T.C. İSTANBUL TİCARET ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

DUYGU ANALİZİYLE MÜŞTERİ PROFİLİ OLUŞTURMA VE ŞİKAYET ÖNCELİKLENDİRME

Muhammed Mehmet AKGÜMÜŞ

Danışman
Dr. Öğr. Üyesi Ali BOYACI

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
İSTANBUL – 2023

KABUL VE ONAY SAYFASI

Muhammed Mehmet AKGÜMÜŞ tarafından hazırlanan "**Duygu Analiziyle Müşteri Profili Oluşturma ve Şikayet Önceliklendirme**" adlı tez çalışması 13/01/2023 tarihinde aşağıdaki jüri üyeleri önünde başarı ile savunularak, İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği Anabilim Dalı**'nda **Yüksek Lisans Tezi** olarak kabul edilmiştir.

Danışman	**Dr. Öğr. Üyesi Ali BOYACI** İstanbul Ticaret Üniversitesi	.....
Jüri Üyesi	**Dr. Öğr. Üyesi Metin TURAN** İstanbul Ticaret Üniversitesi	.....
Jüri Üyesi	**Dr. Öğr. Üyesi Zeynep TURGUT AKGÜN** İstanbul Medeniyet Üniversitesi	.....

Onay Tarihi : 10.02.2023

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsünün 10.02.2023 tarih ve 2023/374 numaralı Yönetim Kurulu Kararının 1. maddesi gereğince, ders yüklerini ve tez yükümlülüğünü yerine getirdiği belirlenen "Muhammed Mehmet AKGÜMÜŞ" adlı öğrencinin mezun olmasına oy birliği ile karar verilmiştir.

Prof. Dr. Doğan KAYA
Enstitü Müdürü

AKADEMİK VE ETİK KURALLARA
UYGUNLUK BEYANI

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

10.02.2023

Muhammed Mehmet AKGÜMÜŞ

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER	i
ÖZET	ii
ABSTRACT	iii
TEŞEKKÜR	iv
ŞEKİLLER DİZİNİ	v
ÇİZELGELER DİZİNİ	vi
SİMGELER VE KISALTMALAR DİZİNİ	vii
1. GİRİŞ	1
2. LİTERATÜR ÖZETİ	5
3. İLGİLİ KAVRAMLAR	12
3.1. Lojistik regresyon	12
3.1.1. İkili lojistik regresyon	13
3.1.2. Multinominal lojistik regresyon	13
3.1.3. Sıralı lojistik regresyon	13
3.2. Lstm	14
3.3. Multinominal Naive Bayes	16
3.4. Svm	16
3.5. Phyton	17
3.6. Nltk	18
3.7. Tensorflow	18
3.8. Scikit learn	18
3.9. Pandas	19
3.10. Seaborn	19
3.11. Matplotlib	19
3.12. Kpi	19
4. METODOLOJİ	21
4.1. Veri toplama	22
4.2. Ön işlem	23
4.3. Eğitim	23
4.4. Test	23
5. ARAŞTIRMA BULGULARI VE TARTIŞMA	24
6. SONUÇ VE ÖNERİLER	31
KAYNAKLAR	34
EKLER	38
EK A. Fotoğraflar	39
ÖZGEÇMİŞ	46

ÖZET

Yüksek Lisans Tezi

DUYGU ANALİZİYLE MÜŞTERİ PROFİLİ OLUŞTURMA VE ŞİKAYET ÖNCELİKLENDİRME

Muhammed Mehmet AKGÜMÜŞ

İstanbul Ticaret Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Ali BOYACI
2023, 46 sayfa

Müşteri şikayetlerini yönetmek için veriler NLP (Doğal Dil İşleme) yöntemleri kullanılarak sınıflandırılabilir. Bu çalışmada banka müşterilerinin talep, istek ve sorularının öncelikli olarak sınıflandırılması amaçlanmıştır. Böylece müşteri talep, istek ve sorularına en hızlı şekilde cevap vermeyi hedefleyerek müşteri memnuniyeti sağlanmaya çalışıldı. Çalışmamız literatür taramasında incelenen çalışmalarda en yüksek doğruluk değerlerinin elde edildiği Lojistik Regresyon, LSTM, Multinominal Naive Bayes ve SVM algoritmalarına odaklanılmıştır. Veri setini sınıflandırmak için detaylı olarak incelendi ve altı sınıfa ayrılması gerektiğine karar verildi. Yüzde Elli dokuz bandında doğruluk sağlayan bu sonuçlar, iki tahminden neredeyse birinin başarısız olduğu düşünüldüğünde başarısız oldu. Bu çalışmada geliştirilen yöntemle iki aşamalı yaklaşımla elde edilen test sonuçlarının, tek aşamalı yaklaşımla elde edilen sonuçlara kıyasla uygulamanın performansını yaklaşık yüzde on artırdığı gözlemlendi. Bu yaklaşımla sınıflandırma, doğruluğu artıracak şekilde iki adımda gerçekleştirilir. Sınırlı sayıda veri seti ile çalışarak bu sonuca ulaşılmıştır.

Anahtar sözcükler: Destek vektör makinesi, doğal dil işleme, lojistik regresyon, makine öğrenimi, Multinominal Naive Bayes, müşteri memnuniyeti, Türkçe metin sınıflandırma, uzun kısa süreli bellek.

ABSTRACT

M.Sc Thesis

CUSTOMER PROFILE CREATION AND COMPLAINT PRIORITIZATION WITH SENTIMENT ANALYSIS

Muhammed Mehmet AKGÜMÜŞ

İstanbul Commerce University
Graduate School of Applied and Natural Sciences
Department of Computer Engineering

Supervisor: Assist. Prof. Dr. Ali BOYACI
2023, 46 pages

Data can be classified using NLP (Natural Language Processing) methods to manage customer complaints. This study is aimed to classify the demands, requests, and questions of bank customers as a priority. Thus, we tried to ensure customer satisfaction by aiming to respond to customer demands, requests, and questions as quickly as possible. Our study focused on Logistic Regression, LSTM, Multinominal Naive Bayes, and SVM algorithms, where the highest accuracy values were obtained in the studies examined in the literature review. In order to classify the dataset, we examined it in detail and decided that we should divide it into six classes. These results, which achieved accuracy in the percent of fifty nine band, were unsuccessful, considering that almost one of the two predictions was unsuccessful. We observed that the test results obtained with the two-step approach with the method we developed in this study increased the performance of the application by about ten percent compared to the results obtained with the onestep approach. With this approach, the classification is performed in two steps, increasing the accuracy. Working with a limited number of data sets, we achieved this result.

Keywords: Customer satisfaction, logistic regression, long short term memory, machine learning, Multinominal Naive Bayes, support vector machine, natural language processing, Turkish text classification.

TEŞEKKÜR

Bu araştırma için beni yönlendiren, karşılaştığım zorlukları bilgi ve tecrübesi ile aşmamda yardımcı olan değerli Danışman Hocam Dr. Öğr. Üyesi Ali BOYACI'ya teşekkür ederim.

Çalışmayı yapabilmem adına gerekli olan veri setine ulaşabilmemi ve banka bilgisayarında olsa da çalışabilmemi sağlayan değerli müdürüm Sami İMAMOĞLU'na teşekkür ederim.

Tezimin her aşamasında beni yalnız bırakmayan kıymetli eşime, aileme sonsuz sevgi ve saygılarımı sunarım.

Muhammed Mehmet AKGÜMÜŞ
İSTANBUL, 2023

ŞEKİLLER

Sayfa

Şekil 3.1. Sigmoid lojistik regresyon grafiği 13
Şekil 3.2. Rnn mimarisi 14
Şekil 3.3. Lstm mimarisi 15
Şekil 3.4. Svm grafiği 17
Şekil 4.1. Metodoloji 21
Şekil 4.2. Veri setinin kategorilere göre adetleri 22
Şekil 5.2. İlk aşamada lojistik regresyon algoritmasıyla elde edilen karışıklık matrisi 25
Şekil 5.3. İlk aşamada Multinominal Naive Bayes algoritmasıyla elde edilen karışıklık matrisi 25
Şekil 5.4. İlk aşamada Svm algoritmasıyla elde edilen karışıklık matrisi 26
Şekil 5.5. İlk aşamada Lstm algoritmasıyla elde edilen doğruluk grafiği 26
Şekil 5.6. İlk aşamada Lstm algoritmasıyla elde edilen karışıklık matrisi 27
Şekil 5.7. İkinci aşamada lojistik regresyon algoritmasıyla elde edilen karışıklık matrisi 27
Şekil 5.8. İkinci aşamada Multinominal Naive Bayes algoritmasıyla elde edilen karışıklık matrisi 28
Şekil 5.9. İkinci aşamada Svm algoritmasıyla elde edilen karışıklık matrisi 28
Şekil 5.10. İkinci aşamada Lstm algoritmasıyla elde edilen doğruluk grafiği 29
Şekil 5.11. İkinci aşamada Lstm algoritmasıyla elde edilen karışıklık matrisi 29

Şekil A.1. Veri setinin temizlendiği eğitim ve test verisinin ayrıldığı kod bloğu .. 39
Şekil A.2. Temel yöntemde altı ayrı sınıfa ayrılmış verinin Lstm algoritmasıyla eğitimi ve test aşamasının kodlaması 40
Şekil A.3. İlk adımda iki ayrı sınıfın Lstm algoritmasıyla eğitim ve test aşamasının kodlaması 42
Şekil A.4. İkinci adımda üç ayrı sınıfın Lstm algoritmasıyla eğitim ve test aşamasının kodlaması 43
Şekil A.5. Multinominal Naive Bayes algoritmasının kodlaması 44
Şekil A.6. Svm algoritmasının kodlaması 45
Şekil A.7. Lojistik regresyon algoritmasının kodlaması 45

ÇİZELGELER

Sayfa

Çizelge 3.1. Bayes formulü 16
Çizelge 5.1. Test sonuçları 24

SİMGELER VE KISALTMALAR

Ar-Ge	Araştırma – Geliştirme
ATM	Automatic Teller Machine
BiLSTM	Bidirectional Long Short Term Memory
CCMS	Component Content Management Systems
CNN	Convolutional Neural Network
CUDASOM	Compute Unified Device Architecture Self Organizing Map
CPU	Central Processor Unit
DTM	Document Term Matrix
GPU	Graphics Processing Unit
IJCNLP	International Joint Conference on Natural Language Processing
KNN	K Nearest Neighbour
KPI	Key Process Indicator
KVKK	Kişisel Verileri Koruma Kanunu
LSTM	Long Short Term Memory
NLP	Natural Language Processing
ODI	Outcome-Driven Innovation
RNN	Recurrent Neural Network
SOR	Stimulus Organism Response
SVM	Support Vector Machine
TF-IDF	Term Fruqency – Inverse Document Fruqency

1. GİRİŞ

Türkiye'de bankacılık sektörü gelişmeye ve büyümeye devam etmektedir. Teknolojinin her geçen gün daha da gelişmesi insanların birçok şeye ulaşmasını kolaylaştırmıştır. Birçok bankacılık hizmetini müşterinin bankaya gitmesine gerek kalmadan yapabiliyor olması müşteri memnuniyetini artırır. Bankalar bu etmenler sebebiyle mevcut ürünlerini teknolojiye uyacak şekilde geliştirmeli ve müşterileri için kullanımı sürdürülebilir kılması gerekir.

Bankacılık sektöründe kullanılan E-CRM'in müşteri ilişkilerine etkisini inceleyen bir çalışmada, müşteri memnuniyeti sağlamada E-CRM'in pozitif etki yarattığı ifade edilmiştir. Çalışmada ifade edilene göre şimdiye kadar çalışılmamış olan bankacılık sektöründe müşteri deneyiminin aracı rolü aracılığıyla E-CRM, müşteri deneyimi ve müşteri memnuniyeti arasındaki ilişkiyi araştırmıştır. Bu araştırmada, veriler varsayılan modeli desteklemiştir. SOR modeli, çevresel uyaranların bireylerin duygusal ve bilişsel durumlarını etkilediğini belirtmektedir. E-CRM, şirketlere veya kuruluşlara, müşterilerinin görüşlerini 360 derecelik bir ızgarada değerlendirerek onlara iyi bir unutulmaz deneyim sunmalarına yardımcı olur. Sonuç olarak, müşterilerin hizmet sağlayıcılarıyla uzun vadeli ilişkiler kurma olasılığı daha yüksektir. Bu durum bankaların sık sık etkileşimde bulunmaları ve banka kaynaklı sorunlarla karşılaştıklarında müşterilerine en iyi hizmeti sunmaları gerektiği sonucuna varmamıza neden olmuştur. Bu etkileşim ve destek, olumlu bir müşteri deneyimine yol açar. SOR modelinin aynı teorisi uygulanırsa, olumlu ve akılda kalıcı deneyimler yaşayan müşteriler daha memnun olacak ve bankalarıyla kalıcı bir ilişki sürdürmeye istekli olacaklardır. Bu sonucun bulunması çalışmada belirtilen önceki çalışmalara göre uyumlu bulunduğu ifade edilmiştir.

Müşteri deneyimi ile müşteri memnuniyeti arasındaki pozitif ilişki, bankaların müşterilerle uzun vadeli ilişkiler geliştirmesi ve sürdürmesi gerektiğini göstermiştir. E-CRM faaliyetlerini geliştirmek, bankalara müşteri ilişkilerini geliştirmek için rekabet fırsatı verecektir. E-CRM bu nedenle müşteri memnuniyeti ile pozitif olarak ilişkilidir (Kumar, 2021).

Müşterilerine en iyi hizmeti sunmak için oldukça rekabetçi bir ortamda faaliyet gösteren çok sayıda banka bulunmaktadır. Bu rekabet ortamında müşteri taleplerine en hızlı şekilde cevap vermek esastır. Bu kapsamda dört farklı bankanın şikâyet yönetiminin iyileştirilmesine yönelik bir çalışma yapılmıştır (Hakiri, 2012). Ayrıca Endonezya'da kamu bankalarının bankacılık hizmetlerine müşteri şikayetlerinin etkisini incelemek için yapılan çalışmalarda şikâyet yönetiminin müşteri sadakati ve memnuniyeti üzerinde önemli bir etkiye sahip olduğu sonucuna varılmıştır (Salim, 2018) (Lu, 2010).

Bankacılık alanında müşteri işlemlerini kolaylaştırmak için yapay zekâ kullanılarak yapılmış birçok çalışma bulunmaktadır. Berke Oral ve diğerleri tarafından Bankacılık belgelerinden yapılan işlemleri çıkarmak için bir yöntem tanıtılmıştır (Oral, 2019). Yöntem, her varlık çifti arasındaki ilişkileri tahmin etmek için BiLSTM tabanlı derin sinir ağını kullanır ve bir ilişki grafiği oluşturur. Bu grafikten, kod çözme algoritması bir dizi alt grafik oluşturur ve işlem sayısını belirlemek ve boşluklarını doldurmak için maksimum klik çarpanlarına ayırma uygular. Yöntemin, işlemleri ve yuvalarını önceden önerilen yöntemlere göre tahmin etmede daha doğru olduğu ifade edilmektedir.

Türkiye'deki bankaların mevduat payları ve şikâyet oranları incelendiğinde, şikayetlerin sistemli bir şekilde yönetilmesinin ve müşterilerin memnuniyetinin bankaların nakit akışlarının devamlılığında önemli rol oynadığı iddia edilmektedir (Eser, 2016). Bu çalışma müşterileri memnun etmenin müşteriyi kaçırmamak adına çok önemli bir etken olduğundan bankaların bu alanda müşteri memnuniyetini sağlamak için çalışma yapmasının gerekliliğini ortaya koyar.

Banka Müşteri ilişkilerinin öneminin vurgulandığı bir çalışma, bankaların müşteri yönetimine ilişkin birtakım çıkarımlara sahiptir (Ndubisi, 2007). İlk olarak, sadık müşteriler kazanmak ve elde tutmakla ilgilenen özelde bankalar ve genel olarak hizmet kuruluşları, müşterilerle kaliteli ilişkiler kurmanın yanı sıra müşterilerin güvenini kazanmaya çalışmalıdır. Kârlılığın yanı sıra müşteri sadakati ile ilgili bir dizi fayda vardır. Müşteri sadakati, iş operasyonlarının maliyetini beş ila altı kat azaltabilir.

Çalışmada bahsedilen daha önce yayınlanmış olan raporlara göre yeni bir müşteriyi çekmenin mevcut bir müşteriyi elde tutmaktan beş ila altı kat daha maliyetli olduğunu gösterdiği ifade edilmektedir. Ayrıca, sadık müşteriler, işletme hakkında olumlu ağızdan ağıza iletişim sağlayarak yeni müşterileri de çekebilir. Böylece organizasyona yeni ürün fikirleri de katabilirler. Banka, güvenilir davranarak ve müşterilerle kaliteli ilişkiler kurarak, sadakati sağlamak için değiştirme maliyetini yükseltebilir.

Güven ve kaliteli bir ilişki sağlamak için, bankaların etkili bir şekilde iletişim kurması, kararlı olması ve çatışmaları iyi yönetmesi gerekir. Sözlerini vermeli ve tutmalı, müşterilere yeni hizmetler, hizmetlerdeki değişiklikler hakkında zamanında ve güvenilir bilgi iletmeli ve sorunlar meydana geldiğinde bankanın sorunla ilgili ne yaptığı konusunda müşterileri bilgilendirmelidir. Bunlar, bankaların güveni artırmak için iletişimi kullanabilecekleri yollardan bazılarıdır.

Bankalar, güven ve kalite ilişkisi yaratma stratejileri olarak müşterilerin ihtiyaçlarına göre ayarlamalar yapmaya, hizmet sağlamada esnek ve yenilikçi olmaya, müşterilere hizmet vermede esnek olmaya ve ayrıca kişiselleştirilmiş hizmetler sunmaya istekli olmalıdır. Son olarak, çatışmaları yetenekli bir şekilde ele almak da güveni ve ilişki kalitesini artıracaktır. Çatışmaların üstesinden gelmenin hem proaktif hem de tepkisel yolları, yani potansiyel çatışmalardan kaçınmak, çatışmaları ortaya çıkmadan önce çözmek ve sorunları açıkça tartışmak, bankanın müşteri güvenini kazanmasına ve onlarla kaliteli ilişkiler kurmasına yardımcı olacaktır.

Harkiranpal Singh'in birçok araştırmacı ve akademisyenin görüş ve araştırmalarına dayanarak yaptığı çalışmada, müşteri memnuniyetinin kurumlar için olmazsa olmaz kriterlerden biri olduğu sonucuna varılmıştır (Singh, 2006). Bu nedenle müşteri memnuniyeti, müşteriler için tekrar satın alma garantisi vermese bile müşteri sadakati ve müşteriyi elde tutmada kritik bir rol oynamaktadır. Kuruluşlar, müşterilerinin ürünlerinden memnun olmalarını sağlamak için çaba göstermelidir. Ayrıca, kullanıcı şikâyet verileri bir banka için KPI (Anahtar Performans Göstergesi) olarak kullanılabilir. Müşteri memnuniyeti de kullanıcı şikayetleri ile yüksek oranda ilişkilidir. Müşteriler ne kadar memnun olursa, şikayetlerinin o kadar az olacağı açıktır.

Müşteri memnuniyeti için yapılan bir çalışmada bu KPI 'lar kullanılarak müşteri puan kartı oluşturulur (Ibrahim, 2018).

Çağrı merkezi aracılığıyla kurum müşterilerinden kurumlara iletilen taleplerin kapsamlı olarak analiz edildiği uygulamalar da vardır (Sestek, 2023). Konuşma süresinin tutulduğu, konuşmanın monotonluk analizi, müşteri sinirlilik yüzdesi, metin içeriği filtreleme gibi özellikleri olan bu uygulama çok sayıda müşteriye hitap eden kurumlar tarafından kullanılmaktadır. Bu uygulamada, çağrı merkezleri için birçok kolaylığı sağlasa da müşterilerden toplanan verileri sınıflayamadığından müşteri bazlı tek tek veri inceleme gerekliliği vardır. Bu da öncelikli ulaşılmak istenen müşterilere ulaşmayı sağlamaz.

Müşteri şikayetlerinin ve taleplerinin yönetiminde müşterilerden bankaya gelen banka uygulamaları hakkındaki problem kayıtları, müşteri işlemleriyle ilgili istekler ve müşterilerin banka uygulamaları hakkında sorularının ayrıştırılması ve hızlı şekilde müşteriye ulaşılması önem arz etmektedir.

2. LİTERATÜR ÖZETİ

Telekom ve internet servis sağlayıcı şirket için yapılan bir araştırma; müşteri seslerinden özellikle öfke ve mutluluk duygusu olmak üzere kullanıcı duygularını doğru bir şekilde tanımlanabildiğini gösteriyor (Gong, 2015). Bu tanımlama, bireysel müşteriler için uygun bir yaklaşım oluşturmaya yardımcı olabilir. Ek olarak, bu bilgiler şirketlerin kullanıcı memnuniyetine yardımcı olabilecek özel müşteri hizmetlerine sahip olmalarına yardımcı olabilir.

Başka bir çalışmada ise İngilizce ve Tagalog dilinde 1752 kayıttan oluşan veri seti üzerinde müşteri şikayetlerinin sınıflandırılması yapılmıştır (Corpuz, 2021). Bu çalışmalarda müşteri şikayetleri için uluslararası bir kalite standardı bulunmaktadır. Çalışmada veri seti dört kategoriye ayrılmıştır (geri bildirim, şikâyet, övgü ve öneri). Bu çalışma sonucunda doğruluk değeri SVM (Destek Vektör Makinesi) algoritması ile %98,63, LSTM (Uzun Kısa Süreli Bellek) algoritması ile %99,32 olarak ölçülmüştür. Ancak verilen değerler sadece eğitim adımlarının sonuçları olduğu için belirsizdir.

Şikayetler müşteriler hakkında bilgiye dönüştürülürse, işletmeler için değerli bir iş zekâsı sağlayabilirler. Şu anda, çoğu müşteri şikâyet yönetimi çözümü, bir şikâyeti belirli bir alana özgü sınıfla ilişkilendirmek için anahtar kelime işlemenin kullanımıyla sınırlıdır. Makalede belirtildiği üzere mevcut endüstriyel CCMS platformları, daha yavaş performans, kalite güvencesi ve sürdürülebilirlik maliyetlerinden kaçınmak için doğal dil işleme veya makine öğrenimi tekniklerini kullanmaz, bu nedenle şikâyet işleme işlevlerinin çoğu hala manueldir.

Boris A. Galitsky ve diğerlerinin yaptığı çalışmada, müşteri-şirket diyaloglarıyla ilişkili şikâyet senaryolarının modellenmesi ve sınıflandırılması için yeni bir yaklaşım sunulmaktadır (Galitsky, 2009). Bu tür diyaloglar hem şirketin hem de müşterinin iletişimsel eylemler yoluyla etkileşime girdiği ve puanlarını destekleyen argümanlar sunduğu etiketli grafikler olarak resmileştirilir. Bu tür argümantasyonun, iletişimsel eylemler üzerinde makine öğrenimi akıl yürütmesini gerçekleştirmek için bir

tamamlayıcı sağladığını ve sonuçta ortaya çıkan sınıflandırma doğruluğunu iyileştirdiğini göstermektedir.

Junegak Joung ve diğerleri tarafından yapılan bir çalışmada veri setindeki çok sık veya çok nadiren meydana gelen sözcükler kaldırılır (Joung, 2018). Tek bir anahtar kelime isimlerle tanımlanır ve birden fazla anahtar kelime, istatistiksel özellikleri kullanarak doğal dil dizilerini modelleyen n-gram modelleme ile tanımlanır.

Bu tanımla birlikte Sonuç Odaklı Yenilik (ODI) yöntemini kullanarak müşterilerin gerçek ihtiyaçlarını belirlemek için müşteri şikayetlerinin veri madenciliği analizinin uygulamasını sunar. Yapılan bu çalışmada iş kavramını kullanarak müşteri şikayetlerini analiz etmek için bir yöntem sunulmaktadır. ODI tabanlı analiz, müşteriler tarafından önceki yöntemlerin keşfedemediği ürün kullanımının uygulama öncesi ve uygulama sonrası adımları sırasında müşteri gizli ihtiyaçlarının belirlenmesine katkıda bulunur. Önerilen yöntemin müşteri gereksinimlerini nasıl tanımlayabileceğini açıklamak için stand tipi klimalarla ilgili bir vaka çalışması sunulmuştur. Analiz, uzmanların belirlemediği ancak önemli olarak gördüğü iki ihtiyacı belirlediği ifade edilmiştir. Bu araştırma, müşterilerin üründen yardım almak istedikleri tüm noktaların gereksinimlerini belirlemeye yardımcı olur.

Joao Filgueiras ve diğerleri tarafından ele alınmış başka bir makalede, Portekiz Ekonomik ve Gıda Güvenliği Kurumu bağlamında şikayetleri analiz etmek ve sınıflandırmak için hangi tekniklerin kullanıldığı hakkında bilgi verilmiştir (Filgueiras, 2019) Operasyonel sürecini temel alarak, üç farklı sınıflandırma problemi ele alınmış: hedef ekonomik aktivite, zımni ihlal şiddet seviyesi ve kurumsal yeterlilik. %70'in üzerinde doğruluk elde edilmiştir. Şikayetle ilgili ekonomik faaliyet türüne göre, ihlalin ciddiyeti ve yetkinliğe göre sınıflandırılmıştır. Birçok algoritma kullanılmış en yüksek doğruluk değeri SVM ile elde edilmiştir.

Rachmawan Adi Laksono ve diğerleri tarafından yapılan bir araştırma sonucuna göre; finans sektöründeki para akışının daha yüksek olmasından kaynaklı olarak ufak bir problem bile büyük krizlere sebep olabilir. Bu doğrultuda bankalar gibi finans

sektöründe hizmet veren kurumlar için şikâyet yönetimi daha kritik hale gelir. İlgili kuruma maliyeti çok yüksek olabilir (Laksono, 2019).

Araştırmalar, müşteri şikayetlerini etkin bir şekilde yönetmek için verilerin uygun şekilde sınıflandırılması gerektiğini göstermektedir. Bu sınıflandırma NLP yöntemleri kullanılarak yapılabilir. Doğal dil işlemede sıklıkla kullanılan yöntemlerden biri, Zellig Harris'in tanıttığı metindeki noktalama işaretlerini kaldırmak, kelimelerin kökünü bulmak ve her metindeki kelimelerden bir bag of words(kelime çantası) oluşturmaktır (Harris, 1954). Bu yöntem ile veri setindeki her kelimenin metinde geçme sıklığı bulunur. Bu yöntem yazım hatalarının göz ardı edileceği varsayılarak çoğu dilde iyi çalışır. İncelediğimiz bir çalışmada; metinleri ve görsel içerikleri sınıflandırmak için Bag of Words yöntemi kullanılmıştır (Qader, 2019).

Metnin dili ve olası yanlış yazılmış kelimeler, sınıflandırmanın doğruluğunu etkiler. Diller yapılarına göre farklılık gösterir. Türkçe, Moğolca, Korece ve Japonca gibi diller sondan eklemeli diller grubuna girer. Sondan eklemeli diller; kelimenin köküne son ekler ekleyerek farklı kelimeler oluşturabilir. Örneğin; ”Süt” kelimesi ”-çü” ekini alarak ”Sütçü” olur ve yeni bir kelime oluşturur. Bu tür sondan eklemeli dillerde bir kelimenin kökünü bulmak kolay değildir. Sondan eklemeli dillerin makine öğrenmesi ile morfolojik analizini yapan bir çalışmada, SVM algoritmasının diğer algoritmalardan daha iyi performans gösterdiği görülmüştür (Kumar, 2009).

Çok sayıda şikâyet mesajı alan şirketler için bu mesajları sınırlı kaynaklarla işlemek zordur. Bu konu, kamu tarafından finanse edilen bir araştırma projesinin bir parçası olarak ele alınmıştır (Kintz, 2020). Şikâyet yönetimi NLP ile sınıflandırılarak otomatik bir yanıt sistemi tasarlanmıştır. Çeşitli uzmanlara göre bu sistemin uygulanabilir olduğu sonucuna varılsa da müşterilerden toplanan verilerle test edilmediği için doğruluğu tartışmaya açık kalmıştır.

Sosyal medyadan toplanan 12790 veriden oluşan Türkçe bir metin üzerinde yürütülen çalışmada (Yıldırım, 2015), tüm deneylerde SVM kullanılmış ve duyarlılık analizinin güven düzeyini artırmak için cross validation(çaprazlama tekniği) uygulanmıştır.

Pozitif, negatif ve nötr olmak üzere üç sınıfa ayrılan veri setinde %78 doğruluk elde edilmiştir.

Türkçe Haber Metinlerinin sınıflandırıldığı bir çalışmada, CNN ve Word2Vec metodu kullanılarak Turkish Text Classification 3600 (TTC-3600) veri kümesi üzerinde metin sınıflandırma çalışması yapılmıştır (Çiğdem, 2019). Bu veri kümesi kullanılarak yapılan başka bir çalışma ile kıyaslanmıştır. Çalışmada TTC-3600'ün ham ve Zemberek yazılımıyla gövdelenmiş halleri üzerinde iki farklı KSA eğitilmiş ve test edilmiştir. CNN (Convolutional Neural Network) ve Word2Vec metodu, klasik istatistiksel ve makine öğrenmesine dayalı sınıflandırma algoritmalarından daha iyi bir performans (%93,3 doğruluk) gösterdiği vurgulanmıştır. Türkçe doğal dil işleme çalışmalarının azlığı ve bu alandaki özellik çıkarma yöntemlerinin limitli olması sebebiyle, kelimelerin semantik değerlerinin önceden eğitilmiş Word2Vec ağı ile sınıflandırmaya katılabilmesi CNN modellerinin doğruluk değerlerini arttırdığı belirtilmiştir. Veri setinde yazım yanlışı olmaması çalışmanın performansını yüksek oranda etkilediği düşünülmektedir.

Rohit Gavval et al. CUDASOM (CUDA-tabanlı Kendi Kendini Düzenleyen Özellik Haritası) ile müşteri şikayetlerinin Görsel Farkındalık Analizi için SOM (Kendini organize eden özellik haritası) özelliğini NVIDIA'nın paralel hesaplama platformu CUDA (Compute Unified Device Architecture) kullanan tanımlayıcı-analitik bir çerçeve önermiştir (Gavval, 2019). Önerilen modelin etkinliği, önde gelen dört Hint bankasının ürün ve hizmetlerine ilişkin müşteri şikayet verileriyle kanıtlanmıştır. Modelin performansı, belgelerin iki vektör uzayı modeli temsili, yani terim frekansı gösterimi ve TF-IDF (Terim Frekansı-Ters Belge Frekansı) gösterimi ile karşılaştırılır. Yazar modelin TF-IDF tabanlı DTM ile frekans tabanlı DTM terimine göre daha spesifik bir segmentasyon ürettiğini belirtmiştir. Görselleştirme, CUDASOM'un ürünler-hizmetlerle ilgili şikayetleri etkin bir şekilde farklı gruplara ayırdığını göstermektedir. Yazar algoritmanın ayrıca seri versiyona kıyasla önemli bir hızlanma gösterdiğini ve büyük boyutlu verilerin daha hızlı analizi için faydalı olarak kabul edilebileceğini ifade etmiştir.

Deepak Gupta ve diğerleri tarafından yapılan bir çalışmada, kullanıcıların herhangi bir dilde geri bildirim yazabileceklerinden bahsedilmiştir. Bu nedenle uygun bilgiyi aramak genellikle zordur (Gupta, 2017). Özellikle geleneksel öznitelik tabanlı kontrollü bir modelde, ilgili öznitelikleri bulmak için ilgili dili anlamak gerektiğinden genel bir sistem oluşturmanın zor olduğu ifade edilmiştir. Bu problemlerin üstesinden gelmek için CNN ve RNN (Repetitive Neural Network) tabanlı yaklaşımlar önerilmiştir. Bu teknikler; müşteri geri bildirim cümlelerini İngilizce, Fransızca, Japonca ve İspanyolca olarak analiz etmek için değerlendirildi. Çalışmada ifade edilen deneysel analiz, modellerin IJCNLP Müşteri Geri Bildirim Analizi üzerine Paylaşılan Görev kurulumlarında dört dilde de iyi performans gösterdiğini göstermektedir. Bu model; IJCNLP Fransızca için %71,75 doğruluk oranıyla ikinci, diğer tüm diller için üçüncü sırada yer almıştır.

Metin sınıflandırmada sıklıkla kullanılan Lojistik Regresyon algoritması, bir sonucu belirleyen bir veya daha fazla bağımsız değişken içeren bir veri setini analiz etmek için kullanılan istatistiksel bir yöntemdir (Hosmer, 2013). Lojistik Regresyon ile diğer bir sınıflandırma algoritması olan Multinominal Naive Bayes algoritmasının performansını karşılaştıran bir chatbot uygulamasında Lojistik Regresyonun Multinominal Naive Bayes algoritmasından daha başarılı çalıştığı gözlemlenmiştir (Setyawan, 2018).

LSTM, bir RNN mimarisidir (Hochreiter, 1994). Öğrenilen ilerleme kaydedildiğinde saklanan değerler değişmez. RNN'ler, nöronlar arasında ileri ve geri bağlantılara izin verir. Bir metin sınıflandırma çalışmasında CNN, alternatif RNN ve LSTM algoritmaları karşılaştırılmış ve LSTM'nin en iyi performansı gösterdiği belirtilmiştir (Bai, 2018).

Naive Bayes Algoritması, çalışma yöntemleri açısından farklılık gösterir. Metin sınıflandırmada en çok kullanılan yöntemlerden ikisi Multinominal Naive Bayes algoritması ve Bernoulli Naive Bayes algoritmasıdır. Bu iki algoritmanın karşılaştırıldığı bir metin sınıflandırma çalışmasında 312 kayıttan oluşan bir veri seti ön işleme tabi tutulmuş ve sonuç olarak Multinominal Naive Bayes algoritmasının çok az bir farkla başarılı bir şekilde çalıştığı görülmüştür (Singh, 2019). Naive Bayes algoritması koşullu

olasılıklara dayanır, hedef sınıfta belirli bir değerin oluşma olasılığını inceler ve buna göre hedef sınıfın değerini tahmin eder.

SVM ilk olarak 1995 yılında Corina Cortes ve Vladimir Vapnik tarafından piyasaya sürülmüştür (Cortes, 1995). Sınıflandırma ve Regresyon problemleri için kullanılan en popüler Denetimli Öğrenme algoritmalarından biridir. Ancak, öncelikle Makine Öğrenimindeki Sınıflandırma problemleri için kullanılır. SVM algoritmasının amacı, aşağıdaki yeni veri noktasını doğru kategoriye hızlı bir şekilde koyabilmek amacıyla n-boyutlu uzayı sınıflandırmak için en iyi çizgi veya karar sınırını oluşturmaktır. Bu en iyi karar sınırına hiper düzlem denir. SVM, hiper düzlemin oluşturulmasına yardımcı olan uç noktaları (vektörleri) seçer. Bu uç durumlara destek vektörleri denir. Bu nedenle algoritma destek vektör makinesi olarak adlandırılır. KNN, Naive Bayes ve SVM algoritmalarını karşılaştıran bir metin sınıflandırma çalışmasında; üç algoritma için yakın sonuçlar elde edilmesine rağmen SVM'nin F1 Skoru, kesinlik ve geri çağırma değerlerinin daha üstün olduğu görülmüştür (Liu, 2010).

Bu çalışmada doğal dil işleme yöntemleri üzerine yapılan çalışmalar araştırılmış ve doğal dil işlemede kullanılan birçok algoritma incelenmiştir. Müşteri şikayetlerinin analizi ve ayrıştırılmasına yönelik bu çalışmada Türkçe metinleri sınıflandırmak için farklı algoritmalar kullanıldı. Ancak belirlenen sınıf sayısının fazla olması nedeniyle algoritmaların elde ettiği doğruluk değerleri düşmüş ve istenilen sonuçlar elde edilememiştir.

Müşterilerden bankaya gelen geri bildirim verilerinin bankaya maliyeti ve müşteri memnuniyeti göz önünde bulundurularak veri en optimal şekilde sınıflandırılmaya çalışılmış ve sınıflara işlem aciliyeti belirlenerek öncelik verilmesi hedeflenmiştir. Bu nedenle çalışmada gerçek banka verilerinin daha detaylı sınıflandırılması yapılmıştır. Bu çalışmaları daha etkin uygulayabilecek ve önceliklendirilmesi istenen sınıflarda daha yüksek performans elde edebilecek bir yöntem sunulmuştur. Bu yöntem literatür taramasındaki tek aşamalı sınıflandırma yönteminden farklı olarak iki aşamada yapılmış hem doğruluğun artırılması hem de öncelikli ulaşılmak istenen kayıtların ayrıştırılması gibi durumların daha efektif yapılması amaçlanmıştır.

Yazının devamında: İlgili kavramlar bölümü; çalışmayı yaparken kullanılan materyalleri, Metodoloji bölümü; kullanılan veri seti, önerilen yöntem, veri sınıflandırma işleminin detayları hakkında bilgi vermektedir. Sonuç bölümünde ise sonuçlar verilmektedir. Son olarak sonuç bölümünde, elde edilen sonuçlar daha önce yapılmış çalışmalarla karşılaştırılmış ve ileriye dönük çalışmalar ifade edilerek tez sonlandırılmıştır.

3. İLGİLİ KAVRAMLAR

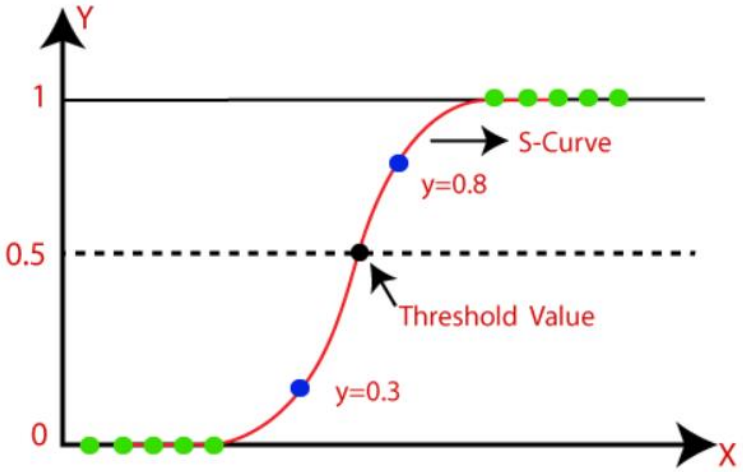
3.1. Lojistik Regresyon

Lojistik regresyon, Denetimli Öğrenme tekniği kapsamında yer alan en popüler Makine Öğrenimi algoritmalarından biridir. Belirli bir bağımsız değişken kümesini kullanarak kategorik bağımlı değişkeni tahmin etmek için kullanılır (Saishruthi, 2018).

Lojistik regresyon, kategorik bir bağımlı değişkenin çıktısını tahmin eder. Bu nedenle, sonuç kategorik veya ayrık bir değer olmalıdır. Evet veya Hayır, 0 veya 1, doğru veya Yanlış vb. olabilir, ancak 0 ve 1 olarak kesin değeri vermek yerine, 0 ile 1 arasında kalan olasılık değerlerini verir.

Lojistik Regresyon, Doğrusal Regresyona çok benzer. Regresyon problemlerini çözmek için Lineer Regresyon, sınıflandırma problemlerini çözmek için Lojistik Regresyon kullanılır. Lojistik regresyonda, bir regresyon çizgisi uydurmak yerine, iki maksimum değeri (0 veya 1) öngören "S" şeklinde bir lojistik fonksiyon üretilir. Lojistik fonksiyondan gelen eğri, bir şeyin olup olmama olasılığını gösterir. Lojistik Regresyon, sürekli ve ayrık veri kümeleri kullanarak olasılıklar sağlama ve yeni verileri sınıflandırma yeteneğine sahip olduğu için önemli bir makine öğrenimi algoritmasıdır.

Lojistik Regresyon, farklı veri türlerini kullanarak gözlemleri sınıflandırmak için kullanılabilir ve sınıflandırma için kullanılan en etkili değişkenleri kolayca belirleyebilir (Javatpoint, 2022). Şekil 3.1.'de Lojistik Regresyon grafiği gösterilmiştir.



Şekil 3.1. Sigmoid lojistik regresyon grafiği

Üç ayrı regresyon çeşidi vardır.

3.1.1 İkili lojistik regresyon

İki ayrı sınıfa bölünmüş veri setlerinde uygulanabilen algoritmadır.

3.1.2 Multinominal lojistik regresyon

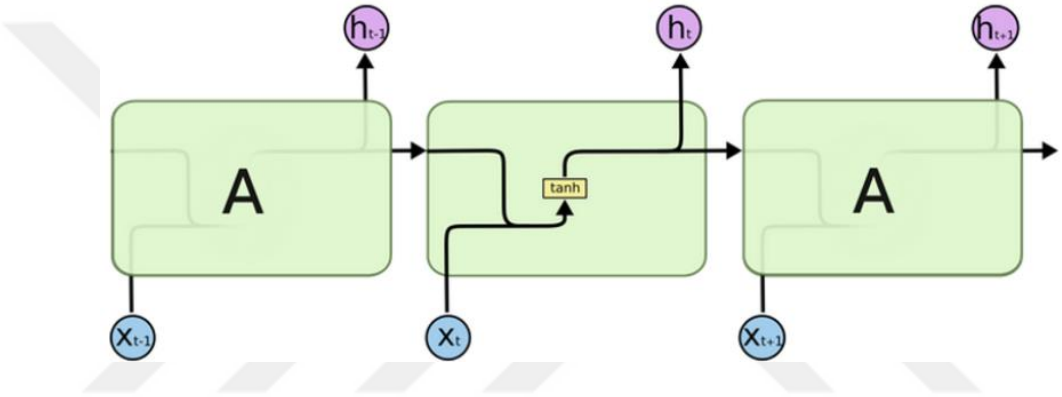
Üç veya daha fazla sınıfa ayrılmış veri setlerinde uygulanabilen algoritmadır.

3.1.3 Sıralı lojistik regresyon

Sıralı Lojistik Regresyon modeli, McCullagh tarafından geliştirilmiş bir modeldir. Yanıt değişkenin kategorilerinin sıralı ölçekle ölçüldüğü durumlarda uygulanan bir tekniktir. Yanıt değişken en az üç kategoriye sahiptir. Kategoriler doğal bir sıraya göre ve küçükten büyüğe doğru kodlanmalıdır. Sıralı kategorik değişkene, müşteri memnuniyetinin puanlanması durumunda en çok memnundan en memnun olmayana doğru sıralanması örneği verilebilir.

3.2. Lstm (Long short term memory)

İnsanlar yaşadıkları ve yaptıkları eylemlerin sonuçlarından bazı öğrenimler çıkarırlar. Bu çıkarımlar hafızada yedeklenir ve gerektiğinde tekrar kullanarak ilerleme sağlanır. Sinir ağlarında da bazı girdiler gerektiğinde tekrar tekrar kullanılarak yeni çıktılar elde edilir ve hafızada tutulur. Geleneksel sinir ağlarında geçmişteki bilgileri hatırlama mekanizması bulunmamaktadır. Tekrarlanan sinir ağları bilginin farklı adımlarda aktarılmasını sağlamak için tasarlanıp zaman içerisindeki ve dizi halinde olan verilerin işlenmesi için tasarlanmıştır.



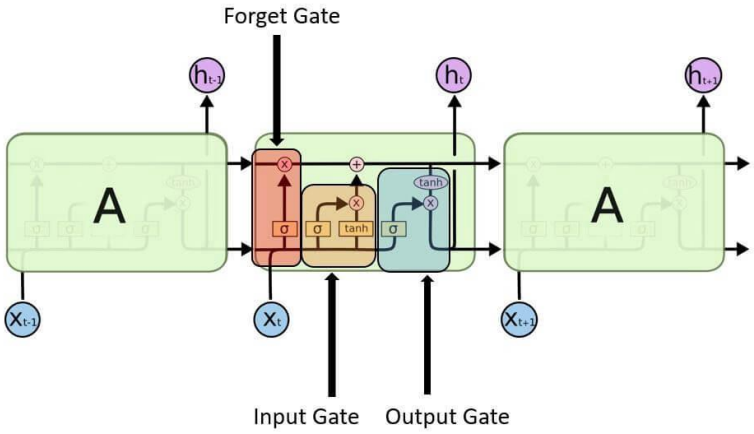
Şekil 3.2. Rnn mimarisi

Şekil 3.2. ‘deki gibi bir sinir ağı modeli olan A ünitesi t zamanındaki xt girdisini alıp ht çıktısını üretir. RNN’lerdeki döngü yapısı, üretilen çıktını t+1 zamanında girdi olarak kullanılabilmesini sağlamaktadır. Bu yapı, zaman serisi halinde olan ve zaman adımları arasında ilişki bulunan verilerin işlenmesini bir sinir ağı ile mümkün kılmaktadır. RNN modeli, birden fazla sinir ağının zaman içerisindeki tekrarlanması ile elde edilen tekrarlayan bir sinir ağıdır. Her bir ünite o andaki veriyi işleyip elde ettiği çıktıyı bir sonraki adıma ileterek bu ağı oluşturur.

RNN’lerin uzun bir geçmiş ile karşı karşıya kaldığı durumlarda istenen önceki girdiyi eksik veya yanlış hatırlaması gibi hatalarla karşılaşılmaktadır. Bu problem RNN’de unutkanlık problemi olarak bilinmektedir. Teorik olarak; RNN’ler bu tür uzun bağımlılıkları hatırlayıp unutmamaları gerekmektedir. Fakat gerçek kullanımlarda bu

her zaman doğru olmayıp model uzun girdilerde unutkan olmaya başlar. Bu sorunu çözmek için Uzun Kısa Süreli Bellek üniteleri geliştirilmiştir.

Temel olarak RNN içerisindeki normal bir adımın farklılaştırılmasıdır. Bu farklılaşma düz ilerleyen bir sinir ağı yapısı yerine içerisinde farklı birkaç matematiksel işlem daha yapılarak yeni parametreler hesaplanır ve saklanır. Böylece daha önceki adımlardaki girdi ya da çıktı bilgilerine gerek duyulduğunda ulaşılabilmektedir (Dobilas, 2022).



Şekil 3.3. Lstm mimarisi

Forget Kapısı: 0 ve 1 den oluşmaktadır. Önceki hidden state ve mevcut inputlar bu kapıya girip vektörel bir sonuç elde edilir. Vektör değerleri 0 ile 1 arasındadır. Kapının çıktısındaki vektör ile önceki c element tabanlı çarpılır eğer sonuç 0 ise değerler atılır/dikkate alınmaz, 1 ise bilgiler saklanır.

Input Kapısı, klasik bir RNN'de gerçekleştirilen işlemleri gerçekleştiren kapı, güncellenecek durumları oluşturur. Output Gate ise sonraki hidden state'i üreten kapıdır.

3.3. Multinominal Naive Bayes

Bu algoritma, koşullu olasılıklara dayanarak hedef sınıftaki belirli bir değerin gerçekleşmesi ihtimalini inceler ve buna dayalı olarak hedef sınıfın değerini tahmin eder. Bu hesaplama Bayes Formülü marifetiyle gerçekleştirilir (Great, 2022).

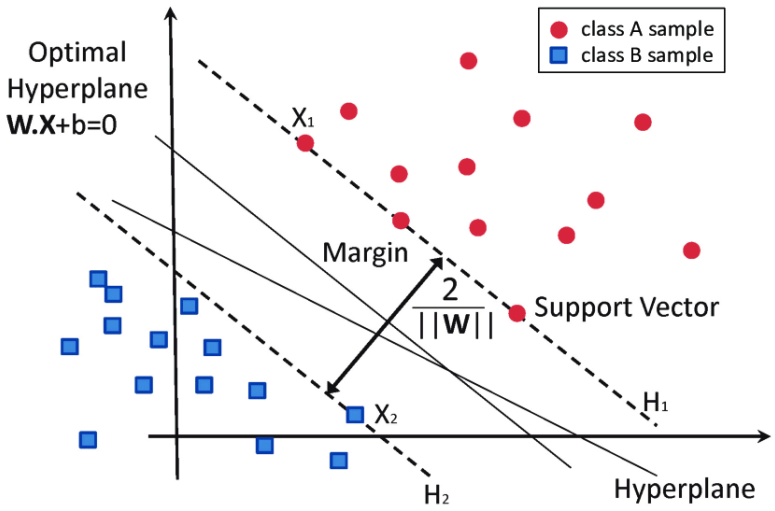
$$P(c|x) = \frac{P(C|X)P(c)}{P(x)} \quad (3.1)$$

Burda c Tahmin edilmeye çalışılan sınıfı x, tahmin eden sınıfı P(c|x); x olayı gerçekleştiğinde c olayının gerçekleşme olasılığını P(x|c); c olayı gerçekleştiğinde x olayının gerçekleşme olasılığını P(c) c olayının gerçekleşme olasılığını P(x) ise x olayının gerçekleşme olasılığını ifade etmektedir.

3.4. Svm (Destek vektör makinesi)

Destek Vektör Makineleri genellikle sınıflandırma problemlerinde kullanılan gözetimli öğrenme yöntemlerinden biridir (Sunil, 2017). Bir düzlem üzerine yerleştirilmiş noktaları ayırmak için bir doğru çizer. Bu doğrunun, iki sınıfının noktaları için de maksimum uzaklıkta olmasını amaçlar. Karmaşık ama küçük ve orta ölçekteki veri setleri için uygundur.

Sınıflandırma problemlerindeki asıl amaç gelecek verinin hangi sınıfta yer alacağına karar vermektir. Bu sınıflandırmanın yapılabilmesi için Şekil 3.4.'de belirtilen iki ayrı sınıfa ait A ve B verilerinden oluşan sınıfları ayıran bir doğru çizilir. Bu doğrunun dışında doğruya en yakın noktalarda bulunan destek vektörlerinin doğruya paralel çizilen (kesik çizgiyle gösterilmiş) iki doğru düzlemi arasında kalan bölgeye Margin adı verilir. Margin ne kadar geniş ise iki veya daha fazla sınıf o kadar iyi ayrıştırılır.



Şekil 3.4. Svm grafiği

3.5. Phyton

Python esas olarak nesne tabanlı programlamayı, belli bir oranda da fonksiyonel programlamayı destekleyen genel amaçlı bir programlama dilidir (Phyton, 2022). 1991 yılında ilk sürümü geliştirilen Python; görece kolaylığı, geniş standart kütüphanesi ve dinamik yapısı nedeniyle günümüzde oldukça popülerleşmiş ve yaygın bir kullanıcı kitlesine ulaşmıştır.

Özellikle bilimsel ve hesaplamalı mühendislik konularında sıkça tercih edilen Python programlama dili, 2000'li yıllardan itibaren yaşanan donanımsal ve yazılımsal gelişmeler ile paralel olarak geniş bir kullanım alanına kavuşmuştur. Python gibi yorumlanan diller yüksek seviyede veri yapıları kullandığından hafızada çok yer tutar ancak günümüzde yüksek bellek seviyelerine ulaşmak kolay olduğundan, bu durum önemli bir olumsuzluk olmaktan çıkmıştır.

Özgür ve açık kaynak kod mantığına dayanan Python'ın standart kütüphanesi, geliştirme araçları ve diğer birçok kütüphane, lisansa gerek duyulmaksızın açık kaynak kod olarak ücretsiz şekilde indirilebilmektedir. İnsanlar aracılığıyla toplanan verinin işlenmesi eğitime sokulup ilgili yapay zekâ algoritmalarıyla bir sonuca ulaşılması için kolayca kullanılabilecek olan bu programlama dili kullanılmıştır.

3.6. Nltk

İnsanların oluşturduğu verilerle çalışacak Python programları oluşturmaya yönelik bir programlama kütüphanesidir (Natural Language Toolkit, 2022). Sınıflandırma, tokenizasyon, gövde oluşturma, etiketleme, ayrıştırma ve anlamsal akıl yürütme için bir metin işleme kitaplıkları paketi, WordNet gibi 50'den fazla derlem ve sözcüksel kaynağa kullanımı kolay arabirimler sağlar.

Hesaplamalı dilbilim konularının yanı sıra programlama temellerini tanıtan uygulamalı bir kılavuz ve ayrıca kapsamlı API belgeleri sayesinde NLTK, dilbilimciler, mühendisler, öğrenciler, eğitimciler, araştırmacılar ve benzer şekilde endüstri kullanıcıları için uygundur. NLTK, Windows, Mac OS X ve Linux için kullanılabilir. Hepsinden iyisi NLTK ücretsiz, açık kaynaklı, topluluk odaklı bir projedir.

3.7. Tensorflow

Açık kaynak kodlu bir deep learning(derin öğrenme) kütüphanesidir (Tensorflow, 2022). Esnek yapısı sayesinde tek bir API ile platform fark etmeksizin hesaplamaları; bir veya birden fazla CPU, GPU kullanarak deploy etmenize olanak sağlar. Temelinde Python kullanılarak geliştirilen bu framework; günümüzde Python'ın yanısıra C++, Java, C#, Javascript ve R gibi birçok dili desteklemektedir. LSTM Algoritmasının uygulanmasında bu kütüphaneden yararlanılmıştır.

3.8. Scikit Learn

Scikit-learn sklearn olarak da bilinir. Python programlama dili için ücretsiz bir makine öğrenimi kütüphanesidir (Aryan, 2022). SVM, Random Forest, gradyan artırma, k-means ve DBSCAN dahil olmak üzere çeşitli sınıflandırma, regresyon ve kümeleme algoritmalarına sahiptir. NumPy, SciPy Python sayısal ve bilimsel kitaplıklarıyla birlikte çalışacak şekilde tasarlanmıştır. Scikit-learn, NumFOCUS tarafından mali olarak desteklenen bir projedir. Tezimizde Lojistik regresyon, Multinominal Naive Bayes ve SVM algoritmalarını kullanabilmek için bu kütüphane kullanılmıştır.

3.9. Pandas

Phyton programlama dilinde yazılmış olan veri işleme ve analiz için kullanılan bir kütüphanedir (Phyton Software Foundation, 2022). Bu çalışmada csv formatındaki veri setini phyton programlama dilinde işleyebilmek için kullanılmıştır.

3.10. Seaborn

Seaborn, matplotlib tabanlı bir Python veri görselleştirme kütüphanesidir (Waskom, 2022). Çekici ve bilgilendirici istatistiksel grafikler çizmek için üst düzey bir arayüz sağlar. Tezimizde Confusion Matrixleri oluşturmak için kullanılmıştır.

3.11. Matplotlib

Python'da statik, animasyonlu ve etkileşimli görselleştirmeler oluşturmak için kapsamlı bir kütüphanedir (Phyton Software Foundation, 2022). Projemizde LSTM algoritmasıyla elde edilen doğruluk değerlerinin grafik gösteriminde kullanılmıştır.

3.12. Kpi

Key Performance Indicator yani Anahtar Performans Göstergesi diye Türkçeye çevirebileceğimiz ve kısaca KPI olarak adlandırılan terim en genel anlamıyla ortaya konmuş olan temel hedeflerin ölçülebildiği bir sistemi ifade etmektedir (Gazitepe, 2022). Hedefi işaret eden anahtarların hangi performans ile sonuçlandığını takip etmek için tanımlanan terim hem kısa hem de uzun vadeli süreçler için uygulanmaktadır.

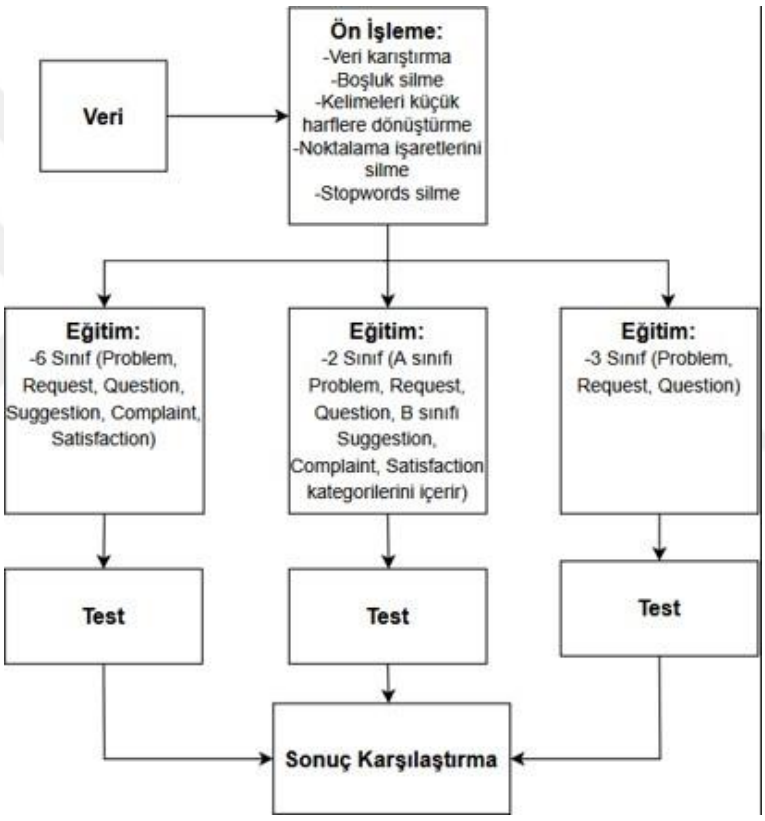
Temel Performans Göstergeleri yani KPI'ler bir diğer deyişle de amaçlanan bir sonuca doğru gidebilmek için belirlenmiş kritik yani anahtar göstergeleridir. Amaç ilerlemek olsa da başarılı KPI'lar stratejik ve eylemsel bir iyileştirme sonucu yaratmak üzere odaklanmayı destekler. Karar vermek adına analitik bir veri ortaya koyarken aynı zamanda da ölçülebilir bir durum sunar.

Hedef kelimesi iş gücünden ve operasyondan beklenen performans düzeyini ifade ederken KPI'lar bunun ölçülmesine hizmet eder. Ayrıca belirlenen hedefe ulaşmak için yapılan çalışmaların hangi noktasında etkin sonuçlar elde edildiğini de raporlanmasını sağlar (Ticimax, 2022). Bu sayede işletmeler mevcut veriler ile sonuçları kıyaslarken aynı şekilde gelecek beklentilerini de belirleyebilir. Bu çalışmada Bankaların KPI kullanarak müşteri şikayetlerine yönelik adaptif yaklaşımlar uygulayabileceği vurgulanmıştır.

4. METODOLOJİ

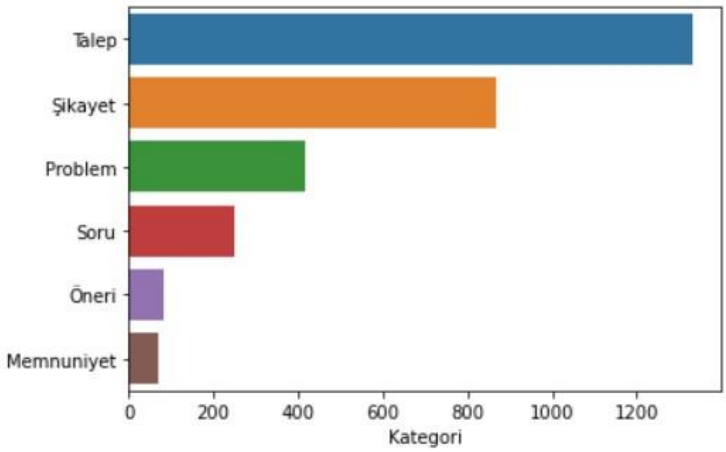
Bu çalışmada literatür taramasında incelenen çalışmalarda Doğal Dil İşleme yöntemlerinde en yüksek doğruluk değerlerinin elde edildiğinin saptandığı Lojistik Regresyon, LSTM, Multinominal Naive Bayes ve SVM algoritmalarına odaklanılmıştır.

Veri setini sınıflandırmak için detaylı olarak incelenmiş ve altı sınıfa ayrılması gerektiğine karar verilmiştir. Uygulanan metodoloji Şekil 4.1'de gösterilmektedir.



Şekil 4.1. Metodoloji

Her sınıf için veri miktarı Şekil 4.2.'de gösterilmiştir.



Şekil 4.2. Veri setinin kategorilere göre adetleri

Veri seti kategorize edilmeden önce detaylı şekilde incelenmiştir. Müşterinin bankadan aldığı hizmetlerde sorun olduğunu düşündüren metinler "Problem" olarak etiketlendi. Müşterinin bankadan herhangi bir konuda talepte bulunulduğunu düşündüren metinler "Talep", bankaya iletilmiş olan tüm sorular "Soru" olarak etiketlendi. Veri setinde bankaya gönderilen teklifleri ve banka hizmetlerinin kullanımına ilişkin öneriler (örneğin ATM yeri önerisi) "Öneri" olarak etiketlendi. Müşterinin beyanına göre; Tehdit, personelin tutumundan memnuniyetsizlik ve mağdur oldukları belirtilmiş olan ifadeler "Şikâyet" olarak etiketlendi. Son olarak müşterilerden gelen memnuniyet beyanları olduğu düşünülen metinler "Memnuniyet" olarak etiketlendi.

4.1. Veri Toplama

Veri seti, gerçek bir banka müşterilerinin farklı kanallardan bankaya ilettiği 6 aylık verilerden oluşmaktadır. 3020 kayıttan oluşan bu veri banka operatörleri tarafından kişisel veri içermeyecek şekilde maskelenerek csv formatında kullanılmak üzere iletilmiştir. Veri seti, halka açık bir veri seti olmayıp veriyi sağlayan kurumun ticari kaygıları nedeniyle paylaşılamaz. Veri müşterilerden gelme tarihine göre sıralanmaktadır.

4.2. Ön İşlem

Veri seti tarihe göre artan sırada olduğundan, benzer bir zamanda meydana gelen olaylar için yanılmayı en aza indirmek amacıyla karıştırıldı. Stopwords belirli bir anlamı olmayan ancak sözcükleri birbirine bağlamak için kullanılan sözcükleri ifade eder. İlk olarak tek başına bir anlam ifade etmeyen kelimeler (stopwords) veri setinden çıkarıldı. Veri setinden çıkarılacak stopwords için python programlama dilindeki NLTK kütüphanesini kullanıldı. Ardından metindeki kelimelerde kullanılan büyük harfler küçük harfe dönüştürüldü ve noktalama işaretleri kaldırıldı. Son olarak metindeki her kelimenin terim sıklığı bulundu ve Bag of Words oluşturuldu. Her kelime eşsiz olacak şekilde kodlanarak sayısal değerlere dönüştürüldü.

4.3. Eğitim

Tüm algoritmalar aynı eğitim ve test verileriyle çalışacak şekilde kodlandı. Veri setinin başlangıcından itibaren her bir algoritma için %70'i eğitim verisi ve kalan %30 'u test verisi olacak şekilde belirlendi. LSTM algoritması 10 adımda eğitim yaparken diğer algoritmalarda bir adımda eğitim aşaması tamamlandı. Phyton'daki Tensorflow kütüphanesi yardımıyla LSTM algoritması kodlandı. Phyton'daki Scikit-learn kütüphanesiyle de Lojistik Regresyon, Multinominal Naive Bayes ve SVM algoritmalarının eğitimi yapılacak şekilde kodlama yapıldı.

4.4. Test

Her algoritma için eğitim işlemi gerçekleştirildikten sonra elde edilen sonuç %30'luk test verisi üzerinde test edildi. Elde edilen sonuçlar Seaborn kütüphanesiyle Karışıklık matrisi oluşturulup görselleştirildi. LSTM algoritmasıyla 10 adımda eğitilen veri seti için her bir adımda oluşan doğruluk değerleri grafiği Matplotlib kütüphanesiyle çizilmiştir.

5. ARAŞTIRMA BULGULARI VE TARTIŞMA

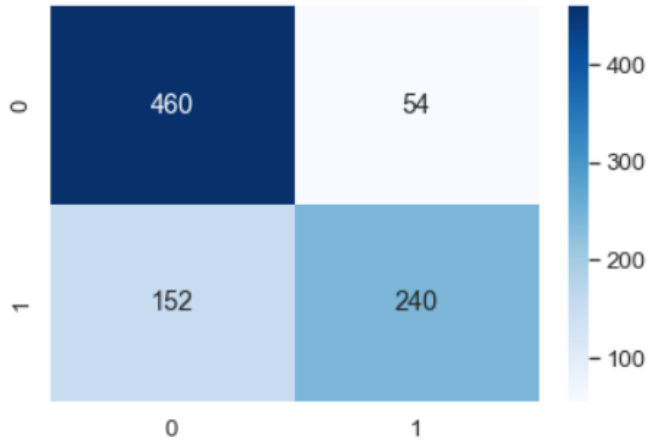
Literatür taramasında öne çıkan algoritmalar ile çalışmamızda elde edilen sonuçlar Çizelge 5.1'de Temel Yöntem sütununda her bir algoritma için gösterilmiştir.

Çizelge 5.1. Test Sonuçları

Algoritma	**Doğruluk**		
	Temel Yöntem	**Önerilen Yöntem**	
Lojistik Regresyon	%61,47	%77,26	%72,45
LSTM	%39,07	%65,89	%66,44
Multinominal Naive Bayes	%55,18	%69,20	%67,94
SVM	%62,36	%74,83	%71,45

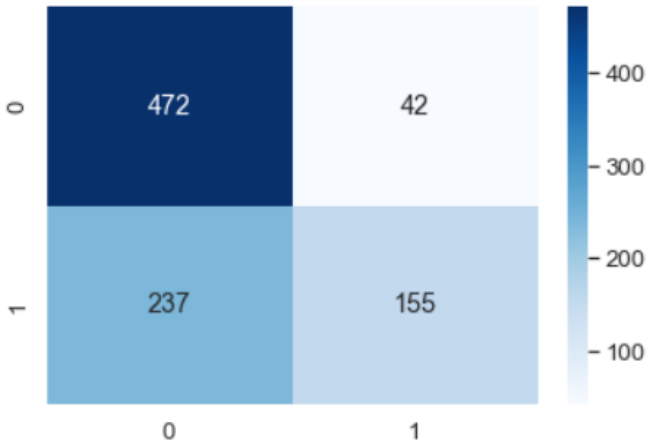
%39-%63 bandında doğruluk elde edilen bu sonuçlar, iki tahminden neredeyse birinin başarısız olduğu düşünüldüğünde başarısız olmuştur. Ancak Türkçenin sondan eklemeli bir dil olması, kelimelerin kök ve eklerinin ayrıştırılamaması nedeniyle bu başarı oranı düşük olmuş olabilir.

Yapılan çalışmalar sonucu Şekil 4.1.'deki gibi bir metodoloji geliştirildi. Problem, Talep, Soru sınıfında bankaların önem verdiği ve tazmin edilmesi muhtemel olan, finansal kayıp riskine neden olabilecek mesajlar genellikle bu sınıftaki veriler olduğundan bu verilerin ayrıştırılması esastır. Bu nedenle önerilen yaklaşım; bu üç gruba ait veri setini bir sınıf olarak, geriye kalan Şikâyet, Memnuniyet, Öneri sınıfına ait veriler ayrı bir sınıf olarak bölündükten sonra iki ayrı sınıfa bölünen tüm veri seti için ön işleme adımı gerçekleştirildi. Son olarak veri seti eğitim ve test aşamasından geçirildi. Lojistik regresyon algoritması kullanılarak test verisi üzerinde elde edilen sonuçlar Şekil 5.2'teki gibidir.



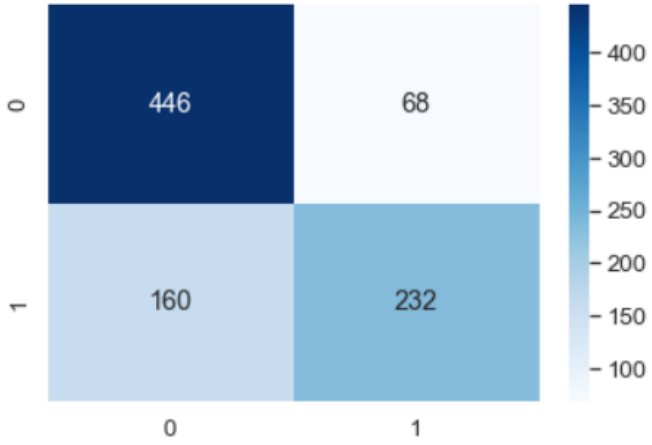
Şekil 5.2. İlk aşamada lojistik regresyon algoritmasıyla elde edilen karışıklık matrisi

Multinominal Naive Bayes algoritması ile test verileri üzerinde elde edilen sonuçlar Şekil 5.3.'te gösterilmektedir.



Şekil 5.3. İlk aşamada Multinominal Naive Bayes algoritmasıyla elde edilen karışıklık matrisi

SVM algoritması ile test verileri üzerinde elde edilen sonuçlar Şekil 5.4.'te gösterilmektedir.

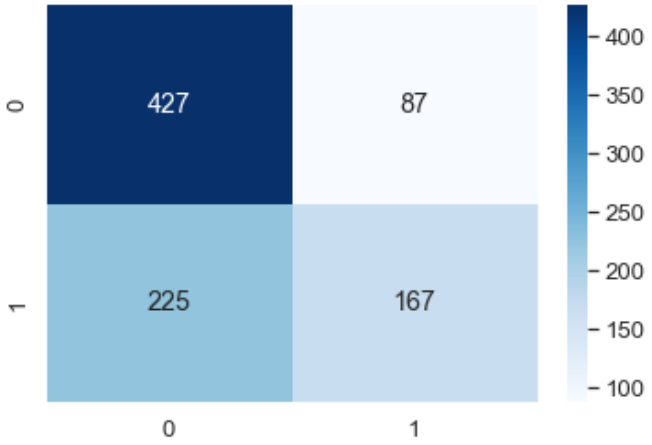


Şekil 5.4. İlk aşamada Svm algoritmasıyla elde edilen karışıklık matrisi

LSTM algoritmasını kullanan çalışmada, eğitim verisiyle 10 kez yapılan eğitim sonucunda %99,43 doğruluk elde edildi. Test verisiyle 10 kez(epoch) yapılan eğitim sonucunda elde edilen sonuçlar Şekil 5.5.'de, karmaşıklık matrisi Şekil 5.6'da gösterilmektedir.



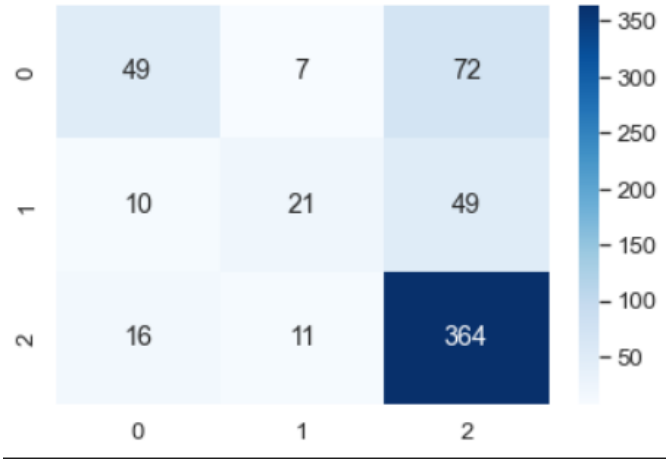
Şekil 5.5. İlk aşamada Lstm algoritmasıyla elde edilen doğruluk grafiği



Şekil 5.6. İlk aşamada Lstm algoritmasıyla elde edilen karmaşıklık matrisi

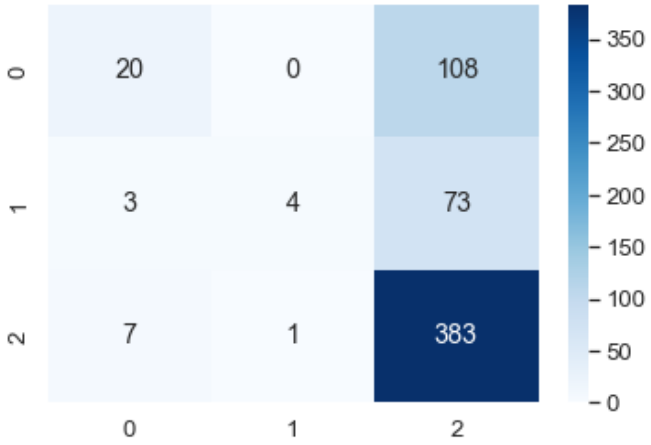
Bu adım her bir algoritma için uygulandıktan sonra ikinci aşamada Problem, Talep, Soru olarak etiketlenen satırlardan oluşan veri setinin %70'i eğitim ve %30 test verisi olarak ayrıştırılıp üç ayrı sınıftan oluşan bu veri seti için eğitim ve test adımları uygulandı.

Lojistik Regresyon algoritması ile elde edilen sonuçlar Şekil 5.7.'de gösterilmektedir.



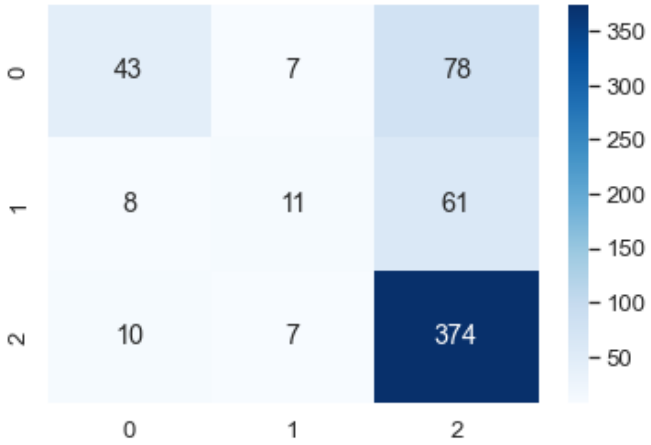
Şekil 5.7. İkinci aşamada lojistik regresyon algoritmasıyla elde edilen karışıklık matrisi

Multinomial Naive Bayes algoritması ile test verisi üzerinde elde edilen sonuçlar Şekil 5.8.'de gösterilmektedir.



Şekil 5.8. İkinci aşamada Multinominal Naive Bayes algoritmasıyla elde edilen karışıklık matrisi

SVM algoritması ile test verisi üzerinde elde edilen sonuçlar Şekil 5.9.'da gösterilmektedir.

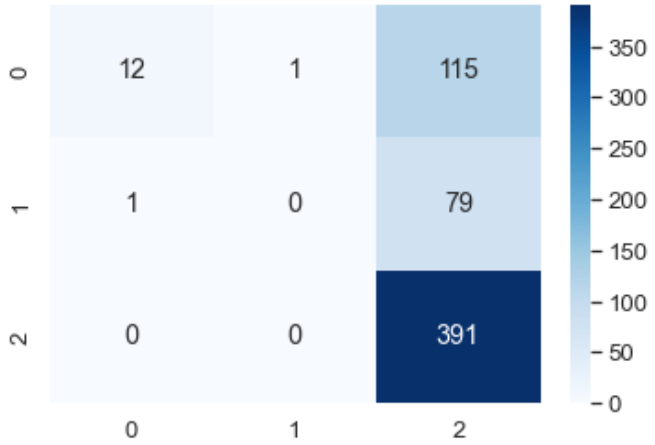


Şekil 5.9. İkinci aşamada Svm algoritmasıyla elde edilen karışıklık matrisi

LSTM algoritmasını kullanarak yaptığımız çalışmada, Eğitim setini 10 kez(epoch) eğitimden geçirilerek %80,21 doğruluk değeri ölçüldü. Test verilerinin eğitimi sonucunda elde edilen doğruluk değerleri Şekil 5.10.'da, karmaşıklık matrisi 5.11.'de gösterilmiştir.



Şekil 5.10. İkinci aşamada Lstm algoritmasıyla elde edilen doğruluk grafiği



Şekil 5.11. İkinci aşamada Lstm algoritmasıyla elde edilen karmaşıklık matrisi

Literatür taramasına göre metin sınıflandırmada başarılı sonuçlar elde edilmiş olan Lojistik Regresyon, Multinominal Naive Bayes, LSTM ve SVM algoritmaları kullanılarak gerçek bir banka veri seti üzerinde yapılan bu çalışmadan elde edilen sonuçlar Çizelge 5.1.'de gösterilmektedir. “Temel Yöntem” sütunu, uygun görülen sınıflandırma yöntemine göre Çizelge 5.1.'de “Algoritma" sütununda belirtilen algoritmalar uygulandıktan sonra test verisi üzerinde elde edilen sonuçları ifade eder.

Bu çalışmada geliştirilen yöntemle iki aşamalı yaklaşımla elde edilen test sonuçlarının, tek adımlı yaklaşımla elde edilen sonuçlara (Çizelge 5.1.’de “Temel Yöntem” sütununda gösterilen sonuçlar) kıyasla uygulamanın performansının yaklaşık yüzde

onun üzerinde artırdığı gözlemlendi. İki aşamalı yaklaşımda, her iki adımın doğruluk değerleri “Önerilen Yöntem” sütununda gösterilmiştir. Bu çalışmada sınıf sayısı artırılmış, sınıflandırma iki aşamalı yaklaşımla yapılmış, sınıflara öncelik verilerek müşteriye hızlı bir şekilde ulaşılması ve operatörlerin yapması gereken işlerin makineye aktarılması hedeflenmiştir.

6. SONUÇ VE ÖNERİLER

Günümüzde hızla gelişen ve değişen teknolojilere birlikte firmalar arasında teknolojik ürünler çıkarmada rekabet kızışmaktadır. En yeni teknolojileri ihtiyaca uygun olarak sistemine entegre eden kurumlar bu rekabet ortamında ön sıralarda yer almaktadır. Buradan hareketle bankacılık sektörünün de bu değişim ve gelişime en hızlı ve efektif şekilde ayak uydurması, bankanın gelişimini doğrudan etkilemektedir. Bunu yaparken müşteri bağlılığı odak noktası olarak ele alınmalıdır. Müşterilere hızlı dönüş yapılması ve müşteri memnuniyetini artırmayı sağlayacak çalışmalar bankaların müşteri kazanımını artıracaktır. Bu gibi kazanımların sağlanmasına yönelik olarak "DUYGU ANALİZİYLE MÜŞTERİ PROFİLİ OLUŞTURMA VE ŞİKAYET ÖNCELİKLENDİRME" çalışması yapılmıştır.

Bu çalışmada ilk olarak; banka müşterilerinin talep, istek ve sorularını öncelikli olarak sınıflandırılması ve başarılı şekilde ayrıştırılarak müşteriye geri bildirim verilmesi hedeflenmiştir. Aynı zamanda müşterilerin memnuniyetinin sağlanması, banka için olası finansal kayıp riskinin en aza indirilmesi ve bankaların temel kaynağı olan müşterilerin ilgili bankaya olan bağlılığının korunması hedeflenmiştir.

Yapılan çalışmanın sonuçları; Lojistik Regresyon, Multinominal Naive Bayes, LSTM ve SVM algoritmaları uygulanarak gerçekleştirilen bu çalışmada, iki aşamalı yaklaşımla elde edilen test sonuçlarının, literatür taramasında incelenmiş çalışmalardan yola çıkarak yapılan tek aşamalı yaklaşımın test sonuçlarından ortalama olarak yüzde onun üstünde daha başarılı olduğunu göstermektedir. Bu yaklaşımla sınıflandırma, doğruluğu artıracak şekilde iki adımda gerçekleştirilir. İlk adımda müşteri talep istek ve soru sınıfındaki veriler bir sınıf; müşteri memnuniyeti, bankaya gelen öneri ve müşteri şikayetleri sınıflarındaki veriler diğer ikinci bir sınıfa dahil edilerek iki ayrı sınıf için eğitim ve test adımları uygulandı. İkinci adımda ise bankanın asıl önem vermesi gerektiği düşünülen yani bahsi geçen ilk sınıftaki veriler sınıflandırılarak öncelikli dönüş yapılması gerektiği düşünülen müşteriye daha hızlı geri dönüş yapılması

sağlanmış olur. Veri seti bu şekilde iki adımlı eğitim ve test aşamasından geçirilerek doğruluk oranı arttırılmış olur.

Bu çalışma yapılırken bazı sorunlarla karşılaşılmıştır. Çalışmaya başlanılmadan önce yaşanan zorluk, araştırma ve geliştirmeye konu olacak veri setinin elde edilmesinin güç olmasıdır. Bunun sebebi kurumların, müşterilerinden gelen geri bildirim verilerinin kişisel veri içerebilmesi sebebiyle KVKK kanunu kapsamında veri işleyici olarak ilgili verinin müşteri rızası dışında üçüncü şahıslarla paylaşmaması gereklidir. Müşteri geri bildirimlerinden oluşan bu veri seti maskelenmiş yani müşterilerin kişisel verilerini içermeyecek şekilde paylaşılabilir fakat bunun için veri sahibi kurum tarafında maskeleme çalışma yapılması gerekir. Herhangi bir kişinin bu çalışmayı yapabilmesi için gerçek bir müşteri geri bildirim verisi elde etmesi bu nedenle mevcut koşullarda zordur. Ar-Ge faaliyetinde bulunan bir banka çalışanı olarak maskelenmiş bir veri seti elde etmek her ne kadar zorlayıcı olsa da çalışma banka bilgisayarında yapılması koşuluyla veri seti elde edilerek yapılabilmiştir.

Çalışma sırasında yaşanan zorluklara değinilmesi gerekirse; Türkçe'nin sondan eklemeli bir dil olması sebebiyle veri setindeki her bir veride kelimenin kökünün bulunup eklerinden ayrıştırılması yapılamamıştır. Ayrıca insan eliyle oluşturulmuş olan bu verilerin yazım yanlışları içermesi gibi nedenlerle sunulan yöntemin doğruluk oranı artırılmış olsa da çalışmanın başarı oranı daha fazla yükseltilememiştir. Yine de sınırlı sayıda veri seti ile çalışılarak bu sonuca ulaşılmıştır.

İleride yapılacak çalışmalarda her kelimenin kökü bulunup o kelimelerdeki ekleri kaldırarak daha fazla veri ile daha yüksek performans elde edilebilir. Kullanılan veri setindeki veride yazım yanlışları, imla hataları giderilebileceği bir yöntemle imla ve yazım hatası içermeyen bir veri seti elde edilebilir. Çalışmada kullanılan veri setindeki bazı sınıfların çok veriye, bazı sınıfların az veriye sahip olması nedeniyle; bu tür veri setlerinde daha yüksek doğrulukla çalışabilecek olan K En Yakın Komşu algoritması kullanılarak başarım oranı arttırılabilir. Buda sadece Literatür taramasına dayanarak Doğal Dil İşlemede başarılı bulunan algoritmaları değil, kullanılan veri setine uygun sınıflandırma algoritması seçilebileceğini gösterir. Veri seti, sınıflar kullanılarak KPI'lar

belirlenir ve müşteri memnuniyeti için elde edilen müşteri puanı kullanılabilir. Altı farklı sınıflandırmaya ek olarak, veri seti mevcut banka müşterilerini bankanın belirlediği herhangi bir standarda ve sahip olacakları müşteri puanına göre sıralanacak şekilde düzenlenebilir. Böylece banka, puanı yüksek olan müşterileri belirlediği standarda göre sıralayabilir ve bu sırayla müşterilere geri dönüş yapabilir. Bu öneri bankalar için avantajlı gibi görünse de banka müşterilerini belirli kurallara göre sıralayacağından düşük puanlı müşterileri memnun etmeyecektir.

KAYNAKLAR

Aryan, G., 2022, An Introduction to Scikit-Learn: Machine Learning in Python. Erişim Tarihi: 04.07.2022. https://www.simplilearn.com/tutorials/python-tutorial/scikit-learn

Bai, X., 2018.Text classification based on LSTM and attention. In 2018 Thirteenth International Conference on Digital Information Management (ICDIM) (pp. 29-32). IEEE.

Corpuz, R. S. A., 2021. An application method of long short-term memory neural network in classifying english and tagalog-based customer complaints, feedbacks, and commendations. International Journal on Information Technologies and Security, 13(1), 2021.

Corpuz, R. S. A., 2021. Categorizing natural language-based customer satisfaction: an implementation method using support vector machine and long short-term memory neural network. International Journal of Integrated Engineering, 13(4), 77-91.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine learning, 20(3), 273-297.

Çiğdem, A. C. I., Çırak, A., 2019. Türkçe haber metinlerinin konvolüsyonel sinir ağları ve Word2Vec kullanılarak sınıflandırılması. Bilişim Teknolojileri Dergisi, 12(3), 219-228.

Dobilas, S., 2022. LSTM Recurrent Neural Networks — How to Teach a Network to Remember the Past. Erişim Tarihi: 14.06.2022 https://towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e

Eser, Z., Pınar, M., Girard, T., 2016. Importance of Customer Complaints: A study of Banking Industry Utilising the Services Marketing and Branding Triangle Framework. Journal of Theory and Practice in Marketing, 2(2), 23-49.

Filgueiras, J., Barbosa, L., Rocha, G., Cardoso, H. L., Reis, L. P., Machado, J. P., Oliveira, A. M., 2019. Complaint analysis and classification for economic and food safety. In Proceedings of the Second Workshop on Economics and Natural Language Processing (pp. 51-60).

Galitsky, B. A., González, M. P., Chesñevar, C. I., 2009. A novel approach for classifying customer complaints through graphs similarities in argumentative dialogues. Decision Support Systems, 46(3), 717-729.

Gavval, R., Ravi, V., Harshal, K. R., Gangwar, A., Ravi, K., 2019. CUDA-Self-Organizing feature map based visual sentiment analysis of bank customer complaints for Analytical CRM. arXiv preprint arXiv:1905.09598.

Gazitepe, H., 2022. KPI Nedir? Nasıl Hesaplanır? Belirleyici Metrikler Nelerdir? Erişim Tarihi: 09.08.2022 https://digitalakademi.net/kpi-nedir-nasil-hesaplanir/

Gong, S., Dai, Y., Ji, J., Wang, J., Sun, H., 2015. Emotion analysis of telephone complaints from customer based on affective computing. Computational intelligence and neuroscience, 2015.

Great Learning Team, 2022. Multinomial Naive Bayes Explained. Erişim Tarihi: 17.06.2022. https://www.mygreatlearning.com/blog/multinomial-naive-bayes-explained/

Gupta, D., Lenka, P., Bedi, H., Ekbal, A., Bhattacharyya, P., 2017. Auto Analysis of Customer Feedback using CNN and GRU Network. arXiv preprint arXiv:1710.04600.

Hakiri, W., 2012. For an efficient complaints management system for banks: A conceptual framework and an exploratory study. Journal of Marketing Research & Case Studies, 2012, 1.

Harris, Z. S., 1954. Distributional Structure, WORD, vol. 10, no. 2–3.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), 1735-1780.

Hosmer Jr, D. W., Lemeshow, S., Sturdivant, R. X., 2013. *Applied logistic regression* (Vol. 398). John Wiley & Sons.

Ibrahim, A., Pratiwi, A., Meytri, D. I., Kurniawan, M. A., Yuniarti, N., 2018. Measuring Customer Satisfaction Using CRM Scorecard in Canteen FASILKOM UNSRI. In 2018 International Conference on Electrical Engineering and Computer Science (ICECOS) (pp. 403-408). IEEE.

Javatpoint, 2022. Logistic Regression in Machine Learning. Erişim Tarihi: 12.06.2022. https://www.javatpoint.com/logistic-regression-in-machine-learning

Joung, J., Jung, K., Ko, S., Kim, K., 2018. Customer complaints analysis using text mining and outcome-driven innovation method for market-oriented product development. Sustainability, 11(1), 40.

Kumar, A., Dhanalakshmi, V., Rekha, R. U., Soman, K. P., Rajendran, S., 2009. Morphological analyzer for agglutinative languages using machine learning approaches. In 2009 International Conference on Advances in Recent Technologies in Communication and Computing (pp. 433-435). IEEE.

Kumar, P., Mokha, A. K., Pattnaik, S. C., 2021. Electronic customer relationship management (E-CRM), customer experience and customer satisfaction: evidence from the banking industry. Benchmarking: An International Journal, 29(2), 551-572.

Kintz, M., Dukino, C., Blohm, M., Hanussek, M., 2020. Make Your Customers Happy Again: AI and NLP for a Customer Complaint Management Platform.

Laksono, R. A., Sungkono, K. R., Sarno, R., Wahyuni, C. S., 2019. Sentiment analysis of restaurant customer reviews on TripAdvisor using Naïve Bayes. In 2019 12th International Conference on Information & Communication Technology and System (ICTS) (pp. 49-54). IEEE.

Liu, Z., Lv, X., Liu, K., Shi, S., 2010. Study on SVM compared with the other text classification methods. In 2010 Second international workshop on education technology and computer science (Vol. 1, pp. 219-222). IEEE.

Lu, X., Yang, Y., Qin, H., 2010. The Research on Personal Internet Banking Service Quality and Customer Loyalty Based on Complaints Handling. In 2010 International Conference on Management and Service Science (pp. 1-4). IEEE.

Natural Language Toolkit, 2022. NLTK ToolKit. Erişim Tarihi: 01.03.2022. https://www.nltk.org/

Ndubisi, N. O., Wah, C. K., Ndubisi, G. C., 2007. Supplier-customer relationship management and customer loyalty: The banking industry perspective. Journal of Enterprise Information Management.

Oral, B., Emekligil, E., Arslan, S., Eryiğit, G., 2019. Extracting complex relations from banking documents. In Proceedings of the Second Workshop on Economics and Natural Language Processing (pp. 1-9).

Python, 2022. What is Python? Executive Summary. Erişim Tarihi: 10.07.2022. https://www.python.org/doc/essays/blurb/

Python Software Foundation, 2022. Matplotlib, Erişim Tarihi: 10.07.2022 https://pypi.org/project/matplotlib/

Python Software Foundation, 2022. Pandas, Erişim Tarihi: 09.07.2022. https://pypi.org/project/pandas/

Qader, W. A., Ameen, M. M., Ahmed, B. I., 2019. An overview of bag of words; importance, implementation, applications, and challenges. In 2019 International Engineering Conference (IEC) (pp. 200-204). IEEE.

Saishruthi, S., 2018, Logistic Regression — Detailed Overview. Erişim Tarihi: 12.06.2022. https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc

Salim, A., Setiawan, M., Rofiaty, R., Rohman, F., 2018. Focusing on complaints handling for customer satisfaction and loyalty: The case of Indonesian public banking. European Research Studies, 21(3), 404-416.

Sestek, 2022. Konuşma ve Metin Analiz Uygulaması, Erişim Tarihi: 10.06.2022. https://www.sestek.com/tr/konusma-ve-metin-analizi/

Setyawan, M. Y. H., Awangga, R. M., Efendi, S. R., 2018. Comparison of multinomial naive bayes algorithm and logistic regression for intent classification in chatbot. In 2018 International Conference on Applied Engineering (ICAE) (pp. 1-5). IEEE.

Singh, G., Kumar, B., Gaur, L., Tyagi, A., 2019. Comparison between multinomial and Bernoulli naïve Bayes for text classification. In 2019 International Conference on Automation, Computational and Technology Management (ICACTM) (pp. 593-596). IEEE.

Singh, H., 2006. The importance of customer satisfaction in relation to customer loyalty and retention. Academy of Marketing Science, 60(193-225), 46.

Sunil, R., 2017. Understanding Support Vector Machine(SVM) algorithm from examples (along with code). Erişim Tarihi: 01.07.2022. https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

Tensorflow, 2022. Tensorflow overview. Erişim Tarihi: 04.03.2022. https://www.tensorflow.org/overview

Ticimax Bilişim Teknolojileri A.Ş, 2022. KEY Performance Indicator (KPI) Nedir? Nasıl Ölçülür? Ticimax Bilgi Bankası. Erişim Tarihi: 10.08.2022. https://www.ticimax.com/blog/key-performance-indicator-kpi-nedir

Waskom, M., 2022. Seaborn, Erişim Tarihi: 05.07.2022. https://seaborn.pydata.org/

Yıldırım, E., Çetin, F. S., Eryiğit, G., Temel, T., 2015. The impact of NLP on Turkish sentiment analysis. Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, 7(1), 43-51.

EKLER

EK A. Fotoğraflar

EK A. Fotoğraflar

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g.
pd.read_csv)
import re # regular expression libary.
import nltk # Natural Language toolkit
nltk.download("stopwords")  #downloading stopwords
nltk.download('punkt')
from nltk import word_tokenize,sent_tokenize
nltk.download('wordnet')
import nltk as nlp
from sklearn.preprocessing import LabelEncoder
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from sklearn.feature_extraction.text import CountVectorizer
from wordcloud import WordCloud
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import
confusion_matrix,accuracy_score,classification_report,precision_scor
e
from sklearn import preprocessing
from sklearn.feature_extraction.text import CountVectorizer #Bag of
Words
import seaborn as sns
import collections
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
file_name="route"
df = pd.read_excel(file_name, sheet_name=None)
df_copy=df.copy()
df_copy["YL"].head()

Encoder = preprocessing.LabelEncoder()
y_train=Encoder.fit_transform(df_copy["YL"][:2114].categoryText)
y_test=Encoder.fit_transform(df_copy["YL"][2114:].categoryText)

import seaborn as sns
import matplotlib.pyplot as plt
data =df_copy["YL"].Category.value_counts()
sns.barplot(y = data.index, x = data, orient='h')

from nltk.corpus import stopwords
nltk.download("stopwords")
nltk.download('omw-1.4')
def clearText(textList):
    text_list=[]

    for text in textList:
        text = text.lower()  #Büyük harften -Küçük harfe çevirme
        text = re.sub("[^abcçdefgğhıijklmnoöprsştuüvyz]"," ",text)
        text=nlp.word_tokenize(text) # splits the words that are in
the sentence from each other.
```

```
        text =[word for word in text if not word in
set(stopwords.words("turkish"))]
        lemma=nlp.WordNetLemmatizer()
        text=[lemma.lemmatize(word) for word in text] # this code
finds the root of the word for a word in the sentence and change
them to their root form.
        text=" ".join(text)
        text_list.append(text) # store sentences in list

    return text_list

text_list=clearText(df_copy["YL"].text)
train_text_list=clearText(text_list[:2114])
test_text_list=clearText(text_list[2114:])

max_features=500 # "number" most common(used) words in reviews
count_vectorizer=CountVectorizer(max_features=max_features)
X_train=count_vectorizer.fit_transform(train_text_list).toarray()
X_test=count_vectorizer.fit_transform(test_text_list).toarray()

X_train=train_text_list
X_test=test_text_list
```

Şekil A.1. Veri setinin temizlendiği eğitim ve test verisinin ayrıldığı kod bloğu

```
import tensorflow as tf
max_lenght=100

tokenizer = Tokenizer()
tokenizer.fit_on_texts(X_train)
word_index = tokenizer.word_index # creating word dict for words in
training

sequences = tokenizer.texts_to_sequences(X_train)  # replacing words
with the number corresponding to them in the dictionary(word_index)

X_train_padded = pad_sequences(sequences,
padding='post',maxlen=max_lenght) # padding words

print(len(word_index)) # I have 94836 words in my dictionary
X_test_sequences = tokenizer.texts_to_sequences(X_test)
X_test_padded =
pad_sequences(X_test_sequences,padding="post",maxlen=max_lenght)
vocab_size = len(tokenizer.word_index)+1
embedding_dim=16
model = tf.keras.Sequential([

    tf.keras.layers.Embedding(vocab_size, embedding_dim,
input_length=100),

    tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(64)),
    tf.keras.layers.Dropout(0.3),
```

```
    tf.keras.layers.BatchNormalization(),

    tf.keras.layers.Flatten(),

    tf.keras.layers.Dense(64, activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dropout(0.2),

    tf.keras.layers.Dense(6, activation='softmax')
])

adam= tf.keras.optimizers.Adam(lr=0.01)


model.compile(loss='sparse_categorical_crossentropy',optimizer=adam,
metrics=['accuracy'])
model.summary()

y_train_label=np.asarray(y_train).reshape(-1,1)
y_test_label=np.asarray(y_test).reshape(-1,1)

num_epochs = 10
BATCH_SIZE=64
history=model.fit(X_train_padded,y_train_label,batch_size=BATCH_SIZE
,epochs=num_epochs, validation_data=(X_test_padded,y_test_label))

plt.plot(history.history["accuracy"],color="green")
plt.plot(history.history["loss"],color="red")
plt.title("Train accuracy and Train loss")
plt.legend(["Accuracy","Loss"])
plt.grid()
plt.savefig('lstmtektrain.eps', format='eps')
plt.plot(history.history["val_accuracy"],color="blue")
plt.plot(history.history["val_loss"],color="orange")
plt.title("Test accuracy and Test loss")
plt.legend(["Val_accuracy","Val_loss"])
plt.grid()
plt.savefig('lsttek.eps', format='eps')

print("Accuracy of the model on Training Data is - " ,
model.evaluate(X_train_padded,y_train_label)[1]*100 , "%")
print("Accuracy of the model on Testing Data is - " ,
model.evaluate(X_test_padded,y_test_label)[1]*100 , "%")
```

Şekil A.2. 6 Ayrı sınıfa ayrılmış verinin Lstm algoritmasıyla eğitim ve test aşamasının kodlaması

```
import tensorflow as tf
max_lenght=100
tokenizer = Tokenizer()
tokenizer.fit_on_texts(X_train)
```

```
word_index = tokenizer.word_index # creating word dict for words in
training

sequences = tokenizer.texts_to_sequences(X_train)  # replacing words
with the number corresponding to them in the dictionary(word_index)

X_train_padded = pad_sequences(sequences,
padding='post',maxlen=max_lenght) # padding words

print(len(word_index)) # I have 94836 words in my dictionary
X_test_sequences = tokenizer.texts_to_sequences(X_test)
X_test_padded =
pad_sequences(X_test_sequences,padding="post",maxlen=max_lenght)
vocab_size = len(tokenizer.word_index)+1
embedding_dim=16

model = tf.keras.Sequential([

    tf.keras.layers.Embedding(vocab_size, embedding_dim,
input_length=100),

    tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(64)),
    tf.keras.layers.Dropout(0.3),
    tf.keras.layers.BatchNormalization(),

    tf.keras.layers.Flatten(),

    tf.keras.layers.Dense(64, activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dropout(0.2),

    tf.keras.layers.Dense(2, activation='softmax')
])

adam= tf.keras.optimizers.Adam(lr=0.01)


model.compile(loss='sparse_categorical_crossentropy',optimizer=adam,
metrics=['accuracy'])
model.summary()
y_train_label=np.asarray(y_train).reshape(-1,1)
y_test_label=np.asarray(y_test).reshape(-1,1)
num_epochs = 10
BATCH_SIZE=64
history=model.fit(X_train_padded,y_train_label,batch_size=BATCH_SIZE
,epochs=num_epochs, validation_data=(X_test_padded,y_test_label))
plt.plot(history.history["accuracy"],color="green")
plt.plot(history.history["loss"],color="red")
plt.title("Train accuracy and Train loss")
plt.legend(["Accuracy","Loss"])
plt.grid()
plt.savefig('lstmtektrain.eps', format='eps')
plt.plot(history.history["val_accuracy"],color="blue")
plt.plot(history.history["val_loss"],color="orange")
plt.title("Test accuracy and Test loss")
plt.legend(["Val_accuracy","Val_loss"])
plt.grid()
```

```
plt.savefig('lsttek.eps', format='eps')
print("Accuracy of the model on Training Data is - " ,
model.evaluate(X_train_padded,y_train_label)[1]*100 , "%")
print("Accuracy of the model on Testing Data is - " ,
model.evaluate(X_test_padded,y_test_label)[1]*100 , "%")
```

Şekil A.3. İlk adımda iki ayrı sınıfın Lstm algoritmasıyla eğitim ve test aşamasının kodlaması

```
import tensorflow as tf
max_lenght=100
tokenizer = Tokenizer()
tokenizer.fit_on_texts(X_train)

word_index = tokenizer.word_index # creating word dict for words in
training

sequences = tokenizer.texts_to_sequences(X_train)  # replacing words
with the number corresponding to them in the dictionary(word_index)

X_train_padded = pad_sequences(sequences,
padding='post',maxlen=max_lenght) # padding words

print(len(word_index)) # I have 94836 words in my dictionary
X_test_sequences = tokenizer.texts_to_sequences(X_test)
X_test_padded =
pad_sequences(X_test_sequences,padding="post",maxlen=max_lenght)
vocab_size = len(tokenizer.word_index)+1
embedding_dim=16

model = tf.keras.Sequential([

    tf.keras.layers.Embedding(vocab_size, embedding_dim,
input_length=100),

    tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(64)),
    tf.keras.layers.Dropout(0.3),
    tf.keras.layers.BatchNormalization(),

    tf.keras.layers.Flatten(),

    tf.keras.layers.Dense(64, activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dropout(0.2),

    tf.keras.layers.Dense(3, activation='softmax')
])

adam= tf.keras.optimizers.Adam(lr=0.01)


model.compile(loss='sparse_categorical_crossentropy',optimizer=adam,
metrics=['accuracy'])
```

```
model.summary()
y_train_label=np.asarray(y_train).reshape(-1,1)
y_test_label=np.asarray(y_test).reshape(-1,1)
#y_train_label = np.asarray(y_train).reshape((-1,1))
#y_test_label = np.asarray(y_test).reshape((-1,1))
num_epochs = 10
BATCH_SIZE=64
history=model.fit(X_train_padded,y_train_label,batch_size=BATCH_SIZE
,epochs=num_epochs, validation_data=(X_test_padded,y_test_label))
plt.plot(history.history["accuracy"],color="green")
plt.plot(history.history["loss"],color="red")
plt.title("Train accuracy and Train loss")
plt.legend(["Accuracy","Loss"])
plt.grid()
plt.savefig('lstmtektrain.eps', format='eps')
plt.plot(history.history["val_accuracy"],color="blue")
plt.plot(history.history["val_loss"],color="orange")
plt.title("Test accuracy and Test loss")
plt.legend(["Val_accuracy","Val_loss"])
plt.grid()
plt.savefig('lsttek.eps', format='eps')
print("Accuracy of the model on Training Data is - " ,
model.evaluate(X_train_padded,y_train_label)[1]*100 , "%")
print("Accuracy of the model on Testing Data is - " ,
model.evaluate(X_test_padded,y_test_label)[1]*100 , "%")
```

Şekil A.4. İkinci adımda üç ayrı sınıfın Lstm algoritmasıyla eğitim ve test aşamasının kodlaması

```
from sklearn import naive_bayes

Train_X=train_text_list
Test_X=test_text_list
Test_Y=test_y
Train_Y=train_y

Tfidf_vect = TfidfVectorizer(max_features=500)
Tfidf_vect.fit(text_list)

Train_X_Tfidf = Tfidf_vect.transform(Train_X)
Test_X_Tfidf = Tfidf_vect.transform(Test_X)

Naive = naive_bayes.MultinomialNB()
Naive.fit(Train_X_Tfidf,Train_Y)

# predict the labels on validation dataset
predictions_NB = Naive.predict(Test_X_Tfidf)

# #Use accuracy_score function to get the accuracy
print("Naive Bayes Accuracy Score ->
",accuracy_score(predictions_NB, Test_Y)*100)

naiveb=confusion_matrix(test_y,predictions_NB)
```

```
plt.figure(figsize=(10,10))
sns.heatmap(confusion_matrix(test_y,predictions_NB),annot=True,fmt='
d')
plt.savefig('NaiveBayestek.eps', format='eps')
```

Şekil A.5. Multinominal Naive Bayes algoritmasının kodlaması

```
# Classifier - Algorithm - SVM
# fit the training dataset on the classifier
SVM = svm.SVC(C=1.0, kernel='linear', degree=3, gamma='auto')
SVM.fit(Train_X_Tfidf,Train_Y)

# predict the labels on validation dataset
predictions_SVM = SVM.predict(Test_X_Tfidf)

# Use accuracy_score function to get the accuracy
print("SVM Accuracy Score -> ",accuracy_score(predictions_SVM,
Test_Y)*100)

plt.figure(figsize=(10,10))
sns.heatmap(confusion_matrix(Test_Y,predictions_SVM),annot=True,fmt=
'd')
plt.savefig('Svmtek.eps', format='eps')
```

Şekil A.6 Svm algoritmasının kodlaması

```
from sklearn.feature_extraction.text import CountVectorizer
sentences_train=Train_X
sentences_test=Test_X
y_train=train_y
y_test=test_y
vectorizer = CountVectorizer()
vectorizer.fit(text_list)
X_train = vectorizer.transform(sentences_train)
X_test  = vectorizer.transform(sentences_test)
from sklearn.linear_model import LogisticRegression

classifier = LogisticRegression()
classifier.fit(X_train, y_train)
score = classifier.score(X_test, y_test)
predictions_LR = classifier.predict(X_test)
print("Accuracy:", score)
plt.figure(figsize=(10,10))
sns.heatmap(confusion_matrix(y_test,predictions_LR),annot=True,fmt='
d')
plt.savefig('LogisticRegtek.eps', format='eps')
```

Şekil A.7 Lojistik regresyon algoritmasının kodlaması

ÖZGEÇMİŞ

Adı Soyadı : Muhammed Mehmet AKGÜMÜŞ

Eğitim Durumu

Lise : Haydarpaşa Anadolu Teknik Lisesi, 2013

Lisans : Sakarya Üniversitesi, Bilgisayar Ve Bilişim Bilimleri Fakültesi, Bilgisayar Mühendisliği Bölümü

Yüksek Lisans : İstanbul Ticaret Üniversitesi,
Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı

Mesleki Deneyim

Kuşman Bilgisayar Yazılım	2018-2020
Türkiye Finans Katılım Bankası	2020-2021
Innovance Bilgi Teknolojileri A.Ş	2021-(Devam Ediyor)

Yayınları

Akgümüş, M. M., Boyacı, A., 2023. Bankacılık Sektörü İçin Topluluk Öğrenimini Kullanan İki Aşamalı Bir Müşteri Şikâyet Yönetimi, Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi.