T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KURAL TABANLI ŞÜPHELİ İŞLEM ÖNLEME SİSTEMLERİNDE KULLANILMAK ÜZERE ÇİZGE VERİTABANI MODELİ ÖNERİSİ

YÜKSEK LİSANS TEZİ

Bahadır Esad DEMİR

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

OCAK 2024

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KURAL TABANLI ŞÜPHELİ İŞLEM ÖNLEME SİSTEMLERİNDE KULLANILMAK ÜZERE ÇİZGE VERİTABANI MODELİ ÖNERİSİ

YÜKSEK LİSANS TEZİ

Bahadır Esad DEMİR

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Dr.Öğr.Üyesi Veysel Harun ŞAHİN

OCAK 2024

Bahadır Esad Demir tarafından hazırlanan "Kural Tabanlı Şüpheli İşlem Önleme Sistemlerinde Kullanılmak Üzere Çizge Veritabanı Modeli Önerisi" adlı tez çalışması 18.01.2024 tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı :	**Dr. Öğr. Üyesi Veysel Harun ŞAHİN**(Danışman) ..... Sakarya Üniversitesi	
Jüri Üyesi :	**Dr. Öğr. Üyesi İsmail ÖZTEL** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Dr. Öğr. Üyesi Soydan SERTTAŞ** Dumlupınar Üniversitesi	.....

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "Kural Tabanlı Şüpheli İşlem Önleme Sistemlerinde Kullanılmak Üzere Çizge Veritabanı Modeli Önerisi" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(18/01/2024).

(imza)

Bahadır Esad Demir

Eşime ve aileme

TEŞEKKÜR

Öncelikle beraber yaptığımız bu çalışmadaki yardımları, yol göstericiliği ve sabrı için değerli danışman hocam Veysel Harun ŞAHİN'e, süreçte yardımlarını esirgemeyen ve fedakarlık yapan değerli eşim Şeyma DEMİR'e teşekkürlerimi sunarım. Ayrıca bu çalışmaya imkan sağladıkları için Architecht Bilişim Sistemlerine teşekkür ederim.

Bahadır Esad DEMİR

İÇİNDEKİLER

Sayfa

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ v
TEŞEKKÜR ix
İÇİNDEKİLER xi
KISALTMALAR xiii
SİMGELER xv
TABLO LİSTESİ xvii
ŞEKİL LİSTESİ xix
ÖZET xxi
SUMMARY xxiii
1. GİRİŞ 1
2. LİTERATÜR ARAŞTIRMASI 3
3. ŞÜPHELİ İŞLEM 5
3.1. Şüpheli İşlem Nedir ? 5
3.2. Şüpheli İşlem Türleri 6
3.2.1. Kimlik hırsızlığı 6
3.2.2. Kart dolandırıcılığı 6
3.2.3. Çek dolandırıcılığı 6
3.2.4. Balık avı (Phishing) 6
3.2.5. Sosyal mühendislik 6
3.2.6. ATM dolandırıcılığı 6
3.2.7. İnternet bankacılığı dolandırıcılığı 6
3.2.8. Fon transferi dolandırıcılığı 7
3.2.9. Çevrimiçi alışveriş dolandırıcılığı 7
3.2.10. Mevduat dolandırıcılığı 7
3.3. Kredi Kartı Şüpheli İşlemleri 7
3.3.1. Kredi kartı bilgilerinin çalınması 7
3.3.2. Sahte alışveriş işlemleri 7
3.3.3. İzinsiz kredi kartı hesap açma 7
3.3.4. Kart bilgilerinin çalınması ve skimming 8
3.3.5. İkinci el kredi kartı piyasası 8
3.3.6. Kredi kartı dolandırıcılığı içeren e-postalar 8
3.4. Şüpheli İşlem Sistemleri 8
3.4.1. Şüpheli işlem tespit sistemleri 8
3.4.2. Şüpheli işlem önleme sistemleri 8
3.4.3. Şüpheli işlem önleme sistemleri türleri 9
3.5. Örnek Bir Kural Tabanlı Şüpheli İşlem Önleme Sistemi 11
4. ÇİZGE VERİTABANLARI 13
4.1. Çizge Veritabanları Nedir ? 13
4.2. İlişkisel Veritabanlarından Farkları Nelerdir ? 13
4.3. Çizge Veritabanı Türleri 14
4.3.1. Veri modeline göre çizge veritabanı türleri 14

4.4. Neo4j 16
4.5. Cypher Sorgu Dili 17
5. MATERYAL 19
5.1. Veri Seti 19
5.2. Verinin Hazırlanması 20
6. YÖNTEM 25
6.1. Neo4j Uygulamasına Verilerin Aktarılması 25
6.1.1. Indexlerin oluşturulması 25
6.1.2. CSV türün deki dosyayı içe aktararak model oluşturulması 25
6.2. Önerilen Model 28
6.3. Örnek Sorgular 28
7. SONUÇ 33
KAYNAKLAR 35
ÖZGEÇMİŞ 39

KISALTMALAR

SSD : Kareler toplamı sapmaları

SSW : Kareler toplamı farkları

SİMGELER

f	: Fonksiyon
∨	: Veya
∧	: Ve
≥	: Büyük eşit
≤	: Küçük eşit
$g1$	: Veri setinin 25. ve 75. yüzdelik dilimleri arasındaki mesafe
σ	: Standart Sapma
$\bar{x}$	: Aritmetik ortalama
Σ	: Toplam
μ	: Popülasyon ortalaması

TABLO LİSTESİ

Sayfa

Tablo 3.1. Şüpheli işlem tespit ve önleme sistemleri arasındaki farklar. 9
Tablo 3.2. İşlem tiplerinin Pos ve Kart sahipliklerine göre belirlenmesi. 11
Tablo 4.1. Çizge ve İlişkisel veritabanları arasındaki temel farklar. 14
Tablo 5.1. BankSim veri kümesindeki şüpheli olan ve olmayan işlem sayıları. 19
Tablo 5.2. BankSim veri kümesindeki kolon detayları. 19
Tablo 5.3. Veri zenginleştirilmesi yapıldıktan sonra veri kümesinin durumu. 23
Tablo 6.1. Tutar gruplarının aralıkları. 26
Table 6.2. Belirli bir kategorideki şüpheli işlemlerin tutar aralıkları. 30
Tablo 6.3. Tutar grubuna göre şüpheli işlemlerin en çok yapıldığı iş yerleri. 30

ŞEKİL LİSTESİ

Sayfa

Şekil 3.1. İşlem tiplerine göre şüpheli işlem kontrol sisteminin çağırılması. 12
Şekil 4.1. Kaynak tanımlama çerçevesi model gösterimi. 15
Şekil 4.2. Etiketlenmiş özellik çizgesi model örneği. 16
Şekil 5.1. Tutar bilgisinin gruplanmadan önce oluşturulan histogram grafiği. 21
Şekil 5.2. Gruplama yapıldıktan sonra tutar bilgisinin dağılım grafiği. 23
Şekil 6.1. Tutar grubu minimum ve maksimum değer bilgileri. 28
Şekil 6.2. Önerilen çizge veritabanı modeli. 28
Şekil 6.3. Neo4j uygulamasının arayüzünde veri setimizdeki bazı alanların genel bir görünümü. 29
Şekil 6.4. Şüpheli işlemleri olan bir müşterinin şüpheli olduğu netleşmemiş diğer işlemleri. 29

KURAL TABANLI ŞÜPHELİ İŞLEM ÖNLEME SİSTEMLERİNDE KULLANILMAK ÜZERE ÇİZGE VERİTABANI MODELİ ÖNERİSİ

ÖZET

Finansal dünyada şüpheli işlemlerin yöntemlerinde ve sayılarında her geçen gün artış olmaktadır. Kredi kartı işlemlerindeki şüpheli işlemler de bu yöntemlerden biridir. Kredi kartı şüpheli işlemleri, kredi kartı işlemlerinde yetkisiz veya aldatıcı yöntemlerin kullanıldığı ve başkasının kredi kartı bilgilerini kullanarak finansal kazanç elde etmeye çalışılan sahtekarlık faaliyetlerine denir. Artan kredi kartı işlemlerinin sayısıyla birlikte birçok banka ve kuruluş, sahtekarlık vakalarını tespit etmek ve önlemek amacıyla sistemler kullanmakta ve bu sistemleri geliştirecek çalışmalar yapmaktadır. Son zamanlarda, bu konuyla ilgili makine öğrenme algoritmalarını kullanan birçok yöntem önerilmiş olsada, kural tabanlı sistemler halen tercih edilmektedir. Kural tabanlı sahtekarlık tespiti ve önleme sistemlerinde meydana gelen vakalar işlem anında veya sonrasında analiz edilir ve işlemin şüpheli olup olmadığına bağlı olarak sistemdeki yetkililerin incelemesine yardımcı olmak ve gerektiğinde işlemleri engelleyebilmek için kurallar yazılır. Bu verilerin sonradan incelenerek analiz edilmesi sayesinde, sahtekarlık işlemleri ve tekrar eden şüpheli işlem desenleri tespit edilebilir ve kurallar bu doğrultuda zenginleştirilebilir. Kural tabanlı sistemlerde farklı veritabanı teknolojileri kullanılmaktadır. İlişkisel veritabanları ve doküman tabanlı veritabanları bu teknolojilere örnek olarak verilebilir. Bu sistemlerde asıl önemli olan görülemeyen, yapısal olarak belirtilmemiş ilişki ağlarını tespit edebilmektir. Diğer veritabanı teknolojilerine alternatif olarak çizge veritabanları da bu alanda kullanılmaktadır. Bu çalışmada öncelikle şüpheli işlemlerin nelere dendiğinden ve bu şüpheli işlemlerin türlerinden bahsedilmiş ve şüpheli işlem tespit ve önleme sistemlerinin özellikleri aktarılmıştır. Ardından çizge veritabanlarının özellikleri açıklanmış ve diğer veritabanı teknolojilerinden farklarına değinilmiştir. Günümüzde sektörde kullanılan çizge veritabanı uygulamalarından örnekler verilmiş ve bu çalışmada da kullanılan Neo4j uygulamasından detaylı bahsedilmiştir. Kaggle üzerinde paylaşılan bir bankanın kredi kartı işlemlerinin örnekleminden sentetik olarak üretilen bir kredi kartı sahtekarlık veri kümesini kullanarak bu veri seti açıklanmış, Neo4j modeline eklemeden önce incelemelerde faydası olacak şekilde tutar bilgisi aralıklı tutar gruplarına dönüştürülmüştür. Bu veri seti için bir model önerilmiş ve bu modele göre Neo4j uygulamasına aktarım gerçekleştirilmiştir. Son olarak önerilen model kullanılarak Neo4j üzerinde örnek sorgular gerçekleştirilmiştir. Bu sorguların çıktılarını analiz ederek kural tabanlı sistemlerdeki kural yazımlarında ek bir metrik ve kural sunulabileceği gösterilmiştir.

A GRAPH DATABASE MODEL PROPOSAL FOR USE IN RULE BASED FRAUD TRANSACTION PREVENTION SYSTEMS

SUMMARY

In the financial world, there is a continuous increase in both the methods and numbers of suspicious transactions. Various methods, such as identity theft, check fraud, ATM fraud, online banking fraud, online shopping fraud, and credit card fraud, contribute to this rise. This study specifically focuses on credit card fraud among these fraudulent methods. Credit card fraud involves fraudulent activities in credit card transactions where unauthorized or deceptive methods are employed. These activities encompass the theft of credit card information, the unauthorized creation of credit cards, the utilization of second-hand credit cards, and methods involving social engineering.

With the increasing number of credit card transactions every day, many banks and organizations are developing systems to detect and prevent fraudulent cases. They integrate these systems and conduct efforts to enhance them. Systems developed to counteract suspicious transactions can be categorized into two main types: suspicious transaction detection systems and prevention systems.

Suspicious transaction detection systems focus on identifying both past and ongoing suspicious transactions. On the other hand, suspicious transaction prevention systems not only aim to detect these transactions but also adopt an approach to prevent and block them in advance.

While suspicious transaction detection systems serve as a security layer, prevention systems function as a security firewall, aiming to both identify and proactively thwart suspicious activities.

Although various methods, such as machine learning, analytics, data mining, behavior analysis, and artificial intelligence, have been proposed for suspicious transaction prevention systems in recent times, many banks and organizations still prefer and use rule-based systems. In rule-based fraud detection and prevention systems, incidents are analyzed, and rules are written by authorized personnel based on whether the transaction is considered suspicious or not.

These processes allow the prevention of transactions based on predefined rules. Additionally, through post-analysis, previously unnoticed methods can be detected, and rules can be enriched accordingly. The study also shares an example of a rule-based suspicious transaction prevention system model.

These systems utilize relational databases, document-based databases, and recently popular graph databases as their database technology. Graph databases, by focusing on relationships, provide an advantage in revealing unseen networks in suspicious transaction analyses. In contrast to relational databases, where entities are stored in tables as rows and columns and results are obtained by combining different tables using 'join' operations during query time, graph databases store entities in a way where nodes are physically connected by edges.

Unlike relational databases, there is no need for a join operation at query time in graph databases because the desired relationships are already represented as physical entities between nodes. The data size for graph databases does not pose a problem due to this structure, making them effective in handling large datasets. Consequently, they have started to be preferred in social networks, behavior analysis, and suspicious transaction detection and prevention systems.

One of the most popular graph database applications is Neo4j. Neo4j assists users in conducting analyses and comprehending them through its user-friendly interface. The Cypher query language, developed by the same team for Neo4j, is used in Neo4j applications. This query language, specially designed for graph databases, allows for the creation of shorter and more understandable queries when querying complex relationships.

In this study, the BankSim dataset, obtained from Kaggle and containing transaction data related to credit cards, has been used. The BankSim dataset is synthetically generated based on fraud models in customer payments for a bank and is shared for academic research purposes. To ensure data security, real personal information is not present in the dataset. The dataset includes approximately 600,000 records, with 7,200 of them being suspicious transactions. The fields in the dataset can be listed as follows: step, customer, customer age, customer gender, customer zipcode, merchant, merchant zipcode, transaction category, transaction amount, and information about whether the transaction is suspicious or not.

Considering the large volume of data in the dataset, it is not necessary to know the exact transaction amount for the checks we will perform. Instead, dividing transaction amounts into more functional and defined groups will make our model and query outputs more meaningful. However, the transaction amount information in our dataset is concentrated within a specific range. When grouping asymmetric and skewed data, the aim is to minimize this skewness by using some mathematical formulas. Therefore, for the BankSim dataset, it was determined that the dataset needs to be divided into 21 groups using Doane's Rule and Jenks Natural Breaks formulas. To ensure balance in these groups, meaning that the transactions are distributed in a way that minimizes skewness, even if not equally in each group, the group intervals were determined. These amount groups were added to the original BankSim dataset as a new column called "Amount Group."

Following this dataset, a model has been proposed for use in graph databases. In the model, workplaces, customers, transactions, categories, and amount groups are represented as separate nodes. Relationships between these nodes have also been specified to accurately convey the relevant dataset. Other fields in the dataset are directly represented as sub-properties of the nodes in the model. After these operations, the dataset has been imported into the Neo4j desktop application using the Cypher language in CSV format. Index definitions were primarily made for nodes related to workplaces, customers, categories, and amount groups, considering that queries would be performed on these nodes to ensure better performance. Subsequently, using the Cypher query language, the available data nodes and relationship definitions were imported according to the proposed model. Code blocks for these sections are detailed within the code. Once the dataset was transferred to the database in line with the proposed model, various queries were executed in the Neo4j application. The results of these queries were analyzed as they could be used as rules for credit card fraud prevention systems. Rule functions were also presented based on the outputs.

In conclusion, a graph database model has been proposed for examining suspicious transactions in rule-based credit card fraud detection and prevention systems within the Neo4j application. The model utilizes the BankSim dataset. The Cypher commands executed in the Neo4j application have generated sample outputs for transaction reviews in the rule-based system. Additionally, suggestions have been made to add new rules based on the analysis results. The aim is to demonstrate the method and benefits of using Graph Databases as an alternative to relational databases in such systems.

1. GİRİŞ

Şüpheli işlem türlerinden biri olan kredi kartı şüpheli işlemlerinin son dönemde e-ticaret alışverişlerinin ve pandemi sonrası kredi kartı kullanımlarının yaygınlaşması ile artış göstermesi, dolandırıcıların bu işlemleri hedef almasına neden oldu. Bir çok kurum ve kuruluş gibi bankalar da bu dolandırıcıları engellemek ve şüpheli işlemleri önlemek adına çeşitli sistemler kullanmakta ve bu sistemleri geliştirmeye çalışmaktadırlar. Kural tabanlı sistemler bankaların yaygın olarak kullanmakta olduğu sistemlerden biridir. Bu çalışmada kredi kartı işlemlerinde gerçekleşen dolandırıcılık vakalarını önlemeye çalışan kural tabanlı sistemlere veritabanı seviyesinde yeni bir teknoloji olan çizge veritabanları ile bir model önerilmektedir. Çalışmada gerçek müşteri verisinden sentetik olarak üretilmiş bir veri seti kullanılmıştır. Kural tabanlı bir şüpheli işlem önleme sisteminde kullanılmak üzere, bu veri setinin çizge veritabanına nasıl bir model ile aktarılabileceği ve nasıl çıktılar elde edilebileceğine değinilmiştir.

2. LİTERATÜR ARAŞTIRMASI

Şüpheli işlemler ve dolandırıcılık tespiti üzerine farklı çalışmalar bulunmaktadır. Mangala ve arkadaşları, bankacılık sektöründeki şüpheli işlemler hakkında sistematik bir literatür taraması yapmışlardır [1]. Hilal ve arkadaşları, finansal şüpheli işlemleri tespit edebilmek için uygulanan çeşitli anormallik tespiti tekniklerini araştırmış ve bu teknikleri detaylı bir şekilde çalışmalarında aktarmışlardır [2]. Abdallah ve arkadaşları, şüpheli işlem tespit sistemlerinin performansını etkileyen sorunlar hakkındaki araştırmaları ile bu sorunlara genel bir bakış sunmuşlardır [3]. Pourhabibi ve arkadaşları, şüpheli işlemlerin tespiti için çizge tabanlı anormallik tespiti yaklaşımlarına dair sistematik bir literatür taraması yapmışlardır [4].

Correa Bahnsen ve arkadaşları, kredi kartı şüpheli işlemlerinin tespiti için işlem birleştirme yöntemini genişleterek işlemlerin ayırt edici özelliklerini ortaya çıkarma konusuna odaklanmışlardır [5]. Carcillo ve arkadaşları, kredi kartı işlemlerindeki dolandırıcılıkların tespit hassasiyetini artırmak için denetimli ve gözetimsiz teknikleri birleştirerek birlikte kullanımını sağlayan bir hibrit teknik sunmuşlardır [6]. Ceronmani Sharmila ve arkadaşları, yerel aykırı faktör ve izolasyon ormanı algoritmasına kullanarak kredi kartı şüpheli işlemlerini tespit etmek için bir yaklaşım oluşturmuşlardır [7]. Huang ve arkadaşları, hem işlem ağ bilgilerini hem de ağdaki varlıkların özellik bilgilerini kullanan CoDetect adlı bir finansal dolandırıcılık tespit çerçevesi geliştirmişlerdir [8].

Vorobyev ve Krivitskaya, kural temelli dolandırıcılık tespiti için yüksek sayıda yanlış pozitifle başa çıkabilmek adına bir kural oluşturma çerçevesi önermişlerdir [9]. Bu çerçeve ile karar ağacı, rastgele orman ve gradyan artırma algoritmalarını kullanarak otomatik kurallar oluşturmuşlardır. Gianini ve arkadaşları, dolandırıcılık tespiti için kullanılan kuralları oluşturmak ve yönetmek için oyun teorisini kullanan bir çalışma yapmışlardır [10].

Ozcan ve Genc, makine öğrenimi bakış açısı ile şüpheli işlemlerin tespitine odaklanmışlardır [11]. Bir otokodlayıcı ve sınıflandırıcıdan oluşan derin öğrenme modeli önermişlerdir. Yuksel ve arkadaşları, kredi kartı dolandırıcılığı tespiti için

boyut indirgeme amacıyla komşuluk bileşen analizi kullanarak yeni bir yaklaşım önermişlerdir [12]. Van Vlasselar ve arkadaşları, müşterilerin ve ağların birkaç özelliğini birleştiren APATE adlı bir kredi kartı dolandırıcılık tespiti yaklaşımı oluşturmuşlardır [13].

Jing ve arkadaşları, çizge tabanlı yarı gözetimli dolandırıcılık tespit çerçevesi oluşturmuşlardır. Yapısal verileri çizge biçimine çevirmişler ve GraphSAGE algoritmasını kullanmışlardır [14]. Prusti ve arkadaşları bir çizge veritabanından yardım alarak dolandırıcılık tespiti sistemi önermişlerdir. Çizge modelinin özelliklerini Neo4j uygulamasını kullanarak çıkarmışlardır [15]. Molloy ve arkadaşları, çapraz kanallardan gerçekleştirilen dolandırıcılıkların tespiti üzerine odaklanmışlardır. Yaklaşımlarında tespitlere yardımcı olmak için çizge yöntemini kullanmışlardır [16]. Kurshan ve arkadaşları, şüpheli işlem tespit sistemleri için çizge yönteminin sistemlere uygulanmasında ne tür zorluklar yaşanabileceğini incelemişlerdir [17]. Çavşi Zaim ve arkadaşları, şüpheli işlem tespit sistemlerinde çizge veritabanı ve makine öğrenme tekniklerinin beraber kullanımına odaklanmışlardır [18]. Van Belle ve arkadaşları, şüpheli işlem tespiti açısından GraphSAGE ve Hızlı Yükseltilmiş Çizge Temsil Öğrenme yeteneklerini değerlendiren indüktif çizge temsil öğrenme tekniklerini de kullanan yöntemlere dair bir inceleme yapmışlardır [19]. Henderson, finansal şüpheli işlemlerin tespitinde kullanmak için çizge veritabanlarının özelliklerini ve faydalarını aktaran bir çalışma yapmıştır [20].

Bu tez kapsamında ise örnek bir veri setinin çizge veritabanlarına aktarımı konusunda model önerisi yapılmıştır. Ayrıca veri setinin zenginleştirilmesi için tutar bilgisinin aralıklı tutar grupları olarak kullanılması sağlanmış, bunu yaparkende normalde coğrafi haritalandırmalarda kullanılan Jenk'in Doğal Kırılım algoritması veri setine uygulanmıştır.

3. ŞÜPHELİ İŞLEM

Bu bölümde genel olarak şüpheli işlemler ile alakalı bilgi verilip, bankacılık özelinde şüpheli işlemler neleri kapsar, nelere şüpheli işlem denilir bunlara değinilecektir. Ayrıca şüpheli işlem önleme ve tespit sistemlerinden bahsedilip kurumların bu sistemlerde nasıl yöntemler kullandıkları aktarılacaktır. Tezimizin kapsamındaki kural tabanlı sistemlere detaylı açıklanarak örnek bir sistemin modeli paylaşılacaktır.

3.1. Şüpheli İşlem Nedir ?

Şüpheli işlemler, genellikle para gibi bir tür maddi yarar elde etmek amacıyla uygulanan dolandırıcılık olarak tanımlanabilir. Tek bir kişi veya bir grup insan tarafından gerçekleştirilebilir. Sigorta şüpheli işlemleri ve finansal şüpheli işlemler gibi çeşitli işlem türleri bulunmaktadır. Bu tez kapsamında, finansal şüpheli işlemlere değinilecektir. Finansal şüpheli işlemler ile alakalı çalışmalara dair detaylı bir inceleme Mangala ve arkadaşlarının yaptığı [1] makalesinde bulunabilir.

Şüpheli işlemlerin tespiti, bankacılık sektörü gibi diğer sektörlerde olduğu gibi kritik bir öneme sahiptir. Günümüzde finansal sistemler için çeşitli şüpheli işlem tespiti yaklaşımları bulunmaktadır [2]. Ancak, gelişen teknolojiler, artan işlem sayıları, çeşitli işlem kanalları ve dolandırıcılar tarafından geliştirilen yeni yöntemler, sahtekarlığı tespit etmeyi ve önlemeyi daha zor hale getirmektedir. Bu nedenle kurumlar ve çalışanları, bu sistemler üzerinde çalışmaya devam etmeli, yeni yöntemler geliştirmeye çalışmalı, yeni yaklaşımlar denemeli ve yeni dolandırıcılık yöntemleri karşısında sahtekarlık tespiti ve önleme sistemlerinin etkinliğini artıracak yeni yollar bulmalıdır.

Finansal şüpheli işlem tespiti sistemi farklı yöntemler ve teknolojiler kullanılarak geliştirilebilir. Günümüzde bankacılık sektöründe kural tabanlı yöntemler yaygın bir şekilde kullanılmaktadır [11]. Kural tabanlı sistemler hızlı bir şekilde yapılandırılabilir ve çalıştırılabilir. Kural tabanlı sistemlerin önemli bir yönü insan kaynağıdır. Nitelikli çalışanlar bu sistemler için kuralları yazmaktadır. Günümüzde, makine öğrenimi teknikleri ve çizge veritabanları da şüpheli işlem tespit sistemlerine entegre edilmektedir.

3.2. Şüpheli İşlem Türleri

Bankacılık sektöründe şüpheli işlem türleri oldukça çeşitli olabilir. Bu tür işlemler, finansal kurumların ve müşterilerin mali varlıklarını koruma amacıyla sürekli olarak göz önünde bulundurulmalı ve önlenmelidir.

3.2.1. Kimlik hırsızlığı

Dolandırıcılar, sahte kimlikler veya çalınmış kimlik bilgileri kullanarak sahte hesaplar açabilirler. Bu tür dolandırıcılık, kişisel bilgilerin çalınması ve kötü amaçlı kullanılmasını içerir.

3.2.2. Kart dolandırıcılığı

Kredi kartı veya banka kartı bilgilerini ele geçirerek, dolandırıcılar sahte işlemler yapabilirler. Bu, kart kopyalama, çalınmış kart bilgileri ile çevrimiçi alışveriş yapma veya ATM'lerden para çekme gibi şekillerde gerçekleşebilir.

3.2.3. Çek dolandırıcılığı

Sahte çeklerin yazılması veya çalınan çeklerin kötüye kullanılması, bankalar ve müşteriler için bir tehdit oluşturabilir.

3.2.4. Balık avı (Phishing)

Sahte e-posta veya internet siteleri aracılığıyla, dolandırıcılar kişisel ve finansal bilgileri elde etmeye çalışırlar. Bu tür dolandırıcılık, kurbanları yanıltarak sahte bir finansal kuruluşa bilgi sağlamalarını sağlar.

3.2.5. Sosyal mühendislik

Dolandırıcılar, insanların güvenini kazanarak, telefonla veya yüz yüze iletişim kurarak kişisel bilgileri elde etmeye çalışabilirler.

3.2.6. ATM dolandırıcılığı

ATM'lere yerleştirilen sahte kart okuyucular veya kameralar aracılığıyla, kart bilgileri ve PIN kodları çalınabilir.

3.2.7. İnternet bankacılığı dolandırıcılığı

Kötü niyetli yazılımlar veya kimlik avı internet siteleri aracılığıyla, dolandırıcılar müşterilerin internet bankacılığı hesaplarını ele geçirebilirler.

3.2.8. Fon transferi dolandırıcılığı

Dolandırıcılar, sahte işlemler veya yanıltıcı bilgilerle banka hesapları aracılığıyla para transferlerini manipüle edebilirler.

3.2.9. Çevrimiçi alışveriş dolandırıcılığı

Kredi kartı bilgilerini ele geçirerek veya sahte ürünler satarak, dolandırıcılar çevrimiçi alışveriş dolandırıcılığı yapabilirler.

3.2.10. Mevduat dolandırıcılığı

Sahte mevduatlarla, banka hesaplarına para yatırma işlemleri yaparak, dolandırıcılar finansal kurumları yanıltabilirler.

3.3. Kredi Kartı Şüpheli İşlemleri

Şüpheli işlem türleri ile alakalı verilen genel bilgilerden sonra bu bölümde tez kapsamında odaklanacağımız kredi kartı şüpheli işlemlerinde gerçekleşen dolandırıcılık yöntemlerine değinilecektir.

3.3.1. Kredi kartı bilgilerinin çalınması

Dolandırıcılar, kredi kartı bilgilerini çeşitli yollarla ele geçirebilirler. Bu bilgiler, fiziksel kartın kopyalanması, çalınması veya çevrimiçi ortamda çalınması gibi yöntemlerle elde edilebilir. Ayrıca, kimlik avı veya kötü niyetli yazılımlar aracılığıyla da kredi kartı bilgileri toplanabilir.

3.3.2. Sahte alışveriş işlemleri

Kredi kartı bilgileri elde edildikten sonra, dolandırıcılar sahte alışveriş işlemleri gerçekleştirebilirler. Bu işlemler, çalınmış kart bilgileri ile online alışveriş yapma veya fiziksel mağazalarda kart kullanma şeklinde olabilir.

3.3.3. İzinsiz kredi kartı hesap açma

Dolandırıcılar, kişilerin kimliklerini çalarak sahte kredi kartı hesapları açabilirler. Bu tür dolandırıcılık, kişisel bilgilerin çalınması ve bu bilgilerin kullanılması ile gerçekleşir.

3.3.4. Kart bilgilerinin çalınması ve skimming

Dolandırıcılar, ATM'lerde veya ödeme terminallerinde kullanılan kart okuyucularına sahte cihazlar ekleyerek kart bilgilerini çalabilirler. Bu yöntem "skimming" olarak

adlandırılır ve genellikle kart okuyucunun üstüne yerleştirilen sahte bir cihazla gerçekleştirilir.

3.3.5. İkinci el kredi kartı piyasası

Kredi kartı bilgileri, siber suçlular arasında ticaret konusu olabilir. Çalınan bilgiler, siber pazarlarda satılır ve başka kişiler tarafından kötü amaçlı kullanılabilir.

3.3.6. Kredi kartı dolandırıcılığı içeren e-postalar

Dolandırıcılar, e-posta yoluyla kişilerden kredi kartı bilgilerini vermesini veya sahte banka veya kredi kartı kurumlarının internet sitelerine yönlendirmelerini isteyebilirler. Bu tür dolandırıcılık olaylarına "phishing" denir.

3.4. Şüpheli İşlem Sistemleri

Şüpheli işlem sistemlerinde temel olarak iki yaklaşım vardır. Şüpheli işlemi tespit etme ve şüpheli işlemi önleme. Şüpheli işlem tespit sistemleri ve önleme sistemleri genellikle birbirini tamamlayan ancak farklı odaklara sahip olan iki farklı güvenlik yaklaşımıdır.

3.4.1. Şüpheli işlem tespit sistemleri

Bu sistemlerin temel amacı, potansiyel olarak kötü niyetli veya yasa dışı işlemleri tespit etmektir. Yani, bir kez olmuş veya gerçekleşmekte olan şüpheli işlemleri belirlemeye odaklanır.

Sistem, önceki işlem verilerini analiz eder ve belirlenmiş şüpheli desenlere veya kriterlere uyan işlemleri belirler. Örneğin, büyük miktarda para transferi, belirli coğrafi konumlardan yapılan alışverişler veya tipik kullanıcı davranışlarından sapmalar gibi durumlar bu sistem tarafından belirlenebilir.

3.4.2. Şüpheli işlem önleme sistemleri

Bu sistemlerin ana amacı, şüpheli işlemleri tespit etmenin yanı sıra, bunları engellemek ve önceden önlemek için proaktif bir yaklaşım benimsemektir. Şüpheli işlem önleme sistemleri, tespit edilen potansiyel tehditlere karşı önleyici önlemler alır. Bu, ödeme işlemlerini durdurma, kullanıcı doğrulamasını güçlendirme veya riskli işlemleri engelleme gibi aksiyonları içerebilir.

İki sistem arasındaki farklar Tablo 3.1. de gösterilmiştir.

Tablo 3.1. Şüpheli işlem tespit ve önleme sistemleri arasındaki farklar.

	Tespit Sistemleri	Önleme Sistemleri
Zaman	Geçmiş işlemleri analiz eder.	Anlık veya gerçek zamanlı işlemlere müdahale eder.
Reaktif ve Proaktif Yaklaşım	Reaktif bir yaklaşım benimser, yani bir tehdit tespit edildikten sonra müdahale eder.	Proaktif bir yaklaşım benimser, potansiyel tehditleri önceden tespit etmeye ve önlemeye çalışır.
Güvenlik	Genellikle bir güvenlik katmanı olarak kullanılır.	İşlemi durdurma veya engelleme yetenekleri ile bir güvenlik duvarı işlevi görür.

3.4.3. Şüpheli işlem önleme sistemleri türleri

Şüpheli işlem önleme sistemlerinde farklı yaklaşımlar bulunmaktadır.

3.4.3.1. Analitik ve veri madenciliği tabanlı sistemler

Bazı sistemler, büyük veri analitiği ve veri madenciliği tekniklerini kullanarak anormal desenleri tespit etmeye odaklanır. Bu sistemler genellikle önceki işlem verilerini analiz eder ve alışılmadık aktiviteleri belirlemeye çalışır.

3.4.3.2. Davranış analizi tabanlı sistemler

Bu tür sistemler, kullanıcı davranışlarına dayanarak normal ve anormal aktiviteleri ayırt etmeye çalışır. Bir kullanıcının tipik alışkanlıklarından sapma durumunda uyarılar verebilir.

3.4.3.3. Makine öğrenimi ve yapay zeka tabanlı sistemler

Makine öğrenimi ve yapay zeka kullanımıyla güçlendirilen sistemler, zaman içinde öğrenme yeteneğine sahiptir. Bu sistemler, sürekli olarak yeni tehditleri tanımayı ve adapte olmayı amaçlar.

3.4.3.4. Çoklu katmanlı güvenlik sistemleri

Güvenlik, genellikle çok katmanlı bir yaklaşımla ele alınır. Birden fazla şüpheli işlem önleme sistemi ve güvenlik katmanı bir araya getirilerek daha etkili bir koruma sağlanır.

3.4.3.5. Kural tabanlı sistemler

Belirli kurallara dayanan sistemler, belirli durumları tanımlayan ve bu durumlar ortaya çıktığında uyarılar oluşturan veya işlemleri engellemeye yarayan kural tabanlı bir yaklaşımı benimser. Örneğin, belirli bir tutarın üzerindeki işlemleri kontrol etmek gibi veya bir grup kurala ait filtreye takılan işlemlerin engellendiği sistemler oluşturulabilir.

Bu çalışmada ilgileneceğimiz kısım kural tabanlı sistemlerdir. Bu sistemlerin diğer sistemlere göre öncelikli sağladığı faydalar aşağıda belirtilmiştir.

Bu sistemler kolay ayarlanabilir sistemlerdir. Kural tabanlı sistemler genellikle belirli kurallara dayandığından, ilgili ekipler istekleri doğrultusunda bu kuralları kolayca ayarlayabilirler. Bu, özel ihtiyaçlara ve sektör spesifikasyonlarına uyum sağlamayı kolaylaştırır.

Hızlı kurulum ve uygulanabilirliğe sahip sistemlerdir. Kural tabanlı sistemler, genellikle hızlı bir şekilde uygulanabilir ve kullanılabilir. Belirli kuralların tanımlanması ve uygulanması, daha karmaşık analitik modellerin geliştirilmesinden daha hızlı olabilir. Bir model eğitilerek bu modele göre işlemlerin engellendiği sistemlerde, önceki işlemleri analiz etmek ve buna göre özellişmiş bir model eğitmek gerekirken kural tabanlı sistemlerde sistemin ayağa kaldırıldığı andan itibaren belirtilen kurallara göre işlemler önlenmeye başlanabilir.

Net olan şüpheli durumlar için çok etkilidirler. Bazı işlemlerde engellenmesi gereken kurallar genellikle belirgin ve açıktır, bu nedenle belirli şüpheli durumları tespit etme konusunda kural tabanlı sistemler etkilidirler. Örneğin, çalındığı düşünülen paylaşılmış kredi kartı bilgileri mevcut ise sisteme bu bilgi verilerek bir kural tanımlandığında sistem etkili bir şekilde bu kartlardan gelecek işlemleri engelleyebilecektir.

Küçük ve orta ölçekli sistemlerde etkilidirler. Kural tabanlı sistemler, büyük veri setlerine ihtiyaç duymadan küçük veri setlerinde bile etkili olabilir. Bu, daha küçük ölçekli işletmeler veya belirli sektörler için avantajlı olabilir.

Kural tabanlı sistemlerin avantajlı olduğu durumların dışında bazı sınırlamaları da bulunmaktadır.

Bu sistemler genellikle belirli kurallara bağlı oldukları için esneklikleri sınırlı olabilir. Değişen tehditlerde hızlı bir şekilde inceleme yapılarak kural tanımlanması gerekmektedir.

Belirli kuralların katı uygulanması, yanlış pozitif veya yanlış negatif sonuçlara yol açabilir. Bu, gerçek olmayan şüpheli durumları tespit etme veya gerçek tehditleri gözden kaçırma riskini artırabilir.

Kural tabanlı sistemler, karmaşık desenleri ve ilişkileri tespit etme konusunda daha sınırlı olabilir. Bu, daha sofistike hile ve sahtekârlık girişimlerini tespit etmede zorlanabilirler. Görünen belirgin dolandırıcılıkları kolaylıkla engelleyebilirken, görünemeyen dolandırıcılık desenlerini tespit edemezler. Bu çalışmada kural tabanlı sistemlerin kısmen eksik olduğu bu tarz görünmeyen, belirgin olmayan dolandırıcılık yöntemlerinde kullanımına ek bir metrik ile katkı sağlamak için çizge veritabanlarının bu konudaki özellikleri anlatılacak ve bir model önerilerek, mevcut bir sisteme ek olarak hangi kuralların yazılabileceği önerilecektir.

3.5. Örnek Bir Kural Tabanlı Şüpheli İşlem Önleme Sistemi

Bu bölümde yapılacak çalışmanın kullanılabileceği ve bir banka tarafından geliştirilen Fraud tespit ve önleme sistemine genel bir bakış yapılacaktır. Bankanın kredi kartı ve pos işlemlerinde şüpheli işlem tespiti için kullandığı bu sistem aşağıdaki tablolarda anlatılmıştır. Kredi kartı ve pos işlemlerinde işlem tiplerinin nasıl belirlendiği hakkında Tablo 3.2'de detay verilmiştir.

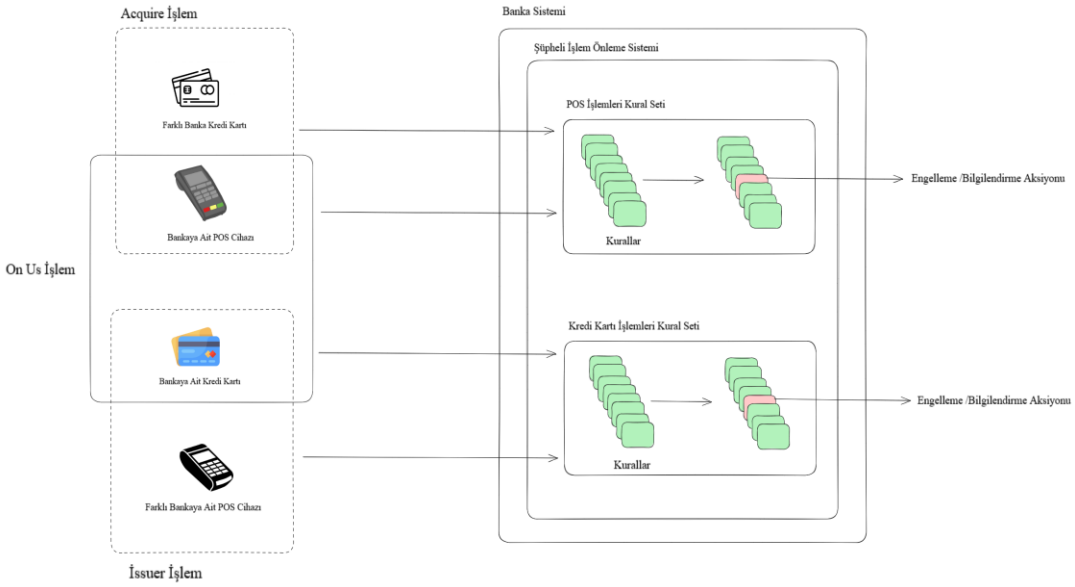
Tablo 3.2. İşlem tiplerinin Pos ve Kart sahipliklerine göre belirlenmesi.

	POS	Card
Acquire	Bizim	Farklı Banka
Issuer	Farklı Banka	Bizim
On Us	Bizim	Bizim

Sadece POS sahibi bizim bankamız ise işlemler Acquire işlem, sadece kart sahibi bizim bankamız ise POS farklı bir bankaya ait ise Issuer işlem, hem POS hem kart sahibi bizim bankamız ise On Us işlem olmaktadır.

Bu sistemde POS ve kart işlemleri şüpheli işlem önleme sistemine gelerek buradaki kurallardan geçirilmektedir. Kurallar Pos ve kart işlemlerine göre ayrı kural setlerinde kontrol edilmektedir. Tanımlı kurallardan geçirilen işlem eğer herhangi bir kuralın filtresine takılır ise şüpheli bir durum oluşmuş olur. Şüpheli durumun kritikliğine göre bu işlem anlık olarak engellenebilir veya müşteriye bilgilendirme gönderilebilir. Ayrıca anlık önlenmesine gerek olmayan işlemler de personeller tarafından kontrol edilerek sonrasında bir aksiyon alınabilir.

Bu işlem tiplerine göre Fraud sisteminin çağırılması da Şekil 3.1'de gösterilmektedir.



Şekil 3.1. İşlem tiplerine göre şüpheli işlem kontrol sisteminin çağırılması.

4. ÇİZGE VERİTABANLARI

Bu bölümde çizge veritabanlarının özelliklerinden, ilişkisel veritabanları ile aralarındaki farklardan bahsedilmiştir. Ayrıca çizge veritabanı türlerine de değinilerek bu tez kapsamında kullanılan Neo4j çizge veritabanı uygulamasından ve cypher sorgu dilinden de bilgiler paylaşılmıştır.

4.1. Çizge Veritabanları Nedir ?

Çizge veritabanları NoSQL türü veritabanlarıdır. Çizge veritabanları içerisindeki veriler tablolarda bulunan satır ve sütunlar şeklinde değil, kenarlar, düğümler ve bunların özellikleri kullanılarak temsil edilir. Düğümler arasındaki ilişkileri merkeze alan bir yapıya sahiptir. Bu neden ile veriler arasındaki ilişki, çizge veritabanlarında fiziksel bir bağlantıdır. Sosyal ağ sistemleri, öneri motorları ve sahtekarlık tespiti ve önleme sistemleri gibi uygulamalar için kullanışlıdır.

Veriler arasındaki ilişkiler çizge veritabanlarında fiziksel bağlantılar olarak tutulduğu için, ilişkilere dayalı verileri tutmak ve kolaylıkla analiz edebilmek için kullanılmaktadırlar. Şüpheli işlem desenlerini ve ağlarını açığa çıkarmada diğer veritabanı teknolojilerine göre kolaylık sağlar. Bu nedenle şüpheli işlem önleme sürecine yardımcı olabilirler.

4.2. İlişkisel Veritabanlarından Farkları Nelerdir ?

Çizge veritabanları ile diğer veritabanı türleri arasında bazı temel farklılıklar vardır. Bunların en önemlilerinden biri veriyi tutma şekillerine göredir. Doküman tabanlı veritabanları her bir veriyi ayrı bir doküman olarak tutarken, ilişkisel veritabanları verileri ilişkilerine göre ayırarak farklı tablolarda tutarlar. Çizge veritabanlarında ise ilişkisel veritabanlarındaki satır ve sütunların yerini çizge veritabanlarında düğüm ve kenarlar almıştır. Tablo 4.1'de bu farklar açıklanmıştır.

Tablo 4.1. Çizge ve İlişkisel veritabanları arasındaki temel farklar.

	Çizge Veritabanları	İlişkisel Veritabanları
Format	Düğüm ve kenarlar	Satır ve sütunlardan oluşan tablolar
İlişkiler	İlişkiler düğümler arasındaki kenarlar ile gösterilen fiziksel varlıklardır.	İlişkiler tablolar arasında ikincil anahtarlar ile kurulmuş fiziksel olmayan varlıklardır.
Karmaşık Sorgular	Birleştirme işlemine gerek olmadan çalışır.	Tablolar arasında karmaşık birleştirmeler yapılarak çalışır.
Kullanım Alanları	Verinin değil ilişkinin ön planda olduğu durumlar.	İşlemin ön planda olduğu durumlar.

İlişkisel veritabanlarında veriye göre tablo tasarımı yapısal olarak oluşturulur ve normalleştirme ile tablolar ilişkilerine göre farklı tablolara ayrılır. Sorgu anında ise ayrılmış tablolardan birincil ve ikincil anahtar değerlerine göre birleştirilerek sonuç getirilir. Çizge veritabanlarında bu durum farklıdır. İlişkiler birincil ve ikincil anahtar ile sağlanan bir yapının yerine direkt olarak verilerin arasında kenarlar ile gösterilen fiziksel bir bağlantıdır. Çizge veritabanlarında sorgu esnasında sonuç zaten hazır olan ilişkiler üzerinden direkt getirilir.

4.3. Çizge Veritabanı Türleri

Çizge veritabanları bir çok farklı amaç için kullanılabilmektedir. Veri modeline ve depolama yöntemlerine göre farklılaşan çizge veritabanlarından bahsedilmiştir. Ayrıca ilgili yöntemi kullanan uygulamalar da örnek olarak verilmiştir.

4.3.1. Veri modeline göre çizge veritabanı türleri

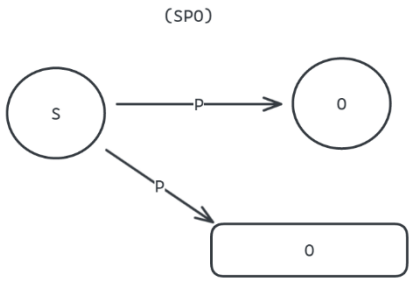
Bahsedilen temel farklılıklardan sonra çizge veritabanı teknolojisinin bulundurduğu veri modeline göre farklı türleri vardır. Bu türler RDF, LPG ve Hypergraph olarak adlandırılırlar.

4.3.1.1. Kaynak tanımlama çerçevesi

Türkçe'de Kaynak Tanımlama Çerçevesi bir semantik web standartıdır. Bu yapı, web üzerindeki kaynakları tanımlamak, ilişkilendirmek ve açıklamak için kullanılan bir veri modeli ve formatını ifade eder. Özellikle semantik web projelerinde ve bağlamsal veri modellerinde kullanılan bir standarttır.

Bu modeldeki veri, üçlü (triple) yapısını kullanır. Bu yapı, bir konu (subject), özne (predicate), ve nesne (object) olmak üzere üç ana öğeden oluşur. Bir kaynağın özelliklerini açıklamak için kullanılır.

Kaynak tanımlama çerçevesi ve çizge veritabanları, özellikle semantik web projeleri, linked data uygulamaları ve ontoloji tabanlı sistemler gibi alanlarda birlikte kullanılarak verilerin anlamlandırılmasını, bağlanmasını ve etkili bir şekilde yönetilmesini sağlar. Örnek bir model şekil 4.1'de gösterilmiştir.



Şekil 4.1. Kaynak tanımlama çerçevesi model gösterimi.

4.3.1.2. HyperGraph

Hypergraph, çizge teorisine dayalı bir veri yapısıdır ve çizge veritabanlarındaki düğümler ve kenarlar konseptini genişleterek, kenarların iki düğüm arasındaki bağlantıyı değil, birden fazla düğüm arasındaki bağlantıyı ifade etmesine izin verir. Bu, veri modelini daha esnek ve kapsamlı hale getirir, çünkü bir kenarın birçok düğümle ilişkilendirilebilmesine olanak tanır.

Hypergraph'in temel unsurları şunlardır:

- **Düğümler**

Hypergraph'de düğümler, genellikle bir varlık veya kavramı temsil eder. Bu düğümler, hyperkenarlar tarafından birleştirilir.

- **Hyperkenarlar**

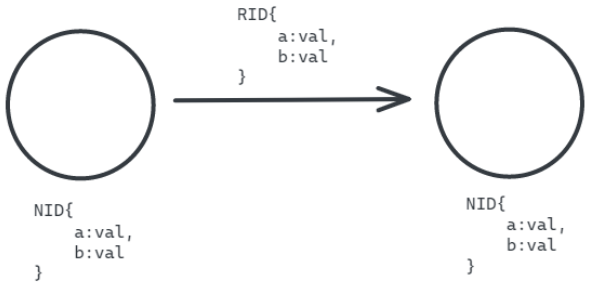
Hypergraph'daki temel farklılık hyperkenarlardır. Bir hyperkenar, iki düğüm arasındaki bağlantıyı değil, birçok düğüm arasındaki bağlantıyı temsil eder. Bu, geleneksel çizge veritabanlarından farklıdır.

Hypergraph bir kenarın birden fazla düğümle ilişkilendirilebilmesi sayesinde karmaşık ve çoklu ilişkileri daha doğrudan ifade edebilir. Bu özellik, bir varlık üzerinde etkileşimde olan çok sayıda diğer varlık bulunduğu durumlarda faydalı olabilir. Ayrıca modelleme açısından daha esnek bir yaklaşım sunar. Her düğümün birbirine bağlı olduğu geniş bir ilişki ağı oluşturabilir.

Hypergraph, özellikle çoklu varlık ilişkilerini ifade etme ihtiyacı olan veritabanları veya bilgi modelleme projeleri için uygun olabilir. Bu tür projelerde, bir kenarın birden fazla düğümle bağlantı kurabilme yeteneği önemli bir avantaj sağlayabilir.

4.3.1.3. Etiketlenmiş özellik çizgesi

Neo4j ve AWS Neptune gibi uygulamalar tarafından kullanılan klasik çizge veritabanı gösterimidir. Düğümler ve ilişkilerden oluşur. Düğüm ve ilişkilerin niteliklerini belli eden etiketler ve özellikler kullanılabilir. Örnek bir model Şekil 4.2'de gösterilmiştir.



Şekil 4.2. Etiketlenmiş özellik çizgesi model örneği.

Bu veri modeli, çizge veritabanlarının kullanıcıların verileri daha açık ve anlamlandırabilir bir şekilde modellemelerini sağlar. Etiketler, düğümleri ve kenarları kategorize ederken, özellikler ise bu verilere ayrıntılı bilgiler eklemelerine olanak tanır.

4.4. Neo4j

Farklı çizge veritabanı türleri ve uygulamalarından bahsettikten sonra bu bölümde tez kapsamında kullandığımız uygulama olan Neo4j'den bahsedilecektir.

Neo4j, 2000'li yılların başlarında İsviçrede kurulan Neo Technology firmasının ilişkisel veritabanlarına alternatif olarak çizge veritabanlarının potansiyelini keşfetmesi ve geliştirmesi sonucunda 2007 de ilk sürümüyle piyasaya çıkmış bir açık kaynak çizge veritabanı yönetim sistemidir ve verileri depolamak, sorgulamak ve yönetmek için özel olarak tasarlanmıştır.

Neo4j etiketlenmiş özellik çizgesi modelini destekler. İncelemeler ve ilişkiler üzerinde çalışmak için özelleştirilmiş güçlü ön yüze sahip bir uygulamadır.

4.5. Cypher Sorgu Dili

Cypher, Neo4j uygulaması ile beraber özel olarak geliştirilen çizge veritabanlarında sorgulama yapmak için tasarlammış bir sorgulama dilidir [21]. Bu çalışmada modeli entegre ederken kullanılan sorgu dili cypher dilidir.

SQL'de yazılan bir çok farklı tabloyu birleştirerek sonuç elde eden karmaşık sorguları cypher dili ile daha basit bir şekilde ifade edilebilir. Aynı verileri getirebilmek için yazılan SQL ve Cypher sorguları aşağıda paylaşılmıştır.

```
SELECT 'C1372042334' AS CustomerId , ct.category FROM Transaction t
INNER JOIN Customer cu ON cu.CustomerId  = t.CustomerId
INNER JOIN Category ct ON ct.CategoryId = t.CategoryId
Where cu.CustomerId = 'C808326652' AND t.isFraud  = 1
```

Yukarıdaki sorguyu Cypher ile yazmak istediğimizde daha anlaşılır ve basit olmaktadır. Çünkü ilişkiler "JOIN" kalıpları ile değil direkt olarak verileri arasındaki ilişkilerin yönü ve isimleri ile sorgulanmaktadır.

```
MATCH (cu:Customer)-[:MAKES]-> (t:Transaction)-[:BELONGS_TO] ->
(ct:Category) WHERE  t.isFraud =1 AND
cu.customerId = "C1372042334" RETURN "C1372042334" as CustomerId ,
ct.category;
```

Yine aşağıdaki sorguda bir müşterinin yaptığı toplam işlem sayısını ilişkisel veritabanlarından alt sorgular kullanılarak nasıl getirebileceğimizi gösteren SQL kodu bulunmaktadır.

```
SELECT CustomerName, TransactionCount
FROM Customers c
JOIN (
    SELECT CustomerId, COUNT(*) AS TransactionCount
    FROM Transactions
```

```
    GROUP BY CustomerId
) t ON c.CustomerId = t.CustomerId
ORDER BY TransactionCount DESC;
```

Yukarıdaki sorgu SQL dilindeki bir sorgu iken, aşağıdaki sorgu cypher sorgusunu temsil etmektedir. Cypher dilinde SQL'de kullanılan alt sorgular "WITH" kalıbı ile gerçekleştirilmektedir.

```
MATCH (c:Customer)-[:MAKES]->(t:Transaction)
WITH c, COUNT(t) AS TransactionCount
RETURN c.customerId, TransactionCount ORDER BYTransactionCount DESC;
```

5. MATERYAL

5.1. Veri Seti

Bu çalışmada, Kaggle'dan alınan BankSim veri seti kullanılmıştır [22,23]. BankSim veri seti, bir bankanın müşteri ödemelerindeki dolandırıcılık desenlerine dayalı olarak sentetik olarak akademik araştırmalarda kullanılmak amacıyla oluşturulmuştur. Veri güvenliğini sağlamak için gerçek kişisel bilgiler veri kümesine dahil değildir. Bu veri kümesinin sayısal detayları, 180 günlük bir örneklem için Tablo 5.1'de gösterilmektedir. Ayrıca, BankSim veri kümesindeki sütunlar, ayrıntıları ve özellikleri Tablo 5.2'de sunulmaktadır.

Tablo 5.1. BankSim veri kümesindeki şüpheli olan ve olmayan işlem sayıları.

Toplam	Şüpheli Olmayan	Şüpheli
594643	587443	7200

Tablo 5.2. BankSim veri kümesindeki kolon detayları.

Kolon	Açıklama
Step	BankSim veri seti 180 günlük verinin örneklemi ile sentetik olarak oluşturulmuştur. “Step” kolonu gün bilgisini ifade eder. 1 değeri ile 180 değeri arasında değişir.
Customer	İşlemi gerçekleştirerek ödemeyi yapan müşterinin eşsiz müşteri numarası bilgisini içeren kolon.
Age	Müşterinin yaş grup bilgisini içeren kolon. Verideki müşterilerin yaşları farklı gruplara ayrılacak bu kolonda ilgili müşterinin yaşının hangi gruba denk geldiği belirtilmiştir. Grup ‘0’ 18 yaş altı kişileri temsil ederken, grup ‘1’ 19 ve 25 yaş aralığındaki kişileri temsil eder.

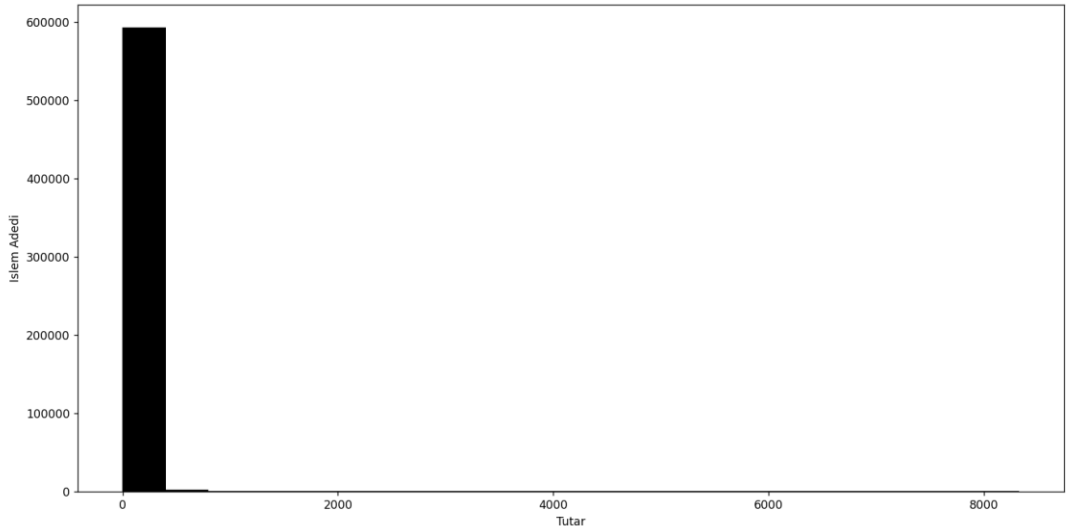
Tablo 5.2. (Devamı) BankSim veri kümesindeki kolon detayları.

Kolon	Açıklama
Gender	Müşterinin cinsiyet bilgisi.
Zipcode	Müşterinin ikametinin bulunduğu yerin adres alan kodu.
Merchant	Müşterinin işlemi yaptığı iş yerinin eşsiz değer bilgisi.
ZipMerchant	İş yerinin bulunduğu bölgenin adres alan kodu.
Category	Müşteri ve iş yeri arasında gerçekleşen işlemin kategorisi.
Amount	Yapılan işlemin tutarı.
Fraud	Yapılan işlemin şüpheli bir işlem olup olmadığına dair bilgi. '0' ve '1' değerini alır. '0' işlemin şüpheli olmadığını gösterirken. '1' işlemin şüpheli olduğunu belirtir.

5.2. Verinin Hazırlanması

Önereceğimiz modelde veri sayısını da göz önünde bulundurduğumuzda tutar bilgisinin net olarak bilinmesine gerek yoktur. Bunun yerine daha işlevsel şekilde bölünmüş ve aralıkları belirlenmiş tutar grupları model ve sorgu çıktılarımızı daha anlamlı hale getirecektir. Örneğin işleme ait 120.32 tutarındaki bilgiyi direkt olarak bu sayısal değer ile değil 100-130 arasındaki tutarlara verdiğimiz 3 numaralı grup ile temsil ettiğimizde son adımda ortaya çıkan kurallarda belli bir aralıkta olan işlem tutarlarının şüpheli veya şüpheli olmadığını belirtebileceğiz.

Bu nedenle öncelikli olarak tutar bilgilerini kaç grupta temsil etmemiz gerektiğini yani en uygun grup sayısını bulmamız gerekmektedir. Bunun için birden fazla yöntem vardır. Sturge's kuralı [24] simetrik ve çarpık olmayan veri setlerinin bölünebileceği gurup sayısını bulmak için kullanılırken Doane's kuralı [25] ise daha çok çarpık olan ve simetrik olmayan veriler için kullanılmaktadır. Bizim verimizin çarpıklığı yüksek ve simetrik olmayan bir veri olduğunu Şekil 5.1'de görebiliriz. Tutar bilgilerini bir histogram grafiğine döktüğümüzde '8000' tutarına kadar değerler yayılmış olsa da tutarların büyük bir çoğunluğunun çok dar bir alanda yoğunlaştığını görebiliyoruz.



Şekil 5.1. Tutar bilgisinin gruplanmadan önce oluşturulan histogram grafiği.

Doane's kuralı yönteminin formülü denklem 5.1 ve denklem 5.2'de gösterilmiştir. Formüle göre bir veri setini gruplanmış frekans tablosu şeklinde göstermek istediğinizde kaç grup olacağınına dair en uygun değeri önermektedir. 5.1 denklemindeki k değeri optimal grup sayısını temsil etmektedir. İki denklemde de kullanılan $g1$ ise veri setinin yirmi beşinci ve yetmiş beşinci yüzdelik dilimleri arasındaki mesafeyi temsil eder. Bu değere göre hesaplanan standart sapma sonrası ilgili logaritmik işlemlerin de yapılması ile k değerine ulaşılmış olur. Bu hesaplamayı Python yardımı ile yaptığımızda optimal grup sayımız 21 olmaktadır.

$$k = 1 + \log_2(n) + \log_2(1 + \frac{|g1|}{\sigma_{g1}}) \tag{5.1}$$

$$\sigma_{g1} = \sqrt{\frac{6(n-2)}{(n+1)(n+3)}} \tag{5.2}$$

Grup sayısı belirlendikten sonra bu sayıya göre verileri aralıklı olarak gruplamalıyız. Veri setimizdeki tutar bilgisi simetrik bir veri içermemektedir. Eğer grup sayısına göre eşit aralıklara bölerek frekans tablosu oluşturacak olursak ilk grubumuz 0-381 aralığında olur ve bu değer aralığında bulunan işlem sayısı 591549'dur. Bu değer veri setimizin %0,995'ine denk gelmektedir. Yine çok kısa bir aralıkta tutar bilgileri yoğunlaşmış olmaktadır. Bu nedenle simetrik olmayan, kırılımı yüksek ve dağınık verisetlerini gruplamada kullanılan matematiksel yol olan Jenk'in Doğal Kırılım (Jenks Natural Breaks) yöntemi ile veri setimizi gruplara ayırıyoruz. Bu matematiksel yöntem her bir grup içerisindeki benzerlik oranlarını yüksek tutmaya çalışırken,

gruplar arasındaki farklılıkları da yüksek tutmaya çalışırak uygun grup aralıklarını bulur. Veriler gruplanırken örnek gruplar seçilerek her seçime göre ilgili aralıkların istenen formüle ne kadar uyduğunun değeri hesaplanır ve en uygun grup dağılımı elde edilmeye çalışılır. İlgili yöntemin matematiksel formülü aşağıda verilmiştir. 5.3 denkleminde bulunan SSD ifadesi her bir değerin kendi sınıf ortalaması ile arasındaki kare farklarının toplamını ifade eder. Bu değer minimize edilmeye çalışılır. 5.4 denkleminde ise SSW ifadesi oluşturulan her bir sınıfın ortalaması ile genel ortalama arasındaki kare farklarının toplamını ifade eder. Bu değer ise maksimize edilmeye çalışılır. Son denklem olan 5.5'de ise toplam varyans SSD ve SSW değerleri kullanılarak hesaplamaya çalışılır.

$$SSD = \Sigma (x_i - \mu)^2 \tag{5.3}$$

$$SSW = \Sigma (x_i - \bar{x}_i)^2 \tag{5.4}$$

$$\chi^2 = (SSD - SSW) / SSW \tag{5.5}$$

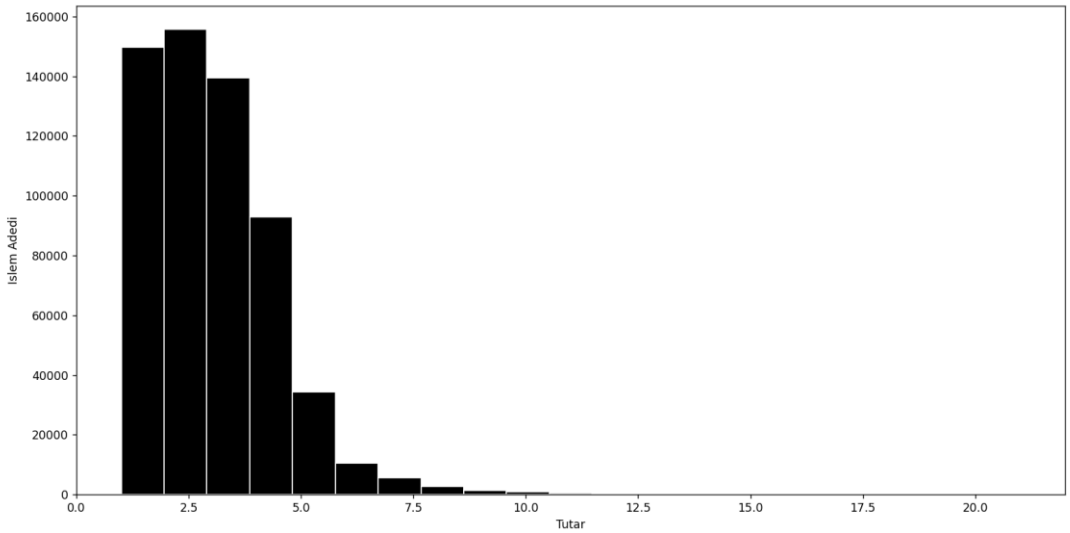
İlgili işlemi yapan ve ardından her işleme ait tutar bilgisine göre oluşan grubu yeni bir kolon olarak verisetimize ekleyen python dilindeki kod bloğu aşağıda paylaşılmıştır. Bu python kodunda jenkspy kütüphanesinin jenks_breaks methodu kullanılmıştır.

```
import pandas as pd
import jenkspy
f = pd.read_csv("DataSet/bs140513_032310.csv")
dataFrame = pd.DataFrame(f)
dataFrame ['amountGroup'] = pd.cut(
dataFrame ['amount'],
bins=jenkspy.jenks_breaks(dataFrame ['amount'], 21),
labels= [str(i) for i in range(1, 22)],
include_lowest=True)
dataFrame = dataFrame.sort_values(by='amount')
dataFrame.to_csv('bs140513_032310_updated.csv', index=False)
```

Bu hesaplama ve ayrımlara göre yeni histogram grafiğimiz Şekil 5.2'de ve güncellenmiş BankSim veri setimizden temel kolonlar ile bir örnek Tablo 5.3'de paylaşılmıştır. İhtiyacımız olan modele daha uygun ve işlevsel bir gruplama oluşturulmuştur.

Tablo 5.3. Veri zenginleştirilmesi yapıldıktan sonra veri kümesinin durumu.

Step	Customer	Merchant	Category	Amount	Fraud	Amount Group
66	'C1659350842'	'M1823072687'	'es_transportation'	0	0	1
169	'C1156808163'	'M1535107174'	'es_wellnessandbeauty'	23.4	0	2
103	'C1883962820'	'M480139044'	'es_health'	69.52	0	5
153	'C616715154'	'M732195782'	'es_travel'	6110.23	1	2 0



Şekil 5.2. Gruplama yapıldıktan sonra tutar bilgisinin dağılım grafiği.

6. YÖNTEM

Bu bölümde hazır hale getirdiğimiz veri setimizi Neo4j uygulamasında önerdiğimiz modele göre içe aktaracağız. Ardından Neo4j uygulamasının arayüzünde verilerimizi gördükten sonra kural tabanlı bir sisteme örnek kurallar üretebilmek için örnek sorgular paylaşacağız.

6.1. Neo4j Uygulamasına Verilerin Aktarılması

İlk olarak Neo4j uygulaması üzerinde çalışmalarımızı yapabileceğimiz "BankSim Fraud" adında bir database oluşturduk. Bu çalışma yapılırken Neo4j uygulamasının güncel versiyonu olan 5.3.0 sürümü kullanılmıştır.

Veriyi uygulamaya eklemeden önce uygulamanın bu veriye daha rahat erişmesi için veritabanının kurulu olduğu klasör altındaki "import" isimli klasöre '.csv' uzantılı dosya atılmıştır.

Ardından veritabanı ayağa kaldırılarak başlatılması sağlanmıştır. Gelen kullanım paneli üzerinden cypher dili ile sorgular yazarak verimizi import edebilir ve üzerinde analiz yapabiliriz.

6.1.1. Indexlerin oluşturulması

Cypher komutları ile iş yerleri ve müşterilerin eşsiz değerleri için indeks kaydı oluşturuluyor. Tanımlanan indeksler sayesinde her biri eşsiz olan bu düğümlerde sorgulama performansı artmaktadır.

```
CREATE INDEX FOR (m:Merchant) ON (m.merchantId)
CREATE INDEX FOR (c:Customer) ON (c.customerId)
CREATE INDEX FOR (ct:Category) ON (ct.category)
CREATE INDEX FOR (a:AmountGroup) ON (a.id)
```

6.1.2. CSV türün deki dosyayı içe aktararak model oluşturulması

Aşağıdaki komut ile ham veri Neo4j uygulamasına yani çizge veritabanı içerisine eklenebilir.

```
:auto LOAD CSV WITH HEADERS FROM
'file:///bs140513_032310_updated.csv' AS bankSimCsv
```

```
CALL{
    WITH bankSimCsv
  MERGE (c:Customer {customerId:bankSimCsv.customer, gender:bankSimCs
v.gender,ageGroup:bankSimCsv.age,zipCode:bankSimCsv.zipcodeOri})
  MERGE (m:Merchant {merchantId: bankSimCsv.merchant, zipCode:bankSim
Csv.zipMerchant})
  MERGE (ct:Category {category:bankSimCsv.category})
  MERGE (a:AmountGroup {groupId:toInteger(bankSimCsv.amountGroup)})
  CREATE (t:Transaction {day:toInteger(bankSimCsv.step),isFraud:toInt
eger(bankSimCsv.fraud)})
  CREATE (c)- [:MAKES]->(t)
  CREATE (t)- [:TO]->(m)
  CREATE (t)- [:BELONGS_TO]->(a)
  CREATE (t)- [:BELONGS_TO]->(ct)
} IN TRANSACTIONS
```

Tutar gruplarının aralıkları sorgu çıktılarını anlamlandırabilmek için düğüm üzerinde bize gereklidir. Aşağıda her bir grup için aralık bilgisi Tablo 6.1'de gösterilmiştir.

Tablo 6.1. Tutar gruplarının aralıkları.

Amount Group	Range
1	0.0 - 13.84
2	13.85 - 27.64
3	27.65 - 42.4
4	42.41 - 61.16
5	61.17 - 94.62
6	94.63 - 149.65
7	149.66 - 227.55
8	227.73 - 335.07
9	335.21 - 479.75
10	479.95 - 657.64
11	658.51 - 876.23
12	876.97 - 1201.96
13	1209.46 - 1673.54

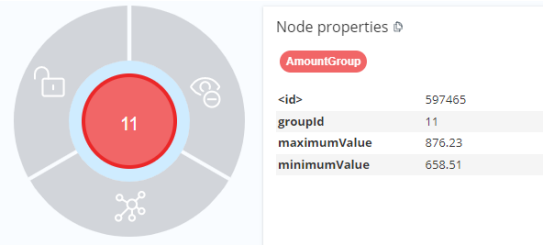
Tablo 6.1. (Devamı) Tutar gruplarının aralıkları.

Amount Group	Range
14	1688.02 - 2225.62
15	2241.69 - 2798.94
16	2832.02 - 3491.53
17	3525.98 - 4265.41
18	4308.89 - 5070.57
19	5107.7 - 5856.7
20	6110.23 - 6888.3
21	7134.39 - 8329.96

Bu tablodaki verileri "AmountGroup" düğümünün bir özelliği olarak verebilmek için aşağıdaki cypher dilindeki kodu çalıştırıyoruz.

```
MATCH (ag:AmountGroup)
SET ag.minimumValue =
CASE ag.groupId
WHEN 1 THEN 0.0
WHEN 2 THEN 13.85
…
WHEN 21 THEN 7134.39
ELSE null
END,
SET ag.maximumValue =
CASE ag.groupId
WHEN 1 THEN 13.84
WHEN 2 THEN 27.64
…
WHEN 21 THEN 8329.96
ELSE null
END;
```

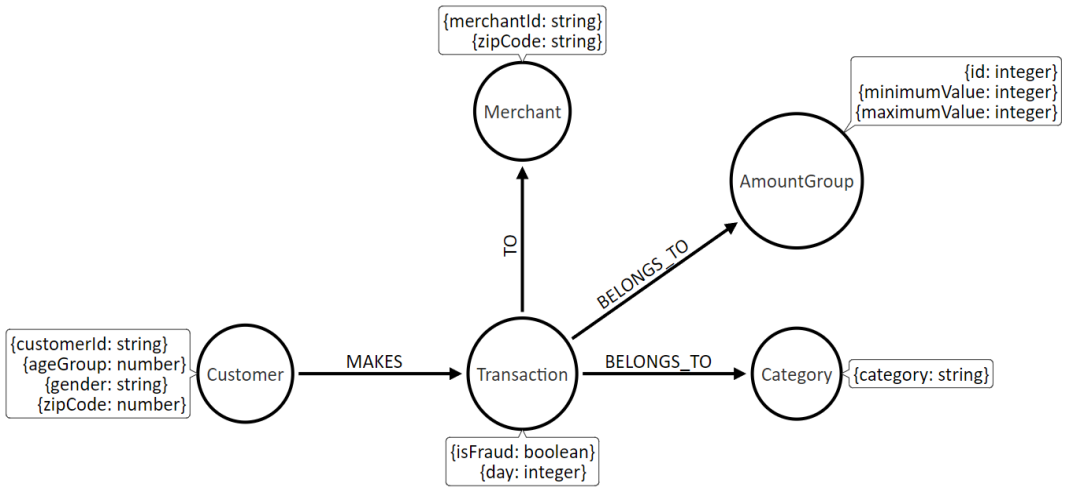
Sorguyu çalıştırdıktan sonra Şekil 6.1'de de görüleceği üzere düğüm üzerinde maksimum ve minimum değerleri görebiliyoruz.



Şekil 6.1. Tutar grubu minimum ve maksimum değer bilgileri.

6.2. Önerilen Model

Bu çalışmada, BankSim veri kümesinin bir çizge veritabanına eklenmesi ve kullanılması sunulmuş, önerdiğimiz model Şekiş 6.2'de gösterilmiştir. Bu modelle birlikte işlemi gerçekleştiren müşteri, işlemin yapıldığı satıcı, işlem tutarının ait olduğu miktar grubu, işlem kategorisi ve işlemin kendisi düğüm olarak temsil edilebir. Bu düğümlere özellikler olarak ek bilgiler eklenebilir. Daha fazla düğüm ekleyerek daha net ve tanımlı sorgular oluşturulması sağlanabilir.



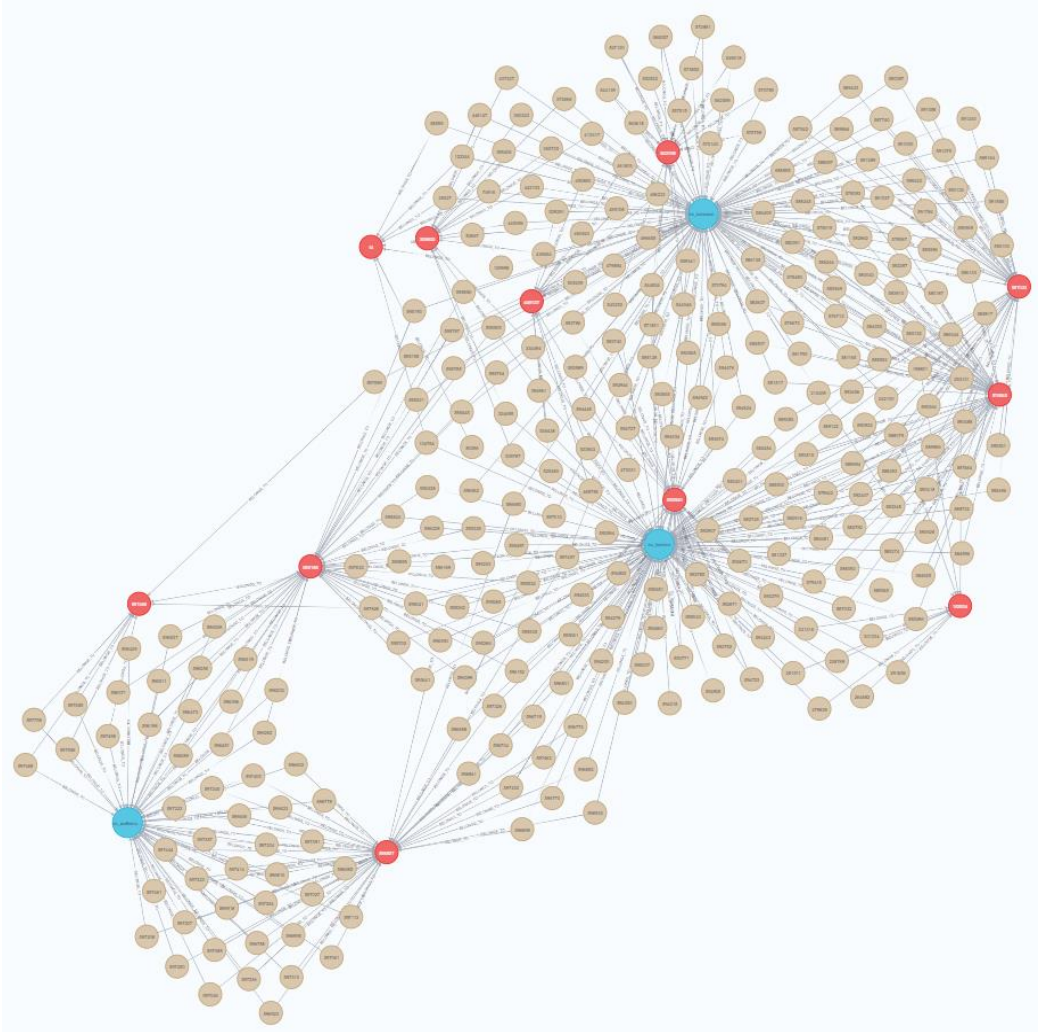
Şekil 6.2. Önerilen çizge veritabanı modeli.

6.3. Örnek Sorgular

Veri kümesi zenginleştirme ve çizge veritabanı modelinin belirlenmesinin ardından, 'CSV' formatındaki veri kümesi, Neo4j masaüstü uygulaması üzerinden Cypher dilini kullanarak içe aktarılmıştır. Artık elimizdeki veri setine göre kural tabanlı bir sisteme ek metrikler ve analizler sağlayabilir ayrıca kural yazımlarında yardımcı olacak sorgular çalıştırabiliriz.

İlk olarak şüpheli işlemlere ait olan kategori ve tutar gruplarını çekmek istediğimizde gelen sonuç Şekil. 6.3'de verilmiştir. Gözükeceği üzere Neo4j uygulamasının

arayüzünde düğümleri ve düğümlerin birbiri ile ilişkilerini çok daha anlaşılabilir şekilde görebilmekteyiz. Kategoriler mavi, müşteriler kırmızı renkte gösterilirken bu müşterilerin yapıtığı işlemler bej renkte gösterilmiştir.



Şekil 6.3. Neo4j uygulamasının arayüzünde veri setimizdeki bazı alanların genel bir görünümü.

Belirli bir kategoride yapılan şüpheli işlemlerin ait olduğu tutar gruplarının maksimum ve minimum değerleri bir kural teşkil eder. Örneğin, "es_fashion" kategorisindeki bir işlem için aşağıdaki cypher dilindeki sorguyu çalıştırdığımızda, Tablo 6.2'de gösterilen çıktıyı elde ederiz.

```
MATCH (c:Category)<--(t:Transaction)--> (a:AmountGroup)
WHERE c.category="es_fashion" AND t.isFraud=1
RETURN c.category as CategoryName ,
       count(t) as FraudCount,
       a.groupId as AmountGroup,
```

```
        a.maximumValue as MaximumValue,
        a.minimumValue as MinimumValue
ORDER BY count(t) desc
```

Table 6.2. Belirli bir kategorideki şüpheli işlemlerin tutar aralıkları.

Kategori	Şüpheli İşlem Sayısı	Tutar Grubu	Değer Aralığı
"es_fashion"	22	8	227.73-335.07
"es_fashion"	19	9	335.21-479.75
"es_fashion"	19	6	94.63-149.65
"es_fashion"	17	7	149.66-227.55

Bu sorgunun çıktısına göre, bir kategoriye ait şüpheli işlemlerin en sık gerçekleştiği tutar gruplarına göre bir şüpheli işlem tespit kuralı yazılabilir. İlgili kural 6.1 numaralı denklemde gösterilmiştir.

$$f\,(Kategori\,, Tutar\,) = \begin{cases} 1, & eğer\ Kategori = \text{"es\_fashion"}\ \wedge\ (335.07 \geq Tutar\ \geq 227.73) \\ 0, & aksi\ takdirde \end{cases} \tag{6.1}$$

Başka bir örnekte, belirli bir tutar grubunda şüpheli işlemlerin en çok gerçekleştiği ilk 3 iş yerini bulmak istediğimizde aşağıdaki gibi bir cypher sorgu kodu ile bu sonuca ulaşabiliriz. İlgi sorgunun çıktısı Tablo 6.3 de görülebilmektedir.

```
MATCH (m:Merchant)<--(t:Transaction)--> (a:AmountGroup)
WHERE  t.isFraud=1 AND a.groupId = 3
RETURN m.merchantId as Merchant ,
       count(t) as FraudCount
ORDER BY count(t) desc
LIMIT 3
```

Tablo 6.3. Tutar grubuna göre şüpheli işlemlerin en çok yapıldığı iş yerleri.

İş Yeri	Toplam Şüpheli İşlem Sayısı
M480139044	46
M980657600	36
M855959430	18

Aynı şekilde bu çıktı da aşağıdaki gibi formülüze edilerek şüpheli işlem önleme sistemi için bir kural haline getirilebilir. İlgili kural denklem 7'de görülebilmektedir.

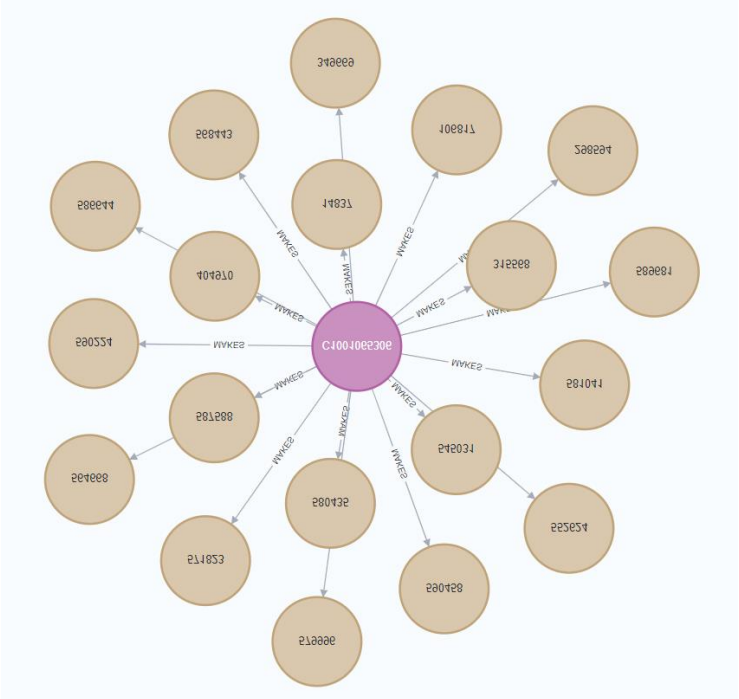
$$f\,(\text{İş}\,Yeri, Tutar\,) = \begin{cases} 1, & eğer\ \text{İş}\,Yeri\ = \text{"M480139044"} \wedge (42.4 \geq Tutar\ \geq 27.65) \\ 0, & aksi\ takdirde \end{cases} \tag{6.2}$$

Ek olarak kural tabanlı sistemlerde eğer bir müşteriye ait şüpheli işlemler var ise bu müşterinin şüpheli olmayan diğer işlemleri de risk teşkil etmektedir ve incelenmesi gerekir. Bu bilgiye aşağıdaki cypher kodu çalıştırılarak erişilebilir.

```
MATCH (suspiciousTransaction:Transaction {isFraud: 1})<-[:MAKES]-
(suspiciousCustomer:Customer)
WITH COLLECT(DISTINCT suspiciousCustomer.customerId) AS fraudCustomer
IDList


MATCH (c:Customer)-[:MAKES]->(t:Transaction)
WHERE c.customerId IN fraudCustomerIDList AND t.isFraud = 0
RETURN c, t
```

Sorgu çalıştırıldıktan sonra elde edilen çıktının çizge gösteriminin bir müşteri için örneği aşağıda verilmiştir.



Şekil 6.4. Şüpheli işlemleri olan bir müşterinin şüpheli olduğu netleşmemiş diğer işlemleri.

7. SONUÇ

Bu tez çalışmasında öncelikle şüpheli işlem, şüpheli işlem önleme sistemleri ve çizge veritabanları ile alakalı literatür araştırması yaptık. Ardından şüpheli işlemin ne olduğu ve türlerinden bahsettik. Şüpheli işlem önleme sistemlerinin nasıl sistemler olduklarından, çeşitlerinden ve tezimizin odaklandığı kural tabanlı şüpheli işlem önleme sistemlerinden de bilgi aktararak, örnek bir şüpheli işlem önleme sisteminin modelini paylaştık. Sonraki bölümde çizge veritabanlarının detaylarından ve ilişkisel veritabanlarıyla aralarındaki farklardan bahsettik. Bu çalışmada kullanacağımız Neo4j uygulaması ve cypher sorgu diliyle alakalı bilgi verdikten sonra veri setimizden ve veri setimizin önereceğimiz modelde daha kullanışlı olması için matematiksel formüller ile nasıl yeni bir bilgi türettiğimizi aktardık. Son olarak da bu veri setini Neo4j uygulamasında cypher sorgu dili ile içe aktararak örnek sorgulamalar yaptık, ek olarak bu sorgulamaların çıktılarını kural tabanlı sistemlerin kullanabileceği kurallar olarak ifade etmeye çalıştık.

İlerleyen çalışmalarda çizge veritabanlarından analizler ile üretilen kurallar otomatikleştirilmiş bir süreç ile uzman personelin bir kontrolü sonrası kural havuzuna eklenerek işlemleri kontrol etmeye başlayabilir. Bu şekilde çalışan bir şüpheli işlem önleme sistemine daha önceki işlemler analiz edilerek kural üretilmiş ve kontroller sonrasında kural beslemesi yapılabilmiş olur.

KAYNAKLAR

[1] Mangala, D., & Soni, L. (2023). A systematic literature review on frauds in banking sector. *Journal of Financial Crime*, *30*(1), 285–301. https://doi.org/10.1108/JFC-12-2021-0263

[2] Hilal, W., Gadsden, S. A., & Yawney, J. (2022). Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances. *Expert Systems with Applications*, *193*, 116429. https://doi.org/10.1016/j.eswa.2021.116429

[3] Abdallah, A., Maarof, M. A., & Zainal, A. (2016). Fraud detection system: A survey. *Journal of Network and Computer Applications*, *68*, 90–113. https://doi.org/10.1016/j.jnca.2016.04.007

[4] Pourhabibi, T., Ong, K.-L., Kam, B. H., & Boo, Y. L. (2020). Fraud detection: A systematic literature review of graph-based anomaly detection approaches. *Decision Support Systems*, *133*, 113303. https://doi.org/10.1016/j.dss.2020.113303

[5] Correa Bahnsen, A., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. *Expert Systems with Applications*, *51*, 134–142. https://doi.org/10.1016/j.eswa.2015.12.030

[6] Carcillo, F., Le Borgne, Y.-A., Caelen, O., Kessaci, Y., Oblé, F., & Bontempi, G. (2021). Combining unsupervised and supervised learning in credit card fraud detection. *Information Sciences*, *557*, 317–331. https://doi.org/10.1016/j.ins.2019.05.042

[7] Ceronmani Sharmila, V., R., K. K., R., S., D., S., & R., H. (2019). Credit Card Fraud Detection Using Anomaly Techniques. *2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, 1–6. https://doi.org/10.1109/ICIICT1.2019.8741421

[8] Huang, D., Mu, D., Yang, L., & Cai, X. (2018). CoDetect: Financial Fraud Detection With Anomaly Feature Detection. *IEEE Access*, *6*, 19161–19174. https://doi.org/10.1109/ACCESS.2018.281656

[9] Vorobyev, I., & Krivitskaya, A. (2022). Reducing false positives in bank anti-fraud systems based on rule induction in distributed tree-based models. *Computers & Security*, *120*, 102786. https://doi.org/10.1016/j.cose.2022.102786

[10] Gianini, G., Ghemmogne Fossi, L., Mio, C., Caelen, O., Brunie, L., & Damiani, E. (2020). Managing a pool of rules for credit card fraud detection by a Game Theory based approach. *Future Generation Computer Systems*, *102*, 549–561. https://doi.org/10.1016/j.future.2019.08.028

[11] Ozcan, F., & Genc, Y. (2022). Increasing Transaction Fraud Prediction Ability by Using Multi-Task Learning and Pruning. *2022 30th Signal Processing and Communications Applications Conference (SIU)*, 1–4. https://doi.org/10.1109/SIU55565.2022.9864949

[12] Yuksel, B., Bahtiyar, S., & Yilmazer, A. (2020). *Credit Card Fraud Detection with NCA Dimensionality Reduction*. 1–7. https://doi.org/10.1145/3433174.3433178

[13] Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., & Baesens, B. (2015). APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions. *Decision Support Systems*, *75*, 38–48. https://doi.org/10.1016/j.dss.2015.04.013

[14] Jing, R., Zheng, X., Tian, H., Zhang, X., Chen, W., Wu, D. D., & Zeng, D. D. (2019). A Graph-Based Semi-Supervised Fraud Detection Framework. *2019 4th IEEE International Conference on Cybernetics (Cybconf)*, 1–5. https://doi.org/10.1109/Cybconf47073.2019.9436573

[15] Prusti, D., Das, D., & Rath, S. K. (2021). Credit Card Fraud Detection Technique by Applying Graph Database Model. *Arabian Journal for Science and Engineering*, *46*(9), 1–20. https://doi.org/10.1007/s13369-021-05682-9

[16] Molloy, I., Chari, S., Finkler, U., Wiggerman, M., Jonker, C., Habeck, T., Park, Y., Jordens, F., & Van Schaik, R. (2017). Graph Analytics for Real-Time Scoring of Cross-Channel Transactional Fraud. In J. Grossklags & B. Preneel (Eds.), *Financial Cryptography and Data Security* (Vol. 9603, pp. 22–40). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-54970-4_2

[17] Kurshan, E., Shen, H., & Yu, H. (2020). Financial Crime & Fraud Detection Using Graph Computing: Application Considerations & Outlook. *2020 Second InternationalConference on Transdisciplinary AI (TransAI)*, 125–130. https://doi.org/10.1109/TransAI49837.2020.00029

[18] Çavşi Zaim, H., Yolaçan, E. N., & Gülbandilar, E. (2021). Banka Ödemelerinde Dolandırıcılığın Çizge Madenciliği ve Makine Öğrenimi Algoritmalarıyla Tespiti. *DÜMF Mühendislik Dergisi*, 615–625. https://doi.org/10.24012/dumf.1002110

[19] Van Belle, R., Van Damme, C., Tytgat, H., & De Weerdt, J. (2022). Inductive Graph Representation Learning for fraud detection. *Expert Systems with Applications*, 116463. https://doi.org/10.1016/j.eswa.2021.116463

[20] Henderson, R. (2020). Using graph databases to detect financial fraud. *Computer Fraud & Security*, *2020*(7), 6–10. https://doi.org/10.1016/S1361-3723(20)30073-7

[21] *Neo4j Cypher Query Language*. (n.d.). Neo4j Cypher Query Language. https://neo4j.com/product/cypher-graph-query-language/

[22] Lopez-Rojas, E. A., & Axelsson, S. (2014). *Banksim: A bank payments simulator for fraud detection research*. 144–152. https://www.kaggle.com/datasets/ealaxi/banksim1

[23] Lopez-Rojas, E. A., & Axelsson, S. (2014). *Synthetic data from a financial payment system | Kaggle*. Synthetic Data from a Financial Payment System. https://www.kaggle.com/datasets/ealaxi/banksim1

[24] Sturges, H. A. (1926). The Choice of a Class Interval. *Journal of the American Statistical Association*, *21*(153), 65–66. https://doi.org/10.1080/01621459.1926.10502161

[25] Doane, D. P. (1976). Aesthetic Frequency Classifications. *The American Statistician*, *30*(4), 181–183. https://doi.org/10.1080/00031305.1976.10479172

ÖZGEÇMİŞ

Ad-Soyad : Bahadır Esad DEMİR

ÖĞRENİM DURUMU:

- **Lisans :** 2019, Sakarya Üniversitesi, Bilgisayar ve Bilişim Bilimleri Fakültesi, Bilgisayar Mühendisliği

TEZDEN TÜRETİLEN ESERLER:

- Demir, B.E., ve Şahin, V.H. (2023, 19-20, Ağustos). A Graph Database Model for Rule Based Credit Card Fraud Prevention System. The 17th International Scientific Research Congress, Ankara, Turkey.