet COŞKUNÇAY

Purpose: As part of this study, the aim is to examine whether it is possible to improve Role-Based Access Control systems, which have become increasingly dirty due to their long-term use and are increasingly challenging to manage and sustain, through the use of AHC in Role Mining, and to determine if the results obtained can be used as a reference in the development of a new system.

Method: In this study, an experimental approach was applied in cooperation with a company in the software industry to improve the role-based access control system tables used by 10 customers of the company, in line with the clustering of existing roles. First of all, data preprocessing and feature extraction were performed on the presented customer tables. In the clustering phase, a widely used and unsupervised hierarchical clustering algorithm, AHC, has been preferred to cluster roles according to their similarities. In order to predict the number of clusters to be formed, firstly, a dendrogram was created for each table, and the number of clusters was determined according to the results obtained through the dendrogram for different threshold values. The trueness of the obtained clusters was calculated by means of pairwise comparison metrics.

Findings: It was determined that the obtained role clusters increased in number as the threshold value of the dendrogram decreased, but the roles in the same clusters were more similar to each other. On the other hand, as the threshold value of the dendrogram increased, it was observed that the number of role clusters decreased, but the roles in the same clusters were less similar to each other. Especially for some customers who have defined a small number of roles in the system they use, the ideal outputs determined by the experts have been obtained in some of the threshold values.

Results: The findings showed that the clusters obtained as a result of the applied approach can be used as a reference to improve the access control systems in use. In addition, it has been observed that the intuitive determination of the threshold value of the dendrogram and the creation of the tables depending on the initiative of the users create limitations in obtaining excellent results.

Keywords: Role-based access control, Role mining, Pairwise evaluation, Agglomerative hierarchical clustering, Data mining, Hierarchical clustering, Access control, Performance evaluation

July 2023, 66 pages

İÇİNDEKİLER

TABLOLAR DİZİNİ

ŞEKİLLER DİZİNİ

KISALTMALAR VE SİMGELER DİZİNİ

AHC : Agglomerative Hierarchical Clustering
API : Application Programming Interface
BAUM : Bilgisayar Bilimleri Uygulama ve Araştırma Merkezi
C : Create Permission
D : Delete Permission
ERP : Enterprise Resource Planning
FN : False Negative
FP : False Positive
gT : Ground Truth
KTÜ : Karadeniz Teknik Üniversitesi
NIST : National Institute of Standards and Technology
NP : Nondeterministic Polynomial
PA : Permission Assignments
pF1 : Pairwise F1
pP : Pairwise Precision
pR : Pairwise Recall
R : Read Permission
RBAC : Role Based Access Control
RMP : Role Mining Problem
TN : True Negative
TP : True Positive
U : Update Permission
UA : User Assignments
UPA : User-Permission Assignments
URL : Uniform Resource Locator

GİRİŞ

Rol tabanlı erişim kontrolü (Role-Based Access Control, RBAC), günümüzde birçok kuruluş ve sistem tarafından güvenliği sağlamak amacıyla yaygın bir şekilde kullanılan (Dekker vd, 2008), yönetimi kolay ve verimliliği yüksek bir erişim denetleme yöntemidir (Hu vd, 2010; Yang vd, 2013). RBAC sistemlerinde kullanıcıların sahip olacakları işlevlerle ilişkilendirilmesi işlemi sistemde tanımlanmış roller aracılığıyla gerçekleştirilmektedir (Sandhu vd, 1996). Geleneksel yöntemde, sahip olacakları izinler kullanıcılara tek tek atanmakta ve bu ilişkilendirmelerin kayıtları kullanıcı-izin tabloları üzerinde tutulmaktadır. RBAC sistemlerinde ise kuruluşun veya sistemin işlevleri roller aracılığıyla tanımlanmakta ve kullanıcılar bu rollerle ilişkilendirilmektedir (Hu vd, 2010). Yani RBAC sistemlerinde kullanıcılara izinlerin atanması işlemi 2 basamaklı bir şekilde gerçekleştirilmektedir.

RBAC sistemlerinde oluşturulan rollerin sayısı genellikle kullanıcı sayısından daha az olduğu için geleneksel yöntemin karmaşıklık seviyesi azalmakta (Vaidya vd, 2010a) ve böylece sistemin yönetilebilirliği büyük ölçüde iyileşmiş olmaktadır (Anderer vd, 2021). RBAC tabanlı sistemlerin kullanımının tüm farklı yönetim faaliyetleri için zaman tasarrufu sağladığını bir NIST raporu göstermektedir (Gallaher vd, 2002). Rollerin iş birimlerini veya belirli işlev kümelerini tanımlaması, sistemden bir kullanıcının ayrılması veya birim değiştirmesi durumunda ortaya çıkabilecek karışıklığın iyi bir şekilde yönetilebilmesini sağlamaktadır (Anderer vd, 2020). Kullanıcının izinlerinin silinmesi veya değiştirilmesi yerine kullanıcı ile daha önce atanmış olduğu rol arasındaki ilişki kesilmekte veya kullanıcı farklı bir role atanmaktadır. RBAC sistemlerinin önemi ve faydası, büyük ölçekli sistem ve kuruluşlarda uygulandıklarında gerçek anlamda ortaya çıkmaktadır. RBAC, faydası ve yönetilebilirliği birçok şirket ve sistem tarafından kabul edilmiş bir yaklaşım olarak günümüzde yaygın bir şekilde kullanılmaktadır (Liu vd, 2022).

RBAC sistemlerinin faydasının gerçek anlamda ortaya çıkabilmesi için rollerin eksiksiz, doğru ve verimli bir şekilde tanımlanması görevi olan rol mühendisliği süreci, önemli bir gereklilik olarak görülmektedir (E. J. Coyne, 1996). Rolleri belirleme ve onlara izinler atama süreçlerinin dahil olduğu RBAC yapısının oluşturulma süreci rol mühendisliği olarak tanımlanmaktadır (E. Coyne vd, 2011). Rol mühendisliği, daha önce bahsedilen NIST çalışmasına göre RBAC sisteminin hayata geçme sürecindeki en maliyetli bileşenlerden biri ve yinelemeli bir süreç olarak kabul edilmektedir (Gallaher vd, 2002). Rol mühendisliğinin zaman alıcı ve maliyetli bir süreç olması, hali hazırda alışılagelmiş yöntem ile erişim kontrol

sürecini sağlayan şirketleri ve kuruluşları RBAC sistemine geçiş konusunda çekingen davranmaya itmektedir.

Rol mühendisliği için aşağıdan yukarıya ve yukarıdan aşağıya olmak üzere temel olarak 2 yaklaşım bulunmaktadır (Sun vd, 2019). Yukarıdan aşağıya yaklaşımda RBAC sisteminin oluşturulması, iş süreçlerinin çok dikkatli ve titiz bir şekilde analiz edilmesiyle başlayıp, belirli iş birimlerini en iyi şekilde temsil edecek rollerin oluşturulmasıyla devam eden ve son olarak oluşturulmuş rollere ilgili izinlerin atanması ile tamamlanan bir süreç olarak ele alınmaktadır. Fakat bu süreç, özellikle büyük sistemler göz önüne alındığında, çok fazla sayıda kullanıcı, iş birimi ve izinler üzerinde gerçekleştirilmeye çalışıldığı için zorlu ve zaman alıcı olabilmektedir. Rol madenciliği olarak adlandırılan (Ene vd, 2008; Lu vd, 2008; Vaidya vd, 2010a) aşağıdan yukarıya yaklaşımda ise hali hazırda kullanılan geleneksel erişim kontrol sisteminde tanımlanmış izinlerden yola çıkarak, veri madenciliği algoritmaları ile rollerin sistematik bir şekilde oluşturulma çabası yer almaktadır. Aşağıdan yukarıya yaklaşım modeli RBAC sisteminin oluşturulma sürecini otomatikleştirdiği için avantajlı olsa da var olan kullanıcı-izin tablosunda iş birimlerine yönelik anlamlı bir veri bulunmaması durumunda yaklaşımın beklenen sonuçları üretme konusundaki başarı oranı düşebilmektedir. RBAC sistemlerini oluşturma sürecinde bu iki yaklaşımın ortak olarak değerlendirildiği hibrit bir yaklaşım da kullanılabilmektedir (Molloy vd, 2010).

RBAC'ın verimliliği yıllar boyunca birçok kuruluş tarafından kabul görmüştür (Kirkpatrick ve Bertino, 2010). Öte yandan, çok sayıda kullanıcı veya iş biriminin bulunduğu büyük ölçekli şirketlerde iş gereksinimlerinin sürekli değişmesinin yanı sıra büyük miktarda verinin işlenmesi ve depolanması görevinin karmaşıklığı artmaktadır (Leblebici vd, 2022). Bu durumda erişim kontrol sistemlerinde RBAC yapısı kullanıldığında bile sistemin yönetiminin zorlaştığı görülebilmektedir (Hu vd, 2010). RBAC sistemleri, sisteme yeni kullanıcılar veya iş birimleri eklendikçe büyümeye devam etmektedir. Ayrıca, ilişkili oldukları rollerin veya izinlerin sistemde varlığını sürdürmeye devam etmesi gibi bazı durumlarda, sistemden kullanıcı veya iş birimlerinin silinmesi de sistemin büyümesine neden olabilmektedir. Erişim kontrol sistemi yöneticileri, sisteme eklenmesi gereken her yeni kullanıcı veya izin için yeni roller tanımlayabilmektedir (Anderer vd, 2020). Ayrıca aynı iş pozisyonunu paylaşan kullanıcıların aynı izinlere sahip olup olmayacağını düşünmeden her yeni iş pozisyonu için yeni bir rol oluşturabilmektedirler (Calo vd, 2008). Bu gibi durumlarda kullanıcılar kuruluştan ayrıldığında veya iş pozisyonlarını değiştirdiğinde artık kullanılmayan, yani atanmış herhangi bir kullanıcısı olmayan roller de ortaya çıkabilmektedir. Üst yönetim ayrıca izin değişiklikleri getirerek veya yeni rol oluşturma zorunluluğu getirerek RBAC sistem tasarımına müdahale

edebilmekte ve böylece RBAC yöneticisinin sistem üzerindeki kontrolünü kaybetmesine neden olabilmektedir (Saenko ve Kotenko, 2017). Bu tür değişiklikler RBAC sisteminin zamanla yıpranmasına ve dağınık hale gelmesine neden olmaktadır. Mevcut RBAC sistemlerinin yeniden yapılandırılması, yeni bir RBAC sistemi tasarlamak kadar maliyetli olmaktadır (Calo vd, 2008). Mevcut RBAC sistemlerini geliştirmek için önerilen yaklaşımda amaç, mevcut rollerin izin benzerliklerine dayalı olarak kümelenmesiyle oluşan rol kümelerinin, hali hazırda var olan RBAC sistemlerinin iyileştirilmesi aşamasında rol-izin matrislerini tanımlamak için veya oluşturulabilecek yeni bir sistem için referans olarak kullanılıp kullanılamayacağına yönelik deneysel bir çalışma yapmaktır. Bu yaklaşımın, rol madenciliği teknikleri aracılığıyla mevcut olan RBAC sistemlerinin iyileştirilmesi konusunda var olan yöntemlere kıyasla potansiyel olarak daha az maliyetli olacağı düşünülmektedir. Literatür incelendiğinde, rol sayısını belirli bir izin benzerliği ölçüsüyle kümeleyerek azaltma ve dolayısıyla sistem yönetilebilirliğini iyileştirme yönünde araştırma yapan bir çalışmaya rastlanmamıştır.

Bu tezde, sahip oldukları izinlerin benzerliklerine göre rolleri kümeleyerek, hali hazırda kullanılan RBAC sistemlerini iyileştirmek için RBAC sistemlerini yeniden yapılandırmayı amaçlayan deneysel bir çalışma sunulmaktadır. Bu çalışma kapsamında, 40'tan fazlası Ar-Ge çalışanı olmak üzere 160'ın üzerinde çalışanı bulunan, 130'dan fazla projeyi hayata geçirmiş ve 25000'den fazla mobil kullanıcısı olan bir yazılım firması ile görüşülmüştür. Bu çalışmanın deneysel kısmı, sentetik veriler kullanmak yerine ilgili şirketin 10 farklı müşterisinden elde edilen mevcut maskelenmiş veriler üzerinde gerçekleştirilmiştir. Çalışmada sırasıyla veri ön işleme, öznitelik çıkarımı, benzerlik mesafesi ölçümü, mesafe tabanlı dendrogram çıkarımı, dendrogram kullanılarak küme sayısı tahmini yapılmış ve Toplayıcı Hiyerarşik Kümeleme (Agglomerative Hierarchical Clustering, AHC) işlemleri için Python tabanlı yardımcı kütüphanelerden faydalanılarak rol kümeleri oluşturulmuştur. Daha sonra sonuçların doğruluğu ve yeterliliği hakkında bir tartışma yapılmıştır.

Alan yazındaki çalışmalar aşağıdan yukarıya, yukarıdan aşağıya ve hibrit yaklaşımların örneklerini temsil etmektedir. Mevcut izinlerden aday rollerin keşfedilmesi, hali hazırda varlığını sürdüren RBAC sistem yapılarının iyileştirilmesi ve bu işlemlerin nasıl daha verimli gerçekleştirilebileceğine yönelik çalışmalar bulunmaktadır. Bununla birlikte, rollerin birkaç örtüşen ayrıcalığa sahip bir şekilde bulunduğu ve eskimiş bir RBAC sisteminin ortaya çıktığı yaygın bir durum için, çağdaş çalışmalar, kümeleme yoluyla rol sayısını azaltarak ve böylece yönetim kolaylığını artırarak RBAC sistemlerini yeniden yapılandırma konusunda sınırlı bir rehberliğe sahiptir. Bu doğrultuda çalışmanın amacı ve temel motivasyonu, mevcut

RBAC sistemlerini yeniden yapılandırılması, ayrıcalık benzerliğine dayalı olarak rollerin kümelenmesi, böylece var olan rollerin değiştirilmesi veya AHC yoluyla kümelenmiş bir dizi rol ile yeni bir RBAC sistemi oluşturulması konusunda rehberlik sağlamaktır.

Bu tezin sonraki bölümleri aşağıdaki gibi organize edilmiştir. Kuramsal çerçeve ve ilgili çalışmalar kuramsal çerçeve başlığı altında ikinci bölümde anlatılmaktadır. Üçüncü bölümde materyal ve yöntem detayları ile ele alınmaktadır. Dördüncü bölümde deneysel çalışma sonucunda elde edilmiş araştırma bulguları sunulmaktadır. Beşinci bölümde tartışma yer almaktadır. Altıncı ve son bölümde ise sonuçlar ve öneriler yer almaktadır.

KURAMSAL TEMELLER

Bu bölümde erişim kontrol sistemleri, geleneksel erişim kontrol sistemleri, RBAC sistemleri, RBAC sistemlerinin kullanımının sağladığı kolaylıklar ve önemi, rol madenciliği ve rol mühendisliği, kümeleme, AHC, Jaccard metriği ve mesafesi, AHC için kullanılan bağlantı yöntemleri ve dendrogramlar literatürde bulunan mevcut çalışmalardan yararlanılarak anlatılmaktadır.

Erişim Kontrol Sistemleri

Kullanıcıların bilgilere, kaynaklara, hizmetlere vb. erişimini kontrol etmek ve kötü niyetli saldırılara engel olmak için kullanılan modellerin ve sistemlerin tamamına atıfta bulunan (Zhang vd, 2015) ve bilgisayar sistemlerinin temeli olan erişim kontrolü, ilk olarak 1970'li yılların başlarında Lampson tarafından önerilmiştir (Lampson, 1971). Erişim kontrolü, işletim sistemlerinde, veri tabanı yönetim sistemlerinde, internet tabanlı çözümlerde vb. uygulamalarda sistem güvenliğini sağlamak ve hassas kaynaklara erişimi kısıtlamak amacıyla kullanılan önemli bir güvenlik mekanizmasıdır (Le vd, 2022).

Erişim kontrolü ve sistem güvenliğini anlayabilmek için öncelikle kullanıcılar ve özneler arasındaki fark iyi anlaşılmalıdır. Bu ayrım temel olmasına rağmen genellikle kesin olmayan bir şekilde ele alınmakta ve güvenliğin amaçları hakkında gereksiz bir kafa karışıklığına yol açmaktadır. Kullanıcının insan olduğu açıktır ve sistem tarafından bilinen her insanın benzersiz bir kullanıcı olduğu varsayılmaktadır. Başka bir deyişle Ayşe adlı benzersiz bir insan sistemde birden fazla kullanıcı kimliğine sahip olamamaktadır. Eğer Ayşe, sistemin yetkili bir kullanıcısı değilse kullanıcı kimliği de bulunmayacaktır. Öte yandan, eğer Ayşe yetkili bir kullanıcı ise sistemde tam olarak kendisi ile ilişkili bir kullanıcı kimliği var olacaktır. Bir özne ise sistemdeki bir süreç olarak tanımlanmaktadır. Yani bir özne, bilgisayar sistemleri için yürütülmekte olan bir program veya web tabanlı sistemler için çalıştırılan bir fonksiyon veya süreçtir. Her özne o özneyi çalıştıran veya tetikleyen kullanıcı adına yürütülmektedir. Genel olarak, bir kullanıcı sistemde aynı anda çalışan birçok özneye sahip olabilir. Dolayısıyla her özne de o özneyi yürüten kullanıcı ile ilişkilidir (Sandhu, 1993).

Erişim kontrolü (Sandhu ve Samarati, 1994), Zhu ve arkadaşları tarafından hassas bilgilerin ve kritik sistem kaynaklarının korunması için önemli bir güvenlik mekanizması olarak tanımlanmıştır (Zhu vd, 2019). Lazouski ve arkadaşları ise erişim kontrolünü belirli bir varlık tarafından belirli bir kaynağa erişim izni verme veya reddetme kabiliyeti olarak

tanımlamışlardır (Lazouski vd, 2010). Diğer bir yandan Li ve arkadaşları, erişim kontrolünü, yasa dışı kullanıcıların sistemdeki kaynaklara erişimlerini yasaklayabilen ve sistemdeki meşru kullanıcıların yetkisiz işlemlerini engelleyen bir yapı olarak yorumlamışlardır (Li vd, 2022).

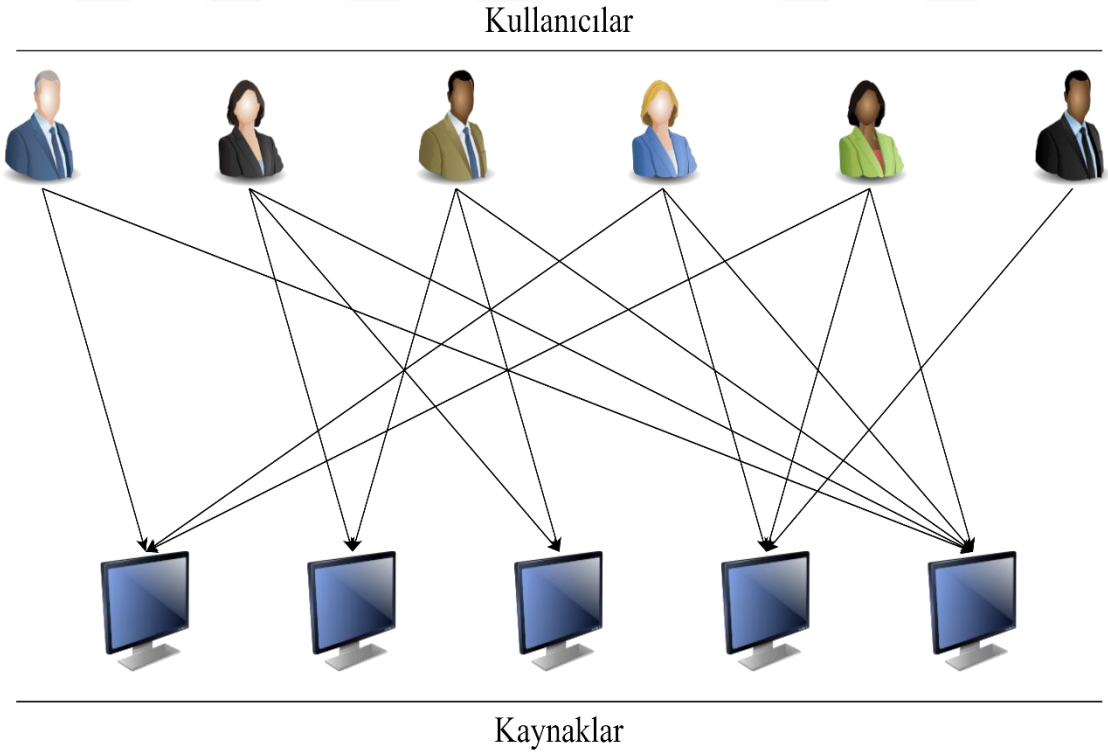
Erişim kontrol sistemlerinde; erişim talebinde bulunan özneler, öznelerin isteklerini alan ve yetki verilip verilmediğini belirleyen nesneler ve öznelerin kaynaklara erişim hakkının olup olmadığını belirleyen izin matrisleri olmak üzere 3 temel unsur bulunmaktadır (Shahen vd, 2021). Bu unsurların temel amaçları aşağıdaki gibi özetlenebilir:

- Kullanıcıların ulaşması gereken kaynaklara erişim sağlamasına izin vermek,
- Kullanıcıların ulaşmaması gereken kaynaklara erişimlerini engellemek,
- Kullanıcı olmayan öznelerin kaynaklara erişimlerini engellemek.

Erişim kontrol sistemleri geleneksel erişim kontrol sistemleri ve rol tabanlı erişim kontrol istemleri olmak üzere genel olarak ikiye ayrılmaktadır.

Geleneksel erişim kontrol sistemleri

Geleneksel erişim kontrol sistemlerinde kullanıcılar, erişmek istedikleri kaynaklara ancak ilgili kaynağa erişim izinleri olduğunda erişebilirler (Cai vd, 2020). Şekil 1'de geleneksel erişim kontrol sistemi yapısı grafiksel olarak ifade edilmiştir.



Şekil 1. Geleneksel erişim kontrol sistemi

Geleneksel erişim kontrol sistemlerinde kullanıcıların ilgili kaynaklara erişim izni hakları 1 ve 0 değerlerinden oluşan tablolardan faydalanılarak yönetilmektedir. Bu tablo veya matris, kullanıcı izin ataması (User-Permission Assignments, UPA) tablosu olarak adlandırılmaktadır (John vd, 2012). Şekil 1'de örneği verilen geleneksel kontrol sisteminin matrise aktarılmış UPA formundaki bir örneği Tablo 1'de gösterilmiştir.

Tablo 1. UPA Tablosu Örneği

Kullanıcı	İzin 1	İzin 2	İzin 3	İzin 4	İzin 5
Kullanıcı 1	1	0	0	0	1
Kullanıcı 2	0	1	1	0	1
Kullanıcı 3	0	1	1	0	1
Kullanıcı 4	1	0	0	1	1
Kullanıcı 5	1	0	0	1	1
Kullanıcı 6	0	0	0	1	0

Çok sayıda kullanıcıya ve yapılan işlerin büyüklüğü doğrultusunda çok sayıda izne sahip bir kurum veya şirkette kullanılan erişim kontrol sistemlerindeki UPA tablolarının boyutları yıllar geçtikçe büyük ölçüde büyüyebilmekte ve yönetimi çok zor bir hâle gelebilmektedir. Uzun süre boyunca hizmet veren erişim kontrol sistemlerinde zamanla birbirini tekrarlayan satırlar veya gözden kaçan gereksiz veriler birikerek sistemin yönetilebilirlik seviyesini önemli ölçüde azaltmaktadır. Ayrıca Saxena ve Alam, geleneksel erişim kontrol sistemlerinin çoklu ilişkiler, dinamik değişiklikler, sanallaştırma vb. gereksinimleri karşılamak için yeterli olmadığını ileri sürmüştür (Saxena ve Alam, 2022).

Rol tabanlı erişim kontrol sistemleri

Bilgisayar sistemlerinde, 1970'li yıllardan itibaren zamanla birden çok kullanıcıya hizmet veren uygulama sayısı çoğalmış ve bu durum veri güvenliği sorunlarının farkındalığını artırmıştır (Kamboj vd, 2021). Öyle ki, üstünkörü bir bakış bile, modern organizasyonların genellikle departman ve organizasyonel sınırlarını aşan karmaşık faaliyetlerinin olduğunu ortaya çıkarmak için yeterli olmaktadır (Thomas ve Sandhu, 1994). Sistem yöneticileri ve yazılım üreticiler, bazı verilere ve belgelere yalnızca yetkili kullanıcıların erişebilmesini sağlamak amacıyla yeni erişim kontrol sistemleri üretmeye çalışmıştır. Bu doğrultuda ortaya çıkan erişim kontrol sistemlerinden biri RBAC olmuştur (Sandhu vd, 1996). RBAC sistemleri ilk ortaya atılmasından (D. Ferraiolo ve Kuhn, 1992) bu yana, büyük kurumsal ortamlarda güvenlik kontrolü için önerilen bir uygulama halinde gelmiştir (Ghazal vd, 2021; D. Zhang vd, 2008) ve erişim kontrolü için etkili ve popüler bir araç olarak kullanılmaktadır (D. F.

Ferraiolo vd, 2001; Zarnett vd, 2010). RBAC sistemlerinin odak noktası öznelerden çok roller için haklar ve ayrıcalıklar tanımlamaktır (Hassen vd, 2007).

Rol, erişim kontrol politikasının temelini oluşturan anlamsal bir yapıdır. RBAC ile sistem yöneticileri, bir şirkette veya kuruluşta gerçekleştirilen ve gerçekleştirilecek olan iş ve görevlere uygun roller oluşturmakta, bu rollere izinler (erişim yetkisi) vermekte, ardından kullanıcıları belirli iş sorumluluklarına ve niteliklerine göre uygun olan rollerle ilişkilendirmektedir (Vaidya vd, 2010b).

Bir rol; bir doktorun, bir eczacının veya bir proje danışmanının belirli görev yetkinliklerini temsil ederek ilgili öznelerin yetki ve sorumluluklarını somutlaştırabilir. Yetki ve sorumluluk, yeterlilikten veya yetkinlikten farklıdır. Bir kişi birkaç departmanı yönetme konusunda yetkin bir durumda olabilir ancak yalnızca fiilen yönettiği departmanın sorumluluğuna sahip olabilir. Roller ayrıca görevli bir hekimi veya bir vardiya yöneticisini de yansıtabilir. RBAC modelleri ve uygulamaları, rol kavramının tüm bu tezahürlerini barındırarak uygun bir şekilde oluşturulmalıdır (Ferraiolo vd, 1993).

Roller, hem kaynaklara erişmesine izin verilen belirli kişileri hem de kaynaklara ne ölçüde erişilebileceğini tanımlamaktadır (Anderer vd, 2020). Örneğin; bir operatör rolü tüm bilgisayar kaynaklarına erişebilirken erişim izinlerini değiştiremeyecek şekilde ayarlanabilir, bir güvenlik görevlisi rolü bazı izinleri değiştirebilirken aynı rolün kaynaklara erişimi kısıtlanabilir ve bir denetçi rolü yalnızca denetim yollarına erişebilecek duruma getirilebilir. Novell NetWare ve Microsoft Windows NT gibi ağ işletim sistemlerinde sistem yönetimi için roller kullanılmaktadır (Sandhu vd, 1996).

RBAC, bilgi işlem sistemlerinde kullanıcıların ve kullanıcıların sahip oldukları ayrıcalık veya izinlerin yönetimini basitleştirmek için kullanılan etkili bir mekanizmadır (D. F. Ferraiolo vd, 2001; Sandhu vd, 1996). Büyük kuruluşların çok sayıda kullanıcıya güvenli bir erişim kontrol sistemi kurma ihtiyacına yönelik bir çözüm olarak ortaya çıkan (Fragkos vd, 2022) rol tabanlı erişim kontrol sistemlerinde, geleneksel erişim kontrol sistemlerinden farklı olarak kullanıcılara izinler doğrudan atanmamaktadır (Colantonio vd, 2009). Bunun yerine izinler gruplandırılarak roller oluşturulmaktadır (Abdul vd, 2022). Daha sonra oluşturulan rollerin kullanıcılara atanmasıyla, kullanıcıların doğal olarak ilgili izinlerle ilişkilendirilmiş ve yetkilendirilmiş duruma gelmesi sağlanmaktadır (Hasebe vd, 2010). RBAC sistemlerinin faydası, kullanılan rol setinin organizasyonun ihtiyaçlarına uygun olması durumunda belirgin bir şekilde ortaya çıkmaktadır (Mitra vd, 2016). Yetkilendirme işlemleri için RBAC sistemleri tercih edildiğinde değişen organizasyonel ihtiyaçlarla başa çıkma durumu daha kolay bir hâl almaktadır (Garg vd, 2023). Ayrıca izinlerden oluşan roller,

organizasyonun fonksiyonel gereksinimlerini izinler aracılığıyla kapsadığı için ilgili görevleri yürüten personeller değişse, yer değiştirse veya işten ayrılsa bile personellere atanan roller değiştirilerek veya kaldırılarak personellerin yetkileri kolaylıkla yönetilebilmektedir (Vaidya vd, 2010b). Böylece sistemin yönetim maliyeti ciddi ölçüde azalmaktadır (Mitra vd, 2016). Tüm bu işlemlerin kolaylıkla ve başarılı bir şekilde yönetilebilmesi için rollerin eksiksiz ve doğru bir şekilde oluşturulması gerekmektedir (Vaidya vd, 2010b).

RBAC oluşturmadaki en temel görev, kullanıcı rolü atamalarının ve rollere atanacak izinlerin en uygun şekilde yapılandırılmasıdır (Mitra vd, 2016). Geleneksel yapıya uygun bir şekilde oluşturulmuş olan sistemde, kullanıcıların hangi kaynaklara erişimi olduğunu belirten kullanıcı-izin ilişki matrisi, bir Boole matrisi olarak gösterilebilir ve UPA şeklinde adlandırılmaktadır. Bu matrisin satırları kullanıcılara, sütunları ise izinlere karşılık gelmektedir. Bir kullanıcıya belirli bir kaynağa ulaşabilmesi veya belirli bir fonksiyonu yerine getirebilmesi amacıyla izin verileceği zaman, UPA matrisinde ilgili kaynak ve kullanıcının kesiştiği hücrenin değeri 1 olacak şekilde ayarlanmaktadır. Aynı şekilde belirli bir kaynağa belirli bir kullanıcının erişimine engel getirileceği zaman yine UPA matrisinde ilgili kaynak ve kullanıcının kesiştiği hücreye 0 değeri atanarak kullanıcının ilgili kaynağa erişimi kısıtlanmaktadır.

Tablo 2. PA Tablosu Örneği

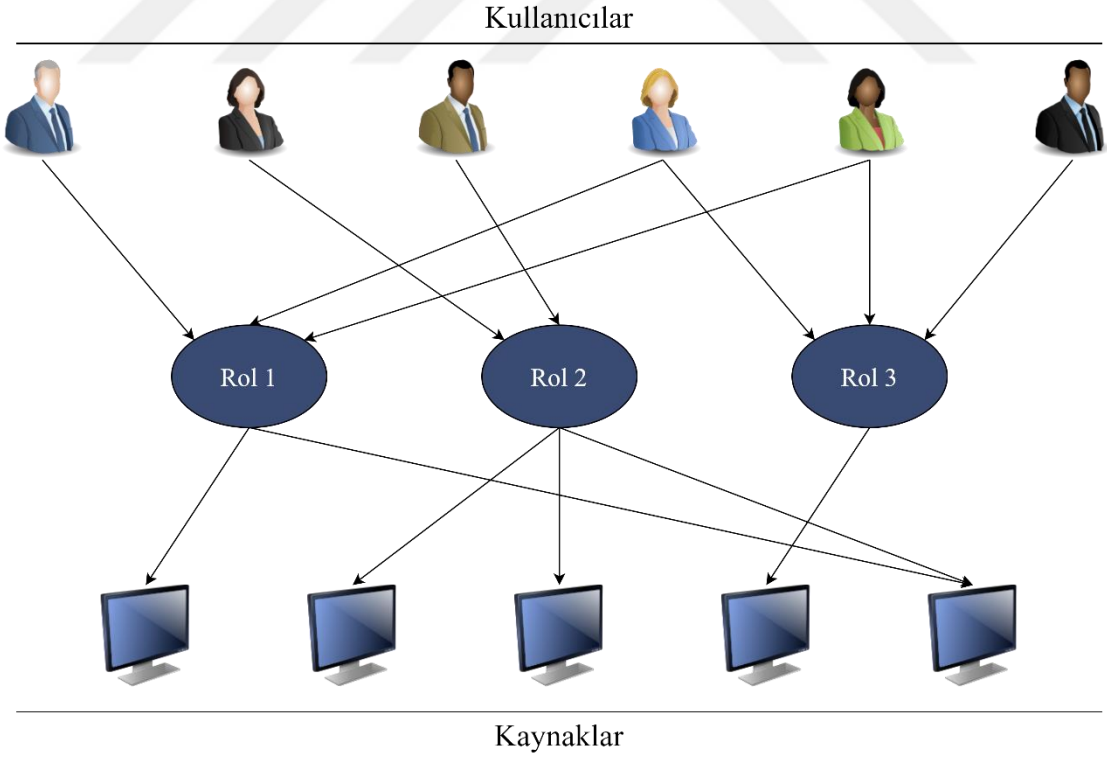
Rol	İzin 1	İzin 2	İzin 3	İzin 4	İzin 5
Rol 1	1	0	0	0	1
Rol 2	0	1	1	0	1
Rol 3	0	0	0	1	0

Tablo 3. UA Tablosu Örneği

Kullanıcı	Rol 1	Rol 2	Rol 3
Kullanıcı 1	1	0	0
Kullanıcı 2	0	1	0
Kullanıcı 3	0	1	0
Kullanıcı 4	1	0	1
Kullanıcı 5	1	0	1
Kullanıcı 6	0	0	1

RBAC sistemlerinde belirli kaynaklara belirli kullanıcıların erişim izinleri benzer şekilde yine Boole matrisleri aracılığıyla yönetilmektedir. Fakat RBAC sistemlerinde,

geleneksel yaklaşımdan farklı olarak yalnızca UPA tablosu yerine User-Assignments (UA) ve Permission-Assignments (PA) olmak üzere 2 farklı tablo kullanılmaktadır (Bonatti vd, 2015). PA tablosunda, oluşturulan roller ve bu rollerin sahip olacakları ayrıcalık veya izinler belirtilmektedir. UA tablosu ise kullanıcı ve rollerden oluşan bir tablodur ve bu tablo aracılığıyla kullanıcılar rollerle ilişkilendirilmektedir. Bir kullanıcı, UA tablosunda bir rolle ilişkilendirildiğinde, o rolün PA tablosunda belirtilen ayrıcalıklarına doğal olarak sahip olmuş olacaktır. RBAC sistemlerinde erişim kontrolünü yönetmek için kullanılan bu 2 tablo, kullanıcılara verilecek olan izinlerin roller aracılığıyla yönetilmesine odaklanan bir amaç doğrultusunda hizmet etmektedir (Rao vd, 2021). RBAC sistemlerinde kullanıcıların roller aracılığıyla izinleri devralması 2 adımlı bir süreç olarak ele alınabilir. İlk adımda kullanıcıya verilmek istenen izinlere veya ayrıcalıklara sahip olan bir rol daha önce oluşturulmuş mu diye kontrol edilir ve oluşturulmamışsa oluşturulur. İkinci adımda ise kullanıcı ilgili rolle ilişkilendirilir veya ilgili role atanır. Bu atama veya ilişkilendirme işlemi aracılığıyla kullanıcı doğal olarak rolün sahip olduğu yetkilere sahip olacaktır. Bir önceki bölümde verilen UPA tablosu örneğine karşılık gelen UA (Tablo 2) ve PA (Tablo 3) tablolarının matris formülasyonu Tablo 2 ve Tablo 3'te gösterilmiştir. Şekil 2'de ise RBAC sistemi örneği verilmiştir.



Şekil 2. Rol tabanlı erişim kontrol sistemi

RBAC sistemlerinin başarılı bir şekilde uygulanması, bazı hatalı bilgiler içermesi durumunda girdi verilerinin temizlenmesi, oluşturulan rollerin yönetimi ve herhangi bir anormalliği tespit etmek amacıyla denetim yapılması gibi uygulamaları da gerektirebilmektedir.

RBAC sistemleri geliştirilmeye ve farklı yaklaşımlarla entegre bir şekilde yapılandırılarak sistemlerin güvenliğini sağlamak amacıyla kullanılmaya devam etmektedir. Akıllı sözleşmelerde de kullanılan RBAC için, akıllı sözleşmelere daha uygun olduğu ileri sürülerek hiyerarşik bir model önerilmiştir (Crass vd, 2022). Hyder ve arkadaşları, kullanıcıların mahremiyetini koruyan, adli bilişim sürecine müdahale etmeyen ve kolayca uygulanabilen bir sistem geliştirmişlerdir. Kriptografi ve rol tabanlı erişim kontrolü kullanarak geliştirdikleri 5 katmanlı sistemi kriptografili rol tabanlı mobil adli tıp çerçevesi olarak adlandırmış ve erişim kontrolü katmanında RBAC kullanmışlardır (Hyder vd, 2022). Jha ve arkadaşları ise rol tabanlı erişim kontrol politikalarının spesifikasyonu ve yönetiminin mevcut durumunu iyileştirmek amacıyla erişim kontrol sistemleri için resmi doğrulama teknikleri geliştirmeyi önermiş ve bu doğrultuda güvenlik analizi problemlerinin sınıflarını resmileştirdiklerini iddia etmişlerdir (Jha vd, 2008). Bir başka çalışmada, iş akışı tabanlı ERP sistemlerinde güvenlik sorunlarının analizine yardımcı olmak amacıyla, yetkilendirme ve iptal işlemlerinin otomatik analizi için model kontrol tabanlı bir yaklaşım sunulmuştur (Schaad vd, 2006). Reith ve arkadaşları, güvenlik özelliklerinin RT (rol tabanlı güven yönetimi politikası dili) politikaları tarafından karşılanıp karşılanmadığını belirlemede yararlı olan bir azaltma, optimizasyon ve doğrulama teknikleri koleksiyonu sunmuşlardır (Reith vd, 2009). Yazılım evrimi sırasında, iki sürüm arasında meydana gelen ve sistem güvenliğini etkileyen tüm farklılıkları, diğer durumların da güvenliğini potansiyel olarak değiştiren ifadeler olarak tanımlayan Laverdière ve arkadaşları, bir uygulamanın iki versiyonu arasındaki güvenliği etkileyen değişiklikleri tanımlayan otomatik bir statik analiz yöntemi önermişlerdir (Laverdière vd, 2021). RBAC veya türevlerine model kontrol teknikleri uygulayan alternatif çalışmalar da literatürde mevcuttur (Ahmed ve Tripathi, 2003; Mondal vd, 2009).

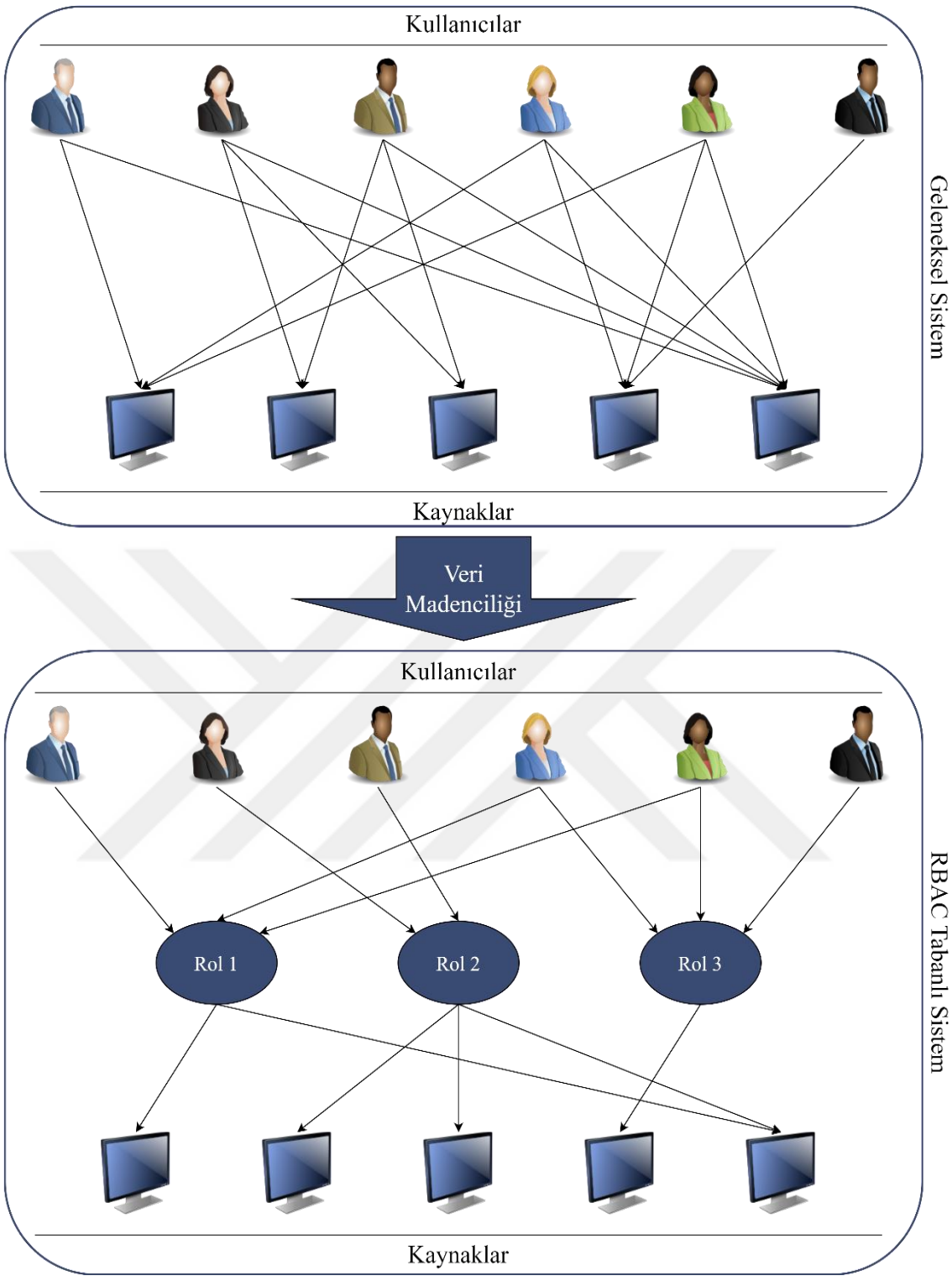
Rol Madenciliği ve Rol Mühendisliği

Farklı izinleri bir araya getirerek gerekli roller setini belirleme süreci, Rol Mühendisliği olarak bilinmektedir (Baumgrass vd, 2011; Colantonio vd, 2008; E. J. Coyne, 1996; Shin vd, 2003). Rol mühendisliğinin, RBAC sistemlerini oluşturma aşamasında kullanılabilecek en maliyetli ve zaman alıcı yaklaşımlardan biri olduğu belirlenmiştir (O'Connor ve Loomis, 2010; Vaidya vd, 2010b).

Rol mühendisliği geleneksel olarak yukarıdan aşağıya yöntemle gerçekleştirilmektedir (Neumann ve Strembeck, 2002; Roeckle vd, 2000). Yukarıdan aşağıya yaklaşımda öncelikle kurumsal iş süreçleri analiz edilmektedir ve daha küçük birimlere ayrıştırılmaktadır. Belirli görevleri yerine getirebilmek için gerekli olan izinler belirlendikten sonra, ilgili izinler onlara uygun olan fonksiyonel roller halinde gruplandırılmaktadır. Bu yaklaşım tüm iş fonksiyonları tamamlanıncaya kadar tekrarlanmaktadır. İş süreçlerinin, kullanıcıların ve izinlerin sayısı zamanla artarak çok yüksek sayılara ulaştığında, yukarıdan aşağıya yaklaşımın ölçeklenebilirlik açısından önemli bir dezavantajı ortaya çıkmaktadır. Aynı zamanda bu yaklaşım zaman alıcı olmaktadır ve önemli ölçüde insan müdahalesi gerektirmektedir. Bu sorunları ele almak amacıyla Narouei ve Takabi tarafından doğal dil işleme tekniklerinden yararlanan otomatik bir yukarıdan aşağıya rol mühendisliği yaklaşımı önerilmiştir (Narouei ve Takabi, 2015). Ayrıca, modern erişim kontrol yönetimi araçları, yukarıdan aşağıya rol modelleme işlemini kolaylaştıran iş akışı ve doğrulama fonksiyonları sağlamaktadır (Mitra vd, 2016).

İlerleyen yıllarda, otomatikleştirilebilen bir süreç kullanarak rolleri tanımlamak için mevcut kullanıcı izin veya ayrıcalıklarını dikkate alan, veri madenciliği tekniklerini uygulayan (D. Zhang vd, 2008), rol madenciliği olarak adlandırılan (Ene vd, 2008; Lu vd, 2008; Vaidya vd, 2010a), alternatif bir aşağıdan yukarıya yaklaşım önerilmiştir. Kuhlmann, rol madenciliğini ilk ortaya atan isimlerden biri olmuştur (Kuhlmann vd, 2003). RBAC sistemlerinin rol madenciliği aracılığıyla oluşturulması sürecinin grafiksel temsili Şekil 3'te verilmiştir. Hem yukarıdan aşağıya hem de aşağıdan yukarıya yaklaşımların faydalarını bir araya getirmek amacıyla hibrit rol mühendisliği adı altında teknikler de önerilmiştir (Frank vd, 2009; Fuchs ve Pernul, 2008; Molloy vd, 2010b).

Rol madenciliği bir matris ayrıştırma süreci olarak da ele alınmaktadır (Mitra vd, 2016). Bu açıdan bakıldığında rol madenciliği, daha önce bahsedilen UPA matrisinin, erişim kontrol politikasını eksiksiz bir şekilde ifade eden iki Boole matrisine (UA ve PA) dönüştürülmesidir (Lu vd, 2014). UA ve PA matrislerinin yanı sıra, rol madenciliği aracılığıyla elde edilen çıktılar bazen bir rol hiyerarşisi oluşturacak şekilde rol-rol ilişkileri içerebilmektedir (Mitra vd, 2016). Rol hiyerarşisi, kıdemli ve daha az kıdemli rollerin belirlenmiş olmasıyla, bir organizasyonun veya kurumun hiyerarşik yapısını doğal bir şekilde temsil etmektedir. Bir rol hiyerarşisinde, alt seviyedeki rollerin izinleri üst seviyedeki roller tarafından devralınmakta ve üst seviyedeki rollerin üyesi olan kullanıcılar dolaylı olarak karşılık gelen alt seviyedeki rollerin de üyesi olmaktadırlar.



Şekil 3. Rol madenciliğine genel bir bakış

Rol güncelleme işlemi, rol yaşam döngüsünde rol bakımının temel bir bileşenidir (Kern vd, 2002) ve rol bakımının toplam maliyetinin büyük bir kısmını oluşturmaktadır (Ni vd, 2009). Literatürde bulunan birçok RBAC yönetim modeline rağmen (Crampton, 2005; N. Li ve Mao, 2007; R. Sandhu vd, 1999), genellikle ortaya çıkan durumun çeşitli kısıtları karşılaması beklendiği için rol güncellemesi zordur ve hataya açık bir süreç olarak ele

alınmaktadır (Hu vd, 2010). Örneğin güncelleme işleminin, kullanıcıların rol setini veya izin setini beklenmedik bir şekilde değiştirmemesi ve böylece sistemde yapılan görevleri engellememesi gerekir; gerçek hayatta profesör olan bir kullanıcıya, izinler veya ayrıcalıkları tam olarak aynı olsa bile öğrenci rolü atanmamalıdır.

Yararlı bir RBAC yapılandırması elde etmek için kullanıcı izin atamalarından en uygun roller kümesini belirleme sorunu, Rol Madenciliği Problemi (RMP) olarak adlandırılmaktadır (Huang vd, 2015; Vaidya vd, 2010c; Vaidya vd, 2007). Temel olarak RMP, bir UPA matrisinden faydalanarak, minimum sayıda rol oluşacak şekilde UA ve PA matrislerini üretmeye odaklanmaktadır (Vaidya vd, 2009). Rol madenciliği aracılığıyla üretilen veya hali hazırda var olan RBAC konfigürasyonunun kalitesi, seçilen optimallik ölçüsü, mevcut bir RBAC konfigürasyonuyla benzerlik derecesi veya rol madenciliği aşamasında kullanılan UPA matrisini doğru şekilde tanımlama yeteneği açısından ifade edilebilir. Ayrıca rol madenciliğinde nicelik ve görev ayrımı gibi çeşitli kısıtlamalar da dikkate alınabilir (Sandhu vd, 1996). Seçilen belirli optimizasyon metriğine ve diğer hususlara bağlı olarak, bir dizi RMP varyantı ve bunları çözmek için çeşitli algoritmalar önerilmiştir. Fuchs ve Meier, bu algoritmaların birçoğunun sistematik bir sınıflandırmasını sağlamıştır (Fuchs ve Meier, 2011). Molloy ve arkadaşları, mevcut farklı gerçek dünya veri kümeleri üzerinde çeşitli yaklaşımların performans karşılaştırmasını sunmuşlardır (Molloy vd, 2009).

Daha genel amaçlı sistemler için otomatikleştirilmiş rol hiyerarşisi üretimi göz önüne alındığında, Schlegelmilch ve arkadaşları tarafından önerilen yaklaşım, birlikte gerçekleşen izinleri rol kümelerine ayırmakta ve küme çiftleri arasında maksimum eşleşmeyi kullanmaktadır. Birden fazla küme çifti maksimum örtüşmeye sahip olduğunda ve her izin, hiyerarşinin yalnızca bir rol yolu boyunca bulunabildiğinde bir küme çifti rastgele seçilmektedir (Schlegelmilch ve Steffens, 2005).

İzinlerin birden fazla role ait olmasına izin vermek amacıyla Vaidya ve arkadaşları, aday rolleri belirlemek için alt küme numaralandırmasını kullanarak RoleMiner isimli bir araç önermişlerdir (Vaidya vd, 2006). Yaklaşımda her bir kullanıcının izin seti, küme çiftleri arasında paylaşılan alt kümeler olarak kullanılmaktadır. Yaklaşım, daha fazla örtüşen alt küme bulunamayana kadar yeni tanımlanan kümelere yinelemeli olarak uygulanmaktadır. İlk yineleme, her kullanıcıya yalnızca bir veya iki rol atanırsa ve her kullanıcıya birden fazla rol atandığında düşük performans gösterirse, simüle edilmiş verilerdeki rollerin çoğunu belirleme yeteneğine sahiptir. Tam algoritma, kullanıcılara birden çok rol atandığında bile olası tüm rolleri ayıklayabilir ancak tam algoritmanın çalışma süresi üsteldir. Önerilen aday rollerin

hiyerarşik RBAC için nasıl uyarlanabileceğine dair bir yöntem yoktur ve tanımlanmış aday rollerin sayısı son derece fazladır.

Otomatik rol çıkarma için grafik tabanlı yaklaşımlar da önerilmiştir. Kullanıcıları, izinleri ve aralarındaki ilişkileri düğümler ve kenarlar olarak bir grafiğe yerleştiren yaklaşımlar bulunmaktadır (Ene vd, 2008; D. Zhang vd, 2007). Zhang ve arkadaşlarının buluşsal yöntemi, grafik yapısının karmaşıklığını azaltmak için kullanıcılar ve izinler arasına rol düğümleri eklemekte ve rolleri birleştirmektedir. Yaklaşımlarına göre grafiği optimize etmek, kuruluş için gereken yönetimin optimize edilmesini temsil etmektedir. Ene ve arkadaşları ise tüm izin atamalarını kapsamak için gereken minimum rol sayısını bulmak amacıyla grafik temsillerini azaltmak için buluşsal yöntemleri ve grafik teorisini kullanmaktadır (Ene vd, 2008).

RMP teorik analizi, izin atamalarını bozmadan asgari tanımlayıcı roller ve ilişkiler kümesini bulma sorununun NP-tam bir problem olduğunu göstermektedir (Lu vd, 2008; Vaidya vd, 2007). Mevcut yaklaşımlar, tüm kullanıcı izin atamalarını, izin atamalarını eklemeden veya kaldırmadan en son elde edilen aday rollerle analiz etmektedir. Bu koşullar altında rolleri ve rol ilişkilerini en aza indiren çözümler, uygulama için istenen sonuçları vermemektedir. Teoride tartışılan alternatif bir teknik, kullanıcı atamaları aşamasında başlangıçtaki sapmalara izin verme fikridir. Fakat henüz bu teoriyi test edecek bir uygulama bulunmamaktadır (D. Zhang vd, 2008).

Kümeleme

Kümeleme analizi, veri madenciliğinin en önemli araştırma alanlarından biridir (Zou, 2020). Benzer ögeleri bir arada gruplandırma amacına sahip, iyi bilinen ve denetimsiz öğrenmenin en önemli yaklaşımlarından biri olarak kabul edilen kümeleme veya küme analizi, veri yapısının bilinmeyen bir alanda gruplanması ile ilgilenmektedir (Xu Dongkuan ve Tian, 2015). Kümelemede nihai görev, ögeleri benzer oldukları ögelerle aynı kümeye, benzerliği düşük ögelerle ise farklı kümelere yerleştirmektir (Cochez ve Neri, 2015).

Kümeleme, ögelerin yerleştirilmesi gereken sınıfların önceden belirlendiği sınıflandırma işleminin tersi olarak kabul edilmektedir (Rokach, 2010). Sınıflandırmanın amacı tahmin etmek, kümelenin amacı ise tanımlamaktır (Vayssieres ve Plant, 1998). Kümeleme aracılığıyla oluşturulan kümeler, aynı kümedeki ögeler arasındaki benzerlikler yüksek, farklı kümelerdeki ögeler arasındaki benzerlikler nispeten düşük olacak şekilde meydana gelmektedir (Aidagulov vd, 2022). Kümeleme işleminde genellikle nesnelerin atanacağı kümelerin sayısı veya doğası hakkında önceden bir belirtim bulunmamaktadır.

Kümeleme genellikle yalnızca benzerlik ölçümlerine dayalı olarak yapılmaktadır ve ideal küme sayısı (genellikle geçici bir şekilde) genel kümeleme süreci içerisinde, ancak kümeleme algoritmasının kendisinden oldukça ayrı bir şekilde belirlenmektedir (Rokach, 2010). Bu özellikler küme analizini zorlaştırabilmektedir. Bu nedenle, bir dizi gözlemlenen veriye dayalı olarak ögelerin veya nesnelerin en iyi şekilde kümelenmesini üretebilmek için çok çeşitli algoritmalar önerilmiştir (Ferreira ve Hitchcock, 2009).

Yaygın olarak kullanılan birçok kümeleme algoritması, nesneler arasındaki ikili farklılıklara dayanmaktadır. Nesneler arasında benzerliği veya farklılığı tanımlamanın birçok farklı yolu bulunmaktadır. Benzerlik veya farklılık ölçüsünün seçimi büyük ölçüde kişinin üzerinde çalıştığı veri türüne bağlı olmaktadır (Rokach, 2010). Genellikle ögeler kümelendiğinde, herhangi iki öge arasındaki benzerlik veya benzemezlik ölçüsü, mesafe metrikleri ile gösterilmektedir (Ferreira ve Hitchcock, 2009).

Kümeleme analizi, modern veri analizinde merkezi bir rol oynamaktadır ve finans, pazarlama, genetik, tıp, psikoloji, arkeoloji, sosyal ve siyaset bilimi, kimya, mühendislik veya makine öğrenimi gibi verilerin ortaya çıktığı hemen hemen her alanda uygulanmaktadır. Kümeleme analizi, biyolojide türlerin sınırlandırılması gibi iyi tanımlanmış araştırma amaçlarına sahip olabilir veya bir şehrin semtlerini kümelerken bir veri setindeki yapı hakkında bilgi edinmek için oldukça keşifsel bir tarzda uygulanabilir (Hennig, 2022).

Literatür incelendiğinde birçok kümeleme yöntemine rastlanmaktadır ve küme kavramının tam olarak tanımlanmamış olması bunun temel nedeni olarak görülmektedir (Estivill-Castro Vladimir and Yang, 2000). Her biri farklı bir tümevarım ilkesi kullanan birçok kümeleme yöntemi geliştirilmiştir. Kümeleme yöntemlerinin, hiyerarşik ve bölümleme yöntemleri olmak üzere iki ana gruba ayrılması önerilmiştir (Fraley ve Raftery, 1998).

Hiyerarşik kümeleme yöntemleri 1960'lı yıllarda geliştirilmiştir ve örüntü tanıma, hesaplamalı biyoloji ve veri madenciliği gibi çok sayıda uygulamada kullanılmaktadır (Gronau ve Moran, 2007). Hiyerarşik kümeleme, denetimsiz makine öğreniminde önemli bir yere sahiptir ve kümelemenin önemli bir alt kategorisidir (Müllner, 2013). Biyolojideki yüksek verimli deneyler, dünya çapındaki ağın (world wide web, www) veri madenciliği gibi çalışmalarda hiyerarşik kümeleme yöntemlerinden yararlanılmaktadır (Filkov Vladimir ve Skiena, 2004; Larsen vd, 2002). Teorik bilgisayar bilimi, bilimsel simülasyon gibi alanlarda da hiyerarşik kümeleme yöntemleri kullanılmaktadır (Eppstein, 2001).

Hiyerarşik kümeleme yöntemleri, nesneleri yukarıdan aşağıya ve aşağıdan yukarıya yinelemeli olarak bölümleyerek kümeleri oluşturmaktadır (Rokach, 2010). Temelde toplayıcı hiyerarşik kümeleme (AHC) ve bölücü hiyerarşik kümeleme (Divisive Hierarchical

Clustering) olmak üzere iki önemli hiyerarşik kümeleme yöntemi bulunmaktadır (Jain ve Dubes, 1988):

- Bölücü hiyerarşik kümeleme, kümelenecek nesnelerin bir bütün yani tek bir küme olarak ele alındığı, bu kümenin iki parçaya veya alt kümelere ayrıldığı yaklaşımdır. Bu parçaların her biri, kendi kümesinin içine dahil olana kadar veya her bir nesnenin kendi kümesi belirleninceye kadar tekrar tekrar daha küçük alt kümelere bölünmektedir.
- Toplayıcı hiyerarşik kümelemede ise başlangıçta her nesne kendi kümesini temsil etmektedir ve ardından iki küme yeni bir küme oluşturmak için tekrar tekrar birleştirilmektedir. Bu işlem, yalnızca bir küme kaldığında veya istenen küme yapısı elde edildiğinde sona ermektedir.

Bu çalışmada AHC yönteminden yararlanıldığı için yalnızca AHC yöntemi detay içerecek şekilde açıklanmıştır.

Toplayıcı hiyerarşik kümeleme (AHC)

AHC (Cochez ve Mou, 2015; Gilpin vd, 2013; Kull ve Vilo, 2008), hiyerarşik kümelemenin aşağıdan yukarıya yaklaşımını temsil etmektedir ve her nesnenin kendi kümesini temsil etmesiyle başlamaktadır (Müllner, 2011). Yani başlangıçta her bir nesne kendi kümesini oluşturmaktadır ve dolayısıyla nesne sayısı kadar küme oluşmaktadır (Billard ve Diday, 2020). Daha sonra tüm veri kümesi tek bir küme hâline gelene kadar birbirine en çok benzeyen kümeler art arda birleştirilmektedir (Rokach, 2010).

AHC için yalın bir algoritmanın sözde kodu Tablo 4'te verilmiştir. Verilen algoritmada girdi olarak gösterilen *clusters* değişkeni, başlangıçta her bir kümenin tek bir kümeye konulduğu ve tüm kümelerin bulunduğu bir diziyi temsil etmektedir. Dolayısıyla başlangıçta eleman sayısı nesne sayısına eşit olmaktadır. Algoritmaya girdi olarak verilen *distance_metric* değişkeni ise benzerlik ölçümü için kullanılacak mesafe metriğini belirtmektedir. AHC yaklaşımında iki küme arasındaki benzerlik ölçüsü hesaplanırken mesafe metriklerinden yararlanılmaktadır. İki küme arasındaki mesafe veya benzerlik bu metriklere dayalı bir şekilde hesaplanmaktadır. AHC yaklaşımında, başlangıçta her bir nesne bir küme olarak ele alınmakta ve birbirine benzeyen kümeler her adımda birleştirilerek sonuçta tek bir küme elde edilene kadar işleme devam edilmektedir.

AHC yaklaşımında çıktı olarak kümeler ve dendrogramlar üretilmektedir (Mucha vd, 2005) ve üretilen dendrogramlar kümeleme hiyerarşisini görselleştirmektedir (Chen vd, 2009). Kümeleme yaklaşımlarında genel olarak oluşturulacak küme sayısı net olarak

bilinmediği için dendrogramlar aracılığıyla sezgisel kestirimler yapılmaktadır. Sezgisel olarak yapılan kestirim sonucunda kaç küme elde edileceği dendrogram aracılığıyla belirlenmekte ve böylece kümeleme işlemine, elde edilmesi beklenen küme sayısı girdi olarak verilebilmektedir.

Tablo 4. Saf AHC Algoritması

Algoritma: Toplayıcı Hiyerarşik Kümeleme (AHC)

```
    Girdiler: clusters, distance_metric // kullanılacak mesafe metriği
    Çıktılar: dendrogram
 1  while size(clusters) > 0 do
 2      minimum_distance ← ∞
 3      for index1 ← 0 to length(clusters) do
 4          for index2 ← index1 + 1 to length(clusters) do
 5              distance ← distance_metric(clusters[index1], clusters[index2])
 6              if distance < minimum_distance then // birleşecek iki kümenin tespit edilmesi
 7                  minimum_distance ← distance
 8                  left_cluster ← clusters[index1]
 9                  right_cluster ← clusters[index2]
11              end if
12          end for
13      end for
14      clusters ← clusters - [left_cluster, right_cluster] // birleşecek iki kümeyi çıkar
15      new_cluster ← merge(left_cluster, right_cluster) // iki kümeyi birleştir
16      dendrograma birleşecek iki küme için bağlantı ekle
17      clusters ← clusters + new_cluster
18  end while
```

AHC yaklaşımında hangi kümelerin birleştirilmesi gerektiğini belirlemek için çeşitli bağlantı yöntemleri kullanılmaktadır. Bu bağlantı yöntemleri en yakın iki kümenin birleştirilmesi aşamasında kullanılmaktadır ve nesne kümeleri arasındaki mesafeyi tanımlamaktadır (Müllner, 2011). Kümeler arasındaki benzerlikler hesaplandıktan sonra bu bağlantı yöntemlerinden faydalanılarak hangi iki kümenin birleştirileceği belirlenmektedir. Yaklaşımda Jaccard benzerlik ölçütü kullanıldığı için aşağıda Jaccard metriği ve Jaccard mesafesi açıklanmıştır. Ayrıca yaygın olarak kullanılan bazı bağlantı yöntemleri ve dendrogramlar anlatılmıştır.

Jaccard Metriği ve Mesafesi

Jaccard (Gower ve Warrens, 2014), iki küme arasındaki mesafeyi ölçmek için yaygın olarak kullanılan bir benzerlik metriğidir (Hwang ve Yang, 2014; Yan vd, 2019) ve ilk olarak 1908'de Paul Jaccard tarafından tanımlanmıştır (Jaccard, 1908; Yeh ve Yang, 2016). Jaccard metriği, bilgi alma, makine öğrenimi, veri madenciliği, tıbbi görüntüleme gibi birçok konu ve yaklaşımlarda pratik uygulamaları olan, kümelerdeki veya ikili verilerdeki benzerlikleri bulmak için kullanılabilecek bir kriterdir (Eelbode vd, 2020; Kosub, 2019).

Jaccard metriği, iki kümenin kesişiminin eleman sayısının, birleşiminin eleman sayısına oranıdır. Jaccard metriği 0 ile 1 arasında bir değer üretmektedir. Üretilen çıktının 1 değerine eşit olması durumu, iki küme veya benzerlik ölçümü yapılan nesnelerin birbirlerinin tam olarak aynısı olduğunu göstermektedir. Aynı şekilde, üretilen çıktının 0 değerine eşit olduğu durumda ise iki küme veya nesnenin birbirlerine hiç benzemediği sonucuna ulaşılmaktadır. Üretilen çıktı, 1 değerine ne kadar yakınsa nesneler birbirlerine o ölçüde benzemektedir. Jaccard metriği aşağıdaki denklemde gösterildiği gibi tanımlanmaktadır:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Yaklaşımda, dendrogramların oluşturulması aşamasında Jaccard mesafesi kullanılmıştır. Jaccard mesafesi, Jaccard metriğinin 1'e tümleyenidir ve Jaccard metriğinin tersi olacak şekilde yorumlanmaktadır. Yani, Jaccard mesafesinin ürettiği çıktı 0 değerine ne kadar yakınsa iki nesne birbirine o ölçüde benzemektedir. Çıktının 0 değerine eşit olması durumu iki nesnenin birbirinin tam olarak aynısı olduğunu göstermektedir. Aynı şekilde, üretilen çıktı 1 değerine ne kadar yakınsa iki nesne birbirine o ölçüde benzememektedir. Jaccard mesafesi aşağıdaki denklemde gösterildiği gibi tanımlanmaktadır:

$$J_d(A,B) = 1 - \frac{|A \cap B|}{|A \cup B|}$$

İkili verilerden oluşan iki küme için Jaccard metriği, iki kümede aynı sırada bulunan ve iki kümede de 1 değerine sahip olan nesnelerin sayısının (kesişim), iki kümeden en az birinde 1 değerine sahip olan nesnelerin sayısına (birleşim) oranı ile hesaplanmaktadır.

Tablo 5. İkili Verilerden Oluşan Örnek Bir Küme

	$Nesne_1$	$Nesne_2$	$Nesne_3$	$Nesne_4$	$Nesne_5$
$Küme_1$	0	1	0	1	0
$Küme_2$	0	0	1	1	0

Tablo 6. Jaccard Metriği ve Jaccard Mesafesi Algoritmasının Sözde Kodu

Algoritma: Jaccard Metriği ve Jaccard Mesafesi

Girdiler: cluster1, cluster2 // mesafesi hesaplanacak aynı uzunluktaki iki küme
Çıktılar: jaccard_metric, jaccard_distance

```
length_of_clusters ← length(cluster1)
X ← 0
Y ← 0
Z ← 0
for index ← 0 to length_of_clusters do
    if cluster1[index] is 1 and cluster2[index] is 1 then
        X ← X + 1
    end if
    if cluster1[index] is 1 and cluster2[index] is 0 then
        Y ← Y + 1
    end if
    if cluster1[index] is 0 and cluster2[index] is 1 then
        Z ← Z + 1
    end if
end for
jaccard_metric ← X / (X + Y + Z)
jaccard_distance ← 1 – jaccard_metric
```

Yukarıda örnek olarak ikili verilerden oluşan bir tablo verilmiştir (Tablo 5). Verilen tablodaki iki küme için aşağıdaki eşitlikler verilmiş ve daha sonra Jaccard mesafesi hesaplanmıştır.

- X = İki kümede de 1'e eşit olan nesnelerin sayısı
- Y = Birinci kümede 1'e, ikinci kümede 0'a eşit olan nesnelerin sayısı
- Z = Birinci kümede 0'a, ikinci kümede 1'e eşit olan nesnelerin sayısı

Yukarıdaki eşitliklere göre ikili verilere sahip A ve B kümeleri için Jaccard metriği aşağıdaki denklemde gösterildiği gibi hesaplanmaktadır:

$$J(A,B) = \frac{X}{X+Y+Z}$$

Buna göre, Tablo 5'te verilen iki küme arasındaki Jaccard metriği ve mesafesi aşağıdaki gibi olacaktır:

$$J(Küme_1, Küme_2) = \frac{X}{X+Y+Z} = \frac{1}{3}$$

$$J_d(Küme_1, Küme_2) = 1 - \frac{1}{3} = \frac{2}{3}$$

İki küme arasındaki Jaccard metriği ve Jaccard mesafesini hesaplayan algoritmanın sözde kodu Tablo 6'da gösterilmiştir.

Tek (Single) Bağlantı Yöntemi

Tek bağlantı yöntemi (Jarman, 2020; Sneath, 1957), iki kümedeki iki nesne arasındaki en kısa mesafeye dayalı olarak kümeleri birleştirmektedir (Şekil 4). Tek bağlantı yöntemine göre; *d(i, j)* i numaralı ve j numaralı nesneler arasındaki mesafeyi temsil etmek üzere A ve B kümeleri arasındaki mesafe aşağıdaki şekilde tanımlanmaktadır:

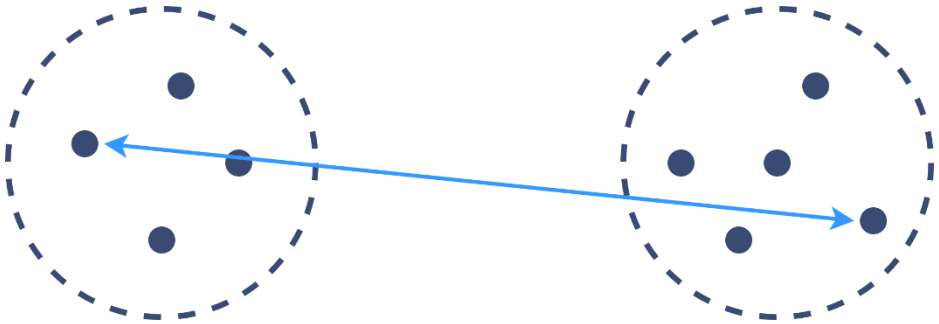
$$d(A,B) = \min_{i \in A, j \in B} d(i,j)$$

Şekil 4. Tek bağlantı yöntemi

Tam (Complete) Bağlantı Yöntemi

Tam bağlantı yöntemi (McQuitty, 1960; Sokal vd, 1963), kümeleri iki kümedeki iki nesne arasındaki maksimum mesafeye dayalı olarak birleştirmektedir (Şekil 5). Tam bağlantı yöntemine göre; *d(i, j)* i numaralı ve j numaralı nesneler arasındaki mesafeyi temsil etmek üzere A ve B kümeleri arasındaki mesafe aşağıdaki şekilde tanımlanmaktadır:

$$d(A,B) = \max_{i \in A, j \in B} d(i,j)$$

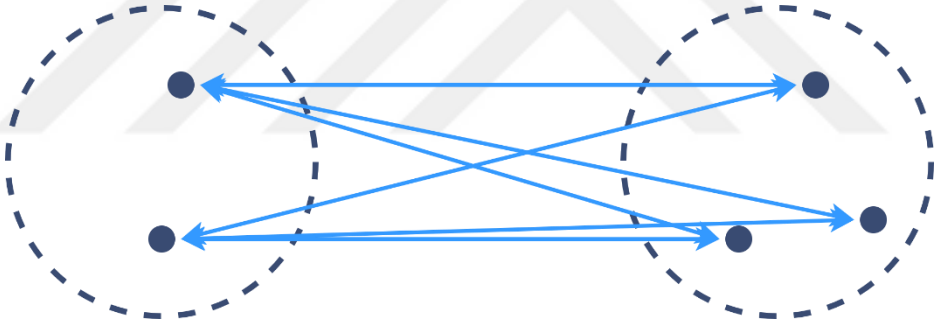


Şekil 5. Tam bağlantı yöntemi

Ortalama (Average) Bağlantı Yöntemi

Ortalama bağlantı yöntemi (Bu vd, 2020; Sokal, 1958), bir kümedeki tüm nesnelerin diğer kümedeki tüm nesnelere olan ortalama mesafesine göre kümeleri birleştirmektedir (Şekil 6). Ortalama bağlantı yöntemine göre; *d(i, j)* i numaralı ve j numaralı nesneler arasındaki mesafeyi ve |A| ifadesi A kümesindeki nesnelerin sayısını temsil etmek üzere, A ve B kümeleri arasındaki mesafe aşağıdaki şekilde tanımlanmaktadır:

$$d(A,B) = \frac{1}{|A||B|} \sum_{i \in A, j \in B} d(i,j)$$



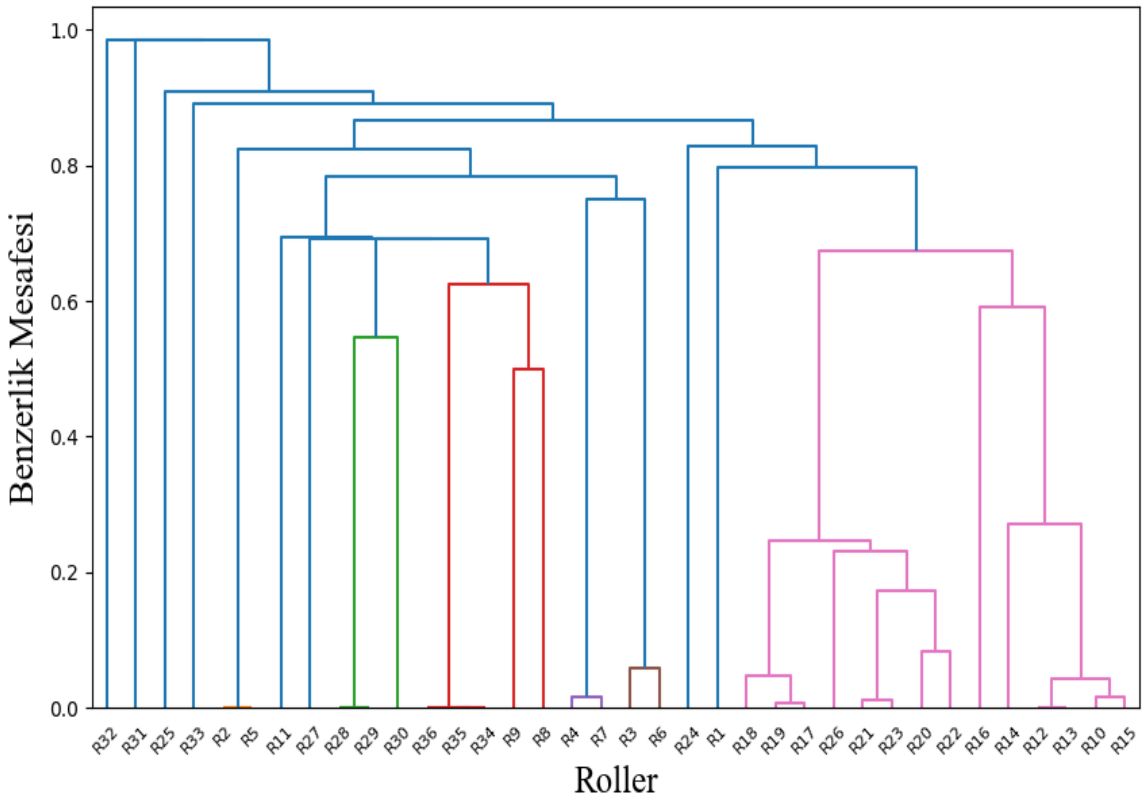
Şekil 6. Ortalama bağlantı yöntemi

Dendrogram

Hiyerarşik kümeleme algoritmaları sadece nesnelerin bir bölümünü veya kümelerini değil, ayrıca bir dendrogram da üretmektedir (Şekil 7). Üretilen dendrogram kümeleme işleminin sonucunun eşdeğeridir (Zheng vd, 2014). Dendrogram yaklaşımında, nesneler veya kümeler arasındaki bir miktar mesafe ölçüsünden ziyade aynı veya benzer nesne kümelerinin tekrarı inceleme altında olduğundan, dendrogramların topolojisi veya dallanma modeli, kümelerin sınıflandırılmalarını karşılaştırmanın doğal bir yolunu sağlamaktadır (Phipps, 1976). Bir dendrogram, kümelenecek nesnelerin birer birer gösteriminden, kümeleme sonucunda elde edilecek kümelerde bulunacak nesne sayısına kadar okunurluğu yüksek bir çıktı (ağaç) üretmektedir (Şekil 7) (Moore vd, 1973). Yani bir dendrogram, tipik olarak,

verilerdeki hiyerarşik ilişkileri (örneğin hiyerarşik kümeleme sonuçları) görselleştirmek için kullanılan bir ikili ağaç biçimidir (Chen vd, 2009).

Bir dendrogram, alt dendrogramlardan oluşan bir ağaç olarak görülebilir (Sieger vd, 2017). Böyle bir ağaçta, bir düğümün çocukları onun varisi olarak ele alınabilir (Kikugawa vd, 2010). Böylece, yapraklar dendrogramın minimum elemanlarına karşılık gelirken kök, maksimum elemanına karşılık gelmektedir (Tadonki vd, 2014).



Şekil 7. Dendrogram örneği

Oluşturulacak küme sayısı, dendrogramın belirli bir yükseklikte (dendrogram oluşturulurken kullanılan mesafe metriğinin belirli bir ölçüsünde) sezgisel olarak kesilmesiyle elde edilmektedir. Bu işlem, elde edilecek küme sayısının önceden bilinmediği uygulamalarda son derece kullanışlı olmaktadır. Bazı kümeleme yaklaşımları, bir buluşsal yöntem kullanarak küme sayısını otomatik olarak belirlemektedir ancak kullanıcılar bunların güvenilirliği konusunda tereddüt yaşayabilir veya uygun bir yöntemi bulmak zor olabilir. Ayrıca hiyerarşik yaklaşımlar, kullanıcının üretilen hiyerarşik çıktıda gezinmesine izin vermektedir ve bu da kullanıcılara verilerin yapısı hakkında daha derin bir fikir edinme olanağı sağlamaktadır.

MATERYAL VE YÖNTEM

Literatür kapsamında geleneksel erişim kontrol sistemleri, rol tabanlı erişim kontrol sistemleri, rol madenciliği ve rol mühendisliği, kümeleme ve AHC kavramları incelendikten sonra bu bölümde, hali hazırda sektörde faaliyet gösteren bir yazılım şirketinin sunduğu veri seti ve uygulanan yöntem anlatılmıştır. Bu başlık 2 bölümden oluşmaktadır. Birinci bölümde, yazılım şirketinin sunduğu ve erişim kontrol sistemlerinde hali hazırda kullandıkları veri seti maskelenmiş bir şekilde gösterilmiştir. İkinci bölümde ise uygulanan yöntem detaylarıyla anlatılarak kullanılan algoritmalar açıklanmıştır.

Veri Seti

Uygulama geliştirilirken kullanılan veri seti; müşterilerinin satış, lojistik ve servis konularında dijitalleşmesini sağlayan yazılımlar üreten, müşterilerinin ihtiyaçları doğrultusunda projeler için gerekli olabilecek yazılım, donanım ve profesyonel hizmetler konularında hizmet sunan, 25000'den fazla mobil kullanıcıya sahip, 130'dan fazla hayata geçirilmiş ve canlılığını devam ettiren projesi bulunan, 160'dan fazla çalışanı ve 40'tan fazla Ar-Ge çalışanı olan ve bir yazılım şirketi olan Univera tarafından sağlanmıştır.

Tablo 7. Sunulan Müşteri Tablolarının Maskelenmiş Bir Örneği

Şirket Kodu	Ürün No	Grup Kodu	C*	R*	U*	D*	Menü	URL	Rol
1	1	1	1	1	1	0	A Menüsü	X	A
1	1	2	0	1	0	0	B Menüsü	Y	A
1	2	3	1	1	1	1	B Menüsü	Y	B
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.

C: Create (Oluşturma) izni
R: Read (Okuma) izni
U: Update (Güncelleme) izni
D: Delete (Silme) izni

Yazılım şirketi, farklı müşterileri için geliştirmiş ve yayınlamış olduğu uygulamaların her biri için ayrı bir erişim kontrol sistemi oluşturmuş ve her bir müşterinin kendi

uygulamasının güvenliğini sağlayabilmesi amacıyla her müşteriye kendi erişim kontrol sistemini yönetme ayrıcalığı tanımıştır. Her müşteri, kendi erişim kontrol sistemi için roller oluşturabilmekte, ilgili roller için yetkiler belirleyebilmekte, var olan yetkileri sınırlayabilmekte ve şirket çalışanlarına gerekli rolleri atayarak çalışanlarını belirli modüller veya web sayfaları üzerinde yetkin kılabilmektedir.

Paylaşılan her bir veri tablosu, müşterilerin RBAC tabanlı erişim kontrol sistemlerinin bilgilerini içermektedir. Tabloların her satırında menü numarası, şirket kodu, ürün numarası, grup kodu, oluştur (create), oku (read), güncelle (update), sil (delete) yetkileri (yani CRUD), menü açıklaması, URL ve rol adı bilgileri bulunmaktadır. URL sütunu, ilgili rolün sahip olduğu yetkilerin hangi kaynakla veya web sayfası ile ilgili olduğunu belirten bir köprü veya hedef belirteci gibidir. Menü sütunu, URL'lerin hangi menüyü hedeflediği hakkında bir açıklama içermektedir. Menü numarası, menü sütunlarına karşılık gelen eşsiz değerlerden oluşmaktadır. CRUD sütunları ise rollerin ilişkili oldukları URL'ler üzerindeki yetkilerini belirlemektedir. Müşterilerden birinin RBAC tablosu, anonimliği sağlayabilmek ve bu sayede müşterilerin sistemlerinin güvenliğini tehlikeye atmamak amacıyla maskelenmiş bir şekilde gösterilmiştir (Tablo 7).

Tablo 8. Sunulan Müşteri Tablolarının Özeti

Tablo Adı	Satır Sayısı	URL Sayısı	Rol Sayısı	Sektör
Müşteri 1	828	299	28	Otomotiv
Müşteri 2	2485	348	36	Otomotiv
Müşteri 3	2432	344	31	Otomotiv
Müşteri 4	5160	295	68	Yiyecek
Müşteri 5	151	59	14	Yiyecek
Müşteri 6	963	278	44	Yiyecek
Müşteri 7	197	42	28	Yiyecek
Müşteri 8	196	60	14	Yiyecek
Müşteri 9	8112	743	94	Yiyecek
Müşteri 10	227	41	19	Yiyecek

Roller, müşteriler tarafından kendi inisiyatifleriyle eklenebildiğinden veya çıkarılabildiğinden dolayı erişim kontrol sistemlerinin bakımı ve kontrolü son yıllarda giderek zorlaşmıştır. Yazılım şirketi ile yapılan görüşmeler doğrultusunda hali hazırda kullanılan erişim kontrol sistemlerinde daha kolay yönetilebilecek bir yapıya ulaşılıp ulaşılamayacağı hakkında tartışılmıştır. Yazılım şirketi, deneysel çalışmanın yürütülebilmesi adına müşteriler tarafından aktif olarak kullanılan bir yazılım ürününün 10 müşteriye ait erişim kontrol sistemi

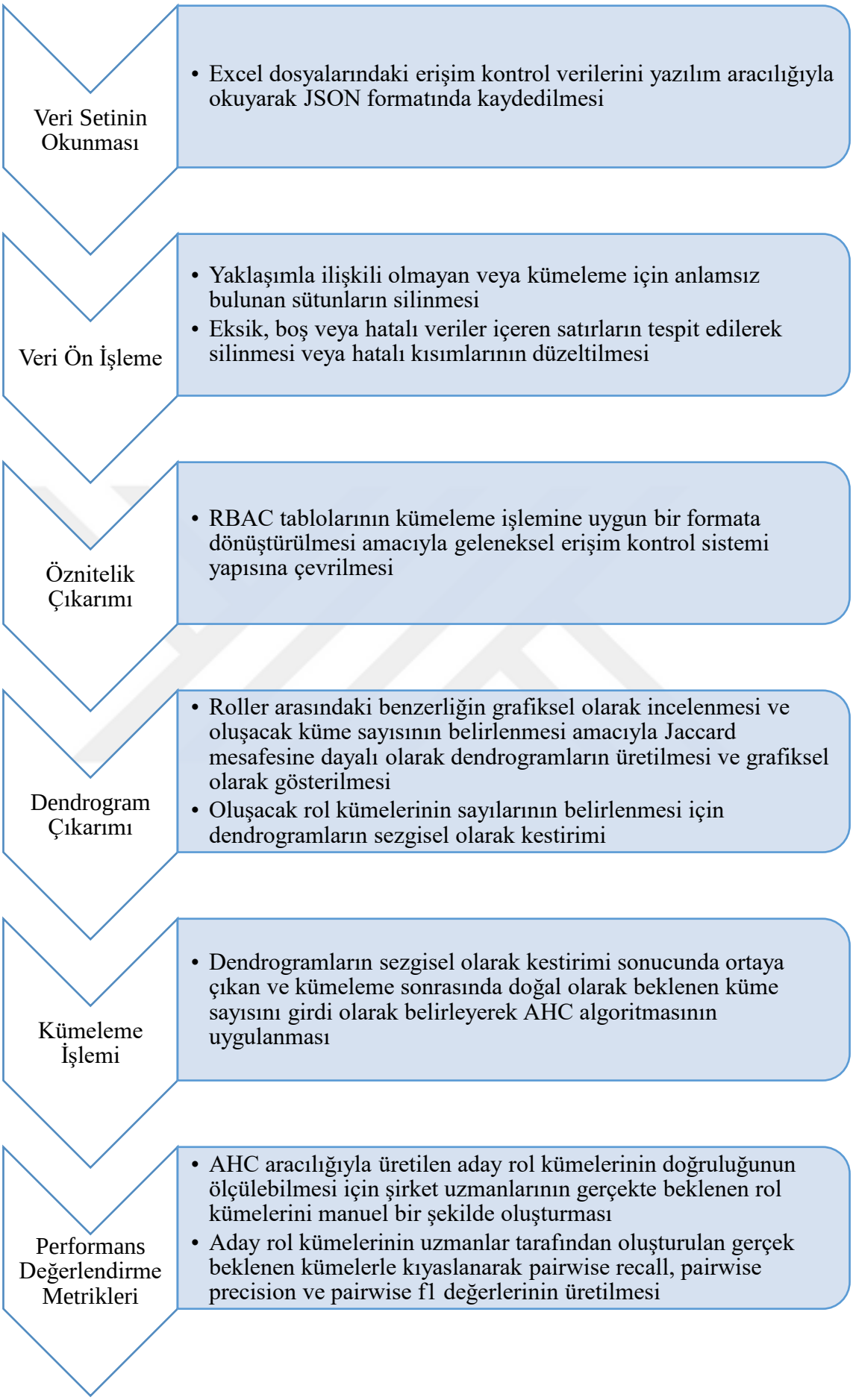
verilerini Excel tabloları formatında paylaşmıştır. Yazılımdaki her müşteri için, sistem yöneticileri tarafından müşteriye özel yeni bir RBAC tablosu oluşturulmuştur. Yazılım şirketinin 10 müşterisinin RBAC verileri, şirketin ve müşterilerinin güvenliğini ve gizliliğini sağlamak amacıyla maskelenmiş bir şekilde özet olarak gösterilmiştir. Özet olarak gösterilen tablonun her satırında sırasıyla, sunulan tablonun adı, tabloda bulunan satır sayısı, tabloda bulunan toplam URL sayısı, tabloda bulunan toplam rol sayısı ve tablosu sunulmuş müşterinin faaliyet gösterdiği sektör bilgisi bulunmaktadır (Tablo 8). Sektör bilgisi, şirket tarafından sunulan erişim kontrol verilerinde hali hazırda bulunmamaktadır. Şirket uzmanları ile yapılan görüşmeler doğrultusunda erişim kontrol verilerinin ait olduğu müşterilerin sektör bilgileri belirlenmiş ve tabloda gösterilmiştir.

Yöntem

Bu başlık 6 bölümden oluşmaktadır. Sırasıyla, ilk bölümde veri setinin okunması, ikinci bölümde veri ön işleme, üçüncü bölümde öznitelik çıkarımı, dördüncü bölümde dendrogram çıkarımı, beşinci bölümde kümeleme işlemi, son bölümde ise yaklaşımın performansının nasıl ölçüldüğü anlatılmış ve kullanılan algoritmalar detaylı bir şekilde açıklanmıştır.

Uygulanan yaklaşımda, bir yazılım firmasının sunduğu RBAC tabanlı erişim kontrol sistemlerinin iyileştirilmesi ve sistemle uyumlu rol kümelerine ulaşılması amacıyla, sistemde zaten var olan roller ve rollere önceden tanımlanmış izinlere dayalı olarak AHC algoritması aracılığıyla kümeleme çalışması yapılmıştır. Böylece sistemin yeniden yapılandırılmasında veya yeni oluşturulacak sistemlerde referans olarak kullanılmak üzere uygun rol kümelerine ulaşılması amaçlanmaktadır.

AHC, veri setindeki her bir ögeyi tek bir küme olarak ele alan ve bu kümeleri sırayla birleştirerek sonunda tek bir küme elde eden, hiyerarşik kümeleme algoritmalarından aşağıdan yukarıya yaklaşımı temsil eden (Jafarzadegan vd, 2022), en geleneksel ve popüler kümeleme algoritmalarından biri olan (N. Liu vd, 2021) danışıksız (denetimsiz) bir algoritmadır. Uygulanan yöntem, daha önce belirlenmiş olan rolleri, ayrıcalık benzerliklerine dayalı bir şekilde kümeleyerek daha az sayıda roller üretme problemiyle ilgilendiği için aşağıdan yukarıya bir yaklaşım olarak değerlendirilmiştir. Ayrıca hedef roller belirlenmemiş olduğundan dolayı sınıflandırma problemi olarak ele alınmamıştır. Tüm bunlar göz önünde bulundurulduğunda, denetimsiz ve aşağıdan yukarıya yaklaşımla eşleştirilen bir öğrenme yöntemi olduğu için çalışmada AHC algoritması tercih edilmiştir. Uygulanan yaklaşımın akış şeması Şekil 8'de gösterilmektedir.



Şekil 8. Kullanılan yaklaşımın akış şeması

Yazılım firmasının sunmuş olduğu RBAC tabanlı veri setinin ve verilerin ne anlama geldiklerinin açıkça anlaşılabilmesi ve çalışmaya doğru bir şekilde uyarlanabilmesi amacıyla, şirketin Ar-Ge birimiyle günde 2 saat olmak üzere 10 hafta boyunca haftada 1 gün görüşme yapılmıştır. Şirketin verilerini paylaştığı yazılımın kullanıcıları, yazılımı lisanslı bir şekilde kullanan şirketlerdir. RBAC tabanlı oluşturulmuş 10 tablo, müşteri olan şirketler tarafından kullanılan rol izin atamalarını barındıran tablolardır. Tabloların sütunlarında bulunan bilgiler veri seti başlığında açıklanmıştır.

Tabloların satırlarında bulunan rollerin izin yetkilerini tam olarak belirleyen sütunlar URL ve CRUD sütunlarıdır. Ancak URL sütunu, bir rolün izinlerinin tanımındaki en belirgin sütundur. Çünkü URL sütunu olmadığında rolün sadece CRUD ayrıcalıkları kalırken bu izinlerin hangi web sayfasını veya API noktasını hedeflediği bilgisi bulunmamaktadır. Yalnızca URL sütunu olduğunda ise rolün URL üzerindeki yetkilerinin belirlendiği bilgilere ulaşılamasa bile en azından rolün hedefi netleşmektedir ve yalnızca o hedef üzerindeki davranışını sınırlayan bilgiler ortadan kaybolmaktadır. Ancak kümeleme aşamasında CRUD ayrıcalıkları dikkate alınmadan sadece URL sütunu kullanıldığında, tamamen farklı işlem ayrıcalıklarıyla kullanılan roller aynı URL hedefini işaret ettikleri için birbirlerine tam olarak benzeyen roller olarak görüneceklerdir. Ayrıca aynı rol, şirket tarafından sunulan tablo yapılarında farklı URL hedeflerine farklı ayrıcalıklarla erişebildiğinden dolayı aynı rol aynı tabloda birden fazla satırda farklı ayrıcalıklarla yer alabilmektedir. Bunlar göz önünde bulundurularak şirket tarafından sunulan RBAC tablolarının yaygın kullanımdan uzak bir şekilde oluşturulduğu, veri ön işleme ve öznitelik çıkarımı yapılmadan kullanılmalarının yaklaşımı uygulama aşamasında zorluklara neden olacağı belirlenmiştir.

Veri setinin okunması

Yazılım şirketi tarafından RBAC verileri, Excel tablosu formatında sunulmuştur. Excel tablolarını okumak için Python tabanlı xlrd kütüphanesi kullanılmıştır. Tablolar okunurken şirket uzmanlarının da görüşü doğrultusunda gereksiz ve doğruluk açısından performans kaybı oluşturabileceği tespit edilen satır ve sütunlar silinerek veri seti Json sözlük tipine dönüştürülmüştür. Veri seti okunurken uygulanan algoritmanın sözde kodu Tablo 9'da gösterilmiştir.

Tablo 9. Veri Seti Okunurken Uygulanan Algoritmanın Sözde Kodu

Algoritma: Müşteri Tablolarını Oku

Girdiler: file_path, sheet_name, roles_dict (boş sözlük)

Çıktılar: readed_rows // okunan tablonun satırları (dizi)

```
wb ← xlrd.open_workbook(file_path) // dosya yolu verilen workbook'u aç
sh ← wb.sheet_by_name(sheet_name) // workbook içinde belirtilen sayfayı al
rows ← sh.rows // sayfanın satırlarını rows değişkenine ata
readed_rows ← boş dizi
for row_index ← 1 to length(rows) do // ilk satır başlıkları içerdiği için atla
    row ← sh.row_values(row_index) / sayfadan sıradaki satırı al
    readed_rows ← readed_rows + row
end for
```

Veri ön işleme

Ön işlemenin ilk aşamasında URL, rol adı ve CRUD ayrıcalıklarını içermeyen sütunlar, problemin çözümüne yönelik anlamlı ve faydalı bilgiler içermediğinden dolayı silinmiştir. Ayrıca şirket uzmanlarının yönlendirmeleri doğrultusunda, kümeleme algoritmasının performansını ve değerlendirmesini olumsuz etkilememek amacıyla, algoritma için önem arz eden sütunlarda hatalı veri bulunduran ve artık kullanımda olmayan eski web sayfalarına ait URL'leri içeren satırlar da tespit edilerek silinmiştir.

Tablo 10. Veri Ön İşleme Sonucunda Elde Edilen Tabloların Yapısı

C	R	U	D	URL	Rol
1	1	1	1	URL_1	Rol_1
1	1	1	0	URL_2	Rol_2
0	1	0	0	URL_2	Rol_1
0	1	0	1	URL_1	Rol_3
1	1	1	0	URL_3	Rol_1
1	1	0	0	URL_2	Rol_4
.	.	.	.	.	.
.	.	.	.	.	.

Yalnızca kümeleme algoritması için önemli ve kullanılabilir verileri içerecek şekilde işlenen veri seti, benzersiz değerlerle çalışılması, okunması, değiştirilmesi, oluşturulması ve ayrıştırılması kolay bir yapı olduğu için JSON formatına dönüştürülmüştür (Lin vd, 2012). Nihai olarak elde edilen veri kümesinde, yalnızca API noktalarını işaret eden URL'lere bağlı

ayrıcalıklar, roller ve her rol için hedeflerini doğal olarak belirleyen URL'ler mevcuttur. Veri ön işleme sonucunda elde edilen tablolar Tablo 10'daki gibi görünmektedir. Veri ön işleme sırasında uygulanan algoritmanın sözde kodu ise Tablo 11'de gösterildiği gibidir.

Tablo 11. Veri Ön İşleme Adımının Sözde Kodu

Algoritma: Eksik veya Hatalı Satır ve Sütunları Tespit Et ve Sil

Girdiler: readed_rows // okunan tablonun satırları (dizi)
Çıktılar: URL_count

```
URLs ← yeni boş sözlük // değişken ataması
URL_count ← 0
URL_column ← URL sütununun indeksini ata
role_column ← rol sütununun indeksini ata
C_column ← oluşturma yetkisi bilgisini içeren sütunun indeksini ata
R_column ← okuma yetkisi bilgisini içeren sütunun indeksini ata
U_column ← güncelleme yetkisi bilgisini içeren sütunun indeksini ata
D_column ← silme yetkisi bilgisini içeren sütunun indeksini ata
for row_index ← 1 to length(readed_rows) do // ilk satır başlıkları içerdiği için atla
    row ← readed_rows[row_index] // sayfadan sıradaki satırı al
    role ← row[role_column]
    URL ← row[URL_column]
    if role is NULL or URL is NULL then
        continue // atla ve sıradaki satıra geç
    end if
    if URL includes dirty_data then
        fix the URL or continue // URL'i düzelt veya satırı atla
    end if
    C ← row[C_column]
    R ← row[R_column]
    U ← row[U_column]
    D ← row[D_column]
    URL_with_CRUD ← [C, R, U, D] // dizi ataması
    if URL not in URLs then
        URL_count ← URL_count + 1
        order_of_URL ← URL_count
        URLs[URL] ← order_of_URL
    end if
```

Tablo 12. Veri Ön İşleme Adımının Sözde Kodu (Devam)

```
29      URL_with_CRUD ← URL_with_CRUD + URLs[URL]
30      if role not in roles_dict then
31          roles_dict[role] ← boş dizi
32      end if
33      if URL_with_CRUD not in roles_dict[role] then
34          roles_dict[role] ← roles_dict[role] + URL_with_CRUD
35      end if
36  end for
```

Öznitelik çıkarımı

Firma uzmanları ile yapılan görüşmeler doğrultusunda gereksiz olduğu tespit edilen satır ve sütunlardaki veriler silindikten sonra öznitelik çıkarımı (feature extraction) yapılmıştır. Öznitelik çıkarımı aşamasında her bir müşteri için sunulan tablolar, kümeleme işlemini gerçekleştirebilecek bir yapıda olmadığı için geleneksel geçiş kontrol sistemlerine benzer bir biçime dönüştürülerek kümeleme işlemini uygulamaya uygun bir hale getirilmiştir.

Tablo 13. Öznitelik Çıkarımı Sonucunda Elde Edilen Tabloların Genel Yapısı

Rol	**URL_1**				**URL_2**				**URL_3**				**…**	**URL_n**			
	C	**R**	**U**	**D**	**C**	**R**	**U**	**D**	**C**	**R**	**U**	**D**	**…**	**C**	**R**	**U**	**D**
Rol_1	1	1	1	1	0	1	0	0	1	1	1	0	…	1	1	1	1
Rol_2	0	0	0	0	1	1	1	0	0	0	0	0	…	0	0	0	0
Rol_3	0	1	0	1	0	0	0	0	0	0	0	0	…	0	1	0	0
Rol_4	0	0	0	0	1	1	0	0	0	0	0	0	…	0	0	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.	…	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	…	.	.	.	.
Rol_m	0	1	0	0	0	1	1	1	1	1	0	0	…	0	1	0	1

Tablolar, öznitelik çıkarımı işlemi yapılmadan önce, yani Tablo yapısını geleneksel erişim kontrol sistemi yapısına benzer bir forma dönüştürmeden önce her rol farklı sayıda satırlarda bulunabiliyordu. Dönüştürme işleminden sonra, her rol yalnızca bir satırda bulunabilir hale getirilmiş ve her rol için toplamda URL sayısı çarpı 4 (yani CRUD sütunları) kadar sütun oluşturulmuştur. Yani n tane URL için n x 4 kadar sütun oluşturulmuştur.

Tablo 14. Öznitelik Çıkarımı Aşamasının Sözde Kodu

Algoritma: Tabloyu Kümelemeye Uygun Formata Dönüştür

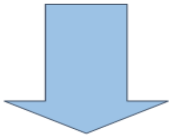
Girdiler: roles_length, URL_count, roles_dict

Çıktılar: roles_list_with_all_URLs

```
CRUD_length ← 4 // yetki sayısı
column_count ← URL_count * CRUD_length
roles_list_with_all_URLs[roles_length][column_count] ← 0 // 0'larla doldur
roles_list ← roles_dict to list
roles_length ← length(roles_list)
role_name_index ← 0
URL_with_CRUD_index ← 1
role_names ← boş dizi
for index ← 0 to roles_length do
    role_name ← roles_list[index][role_name_index]
    role_names ← role_names + role_name
    URLs_with_CRUD ← roles_list[index][URL_with_CRUD_index]
    for inner_index ← 0 to length(URLs_with_CRUD) do
        C ← URLs_with_CRUD[inner_index][0]
        R ← URLs_with_CRUD[inner_index][1]
        U ← URLs_with_CRUD[inner_index][2]
        D ← URLs_with_CRUD[inner_index][3]
        URL_order ← URLs_with_CRUD[inner_index][4]
        URL_index ← (URL_order - 1) * CRUD_length
        C_index ← URL_index + 0
        R_index ← URL_index + 1
        U_index ← URL_index + 2
        D_index ← URL_index + 3
        roles_list_with_all_URLs[index][C_index] = C
        roles_list_with_all_URLs[index][R_index] = R
        roles_list_with_all_URLs[index][U_index] = U
        roles_list_with_all_URLs[index][D_index] = D
    end for
end for
```

Öznitelik çıkarımı işlemi yapıldıktan sonra elde edilen tablolar, Tablo 13'te gösterildiği gibi bir yapıda bulunmaktadır. İlgili tablo, örnek olması açısından maskelenmiş bir şekilde paylaşılmış ve müşterilerin güvenliği de göz önünde bulundurularak izinler tamamen rastgele oluşturulmuştur.

C	R	U	D	URL	Rol
1	1	1	1	URL_1	Rol_1
1	1	1	0	URL_2	Rol_2
0	1	0	0	URL_2	Rol_1
0	1	0	1	URL_1	Rol_3
1	1	1	0	URL_3	Rol_1
1	1	0	0	URL_2	Rol_4
.	.	.	.	.	.
.	.	.	.	.	.



Rol	URL_1				URL_2				URL_3				...	URL_n			
	C	R	U	D	C	R	U	D	C	R	U	D	...	C	R	U	D
Rol_1	1	1	1	1	0	1	0	0	1	1	1	0	...	1	1	1	1
Rol_2	0	0	0	0	1	1	1	0	0	0	0	0	...	0	0	0	0
Rol_3	0	1	0	1	0	0	0	0	0	0	0	0	...	0	1	0	0
Rol_4	0	0	0	0	1	1	0	0	0	0	0	0	...	0	0	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.	...	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	...	.	.	.	.
Rol_m	0	1	0	0	0	1	1	1	1	1	0	0	...	0	1	0	1

Şekil 9. RBAC tablolarının geleneksel tablo yapısına dönüştürülmesi

Öznitelik çıkarımı sırasında tablo verilerinin geleneksel erişim kontrol sistemlerine benzetilme aşamasının grafiksel temsili Şekil 9'da ve uygulanan algoritmanın sözde kodu Tablo 14'de gösterildiği gibidir.

Dendrogram çıkarımı

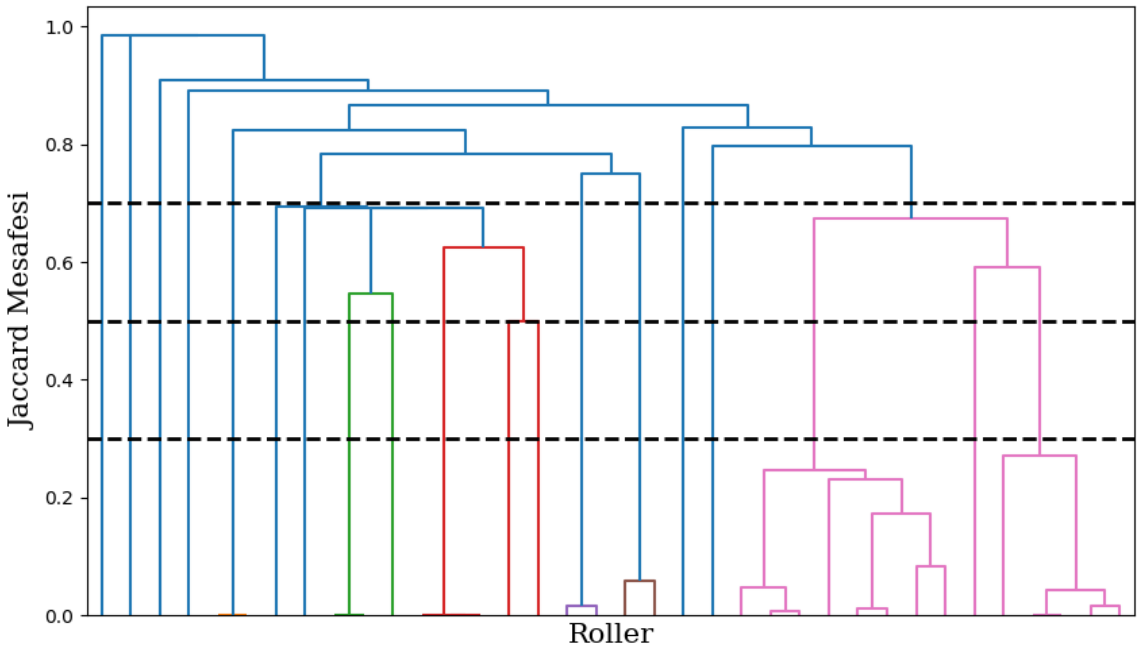
Tabloları içeren dosyalar okunduktan sonra Python tabanlı SciPy kütüphanesi aracılığıyla, elde edilen satır verilerinden Jaccard mesafeleri hesaplanarak dendrogram çıkarımı yapılmıştır. Uygulaması basit ve hızlı sonuçlar veren bir mesafe metriği olduğu için Jaccard indeksi veya mesafesi avantajlı bir metriktir (Aziz vd, 2010). Yaklaşımda roller arasındaki benzerlikleri bulmak için Jaccard mesafesi ve kümeleri birleştirme aşamasında referans olması için tek bağlantı yöntemi kullanılmıştır. Benzerlik mesafesinin hesaplanması aşamasında roller arasındaki benzerlik dikkate alınmıştır. Her bir rolle ilişkilendirilmiş olan ve rollerin hedefini belirten URL'lerin CRUD ayrıcalıkları, rollerin benzerlik kriterleri olarak

seçilmiştir. Dendrogram çıktısı, rollerin benzerliklerinin bir ağaç grafik temsilidir. Dendrogram, Python tabanlı SciPy kütüphanesi kullanılarak elde edilmiş ve matplotlib (Harding, 2016) kütüphanesi aracılığıyla görselleştirilmiştir. Dendrogram çıkarımı aşamasında uygulanan algoritmanın sözde kodu Tablo 15'te verilmiştir.

Tablo 15. Dendrogram Çıkarımı Aşamasının Sözde Kodu

Algoritma: Dendrogramı Göster, Uygulanan Eşik Değerinde Oluşacak Küme Sayısını Ver	
	Girdiler: roles_list_with_all_URLs, role_names, threshold
	Çıktılar: dendrogram, cluster_count
1	roles ← roles_list_with_all_URLS
2	similarity_measure ← "jaccard"
3	linkage_method ← "single"
4	linkage_result ← linkage_with_scipy**(**roles, linkage_method, similarity_measure**)**
5	made_dendrogram_with_scipy**(**linkage_result, labels ← role_names**)**
6	show_dendrogram_with_matplotlib**()**
7	cluster_count ← get_number_of_clusters_with_linkage_result**(**threshold**)**

Şekil 7, yazılım şirketinin sunduğu tablolardan ikinci müşteri verilerini temsil eden tablo için çıkarılan dendrogramı göstermektedir (Diğer müşteri dendrogramları için bk. Ek-1, Ek-2, …, Ek-9). Dendrogramın yatay ekseni, tabloda tanımlanan rolleri benzersiz bir şekilde yani her rolden yalnızca bir tane olacak şekilde temsil eder. Dendrogramın dikey ekseni ise roller arasındaki Jaccard mesafesini gösterir ve 0 ile 1 arasında bir değer alır.



Şekil 10. İkinci müşteri dendrogramının 0,3, 0,5 ve 0,7 noktalarından kestirimi

İki rol birbirine ne kadar benzerse, aralarındaki bağlantı hattının dikey eksen uzunluğu o kadar kısa ve aralarındaki Jaccard mesafesi de o kadar az olur. İki rolün birbirlerine daha az benzediği durumlarda aralarındaki Jaccard mesafesi büyür ve dolayısıyla iki rol arasındaki bağlantı çizgisi dikey eksende daha uzun olur. İki rolün tam olarak aynı olduğu yani birbirlerine tam olarak benzedikleri durumlarda ise aralarındaki mesafe ve dikey eksendeki bağlantı çizgilerinin uzunluğu 0'a eşit olacaktır. Benzer şekilde, iki rolün ayrıcalıklarının tamamen farklı olduğu, yani birbirlerine hiç benzemedikleri durumlarda aralarındaki Jaccard mesafesi ve dikey eksende rolleri birleştiren çizginin uzunluğu 1'e eşit olacaktır.

AHC algoritmasında kullanılacak rol kümelerinin sayısını belirlemek için dendrogramlara bazı eşik (threshold) değerleri uygulanmıştır (Hai vd, 2022). Dendrogram üzerinde bir eşik değeri ayarlamak için en ideal kesim noktası konusunda fikir birliği yoktur ve eşik değeri ayarı sezgisel olarak yapılmaktadır(C. Li ve Biswas, 2002). Şekil 10'daki dendrogram üzerinde örnek olması açısından sezgisel olarak üç farklı eşik değeri uygulanarak sonuçlar karşılaştırılmıştır.

Kümeleme işlemi

Kümeleme adımında ilk olarak birbirine 0,7 Jaccard mesafesinden yakın olan roller için kümeleme işlemi yapılmıştır. Dendrogram üzerinde kesim yapıldığında, yatay eksendeki eşik (kesim) çizgisini dikey eksende kesen çizgi sayısı, bu eşik değerinin oluşturduğu küme sayısını belirlemektedir. Örnek olarak gösterilen Müşteri 2 tablosu verileri için dendrogram kesimi, dendrogramda roller arasındaki Jaccard mesafesinin 0,7 değerine denk gelen çizgi ile yapıldığında sonuç olarak beklenen küme sayısı 11 olarak görülmektedir. Kümeleme işleminde uygulanan algoritmanın sözde kodu Tablo 16'da gösterildiği gibidir.

Dendrograma eşik değeri uygulandığında, kümeleme sonucunda oluşacak rol kümelerinin sayısı kullanılan kütüphane aracılığıyla üretilmektedir. Kümeleme fonksiyonunun girdi olarak beklediği rol kümelerinin sayısı elde edildikten sonra, Python tabanlı scikit-learn kütüphanesi kullanılarak 36 rolden oluşan bir müşteri tablosuna AHC algoritması uygulanmıştır. Daha sonra Şekil 10'da görüldüğü gibi 11 kümenin oluşması beklenmiş ve ortaya çıkan rol kümeleri Şekil 11'de gösterilmiştir.

Tablo 16. Kümeleme Adımının Sözde Kodu

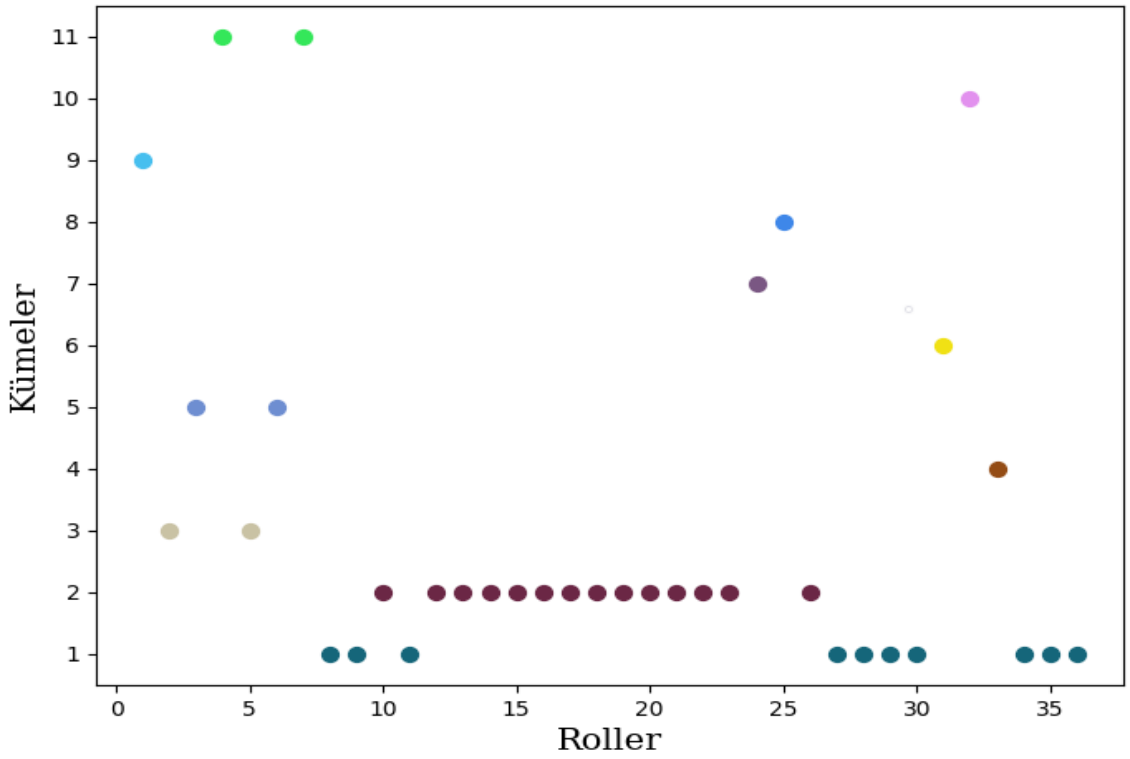
Algoritma: AHC Uygula ve Sonuçları Görsel Bir Şekilde Göster

Girdiler: roles_list_with_all_URLs, role_names, cluster_count

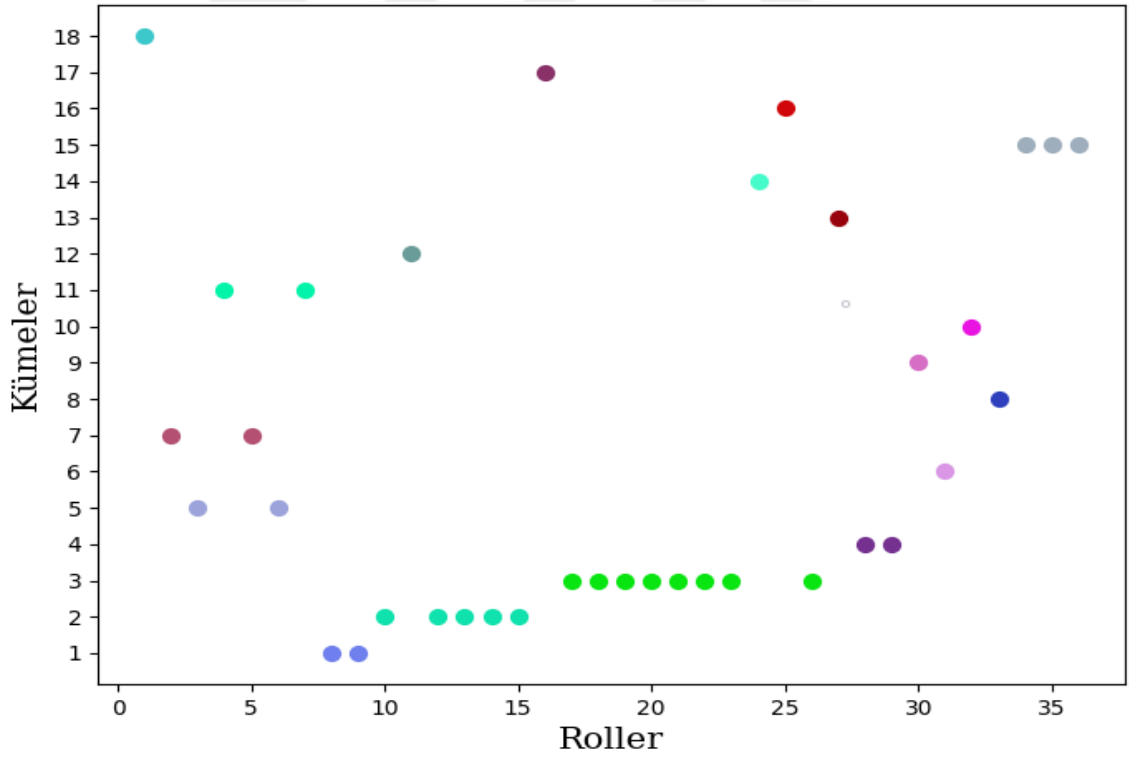
Çıktılar: kümeleme işlemi sonucunun grafiksel gösterimi

```
1  roles ← roles_list_with_all_URLS
2  similarity_measure ← "jaccard"
3  linkage_method ← "single"
4  model ← AHC(cluster_count, similarity_measure, linkage_method)
5  fit the model by using roles_list_with_all_URLs
6  roles_with_cluster_number ← model.labels_ // kümeleme sonucu
7  color_list ← boş dizi
8  for index ← 0 to length(cluster_count) do
9      r ← get_random_integer(0, 255)
10     g ← get_random_integer(0, 255)
11     b ← get_random_integer(0, 255)
12     new_random_color ← [r / 255, g / 255, b / 255]
13     color_list ← color_list + new_random_color
14 end for
15 for index ← 0 to length(role_names) do
16     color ← color_list[roles_with_cluster_number[index]]
17     x ← index + 1
18     y ← roles_with_cluster_number[index] + 1
19     set_coordinates_for_roles_with_specific_colors(x, y, color)
20 end for
21 draw_result_with_matplotlib()
```

Daha sonra birbirine 0,5 Jaccard mesafesinden daha yakın olan rollerin benzerliklerini belirlemek için dendrogram kesim aşamasında 0,5 noktasına denk gelen eşik değeri kullanılmıştır. Dendrogram 0,5 eşik değerinde kesildiğinde, Şekil 12'de gösterildiği gibi AHC sonucunda 18 küme oluşmuştur.

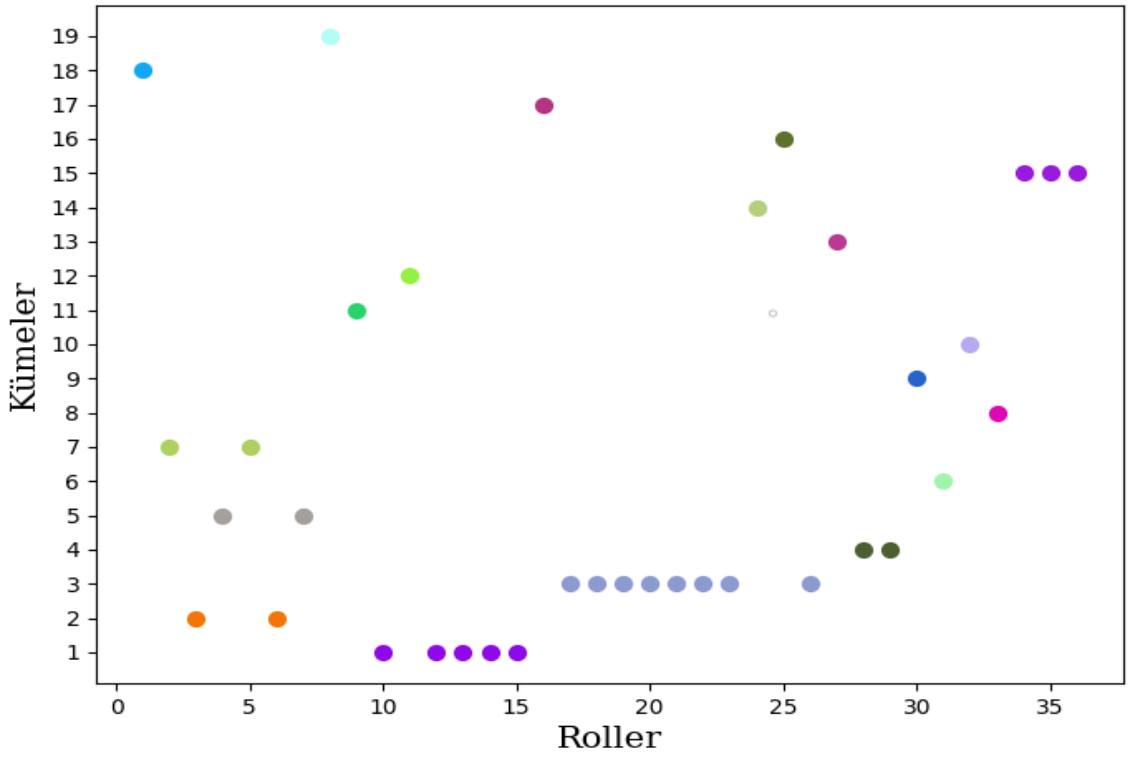


Şekil 11. Dendrogramın 0,7 noktasından kestirimiyle elde edilen AHC sonucu



Şekil 12. Dendrogramın 0,5 noktasından kestirimiyle elde edilen AHC sonucu

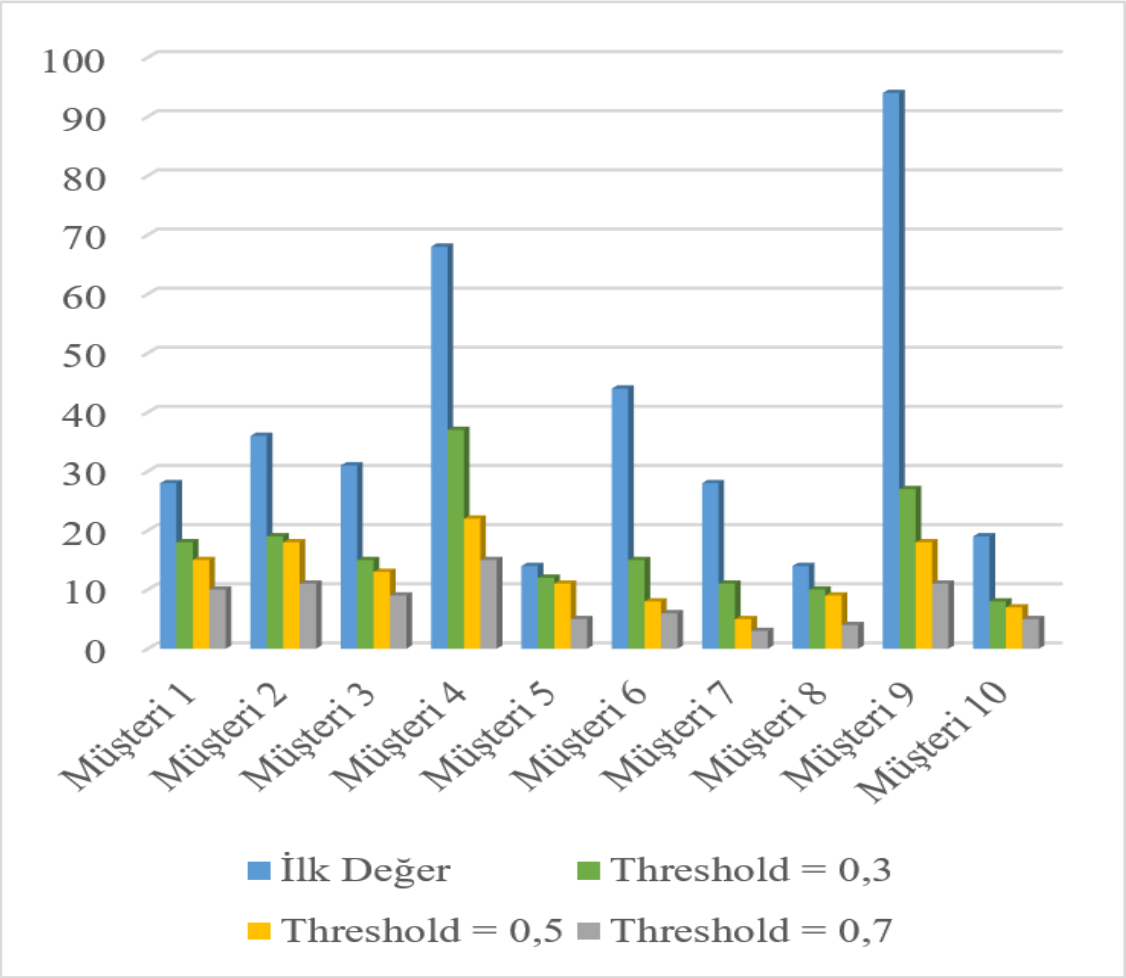
Son olarak dendrogram üzerinde 0,3 noktasından kesim yapılarak oluşturulacak rol kümesi sayısı belirlenmiştir. Şekil 13'de gösterildiği gibi dendrogram üzerinde 0,3 eşik değerinde kesim yapıldığında ise 19 rol kümesinin oluşacağı gözlenmiştir.



Şekil 13. Dendrogramın 0,3 noktasından kestirimiyle elde edilen AHC sonucu

Beklendiği gibi, dendrogram kestirimi aşamasında seçilen eşik değerler düşük tutulduğunda, daha fazla rol kümesi oluşmaktadır ve bu durum sistem yönetiminin kolaylaştırılması açısından pek istenen bir durum değildir. Ancak, birbirlerine yüksek oranda benzeyen rollerin bulunduğu aday kümeler meydana gelmektedir. Bu nedenle, eşik değerleri düşük tutulduğunda elde edilen kümelerin çok daha kullanışlı ve oluşturulacak yeni sistem veya geliştirilecek eski sistem için referans olarak kullanılmaya daha uygun oldukları gözlenmiştir. Öte yandan, dendrogramda eşik değerler yüksek tutulduğunda daha az sayıda küme oluşmuştur ve bu durum da sistem yönetiminin kolaylaştırılması açısından istenen bir durumdur. Ancak oluşturulan kümelerdeki roller birbirlerine daha az benzedikleri için aday rollerin referans olarak kullanılmaya daha uzak oldukları göze çarpmaktadır.

Bu yaklaşım, dendrogram kesimi için 0,1 ile 0,7 değerleri arasında 0,1 adımla artan eşik değerleri seçilerek tekrarlanmıştır. Örnek olması açısından, yalnızca 0,3, 0,5 ve 0,7 değerlerindeki dendrogram kesme noktaları için ortaya çıkan aday rol kümeleri ve içerdikleri rol sayıları Şekil 14'te gösterilmiştir.



Şekil 14. Her müşteri için, 0,3, 0,5 ve 0,7 threshold değerlerinde rol küme sayısı

Performans değerlendirme metrikleri

Müşteri veri tablolarının dendrogramları çıkarılmış ve her bir tablo için 0,1 ile 0,7 değerleri arasında, her adımda 0,1 kadar artacak şekilde eşik değerleri belirlenerek kümeleme işlemi yapılmıştır. Kümeleme işlemi sonucunda elde edilen rol kümeleri altın standart (yazılım firmasının Ar-Ge birimindeki uzmanlar tarafından hazırlanan gerçekte beklenen ideal rol kümeleri) (Palma vd, 2015) yani ground-truth (gT) (Lemoigne, 2008) verileri ile karşılaştırılmıştır.

Müşterilere ait tablolarda, 0,1 ile 0,7 değerleri arasında 0,1 adımlık artışla değişen her bir eşik değeri için ikili precision (pP), ikili recall (pR) ve ikili f1 skoru (pF1) değerleri (Kim, 2019; Kim vd, 2019) hesaplanmıştır (Tablo 19). Bir gT kümesindeki bir rolün diğer rollerle aynı gT kümesinde olmaması durumu, beklenen gerçek negatif (N) değeri temsil etmektedir. Aynı rolün, kümeleme algoritması tarafından tahmin edilen kümelerdeki diğer rollerle aynı kümede olmama durumu ise doğru negatif (True Negative, TN) değerlerini belirlemektedir. Elde edilen TN değerleri diğer değerlerle (TP, FP, FN) karşılaştırıldığında gözle görülür ve azımsanamayacak şekilde yüksek sayıda çıkmaktadır. Değerlendirme aşamasında ikili

karşılaştırma kullanıldığından dolayı bir rol aynı kümede olduğu her bir rol için yalnızca 1 doğru pozitif (True Positive, TP) değeri ve aynı kümede olmadığı her bir rol için çok sayıda TN değeri üretmektedir. Bu durumda accuracy (doğruluk) değerleri, yanıltıcı sonuçlar verme olasılığı yüksek olduğu için değerlendirme sonuçlarını içeren tabloya dahil edilmemiştir (Kanani ve Mccallum, 2023; Mower, 2005).

Tablo 17. Performans Değerlendirme Aşamasının Sözde Kodu

Algoritma: Pairwise Metriklerini Hesapla

Girdiler: predicted_clusters (2 boyutlu dizi), gT_clusters (2 boyutlu dizi)
Çıktılar: pP, pR, pF1

```
predicted_pairs ← yeni boş dizi // değişken ataması
gT_pairs ← yeni boş dizi // değişken ataması
for predicted_cluster_index ← 0 to length(predicted_clusters) do
    predicted_cluster ← predicted_clusters[predicted_cluster_index]
    if length(predicted_cluster) > 1 then
        for index ← 0 to length(predicted_cluster) - 1 do
            for inner_index ← index + 1 to length(predicted_cluster) do
                pair ← [predicted_cluster[index], predicted_cluster[inner_index]]
                predicted_pairs ← predicted_pairs + pair
            end for
        end for
    end if
end for
for gT_cluster_index ← 0 to length(gT_clusters) do
    gT_cluster ← gT_clusters [gT_cluster_index]
    if length(gT_cluster) > 1 then
        for index ← 0 to length(gT_cluster) - 1 do
            for inner_index ← index + 1 to length(gT_cluster) do
                pair ← [gT_cluster[index], gT_cluster[inner_index]]
                gT_pairs ← gT_pairs + pair
            end for
        end for
    end if
end for
length_of_predicted_pairs ← length(predicted_pairs)
length_of_gT_pairs ← length(gT_pairs)
```

Tablo 18. Performans Değerlendirme Aşamasının Sözde Kodu (Devam)

```
length_of_intersected_pairs ← 0
for index ← 0 to length(predicted_pairs) do
    if predicted_pairs[index] in gT_pairs then
        length_of_intersected_pairs ← length_of_intersected_pairs + 1
    end if
end for
pP ← length_of_intersected_pairs / length_of_predicted_pairs
pR ← length_of_intersected_pairs / length_of_gT_pairs
pF1 ← 2 * pR * pP / (pR + pP)
```

İkili karşılaştırma metriklerinin nasıl hesaplandığına ilişkin formüller aşağıdaki ifadelerde gösterilmiştir:

$$pP = \frac{|Tahmin\ Edilen\ K\ddot{u}melerdeki\ Rol\ \c{C}iftleri\ \cap\ gT\ K\ddot{u}melerdeki\ Rol\ \c{C}iftleri|}{Tahmin\ Edilen\ K\ddot{u}melerdeki\ Rol\ \c{C}iftleri}$$

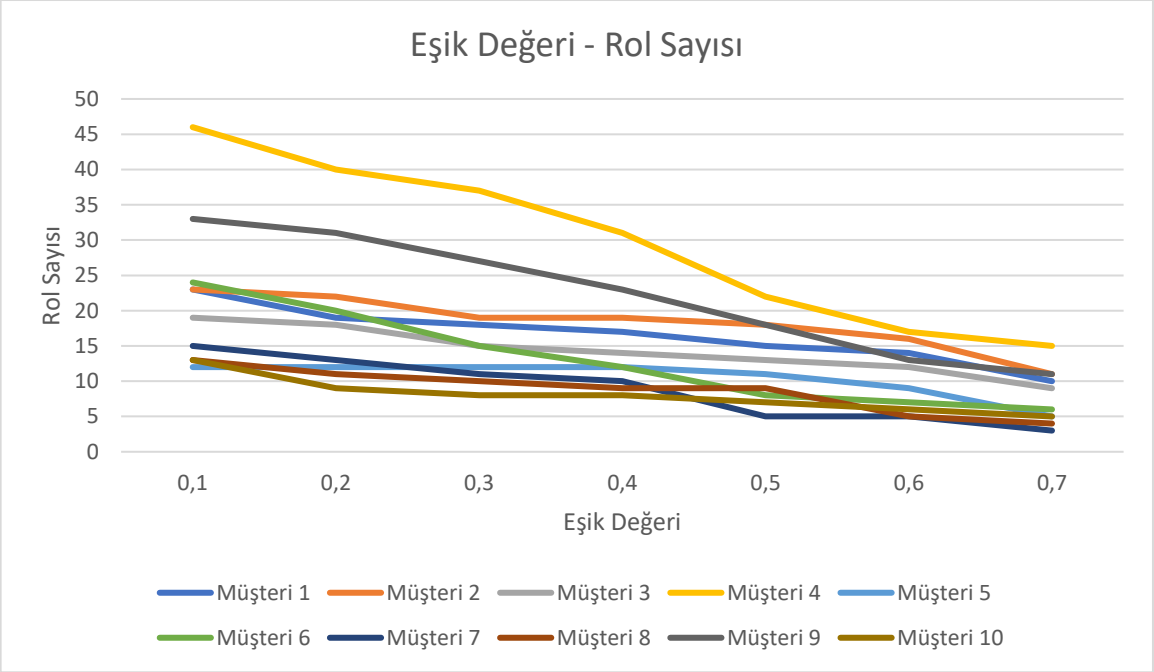
$$pR = \frac{|Tahmin\ Edilen\ K\ddot{u}melerdeki\ Rol\ \c{C}iftleri\ \cap\ gT\ K\ddot{u}melerdeki\ Rol\ \c{C}iftleri|}{gT\ K\ddot{u}melerdeki\ Rol\ \c{C}iftleri}$$

$$pF1 = \frac{2\ x\ pP\ x\ pR}{pP + pR}$$

İkili karşılaştırma metriklerinin hesaplanması işlemi için oluşturulan fonksiyonun algoritmasının sözde kodu yukarıda gösterildiği gibidir (Tablo 17). Gösterilen algoritmada gerçekte beklenen ideal kümeler gT_clusters, kümeleme işlemi sonucunda elde edilen tahmin edilmiş kümeler ise predicted_clusters isimli değişkenlerle belirtilmiştir.

ARAŞTIRMA BULGULARI

Her biri yazılım şirketinin farklı bir müşterisine ait RBAC sistem verilerini içeren 10 veri tablosu kullanılarak yazılım firmasının erişim kontrol sistemi verileri üzerinde deneysel bir çalışma yapılmıştır. Bir önceki bölümde örnek teşkil etmesi açısından yalnızca ikinci müşteriye ait verilerin analizlerinin sonuçları grafiksel olarak sunulmuştur.



Şekil 15. Her bir müşteri için eşik değeri – rol sayısı

Dendrogramlarda, 0,1 noktasından başlayıp 0,1 adımlık değerlerle artan kesmeler yapılarak gerçekleştirilen deneyde kesme değeri arttıkça, elde edilen rol kümelerinin sayılarının azaldığı ancak birbirlerine çok daha az benzeyen rollerden oluştuğu görülmüştür. Dendrogram kesme değeri 0,1 noktasına yaklaştıkça ise aynı kümeye atanan rollerin birbirine çok daha benzer olduğu ancak oluşan küme sayısının çok daha fazla olduğu görülmüştür. Yani, dendrogram kestirimi için seçilen eşik değer azaldıkça, aday kümeler (roller) daha kullanılabilir olmaktadır. Fakat oluşan küme sayısı (aday rol sayısı) arttığı için sistem yönetiminin kolaylığı açısından bir dezavantaj meydana gelmektedir. Öte yandan, dendrogram kestirimi için seçilen eşik değer arttıkça, meydana gelen küme sayısı (aday rol sayısı) azaldığı için sistem yönetiminin kolaylığı açısından bir iyileşme söz konusu olmaktadır. Fakat bu durumda, oluşan kümeler birbirlerine düşük oranda benzeyen rollerden meydana geldiği için bu kümeler aday rol olarak daha az kullanılabilir, diğer bir deyişle daha az tercih edilebilir olmaktadır (Şekil 15).

Tablo 19. Farklı Metrik Değerleri İçin Doğrulama Sonuçları

Tablo Adı	Metrik	Kesim Noktası (Threshold) Değeri						
		0,1	**0,2**	**0,3**	**0,4**	**0,5**	**0,6**	**0,7**
Müşteri 1	**pP**	**1,00**	**1,00**	**1,00**	0,90	0,73	0,66	0,35
	pR	0,58	0,84	0,95	0,95	**1,00**	**1,00**	**1,00**
	pF1	0,73	0,91	**0,97**	0,92	0,84	0,79	0,51
Müşteri 2	**pP**	**1,00**	**1,00**	0,58	0,58	0,59	0,51	0,21
	pR	0,60	0,73	0,87	0,87	0,90	0,90	**0,97**
	pF1	0,75	**0,85**	0,69	0,69	0,71	0,65	0,34
Müşteri 3	**pP**	**1,00**	**1,00**	0,64	0,64	0,65	0,59	0,27
	pR	0,57	0,70	0,93	0,97	**1,00**	**1,00**	**1,00**
	pF1	0,72	**0,82**	0,76	0,77	0,79	0,74	0,43
Müşteri 4	**pP**	0,75	0,79	**0,83**	0,52	0,30	0,14	0,08
	pR	0,57	0,70	0,91	0,94	0,96	**1,00**	**1,00**
	pF1	0,65	0,75	**0,87**	0,67	0,46	0,24	0,15
Müşteri 5	**pP**	**1,00**	**1,00**	**1,00**	**1,00**	0,50	0,18	0,04
	pR	**1,00**	**1,00**	**1,00**	**1,00**	**1,00**	**1,00**	**1,00**
	pF1	**1,00**	**1,00**	**1,00**	**1,00**	0,67	0,31	0,09
Müşteri 6	**pP**	**1,00**	**1,00**	0,58	0,55	0,46	0,44	0,22
	pR	0,39	0,57	0,65	0,88	**0,99**	**0,99**	**0,99**
	pF1	0,56	**0,73**	0,62	0,68	0,63	0,61	0,35
Müşteri 7	**pP**	**1,00**	**1,00**	**1,00**	**1,00**	0,68	0,68	0,32
	pR	0,47	0,55	0,76	0,88	**1,00**	**1,00**	**1,00**
	pF1	0,64	0,71	0,87	**0,93**	0,81	0,81	0,48
Müşteri 8	**pP**	**1,00**	**1,00**	**1,00**	**1,00**	**1,00**	0,41	0,35
	pR	0,09	0,36	0,45	**1,00**	**1,00**	**1,00**	**1,00**
	pF1	0,17	0,53	0,63	**1,00**	**1,00**	0,58	0,52
Müşteri 9	**pP**	**0,81**	0,77	0,76	0,68	0,50	0,40	0,35
	pR	0,99	0,99	**1,00**	**1,00**	**1,00**	**1,00**	**1,00**
	pF1	**0,89**	0,87	0,86	0,81	0,67	0,57	0,52
Müşteri 10	**pP**	**1,00**	0,62	0,65	0,65	0,66	0,46	0,39
	pR	0,33	0,74	0,96	0,96	**1,00**	**1,00**	**1,00**
	pF1	0,50	0,68	0,78	0,78	**0,79**	0,63	0,56

Elde edilen F1 skor değerleri yazılım firmasının Ar-Ge birimindeki uzmanların da katılımıyla analiz edildiğinde, dendrogramları kesmek için seçilen eşik değeri 0,6 değerinden

daha küçük olduğunda, oluşturulan yeni aday rollerin birbirine daha çok benzeyen ve genel olarak daha uyumlu rollerden oluştuğu görülmüştür. Dendrogram kesme noktası için seçilen eşik değeri arttıkça ve özellikle 0,6 değeri aşıldıktan sonra aday rollerin doğruluğunun önemli ölçüde azaldığı dikkat çekmektedir.

Tablo 19 incelendiğinde pP ve pF1 değerlerinin genellikle 0,4 eşik değerinin altındaki noktalarda en yüksek değerlere sahip olduğu, pR değerlerinin ise genellikle eşik değerinin 0,5'in üzerinde olduğu noktalarda en yüksek değerlere sahip olduğu görülmektedir. Bunun nedeni, Recall değerlerinin gerçek pozitif durumların ne kadar iyi tahmin edildiğini, Precision değerlerinin ise pozitif olarak tahmin edilen durumların ne kadar başarılı olduğunu göstermesidir (Fawcett, 2006). Eşik değeri arttıkça, kümeler birbirlerine daha az benzeyen rollerden oluştuğu için yanlış pozitif (False Positive, FP) olarak tahmin edilen değerlerin sayısı artmaktadır. Bu nedenle eşik değeri arttıkça Precision değeri azalmaktadır. Öte yandan yine eşik değeri arttıkça, yanlış negatif (FN) olarak tahmin edilen değerlerin sayısı azaldığı için Recall değeri artmaktadır. Çünkü eşik değeri yükseldikçe algoritma tarafından aynı kümeye birbirlerine daha az benzeyen roller yerleştirilmektedir. Kısaca olumsuz olarak tahmin edilen durumların sayısı azalmaktadır.

Elde edilen en yüksek değerler, her müşteri ve her bir değerlendirme metriği için Tablo 19'da kalın bir şekilde vurgulanarak gösterilmiştir (Her bir müşteri ve eşik değeri için pP, pR ve pF1 değerlerinin grafiksel gösterimi için bk. Ek-10, Ek-11 ve Ek-12). Tablo 19'da gösterildiği gibi algoritma, bazı eşik değerler uygulandığında, bazı müşteriler için uzmanlar tarafından gerçekte beklenen ideal rol setlerini üretmiştir. Firma uzmanları ile yapılan görüşmeler doğrultusunda üretilen pF1 değerlerinin doğruluğunun beklenen sonuçlarla oldukça uyumlu olduğu gözlemlenmiştir.

TARTIŞMA

Hane halklarının günlük yük eğrilerinin tüketim kalıplarına göre sınıflandırılması (AlMahamid ve Grolinger, 2022), akıllı şebeke sistemlerinde hassas hatların belirlenmesi (L. Yang ve Li, 2023), kentsel yol trafiği gürültü dağılım haritasının incelenmesi (Forouhid vd, 2023), Endonezya havacılık kazalarında insan kaynaklı faktörlerin incelenmesi (Passarella vd, 2022), kömürdeki elementel yakınlıkların değerlendirilmesi (Eminagaoglu vd, 2022) ve agro-morfolojik özelliklere dayalı buğday genotiplerinde genetik farklılık tahmini (Khalid vd, 2023) gibi çalışmalarda AHC algoritması kullanılmıştır.

Organizasyon modelinde yukarıdan aşağıya yaklaşımı doğrulamak için Jin ve arkadaşları, olay günlüklerine dayanan rol tabanlı bir yaklaşım önermişlerdir (Jin vd, 2007). Öte yandan, Baumgrass ve arkadaşları, iş süreçlerinden ve senaryo modellerinden aday roller türetmenin sıkıcı ve zorlu bir işlem olduğunu belirtmiş, bu işlemi otomatikleştirmek için UML aktivite diyagramlarını kullanarak yeni roller türetmeyi önermişlerdir (Baumgrass vd, 2011). Çoğu RMP yaklaşımı, sabit ağırlıklı izinlerin kullanımını uygularken Ma ve arkadaşları, izinleri, işlemlerin doğası veya nesnelerin hassasiyet derecesi ile ilişkili bir şekilde ağırlıklandırarak rolleri tanımlayan ağırlıklı bir rol madenciliği algoritması geliştirmişlerdir (Ma vd, 2010). Daha sonra Vaidya ve arkadaşları, mevcut ayrıcalıklardan tam ve doğru uygun roller seti keşfedilse bile halihazırda kullanımda olan roller ile sistemde oluşturulan roller arasında önemli farklılıklar olabileceğini, bu nedenle eski rolleri devre dışı bırakmanın ve aynı zamanda aday rolleri kullanmanın mümkün olmayacağını belirtmişlerdir (Vaidya vd, 2008). Bu doğrultuda, dağıtılan roller sorununu dikkate alan ve mevcut rolleri bu yönde çoğaltan ideal bir rol seti oluşturmak için Minimal Pertürbasyon RMP yaklaşımını önermişlerdir. Öte yandan Trnecka ve Trneckova, rol madenciliği sürecinin daha çok artımlı bir görev olduğuna dikkat çekmişlerdir ve rollerin yeniden hesaplanmasının verilerdeki küçük değişikliklerden sonra bile verimsiz olduğuna işaret eden artımlı bir algoritma önermişlerdir (Trnecka ve Trneckova, 2020).

Öte yandan, Molloy ve arkadaşlarına göre mevcut rol madenciliği araçları, anlamsal çıkarımlarla rolleri tanımlama yollarını tam olarak ele almamaktadır. Eldeki bilgi bir kullanıcı-ayrıcalık ilişkisi olduğunda, anlamı resmi kavram analizine dayanan rollere bakmayı önermişlerdir. Ayrıca, kullanıcı özniteliklerinde ifade edilebilecek ve kullanıcıya özgü karakteristik bilgiler mevcut olduğunda belirli bir anlamı olan roller üzerinde çalışılmasını önermişlerdir (Molloy vd, 2008).

Pan ve arkadaşları, RBAC sistemlerinin uzun süreli kullanımının veri kirliliğine yol açtığını ve bunun da sistemleri yönetme zorluğunu artırdığını vurgulamak için bir çalışma gerçekleştirmişlerdir. Kuruluşlar, departmanlarda, çalışanlarda ve iş süreçlerinde meydana gelen değişikliklere uyum sağlamak için RBAC sistemlerini değiştirebilmektedir. Ancak böyle bir değişikliğin diğer kullanıcılar ve roller üzerinde herhangi bir etkisinin olmamasının zor olduğundan bahsetmişlerdir. Bu durumda sistem, RBAC sisteminde tanımlanan yeni rollerle giderek daha da genişlemektedir. RBAC sisteminin, sistemin yapısal karmaşıklığını en aza indirmek ve orijinal işlevselliğini korumak için yeniden tasarlanması gerektiğini savunmuşlardır. Bu nedenle, orijinal işlevselliğini korumak ve yapısal karmaşıklığını azaltmak için RBAC sisteminin yeniden tasarlanmasını önermişlerdir (Pan vd, 2018).

Tüm bu çalışma ve yaklaşımlara rağmen literatür incelendiğinde, rol madenciliği için AHC kullanan herhangi bir çalışmaya rastlanmamıştır. Buna rağmen, rol madenciliğinde AHC kullanarak rol tabanlı erişim kontrol sistemlerini iyileştirmeye çalışan yaklaşım doğrultusunda elde edilen sonuçların, yazılım şirketinde uzmanların da görüşleri doğrultusunda, hali hazırda varlığını sürdüren rol tabanlı erişim kontrol sistemlerini geliştirme veya yeni rol tabanlı erişim kontrol sistemleri oluşturma aşamasında referans olarak kullanılabileceği görülmektedir. Öyle ki, yaklaşımımız, bazı kestirim değerlerinde tam olarak altın standart verilerine karşılık gelen sonuçları üretmiştir.

SONUÇ VE ÖNERİLER

Bu çalışmada, bir yazılım firmasının 10 farklı müşterisinin mevcut RBAC tabanlı erişim kontrol sistem verileri üzerinde AHC algoritması uygulanmıştır. Jaccard mesafesi, tek bağlantı yöntemi ve farklı eşik değerleri kullanılarak gerçekleştirilen AHC sonucunda elde edilen rol kümelerinin, geliştirilmiş veya geliştirilmekte olan RBAC sistemleri için yeni aday roller olarak kullanılıp kullanılamayacağı araştırılmıştır.

Yazılım firmasının Ar-Ge birimindeki uzmanların da katılımı ile sonuçlar incelendiğinde, dendrogramları kesmek için seçilen eşik değer düştükçe yeni aday rollerin birbirlerine daha çok benzediği ve genel olarak daha kullanışlı olduğu ancak daha fazla aday rol oluşturulduğu gözlenmiştir. Dendrogram kesme noktası için seçilen eşik değer arttıkça ise oluşturulacak aday rollerin sayıları azalsa da birbirlerine daha az benzeyen rollerin kümelenmesiyle oluştukları için daha az kullanışlı oldukları, yani bir sistemin oluşturulma veya var olan bir sistemin geliştirilme aşamasında aday rol olarak değerlendirilmeye daha uzak oldukları görülmektedir. Bazı kesme değerlerinde elde edilen sonuçlar uzmanlar tarafından belirlenen gerçek beklenen çıktı ile birebir aynı olmasına rağmen sonuçlar genel olarak değerlendirildiğinde bu durumun değişken olduğu ve kullanılabilirlik konusunda en iyi sonuca götürebilecek bir kesme değeri noktası hakkında net bir çıkarımın yapılamayacağı sonucuna varılmıştır. Bu nedenle, RBAC sistemlerini geliştirmek için bir rehber olarak kullanılabilecek bu yaklaşımı uygularken eşik değerini belirleme aşaması sezgi ve alan uzmanlığı gerektiriyor gibi görünmektedir.

Bu yaklaşım doğrultusunda, sistemin bu durumunda, önceki rollerin kümelenmesiyle oluşturulan yeni aday rollere doğrudan bir kullanıcının atanamadığı gözlemlenmiştir. Bu nedenle, şirket uzmanlarıyla da mutabık olunan ortak görüş doğrultusunda, üretilen aday rollerin mevcut bir RBAC sisteminde hali hazırda bulunan rollerle doğrudan yer değiştirmesi için kullanılmasının mümkün olmadığına karar verilmiştir. Ancak elde edilen sonuçların, yine şirket uzmanlarıyla yapılan ortak bir değerlendirme neticesinde, yeni bir RBAC sistemi oluştururken veya mevcut ve yıpranmış RBAC sistemlerini iyileştirirken, rollerin ve ayrıcalıklarının nasıl yapılandırılması gerektiği hakkında karar vermede iyi bir referans ve yardımcı olabileceği görülmüştür.

Gelecekteki çalışmalarda, RBAC sistemlerine uygulanan kümeleme aşamasında, beklenen ideal rol kümelerini veya onlara en yakın rol kümelerini elde etmedeki etkilerini ve

başarılarını incelemek için farklı mesafe ölçütleri, bağlantı yöntemleri ve algoritmalar kullanılacaktır. Ayrıca, aynı hedef sektördeki müşteriler için RBAC sistemleri oluşturulurken sektör bilgilerinin de sisteme dahil edilmesi doğrultusunda yaklaşımın performansı araştırılacaktır. Öte yandan, dendrogram kestirim değerlerinin sezgisel oluşunun yanı sıra, en uygun threshold değerinin belirlenmesi konusunda yardımcı olabilecek bir algoritma veya yaklaşımın geliştirilmesine yönelik araştırma ve deneysel çalışmalar yapılacaktır. Tüm bunların yanı sıra kümeleme algoritmalarına kolaylıkla uygulanabilen bir doğrulama yöntemine literatür incelemesi aşamasında pek rastlanmamıştır. İdeal çıktı olarak beklenen kümelerin sayısı ve ideal çıktı kümelerindeki nesne sayısı ile kümeleme algoritmasının ürettiği küme sayısı ve üretilen her kümedeki nesne sayısının farklı olmalarının mümkün olması, bu durumun nedeni olarak düşünülmektedir. Çünkü bu durumlarda sonuçlar kesin bir eşleşme ile karşılaştırılamamaktadır. Bu nedenle mevcut doğrulama yöntemlerinin iyileştirilmesi veya kümeleme algoritmaları için yeni doğrulama yöntemlerinin geliştirilmesine yönelik çalışmalar yapılabilir. Tüm bunlara ek olarak, elde edilen sonuçların grafikleri üzerinde örüntü analizleri yapılabilir.

KAYNAKÇA

Abdul, A. M., Mohammad, A. A. K., Venkat Reddy, P., Nuthakki, P., Kancharla, R., Joshi, R., & Kannaiya Raja, N. (2022). Enhancing Security of Mobile Cloud Computing by Trust- and Role-Based Access Control. *SCIENTIFIC PROGRAMMING, 2022*. https://doi.org/10.1155/2022/9995023

Ahmed, T., & Tripathi, A. R. (2003). Static Verification of Security Requirements in Role Based CSCW Systems. *Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies*, 196–203. https://doi.org/10.1145/775412.775438

Aidagulov, R., Glavatsky, S., & Mikhalev, A. (2022). Clustering Models. *Journal of Mathematical Sciences*, *262*. https://doi.org/10.1007/s10958-022-05841-9

AlMahamid, F., & Grolinger, K. (2022). Agglomerative Hierarchical Clustering with Dynamic Time Warping for Household Load Curve Clustering. *2022 IEEE CANADIAN CONFERENCE ON ELECTRICAL AND COMPUTER ENGINEERING(CCECE)*, 241–247. https://doi.org/10.1109/CCECE49351.2022.9918481

Anderer, S., Kempter, T., Scheuermann, B., & Mostaghim, S. (2021). The Dynamic Role Mining Problem: Role Mining in Dynamically Changing Business Environments. In T. Back, C. Wagner, J. Garibaldi, H. K. Lam, M. Cottrell, J. J. Merelo, & K. Warwick (Eds.), *PROCEEDINGS OF THE 13TH INTERNATIONAL JOINT CONFERENCE ON COMPUTATIONAL INTELLIGENCE (IJCCI)* (pp. 37–48). https://doi.org/10.5220/0010654000003063

Anderer, S., Kreppein, D., Scheuermann, B., & Mostaghim, S. (2020). The addRole-EA: A New Evolutionary Algorithm for the Role Mining Problem. In J. J. Merelo, J. Garibaldi, C. Wagner, T. Back, K. Madani, & K. Warwick (Eds.), *PROCEEDINGS OF THE 12TH INTERNATIONAL JOINT CONFERENCE ON COMPUTATIONAL INTELLIGENCE (IJCCI)* (pp. 155–166). https://doi.org/10.5220/0010025401550166

Aziz, N. A. A., Salleh, S. S., Mohamad, D., & Omar, M. (2010). Investigating Jaccard Distance similarity measurement constriction on handwritten pen-based input digit. *2010 International Conference on Science and Social Research (CSSR 2010)*, 1181–1185. https://doi.org/10.1109/CSSR.2010.5773712

Baumgrass, A., Strembeck, M., & Rinderle-Ma, S. (2011). Deriving Role Engineering Artifacts from Business Processes and Scenario Models. *Proceedings of the 16th ACM Symposium on Access Control Models and Technologies*, 11–20. https://doi.org/10.1145/1998441.1998445

Billard, L., & Diday, E. (2020). Agglomerative Hierarchical Clustering. In *CLUSTERING METHODOLOGY FOR SYMBOLIC DATA* (pp. 261–316).

Bonatti, P., Galdi, C., & Torres, D. (2015). Event-driven RBAC. *Journal of Computer Security*, *23*, 709–757. https://doi.org/10.3233/JCS-150539

Bu, J., Liu, W., Pan, Z., & Ling, K. (2020). Comparative Study of Hydrochemical Classification Based on Different Hierarchical Cluster Analysis Methods. *International Journal of Environmental Research and Public Health*, *17*, 9515. https://doi.org/10.3390/ijerph17249515

Cai, F., He, J., Zardari, Z. A., & Han, S. (2020). Distributed management of permission for access control model. *JOURNAL OF INTELLIGENT & FUZZY SYSTEMS*, *38*(2, SI), 1539–1548. https://doi.org/10.3233/JIFS-179517

Calo, S., Chen, H., Li, N., Li, T., Lobo, J., Molloy, I., & Wang, Q. (2008). *Some Usability Considerations in Access Control Systems*.

Chen, J., MacEachren, A. M., & Peuquet, D. J. (2009). Constructing Overview plus Detail Dendrogram-Matrix Views. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*, *15*(6), 889–896. https://doi.org/10.1109/TVCG.2009.130

Cochez, M., & Mou, H. (2015). Twister Tries: Approximate Hierarchical Agglomerative Clustering for Average Distance in Linear Time. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 505–517. https://doi.org/10.1145/2723372.2751521

Cochez, M., & Neri, F. (2015). Scalable Hierarchical Clustering: Twister Tries with a Posteriori Trie Elimination. *2015 IEEE Symposium Series on Computational Intelligence*, 756–763. https://doi.org/10.1109/SSCI.2015.12

Colantonio, A., Di Pietro, R., & Ocello, A. (2008). A Cost-Driven Approach to Role Engineering. *Proceedings of the 2008 ACM Symposium on Applied Computing*, 2129–2136. https://doi.org/10.1145/1363686.1364198

Colantonio, A., Di Pietro, R., Ocello, A., & Verde, N. V. (2009). A Formal Framework to Elicit Roles with Business Meaning in RBAC Systems. *Proceedings of the 14th ACM Symposium on Access Control Models and Technologies*, 85–94. https://doi.org/10.1145/1542207.1542223

Coyne, E. J. (1996). Role engineering. *Proceedings of the First ACM Workshop on Role-Based Access Control*, 4–es.

Coyne, E., Weil, T., & Kuhn, D. (2011). Role Engineering: Methods and Standards. *IT Professional*, *13*, 54–57. https://doi.org/10.1109/MITP.2011.105

Crampton, J. (2005). Understanding and Developing Role-Based Administrative Models. *Proceedings of the 12th ACM Conference on Computer and Communications Security*, 158–167. https://doi.org/10.1145/1102120.1102143

Crass, S., Lackner, A., Begic, N., Mirhosseini, S. A. M., & Kirchmayr, N. (2022). Collaborative Administration of Role-Based Access Control in Smart Contracts. *2022 4TH CONFERENCE ON BLOCKCHAIN RESEARCH & APPLICATIONS FOR INNOVATIVE NETWORKS AND SERVICES (BRAINS)*, 87–94. https://doi.org/10.1109/BRAINS55737.2022.9909116

Dekker, M. A. C., Crampton, J., & Etalle, S. (2008). RBAC Administration in Distributed Systems. *SACMAT'08: PROCEEDINGS OF THE 13TH ACM SYMPOSIUM ON ACCESS CONTROL AND TECHNOLOGIES*, 93–101.

Eelbode, T., Bertels, J., Berman, M., Vandermeulen, D., Maes, F., Bisschops, R., & Blaschko, M. B. (2020). Optimization for Medical Image Segmentation: Theory and Practice When Evaluating With Dice Score or Jaccard Index. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, *39*(11), 3679–3690. https://doi.org/10.1109/TMI.2020.3002417

Eminagaoglu, M., Oskay, R. G., & Karayigit, A. I. (2022). Evaluation of elemental affinities in coal using agglomerative hierarchical clustering algorithm: A case study in a thick and mineable coal seam (kM2) from Soma Basin (W. Turkey). *INTERNATIONAL JOURNAL OF COAL GEOLOGY*, *259*. https://doi.org/10.1016/j.coal.2022.104045

Ene, A., Horne, W., Milosavljevic, N., Rao, P., Schreiber, R., & Tarjan, R. E. (2008). Fast Exact and Heuristic Methods for Role Minimization Problems. *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies*, 1–10. https://doi.org/10.1145/1377836.1377838

Eppstein, D. (2001). Fast Hierarchical Clustering and Other Applications of Dynamic Closest Pairs. *ACM J. Exp. Algorithmics*, *5*, 1–es. https://doi.org/10.1145/351827.351829

Estivill-Castro Vladimir and Yang, J. (2000). Fast and Robust General Purpose Clustering Algorithms. In J. Mizoguchi Riichiro and Slaney (Ed.), *PRICAI 2000 Topics in Artificial Intelligence* (pp. 208–218). Springer Berlin Heidelberg.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. https://doi.org/https://doi.org/10.1016/j.patrec.2005.10.010

Ferraiolo, D. F., Gilbert, D., & Lynch, N. (1993). *An Examination of Federal and Commercial Access Control Policy Needs*.

Ferraiolo, D. F., Sandhu, R., Gavrila, S., Kuhn, D. R., & Chandramouli, R. (2001). Proposed NIST Standard for Role-Based Access Control. *ACM Trans. Inf. Syst. Secur.*, *4*(3), 224–274. https://doi.org/10.1145/501978.501980

Ferraiolo, D., & Kuhn, R. (1992). Role-Based Access Control. 15th NIST-NSA National Computer Security Conference,(pp. 554-563). *Baltimore, Maryland, USA*.

Ferreira, L., & Hitchcock, D. B. (2009). A Comparison of Hierarchical Methods for Clustering Functional Data. *Communications in Statistics - Simulation and Computation*, *38*(9), 1925–1949. https://doi.org/10.1080/03610910903168603

Filkov Vladimir and Skiena, S. (2004). Heterogeneous Data Integration with the Consensus Clustering Formalism. In E. Rahm (Ed.), *Data Integration in the Life Sciences* (pp. 110–123). Springer Berlin Heidelberg.

Forouhid, A. E., Khosravi, S., & Mahmoudi, J. (2023). Noise Pollution Analysis Using Geographic Information System, Agglomerative Hierarchical Clustering and Principal Component Analysis in Urban Sustainability (Case Study: Tehran). *SUSTAINABILITY*, *15*(3). https://doi.org/10.3390/su15032112

Fragkos, G., Johnson, J., & Tsiropoulou, E. E. (2022). Dynamic Role-Based Access Control Policy for Smart Grid Applications: An Offline Deep Reinforcement Learning Approach. *IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS*, *52*(4), 761–773. https://doi.org/10.1109/THMS.2022.3163185

Fraley, C., & Raftery, A. E. (1998). How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *The Computer Journal*, *41*(8), 578–588. https://doi.org/10.1093/comjnl/41.8.578

Frank, M., Streich, A. P., Basin, D., & Buhmann, J. M. (2009). A Probabilistic Approach to Hybrid Role Mining. *Proceedings of the 16th ACM Conference on Computer and Communications Security*, 101–111. https://doi.org/10.1145/1653662.1653675

Fuchs, L., & Meier, S. (2011). The Role Mining Process Model - Underlining the Need for a Comprehensive Research Perspective. *Proceedings of the 2011 Sixth International Conference on Availability, Reliability and Security*, 35–42. https://doi.org/10.1109/ARES.2011.12

Fuchs, L., & Pernul, G. (2008). HyDRo — Hybrid Development of Roles. *Proceedings of the 4th International Conference on Information Systems Security*, 287–302. https://doi.org/10.1007/978-3-540-89862-7_24

Gallaher, M., O'Connor, A., Kropp, B., & Tassey, G. (2002). *The economic impact of role-based access control*.

Garg, T., Kagalwalla, N., Puthran, S., Churi, P., & Pawar, A. (2023). A novel approach of privacy-preserving data sharing system through data-tagging with role-based access control. *WORLD JOURNAL OF ENGINEERING*, *20*(1), 12–28. https://doi.org/10.1108/WJE-04-2021-0218

Ghazal, R., Malik, A. K., Raza, B., Qadeer, N., Qamar, N., & Bhatia, S. (2021). Agent-Based Semantic Role Mining for Intelligent Access Control in Multi-Domain Collaborative Applications of Smart Cities. *SENSORS*, *21*(13). https://doi.org/10.3390/s21134253

Gilpin, S., Qian, B., & Davidson, I. (2013). Efficient Hierarchical Clustering of Large High Dimensional Datasets. *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, 1371–1380. https://doi.org/10.1145/2505515.2505527

Gower, J. C., & Warrens, M. J. (2014). Similarity, dissimilarity, and distance, measures of. *Wiley StatsRef: Statistics Reference Online*, 1–11.

Gronau, I., & Moran, S. (2007). Optimal implementations of UPGMA and other common clustering algorithms. *Information Processing Letters*, *104*(6), 205–210. https://doi.org/https://doi.org/10.1016/j.ipl.2007.07.002

Hai, V. Van, Nhung, H. L. L. Le, & Jasek, R. (2022). Toward Applying Agglomerative Hierarchical Clustering in Improving the Software Development Effort Estimation. In R. Silhavy (Ed.), *SOFTWARE ENGINEERING PERSPECTIVES IN SYSTEMS, VOL. 1* (Vol. 501, pp. 353–371). https://doi.org/10.1007/978-3-031-09070-7_30

Harding, C. (2016). Geographic Visualization of Solar Radiation Flux Data for Teaching Purposes. In E. Banissi, M. W. M. Bannatyne, F. Bouali, R. Burkhard, J. Counsell, U. Cvek, M. J. Eppler, G. Grinstein, W. D. Huang, S. Kernbach, C. C. Lin, F. Lin, F. T. Marchese, C. M. Pun, M. Sarfraz, M. Trutschl, A. Ursyn, G. Venturini, T. G. Wyeld, & J. J. Zhang (Eds.), *PROCEEDINGS 2016 20TH INTERNATIONAL CONFERENCE INFORMATION VISUALISATION IV 2016* (pp. 384–389). https://doi.org/10.1109/IV.2016.85

Hasebe, K., Mabuchi, M., & Matsushita, A. (2010). Capability-Based Delegation Model in RBAC. *Proceedings of the 15th ACM Symposium on Access Control Models and Technologies*, 109–118. https://doi.org/10.1145/1809842.1809861

Hassen, H. R., Bouabdallah, A., Bettahar, H., & Challal, Y. (2007). Key management for content access control in a hierarchy. *Computer Networks*, *51*(11), 3197–3219. https://doi.org/https://doi.org/10.1016/j.comnet.2006.12.011

Hennig, C. (2022). An empirical comparison and characterisation of nine popular clustering methods. *Advances in Data Analysis and Classification*, *16*(1), 201–229. https://doi.org/10.1007/s11634-021-00478-z

Hu, J., Zhang, Y., Li, R., & Lu, Z. (2010). Role Updating for Assignments. *Proceedings of the 15th ACM Symposium on Access Control Models and Technologies*, 89–98. https://doi.org/10.1145/1809842.1809859

Huang, H., Shang, F., Liu, J., & Du, H. (2015). Handling least privilege problem and role mining in RBAC. *JOURNAL OF COMBINATORIAL OPTIMIZATION*, *30*(1), 63–86. https://doi.org/10.1007/s10878-013-9633-9

Hwang, C.-M., & Yang, M.-S. (2014). New Similarity Measures Between Generalized Trapezoidal Fuzzy Numbers Using the Jaccard Index. *INTERNATIONAL JOURNAL*

OF UNCERTAINTY FUZZINESS AND KNOWLEDGE-BASED, *22*(6), 831–844. https://doi.org/10.1142/S0218488514500445

Hyder, M. F., Arshad, S., Arfeen, A., & Fatima, T. (2022). Privacy preserving mobile forensic framework using role-based access control and cryptography. *CONCURRENCY AND COMPUTATION-PRACTICE & EXPERIENCE*, *34*(23). https://doi.org/10.1002/cpe.7178

Jaccard, P. (1908). Nouvelles Recherches Sur la Distribution Florale. *Bulletin de La Societe Vaudoise Des Sciences Naturelles*, *44*, 223–270. https://doi.org/10.5169/seals-268384

Jafarzadegan, M., Safi-Esfahani, F., & Beheshti, Z. (2022). An Agglomerative Hierarchical Clustering Framework for Improving the Ensemble Clustering Process. *Cybernetics and Systems*, *53*(8), 679–701. https://doi.org/10.1080/01969722.2022.2042917

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Englewood Cliffs, NJ: Prentice Hall. https://homepages.inf.ed.ac.uk/rbf/BOOKS/JAIN/Clustering_Jain_Dubes.pdf

Jarman, A. (2020). *Hierarchical Cluster Analysis: Comparison of Single linkage,Complete linkage, Average linkage and Centroid Linkage Method*. https://doi.org/10.13140/RG.2.2.11388.90240

Jha, S., Li, N., Tripunitara, M., Wang, Q., & Winsborough, W. (2008). Towards Formal Verification of Role-Based Access Control Policies. *IEEE Transactions on Dependable and Secure Computing*, *5*(4), 242–255. https://doi.org/10.1109/TDSC.2007.70225

Jin, T., Wang, J., & Wen, L. (2007). Organizational modeling from event logs. *SIXTH INTERNATIONAL CONFERENCE ON GRID AND COOPERATIVE COMPUTING, PROCEEDINGS*, 670+. https://doi.org/10.1109/GCC.2007.93

John, J. C., Sural, S., Atluri, V., & Vaidya, J. S. (2012). Role Mining under Role-Usage Cardinality Constraint. In D. Gritzalis, S. Furnell, & M. Theoharidou (Eds.), *INFORMATION SECURITY AND PRIVACY RESEARCH* (Vol. 376, pp. 150–161).

Kamboj, P., Khare, S., & Pal, S. (2021). User authentication using Blockchain based smart contract in role-based access control. *PEER-TO-PEER NETWORKING AND APPLICATIONS*, *14*(5, SI), 2961–2976. https://doi.org/10.1007/s12083-021-01150-1

Kanani, P., & Mccallum, A. (2023). *Efficient strategies for improving partitioning-based author coreference by incorporating Web pages as graph nodes*.

Kern, A., Kuhlmann, M., Schaad, A., & Moffett, J. (2002). Observations on the Role Life-Cycle in the Context of Enterprise Security Management. *Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies*, 43–51. https://doi.org/10.1145/507711.507718

Khalid, A., Hameed, A., & Tahir, M. F. (2023). Estimation of genetic divergence in wheat genotypes based on agro-morphological traits through agglomerative hierarchical clustering and principal component analysis. *CEREAL RESEARCH COMMUNICATIONS*, *51*(1), 217–224. https://doi.org/10.1007/s42976-022-00287-w

Kikugawa, M., Chang, W.-D., Hwang, S., & Shin, J. (2010). A Fast Shape Retrieval using Dendrogram. *2010 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS IJCNN 2010*.

Kim, J. (2019). A fast and integrative algorithm for clustering performance evaluation in author name disambiguation. *Scientometrics*, *120*(2), 661–681. https://doi.org/10.1007/s11192-019-03143-7

Kim, J., Kim, J., & Owen-Smith, J. (2019). Generating automatically labeled data for author name disambiguation: an iterative clustering method. *SCIENTOMETRICS*, *118*(1), 253–280. https://doi.org/10.1007/s11192-018-2968-3

Kirkpatrick, M. S., & Bertino, E. (2010). Enforcing Spatial Constraints for Mobile RBAC Systems. *Proceedings of the 15th ACM Symposium on Access Control Models and Technologies*, 99–108. https://doi.org/10.1145/1809842.1809860

Kosub, S. (2019). A note on the triangle inequality for the Jaccard distance. *Pattern Recognition Letters*, *120*, 36–38. https://doi.org/https://doi.org/10.1016/j.patrec.2018.12.007

Kuhlmann, M., Shohat, D., & Schimpf, G. (2003). Role Mining - Revealing Business Roles for Security Administration Using Data Mining Technology. *Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies*, 179–186. https://doi.org/10.1145/775412.775435

Kull, M., & Vilo, J. (2008). Fast approximate hierarchical clustering using similarity heuristics. *BioData Mining*, *1*, 9. https://doi.org/10.1186/1756-0381-1-9

Lampson, B. (1974). Protection. *ACM SIGOPS Operating Systems Review*, *8*, 18–24. https://doi.org/10.1145/775265.775268

Larsen, J., Hansen, L. K., Have, A. S., Christiansen, T., & Kolenda, T. (2002). Webmining: learning from the world wide web. *Computational Statistics & Data Analysis*, *38*(4), 517–532. https://doi.org/https://doi.org/10.1016/S0167-9473(01)00076-7

Laverdière, M.-A., Julien, K., & Merlo, E. (2021). RBAC protection-impacting changes identification: A case study of the security evolution of two PHP applications. *Information and Software Technology*, *139*, 106630. https://doi.org/https://doi.org/10.1016/j.infsof.2021.106630

Lazouski, A., Martinelli, F., & Mori, P. (2010). Usage control in computer security: A survey. *COMPUTER SCIENCE REVIEW*, *4*(2), 81–99. https://doi.org/10.1016/j.cosrev.2010.02.002

Le, H. T., Shar, L. K., Bianculli, D., Briand, L. C., & Nguyen, C. D. (2022). Automated reverse engineering of role-based access control policies of web applications. *Journal of Systems and Software*, *184*, 111109. https://doi.org/https://doi.org/10.1016/j.jss.2021.111109

Leblebici, O., Kardas, G., & Tuglular, T. (2022). A Domain-Specific Language for the Document-Based Model-Driven Engineering of Business Applications. *IEEE Access*, *10*, 104093–104110. https://doi.org/10.1109/ACCESS.2022.3210530

Lemoigne, Y. (2008). *Molecular Imaging: Computer Reconstruction and Practice: Preface*. v–vii.

Li, C., & Biswas, G. (2002). Unsupervised learning with mixed numeric and nominal data. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, *14*(4), 673–690. https://doi.org/10.1109/TKDE.2002.1019208

Li, N., & Mao, Z. (2007). Administration in Role-Based Access Control. *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security*, 127–138. https://doi.org/10.1145/1229285.1229305

Li, Y., Du, Z., Fu, Y., & Liu, L. (2022). Role-Based Access Control Model for Inter-System Cross-Domain in Multi-Domain Environment. *APPLIED SCIENCES-BASEL*, *12*(24). https://doi.org/10.3390/app122413036

Lin, B., Chen, Y., Chen, X., & Yu, Y. (2012). Comparison between JSON and XML in Applications Based on AJAX. *2012 International Conference on Computer Science and Service System*, 1174–1177. https://doi.org/10.1109/CSSS.2012.297

Liu, B., Michalas, A., & Warinschi, B. (2022). Cryptographic Role-Based Access Control, Reconsidered. In C. Ge & F. Guo (Eds.), *PROVABLE AND PRACTICAL SECURITY, PROVSEC 2022* (Vol. 13600, pp. 282–289). https://doi.org/10.1007/978-3-031-20917-8_19

Liu, N., Xu, Z., Zeng, X.-J., & Ren, P. (2021). An agglomerative hierarchical clustering algorithm for linear ordinal rankings. *Information Sciences*, *557*, 170–193. https://doi.org/https://doi.org/10.1016/j.ins.2020.12.056

Lu, H., Vaidya, J., & Atluri, V. (2008). Optimal Boolean Matrix Decomposition: Application to Role Engineering. *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, 297–306. https://doi.org/10.1109/ICDE.2008.4497438

Lu, H., Vaidya, J., & Atluri, V. (2014). An optimization framework for role mining. *JOURNAL OF COMPUTER SECURITY*, *22*(1), 1–31. https://doi.org/10.3233/JCS-130484

Ma, X., Li, R., & Lu, Z. (2010). Role Mining Based on Weights. *SACMAT 2010: PROCEEDINGS OF THE 15TH ACM SYMPOSIUM ON ACCESS CONTROL AND TECHNOLOGIES*, 65–74.

McQuitty, L. L. (1960). Hierarchical linkage analysis for the isolation of types. *Educational and Psychological Measurement*, *20*(1), 55–67.

Mitra, B., Sural, S., Vaidya, J., & Atluri, V. (2016). A Survey of Role Mining. *ACM COMPUTING SURVEYS*, *48*(4). https://doi.org/10.1145/2871148

Molloy, I., Chen, H., Li, T., Wang, Q., Li, N., Bertino, E., Calo, S., & Lobo, J. (2008). Mining Roles with Semantic Meanings. *SACMAT'08: PROCEEDINGS OF THE 13TH ACM SYMPOSIUM ON ACCESS CONTROL AND TECHNOLOGIES*, 21–30.

Molloy, I., Chen, H., Li, T., Wang, Q., Li, N., Bertino, E., Calo, S., & Lobo, J. (2010a). Mining Roles with Multiple Objectives. *ACM TRANSACTIONS ON INFORMATION AND SYSTEM SECURITY*, *13*(4). https://doi.org/10.1145/1880022.1880030

Molloy, I., Chen, H., Li, T., Wang, Q., Li, N., Bertino, E., Calo, S., & Lobo, J. (2010b). Mining Roles with Multiple Objectives. *ACM Trans. Inf. Syst. Secur.*, *13*(4). https://doi.org/10.1145/1880022.1880030

Molloy, I., Li, N., Li, T., Mao, Z., Wang, Q., & Lobo, J. (2009). Evaluating Role Mining Algorithms. *Proceedings of the 14th ACM Symposium on Access Control Models and Technologies*, 95–104. https://doi.org/10.1145/1542207.1542224

Mondal, S., Sural, S., & Atluri, V. (2009). Towards Formal Security Analysis of GTRBAC Using Timed Automata. *Proceedings of the 14th ACM Symposium on Access Control Models and Technologies*, 33–42. https://doi.org/10.1145/1542207.1542214

Moore, G. W., Goodman, M., & Barnabas, J. (1973). An iterative approach from the standpoint of the additive hypothesis to the dendrogram problem posed by molecular data sets. *Journal of Theoretical Biology*, *38*(3), 423–457. https://doi.org/https://doi.org/10.1016/0022-5193(73)90251-8

Mower, J. P. (2005). PREP-Mt: predictive RNA editor for plant mitochondrial genes. *BMC Bioinformatics*, *6*(1), 96. https://doi.org/10.1186/1471-2105-6-96

Mucha, H. J., Bartel, H. G., & Dolata, J. (2005). Techniques of rearrangements in binary trees (dendrograms) and applications. *MATCH-COMMUNICATIONS IN MATHEMATICAL AND IN COMPUTER CHEMISTRY*, *54*(3), 561–582.

Müllner, D. (2011). *Modern hierarchical, agglomerative clustering algorithms*.

Müllner, D. (2013). fastcluster: Fast Hierarchical, Agglomerative Clustering Routines for R and Python. *Journal of Statistical Software*, *53*(9), 1–18. https://doi.org/10.18637/jss.v053.i09

Narouei, M., & Takabi, H. (2015). Towards an Automatic Top-down Role Engineering Approach Using Natural Language Processing Techniques. *Proceedings of the 20th ACM Symposium on Access Control Models and Technologies*, 157–160. https://doi.org/10.1145/2752952.2752958

Neumann, G., & Strembeck, M. (2002). A Scenario-Driven Role Engineering Process for Functional RBAC Roles. *Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies*, 33–42. https://doi.org/10.1145/507711.507717

Ni, Q., Lobo, J., Calo, S., Rohatgi, P., & Bertino, E. (2009). Automating Role-Based Provisioning by Learning from Examples. *Proceedings of the 14th ACM Symposium on Access Control Models and Technologies*, 75–84. https://doi.org/10.1145/1542207.1542222

O'Connor, A. C., & Loomis, R. J. (2010). 2010 economic analysis of role-based access control. *NIST, Gaithersburg, MD*, *20899*.

Palma, G., Vidal, M.-E., Haag, E., Raschid, L., & Thor, A. (2015). Determining similarity of scientific entities in annotation datasets. *Database : The Journal of Biological Databases and Curation*, *2015*. https://doi.org/10.1093/database/bau123

Pan, N., Sun, L., He, L.-S., & Zhu, Z.-Q. (2018). An Approach for Hierarchical RBAC Reconfiguration With Minimal Perturbation. *IEEE ACCESS*, *6*, 40389–40399. https://doi.org/10.1109/ACCESS.2017.2782838

Passarella, R., Oktariani, G., Kurniawan, D., & Sari, P. (2022). Using the Agglomerative Hierarchical Clustering Method to Examine Human Factors in Indonesian Aviation Accidents. *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, *13*(9), 325–331.

PHIPPS, J. B. (1976). DENDROGRAM TOPOLOGY - CAPACITY AND RETRIEVAL. *CANADIAN JOURNAL OF BOTANY-REVUE CANADIENNE DE BOTANIQUE*, *54*(8), 679–685. https://doi.org/10.1139/b76-072

Rao, K. R., Nayak, A., Ray, I. G., Rahulamathavan, Y., & Rajarajan, M. (2021). Role recommender-RBAC: Optimizing user-role assignments in RBAC. *Computer Communications*, *166*, 140–153. https://doi.org/https://doi.org/10.1016/j.comcom.2020.12.006

Reith, M., Niu, J., & Winsborough, W. H. (2009). Toward Practical Analysis for Trust Management Policy. *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, 310–321. https://doi.org/10.1145/1533057.1533098

Roeckle, H., Schimpf, G., & Weidinger, R. (2000). Process-Oriented Approach for Role-Finding to Implement Role-Based Security Administration in a Large Industrial Organization. *Proceedings of the Fifth ACM Workshop on Role-Based Access Control*, 103–110. https://doi.org/10.1145/344287.344308

Rokach, L. (2010). A survey of Clustering Algorithms. In L. Maimon Oded and Rokach (Ed.), *Data Mining and Knowledge Discovery Handbook* (pp. 269–298). Springer US. https://doi.org/10.1007/978-0-387-09823-4_14

Saenko, I., & Kotenko, I. (2017). Administrating Role-Based Access Control by Genetic Algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 1463–1470. https://doi.org/10.1145/3067695.3082509

Sandhu, R., Bhamidipati, V., & Munawer, Q. (1999). The ARBAC97 Model for Role-Based Administration of Roles. *ACM Trans. Inf. Syst. Secur.*, *2*(1), 105–135. https://doi.org/10.1145/300830.300839

Sandhu, R. S. (1993). Lattice-based access control models. *Computer*, *26*(11), 9–19. https://doi.org/10.1109/2.241422

Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-Based Access Control Models. *Computer*, *29*(2), 38–47. https://doi.org/10.1109/2.485845

Sandhu, R. S., & Samarati, P. (1994). Access Control: Principles and Practice. *IEEE Communications Magazine*, *32*(9), 40 – 48. https://doi.org/10.1109/35.312842

Saxena, U. R., & Alam, T. (2022). Role based access control using identity and broadcast based encryption for securing cloud data. *JOURNAL OF COMPUTER VIROLOGY AND HACKING TECHNIQUES*, *18*(3), 171–182. https://doi.org/10.1007/s11416-021-00402-1

Schaad, A., Lotz, V., & Sohr, K. (2006). A Model-Checking Approach to Analysing Organisational Controls in a Loan Origination Process. *Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies*, 139–149. https://doi.org/10.1145/1133058.1133079

Schlegelmilch, J., & Steffens, U. (2005). Role Mining with ORCA. *Proceedings of the Tenth ACM Symposium on Access Control Models and Technologies*, 168–176. https://doi.org/10.1145/1063979.1064008

Shahen, J., Niu, J., & Tripunitara V, M. (2021). Cree: A Performant Tool for Safety Analysis of Administrative Temporal Role-Based Access Control (ATRBAC) Policies. *IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING*, *18*(5), 2349–2364. https://doi.org/10.1109/TDSC.2019.2949410

Shin, D., Ahn, G.-J., Cho, S., & Jin, S. (2003). On Modeling System-Centric Information for Role Engineering. *Proceedings of the Eighth ACM Symposium on Access Control Models and Technologies*, 169–178. https://doi.org/10.1145/775412.775434

Sieger, T., Hurley, C. B., Fiser, K., & Beleites, C. (2017). Interactive Dendrograms: The R Packages idendro and idendr0. *JOURNAL OF STATISTICAL SOFTWARE*, *76*(10), 1–22. https://doi.org/10.18637/jss.v076.i10

Sneath, P. H. A. (1957). The application of computers to taxonomy. *Microbiology*, *17*(1), 201–226.

Sokal, R. R. (1958). A statiscal method for evaluating systematic relationships. *Univ Kans Sci Bull*, *38*, 1409–1438.

Sokal, R. R., Sneath, P. H. A., & others. (1963). Principles of numerical taxonomy. *Principles of Numerical Taxonomy.*

Sun, W., Su, H., & Liu, H. (2019). Role-Engineering Optimization with Cardinality Constraints and User-Oriented Mutually Exclusive Constraints. *INFORMATION*, *10*(11). https://doi.org/10.3390/info10110342

Tadonki, C., Meyer, F., & Irigoin, F. (2014). Dendrogram Based Algorithm for Dominated Graph Flooding. *Procedia Computer Science*, *29*, 586–598. https://doi.org/https://doi.org/10.1016/j.procs.2014.05.053

Thomas, R. K., & Sandhu, R. S. (1994). Conceptual foundations for a model of task-based authorizations. *Proceedings The Computer Security Foundations Workshop VII*, 66–79. https://doi.org/10.1109/CSFW.1994.315946

Trnecka, M., & Trneckova, M. (2020). An incremental algorithm for the role mining problem. *COMPUTERS & SECURITY*, *94*. https://doi.org/10.1016/j.cose.2020.101830

Vaidya, J., Atluri, V., & Guo, Q. (2007). The Role Mining Problem: Finding a Minimal Descriptive Set of Roles. *SACMAT'07: PROCEEDINGS OF THE 12TH ACM SYMPOSIUM ON ACCESS CONTROL AND TECHNOLOGIES*, 175–184.

Vaidya, J., Atluri, V., & Guo, Q. (2010). The Role Mining Problem: A Formal Perspective. *ACM TRANSACTIONS ON INFORMATION AND SYSTEM SECURITY*, *13*(3). https://doi.org/10.1145/1805974.1895983

Vaidya, J., Atluri, V., Guo, Q., & Adam, N. (2008). Migrating to Optimal RBAC with Minimal Perturbation. *Proceedings of the 13th ACM Symposium on Access Control Models and Technologies*, 11–20. https://doi.org/10.1145/1377836.1377839

Vaidya, J., Atluri, V., Guo, Q., & Lu, H. (2009). Edge-RMP: Minimizing administrative assignments for role-based access control. *JOURNAL OF COMPUTER SECURITY*, *17*(2), 211–235. https://doi.org/10.3233/JCS-2009-0341

Vaidya, J., Atluri, V., Guo, Q., & Lu, H. (2010). Role Mining in the Presence of Noise. In S. Foresti & S. Jajodia (Eds.), *DATA AND APPLICATIONS SECURITY AND PRIVACY XXIV, PROCEEDINGS* (Vol. 6166, p. 97+).

Vaidya, J., Atluri, V., & Warner, J. (2006). RoleMiner: Mining Roles Using Subset Enumeration. *Proceedings of the 13th ACM Conference on Computer and Communications Security*, 144–153. https://doi.org/10.1145/1180405.1180424

Vaidya, J., Atluri, V., Warner, J., & Guo, Q. (2010). Role Engineering via Prioritized Subset Enumeration. *IEEE Trans. Dependable Secur. Comput.*, *7*(3), 300–314. https://doi.org/10.1109/TDSC.2008.61

Vayssieres, M., & Plant, R. E. (1998). *Identification of Vegetation State and-transition Domains in California's Hardwood Rangelands*.

Xu Dongkuan and Tian, Y. (2015). A Comprehensive Survey of Clustering Algorithms. *Annals of Data Science*, *2*(2), 165–193. https://doi.org/10.1007/s40745-015-0040-1

Yan, Z., Wu, Q., Ren, M., Liu, J., & Liu Shaowu and Qiu, S. (2019). Locally private Jaccard similarity estimation. *CONCURRENCY AND COMPUTATION-PRACTICE & EXPERIENCE*, *31*(24, SI). https://doi.org/10.1002/cpe.4889

Yang, L., & Li, C. (2023). Identification of Vulnerable Lines in Smart Grid Systems Based on Improved Agglomerative Hierarchical Clustering. *IEEE ACCESS*, *11*, 13554–13563. https://doi.org/10.1109/ACCESS.2023.3243806

Yang, Z., Yang, L., Luo, X., Ma, L., Kou, B., & Zhang, K. (2013). Model of Domain based RBAC and Supporting Technologies. *JOURNAL OF COMPUTERS*, *8*(5), 1220–1229. https://doi.org/10.4304/jcp.8.5.1220-1229

Yeh, C.-C., & Yang, M.-S. (2016). A Generalization of Rand and Jaccard Indices with Its Fuzzy Extension. *INTERNATIONAL JOURNAL OF FUZZY SYSTEMS*, *18*(6, SI), 1008–1018. https://doi.org/10.1007/s40815-016-0263-0

Zarnett, J., Tripunitara, M., & Lam, P. (2010). Role-Based Access Control (RBAC) in Java via Proxy Objects Using Annotations. *Proceedings of the 15th ACM Symposium on Access Control Models and Technologies*, 79–88. https://doi.org/10.1145/1809842.1809858

Zhang, D., Ramamohanarao, K., & Ebringer, T. (2007). Role Engineering Using Graph Optimisation. *Proceedings of the 12th ACM Symposium on Access Control Models and Technologies*, 139–144. https://doi.org/10.1145/1266840.1266862

Zhang, D., Ramamohanarao, K., Ebringer, T., & Yann, T. (2008). Permission Set Mining: Discovering Practical and Useful Roles. *2008 Annual Computer Security Applications Conference (ACSAC)*, 247–256. https://doi.org/10.1109/ACSAC.2008.21

Zhang, Y., He, J., & Zhao, B. (2015). The Dynamic Access Control Model for Cloud Web Based on Repeated-game Theory. *International Journal of Grid and Distributed Computing*, *8*, 27–36. https://doi.org/10.14257/ijgdc.2015.8.4.03

Zheng, L., Li, T., & Ding, C. (2014). A Framework for Hierarchical Ensemble Clustering. *ACM TRANSACTIONS ON KNOWLEDGE DISCOVERY FROM DATA*, *9*(2). https://doi.org/10.1145/2611380

Zhu, N., Cai, F., He, J., Zhang, Y., Li, W., & Li, Z. (2019). Management of access privileges for dynamic access control. *CLUSTER COMPUTING-THE JOURNAL OF NETWORKS SOFTWARE TOOLS AND*, *22*(4), S8899–S8917. https://doi.org/10.1007/s10586-018-2018-1

Zou, H. (2020). Clustering Algorithm and Its Application in Data Mining. *Wireless Personal Communications*, *110*(1), 21–30. https://doi.org/10.1007/s11277-019-06709-z

ÖZGEÇMİŞ

Kişisel Bilgiler		
Adı Soyadı:	Osman DURDAĞ	
Biyografi		
Erzurum İbrahim Hakkı Fen Lisesinden mezun olduktan sonra lisans eğitimini Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği bölümünden mezun olarak tamamlamıştır. Öğrenim hayatı boyunca lisans dersleri, yapmış olduğu stajlar ve bireysel çalışmaları doğrultusunda birçok projede aktif bir şekilde yer alarak kendini geliştirmeye ve teknik-teorik açıdan ilerlemeye özen göstermiştir. Edinmiş olduğu bilgileri fırsat buldukça çevresine aktarmaya çalışarak, her zaman beğenilen ve takdir edilen anlatım ve aktarım yeteneklerini doğal olarak geliştirmiştir. Yaklaşık 2 yıl özel sektörde çalıştıktan sonra akademik yolculuğuna devam etmeye karar vermiştir. Hâli hazırda Atatürk Üniversitesi Mühendislik Fakültesi Yazılım Mühendisliği Bölümünde Araştırma Görevlisi olarak görev almaktadır. Akademik yolculuğundaki hayali gerçek bir bilim insanı olmak ve bilimin, insanlığın ve insanın sahip olması gereken vicdanın ışığından ayrılmayarak ilerlemek ve bu konuda içinde bulunduğu topluma ışık olmaya çalışmaktır. Her geçen gün var olan yeteneklerini geliştirmek, yeni yetenekler ve bilgiler edinmeye çalışmak ve bu doğrultuda kendine, içinde bulunduğu topluma ve ülkesine faydalı bir birey olma yolunda gayret göstermek inançları, prensipleri ve amaçları arasındadır.		
Deneyim		
Araştırma Görevlisi:	Atatürk Üniversitesi	Ekim 2021 – Devam Ediyor
Solutions Architect:	Quorion Software Center	Haziran 2021 – Eylül 2021
Web Developer:	Quorion Software Center	Kasım 2020 – Haziran 2021
Full Stack Developer:	Teknar Teknoloji	Eylül 2019 – Kasım 2019
Stajyer:	Unicrow	Şubat 2019 – Nisan 2019
Stajyer:	KTÜ – BAUM	Temmuz 2018 – Ağustos 2018
Eğitim		
Yüksek Lisans:	Atatürk Üniversitesi, Bilgisayar Mühendisliği	
Lisans:	Karadeniz Teknik Üniversitesi, Bilgisayar Mühendisliği	
Lise:	Erzurum İbrahim Hakkı Fen Lisesi	
Yabancı Dil Bilgisi		
İngilizce:	B1	