



Teknoloji Fakültesi

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

KAREKOD İÇEREN FATURALARDAN BİLGİ ÇIKARTILMASI VE BU FATURALARIN SINIFLANDIRILMASI, ANALİZ EDİLMESİ

BİTİRME PROJESİ 1.

ARA RAPORU

Bilgisayar Mühendisliği Bölümü

DANIŞMAN

Doç. Dr. AYŞE BERNA ALTINEL GİRGİN

İSTANBUL, 2025

MARMARA ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

Marmara Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Öğrencileri **Yunus Ege Küçük, Emir Binçe ve Farid Bayramov** tarafından “**KAREKOD İÇEREN FATURALARDAN BİLGİ ÇIKARTILMASI VE BU FATURALARIN SINIFLANDIRILMASI, ANALİZ EDİLMESİ**” başlıklı proje çalışması, **xxx** tarihinde savunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

Jüri Üyeleri

Dr. Öğr. Üyesi xxx xxx
Marmara Üniversitesi
Prof. Dr. Xxx xxx
Marmara Üniversitesi
Prof. Dr. Xxx xxx
Marmara Üniversitesi

(Danışman)

(Üye)

(Üye)

(İMZA).....

(İMZA).....

(İMZA).....

ÖNSÖZ

Proje çalışmamız süresince karşılaştığımız bütün problemlerde, sabırla yardım ve bilgilerini, manevi desteğini esirgemeyen, proje çalışmamız sırasında okul içerisinde ve okul dışında her zaman yanımızda olan, tüm desteğini sonuna kadar yanımızda hissettiğimiz değerli hocamız, sayın Doç. Dr. Ayşe Berna Altınel Girgin'e en içten teşekkürlerimizi sunarız.

İÇİNDEKİLER

1. GİRİŞ	12
1.1. Proje Çalışmasının Amacı ve Önemi	12
2. LİTERATÜR TARAMASI	13
2.1. Türkçe Faturaların Sınıflandırılmasında Farklı Öznitelik Seçimi Yöntemleri ile Topluluk Öğrenme Algoritmalarının Etkilerinin İncelenmesi	13
2.2. QR Kod Tanıma ve Bulut Depolama Tabanlı Çevrimiçi Faturalama Sistemi	15
2.3. Extraction of Information from Invoices – Challenges in the Extraction Pipeline	16
3. PROJE TASARIMI	17
3.1. Veritabanı Tasarımı	17
3.1.1. PostgreSQL	17
3.1.2. Tabloların Yapısı	18
3.1.3. Fonksiyonlar ve İşlem Akışları	22
3.1.4. Veritabanı ve Normalizasyon	22
3.1.5. Performans ve Güvenlik	23
3.2. API Tasarımı ve Arka Uç Geliştirme	23
3.2.1. Endpointler ve Kullanılan HTTP Request Türleri	23
3.2.2. Veri Kontrolü ve Validasyonu	24
3.2.3. Veri Güvenliği ve Veritabanı Saldırılarına Karşı Güvenlik	25
3.2.4. JWT Kullanımı ve Kullanıcı Authentication	25
3.2.5. Testler	26
3.3. Web Ön Uç Geliştirme	26
3.3.1. React.js ile Component Tabanlı Yapı	26
3.3.2. State Yönetimi - Redux Toolkit	26
3.3.3. Material UI ile Tasarım ve Kullanıcı Deneyimi	27
3.3.4. Veri Gösterimi ve Filtrelendirme	27
3.3.5. Formlar ve Doğrulama	28
3.3.6. API Entegrasyonu ve Axios Kullanımı	28
3.3.7. Güvenlik Önlemleri ve Kullanıcı Kimlik Doğrulama	28
3.4. Optik Karakter Tanıma ve Yapay Zeka Destekli Fatura Analizi	29
3.4.1. Tesseract	29
3.4.2. PaddleOCR	29
3.4.3. OpenCV	29
3.4.4. Makine Öğrenmesi	30
4. BULGULAR VE TARTIŞMA	30
4.1. Fatura Görsellerinden Optik Karakter Tanıma	30

4.1.1. Veri Ön İşleme Aşamalarının Değerlendirilmesi	33
4.1.2. Tesseract ve PaddleOCR ile Tarih Bilgisi Çıkarımı: Doğruluk, Tespit ve Hız Karşılaştırması	35
4.1.3. Faturalardan Toplam Fiyat Bilgilerinin Çıkarımı	37
4.2. Faturalardan Optik Karakter Tanıma Teknolojisi ile Okunan Faturalardan Fatura Kalemi Çıkarımı	38
4.3. Fatura Kalemi Kategorizasyonu	38
4.3.1. Kullanılan Veri Seti: Amazon Products Sales Dataset 2023	39
4.3.2. Veri Setinden Örneklem Alınması	42
4.3.3. Vektörizasyon	43
4.3.4. Model Seçimi ve Eğitimi	46
4.4. Veritabanı Üzerinden Yapılacak Veri Analizleri	51
4.5. API Geliştirme ve Veri Sunumu	52
5. KAYNAKÇA	53

ÖZET

KAREKOD İÇEREN FATURALARDAN BİLGİ ÇIKARTILMASI VE BU FATURALARIN SINIFLANDIRILMASI, ANALİZ EDİLMESİ

Bu bitirme projesi, kullanıcıların QR kod içeren faturalarını daha kolay analiz edilmesi ve saklanabilmesi amacıyla geliştirilmiştir. Optik Karakter Tanıma (OCR) teknolojisi kullanılarak, QR kodlu faturalardan veri okunmuş ve bu veriler bir veri tabanında saklanmıştır. Verilere erişimi sağlamak için bir API geliştirilmiş ve kullanıcıların verilerini bir web arayüzü üzerinden yönetebilmeleri sağlanmıştır.

Projenin önemli hedeflerinden biri, kullanıcıların önceki faturalarına dayalı olarak gelecekteki harcama düzenlerini tahmin edebilen yapay zeka (AI) destekli bir sistem entegre etmektir. Bu tahmin sistemi, kullanıcıların bütçelerini daha verimli bir şekilde yönetmelerine yardımcı olmayı amaçlamaktadır. Ayrıca, uygulamanın analiz kısmı grafiklerle desteklenmiş olup, kullanıcıların hem faturalarını tek tek görüntüleyebilmelerini hem de aylık harcamalarını grafikler aracılığıyla daha kolay analiz etmelerini sağlamaktadır.

Proje, front-end olarak React, back-end olarak PostgreSQL, API için Flask, OCR için PaddleOCR kullanmakta; kategorizasyon işlemi için makine öğrenmesi algoritmaları kullanılmaktadır.

Bu proje, kullanıcı dostu bir platform sunmanın yanı sıra fatura yönetiminde otomasyonu ve dijitalleşmeyi destekleyerek, kullanıcıların zaman kazanmalarını ve mali verilerini daha etkin bir şekilde yönetmelerini amaçlamaktadır.

Mart, 2025

Öğrenciler

Yunus Ege Küçük

Emir Binçe

Farid Bayramov

ABSTRACT

EXTRACTION OF INFORMATION FROM INVOICES CONTAINING QR CODES, CLASSIFICATION AND ANALYSIS OF THESE INVOICES

This graduation project has been developed to enable users to more easily analyze and store invoices containing QR codes. Using Optical Character Recognition (OCR) technology, data has been extracted from QR-coded invoices and stored in a database. An API has been developed to provide access to the data, allowing users to manage their data through a web interface.

One of the main goals of the project is to integrate an AI-powered system that can predict future spending patterns based on users' previous invoices. This prediction system aims to help users manage their budgets more efficiently. Additionally, the analysis section of the application is supported by graphs, enabling users to not only view their invoices individually but also analyze their monthly expenses more easily through charts.

The project uses React as front-end, PostgreSQL as back-end, Flask for API, PaddleOCR for OCR; machine learning algorithms are used for categorization.

This project aims to provide a user-friendly platform as well as support automation and digitalization in invoice management, saving users time and managing their financial data more effectively.

March, 2025

Students

Yunus Ege Küçük

Emir Binçe

Farid Bayramov

KISALTMALAR

AI	: artificial intelligence
API	: application programming interface
BERT	: bidirectional encoder representations from transformers
CPU	: central processing unit
GPU	: graphics processing unit
HTTPS	: hypertext transfer protocol secure
JWT	: JSON web token
NLP	: natural language processing
OCR	: optical character recognition
OpenCV	: open source computer vision library
ORM	: object to relational mapping
QR Code	: quick response code
RDBMS	: relational database management system
SSL	: secure sockets layer
TLS	: transport layer security
CSV	: comma separated values
TF-IDF	: term frequency – inverse document frequency
BoW	: bag of words
SVM	: support vector machine
NB	: naive bayes

GÖRSEL LİSTESİ

Görsel 1. Profil bilgileri ekranı	26
Görsel 2. Giriş ve kullanıcı oluşturma formları	27
Görsel 3. OCR araçlarının karşılaştırması	29
Görsel 4. Başarı-Yöntem Grafiği	33
Görsel 5. Tespit-OCR Aracı Grafiği	34
Görsel 6. Kategorilerin yüzdelik dağılımlarının pasta grafiği olarak gösterimi	42

TABLO LİSTESİ

Tablo 1. Tedarikçiler tablosu	17
Tablo 2. Kullanıcılar tablosu	18
Tablo 3. Faturalar tablosu	19
Tablo 4. Fatura kalemi tablosu	20
Tablo 5. Fatura görselleri üzerinde OCR araçlarının karşılaştırılması	31
Tablo 6. PaddleOCR ve Tesseract “date” verisi üzerine OCR sonuçları	35
Tablo 7. PaddleOCR ve Tesseract Tahmin Oranı - Zaman karşılaştırması	35
Tablo 8. PaddleOCR ile fatura görsellerinden “total” bilgisinin okunması sonuçları	36
Tablo 9. Amazon Products Sales Dataset 2023 veri setinin sahip olduğu sütunlar	39
Tablo 10. Kategorilerin yüzdelik dağılımları	40
Tablo 11. Farklı sayıda örnek içeren verilerle eğitilen Logistic Regression modellerinin performansları	42
Tablo 12. Names sütununun kısaltılmış hali ve orijinal hali ile eğitilen farklı Logistic Regression modellerinin performansları	43
Tablo 13. TF-IDF ve BoW vektörizasyon algoritmaları kullanılarak 11 bin, 22 bin ve 55 bin sample veri seti ile eğitilen Logistic Regression modellerinin performansları	45
Tablo 14. TF-IDF ve BoW vektörizasyon algoritmaları kullanılarak 220 bin sample veri seti ile eğitilen Logistic Regression modellerinin performansları	46
Tablo 15. 11 bin sample veri seti ile eğitilen farklı klasik makine öğrenmesi modellerinin performansları	47
Tablo 16. 55 bin sample veri seti ile eğitilen farklı klasik makine öğrenmesi modellerin performansları	48
Tablo 17. 220 bin sample veri seti ile eğitilen farklı klasik makine öğrenmesi modellerin performansları	49

Tablo 18. Klasik makine öğrenmesi algoritmalarının zaman karşılaştırması	51
Tablo 19. Amazon Products Dataset 2023 veri setinden BoW vektörizasyon yöntemi kullanılarak elde edilen veri ile eğitilmiş modellerin değerlendirilmesi	51

1. GİRİŞ

Günümüzde, bireyler ve şirketler tarafından yapılan alışverişlerin birçoğu dijital olmuşken, fiziksel faturaların düzenlenmesi, saklanması ve takip edilmesi hala büyük bir zorluk teşkil etmektedir. Faturalar, işletmeler ve kullanıcılar için önemli finansal veriler içerdiğinden, bunların düzenli bir şekilde saklanması ve yönetilmesi gerekmektedir. Ancak mevcut sistemler, fiziksel faturaların dijitalleştirilmesi ve bu verilerin standart bir formatta işlenmesi konusunda ciddi eksiklikler göstermektedir. Özellikle karekod içeren faturalar, farklı firmalar tarafından farklı formatlarda oluşturulmakta ve bu da kullanıcıların faturalarındaki verileri çıkarmalarını ve analiz etmelerini zorlaştırmaktadır. Karekod içeren faturaların işlenmesi ile ilgili en büyük zorluklardan biri, karekoddaki verilerin genellikle standart dışı bir biçimde saklanmasıdır. Her firma, kendi sistemine uygun şekilde karekod verilerini düzenler, bu da verilerin farklı yöntemlerle elde edilmesine yol açar. Örneğin, bazı firmalar karekodda sadece bir URL ya da fatura numarası saklarken, diğerleri daha karmaşık verilerle çalışmaktadır. Bu projeyle amaçlanan, kullanıcıların faturalarını dijital ortamda kolayca saklayabilecekleri ve bu faturalar üzerinden istatistiksel analizler yaparak öneriler alabilecekleri bir platform oluşturmaktır. Kullanıcılar, faturalarındaki verileri rahatça okuyup depolayabilecek, verileri analiz ederek tasarruf önerileri ya da finansal trendler hakkında geri bildirim alabileceklerdir.

1.1. Proje Çalışmasının Amacı ve Önemi

Mevcut sistemlerin çoğu, e-fatura ve dijital ödemeler üzerine odaklanmakta olup, fiziksel faturaların dijitalleştirilmesine yönelik bir çözüm sunmamaktadır. Bu projede, fiziksel faturaların dijital ortamda saklanabilmesi, içerdikleri verilerin okunabilmesi ve analiz edilebilmesi için güçlü bir çözüm önerilmektedir. Kullanıcılar, faturalarındaki verileri standart bir şekilde kaydedebilecek ve bu veriler üzerinde analizler yaparak finansal durumlarını daha iyi takip edebileceklerdir.

Proje, kullanıcıların farklı firmalardan gelen çeşitli formatlardaki faturalarını tek bir sistemde toplayarak verilerini düzenlemeyi ve analiz etmeyi mümkün kılacaktır. Böylece, faturaların karekod içeriğiyle birlikte, görsel verisi de OCR (Optik Karakter Tanıma) kullanılarak çıkarılacaktır. Bu süreç, fiziksel faturalarını dijital ortamda

kolayca yönetilmesini sağlar.

Bu projede kullanılacak teknolojiler arasında Python tabanlı bir back-end (Flask API), PostgreSQL veritabanı, React tabanlı front-end teknolojileri yer alacak; aynı zamanda PaddleOCR, Tesseract ve OpenCV gibi görüntü işleme kütüphaneleriyle de OCR işlemleri gerçekleştirilecektir. Optik karakter tanıma sonucunda elde edilen verilerin sınıflandırılması için ise makine öğrenimi modelleri kullanılacaktır. Kullanıcıların faturalarındaki tüm detaylara kolayca erişmeleri sağlanacaktır.

2. LİTERATÜR TARAMASI

Projemiz kapsamında, literatürde yer alan mevcut çalışmalara dayanarak ilgili alandaki bilgi birikimini genişletmek ve projenin teorik altyapısını güçlendirmek amacıyla üç farklı akademik araştırma detaylı bir şekilde incelenmiştir [1] [2] [3].

2.1. Türkçe Faturaların Sınıflandırılmasında Farklı Öznitelik Seçimi Yöntemleri ile Topluluk Öğrenme Algoritmalarının Etkilerinin İncelenmesi

Bu çalışma Türkçe faturaların sınıflandırılması amacıyla, hem doğal dil işleme hem de makine öğrenmesi tekniklerini topluluk öğrenmesi mimarisi içinde Serpme (Sprinkling) tekniğini de kullanan bu kapsamdaki ilk çalışmadır. Türkçe dilindeki faturaların otomatik olarak analiz edilmesi ve sınıflandırılması için literatürde bulunan çalışma yetersizliği bu çalışmada motivasyon kaynağı olmuştur. Bu çalışma ile Türkçe faturalar üzerine doğal dil işleme ve makine öğrenmesi yöntemleri ile sınıflandırma yapmak ve ayrıca farklı özellik seçimi yöntemlerinin bu sınıflandırma algoritmaları üzerindeki etkilerinin gözlemlenmesi amaçlanmıştır. Aynı zamanda çalışmada kullanılan veri kümeleri ve deney ortamı diğer araştırmacıların erişimine açık hale getirilerek literatüre katkı yapılması amaçlanmıştır.

Çalışmada sınıflandırma algoritmaları olarak DVM (Destek Vektör Makineleri), RO (Rassal Orman Algoritması), NB (Naif Bayes Algoritması), KNN (K-En Yakın Komşu Algoritması) ve Birleştirici Adaboost Algoritması kullanılmıştır. Ayrıca Serpme (Sprinkling) Tekniği kullanılarak Türkçe Faturalar DVM ve NB Algoritmaları ile sınıflandırılmıştır. Öznitelik değerlendirme yöntemleri olan Kazanç Oranı, Ki-Kare Özellik Değerlendirici, Bilgi Kazancı ve Geriye Doğru Eleme yöntemleri farklı

özniteliklerin önem sırasını belirlemek ve katkısını ölçmek amacıyla kullanılmıştır. Çalışmada veri seti olarak kullanılmak üzere 2 farklı veri kümesi hazırlanmıştır. İki veri kümesinde de fatura türlerine göre e-fatura, e-arşiv, irsaliyeli fatura, makbuz ve fatura değil olarak beş farklı sınıf bulunmaktadır. İlk veri kümesi 15000 adet fatura verisinden oluşmaktadır. Bu fatura verilerinden 3352 adet e-fatura, 3168 adet e-arşiv, 2873 adet irsaliyeli fatura, 2514 adet makbuz ve 3093 adet fatura olmayan veri bulunmaktadır. İkinci veri kümesi ise 50000 adet fatura verisinden oluşmaktadır. Bu fatura verilerinden 12152 adet e-fatura, 11359 adet e-arşiv, 9402 adet irsaliyeli fatura, 8264 adet makbuz ve 8824 adet fatura olmayan veri bulunmaktadır.

Veri kümesinde, fatura üzerinde bulunan verilerin anlamı daha net hale getirmek, gürültüyü azaltmak ve veri kümesini daha tutarlı hale getirmek amacıyla metin ön işleme yöntemleri kullanılmıştır. Veri ön işleme aşamasında veri kümesi üzerinde boşluklu yapıların kaldırılması, kelime normalizasyon işlemleri, noktalama işaretlerinin kaldırılması, sayıların kaldırılması ve fatura türünün belirlenmesinde etki etmeyecek fakat kelime frekans sayısı çok olan kelimelerin kaldırılması işlemleri gerçekleştirilmiştir. Bu aşamalardan sonra elde edilen veriler üzerinde tokenize işlemi, morfolojik operasyonlar, kök bulma ve kök normalizasyon işlemleri uygulanmıştır. Elde edilen veri kümesinin %60'ı eğitim, %20'si test ve %20'si validasyon veri kümesi olarak bölünmüştür.

Bu veri kümeleri üzerinde kazanç oranı, ki-kare, bilgi kazancı ve geri doğru eleme özellik seçim yöntemleri kullanılarak K-Yakın Komşu Algoritması, Rassal Orman, Naïf Bayes, DVM ve Birleşik (Ensemble) Adaboost algoritmalarının eğitimleri gerçekleştirilmiştir. 15000 adet faturadan oluşan veri kümesinde gerçekleştirilen eğitim sonucunda en yüksek F1 skoru Birleşik Adaboost ve Geriye Doğru Eleme özellik seçimi yöntemleri kullanılan modelde %96,11 olarak elde edilmiştir. İkinci en yüksek olarak bu değeri %95,91 ile Rassal Orman ve Geriye Doğru Eleme yöntemlerinin kullanıldığı model takip etmektedir. 50000 adet faturadan oluşan veri kümesi üzerinde yapılan eğitim sonucunda en yüksek başarı oranı Geriye Doğru Eleme özellik seçimi yöntemi ve Birleşik Adaboost kullanılarak %94.81 ile elde edilmiştir. Serpme (Sprinkling) tekniğinin etkisini ölçmek için Serpme tekniği kullanılarak DVM deney sonuçları ve geleneksel DVM deney sonuçları kıyaslanmıştır. F1 Skoru açısından incelendiğinde Serpme tekniği kullanılan DVM modelinde geleneksel DVM modeline

kıyasla 3,01 artış görülmektedir. Sınıflandırma başarımındaki bu artışın Serpme (Sprinkling) tekniğinin veri kümesinin vektörleşmesi aşamasında modele kattığı yeni bir etiketin, anlamsal boyutta sınıflandırma pozitif etkisi olmasından ötürü olabilir.

Sonuç olarak özellik seçimlerinden Geriye Doğru Eleme yöntemi kullanıldığında modeller en iyi sonuçları vermektedir. İki veri kümesinde uygulanan yöntemler incelendiğinde en yüksek doğruluk oranının Birleşik Adaboost ve Geriye Doğru Eleme yöntemi ile olduğu görülmüştür.

2.2. QR Kod Tanıma ve Bulut Depolama Tabanlı Çevrimiçi Faturalama Sistemi

“QR Kod Tanıma ve Bulut Depolama Tabanlı Çevrimiçi Faturalama Sistemi” araştırma makalesi, geleneksel faturalama yöntemlerindeki yaygın verimsizliklere modern bir çözüm sunmaktadır. Geleneksel sistemler genellikle vergi mükellefi kimlik numaraları, adresler ve banka bilgileri gibi temel işletme bilgilerinin manuel olarak girilmesini gerektirir. Bu süreçler hem zahmetli ve zaman alıcıdır hem de hatalara açık olup, verimliliği düşürmekte, müşteri memnuniyetini azaltmakta ve vergi politikalarının uygulanmasında zorluklar yaratmaktadır. Bu sorunları aşmak için yazarlar, faturaları daha hızlı ve daha güvenilir bir şekilde oluşturmak amacıyla QR kod teknolojisini ve bulut depolamayı entegre eden bir sistem önermektedir.

Yaklaşım, işletmelere vergi mükellefi kimlik numaralarına dayalı olarak benzersiz QR kodları atanmasını ve isimler, iletişim bilgileri ve banka bilgileri gibi ayrıntılı bilgilerin, vergi dairesinin sistemleriyle senkronize bir bulut veritabanında depolanmasını içerir. Bu veri seti, vergi dairesi tarafından yönetilen işletme bilgilerini içerir ve şu unsurları kapsar: işletme adı, vergi mükellefi kimlik numarası, kayıtlı adres, telefon numarası, banka adı ve hesap bilgileri. Bu bilgiler, bulut depolama sunucusunda saklanır ve sürekli olarak güncellenerek senkronize edilir. Satıcılar, bu QR kodlarını tarayarak alıcının bilgilerini anında alabilir; böylece manuel veri girişine gerek kalmaz ve hatalar azalır. Sistem, satıcının faturalama terminali ile bulut depolama sunucusu arasında kesintisiz bir etkileşim sağlarken veri güvenliği ve gizliliği sağlamak için güvenli iletişim protokolleri kullanır.

Bu sistem, bir dizi dikkat çekici avantaj sunmaktadır. Faturalama süresini önemli ölçüde kısaltarak bir dakikadan fazla süren işlemleri sadece 15 saniyeye indirir, böylece

verimliliği artırır. Otomasyon, insan hatalarını en aza indirerek ve veri bütünlüğünü sağlayarak doğruluğu artırır. Alıcılar da faydalanır; artık işletme bilgilerini hatırlamak veya sağlamak zorunda kalmazlar—satıcılar yalnızca dijital veya basılı bir QR kod tarayarak işlemi tamamlayabilir. Ayrıca, sistem düşük maliyetlidir ve faturalama sürecini basitleştirip düzenleyerek vergi düzenlemelerine uyumu teşvik eder.

QR kod tanıma ve bulut depolama gibi son teknoloji çözümleri kullanan bu sistem, geleneksel faturalama uygulamalarını modernize etmektedir. Sadece verimliliği ve doğruluğu artırmakla kalmaz, aynı zamanda hem işletmeler hem de vergi otoriteleri için kullanım kolaylığı sağlayarak gelecekte daha gelişmiş vergi yönetimi ve operasyonel sistemler için bir yol açar.

2.3. Extraction of Information from Invoices – Challenges in the Extraction Pipeline

Bu makale, iş süreçleri için faturalardan bilgi çıkarma sürecinde karşılaşılan zorlukları ve çözüm önerilerini ele almaktadır. Farklı düzen ve dillerdeki faturalardan yapısal ve dijital veri elde etmek, işletmeler için değerli bir süreçtir; ancak bu süreç çeşitli engeller içermektedir.

Faturalardan bilgi çıkarma, işletmelerin veri analitiği, süreç otomasyonu ve denetim gibi birçok alanda fayda sağlamasına olanak tanır. Buna rağmen, fatura düzenlerinin çeşitliliği, dil farklılıkları ve veri gizliliği sorunları süreci zorlaştırmaktadır. Mevcut çözümler genellikle belirli düzenlere ve dillere özel olarak geliştirilmiştir. Bu durum, genelleştirilebilir bir sistem tasarlamayı güçleştirmektedir. Ayrıca, mevcut yöntemler çoğunlukla manuel etiketleme gibi zaman alıcı işlemler gerektirir ve bu da maliyeti artırır.

Makale, faturalardan bilgi çıkarma sürecinde bir veri işleme boru hattının (pipeline) tasarımında karşılaşılan sorunları analiz etmekte ve bu süreci daha etkili hale getirmek için bir çerçeve sunmaktadır. Araştırmacılar, bu doğrultuda yedi farklı bilgi türünü (segmental, sentaktik, semantik, uzamsal, dışsal, grafiksel, mantıksal) kategorize ederek sistematik bir yaklaşım geliştirmiştir. Ayrıca, süreçte kullanılan genel adımları ve karşılaşılan zorlukları tanımlamıştır. Bu kapsamda, tasarım bilimi araştırma yöntemi (Design Science Research, DSR) kullanılarak hem teorik hem de pratik bir çerçeve

oluşturulmuştur. Çalışma, mevcut yöntemlerin eleştirel bir incelemesini yaparak eksiklikleri ortaya koymuş ve daha kapsamlı bir pipeline tasarımı için rehber niteliğinde katkılar sunmuştur.

Araştırmada, Almanca faturalardan oluşan gerçek dünya veri kümesi kullanılmıştır. Bu veri kümesi, 977 PDF dosyasından oluşmakta ve 60'tan fazla sınıf etiketi içermektedir. Öne çıkan etiketler arasında fatura tarihi, toplam tutar, ödeme bilgileri ve IBAN gibi unsurlar bulunmaktadır. Segmentasyon işlemi kelime düzeyinde gerçekleştirilmiş ve veri kümesi, manuel etiketleme yöntemiyle hazırlanmıştır.

Önerilen pipeline, Almanca faturalardan bilgi çıkarmak için test edilmiştir. Prototip model, %82.3 F1 skoru elde etmiştir. Ancak bu skor, referans alınan bir İngilizce modeline (%90.5 F1 skoru) kıyasla daha düşüktür. Performans farkının, kullanılan kelime modelindeki ve komşuluk algoritmasındaki farklılıklardan kaynaklandığı belirtilmiştir.

Sonuç olarak, bu çalışma, faturalardan bilgi çıkarma süreçlerinin optimize edilmesi ve mevcut çözümlerin iyileştirilmesi için pratik bir rehber sunmaktadır.

3. PROJE TASARIMI

3.1. Veritabanı Tasarımı

Bu bölümde, projemizin veritabanı tasarımını ve verilerin yönetilmesini ele alacağız. Bu veritabanı, faturaların saklanması, kullanıcıların yönetilmesi, faturalarla ilgili analizlerin yapılabilmesi ve sistemin genel işleyişi için gereksinimleri karşılamak üzere tasarlanmıştır. Veritabanı tasarımı, kullanıcı verilerinin güvenli bir şekilde saklanması, faturaların düzenli bir biçimde kaydedilmesi ve ilişkilendirilmesi gibi önemli unsurları içermektedir.

3.1.1. PostgreSQL

PostgreSQL, açık kaynaklı ve nesne ilişkisel veritabanı yönetim sistemi (RDBMS) olup, SQL (Structured Query Language) dili ile veri yönetimi sağlar. Yüksek düzeyde güvenilirlik ve sağlamlık sunan bu sistem, geniş veri kümeleri üzerinde karmaşık sorguların işlenmesini etkin bir şekilde gerçekleştirebilecek şekilde tasarlanmıştır.

PostgreSQL, ACID (Atomicity, Consistency, Isolation, Durability) uyumlu olup, veritabanı işlemlerinin güvenli ve tutarlı bir şekilde gerçekleşmesini sağlar. Ayrıca, hem ilişkisel veri modelini hem de NoSQL özelliklerini destekleyerek, esnek veri yapıları ve büyük ölçekli veritabanı uygulamaları için uygun bir çözüm sunmaktadır.

3.1.2. Tabloların Yapısı

Vendors (Tedarikçiler) Tablosu:

Tedarikçi bilgilerini saklamak için kullanılan bu tablo, faturaların ilişkilendirildiği tedarikçilerin verilerini içerir. Tedarikçi adı, adresi, ülke ve telefon numarası gibi bilgileri tutar. Bu tabloya eklenen veriler, her fatura kaydıyla ilişkilendirilen tedarikçiyi belirtmek için kullanılır.

Vendors		
id	SERIAL	PRIMARY KEY
name	VARCHAR(255)	NOT NULL
address	TEXT	
country	VARCHAR(100)	
phone	VARCHAR(50)	

Tablo 1. Tedarikçiler tablosu

- **id**: Her tedarikçiye özgü benzersiz bir kimlik numarası.
- **name**: Tedarikçi adı.
- **address**: Tedarikçinin adresi.
- **country**: Tedarikçinin bulunduğu ülke.
- **phone**: Tedarikçi telefon numarası.

Bu tablo, tedarikçi bilgilerini yönetir ve diğer tablolarda tedarikçi ile ilişkilendirilmiş veri girişlerine olanak tanır. Tedarikçi bilgileri, faturaların doğru bir şekilde eşleştirilmesi ve analiz edilmesi için kritik bir rol oynar.

Users (Kullanıcılar) Tablosu:

Kullanıcıların kimlik bilgilerini ve hesap verilerini içeren bu tablo, her kullanıcının benzersiz bir kimliğiyle ilişkili çeşitli bilgilere sahiptir. Kullanıcılar, faturalarını sisteme yükleyen ve analizleri yapan kişiler olduğundan, bu verilerin güvenli ve düzenli bir şekilde saklanması gerekmektedir.

Users		
id	SERIAL	PRIMARY KEY
name	VARCHAR(255)	NOT NULL
surname	VARCHAR(255)	NOT NULL
username	VARCHAR(255)	NOT NULL UNIQUE
email	VARCHAR(255)	NOT NULL UNIQUE
password_hash	TEXT	NOT NULL
gender	VARCHAR(10)	
date_of_birth	DATE	NOT NULL
created_at	TIMESTAMP	

Tablo 2. Kullanıcılar tablosu

- **id**: Kullanıcının benzersiz kimlik numarası.
- **name, surname**: Kullanıcının adı ve soyadı.
- **username**: Kullanıcının giriş için kullandığı benzersiz kullanıcı adı.
- **email**: Kullanıcının e-posta adresi.
- **password_hash**: Kullanıcının şifre hash'i. Güvenlik için şifre düz metin olarak saklanmaz.
- **gender**: Kullanıcının cinsiyeti (isteğe bağlı).
- **date_of_birth**: Kullanıcının doğum tarihi.
- **created_at**: Kullanıcının hesap oluşturma tarihi ve saati.

Bu tablo, kullanıcıların sisteme giriş yapmasını sağlayacak temel bilgilere sahiptir. username ve email alanları üzerinde benzersiz kısıtlamalar bulunmaktadır, bu da veri doğruluğunu sağlar.

Invoices (Faturalar) Tablosu:

Faturaların kaydedildiği ana tablodur. Bu tabloda, her faturanın hangi tedarikçiye ait olduğu, hangi kullanıcı tarafından yüklendiği, faturanın tarihi, tutarı gibi önemli bilgiler saklanır. Aynı zamanda, her fatura için karekod verileri de burada tutulur.

Invoices		
id	SERIAL	PRIMARY KEY
vendor_id	INT	REFERENCES vendors(id) ON DELETE SET NULL
user_id	INT	NOT NULL REFERENCES users(id) ON DELETE CASCADE
date	DATE	NOT NULL
total_amount	DECIMAL(10,2)	NOT NULL
currency	VARCHAR(10)	
qr_data	TEXT	
created_at	TIMESTAMP	

Tablo 3. Faturalar tablosu

- **id**: Faturanın benzersiz kimlik numarası.
- **vendor_id**: İlgili tedarikçi ID'si. Bu alan, faturanın hangi tedarikçiye ait olduğunu belirler.
- **user_id**: Faturayı sisteme yükleyen kullanıcının ID'si.
- **date**: Faturanın kesildiği tarih.
- **total_amount**: Faturanın toplam tutarı.
- **currency**: Faturanın para birimi.
- **qr_data**: Faturanın karekodundan çıkarılan ham veri.
- **created_at**: Faturanın sisteme yüklendiği tarih ve saat.

Bu tablo, faturaların yönetilmesi ve sorgulanması için temel bir yapı sunar. Ayrıca user_id ile kullanıcılar ve vendor_id ile tedarikçiler arasında güçlü ilişkiler kurulur.

Invoice Items (Fatura Kalemleri) Tablosu:

Faturaların detaylı bir şekilde saklanmasını sağlar. Her fatura için birden fazla kalem (ürün ya da hizmet) bulunabilir, bu nedenle her bir fatura kalemi ayrı bir satırda saklanır.

Invoice Items		
id	SERIAL	PRIMARY KEY
invoice_id	INT	NOT NULL REFERENCES invoices(id) ON DELETE CASCADE
description	TEXT	NOT NULL
quantity	DECIMAL(10,2)	NOT NULL CHECK (quantity > 0)
unit_price	DECIMAL(15,2)	NOT NULL CHECK (unit_price >= 0)
total_price	DECIMAL(15,2)	GENERATED ALWAYS AS (quantity * unit_price) STORED
category	VARCHAR(100)	

Tablo 4. Fatura kalemi tablosu

- **id**: Fatura kaleminin benzersiz kimlik numarası.
- **invoice_id**: İlgili faturanın ID'si. Bu alan, kalemin hangi faturaya ait olduğunu belirtir.
- **description**: Fatura kaleminin açıklaması (örneğin, ürün adı veya hizmet açıklaması).
- **quantity**: Ürünün ya da hizmetin miktarı.
- **unit_price**: Ürünün ya da hizmetin birim fiyatı.
- **total_price**: Kalemin toplam tutarı (miktar * birim fiyat). Bu alan, generated always as özelliği ile otomatik olarak hesaplanır.

- **category:** Ürün veya hizmet kategorisi.

Bu yapı, faturaların her bir kalemini ayrıntılı bir şekilde saklamamıza olanak tanır, böylece detaylı raporlar ve analizler oluşturulabilir.

3.1.3. Fonksiyonlar ve İşlem Akışları

Veritabanının işlevselliğini artıran fonksiyonlar, kullanıcılar ve faturalar üzerinde işlem yaparken güvenlik ve doğruluk sağlamak amacıyla kullanılır. Örnek olarak, kullanıcı oluşturma, güncelleme, silme, fatura ekleme ve silme işlemleri için yazılmış fonksiyonlar bulunmaktadır. Bu fonksiyonlar, verilerin doğru ve tutarlı bir şekilde işlenmesini sağlayarak iş süreçlerini kolaylaştırır.

- **create_user():** Yeni kullanıcılar eklerken, kullanıcı adı ve e-posta adresinin benzersizliğini kontrol eder ve kullanıcıyı veritabanına ekler.
- **verify_user_login():** Kullanıcı adı ya da e-posta adresi ile giriş yapan kullanıcının doğrulamasını yapar.
- **update_user():** Kullanıcı bilgilerini güncellerken, e-posta adresinin benzersizliğini kontrol eder.
- **delete_user():** Kullanıcıyı veritabanından siler ve bu kullanıcının faturalarını da siler. Silinen kullanıcı ile ilişkili olmayan tedarikçiler de veritabanından temizlenir.
- **insert_invoice():** Yeni bir fatura eklerken, tedarikçi bilgilerini kontrol eder ve gerekirse yeni bir tedarikçi ekler. Fatura kalemleri, JSON formatında sistemde saklanarak işlem kolaylığı sağlanır.
- **delete_invoice():** Fatura silinirken, ilişkilendirilen fatura kalemleri ve tedarikçi verileri de temizlenir.

3.1.4. Veritabanı ve Normalizasyon

Veritabanı, Normalization (Normalizasyon) ilkelerine uygun olarak tasarlanmıştır. Bu, veri tekrarını minimize etmek ve veri tutarlılığını sağlamak için önemlidir. Ayrıca, tablolar arasında güçlü ilişkiler kurularak veri bütünlüğü korunur.

- **Foreign Key** ilişkileri:

- **invoices** tablosu, **vendors** ve **users** tabloları ile ilişkilidir.
- **invoice_items** tablosu ise her bir kalemin hangi faturaya ait olduğunu belirtmek için **invoices** tablosu ile ilişkilidir.
- **ON DELETE CASCADE**: Fatura ve kullanıcı silindiğinde ilişkili verilerin (fatura kalemleri, tedarikçi vb.) otomatik olarak silinmesi sağlanır.

3.1.5. Performans ve Güvenlik

Veritabanı tasarımı, işlem hızı ve veri güvenliği göz önünde bulundurularak yapılmıştır. Kullanıcı bilgileri, şifrelerin güvenli bir şekilde saklanabilmesi için password_hash formatında saklanır. Ayrıca, kullanıcı adı ve e-posta adresinin benzersizliği sağlanarak veri tutarlılığı garantilenir.

Veritabanı sorguları ve fonksiyonlar, optimize edilmiş ve güvenli şekilde tasarlanmıştır. Veri girişlerinde yapılan doğrulamalar, sistemin hatalı veri girişi önlemesini sağlar.

3.2. API Tasarımı ve Arka Uç Geliştirme

Bu proje, kullanıcıların hesap oluşturmaya, giriş yapmaya, fatura eklemesine, faturaları yönetmesine ve kullanıcı hesaplarını yönetmesine olanak tanıyan bir API sunmaktadır. API, RESTful mimariyi kullanarak çeşitli endpoint'ler ve HTTP metodlarıyla kullanıcı etkileşimini yönetir. API'nin temel bileşenlerini ve nasıl çalıştığını şu şekilde detaylandırabiliriz:

3.2.1. Endpointler ve Kullanılan HTTP Request Türleri

API, çeşitli kullanıcı işlemleri ve fatura yönetimi için farklı endpoint'lere sahiptir. Aşağıda, projenin temel endpoint'leri ve bu endpoint'ler ile gerçekleştirilen HTTP işlemleri yer almaktadır:

- **Kullanıcı işlemleri:**
 - **POST /users/add_user**: Kullanıcı kaydı işlemi. Bu endpoint, yeni bir kullanıcı oluşturulmasını sağlar.
 - **Request Type**: POST
 - **Body**: JSON formatında kullanıcının bilgileri (isim, soyisim,

kullanıcı adı, email, vb.)

- **POST /users/login:** Kullanıcı girişi işlemi. Kullanıcı adı ya da email ve şifre ile giriş yapılır.
 - **Request Type:** POST
 - **Body:** JSON formatında kullanıcı bilgileri
- **DELETE /users/delete_user/{user_id}:** Kullanıcıyı silme işlemi.
 - **Request Type:** DELETE
 - **Body:** Kimlik doğrulama için JWT token header'da gönderilir.
- **Fatura işlemleri:**
 - **POST /invoices/add_invoice:** Yeni bir fatura ekler.
 - **Request Type:** POST
 - **Body:** JSON formatında fatura verileri (fatura tarihi, tutar, ödeme bilgileri, satılan ürünler)
 - **DELETE /invoices/delete_invoice/{invoice_id}:** Bir faturayı siler.
 - **Request Type:** DELETE
 - **Body:** Kimlik doğrulama için JWT token header'da gönderilir.
 - **GET /invoices/get_invoices:** Kullanıcıya ait tüm faturaları listeler.
 - **Request Type:** GET
 - **Body:** Kimlik doğrulama için JWT token header'da gönderilir.
 - **POST /invoices/process_qr:** Karekodu okuyup fatura verisini saklar.
 - **Request Type:** POST
 - **Body:** JSON formatında karekod verisi (URL)

Bu endpoint'ler, kullanıcılara işlemlerini gerçekleştirmek için gerekli araçları sunmaktadır. API, RESTful prensiplere göre tasarlanmış ve HTTP metodları doğru şekilde kullanılmıştır.

3.2.2. Veri Kontrolü ve Validasyonu

API'nin doğru çalışabilmesi için gelen verilerin belirli kurallara uygunluğu denetlenir. Bu, kullanıcıların ve sistemin güvenliği için kritik öneme sahiptir. Veri doğrulama, aşağıdaki gibi gerçekleştirilir:

- **Kullanıcı kaydı (Add User):** Kullanıcıdan gelen verilerde eksiklik veya hatalı

veri olup olmadığı kontrol edilir. Örneğin, email ve password gibi kritik alanların boş olmaması gerektiği gibi.

- **Giriş (Login):** Giriş yapılırken kullanıcı adı ya da e-mail ve şifrenin doğru formatta olup olmadığı kontrol edilir.
- **Fatura ekleme (Add Invoice):** Fatura ekleme işlemi sırasında total_amount, currency, ve items gibi alanların doğru formatta olup olmadığı kontrol edilir. Eksik veya hatalı veri gönderildiğinde, API uygun hata mesajı döner.

Veri doğrulama, kullanıcı deneyimini iyileştiren, hata risklerini azaltan ve sistemi güvenli hale getiren bir özelliktir.

3.2.3. Veri Güvenliği ve Veritabanı Saldırılarına Karşı Güvenlik

API, güvenliği en üst seviyeye çıkarmak için SQL injection gibi veritabanı saldırılarına karşı koruma sağlanır. SQL sorguları, ORM (Object-Relational Mapping) aracı ile güvenli bir şekilde oluşturulur, doğrudan kullanıcıdan alınan verilerle SQL sorguları yapılmaz. Kullanıcı şifreleri, veritabanına kaydedilmeden önce güçlü bir şifreleme algoritması ile hash'lenir. Bu sayede, şifreler düz metin olarak saklanmaz ve veritabanı ele geçirilse bile kullanıcı şifreleri güvenli bir şekilde korunur. API, veri iletiminde güvenliği sağlamak için HTTPS (SSL/TLS) kullanır. Bu, verilerin şifreli bir şekilde iletilmesini ve kötü niyetli kullanıcıların ağ üzerinden veri çalmalarını engeller.

3.2.4. JWT Kullanımı ve Kullanıcı Authentication

Kullanıcıların sisteme güvenli bir şekilde giriş yapabilmesi için JSON Web Token (JWT) kullanılır. JWT, kullanıcının kimliğini doğrulamak için bir token oluşturur. Bu token, kullanıcının sisteme giriş yaptıktan sonra her istekte gönderilir ve kimlik doğrulama için kullanılır. JWT kullanımının avantajları:

- **Token tabanlı kimlik doğrulama:** Kullanıcı, giriş yaptıktan sonra bir JWT alır. Bu token, sonraki API isteklerinde kimlik doğrulama amacıyla header'da gönderilir.
- **Token geçerliliği:** JWT belirli bir süre için geçerlidir, bu sayede oturum süresi kontrol edilebilir. Token geçerliliği bittiğinde kullanıcıdan yeniden giriş yapması istenir.

- **Güvenli Veri İletimi:** JWT, veriyi şifreler ve istemci-sunucu arasındaki iletişimin güvenli bir şekilde sağlanmasını sağlar.

3.2.5. Testler

API'nin doğru çalıştığını ve hata durumlarını düzgün bir şekilde yönettiğini test etmek için çeşitli testler yazılmıştır. Bu testler, API'nin her özelliğini kapsayan senaryoları içerir:

- **Kullanıcı işlemleri:** Kullanıcı kaydı, giriş, ve kullanıcı silme işlemleri başarıyla test edilmiştir. Verilerin doğruluğu, eksik alanların kontrolü ve hata mesajlarının doğru döndüğü test edilmiştir.
- **Fatura işlemleri:** Fatura ekleme, fatura silme ve fatura listeleme işlemleri test edilmiştir. Fatura verilerinin doğruluğu, eksik alanlar, ve kullanıcı doğrulama işlemleri başarılı bir şekilde test edilmiştir.
- **Güvenlik testleri:** Hatalı veya eksik token ile yapılan işlemler, 401 Unauthorized hatası ile engellenmiştir.

Testler, API'nin her durum altında düzgün çalıştığını ve tüm endpoint'lerin beklenen şekilde yanıt verdiğini doğrular.

3.3. Web Ön Uç Geliştirme

Web ön uç geliştirme sürecinde, kullanıcı dostu ve işlevsel bir arayüz tasarımı hedeflenmiştir. Bu süreç, HTML, CSS ve TypeScript web teknolojileri ile yapılmış, modern framework'ler kullanılarak uygulama geliştirilmiştir. Proje için React.js, Redux Toolkit ve Material UI gibi güçlü araçlar tercih edilmiştir.

3.3.1. React.js ile Component Tabanlı Yapı

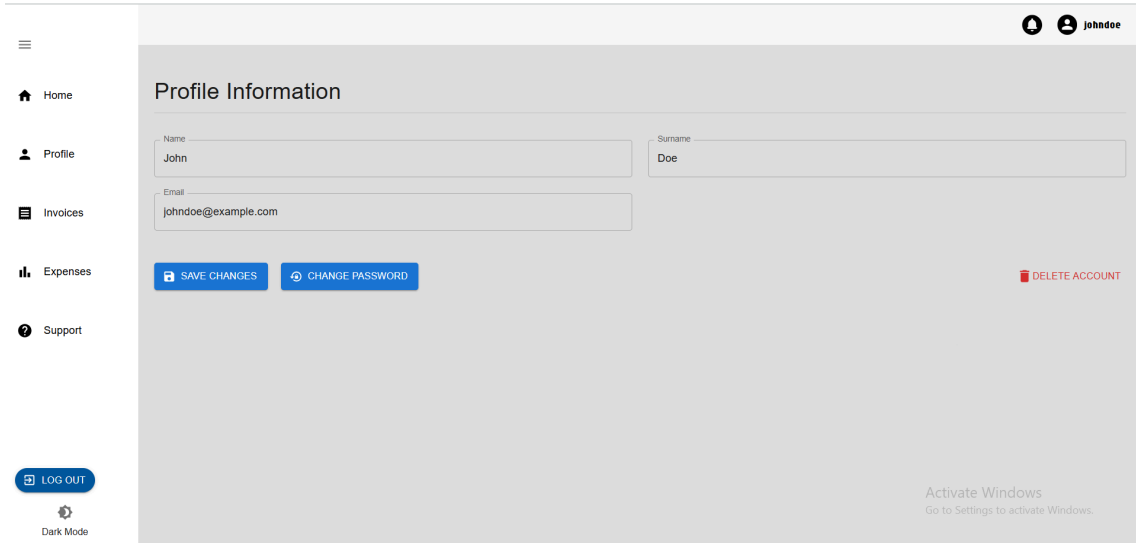
React.js, uygulamanın dinamik ve etkileşimli bileşenler aracılığıyla yönetilmesini sağlamaktadır. Web uygulamasındaki her sayfa ve özellik, bağımsız component'ler olarak geliştirilmiş ve React Router ile sayfalar arasında geçiş sağlanmıştır. Bu yapı, geliştirme sürecinde esneklik ve yeniden kullanılabilirlik sağlamıştır.

3.3.2. State Yönetimi - Redux Toolkit

Uygulama genelinde state yönetimi için Redux Toolkit kullanılmıştır. Kullanıcı oturumu, kullanıcı profili ve dinamik veri yönetimi gibi önemli state'ler Redux üzerinden yönetilmektedir. Redux Toolkit, özellikle kodun sadeleştirilmesine ve performansın artırılmasına yardımcı olmuştur. Kullanıcı giriş işlemleri ve faturaların listeleme, filtreleme gibi işlemleri Redux aracılığıyla verimli bir şekilde gerçekleştirilmiştir.

3.3.3. Material UI ile Tasarım ve Kullanıcı Deneyimi

Web uygulamasında görsel tasarım için Material UI kullanılmıştır. Material UI, modern ve kullanıcı dostu bir arayüz sağlamak adına esnek bileşenler ve tema seçenekleri sunmuştur. Renk paleti, tipografi ve responsive (duyarlı) tasarım özellikleri, kullanıcıların mobil ve masaüstü cihazlarda sorunsuz bir deneyim yaşamasını sağlamıştır. Özellikle fatura verilerinin görüntülediği sayfalarda, tablolar, düğmeler ve formlar gibi UI bileşenleri Material UI ile tasarlanmıştır.



Görsel 1. Profil bilgileri ekranı

3.3.4. Veri Gösterimi ve Filtrelendirme

Web ön uç geliştirmesinde, JSON verilerinin görselleştirilmesi önemli bir yer tutmaktadır. Fatura verilerinin gösterimi için DataGrid bileşeni kullanılmıştır. Ayrıca, kullanıcıların verileri filtrelemesine olanak tanımak için arama ve filtreleme işlevleri eklenmiştir. Bu sayede kullanıcılar, istedikleri tarih aralığında ya da belirli kriterlere

göre verilerini hızlıca görüntüleyebilmekte ve analiz edebilmektedir.

3.3.5. Formlar ve Doğrulama

Fatura ekleme ve kullanıcı kaydı işlemleri gibi formlar, react-hook-form ve YUP kütüphaneleri ile geliştirilmiştir. Bu sayede form doğrulama işlemleri kolayca yapılmış, kullanıcı deneyimi artırılmıştır. Örneğin, kullanıcı giriş ve ya hesap oluşturma işlemi yaparken zorunlu alanların doldurulması ve geçerli veri formatlarının kontrol edilmesi sağlanmıştır.

The image shows two side-by-side form mockups. The left form is titled 'Login' and has two input fields: 'Email or Username' and 'Password'. Below the 'Email or Username' field is a red error message 'Please enter a valid username or email.' Below the 'Password' field is a red error message 'Password must be at least 6 characters'. There is a blue 'LOGIN' button and a checkbox for 'Save password'. At the bottom, there is a link 'Don't have an account?' and a 'CREATE AN ACCOUNT' button. The right form is titled 'Create an Account' and has five input fields: 'Name', 'Surname', 'Username', 'Gender', and 'Doğum Tarihi' (Date of Birth). Each field has a red error message below it: 'Name is required', 'Surname is required', 'Username is required', 'Gender is required', and 'Doğum tarihi zorunludur'. The 'Doğum Tarihi' field has a date picker icon.

Görsel 2. Giriş ve kullanıcı oluşturma formları

3.3.6. API Entegrasyonu ve Axios Kullanımı

Faturaların veritabanından alınması ve kaydedilmesi işlemleri için Axios kütüphanesi kullanılarak API entegrasyonu yapılmıştır. Kullanıcılar, giriş yaptığında, fatura bilgilerini form aracılığıyla girdiğinde, bu veriler sunucuya POST isteği ile gönderilmektedir. Aynı şekilde, fatura listesi ve kullanıcı bilgileri GET isteği ile alınmaktadır.

3.3.7. Güvenlik Önlemleri ve Kullanıcı Kimlik Doğrulama

Kullanıcı oturumu açma ve kimlik doğrulama işlemleri için JWT (JSON Web Token) kullanılmıştır. Bu, kullanıcıların güvenli bir şekilde sisteme giriş yapmalarını sağlamaktadır. Ayrıca, oturum süresi sonunda kullanıcılar yeniden kimlik doğrulama

işlemi yapmak zorundadırlar, bu da uygulamanın güvenliğini artırmaktadır.

3.4. Optik Karakter Tanıma ve Yapay Zeka Destekli Fatura Analizi

Optik Karakter Tanıma (OCR), belgelerde yer alan metinleri dijital ortama aktarmak ve bu metinler üzerinde işlem yapabilmek için kullanılan bir teknolojidir. Projemizde OCR araçları olarak Tesseract ve PaddleOCR incelenmiş, çeşitli kriterlere göre karşılaştırmalar yapılmış ve her iki aracın performansı değerlendirilmiştir. Yapay zeka implementasyonu için çeşitli makine öğrenmesi teknikleri kullanılmıştır.

3.4.1. Tesseract

Tesseract, açık kaynaklı bir Optik Karakter Tanıma (OCR) motorudur. İlk olarak HP tarafından bir PhD projesi olarak başlatılan [4] Tesseract, 1984 ve 1994 yılları arasında HP Labs OCR adı altında geliştirilmiştir ve “1995 UNLV The 4th Annual Test of OCR Accuracy” çalışmasında test edilip popülerite kazanmıştır [5]. 2005 yılının sonlarına doğru HP tarafından kodları açık kaynak olarak paylaşılıp projenin geliştirilmesinin durdurulmasının ardından Google, projeyi geliştirmeye devam etmek için, 2006 yılında projeyi devralmıştır ve 2007 yılında projeyi detaylı olarak tanıtan bir makale yayınlamıştır [6]. Google, Tesseract OCR projesini günümüzde halen aktif olarak geliştirmektedir ve şu anda Tesseract, OCR teknolojileri piyasasında popüler bir konumdadır.

3.4.2. PaddleOCR

PaddleOCR, yapay zeka alanında araştırma ve geliştirme faaliyetleri yürüten Çin merkezli ve Çin'deki ilk bağımsız Ar-Ge derin öğrenme platformu PaddlePaddle tarafından geliştirilen, açık kaynak kodlu bir optik karakter tanıma modelidir [7]. PaddleOCR, 14 Mayıs 2020'de piyasaya sürülmesinden bu yana sürekli olarak optimize edilmiş, iyileştirilmiş ve güncellenmiştir [8].

3.4.3. OpenCV

OpenCV (Open Source Computer Vision Library), 1999 yılında Intel firması altında Gary Bradski tarafından başlatılan ve ilk sürümü 2000 yılında yayınlanan, bilgisayarla görme ve görüntü işleme alanlarında kullanılan açık kaynaklı bir yazılım kütüphanesidir

[9]. Görüntü ve video analizi, nesne tanıma, yüz tanıma, hareket izleme, görüntü düzenleme ve daha birçok bilgisayarla görme işlemi için araçlar sunar. Gerçek zamanlı video işleme, 2D/3D özellikler çıkarımı gibi geniş bir uygulama yelpazesi sunar. OpenCV, özellikle yapay zeka ve makine öğrenmesi ile entegre edilen projelerde sıklıkla tercih edilen bir araçtır.

3.4.4. Makine Öğrenmesi

Makine öğrenmesi (ML), bilgisayar sistemlerinin açıkça programlanmaksızın belirli görevleri yerine getirebilmesini sağlayan algoritmalar ve istatistiksel modeller sayesinde makinen verilerden öğrenmesini ve bu öğrenimle belirli görevleri yerine getirmesini sağlayan yapay zeka alt alanıdır [10].

4. BULGULAR VE TARTIŞMA

Bu bölümde projemiz üzerinde çalışırken geçtiğimiz aşamalar yer almaktadır. OCR yöntemlerinin deneysel olarak uygulanması, karşılaştırılması ve elde edilen sonuçların görselleştirilip tablo haline getirilmesi sağlanmıştır. Büyük dil modelleri ve makine öğrenmesi tekniklerinin projemize nasıl dahil edildiği açıklanmıştır.

4.1. Fatura Görsellerinden Optik Karakter Tanıma

Fatura Görsellerinden anlamlı veri elde edilmesi amacıyla kullanılan optik karakter tanıma (OCR) teknolojilerinden açık kaynaklı iki OCR kütüphanesi olan Tesseract ve PaddleOCR araçlarının karşılaştırması gerçekleştirilmiştir. Tesseract, PaddleOCR, EasyOCR ve MMOCR teknolojileri bilgisayar ortamında yazılmış bir Request metninin okunmasında ve anlamlı verilere dönüştürülmesinde kullanılmıştır. Bu teknolojiler arasındaki karşılaştırmalar sonucunda, PaddleOCR en iyi sonucu vererek daha doğru ve tutarlı veri çıkarmada başarılı olmuştur. PaddleOCR, özellikle karmaşık yapıdaki metinlerde daha yüksek doğruluk oranlarıyla öne çıkmaktadır.

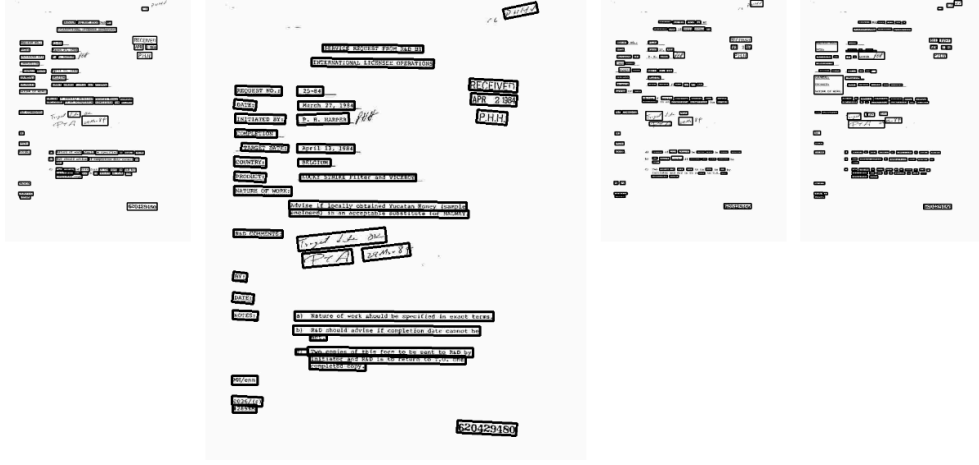
Text detection

Detection with EasyOCR

Detection with PPOCR

Detection with MMOCR

Detection with Tesseract






Görsel 3. OCR araçlarının karşılaştırması

Yukarıdaki görselde de görülebileceği gibi, PaddleOCR metindeki neredeyse tüm satırları, hatta farklı açılarda yazılmış imza gibi öğeleri bile doğru bir şekilde algılayarak diğer OCR araçlarına kıyasla üstün performans sergilemiştir.

Fatura görüntülerinde test etmek amacıyla PaddleOCR ve Tesseract görüntüler üzerinde uygulanıp sonuçlar karşılaştırılmıştır.

Fatura Görselleri Üzerinde OCR Araçlarının Karşılaştırılması

Original	PaddleOCR	Tesseract
 <p>Give us feedback @ survey.walmart.com Thank you! ID #: 78655MCF22</p> <p>Walmart 949-498-6669 Mor: MICHAEL 951 AVENIDA PICO SAN CLEMENTE CA 92673</p> <p>ST# 02527 OP# 009045 TE# 45 TR# 06193 GY OATMEAL 007874243408 F 1.76 0 OT 200Z TOM 081236803115 6.74 X M ATHLETICS 019104967781 24.97 X DEXAS 15X20 008429710921 12.97 X SUBTOTAL 46.44 TAX 1 7.750 X 3.46 TOTAL 49.90 DEBIT TEND 49.90 CHANGE DUE 0.00</p> <p>EFT DEBIT PAY FROM PRIMARY 49.90 TOTAL PURCHASE US DEBIT ***** 9689 I 0 REF # 029200639128 NETWORK ID: 0056 APPR CODE 461500 US DEBIT AID A0000000980840 AAC 7756F4E82B54EED TERMINAL # 50011004 10/18/20 11:30:42 # ITEMS SOLD 4 TOM 2400 4776 4305 8847 1471</p> <p>W+ Introducing Walmart+ Join today at walmart.com/plus Low Prices You Can Trust. Every Day. 10/18/20 11:30:46</p>	 <p>Give us feedback @ survey.walmart.com Thank you! ID #: 78655MCF22</p> <p>Walmart 949-498-6669 Mor: MICHAEL 951 AVENIDA PICO SAN CLEMENTE CA 92673</p> <p>ST# 02527 OP# 009045 TE# 45 TR# 06193 GY OATMEAL 007874243408 F 1.76 0 OT 200Z TOM 081236803115 6.74 X M ATHLETICS 019104967781 24.97 X DEXAS 15X20 008429710921 12.97 X SUBTOTAL 46.44 TAX 1 7.750 X 3.46 TOTAL 49.90 DEBIT TEND 49.90 CHANGE DUE 0.00</p> <p>EFT DEBIT PAY FROM PRIMARY 49.90 TOTAL PURCHASE US DEBIT ***** 9689 I 0 REF # 029200639128 NETWORK ID: 0056 APPR CODE 461500 US DEBIT AID A0000000980840 AAC 7756F4E82B54EED TERMINAL # 50011004 10/18/20 11:30:42 # ITEMS SOLD 4 TOM 2400 4776 4305 8847 1471</p> <p>W+ Introducing Walmart+ Join today at walmart.com/plus Low Prices You Can Trust. Every Day. 10/18/20 11:30:46</p>	 <p>Give us feedback @ survey.walmart.com Thank you! ID #: 78655MCF22</p> <p>Walmart 949-498-6669 Mor: MICHAEL 951 AVENIDA PICO SAN CLEMENTE CA 92673</p> <p>ST# 02527 OP# 009045 TE# 45 TR# 06193 GY OATMEAL 007874243408 F 1.76 0 OT 200Z TOM 081236803115 6.74 X M ATHLETICS 019104967781 24.97 X DEXAS 15X20 008429710921 12.97 X SUBTOTAL 46.44 TAX 1 7.750 X 3.46 TOTAL 49.90 DEBIT TEND 49.90 CHANGE DUE 0.00</p> <p>EFT DEBIT PAY FROM PRIMARY 49.90 TOTAL PURCHASE US DEBIT ***** 9689 I 0 REF # 029200639128 NETWORK ID: 0056 APPR CODE 461500 US DEBIT AID A0000000980840 AAC 7756F4E82B54EED TERMINAL # 50011004 10/18/20 11:30:42 # ITEMS SOLD 4 TOM 2400 4776 4305 8847 1471</p> <p>W+ Introducing Walmart+ Join today at walmart.com/plus Low Prices You Can Trust. Every Day. 10/18/20 11:30:46</p>
 <p>TRADER JOE'S 2001 Greenville Ave Dallas TX 75206 Store #403 - (469) 334-0614 OPEN 8:00AM TO 9:00PM DAILY</p> <p>1-29 U-CARROTS SHREDDED 10 OZ 1.99 U-CLOVERED PERSIAN 1 LB 1.99 TOMATOES CRUSHED NO SALT 1.59 TOMATOES WHOLE NO SALT 1.59 ORGANIC OLD FASHIONED OATMEAL 2.69 MINI-PEARL TOMATOES 2.49 Pkg SHREDDED MOZZARELLA LITE T 3.99 100% V KEY ORGANIC BROWN 3.79 RIANG GARRARZO 0.89 SHREDDED CA STYLE 2.99 A-AVOCADOS WASS BAG ACT 3.99 A-APPLE BAG JAZZ 2 LB 2.99 A-PEPPER BELL LUCK XL RED 0.99 GROCERY NON TAXABLE 0.98 2 # 0.49 BANANAS ORGANIC 0.87 3EA # 0.29/EA CREAMY SALTED PEANUT BUTTER 2.49 WHL WHIT PITTA BREAD 1.59 GROCERY NON TAXABLE 1.58 2 # 0.59</p> <p>SUBTOTAL \$38.68 TOTAL \$38.68 CASH \$40.00 CHANGE \$1.32</p> <p>ITEMS 22 06-28-2014 12:34PM 0403 04 1346 4663 Higgins, Ryan</p> <p>THANK YOU FOR SHOPPING AT TRADER JOE'S www.traderjoes.com</p>	 <p>TRADER JOE'S 2001 Greenville Ave Dallas TX 75206 Store #403 - (469) 334-0614 OPEN 8:00AM TO 9:00PM DAILY</p> <p>1-29 U-CARROTS SHREDDED 10 OZ 1.99 U-CLOVERED PERSIAN 1 LB 1.99 TOMATOES CRUSHED NO SALT 1.59 TOMATOES WHOLE NO SALT 1.59 ORGANIC OLD FASHIONED OATMEAL 2.69 MINI-PEARL TOMATOES 2.49 Pkg SHREDDED MOZZARELLA LITE T 3.99 100% V KEY ORGANIC BROWN 3.79 RIANG GARRARZO 0.89 SHREDDED CA STYLE 2.99 A-AVOCADOS WASS BAG ACT 3.99 A-APPLE BAG JAZZ 2 LB 2.99 A-PEPPER BELL LUCK XL RED 0.99 GROCERY NON TAXABLE 0.98 2 # 0.49 BANANAS ORGANIC 0.87 3EA # 0.29/EA CREAMY SALTED PEANUT BUTTER 2.49 WHL WHIT PITTA BREAD 1.59 GROCERY NON TAXABLE 1.58 2 # 0.59</p> <p>SUBTOTAL \$38.68 TOTAL \$38.68 CASH \$40.00 CHANGE \$1.32</p> <p>ITEMS 22 06-28-2014 12:34PM 0403 04 1346 4663 Higgins, Ryan</p> <p>THANK YOU FOR SHOPPING AT TRADER JOE'S www.traderjoes.com</p>	 <p>TRADER JOE'S 2001 Greenville Ave Dallas TX 75206 Store #403 - (469) 334-0614 OPEN 8:00AM TO 9:00PM DAILY</p> <p>1-29 U-CARROTS SHREDDED 10 OZ 1.99 U-CLOVERED PERSIAN 1 LB 1.99 TOMATOES CRUSHED NO SALT 1.59 TOMATOES WHOLE NO SALT 1.59 ORGANIC OLD FASHIONED OATMEAL 2.69 MINI-PEARL TOMATOES 2.49 Pkg SHREDDED MOZZARELLA LITE T 3.99 100% V KEY ORGANIC BROWN 3.79 RIANG GARRARZO 0.89 SHREDDED CA STYLE 2.99 A-AVOCADOS WASS BAG ACT 3.99 A-APPLE BAG JAZZ 2 LB 2.99 A-PEPPER BELL LUCK XL RED 0.99 GROCERY NON TAXABLE 0.98 2 # 0.49 BANANAS ORGANIC 0.87 3EA # 0.29/EA CREAMY SALTED PEANUT BUTTER 2.49 WHL WHIT PITTA BREAD 1.59 GROCERY NON TAXABLE 1.58 2 # 0.59</p> <p>SUBTOTAL \$38.68 TOTAL \$38.68 CASH \$40.00 CHANGE \$1.32</p> <p>ITEMS 22 06-28-2014 12:34PM 0403 04 1346 4663 Higgins, Ryan</p> <p>THANK YOU FOR SHOPPING AT TRADER JOE'S www.traderjoes.com</p>

Fatura Görselleri Üzerinde OCR Araçlarının Karşılaştırılması		
Original	PaddleOCR	Tesseract
		

Tablo 5. Fatura görselleri üzerinde OCR araçlarının karşılaştırılması

Elde edilen sonuçlara göre, genel olarak, PaddleOCR’ın fatura görsellerinden çıkarılan metinleri genellikle daha doğru bir şekilde okuyarak daha başarılı performans sergilediği sonucuna varılmıştır. Tesseract ise bazen metin olmayan öğeleri de içerebilmektedir, bu da veri doğruluğunu olumsuz yönde etkileyebilmektedir.

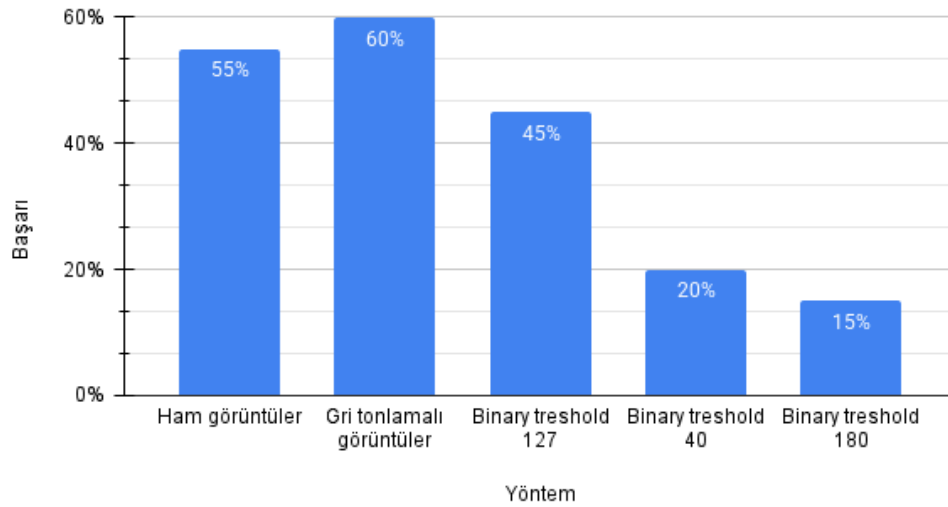
4.1.1. Veri Ön İşleme Aşamalarının Değerlendirilmesi

OCR teknolojisinin doğruluğunu artırmak ve en uygun ön işleme yöntemini belirlemek amacıyla, fatura görselleri farklı ön işleme tekniklerine tabi tutulmuş ve her birinde Tesseract optik karakter tanıma teknolojisi kullanılarak tarih bilgisi okunmaya çalışılmıştır. Bu çalışmada, Tesseract yazılımı kullanılmıştır. Deneyler, 20 farklı fatura görseli üzerinde gerçekleştirilmiş ve farklı ön işleme yöntemlerinin OCR performansına etkisi karşılaştırılmıştır.

İlk olarak, ham (RAW) görüntüler üzerinde OCR uygulanmış ve 20 fatura içerisinde 11'inde tarih bilgisi başarılı şekilde tespit etmiştir. Başarı oranı %55 olarak hesaplanmıştır. Ardından, gri tonlamalı (gray-scale) görüntüler üzerinde yapılan testlerde, başarı oranı %60'a yükselmiş ve 12 fatura üzerindeki tarih bilgisi doğru şekilde tespit edilmiştir. Bu sonuç, gri tonlamanın OCR performansını artırdığına işaret etmektedir.

Bunun yanı sıra, gri tonlamanın ardından uygulanan farklı ikili eşikleme (binary threshold) değerleri de test edilmiştir. 127 eşik değeri kullanılarak yapılan işlem sonucunda tespit oranı %45'e düşmüş ve yalnızca 9 faturada tarih bilgisi tespit edilmiştir. Eşik değeri 40'a düşürüldüğünde başarı oranı %20'ye gerileyerek yalnızca 4 fatura üzerinde veri elde edilmiştir. Benzer şekilde, eşik değeri 180'e yükseltildiğinde en düşük başarı oranı (%15) gözlemlenmiş ve yalnızca 3 fatura üzerindeki tarih bilgisi tespit edilmiştir.

Başarı-Yöntem Grafiği



Görsel 4. Başarı-Yöntem Grafiği

Elde edilen sonuçlar doğrultusunda, Tesseract OCR aracının en iyi performansı gri tonlamalı görüntüler üzerinde gösterdiği belirlenmiştir. Bu yöntemde tespit oranı %60 olarak ölçülmüş ve diğer ön işleme yöntemlerine kıyasla en yüksek tespit oranı sağlanmıştır. Sonuç olarak, faturalardan tarih verisi çıkarmada Tesseract OCR aracının performansını optimize etmek için görüntülerin gri tonlamaya dönüştürülmesi

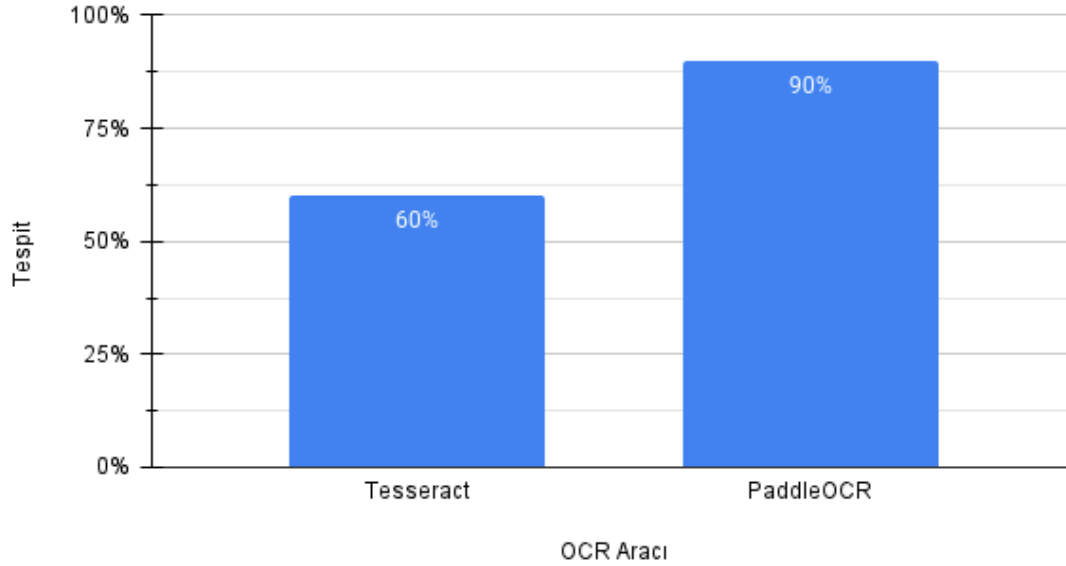
önerilmektedir.

4.1.2. Tesseract ve PaddleOCR ile Tarih Bilgisi Çıkarımı: Doğruluk, Tespit ve Hız Karşılaştırması

Tesseract OCR aracında en doğru sonucu veren gri tonlamalı (gray-scale) görseller üzerinden tarih çıkarma işlemi, aynı koşullar altında PaddleOCR ile de test edilmiştir. Önceki paragrafta belirtildiği üzere, Tesseract OCR tüm faturalardaki tarihlerinin %60'ını doğru şekilde okumayı başarmıştır.

Öncelikle, Tesseract kullanılarak faturalar üzerinde optik karakter tanıma işlemi gerçekleştirilmiş ve 10 saniyede 20 faturadan 12 tarih verisi tespit edilerek %60 tespit oranı elde edilmiştir. Aynı koşullar altında PaddleOCR kullanılarak faturalar üzerinden tarih bilgisi çıkarma yöntemi test edilmiştir. Elde edilen sonuçlara göre, PaddleOCR 56 saniyede 20 faturadan 18 tarih verisini tespit ederek %90 tespit oranına ulaşmıştır.

Tespit-OCR Aracı Grafiği



Görsel 5. Tespit-OCR Aracı Grafiği

Ancak, PaddleOCR'ın CPU üzerinde çalıştırılan sürümü, Tesseract'ın 10 saniyede tamamladığı işlemi 56 saniyede tamamlayarak, Tesseract'a kıyasla önemli ölçüde daha yavaş çalışmıştır. Bu durum, PaddleOCR'ın sunduğu yüksek tespit oranının işlem süresi

açısından bir maliyet getirdiğini göstermektedir. Ayrıca, PaddleOCR genel olarak daha yüksek tespit oranı sunsa da bazen tarih verisini yanlış okuma durumu gözlemlenmiştir. Çalışma sonuçları aşağıdaki tabloda verilmiştir.

Görsel	Doğru "date"	PaddleOCR	Tesseract
0	08/20/10	08/20/10	None
1	06-28-2014	06-28-2014	06-28-2014
2	10/18/20	10/18/20	10/18/20
3	04/27/19	04/27/19	None
4	12/08/15	12/08/15	12/08/15
5	02/10/2021	9/15/2020	9/15/2020
6	26/01/2025	26/01/2015	26/01/2016
7	11/13/17	11/13/1712	11/13/17
8	04/20/2016	None	None
9	09/08/14	09/80/60	09/08/14
10	23.02.21	23.02.21	None
11	None	None	None
12	10/20/07	10/20/07	None
13	08/11/17	08/11/17	None
14	05/04/17	05/04/17	05/04/17
15	07/22/16	07/22/16	07/22/16
16	05.11.2024	05.11.2024	05.11.2024
17	01/15/17	01/15/17	01/18/17
18	10/31/21	10/31/2110	10/31/21
19	10/16/21	10/16/21	None

Tablo 6. PaddleOCR ve Tesseract “date” verisi üzerine OCR sonuçları

Tabloda da belirtildiği gibi bu değerlendirmede, PaddleOCR toplam 19 fatura görselinin 14’ünde doğru tarih tahmini yaparak %73,7 doğruluk oranına ulaşmıştır. Tesseract ise yalnızca 9 görselde doğru sonuç vererek %47,4 doğruluk oranında kalmıştır. Ayrıca PaddleOCR’ın bazı durumlarda yanlış tarih formatları ve tahminleri ürettiği gözlemlense de, Tesseract’a kıyasla daha kararlı bir performans sergilediği

görülmektedir. Süre açısından değerlendirildiğinde, PaddleOCR'nin bu sonuçları 56 saniyede, Tesseract'ın ise 10 saniyede verdiği belirtilmektedir. Bu sonuçlar, PaddleOCR'nin daha yüksek doğruluğa sahip olmasına karşın daha fazla işlem süresi gerektirdiğini ortaya koymaktadır.

OCR Teknolojisi	Tahmin Oranı	Zaman
PaddleOCR	%73,7	56 saniye
Tesseract	%47,4	10 saniye

Tablo 7. PaddleOCR ve Tesseract Tahmin Oranı - Zaman karşılaştırması

Dolayısıyla, PaddleOCR'ın performans avantajlarından tam olarak yararlanmak için GPU hızlandırması ile çalıştırılması önerilmektedir. Bununla birlikte, kritik öneme sahip uygulamalarda yanlış okuma ihtimaline karşı bir doğrulama mekanizması veya ek ön işleme adımları kullanılması faydalı olabilir.

4.1.3. Faturalardan Toplam Fiyat Bilgilerinin Çıkarımı

Önceki bulgulara ek olarak, toplam fiyat verisinin tespitine yönelik yapılan analizde, PaddleOCR'nin 20 farklı görsel üzerinde test edildiği ve bu görsellerden 18'inde toplam fiyatı bir sayı olarak tespit ettiği belirlenmiştir. Ancak, bu 18 tahminden yalnızca 14'ü doğru olup, PaddleOCR'nin toplamda %70 doğruluk oranına ulaştığı görülmüştür.

Görsel	Doğru "total"	PaddleOCR
0	5.11	5.11
1	38.68	40.0
2	49.90	49.9
3	144.02	144.02
4	7.43	7.43
5	28.28	28.28
6	175.0	175.0
7	23.19	23.19
8	89.13	85.61
9	121.92	1.37

10	338.16	338.16
11	45.44	None
12	18.75	18.75
13	50.00	None
14	26.60	7.0
15	12.58	12.58
16	8.93	8.93
17	38.68	38.68
18	86.35	86.35
19	35.05	3.61

Tablo 8. PaddleOCR ile fatura görsellerinden “total” bilgisinin okunması sonuçları

Birçok durumda "TOTAL" yerine başka kelimelerin ("CASH," "SUBTOTAL," "EGGS," "BAL," "TAX," "RED GRAPE") yanındaki rakamlar yanlışlıkla algılanmış, bu da doğruluğu olumsuz etkilemiştir. Ayrıca, bir görselde "TOTAL" kelimesindeki ikinci "T" soluk olduğu için PaddleOCR bu kısım tespit edilememiştir.

4.2. Faturalardan Optik Karakter Tanıma Teknolojisi ile Okunan Faturalardan Fatura Kalemi Çıkarımı

Kullanıcı, POST /invoices/process_qr endpoint'ine QR koddan elde ettiği URL'yi içeren qr_data verisini gönderir. Bu istek, JWT tabanlı kimlik doğrulama kontrolünden geçtikten sonra gönderilen JSON içerisinden qr_data alanı alınır. Ardından, HTTP Authorization başlığında gönderilen JWT token çözülerek kullanıcı kimliği doğrulanır. qr_data içerisinde gelen bağlantıdan fatura ID'si ayrıştırılır ve e-kassa sisteminden ilgili fatura görseli indirilir. Görsel images/ dizinine kaydedilir.

Kaydedilen görsel üzerinde OCR işlemi gerçekleştirilerek belge üzerindeki yazılar tespit edilir. Elde edilen metinler, satır satır bir liste haline getirilir.

OCR ile elde edilen satırlar içerisinden:

- Fatura tarihi,
- Satıcı (vendor) ismi ve adresi,
- Toplam tutar,

- Para birimi (sabit olarak AZN),
- Kalemler (ürün/hizmet açıklaması, miktar, birim fiyat)

gibi temel bilgiler ayrıştırılır.

Kalem çıkarımı sırasında sistem, "Quantity", "Price" ve "Total" gibi anahtar kelimelerin bulunduğu satırları tespit ederek bu bölümden itibaren ürün kalemlerini ayrıştırmaya çalışır. Kalemler çok satırlı açıklamalara sahip olabileceği için açıklamalar birleştirilerek işlenir. Her kalem için açıklama, miktar ve birim fiyat bilgileri elde edilip yapılandırılmış bir formatta kaydedilir.

Elde edilen tüm veriler yapılandırılmış bir JSON formatında toplanır ve kullanıcının hesabıyla ilişkilendirilerek veritabanına kaydedilir. Son olarak, geçici olarak indirilen görsel silinir. OCR sonrası çıkan düzensiz metinleri doğru alanlara ayrıştırmak için özel kurallar ve düzenli ifadeler kullanılmıştır.

4.3. Fatura Kalemi Kategorizasyonu

Bu çalışmada, faturalardan optik karakter tanıma (OCR) ile çıkarılan ürün kalemlerinin anlamlandırılması ve kategorik olarak sınıflandırılması hedeflenmiştir. Metinler, TF-IDF ve Bag of Words (BoW) gibi vektörleştirme yöntemleriyle sayısal formatlara dönüştürülmüş ve sınıflandırma modellerinin girdisi olarak kullanılmıştır.

Model eğitimi aşamasında, Amazon Products Sales Dataset 2023'ten türetilen örneklenmiş veri setleri ile Logistic Regression, Destek Vektör Makineleri (SVM), Naive Bayes, Decision Tree ve Random Forest gibi çeşitli makine öğrenmesi algoritmaları denenmiştir. Yapılan karşılaştırmalarda özellikle BoW ile vektörleştirilen ve Logistic Regression modeliyle eğitilen sistemin eğitim zamanı, doğruluk ve f1-score gibi metriklerde en başarılı sonuçları verdiği gözlemlenmiştir.

Geliştirilen sistem, OCR ile elde edilen ürün kalemlerini önceden belirlenen kategorilere başarıyla atayabilmekte ve bu sayede kullanıcıların harcama analizlerinin daha verimli bir şekilde yapılabilmesine olanak tanımaktadır. Ayrıca, gerçek hayattan toplanan Azerbaycan dilindeki yerel fatura verileriyle sistemin genellenebilirliği test edilmiş ve sonuçlar başarılı bulunmuştur.

Bu kategorizasyon modülü, ilerleyen aşamalarda harcama alışkanlıklarının belirlenmesi,

anomalilerin tespiti ve öneri sistemlerinin geliştirilmesi gibi ileri düzey analizler için de temel teşkil edecektir.

4.3.1. Kullanılan Veri Seti: Amazon Products Sales Dataset 2023

Projemizde kullanılan seti, Amazon Products 2023 veri setidir [11]. Amazon Products Sales Dataset, Amazon web sitesinden web scraping yöntemi kullanılarak elde edilmiş, gerçek dünya verilerini içeren kapsamlı bir veri setidir. Veri seti CSV formatında sunulmuş olup, makine öğrenimi, doğal dil işleme ve veri analizi gibi pek çok alanda kullanılabilecek zengin içerik barındırmaktadır. Toplamda 1.103.170 satır içeren bu veri seti, çeşitli ürün bilgilerini ve bu ürünlere ait kategorileri içermektedir. Özellikle ürün isimleri ('name') ve ana kategoriler ('main_category') gibi alanlar, metin sınıflandırma ve kategorik analiz çalışmaları için güçlü bir temel sunmaktadır. Gerçek kullanıcı verilerinden elde edildiği için, veri seti hem ürün çeşitliliği hem de dağılım açısından oldukça geniş bir yelpazeye sahiptir. Bu durum, model geliştirme ve performans değerlendirme aşamalarında gerçekçi ve anlamlı sonuçlar elde edilmesine olanak sağlamaktadır.

Sütun Adı	Sütun Açıklaması
name	Ürünün ismi
main_category	Ürünün dahil olduğu kategori
sub_category	Ürünün dahil olduğu alt kategori
image	Ürünün resmi
link	Ürünün Amazon web sitesindeki referans linki
ratings	Kullanıcılar tarafından yapılan değerlendirmeler
no of ratings	Bu ürüne yapılan değerlendirme sayısı
discount_price	Ürünün indirimli fiyatı
actual_price	Ürünün asıl fiyatı

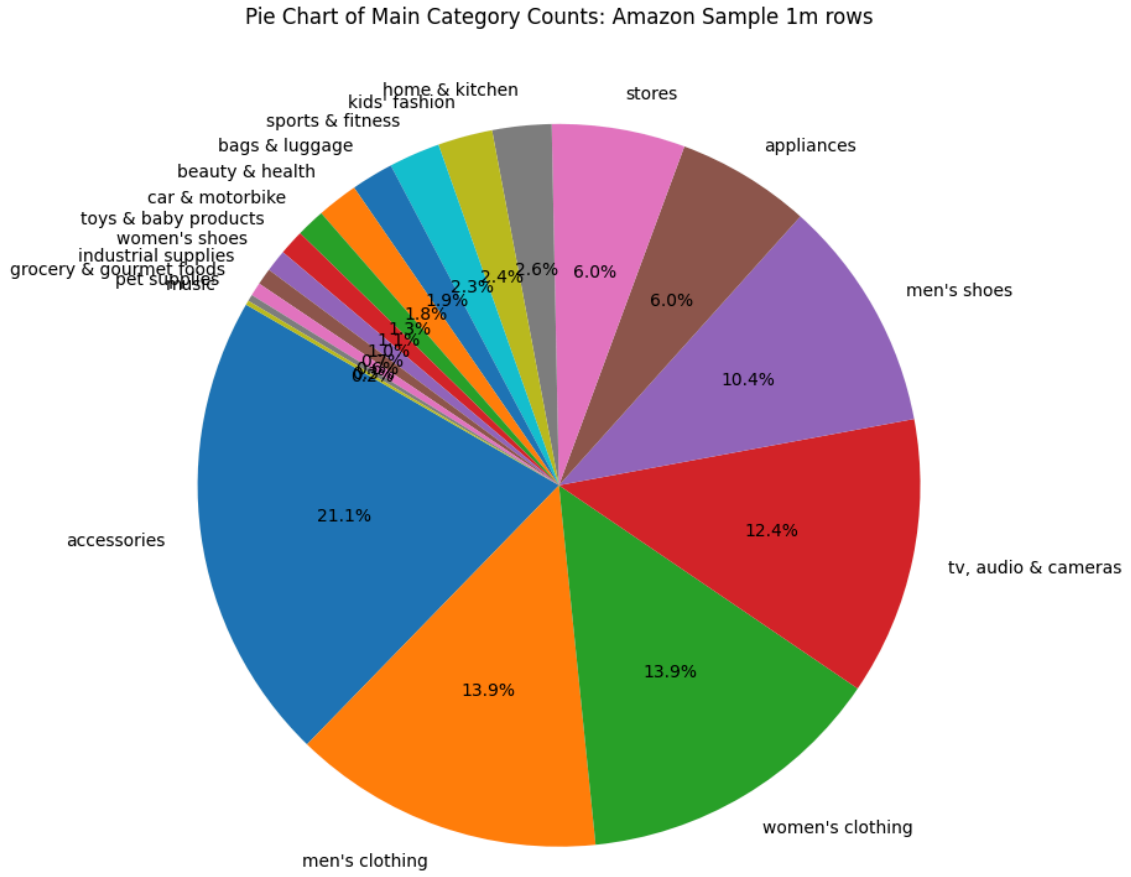
Tablo 9. Amazon Products Sales Dataset 2023 veri setinin sahip olduğu sütunlar

Projemizde kategorizasyon işlemi için, veri setinde yer alan ürün isimlerini içeren

‘name’ sütunu ile her bir ürünün ait olduğu ana kategori bilgisini içeren ‘main_category’ sütunu kullanılmıştır. Bu iki sütun, ürünlerin metin tabanlı olarak sınıflandırılması ve doğru kategoriye atanması amacıyla temel veri kaynakları olarak değerlendirilmiştir. Kategorilerin veri setindeki yüzdelik dağılımları ise aşağıda detaylı bir şekilde sunulmuştur.

Kategori adı	Kategorinin örneklem uzayındaki oranı
accessories	%21.056
men's clothing	%13.897
women's clothing	%13.871
tv, audio & cameras	%12.448
men's shoes	%10.417
appliances	%6.0
stores	%5.965
home & kitchen	%2.645
kids' fashion	%2.445
sports & fitness	%2.293
bags & luggage	%1.888
beauty & health	%1.835
car & motorbike	%1.284
toys & baby products	%1.127
women's shoes	%0.992
industrial supplies	%0.744
grocery & gourmet foods	%0.6
pet supplies	%0.296
music	%0.196

Tablo 10. Kategorilerin yüzdelik dağılımları



Görsel 6. Kategorilerin yüzdelik dağılımlarının pasta grafiği olarak gösterimi

Veri setinin incelenmesi sonucunda, ürünlerin büyük bir kısmının aksesuarlar, erkek giyimi, kadın giyimi ve elektronik (TV, ses sistemleri, kameralar) kategorilerinde yoğunlaştığı görülmektedir. Bu dört kategori toplamda %61'den fazla bir orana sahiptir. Öte yandan, müzik, evcil hayvan ürünleri ve gurme yiyecekler gibi kategoriler ise oldukça düşük bir temsil oranına sahiptir. Bu dağılım, model eğitimi sırasında veri dengesizliğine dikkat edilmesi gerektiğini ve azınlık sınıflar için uygun önlemlerin alınmasının önemli olabileceğini göstermektedir.

4.3.2. Veri Setinden Örneklem Alınması

Amazon Product Sales veri seti, makine öğrenimi modeli eğitimi için kaynak optimizasyonu göz önünde bulundurularak, veri setinin boyutuyla ilişkili hesaplama yükünü en aza indirirken anlamsal bütünlüğü korumak için optimum veri hacmini belirlemeye odaklanmıştır. Bu kapsamlı veri seti, proje kapsamında yürütülen sınıflandırma çalışmalarında kullanılmak üzere örnekleme (sampling) yöntemi ile boyut olarak küçültülmüş, ancak veri dağılımının istatistiksel yapısı korunarak normalizasyon oranı sabit tutulmuştur.

Veri setleri, satır sayısına göre 11 bin, 22 bin, 55 bin, 110 bin, 220 bin, 550 bin olacak şekilde 6 farklı şekilde örneklendirilmiştir ve modeller arasında hızlı karşılaştırma için her örneklendirilmiş veri seti ile farklı bir Logistic Regression modeli eğitilmiştir. Modeller eğitilirken veri setleri %80 train seti ve %20 test seti olacak şekilde ayrılmıştır. Veriler bu çalışmada varsayılan olarak TF-IDF vektörizasyon yolu ile vektörize edilmiştir. Aşağıda, örneklendirilmiş veri setlerinden eğitilmiş modellerden elde edilen değerlendirme sonuçları yer almaktadır.

toplam veri sayısı	accuracy	macro-precision	weighted-precision	macro-recall	weighted-recall	macro-f1-score	weighted-f1-score	total support (test)
11032	0.82	0.75	0.81	0.49	0.82	0.53	0.78	2207
22064	0.85	0.80	0.84	0.61	0.85	0.67	0.84	4413
55159	0.87	0.84	0.86	0.67	0.87	0.72	0.86	11032
110317	0.88	0.84	0.87	0.73	0.88	0.77	0.88	22064
220634	0.89	0.85	0.88	0.76	0.89	0.79	0.88	44127
551585	0.89	0.85	0.88	0.78	0.89	0.81	0.88	110317
1103170	0.89	0.84	0.88	0.79	0.89	0.81	0.88	220634

Tablo 11. Farklı sayıda örnek içeren verilerle eğitilen Logistic Regression modellerinin performansları

Sonuçlar incelendiğinde kaynak/performans açısından 220634 satırlık verinin bir model eğitmek için en uygun veri seti olduğu görülmektedir. Fakat örnekleme işleminde yalnızca doğruluk değil, zaman kazanımı da hedeflenmektedir.

Names sütununun kısaltılması

Veri kümesindeki 'names' sütunu, modelin eğitim sürecini hızlandırmak amacıyla kısaltılmıştır. Bu kısaltma işlemi sırasında Google T5 Tokenizer kullanılmış ve kısaltılmış veri ile orijinal veri arasındaki sonuçlar karşılaştırmalı olarak analiz edilmiştir.

toplam veri sayısı	kısaltılmış	accuracy	macro-precision	weighted-precision	macro-recall	weighted-recall	macro-f1-score	weighted-f1-score	total support (test)
11032	Hayır	0.82	0.75	0.81	0.49	0.82	0.53	0.78	2207
11032	Evet	0.71	0.62	0.72	0.35	0.71	0.38	0.67	2207
22064	Hayır	0.85	0.80	0.84	0.61	0.85	0.67	0.84	4413
22064	Evet	0.74	0.77	0.75	0.42	0.74	0.48	0.71	4413
55159	Hayır	0.87	0.84	0.86	0.67	0.87	0.72	0.86	11032
55159	Evet	0.77	0.77	0.76	0.49	0.77	0.54	0.75	11032

Tablo 12. Names sütununun kısaltılmış hali ve orijinal hali ile eğitilen farklı Logistic Regression modellerinin performansları

İsimleri kısaltma işlemi sonucunda yeni modellerin kötü sonuçlar vermesi sebebiyle bu fikrin uygulanması uygun görülmemiştir.

4.3.3. Vektörizasyon

Projede iki farklı metin vektörizasyon yöntemi kullanılmıştır. Şu ana kadar yürütülen veri setini küçültme ve özetleme çalışmalarında varsayılan olarak TF-IDF algoritması tercih edilmiştir. Bu aşamada ise, Bag of Words ve TF-IDF algoritmalarının performansları karşılaştırmalı olarak değerlendirilmiş, her iki yöntemin sınıflandırma başarımına etkisi analiz edilmiştir.

Bag of Words

Bag of Words (BoW) algoritması, metin madenciliği ve doğal dil işleme (NLP) alanlarında sıklıkla kullanılan, bir metni sabit uzunlukta bir vektörle temsil etmeye yönelik temel ve etkili bir yaklaşımdır. Bu yöntem, metindeki sözcük sırasını veya gramer yapısını dikkate almaksızın, yalnızca kelimelerin frekanslarına odaklanır. Her

metin, önceden tanımlanmış bir kelime sözlüğü doğrultusunda, bu kelimelerin metin içinde kaç kez geçtiğini gösteren bir sayısal vektöre dönüştürülür. Bu sayede, metinler arasında karşılaştırma, sınıflandırma ve benzeri istatistiksel işlemler gerçekleştirilebilir. BoW modeli, basitliği ve uygulama kolaylığı sayesinde özellikle belge sınıflandırma, duygu analizi ve bilgi erişimi gibi çeşitli NLP görevlerinde yaygın olarak kullanılmaktadır.

Bag of Words algoritmasını projemize dahil etmek için sklearn modülünün CountVectorizer sınıfı kullanılmıştır.

TF-IDF (Term Frequency–Inverse Document Frequency)

TF-IDF (Term Frequency–Inverse Document Frequency) algoritması, metin madenciliği ve bilgi erişimi alanlarında sıkça kullanılan, bir kelimenin belirli bir belgede ne kadar önemli olduğunu niceliksel olarak ifade eden istatistiksel bir ağırlıklandırma yöntemidir. Bu yöntemde, "term frequency" (TF) bir kelimenin ilgili belge içerisinde kaç kez geçtiğini ölçerken, "inverse document frequency" (IDF) ise bu kelimenin tüm belge koleksiyonu içerisindeki ayırt ediciliğini değerlendirir. Nadir kullanılan fakat belirli belgelerde sıkça geçen kelimelere daha yüksek ağırlık verilirken, koleksiyondaki birçok belgede geçen yaygın kelimeler düşük ağırlıkla temsil edilir. Bu sayede TF-IDF, metinlerin daha anlamlı ve ayrıştırıcı temsillerinin oluşturulmasına olanak tanır ve belge sınıflandırma, bilgi getirme, kümeleme ve öneri sistemleri gibi pek çok doğal dil işleme görevinde etkili bir şekilde kullanılmaktadır.

TF-IDF algoritmasını projemize dahil etmek için sklearn modülünün TfidfVectorizer sınıfı kullanılmıştır.

Karşılaştırma ve Sonuçlar

Bu çalışmada, veri setinde yer alan 'name' sütunu vektörleştirilerek metinsel verilerin sayısal temsili elde edilmiştir. Bu amaçla Bag of Words ve TF-IDF algoritmaları kullanılmıştır. Söz konusu vektörleştirme yöntemlerinin sınıflandırma performansları karşılaştırılmak üzere, 11 bin, 22 bin ve 55 bin örnekten oluşan üç farklı veri seti üzerinde Logistic Regression modelleri eğitilmiştir. Böylece, her iki yöntemin farklı veri seti boyutlarındaki etkisi sistematik olarak analiz edilmiştir.

toplam	vektörü	accu	macr	weigh	macr	weigh	macro	weigh	total
--------	---------	------	------	-------	------	-------	-------	-------	-------

veri sayısı	zasyon	racy	o-pre cisio n	ted-p recisi on	o-rec all	ted-re call	-f1-sc ore	ted-f1 -score	suppo rt (test)
11032	TF-IDF	0.82	0.75	0.81	0.49	0.82	0.53	0.78	2207
11032	BoW	0.83	0.80	0.82	0.61	0.83	0.66	0.82	2207
22064	TF-IDF	0.85	0.80	0.84	0.61	0.85	0.67	0.84	4413
22064	BoW	0.86	0.82	0.85	0.69	0.86	0.73	0.85	4413
55159	TF-IDF	0.87	0.84	0.86	0.67	0.87	0.72	0.86	11032
55159	BoW	0.87	0.81	0.86	0.73	0.87	0.76	0.86	11032

Tablo 13. TF-IDF ve BoW vektörizasyon algoritmaları kullanılarak 11 bin, 22 bin ve 55 bin sample veri seti ile eğitilen Logistic Regression modellerinin performansları

TF-IDF ve Bag of Words algoritmaları ile elde edilen sonuçlar karşılaştırıldığında, aynı veri seti ve aynı sınıflandırma modeli kullanıldığında, Bag of Words algoritmasının TF-IDF algoritmasına kıyasla daha yüksek performans sergilediği gözlemlenmiştir. Bu durum, özellikle incelenen veri setindeki kelime dağılımı ve örnekleme yapısının, BoW algoritmasının güçlü yönleriyle daha uyumlu olduğunu göstermektedir.

Fakat 55 bin sample veri setinde yapılan incelemede BoW algoritmasının ‘macro-precision’ metriğinde düşük performans sergilemesi, veri setinin boyutu arttıkça BoW algoritmasının performansının düşebileceğine dair soru işaretleri getirmiştir. Bu düşüncenin doğruluğunu test etmek amacıyla projenin ‘Amazon Veri Seti Küçültülmesi’ bölümünde en optimal sample veri seti olarak belirlenen 220k sample veri seti kullanılarak çalışma tekrarlanmıştır.

toplam veri sayısı	vektöri zasyon	accu racy	macr o-pre cisio n	weigh ted-p recisi on	macr o-rec all	weigh ted-re call	macro -f1-sc ore	weigh ted-f1 -score	total suppo rt (test)
220634	TF-IDF	0.89	0.86	0.88	0.76	0.89	0.80	0.88	44127
220634	BoW	0.89	0.86	0.89	0.80	0.89	0.82	0.89	44127

Tablo 14. TF-IDF ve BoW vektörizasyon algoritmaları kullanılarak 220 bin sample veri seti ile eğitilen Logistic Regression modellerinin performansları

Sonuç olarak 220 bin sample veri seti kullanılarak tekrarlanan karşılaştırmada genel olarak bu veri setinde BoW algoritmasının TF-IDF algoritmasından daha iyi sonuç gösterdiği görülmüştür.

4.3.4. Model Seçimi ve Eğitimi

Makine öğrenmesi projelerinde model başarısını doğrudan etkileyen en kritik adımlardan biri uygun algoritmanın seçimi ve bu algoritmanın veriye en iyi şekilde uyum sağlayacak biçimde eğitilmesidir. Hangi makine öğrenmesi algoritmasının daha uygun olduğu bu aşamada tartışılacak ve değerlendirilecektir. Uygun makine öğrenmesi algoritması belirlendikten sonra belirtilen veri seti kullanılarak eğitilecektir. Her eşyanın hangi kategoride olduğunu belirlemek için oluşturulan makine öğrenmesi algoritması kullanılacaktır.

Bu çalışmada, ürün ismine dayalı kategorik sınıflandırma problemi ele alınmış olup, metin verisinin yapısına uygun olarak farklı sınıflandırma algoritmaları değerlendirilmiştir.

Özellikle 11 bin ve 55 bin örnek içeren iki ayrı sample veri seti ile klasik makine öğrenmesi algoritmaları, derin öğrenme modelleri ve Transformer tabanlı modeller en optimal vektörleştirme yöntemi olarak belirlediğimiz Bag-of-Words (BoW) algoritması kullanılarak vektörize edilmiş veriler ile eğitilmiştir.

Çalışmanın nihai karşılaştırma aşamasında ise daha kapsamlı ve temsili sonuçlar elde edebilmek amacıyla 220 bin örnek içeren optimal veri seti kullanılmıştır.

Klasik Makine Öğrenmesi Algoritmalarının Karşılaştırılması

Klasik makine öğrenmesi algoritmaları karşılaştırma çalışmasında Logistic Regression SVM, Naive Bayes sınıflandırıcıları, Decision Tree ve Random Forest modelleri eğitilmiştir.

algoritma	accu racy	macro -precis ion	weighte d-precisi on	macro -recall	weighte d-recall	macro -f1-sco re	weight ed-f1-s core
Logistic	0.83	0.80	0.82	0.61	0.83	0.66	0.82

Regression							
SVM Linear	0.83	0.80	0.82	0.62	0.83	0.68	0.82
SVM Poly	0.65	0.71	0.76	0.29	0.65	0.32	0.60
SVM RBF	0.80	0.71	0.80	0.45	0.80	0.50	0.76
Multinomial NB	0.80	0.72	0.78	0.48	0.80	0.52	0.76
Gaussian NB	0.69	0.62	0.74	0.53	0.69	0.54	0.71
Bernoulli NB	0.70	0.24	0.58	0.27	0.70	0.25	0.62
Complement NB	0.82	0.75	0.80	0.62	0.82	0.65	0.79
Decision Tree	0.75	0.60	0.73	0.52	0.75	0.55	0.74
Random Forest 100	0.80	0.77	0.79	0.52	0.80	0.58	0.78

Tablo 15. 11 bin sample veri seti ile eğitilen farklı klasik makine öğrenmesi modellerinin performansları

Yapılan çalışma sonucunda, 11 bin örnekten oluşan sample veri setiyle eğitilen klasik makine öğrenmesi modelleri arasında tüm değerlendirme metrikleri açısından en yüksek performansı lineer çekirdek fonksiyonu kullanan Destek Vektör Makineleri (SVM) modeli göstermiştir. Çalışmada lineer çekirdek fonksiyonu kullanan SVM modeli ve kendisine çok yakın bir performans gösteren Logistic Regression modeli örnekteki gibi küçük çaplı veri setleri için en uygun modeller olarak seçilmişlerdir.

Buna ek olarak, Gaussian Naive Bayes algoritmasının seyrek (sparse) veri yapısını desteklememesi nedeniyle, veri liste formatına dönüştürülmek zorunda kalmıştır. Bu dönüşüm, hem zaman hem de bellek kullanımı açısından kayıplara yol açmıştır. Söz konusu algoritma, bu nedenlerin yanı sıra düşük performans sergilemesi nedeniyle daha büyük sample veri setleri ile gerçekleştirilen çalışmalarda tercih edilmemiştir.

algoritma	accuracy	macro-precision	weighted-precision	macro-recall	weighted-recall	macro-f1-score	weighted-f1-score
-----------	----------	-----------------	--------------------	--------------	-----------------	----------------	-------------------

Logistic Regression	0.87	0.81	0.86	0.73	0.87	0.76	0.86
SVM Linear	0.86	0.79	0.86	0.73	0.86	0.75	0.86
SVM Poly	0.75	0.85	0.80	0.41	0.75	0.47	0.72
SVM RBF	0.87	0.85	0.86	0.64	0.87	0.69	0.86
Multinomial NB	0.84	0.83	0.83	0.57	0.84	0.63	0.82
Bernoulli NB	0.76	0.47	0.71	0.31	0.76	0.28	0.67
Complement NB	0.84	0.81	0.67	0.82	0.84	0.71	0.81
Decision Tree	0.81	0.69	0.81	0.62	0.81	0.65	0.81
Random Forest 100	0.86	0.84	0.85	0.63	0.86	0.69	0.84

Tablo 16. 55 bin sample veri seti ile eğitilen farklı klasik makine öğrenmesi modellerin performansları

55 bin örnekten oluşan sample veri seti ile yapılan çalışma sonucunda, bu veri seti ile eğitilen klasik makine öğrenmesi modelleri arasında genel olarak değerlendirme metriklerinde Logistic Regression ve radyal baz fonksiyonu (RBF) çekirdek fonksiyonu kullanan Destek Vektör Makineleri (SVM) modelleri öne çıkmaktadır. Fakat, SVM RBF modelinin macro-recall metriğinde düşük olması bu modelin azınlık sınıfları tanımakta zorlandığını gösterir. ve bu durum, aynı modelde macro-f1-score metriğinin düşük olmasına yol açmıştır. Azınlık sınıflarda daha iyi performans gösteren Logistic Regression modelinin en optimal model olduğu söylenebilir.

İki tablo birlikte incelendiğinde, Logistic Regression modeli, 11 bin sample veri seti ile yapılan çalışmada lineer çekirdekli SVM modelinin gerisinde kalırken; 55 bin sample veri seti ile yapılan çalışmada lineer çekirdekli SVM modelinin ötesinde bir performans sergilemiştir. Bu durum, Logistic Regression modelinin genel çıkarım yapabilmesi için daha çok veri setine ihtiyacı olduğunu ve lineer çekirdekli SVM modelinin daha az veride daha iyi genel çıkarım yaptığını gösteriyor olabilir.

Verilerden elde edilen sonuçlara göre 220 bin veri seti kullanılarak yapılan son performans karşılaştırma için en uygun modeller:

1. Logistic Regression
2. SVM Linear
3. SVM RBF
4. Random Forest

olacak şekilde belirlenmiştir.

algoritma	accuracy	macro-precision	weighted-precision	macro-recall	weighted-recall	macro-f1-score	weighted-f1-score
Logistic Regression	0.89	0.86	0.89	0.80	0.89	0.82	0.89
SVM Linear	0.90	0.86	0.89	0.81	0.90	0.83	0.89
SVM RBF	0.90	0.88	0.90	0.78	0.90	0.81	0.89
Random Forest 100	0.89	0.87	0.88	0.75	0.89	0.80	0.88

Tablo 17. 220 bin sample veri seti ile eğitilen farklı klasik makine öğrenmesi modellerin performansları

220 bin sample veri seti ile eğitilen modellerin performansları karşılaştırıldığında Random Forest modeli dışında diğer modellerin birbirlerine çok yakın performans gösterdiği görülmektedir. Random Forest modelinin performansı, diğer modellere göre geride kalmıştır.

Modeller, eğitim zamanı açısından incelendiğinde ise veri setinin yüklenmesi, ‘names’ sütununun vektörizasyonu, “main_category” sütununa label encoding uygulanması, modelin eğitilmesi ve performansının değerlendirilmesi kapsamında geçen toplam zamanda modellerin hazır olma süreleri birbirinden farklılık göstermektedir.

Veri Seti	Model	Eğitim Zamanı
11 bin sample veri seti	Logistic Regression	2.3 saniye
11 bin sample veri seti	SVM Linear	5.1 saniye
11 bin sample veri seti	SVM RBF	10.2 saniye
11 bin sample veri seti	Random Forest	18.6 saniye
55 bin sample veri seti	Logistic Regression	11.5 saniye

55 bin sample veri seti	SVM Linear	69.2 saniye
55 bin sample veri seti	SVM RBF	137.8 saniye
55 bin sample veri seti	Random Forest	246.5 saniye
220 bin sample veri seti	Logistic Regression	28.2 saniye
220 bin sample veri seti	SVM Linear	1587.9 saniye
220 bin sample veri seti	SVM RBF	3031,7 saniye
220 bin sample veri seti	Random Forest	2905 saniye

Tablo 18. Klasik makine öğrenmesi algoritmalarının zaman karşılaştırması

Zaman karşılaştırması incelendiğinde, her iki veri setinde de algoritmalar arasında en hızlı eğitilen model çok büyük bir fark ile Logistic Regression modeli olmuştur. Diğer modellerin eğitim süreleri exponansiyel olarak artarken, Logistic Regression modelinin eğitim süresi lineer olarak artmaktadır. Bu durum, Logistic Regression modelinin 1.1m satırlık büyük veri seti ile eğitilmek için en uygun aday olduğunu gösterir.

Sonuç

Sonuç olarak bu bölümde yapılan tüm çalışmalar incelendiğinde, gerek değerlendirme parametrelerindeki yüksek performansı ile gerekse kaynak optimizasyonu ile ürün kategorizasyonu probleminde **Logistic Regression**, diğer klasik makine öğrenmesi algoritmalarından daha çok öne çıkmaktadır. Şayet kaynak optimizasyonu önemli değilse radyan baz fonksiyonu veya lineer çekirdekli Destek Vektör Makineleri (SVM) modeli de kullanılabilir.

topla m veri	algoritma	accu racy	macr o-pre cision	weighte d-prec ision	macr o-rec all	weight ed-rec all	macr o-f1-s core	weight ed-f1- score
1.1 milyon	Logistic Regression	0.91	0.88	0.90	0.85	0.91	0.86	0.91

Tablo 19. Amazon Products Dataset 2023 veri setinden BoW vektörizasyon yöntemi kullanılarak elde edilen veri ile eğitilmiş modellerin değerlendirilmesi

4.4. Veritabanı Üzerinden Yapılacak Veri Analizleri

Veritabanında biriken işlem verileri, kullanıcı davranışlarını anlamak ve pazarlama

stratejilerini geliřtirmek iin analiz edilebilir. ncelikli olarak, kullanıcıların satın alma alışkanlıkları detaylı şekilde incelenebilir. Hangi ürünlerin en sık birlikte satın alındığı tespit edilerek apraz satış stratejileri oluşturulabilir. rneğın, belirli bir ürünü satın alan kullanıcıların genellikle hangi diğerk ürünleri tercih ettiğı analiz edilerek öneri mekanizmaları geliştirilebilir.

Bunun yanı sıra, demografik faktörlerin satın alma kararları üzerindeki etkisi araştırılabilir. Kullanıcıların yaşı ve cinsiyet gibi özelliklerine göre hangi ürünleri daha çok satın aldığı belirlenerek hedef kitleye özel pazarlama stratejileri geliştirilebilir. rneğın, genç kullanıcıların belirli kategorilere daha fazla ilgi gösterip göstermediğı analiz edilerek kampanyalar ve promosyonlar buna göre şekillendirilebilir.

Ürün fiyatlarının zaman içindeki değıřimi de önemli bir analiz alanıdır. Belirli ürünlerin fiyatlarındaki artış veya azalma, kullanıcıların satın alma kararlarını nasıl etkiliyor sorusu üzerinde durulabilir. Böylelikle fiyatlandırma stratejileri geliştirilerek daha uygun satış politikaları oluşturulabilir.

Ayrıca, kullanıcıların geçmiş alışveriş verilerine dayanarak onlara kişiselleştirilmiş öneriler sunmak mümkün olabilir. Belirli bir kategoriden sıkça alışveriş yapan kullanıcılara benzer ürünler önerilebilir. Bu sayede, müşteri deneyimi iyileştirilerek satış oranlarının artırılması hedeflenebilir.

Son olarak, veri analizi ve makine öğrenmesi yöntemleri kullanılarak geleceğey yönelik tahminler yapılabilir. řu ana kadar toplanan veriler doğırtusunda, hangi ürünlerin popürlüğünün artacağı ya da hangi kategorilerde talebin azalacağı önceden tahmin edilerek işletmelerin stratejik planlamalarına katkı sağlanabilir.

4.5. API Geliřtirme ve Veri Sunumu

API'nin kullanıcıya sunduğı veri erişimini genişletmek ve sistemin ölçeklenebilirliğini artırmak iin yeni geliřtirmeler yapılması gerekmektedir. Mevcut sistemde, kullanıcıların alışveriş geçmişine dair temel verilere erişimi bulunmaktadır. Ancak, API'nin daha kapsamlı analiz verileri sunabilmesi iin yeni endpoint'lerin eklenmesi planlanmaktadır. rneğın, en çok satılan ürünler, kişisel alışveriş istatistikleri ve ürün fiyat değıřim analizleri gibi detaylı veriler API aracılığıyla sağlanabilir.

Bununla birlikte, raporlama ve analiz özelliklerinin de API'ye entegre edilmesi önemlidir. Kullanıcıların alışveriş geçmişleri ve genel satış istatistikleri üzerinden detaylı raporlar oluşturulması, işletme sahipleri ve yöneticiler için büyük bir avantaj sağlayacaktır. Bu sayede, API aracılığıyla doğrudan satış analizlerine erişmek mümkün olacak ve veri odaklı karar alma süreçleri desteklenecektir.

Gerçek zamanlı veri sunma özelliği de sistemin geliştirilmesi gereken noktalarından biridir. Kullanıcıların taleplerine bağlı olarak belirli veri noktalarının dinamik olarak sağlanması planlanmaktadır. Örneğin, ürünlerin stok durumu, fiyat değişiklikleri veya anlık satış trendleri gerçek zamanlı olarak API aracılığıyla sunulabilir.

API'nin güvenliğini artırmak adına yetkilendirilmiş veri paylaşımı üzerinde de durulmalıdır. Kullanıcıların yalnızca kendi satın alma verilerine erişebilmeleri için JWT tabanlı kimlik doğrulama sistemlerinin etkin bir şekilde kullanılması gerekmektedir. Bu sayede, kullanıcı verilerinin güvenliği sağlanarak olası yetkisiz erişimlerin önüne geçilecektir.

Genel olarak, API'nin daha geniş bir veri sunum altyapısına sahip olması, sistemin kullanıcı deneyimini iyileştirmesi ve işletmelere daha fazla veri odaklı içgörüler sunması hedeflenmektedir. Yapılacak geliştirmeler sayesinde, hem kullanıcılar için daha kişiselleştirilmiş hizmetler sunulacak hem de işletmeler için stratejik karar alma süreçleri daha verimli hale getirilecektir.

5. KAYNAKÇA

- [1] Yıldız, İ., Kotan, A.B., Altınel Girgin, A.B. (2023). Türkçe Faturaların Sınıflandırılmasında Farklı Öznitelik Seçimi Yöntemleri ile Topluluk Öğrenme Algoritmalarının Etkilerinin İncelenmesi. Avrupa Bilim ve Teknoloji Dergisi, (52), 272-278.
- [2] W. Zhang, "Online Invoicing System Based on QR Code Recognition and Cloud Storage," 2018 2nd IEEE Advanced Information

- Management,Communicates,Electronic and Automation Control Conference (IMCEC), Xi'an, China, 2018, pp. 2576-2579
- [3] Thiée, Lukas-Walter, Felix Krieger, and Burkhardt Funk. "Extraction of information from invoices—challenges in the extraction pipeline." (2023): 1777-1792.
 - [4] R.W. Smith, The Extraction and Recognition of Text from Multimedia Document Images, PhD Thesis, University of Bristol, November 1987.
 - [5] S. V. Rice, F. R. Jenkins, and T. A. Nartker, "The Fourth Annual Test of OCR Accuracy," Information Science Research Institute, University of Nevada, Las Vegas, Apr. 1995.
 - [6] R. Smith, "An Overview of the Tesseract OCR Engine," Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), IEEE, ss. 1145-1148, 2007.
 - [7] O. Sarkar, S. Sinha, A. K. Jena, A. K. Parida, N. Parida and R. K. Parida, "Automatic Number Plate Character Recognition using Paddle-OCR," 2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET), Nagpur, India, 2024, pp. 1-7, doi: 10.1109/ICICET59348.2024.10616305.
 - [8] D. Zhang and Y. Li, "Research and Application of Health Code Recognition Based on Paddle OCR under the Background of Epidemic Prevention and Control," Journal of Artificial Intelligence Practice, vol. 6, no. 1, pp. 1-16, Feb. 2023
 - [9] Bradski Gary and Kaehler Adrian, “Learning OpenCV: Computer vision with the OpenCV library.”, O'Reilly Media, Inc., 2008.
 - [10] Mahesh, Batta. "Machine learning algorithms-a review." International Journal of Science and Research (IJSR).[Internet] 9.1 (2020): 381-386.
 - [11] Lokesh Parab, Amazon Products Sales Dataset 2023, Kaggle. Accessed online on 24 March 2025:
<https://www.kaggle.com/datasets/lokeshparab/amazon-products-dataset>