
GAZİ ÜNİVERSİTESİ
Gazi
1926

# OPTİMİZASYON TEMELLİ ÖZNİTELİK SEÇME YÖNTEMLERİ İLE DESTEKLENEN TOPLULUK ÖĞRENME YAKLAŞIMINA DAYALI YAZAR TANIMA

**Merve GÜLLÜ**

**YÜKSEK LİSANS**

**BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ**

**FEN BİLİMLERİ ENSTİTÜSÜ**

**NİSAN 2021**

## ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Merve GÜLLÜ

22/04/2021

# OPTİMİZASYON TEMELLİ ÖZNİTELİK SEÇME YÖNTEMLERİ İLE DESTEKLENEN TOPLULUK ÖĞRENME YAKLAŞIMINA DAYALI YAZAR TANIMA

(Yüksek Lisans Tezi)

Merve GÜLLÜ

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Nisan 2021

## ÖZET

İnternet ve özellikle sosyal medya aracılığıyla veri arama, kopyalama ve yayma fırsatlarının artması doğru bilgiye ulaşımı azaltmıştır. Veriden doğru bilgiye ulaşma konusunda metin madenciliği alanında yapılan çalışmalardan biri metin yazarı tahminidir. Bir metin, onu yazan kişinin karakteristik özelliklerini taşır ve bu özellikler metnin yazarını tanımlamak için kullanılabilir. Bu çalışmada 54 adet yazar, değişken sayıda ve değişken uzunlukta toplam 46.837 adet köşe yazısı ile bir derlem oluşturulmuştur. Yazarların karakteristik özelliklerini çıkarmak için iki farklı analiz ve iki analizin birleştirilmesi ile oluşturulmuş karma analiz hazırlanmıştır. Analiz sonuçlarının verimliliğini artırmaya yönelik Rastgele Orman Algoritması ile Genetik Algoritma ve Tavuk Sürü Optimizasyon Algoritması yaklaşımlarına dayalı toplamda iki farklı öznitelik seçim yöntemi sunuldu. Yapılan analizler sonucunda en verimli yazar tahmini çözüm önerisi, Topluluk Öğrenimi Algoritmaları ile desteklendi. En iyi performans, Karma Analiz ve sonrasında Genetik Optimizasyon algoritması ile gerçekleştirilen işlemler sonucu oluşturulan on yazarlı veri kümesi üzerinde Torbalama Algoritmasında sınıflandırıcı metot olarak Karar Ağacı kullanımında % 95,74 olarak elde edilmiştir.

Bilim Kodu : 92432
Anahtar Kelimeler : Yazar Tanıma, Genetik Algoritma, Tavuk Sürü Optimizasyon Algoritması, Topluluk Öğrenimi
Sayfa Adedi : 68
Danışman : Doç. Dr. Hüseyin POLAT

# AUTHOR IDENTIFICATION BASED ON THE ENSEMBLE LEARNING APPROACH SUPPORTED BY OPTIMIZATION-BASED FEATURE SELECTION METHODS

(M.Sc. Thesis)

Merve GÜLLÜ

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

April 2021

## ABSTRACT

Accessing accurate information has been reduced by increasing opportunities for searching, copying and disseminating data via the Internet and especially social media. One of the studies conducted in the field of text mining in obtaining correct information from data is the author identification. A text carries the characteristics of the person who wrote it, and these properties can be used to identify the author of the text. In our study, a collection contained 54 authors, and a total of 46 837 texts with different numbers and different sizes, was used. A mixed analysis was prepared by combining two different analyzes and two analyzes to reveal the characteristics of the author. In order to increase the efficiency of the analysis results two different feature selection methods were offered, based on the combination of Random Forest Algorithm with Genetic Algorithm and Chicken Swarm Optimization Algorithm. As a result of the analysis, the most efficient author identification solution was supported with Ensemble Learning Algorithms. The best performance was achieved as 95.74% in the use of Decision Tree as a classifying method in Bagging Algorithm on a data set with ten authors, which was created as a result of operations performed with the Mixed Analysis and afterwards with the Genetic Optimization algorithm.

| | | |
|---|---|---|
| Science Code | : | 92432 |
| Key Words | : | Author identification, Genetic Algorithm, Chicken Swarm Optimization, Ensemble Learning Algorithm |
| Page Number | : | 68 |
| Supervisor | : | Assoc. Prof. Dr. Hüseyin POLAT |

# TEŞEKKÜR

Yüksek lisans eğitimim boyunca çalışmamda konu, kaynak ve yöntem açısından yol gösteren, danışmanlığımı yapan değerli hocam Doç. Dr. Hüseyin POLAT'a, mensubu olduğum Gazi Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Anabilim Dalı'nda görev yapan değerli hocalarıma, bu süreçte yanımda olan ve desteğini hiç esirgemeyen aileme sonsuz teşekkürlerimi sunuyorum.

# İÇİNDEKİLER

**Sayfa**

ÖZET ..... iv

ABSTRACT ..... v

TEŞEKKÜR ..... vi

ÇİZELGELERİN LİSTESİ ..... ix

ŞEKİLLERİN LİSTESİ ..... xi

SİMGELER VE KISALTMALAR ..... xiii

1. GİRİŞ ..... 1

2. LİTERATÜR TARAMASI ..... 5

3. MATERYAL VE METOT ..... 11

3.1. Derlem (Corpus) ..... 12

3.2. Öznitelik Çıkarımı ..... 13

3.3. Öznitelik Seçme Yöntemleri ..... 18

3.3.1. Genetik Algoritma ..... 18

3.3.2. Tavuk Sürü Optimizasyon Algoritması ..... 21

3.4. Makine Öğrenmesi ..... 24

3.4.1. Topluluk öğrenimi ..... 25

3.4.2. Rastgele Orman Algoritması ..... 27

3.4.3. K En Yakın Komşu ..... 28

3.4.4. Karar Ağacı ..... 28

3.4.5. Çok Terimli Naive Bayes ..... 29

3.4.6. Adaboost ..... 29

3.4.7. XGBoost ..... 30

3.4.8. LightGBM ..... 30

**Sayfa**

3.5. Performans Değerlendirme Ölçütleri ..... 31

3.6. Geliştirme Ortamı ..... 32

4. METODOLOJİ VE DENEYSEL SONUÇLAR ..... 33

5. SONUÇ VE ÖNERİLER ..... 57

KAYNAKLAR ..... 61

ÖZGEÇMİŞ ..... 67

# ÇİZELGELERİN LİSTESİ

**Çizelge** **Sayfa**

Çizelge 3.1. Çalışmada kullanılan sözcüksel, yapısal ve sözdizilimsel özniteliklerin listesi ..... 16

Çizelge 3.1. (devam) Çalışmada kullanılan sözcüksel, yapısal ve sözdizilimsel özniteliklerin listesi ..... 16

Çizelge 3.2. Hata Matrisi ..... 32

Çizelge 4.1. En uygun jeton değerini belirlemek için n değerinin 2 olarak sabit tutulduğu, farklı sayıda yazar ve farklı sayıda metinler ile hazırlanan senaryolarda 6 farklı jeton değeri için ROA kullanılarak hazırlanan modellerin ortalama doğruluk oranları (%)..... 35

Çizelge 4.2. n-gram tekniğinde en uygun n değerinin belirlenmesi için jeton sayısının 300 olarak kabul edildiği ve yazar sayısı, her yazar için kullanılacak metin sayısı, n değerlerinin farklı değerleri ile hazırlanan tüm senaryolar için ROA kullanılarak oluşturulan modellerinin doğruluk oranları (%) ..... 37

Çizelge 4.3. Yazar sayısı ve metin sayısının farklı kombinasyonları ile hazırlanmış toplam 18 durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulmuş verisetlerinin ROA ile gerçekleştirilen modellerin doğruluk oranları (%).. ..... 44

Çizelge 4.4. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz1 ile oluşturulan veri kümesi üzerinde TSOA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranları, model oluşturma süreleri ve 300 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri gösterilmiştir ..... 46

Çizelge 4.5. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz2 ile oluşturulan veri kümesi üzerinde TSOA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranları, model oluşturma süreleri ve 40 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri gösterilmiştir ..... 47

Çizelge 4.6. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Karma Analiz ile oluşturulan veri kümesi üzerinde TSOA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranları, model oluşturma süreleri ve 340 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri gösterilmiştir ..... 48

Çizelge 4.7. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz1 ile oluşturulan veri kümesi üzerinde GA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranı, model oluşturma süreleri ve 300 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri ..... 52

**Çizelge** **Sayfa**

Çizelge 4.8. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz2 ile oluşturulan veri kümesi üzerinde GA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranı, model oluşturma süreleri ve 40 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri ..... 53

Çizelge 4.9. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Karma Analiz ile oluşturulan veri kümesi üzerinde GA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranı, model oluşturma süreleri ve 340 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri..... 54

Çizelge 4.10. Torbalama Yöntemi ile yazar sayısının 10 metin sayısının 2000 olduğu (Karma Analiz kullanılarak özniteliklerin çıkarılıp GA ile öznitelik seçim işlemi uygulandıktan sonra oluşturulan) veri kümesi üzerinde gerçekleştirilen sınıflandırma sonucu başarım metrikleri..... 55

Çizelge 4.11. Artırma Yöntemi ile yazar sayısının 10 metin sayısının 2000 olduğu (Karma Analiz kullanılarak özniteliklerin çıkarılıp GA ile öznitelik seçim işlemi uygulandıktan sonra oluşturulan) veri kümesi üzerinde gerçekleştirilen sınıflandırma sonucu başarım metrikleri..... 56

Çizelge 5.1. Yazar sayısının 10 olduğu ve metin sayısının 5 farklı boyutta (100, 200, 500, 1000 ve 2000) olduğu durumlar için öznitelik seçim işlemi öncesi ve sonrası ROA ile oluşturulan modellerin 5 durum için ortalama doğruluk oranları..... 58

Çizelge 5.2. Bu çalışmada önerilen yöntem ve diğer Türkçe derlemler ile yapılan çalışmaların karşılaştırılması..... 59

# ŞEKİLLERİN LİSTESİ

**Şekil** **Sayfa**

Şekil 3.1. Derlem oluşturma süreçleri ..... 13

Şekil 3.2. Çalışmada oluşturulan veri setlerinin GA ile öznitelik seçim işleminin aşamaları ..... 19

Şekil 3.3. TSOA ile gerçekleştirilen öznitelik seçim işlemi akış çizelgesi ..... 22

Şekil 3.4. Torbalama tekniğinin çalışma prensibi ..... 26

Şekil 3.5. Artırma tekniğinin çalışma prensibi ..... 27

Şekil 4.1. Analiz1-n-gram tekniğinde izlenen süreç ..... 34

Şekil 4.2. Yazar sayısının 5 olduğu, her yazara ait 3000 metin ve 6 farklı jeton sayıları ile oluşturulan senaryolarda model geliştirme süreleri ..... 36

Şekil 4.3. En uygun n değeri hesaplamasında oluşturulan her yazara ait 200 adet metnin kullanıldığı, farklı yazar ve farklı n değeri ile oluşturulan modellerin, sınıflandırma doğruluğunun yazar sayısına göre değişimini ..... 38

Şekil 4.4. En uygun n değeri hesaplamak için oluşturulan modellerin yazar sayısı 5 olarak sabit tutulduğunda metin sayıları değişiminin ve değişen n değerlerinin doğruluk oranı üzerindeki etkisi ..... 39

Şekil 4.5. Farklı yazar ve metin sayıları ile oluşturulan 18 senaryo üzerinde en yüksek sınıflandırma doğruluğunun n değeri üzerindeki dağılımları ..... 40

Şekil 4.6. Yazar sayısı 5 ve 10 olduğu durumda sınıflandırmada kullanılacak metin sayısı değişiminin ve farklı jeton değerlerinin sınıflandırma doğruluğu üzerindeki etkisi ..... 41

Şekil 4.7. Yazar tahmini problemi çözümünde Analiz2, Analiz1 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde gerçekleştirilecek işlemlerde izlenecek süreç ..... 42

Şekil 4.8. Yazar sayısının 10 ve metin sayısının 100 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri ..... 49

Şekil 4.9. Yazar sayısının 10 ve metin sayısının 200 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri ..... 49

**Şekil** **Sayfa**

Şekil 4.10. Yazar sayısının 10 ve metin sayısının 500 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri..... 50

Şekil 4.11. Yazar sayısının 10 ve metin sayısının 1000 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri..... 50

Şekil 4.12. Yazar sayısının 10 ve metin sayısının 2000 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri..... 51

# SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

| **Simgeler** | **Açıklamalar** |
|---|---|
| **sn** | Saniye |

| **Kısaltmalar** | **Açıklamalar** |
|---|---|
| **ÇTNM** | Çok Terimli Naive Bayes |
| **GA** | Genetik Algoritma |
| **KEYK** | K En Yakın Komşu Algoritması |
| **Ort** | Ortalama |
| **ROA** | Rastgele Orman Algoritması |
| **TÖA** | Topluluk Öğrenimi Algoritması |
| **TSOA** | Tavuk Sürü Optimizasyon Algoritması |

# 1. GİRİŞ

Yazının icadı ile günümüze kadar olan süreçte birçok farklı yol ile yazılı doküman üretilmiştir. Özellikle internet kullanımının artması ile yazılı doküman sayısı da hızlı bir oranda artmaktadır. Dijital ortamda gerçekleşen bu artış ile birlikte infobezite kavramı ortaya çıkmıştır.

İnfobezite (infobesity), bilgi/enformasyon (information) ve obezite (obesity) sözcüklerinin birleşimidir. Aşırı yeme hastalığına gönderme yapılarak bilginin de aşırıya kaçılarak tüketilmesi ile oluşan olumsuz durumun ifadesi olarak kullanılmaktadır. Bu görüş, gıda ve bilginin benzerliklerinin bulunduğunu, iki ürünün de üzerindeki farklı etkilerle değiştiği ve tüketici üzerinde kötü etkilerinin bulunduğunu savunmaktadır (Şimşek İşliyen, 2020). İnfobezite sadece bu duruma maruz kalan kişiyi değil toplumu da etkilediği için büyük önem arz etmektedir. Bilgi miktarında artış sağlanırken, doğru bilgi miktarı oranının düşmesi infobeziteyi tetiklemektedir. Sayısal ortamdaki bilgi artışı infobezite etkisine ek olarak bilgi hırsızlığı, dolandırıcılık, eserlerde çalıntı, takma isimler ile hakaret gibi birçok problemi de beraberinde getirmektedir. Ayrıca bu problemler gün geçtikçe de artmaktadır. Bu problemlere çözüm niteliği taşıyan bilginin otomatik analizine olan ilgi de artmaktadır. Bu ilgi, özellikle yazılı doküman sayısının fazla olmasından dolayı metin analizinde yoğunlaşmıştır. Metin analizi, aynı zamanda metin madenciliği alanının da gelişmesini sağlamıştır.

Metin madenciliği; doğal dil işleme, makine öğrenimi ve istatistik gibi birçok alandan faydalanarak metin içinde gerçekleştirdiği analizlerle daha önce bilinmeyen bilgilerin çıkarılmasını veya bilinen bir bilginin doğruluğunun tespitini sağlamaktadır (Kaşıkçı ve Gökçen, 2013). Metin madenciliği; otomatik çeviri, metin özetleme, metin sınıflandırma/tür tanıma, bağlantılı metinlerin keşfi, konuşma tanıma, duygu analizi ve metin yazarı tahmini gibi birçok çalışmada kullanılmıştır (Kaşıkçı ve Gökçen, 2013; Yüksel ve Tan, 2018).

Metinler üzerinde yapılan önemli analizlerden biri, metni üreten kişinin yani metin yazarının tahminidir. Metin yazarı tahmininin kullanım amaçları aşağıdaki gibi özetlenebilir;

- Bir metne sonradan ekleme veya değişiklik yapılıp yapılmadığı tespit edilebilir.

- Daha önceden bilinmeyen eserlerin yazarları ve müzik bestelerinin sahipleri bulunabilir.
- Eposta ve SMS yazarları ile forumlarda ve sosyal medya uygulamalarında propaganda amaçlı takma adla açılan hesaplardaki yazıların sahipleri tespit edilebilir (Alshaher ve Xu, 2020).
- Terörist bildirisi ve intihar notu gibi adli vakaların çözümünde kullanılan belgelerin yazarları tespit edilebilir.
- Akademik çalışmalarda intihal olup olmadığı tespit edilebilir (Soler-Company ve Wanner, 2018),
- Ünlü yazarların karakteristik özellikleri bulunabilir ve eser-yazar ilişkisi incelenebilir.
- Kötücül (malware) yazılımların yazarının belirlenmesinde kullanılabilir (Kalgutkar ve diğerleri, 2019)

Yukarıda sayılan tüm çalışma alanlarına ek olarak küresel bir problem olan infobezite etkisi metin yazarı tahmini ile azaltılabilir. Bu çalışmada daha önce bahsedilen tüm problemlerin çözüm temelinde kullanılabilecek bir yazar tahmini önerisi sunulmuştur.

Yazar tahmini alanında özellikle İngilizce yazılmış metinler üzerinde çok fazla çalışma mevcut olmasına rağmen Türkçe alanında yapılmış çalışma sayısı oldukça azdır. Farklı dillerde yapılan çalışmalar incelendiğinde, dilin yazar tanıma üzerinde etkisi olduğu her dil için farklı çalışmalara ihtiyaç duyulduğu gözlemlenmiştir (Aydemir, 2019; Patrick Juola ve Baayen, 2005; Kukushkina ve diğerleri, 2001; Otoom ve diğerleri, 2014; Zheng ve diğerleri, 2006). Türkçe dilinde çok az çalışmanın olması ve geniş bir derlemin bulunmaması bu çalışmanın gerekliliğini göz önüne sermektedir.

Bu çalışmada, Türkçe metinler üzerinden yazar tahmini için farklı haber sitelerinden 54 farklı yazar ve 46 837 adet köşe yazılarından oluşan bir derlem oluşturulmuştur. Derlemden elde edilen öznitelikler, yazar sayıları ve yazarlara ait kullanılacak metin sayılarındaki değişim, yazar tanıma işlemindeki sınıflandırma başarısı üzerinde etkili olabilmektedir. Bu yüzden çalışmanın kapsamlı analizler sunması amaçlanarak, hazırlanan derlemden farklı boyutlarda ham veri setleri çıkarılmıştır. Bu şekilde aynı analiz birden fazla durum için yorumlanabilir hale getirilmiştir. Hazırlanan her ham veri kümesi için öznitelik çıkarımında n-gram, sözcüksel ve sözdizilimsel analizler kullanılmıştır. Bu şekilde yazar tanıma işlemine farklı pencerelerden bakabilme imkânı sağlanmıştır. Bu sayede derlemden elde edilen

değişik özniteliklerin yazar tanıma başarımını nasıl etkilediğine ve başarım artışı için neler yapılabileceğine dair sorulara cevap aranmıştır.

Derlemden elde edilen farklı özniteliklerin yanında ayrıca hem yazar tanımadaki sınıflandırma başarısını artırmak hem de sınıflandırma maliyetini azaltmaya yönelik öznitelik seçim işlemi uygulanmıştır. Öznitelik seçim işlemi için Genetik Algoritma (GA) ve Tavuk Sürü Optimizasyon Algoritması (TSOA) kullanılmıştır. Her iki algoritmada da uygunluk değeri hesaplamalarında Rastgele Orman Algoritması (ROA) kullanılmıştır. Farklı optimizasyon algoritmasının kullanılması karşılaştırma yapılmasına ve çalışmanın geniş analizler sunmasına olanak sağlamıştır.

Çalışmanın son aşamasındaki yazar sınıflandırma işleminde ise, klasik makine öğrenimi algoritmaları gibi tekli sınıflandırıcı model kullanımının aksine birden fazla sınıflandırıcı model içeren Topluluk Öğrenimi Algoritması (TÖA) kullanılmıştır. Derlem üzerinde gerçekleştirilen tüm analizlerden sonra seçilen veri kümesi üzerinden TÖA ile 6 farklı sınıflandırma işlemi gerçekleştirilmiştir.

Bu çalışma aşağıda belirtilen 7 hususdan dolayı önemli görülmektedir:

1) Metin yazarı tahmini probleminin her geçen gün öneminin artırması ve bu çalışmanın konuya çözüm önerisi sunması
2) Türkçe çalışma sayısının azlığı ve bu çalışmanın geniş bir derlem sunması
3) Literatürdeki diğer Türkçe çalışmalara göre bu çalışmada tahmin edilecek yazar sayısının fazlalığı ve bu fazlalığa rağmen yüksek başarının elde edilmesi
4) Kullanılan her durum için birden fazla analiz yapılıp karşılaştırılması
5) Yazar tanıma süreci için hazırlanan derlem üzerinden farklı açıdan toplam 8 816 veri kümesinin incelenmesi
6) Hazırlanan veri setleri üzerinde öznitelik seçim işleminde GA-ROA ve TSOA-ROA yaklaşımlarının kullanılması
7) Seçilen en uygun veri kümesinin, TÖA ile gerçekleştirilen sınıflandırma başarısının değerlendirilmesi

Bu çalışma, giriş bölümü ile birlikte toplamda 5 bölümden oluşmaktadır. İkinci bölüm, Türkçe ve farklı dillerde daha önce yapılmış çalışmaların özetlerinin sunulduğu literatür

kısmıdır. Üçüncü bölümde, çalışma kapsamında kullanılan materyal ve metotlar verilmiştir. Bu bölümde çalışmada kullanılan tüm analizlerin ve metotların çalışma prensipleri, mimarileri ve çalışmada hangi amaçla kullanıldıkları açıklanmıştır. Dördüncü bölüm, deneysel çalışmaların tüm adımları ve elde edilen bulguların tartışılıp sunulduğu bölümdür. Son bölümde ise bu çalışmanın diğer Türkçe çalışmalar ile karşılaştırılmış ve elde edilen sonuçlar tartışılmıştır. Ayrıca bu bölümde çalışmanın literatüre katkıları ve geliştirilmesi için öneriler yer almaktadır.

# 2. LİTERATÜR TARAMASI

Metin yazarı tanıma alanındaki çalışmaların temeli 1871 yılında, uzunluğu aynı olan kelimelerin kullanım sıklığını inceleme fikri ile atılmıştır. Morgan tarafından ortaya atılan bu fikir, üslup analizi kavramının da temelini oluşturmuştur (De Morgan ve De Morgan, 1882). 19 yüzyılın sonlarına doğru üslup analizinin ilk el ile hesaplama örneği Mendenhall tarafından gerçekleştirilmiştir. Bu örnek, yazma stilinin istatiksel olarak ölçülmesi üzerine geliştirilmiştir (Mendenhall, 1887). Yazar bilgisini elde etmek için kullanılan üslup analizi; Shakespeare oyun metinleri üzerinde de uygulanmıştır. Bu çalışmaları, George Yule'ün 1944'te kelime zenginliği analizi için kelime sıklığını ölçmesi gibi benzer çalışmalar izlemiştir (Yule, 1944). Yazar tanıma alanında yapılan bilgisayar destekli ilk kapsamlı çalışma Mosteller tarafından yazarları tartışma konusu olan ve 146 metin içeren 'The Federalist Papers' üzerinde gerçekleştirilmiştir. Yapılan çalışmada metin içerisinde 'and' ve 'in' gibi kelimelerin kullanım sıklıkları istatistiksel olarak analiz edilmiştir (Mosteller ve Wallace, 1964).

Geçmişten günümüze yazılı eserler çoğalmış ve yazar tanıma problemine ilgi de artmıştır. Yazar tanıma alanında özellikle İngilizce diline ait çalışmalar oldukça fazladır. Çalışma sayısının yüksek olmasına rağmen, dilin yapısal özelliklerinin farklılığından dolayı her dil için uygulanabilecek geçerli bir metot bulunamamıştır. Özellikle son yıllarda İngilizce dışındaki farklı dillerde de çalışmalar artmıştır. Bu çalışma kapsamında son yıllarda yapılmış çalışmalar veri kümeleri boyutları, metin dili, performans değerleri ile ilgili bilgiler verilerek özetlenmiştir.

Arapça üzerinde yapılan bir çalışmada, eski metinler üzerinde Manhattan, Cosine, Stamatatos ve Canberra mesafeleri, Çok Katmanlı Algılayıcı, Destek Vektör Makinesi ve Doğrusal Regresyon ile yazar tanıma işlemi gerçekleştirilmiştir. Çalışmada, toplamda 10 adet yazar üzerinde kelimeler, kelime-bigram, kelime-trigramlar, kelime-tetragramlar ve nadir kullanılan kelimeler için ayrı ayrı analizlerle veri setleri hazırlanmıştır. En iyi performans, nadir kullanılan kelimeler ile gerçekleştirilen analiz sonucu oluşturulan veri kümesi üzerinde Sıralı Minimal Optimizasyona dayalı Destek Vektör Makinesi sınıflandırıcısı kullanılarak % 80 olarak elde edilmiştir (Ouamour ve Sayoud, 2013).

Arapça metinler üzerinde yapılan bir diğer çalışmada 7 yazar ve toplamda 456 metin ile yazar tahmini işlemi gerçekleştirilmiştir. Sözcüksel, sözdizimsel, yapısal ve içeriğe özgü özellikler içeren toplamda 12 öznitelik kullanılarak Destek Vektör Makinesi ve Fonksiyonel Ağaç Algoritması ile sınıflandırma gerçekleştirilmiştir. Holdout testi ve Fonksiyonel Ağaç Algoritması kullanılarak % 82 doğruluk elde edilmiştir (Otoom ve diğerleri, 2014).

İngilizce metinlerde yazar ve cinsiyet tanıma üzerine gerçekleştirilen bir çalışmada blog ve edebi yazıları içeren iki farklı veri kümesi kullanılmıştır. İlk veri kümesi 23 yazar ve onlara ait toplamda 4284 blog metni içermektedir. İkinci veri kümesi 16 yazar ve her yazarın 3 kitabı ile oluşturulmuştur. Metinler üzerinde 15 farklı öznitelik kümesi ve bu setlerin birleşimi ile toplamda 16 farklı öznitelik kümesi oluşturulmuş ve her biri Destek Vektör Makineleri Kitaplığı (LibSVM -with a linear kernel) algoritması ile eğitilmiştir. Yazar tanıma probleminde en yüksek doğruluk oranı blog verileri için %78,16, edebi eserleri içeren veri kümesi için %95,03 olarak elde edilmiştir (Soler-Company ve Wanner, 2018).

Hint bölgesel dillerinden Marathi dili ile yazılmış metinler üzerinden Karar Ağacı ile Sıralı Minimal Optimizasyon algoritmaları kullanarak yazar tanıma işlemi gerçekleştirilmiştir. Çalışmada 5 yazar ve onlara ait eğitim için 10, test için 5 metin ile toplamda 15 metin kullanılmıştır. Metinler üzerinde sözcüksel ve stil özellik analizleri ile öznitelik setleri oluşturulmuştur. Hazırlanan öznitelik setleri ile Sıralı Minimal Optimizasyon Algoritması sınıflandırması sonucunda % 80 doğruluk oranına ulaşılmıştır (Digamberrao ve Prasad, 2018).

Kısa metinlerde yazar tanıma problemi için Yelp Review veri kümesinden 4521 yazar ve her yazar için en az 100 kısa metin olacak şekilde veri kümesi hazırlanmıştır. Hazırlanan veri kümesi üzerinde 3 adımlı özellik azaltma işlemi gerçekleştirilmiştir. Bu işlemler küçük harf normalleştirme, noktalama işaretlerini ayırma ve bir kelimenin tüm veri kümesinde yalnızca bir kez görünmesini kaldırma şeklindedir. Çalışma sonucu, Destek Vektör Makinesi sınıflandırıcısı ile unigram, bigram modelleri ve sözbirimleştirme (lemmatization) kombinasyonları ile % 90,5 doğruluk değerine ulaşılmıştır (Vijayakumar ve Fuad, 2019).

Arapça dilinde yapılmış bir diğer çalışmada 3 farklı veri kümesi ve her veri kümesi içerisinde 8 farklı alt veri kümesi oluşturularak yazar tahmini işlemi gerçekleştirilmiştir. Bu setler;

stilometrik özellikler, kelime torbası yöntemi (bag-of-words) uygulanarak çıkarılan farklı öznitelikler ve iki kümenin kombinasyonu şeklindedir. Üç veri kümesi de kendi içinde dengeli (her yazara ait sırasıyla 50, 100, 300 metin ile) ve dengesiz (yazar sayısı sırasıyla 11, 7, 5, 8 olup her grupta farklı sayıda metin ile) bölünme gerçekleştirilerek alt veri kümeleri oluşturulmuştur. Performans olarak en yüksek değer çoğu durumda Adaboost sınıflandırıcısı ile dengeli veri kümelerinde elde edilmiştir. Çalışma da en yüksek doğruluk % 99,83 oranı ile 3.set ve dengeli dağılımda her yazara 300 metin içeren alt küme ile elde edilmiştir (Al-Sarem ve diğerleri, 2020).

Thai dilinde yapılan bir çalışmada 200 yazar ve onlara ait toplamda 547 metin üzerinde yazar tanıma işlemi gerçekleştirmişlerdir. Çalışmalarında sözcüksel, sözdizimsel ve yapısal özellikler içeren toplamda 46 adet öznitelik kullanmışlar ve metinleri sabit uzunlukta parçalara ayırmışlardır. Çalışmada sınıflandırıcı olarak Olasılıksal K En Yakın Komşu Algoritması ile birlikte standart, kısmi ve motife edilmiş 3 farklı Hausdorff Mesafe yöntemi kullanarak %91,02 doğruluk değerine ulaşmışlardır (Sarwar ve diğerleri, 2020).

Çince üzerinde yapılan bir çalışmada, 4 yazar ve onlara ait toplamda 154 metin içeren derlem üzerinde farklı stilometrik analizler ile veri setleri hazırlanmıştır. Veri setleri üzerinde Destek Vektör Makinası ve ROA ile modeller oluşturulmuştur. Modeller arasında sınıflandırma hata oranı en düşük %60,33 ve en yüksek %70,93 değerleri elde edilmiştir (Hou ve Huang, 2020).

İngilizcede yapılan bir diğer çalışmada 5 farklı veri kaynağı kullanılmıştır. Bunlar;

- The Eron mail veri kümesinde 150 kullanıcı ve toplamda 0.5 milyon eposta verisi
- Farklı bloglardan toplanmış 100 blog yazarı ve toplamda 1000 blog yazısı
- Gutenberg projesi ile hazırlanan kütüphane kullanılarak 100 yazar ve toplamda 600 belge
- Twitter verileri ile 100 yazar ve onlara ait toplamda 10000 metin
- PAN 2018 verileri

5 farklı veri kümesi için terim sıklığı (tf), terim sıklığı-ters belge sıklığı (tf-idf) ve 1/sigma ağırlıklandırmaları ile jeton ve ngram öznitelikleri ile K En Yakın Komşu, Rastgele Orman, En Yakın Ağırlık Merkezi ve Destek Vektör Makinası algoritmaları ile sınıflandırma

işlemleri gerçekleştirilmiştir. Beş farklı veri kümesi için karşılaştırmalı deneysel sonuçlara göre, önerilen terim ağırlıklandırmanın yazar tanımada ümit verici bir performans artışı sağladığını savunmaktadırlar (Alshaher ve Xu, 2020).

Gutenberg veri tabanından alınan 7 yazar ve onlara ait toplamda 28 adet kitap üzerinde 1500 kelime bloğu şeklinde parçalar ile 5 farklı boyutta kümeler kullanılarak yazar tanıma işlemi gerçekleştirilmiştir. Metinler üzerinde karakter tabanlı, kelime tabanlı ve sözdizimsel özelliklerden oluşan on üç stilometrik özellik analiz edilmiştir. 5 farklı boyut içeren kümeler için çıkarılan öznitelikler; Yapay Sinir Ağı, Naive Bayes, Destek Vektör Makineleri, Derin Öğrenme Sinir Ağları ile eğitim ve test işlemine tabi tutulmuştur. Sınıflandırma da en yüksek doğruluk değerine %98 ile ulaşılmıştır (Boran ve diğerleri, 2020).

Metinler, yazıldıkları dile ait özellikler taşımaktadır. Bundan dolayı yazar tanımada kullanılan analizlerin de farklılaşması beklenir. Literatürde Almanca (Diederich ve diğerleri, 2003), İbranice (Koppel ve diğerleri, 2006), Yunanca (Keselj ve diğerleri, 2003), Rusça (Kukushkina ve diğerleri, 2001), Danca (P. Juola, 2005), İtalyanca (Savoy, 2015) gibi birçok dilde çalışma yapılmıştır. Çalışmaların çoğu farklı öznitelik çıkarımı üzerine yoğunlaşmıştır.

Türkçe metinler üzerinde yapılan yazar tanıma çalışmaları incelendiğinde, genelde yazar sayısı az tutulmuş ve metin üzerinde yapılan farklı analizlerle veri kümesi oluşturmaya yoğunlaşmıştır (Agun ve diğerleri, 2017; Aydemir, 2019; Ekinci ve Takci, 2012; Türkoğlu ve diğerleri, 2007). Metin üzerinden elde edilen öznitelikler metin yazarını tanımada başarıyı etkileyen en önemli unsurlar arasındadır. Bunun dışında metin dili, metin konusu ve alanı, yazara ait yeterli metinlerin olup olmadığı, yazarın yazım tarzının zamanla değişme ihtimali, tahmin edilecek yazar sayısı da doğrudan veya dolaylı olarak başarıyı etkilemektedir. Çalışmalarda, genellikle yazar sayısı arttığında başarının düştüğü gözlemlenmektedir (Agun ve diğerleri, 2017; Aydemir, 2019; Ekinci ve Takci, 2012; Türkoğlu ve diğerleri, 2007).

Patton ve Can tarafından cümle uzunlukları, sık kullanılan kelime, hece sayısı gibi özellikler kullanılarak yazar analizi gerçekleştirilmiştir. Yaşar Kemal'e ait İnce Mehmed kitap serisi ham veri kümesi olarak kullanılmıştır. Romanların farklı tarzda olmasının sonucu etkilediği sonucu çıkarılmıştır (Patton ve Can, 2004).

Amasyalı ve Diri'nin yaptıkları bir çalışmada 18 yazar ve her birine ait 35 metin ile oluşturdukları veri kümesi üzerinde n-gram metodu ile yazar, metin türü ve yazar cinsiyeti olmak üzere 3 farklı tahmin işlemini gerçekleştirmişlerdir. Ortalama doğruluk değerleri; yazar tanıma için %83, cinsiyet tahmini için % 96 ve tür tahmini için % 93 olarak elde edilmiştir (Fatih Amasyali ve Diri, 2006).

Türkoğlu ve arkadaşları tarafından gerçekleştirilen çalışmada üç farklı boyutta veri kümesi hazırlanmıştır. Oluşturdukları veri setlerinde sınıflandırılacak yazar sayıları sırasıyla 18, 9 ve 9 olarak verilmiştir. Çalışmalarında sözcüksel, yapısal, fonksiyonel ve n-gram metotlarını kullanarak öznitelik setleri oluşturulmuştur. Sınıflandırma işlemlerinde Destek Vektör Makinaları, Naive Bayes, k En Yakın Komşu, Rastgele Orman ve Çok Katmanlı Algılayıcı Algoritmaları kullanılmış ve en başarılı sonuçlar Destek Vektör Makinaları ve Çok Katmanlı Algılayıcı metotlarında gerçekleşmiştir. Veri setlerinin ortalama sınıflandırma başarısı sırasıyla %72,4, %80,0 ve %82,9 olarak elde edilmiştir. Özellik seçme algoritmalarının da kullanıldığı çalışmada, 9 yazarlı ve 315 adet metin içeren veri kümesi 3 ile en yüksek doğruluk oranı % 96,9'a ulaşmıştır. Çalışma sonuçlarında, yazar sayısının sınıflandırma doğruluğunu doğrudan etkilendiği de vurgulanmıştır (Türkoğlu ve diğerleri, 2007).

Bay ve arkadaşları tarafından gerçekleştirilen çalışmada; noktalama işaretlerinin sayıları, cümle sayıları, metinde Türkçe olmayan kelime sayıları gibi analizler ile yazar tahmini gerçekleştirilmiştir. Çalışmada toplamda 17 yazar ve her yazar için 50 adet metin kullanılmıştır. Çalışma, 10 ve 7 yazar bilgisi ile iki ayrı veri kümesine ayrılarak gerçekleştirilmiştir. Makine öğrenme algoritması olarak Naive Bayes, Destek Vektör Makinesi, K-En Yakın Komşu ve Karar Ağacı algoritmaları ve özellik seçme algoritması olarak Ki-kare özellik seçme yöntemi kullanılmıştır. Çalışmada en yüksek doğruluk oranı K-En Yakın Komşu ile % 99,7 olarak elde edilmiştir (Bay ve Çelebi, 2016).

Aslantürk ve arkadaşları tarafından 9 yazar ve farklı alanda 513 metin verisi kullanılarak sözdizimsel ve sözcüksel analizler ile yazar ve konu tahmini çalışması gerçekleştirilmiştir. Çalışma da metnin konusu bakımından homojen ve heterojen veri setleri oluşturulmuş ve sınıflandırma doğrulukları tartışılmıştır. Hazırlanan veri kümesi bilgileri ve sınıflandırma doğruluk oranları şöyledir;

- 9 yazar ve her birinden 12 metin ile heterojen veri kümesi ile %55,56

- 5 yazar ve her birinden 12 metin ile homojen veri kümesi ile %83,33
- 4 yazar ve her birinden 12 metin homojen veri kümesi ile %81,25

Homojen verilerden elde edilen sınıflandırma doğruluğu heterojen verilerden yüksek olduğu gözlenmiştir. Bu çalışmada konu bakımından homojen ve heterojen kavramları tartışılmış olsa da sınıflandırılan yazar sayısı eşit olmadığı gözlemlenmiştir (Aslantürk ve diğerleri, 2010).

Ekinci'nin yaptığı çalışmada, 5 yazara ait e-posta metinleri üzerinde Çok Katmanlı Yapay Sinir Ağları, Destek Vektör Makinaları ve Karar Ağaçları kullanılarak yazar tahmini gerçekleştirilmiştir. Çalışmada; ortalama kelime uzunluğu, karakter, büyük harf, küçük harf sayılarının da bulunduğu toplamda 43 öznitelik ile en yüksek %84 doğruluk değerine ulaşılmıştır (Ekinci ve Takci, 2012).

Yazar sayısının 25 olduğu bir çalışmada ise Lojistik Regresyon, Naive Bayes ve Çok Katmanlı Yapay Sinir Ağları gibi makine öğrenme algoritmalrı ile yazar tanıma gerçekleştirilmiştir. Çalışmada metinlerin sözdizimsel yapıları ve kelime kökleri öznitelik oluşturmada kullanılmış ve en yüksek F-ölçüm oranı %73,22 değerine ulaşılmıştır (Agun ve diğerleri, 2017).

Aydemir'in yaptığı bir çalışmada, kelime sayısı, cümledeki ortalama kelime sayısı, sıfat sayısı gibi 30 adet öznitelik ile Çok Katmanlı Yapay Sinir Ağı modeli ile yazar tanıma gerçekleştirilmiştir. Yapay sinir ağında gizli katman sayısı bir, iki, üç ve dört katmanlı ve her bir katmanda farklı sayıda yapay sinir hücresi bulunan modeller ile çeşitli deneyler gerçekleştirilerek sınıflandırma doğruluğu incelenmiştir. En yüksek doğruluk tek gizli katmanda 35 yapay sinir hücresi bulunan model ile elde edilmiştir. Yazar sayısının 40 olduğu ve 400 adet köşe yazısının kullanıldığı bu çalışmada en yüksek doğruluk oranına %72 ile ulaşılmıştır (Aydemir, 2019).

# 3. MATERYAL VE METOT

Metinden yazar tahmini probleminin çözümü için Türkçe dilinde yapılmış çalışma sayısı kısıtlıdır ve bu alanda geniş bir derlem bulunmamaktadır. Çevrimiçi Türkçe haber sitelerinden konu ve alanlarına bakılmaksızın köşe yazıları toplanmıştır. Genellikle günlük yazıların olduğu metinlerin, manuel toplanması zor ve zahmetlidir. Bunun için her haber sitesine özel Python programlama dili ile geliştirilen uygulama sayesinde otomatik metin toplama gerçekleştirilmiştir. Bu uygulama sayesinde her haber sitesine, Tekdüzen Kaynak Bulucu (Uniform Resource Loader -URL) bilgileri üzerinden erişim sağlanmıştır. Haber sitelerinin sayfa tasarımları farklılık göstermektedir. Zamanla bazı haber sitelerinde de sayfa tasarımında kendi içinde farklılıkların olduğu gözlemlenmiştir. Bunun için her haber sitesine uygun ve tasarım farklılığının zaman içinde değişebildiği göz önüne alınarak farklı uygulamalar hazırlanmıştır.

Bu uygulamalarda, rastgele seçilen yazarlar ve onların köşe yazılarına ait URL bilgileri tutulmaktadır. Her köşe yazısının bulunduğu sayfa HTML formatında incelenir. Uygulamalar html etiketlerini inceler ve köşe yazısından bağımsız olan içeriklerin (reklam, link, resim, vb.) temizlenmesini sağlar. Temizleme işleminden sonra başlık ve haber içeriğinin alınması ve ilgili yazar bilgisi ile birlikte kayıt işlemini gerçekleştirir.

Çalışma kapsamında hazırlanan uygulamalar sayesinde toplam 54 adet yazar ve onlara ait değişken sayıda ve değişken uzunlukta metinler ile bir derlem oluşturulmuştur.

Derlem üzerinde iki farklı analiz ile öznitelik çıkarım işlemi gerçekleştirilmiştir. İlk yöntem karakter tabanlı n-gram tekniğidir. Bu teknik uygulanırken özellikle iki parametre önemlidir. Bunlar n değeri ve jeton sayısı. Bu parametrelerin farklı değerlerinin sınıflandırma problemi üzerindeki etkisi incelenerek en uygun değerler seçilmiştir. İkinci öznitelik çıkarım işlemi ise metinlerin sözcüksel, yapısal ve sözdizilimsel yapısının analizidir. İki farklı analizin bir arada kullanımındaki başarısını incelemek üzere birleştirilmiş analiz de oluşturulmuştur. Hazırlanan tüm veri setleri ile oluşturulan modellerin doğruluk oranları ve model oluşturma süreleri bakımından birbiri ile kıyaslanmıştır.

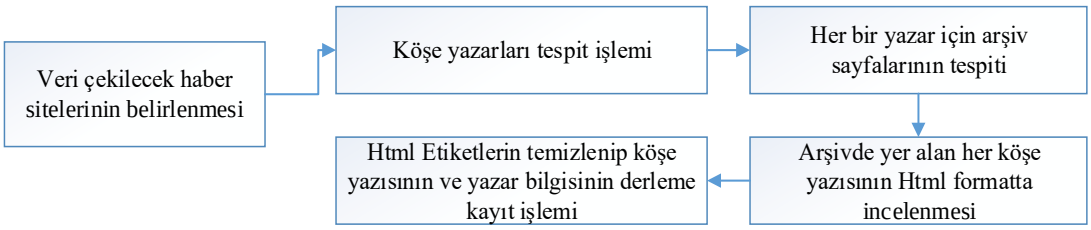
Hazırlanan tüm veri setlerinin sınıflandırma doğruluğunu artırmaya yönelik 2 farklı öznitelik seçim işlemi gerçekleştirilmiştir. Farklı boyut ve farklı analiz ile hazırlanan veri setleri üzerinde 2 farklı yöntem (GA ve TSOA) ile öznitelik seçim işlemi gerçekleştirilmiştir. Her iki öznitelik seçim işlemi için uygunluk fonksiyonu olarak ROA ve uygunluk değeri olarak da sınıflandırıcılardan elde edilen modellerin doğruluk değerleri kullanılmıştır. Öznitelik seçim işlemi sonrasında oluşturulan yeni veri setleri üzerinde TÖA ile sınıflandırma işlemi gerçekleştirilmiştir.

### 3.1. Derlem (Corpus)

Literatürdeki çalışmalar incelendiğinde yazar tanıma için derlem olarak kitap, tweet, köşe ve blok yazıları sıklıkla kullanılmaktadır. İngilizce diline ait çalışmalarda kullanılması için hazırlanmış derlemlere ulaşmak mümkün olsa da Türkçe yazar tanıma alanında hazırlanmış geniş bir derlem bulunmamaktadır. Bu çalışmada, farklı haber sitelerinde yayınlanan köşe yazıları üzerinden yazar tanıma işlemi gerçekleştirilmiştir. Köşe yazıları bir yazara ait birden fazla yazıya kolaylıkla erişimin sağlanabilmesinden dolayı tercih edilmiştir.

Ham veri kümesi için farklı haber sitelerinden köşe yazısı ve yazar bilgisi toplanma işleminde Python 3.0 programlama dili kullanılarak bir konsol uygulaması geliştirilmiştir. Geliştirilen uygulamada 4 farklı haber sitesinden farklı sayılarda yazar ve onlara ait köşe yazıları ile bir derlem oluşturulmuştur. Haber sitelerinin tasarımının birbirinden farklı olması ve sitelerin farklı zamanlarda sayfa tasarımında değişim uygulamalarından dolayı yazılımın bu değişkenliğe uyum sağlaması gerekmektedir. Ayrıca her haber sitesi ve haber sitesindeki köşe yazar listesi, köşe yazarlarının arşiv sayfaları gibi farklı sayfaları ve tasarımdaki değişimler incelenerek ayrı bir okuma ve kayıt fonksiyonu oluşturulmuştur.

Derlem oluşturulurken Şekil 3.1.'de belirtilen süreçler takip edilmiştir. Her haber sitesi için köşe yazarları listesi tespit edilmiş ve bu yazarlara ait arşivlerin URL bilgileri kaydedilmiştir. Her arşiv için içerisinde yer alan her köşe yazısına ait URL bilgisi ile bir liste oluşturulmuştur. Listede yer alan her bir URL ile köşe yazılarının sayfalarına ulaşılmıştır. Sayfalar html formatında kayıt edilmiştir. Web sayfası içerisinde html etiketlere göre haber başlığı, köşe yazısı içeriği ve yazar bilgisi derleme kayıt edilmiştir. Haber ile bağlantısı olmayan reklam sosyal medya linkleri vb. bilgiler ile haber içerisinde yer alan resim, video gibi ekler derleme eklenmemiştir.




Şekil 3.1. Derlem oluşturma süreçleri

Oluşturulan ham veri kümesinde 54 adet yazar ve değişken sayıda ve uzunlukta toplam 46 837 adet köşe yazısı mevcuttur. El ile köşe yazılarından ile bir derlem oluşturmak zor ve zaman alacak bir işlemdir. Her sitenin farklı tasarıma sahip olması ve farklı zamanlarda haber sitelerinin tasarım güncellemesi problemi için farklı fonksiyonların hazırlanması ile çözüm sunulmuştur. Hazırlanan uygulama ile otomatik veri çekme işlemi gerçekleştirilmiştir.

### 3.2. Öznitelik Çıkarımı

Türkçede doğal dil işleme teknikleri birçok çalışma alanında kullanılmıştır. Bunlardan bazıları;

- Metinlerde yazım yanlışlarının bulunması ve düzeltilmesi (Delibaş, 2008)
- Yakın anlamlı sözcüklerin bulunması (Polat ve Körpe, 2018) İstem bilgisi (Doğan, 2016)
- Müzik türü sınıflandırması (Coban ve Karabey, 2017)
- Duygu analizi ve sınıflandırması (Yıldırım ve diğerleri, 2015; Yüksel ve Tan, 2018)
- Sözlük tabanlı uygulama geliştirilmesi (Vural, 2013)

Doğal dil teknikleri, yazar tanıma probleminde de sıklıkla kullanılmıştır. Çünkü her metin kendini oluşturan yazarın izini taşır. Bu iz sözcüksel, sözdizimsel, yapısal veya içeriğe özgü özellikler olarak karşımıza çıkmaktadır. Sözcük ve karakter tabanlı özellikler; büyük harf sayısı, bir kelimede kullanılan ortalama harf sayısı, bir cümlede yer alan ortalama kelime sayısı, metindeki kelime çeşitliliği gibi özellikleri içerir. Sözdizimsel özellikler ise sıfat, fiil, zamir gibi sözcük türlerinin kullanımı ve sıklığı, noktalama işaretleri kullanımını gibi özellikleri kapsar.

Literatürde genel olarak metin üzerinde doğal dil işleme teknikleri ile yapılan analizler 5 seviye de incelenebilir. Bunlar;

1. Karakter ve kelime seviyesi (sözcük)
2. Cümle ve paragraf seviyesi (yapısal)
3. Noktalama işaretleri ve işlev sözcükleri seviyesi (sözdizimsel)
4. Belirli bir yazarın benzersiz unsurları (kendine özgü)
5. Konu seviyesi (içerik)

Genel olarak çalışmalarda sözcük, yapısal ve sözdizilimsel analizler gerçekleştirilmiştir. Yazarın stilini bulmak bir başka deyişle yazarın kendine özgü ifadelerini bulmak yazar tanıma probleminin özellikle ilk yıllarında gerçekleştirilen çalışmalarda gözlemlenmiştir. Bu çalışmada farklı dillerde yapılan çalışmalardaki sık kullanılan sözcüksel ve sözdizilimsel analizlere de yer verilmiştir.

Çalışmada, yazar tanımada kullanılmak üzere doğal dil işleme teknikleri ile 2 farklı öznitelik çıkarım işlemi ile veri setleri oluşturulmuştur. Farklı öznitelik yapılarına sahip veri setlerinin çıkarılmasındaki amaç; yazar tanımada en yüksek doğruluğu sağlayacak özniteliklerin elde edilmesidir. Veri setlerini oluştururken Zemberek ve NLTK (Natural Language Toolkit) kütüphaneleri kullanılmıştır.

Zemberek yazım denetimi, sözcük türü analizi, kelime oluşturma ve önerme, morfolojik ayrıştırma, hece çıkarma gibi doğal dil işlemlerini daha kolay gerçekleştirmek için geliştirilmiş erişime açık bir kütüphanedir (Akın ve Akın, 2007).

NLTK kütüphanesi de Python dilinde geliştirilmiş açık kaynak kodlu bir kütüphanedir. Bu kütüphane kullanılarak cümleleri ve kelimeleri ayırma, kelime kökü bulma, sözcük türü analizi gibi birçok işlem gerçekleştirilebilir (Loper ve Bird, 2002).

Sözcük türleri belirlenmesinde Zemberek; kelime, cümle ve hece çıkarımları ile durak kelime tespitinde NLTK kütüphanelerinden yararlanılmıştır.

Çalışmada kullanılan 2 farklı öznitelik çıkarım işlemleri “Analiz1” ve “Analiz2” olarak adlandırılmış iki yöntemin birleştirilmesi ise “Karma Analiz” olarak ifade edilmiştir.

### 3.2.1. Analiz1-n-gram tabanlı öznitelikler

N-gram yöntemi, metinler üzerinde n büyüklüğünde karakter parçalanması olarak ifade edilebilir. Bu yöntem ile sınıflandırma işlemi metin içerisindeki kullanım sıklığına bakılarak gerçekleştirilir. N-gram, dilden bağımsız olmasından da kaynaklı olarak metin sınıflandırmada sıkça kullanılan basit ve güvenilir bir yöntemdir.

Literatürde n-gram tekniği ile metin içerisinden anahtar kelime çıkarımı (Hossny ve diğerleri, 2020), içerik/konu çıkarımı (Zdonak, 2020), duygu analizi (İlhan ve Sağalltıcı, 2020), dosya türü belirleme (Li ve diğerleri, 2005) ve yazar tahmini gibi birçok metin madenciliği probleminde kullanılmıştır.

N-gram, literatürde kelime tabanlı ve karakter tabanlı olarak ayrılmıştır. Bu çalışmada, karakter tabanlı n-gram yöntemi kullanılmıştır. Kelime tabanlı n-gram yöntemi veri sayısının azlığından dolayı tercih edilmemiştir.

Literatürde iki farklı yöntem ile n-gram jetonlama gerçekleştirilmektedir. Bu yöntemler n adet karakter ile kaydırma ve metindeki her karakter başlangıç kabul edilerek gerçekleştirilen kaydırmadır.

Örneğin; metin= “tespit yöntemi” ve n=4 değeri için kelimeler arasında boşluk karakteri # ile gösterilirse oluşan n-gram listesi;

- n adet karakter ile kaydırma:

“tesp”, “it#y”, “önte”, “mi##”

- her karakter başlangıç kabul edilerek gerçekleştirilen kaydırma:

“tesp”, “espi”, “spit”, “pit#”, “it#y”, “t#yö”, “#yön”, “yönt”, “önte”, “ntem”, “temi”

Bu çalışmada daha ayrıntı analiz sunduğundan her karakter başlangıç kabul edilerek kaydırma işlemi gerçekleştirilmiştir. Kullanılacak n ve jeton parametreleri için farklı senaryolar hazırlanmıştır. Ulaşılan en uygun n ve jeton değerleri ile deneyler gerçekleştirilmiştir. Çalışmanın devamında bu yöntem ile oluşturulan veri setleri, analiz1 olarak adlandırılmıştır.

### 3.2.2. Analiz2-sözcüksel, yapısal ve sözdizilimsel tabanlı öznitelikler

Çalışmada kullanılan ikinci analiz, sözcüksel, yapısal ve sözdizilimsel analizleri kapsar. Çizelge 3.1'de çalışmada kullanılan özniteliklere ait detaylı bilgiler yer almaktadır. Genel bir ifade ile kelime ve cümle bazında yapılan incelemelerle sözcüksel ve yapısal analiz, işlevsel yapıların incelenmesi, noktalama işaretleri ve karakter tabanlı incelemelerle sözdizilimsel analiz gerçekleştirmiştir. Öznitelik çıkarımı, Zemberek ve NLTK kütüphanelerinden yararlanılarak oluşturulmuş Pyton uyguması ile gerçekleştirilmiştir. Kullanılan tüm özniteliklerin kısa kod ve açıklaması Çizelge 3.1'de yer almaktadır. Toplamda 40 adet öznitelik belirlenmiştir.

Çizelge 3.1. Çalışmada kullanılan sözcüksel, yapısal ve sözdizilimsel özniteliklerin listesi

| Kod | Öznitelik | Öznitelik Açıklaması |
|---|---|---|
| ks | Kelime Sayısı | Metinde geçen toplam kelime sayısı |
| oku | Ortalama Kelime Uzunluğu | toplam karakter/ks |
| fks | Farklı Kelime Sayısı | Metinde tek bir kez geçen kelime sayısı |
| ofks | Ortalama Farklı Kelime Sayısı | fks/ks |
| cs | Cümle Sayısı | Metinde geçen cümlelerin sayısı |
| coks | Cümledeki Ortalama Kelime Sayısı | Bir cümlede kullanılan kelime sayısı ks/cs |
| is | İsim Sayısı | Metinde geçen toplam isim sayısı |
| fs | Fiil Sayısı | Metinde geçen toplam fiil sayısı |
| ss | Sıfat Sayısı | Metinde geçen toplam sıfat sayısı |
| zmrs | Zamir Sayısı | Metinde geçen toplam zamir sayısı |
| kslts | Kısaltma Sayısı | Metinde geçen toplam kısaltma sayısı |
| ois | Ortalama İsim Sayısı | Ortalama isim sayısı is/ks |
| ofs | Ortalama Fiil Sayısı | Ortalama fiil sayısı fs/ks |
| Oss | Ortalama Sıfat Sayısı | Ortalama sıfat sayısı ss/ks |
| ozmrs | Ortalama Zamir Sayısı | Ortalama zamir sayısı zmrs/ks |
| okslts | Kısaltma Sayısı | Ortalama kısaltma sayısı kslts/ks |
| shs | Sesli Harf Sayısı | Metinde geçen toplam sesli harflerin sayısı |

Çizelge 3.1. (devam) Çalışmada kullanılan sözcüksel, yapısal ve sözdizilimsel özniteliklerin listesi

| | | |
|---|---|---|
| sszhs | Sessiz Harf Sayısı | Metinde geçen toplam sessiz harflerin sayısı |
| krs | Karakter Sayısı | Metinde geçen toplam karakterlerin sayısı |
| khs | Küçük Harf Sayısı | Metinde geçen toplam küçük harf sayısı |
| bhs | Büyük Harf Sayısı | Metinde geçen toplam büyük harf sayısı |
| bkrs | Boşluk Karakteri Sayısı | Metinde geçen toplam boşluk karakterinin sayısı |
| tns | Toplam Noktalama Sayısı | Metinde geçen toplam noktalama işaretlerinin sayısı |
| ns | Nokta Sayısı | Metinde geçen nokta işaretlerinin sayısı |
| vs | Virgül Sayısı | Metinde geçen virgül işaretlerinin sayısı |
| nvs | Noktalı Virgül Sayısı | Metinde geçen noktalı virgül işaretlerinin sayısı |
| inüs | İki Nokta Üst üste Sayısı | Metinde geçen iki nokta üst üste işaretlerinin sayısı |
| sis | Soru İşareti Sayısı | Metinde geçen soru işareti işaretlerinin sayısı |
| üis | Ünlem İşareti Sayısı | Metinde geçen ünlem işareti işaretlerinin sayısı |
| üns | Üç Nokta Sayısı | Metinde geçen üç nokta işaretlerinin sayısı |
| çts | Çift Tırnak Sayısı | Metinde geçen çift tırnak işaretlerinin sayısı |
| Tts | Tek Tırnak Sayısı | Metinde geçen tek tırnak işaretlerinin sayısı |
| ps | Parantez Sayısı | Metinde geçen parantez işaretlerinin sayısı "(" ," )" |
| oshs | Ortalama Sesli Harf Sayısı | Ortalama bir kelimede geçen sesli harf shs/ks |
| osszhs | Ortalama Sessiz Harf Sayısı | Ortalama bir kelimede geçen sessizli harf sszhs/ks |
| okrs | Ortalama Karakter Sayısı | Ortalama bir kelimede kullanılan karakter sayısı krs/ks |
| okhs | Ortalama Küçük Harf Sayısı | Ortalama bir cümlede geçen küçük harf sayısı khs/cs |
| obhs | Ortalama Büyük Harf Sayısı | Ortalama bir cümlede geçen büyük harf sayısı bhs/cs |
| obkrs | Ortalama Boşluk Karakteri Sayısı | Ortalama bir cümlede geçen boşluk karakteri sayısı bkrs/cs |
| otns | Ortalama Noktalama Sayısı | Ortalama bir cümlede kullanılan noktalama sayısı tns/cs |

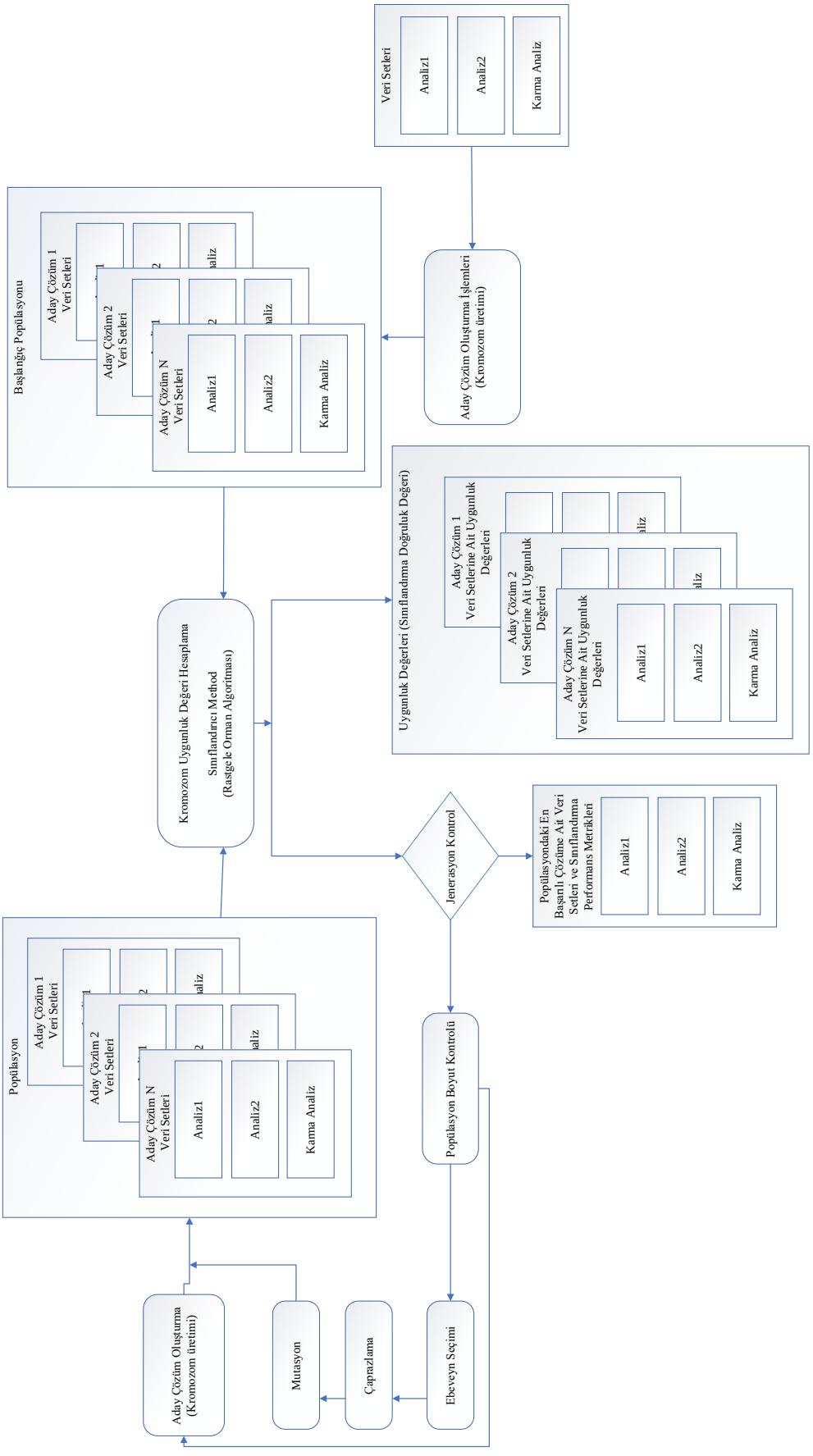
Çalışmanın devamında bu yöntem ile oluşturulan veri setleri, analiz2 olarak adlandırılmıştır.

## 3.3. Öznitelik Seçme Yöntemleri

Öznitelik seçim işlemi, orijinal veri kümesine ait özniteliklerden daha az sayıda ve daha etkili özniteliklere sahip alt küme oluşturma işlemidir. Amaç yazar tahmini için yeterli olabilecek en küçük öznitelik alt kümesini bulmaktır.

### 3.3.1. Genetik Algoritma

Genetik Algoritma (GA), doğal evrimin genetik ve evrimsel süreç benzetimi temeline dayanan bir sezgisel arama tekniğidir. İlk olarak ABD Michigan Üniversitesi'nde Profesör Holland tarafından önerilmiştir (Holland 1992). GA'da her biri olası çözüm olan kromozomlar ve bu kromozomların bağlı olduğu bir popülasyon bulunmaktadır. Her kromozomun, popülasyondaki varlık konumunu belirleyen bir uygunluk değeri vardır. Uygunluk değerini bulmak için kullanılan fonksiyon, her problem için farklı hesaplanabilir. Mevcut popülasyondan özel seçim yöntemleri ile ebeveyn seçimi gerçekleştirilir. Ebeveynler, gelecek nesil kromozomlarını üretmek için kullanılırlar. Art arda gelen nesiller boyunca nüfus yerel optimal çözümler üzerinden geliştirilir. Ebeveyn seçiminde genel olarak, uygunluk değeri en yüksek olan kromozomlar seçilir. Daha sonra en yüksek uygunluk değerlerine sahip ebeveynler ile çaprazlama işlemi gerçekleştirilir. Uygunluk değerleri en düşük olan kromozomlar, popülasyondan atılır. Böylece popülasyondaki kromozom sayısı sabit tutulur. Popülasyon içerisinde bazı kromozomlar üzerinde mutasyon gerçekleşebilmektedir. GA'da gerçekleştirilen çaprazlama ve mutasyon işlemleri sayesinde arama alanı tek yönlü değildir. Bir dizi bireysel çözümü ve testi göz önünde bulundurmasından dolayı küresel optimal çözümü bulma olasılığı yüksektir. GA, fonksiyonun süreksiz, ayırt edilemez veya doğrusal olmayan problemler dahil olmak üzere standart optimizasyon algoritmaları için uygun olmayan çeşitli optimizasyon problemlerini çözmek için uygulanabilir. Özellikle öznitelik seçme işlemleri için çok kullanışlıdır (Yang, 1998; Pappa, 2002).




Şekil 3.2. Çalışmada oluşturulan veri setlerinin GA ile öznitelik seçim işleminin aşamaları

Bu çalışmada, 3 analiz ile oluşturulmuş veri setlerine, ayrı ayrı GA ve ROA modelinin birleşimi ile öznitelik seçim işlemi uygulanmıştır. ROA, GA'nın uygunluk fonksiyonunun (fitness function) sonucunu döndüren fonksiyonu olarak seçildi. GA ile öznitelik seçim işleminde gerçekleştirilen aşamalar Şekil 3.2'de gösterilmiştir.

GA ile öznitelik seçme işleminde her bir veri kümesi için aşağıdaki adımlar gerçekleştirildi:

a. Her bir veri kümesi içerisinden, rastgele olarak seçilen öznitelikler ile farklı boyutlarda (öznitelik/sütun boyutu) alt veri kümeleri (kromozomlar) oluşturuldu. 32 kromozom içeren bir başlangıç popülasyonu hazırlandı. Popülasyondaki kromozom sayısı tüm işlemler süresince sabit tutuldu. Popülasyondaki her bir kromozom veri kümesinin öznitelik boyutu azaltılmış bir alt veri kümesidir ve olası bir çözümü temsil eder.
b. Başlangıç popülasyonu oluşturulduktan sonra, popülasyondaki her kromozomun uygunluk (fitness) değeri hesaplandı. Uygunluk değerinin hesaplanması için, her kromozom, ROA ile eğitildi ve test edildi. Oluşturulan modelinin doğruluğu, kromozomların uygunluk değeri (fitness value) olarak atandı.
c. GA'nın belirlenen sonlandırma kriteri kontrol edildi. Bu çalışmada sonlandırma kriteri olarak nesil sayısı seçildi ve değeri 250 olarak belirlendi. Sonlandırma kriteri sağlandığında i, sağlanmadığında ise d maddesi üzerinden işlemlere devam edildi.
d. Popülasyondaki kromozomlardan bazıları, yeni nesiller üretmek için ebeveyn kromozomlar olarak seçildiler ve bir ebeveyn kromozomlar havuzu oluşturuldu. Kromozom seçim işlemi için sabit durum seçimi kullanıldı. En yüksek uygunluk değerlerine sahip 8 kromozom ebeveyn kromozomlar olarak bu havuzda toplandı.
e. Ebeveyn kromozomlardaki bazı genlerin (özniteliklerin) yer değiştirmesi ile yeni yavru (offspring) kromozomlar oluşturmak için çaprazlama (crossover) işlemi gerçekleştirildi. Çaprazlama metodu olarak çift noktalı gen takası kullanıldı. Her iki nokta da rastgele seçildi. Ebeveyn havuzundan rastgele iki kromozom seçildi yine rastgele genler seçilerek yeni yavru kromozomlar oluşturuldu.
f. Sürekli yeni nesil üretimi sonucunda, belli bir süre sonra nesildeki kromozomlar tekrar edebilir ve farklı kromozom üretimi azalabilir. Bu yüzden yeni nesildeki kromozom çeşitliliğini artırmak için yavru kromozomlardan bazılarına mutasyon (mutation) uygulandı. Mutasyon erken yakınsamayı önlemek ve daha geniş arama alanını keşfetmek için kullanılmıştır. Mutasyon, kromozomlar üzerinde rastgele olarak gen değişimi evirme (inversion) yöntemi ile gerçekleştirildi.
g. Çaprazlama ve mutasyon ile oluşturulan yeni yavru kromozomlar popülasyona eklendi.

h. İşlem b maddesi üzerinden devam ettirildi.
i. Sonlandırma kriteri sağlandığında son popülasyonda en yüksek uygunluk değerine sahip olan kromozom, çözüm veri kümesi olarak seçildi.

Bu yeni veri setlerini kullanarak TÖA ile yeni sınıflandırma işlemleri gerçekleştirildi. Öznitelik seçim işlemi öncesi sınıflandırma sonuçları ile öznitelik seçim işlemi sonrası sınıflandırma sonuçları karşılaştırıldı. GA ve ROA modelinin birleşimi ile öznitelik seçim işleminin başarısı tartışıldı.

### 3.3.2. Tavuk Sürü Optimizasyon Algoritması

Tavuk Sürü Optimizasyon Algoritması (TSOA), tavuk sürüsündeki hiyerarşik düzeni ve sürü davranışlarını taklit eden bir optimizasyon tekniğidir. TSOA, sürüde yer alan horoz, civciv ve tavukların yemek arama davranışlarına dayanır.

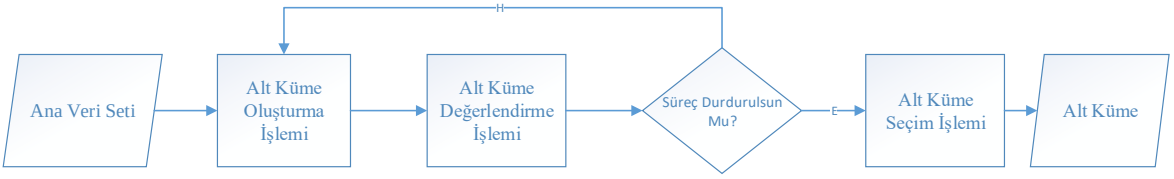
TSOA, yakınsama hızlılığı ve doğruluğu nedeniyle farklı birçok araştırma alanında kullanılmıştır. Bunlardan bazıları; robot rota planlanmasında (X. Liang ve diğerleri, 2020), anten dizilerinin maksimum yan lob seviyesi bastırma problemini çözmede (S. Liang ve diğerleri, 2020), sanal makine konsolidasyonu için kilitlenmeyen göç problemini çözmede (Tian ve diğerleri, 2017), endüstride doğrusal olmayan sistemlerin parametre tahmininde (Ren ve diğerleri, 2019), kablosuz sensör ağlarında enerji verimliliğini artırma (Osamy ve diğerleri, 2020) gibi çalışmalarıdır.

Algoritma 4 ana kural üzerine kurulur. Bunlar;

(1) Sürüde birden fazla alt grup olabilir. Her alt grup bir horoz, adedi parametrik olarak belirlenebilen tavuklar ve civcivlerden oluşur.
(2) Sürüde, en iyi uygunluk değerine sahip bireyler horoz olarak atanır ve atanan alt grupta lider olarak hareket ederler. En kötü uygunluk değerine sahip bireyler civciv olarak atanır, geriye kalan bireyler ise tavuk olarak belirlenir. Tavukların yaşayacağı alt grup rastgele seçilir. Anne tavuklar ve civcivler arasındaki ilişki de yine rastgele kurulur.
(3) Hiyerarşik düzen yine parametrik olarak belirlenen nesil sayısından sonra güncellenir.
(4) Her alt gruptaki tavuklar, yiyecek aramak için grup lideri olan horozlarını takip ederler. Bazı zamanlarda diğer tavukların yiyeceklerini çalma durumları yaşanabilir. Civcivlerin

diğer bireyler tarafından bulunan yiyecekleri rastgele çalabileceği ve her civcivin yiyecek aramak için annelerini takip ettiği varsayılmaktadır.

Genel kuraları verilen TSOA'nın öznitelik seçim işleminde de kullanılabileceği düşünülmüştür. Bu amaçla, TSOA'yı ROA ile destekleyerek bir yaklaşım önerilmiştir. Bu yaklaşım sayesinde sınıflandırma doğruluğunun artırılması için öznitelik seçim işleminde optimal çözüme ulaşılması hedeflenmektedir.




Şekil 3.3. TSOA ile gerçekleştirilen öznitelik seçim işlemi akış çizelgesi

TSOA ile gerçekleştirilen öznitelik seçim işlemi Şekil 3.3'de gösterildiği gibi genel olarak 3 ana fonksiyon üzerine kuruludur. Bu fonksiyonlardan ilki, orijinal veri kümesinden farklı alt kümeleri oluşturma işlemini kapsar. Oluşan alt kümeler ikinci ana fonksiyon olan değerlendirme işlemine tabi tutulur. Durdurma kriteri sağlanana kadar bu iki ana fonksiyon tekrar edilir. Durdurma kriteri, parametrik olarak önceden belirlenen uygunluk değerine ya da döngü sayısına ulaşılması olarak tasarıma göre farklılık gösterebilir. Durdurma kriteri sağlandığında son ana fonksiyon olan alt küme seçim işlemi gerçekleştirilir. Fonksiyonda oluşan tüm alt kümelerin en yüksek uygunluk değerine sahip olanı, çözüm olarak kabul edilir.

Alt küme değerlendirme işlemi, alt kümelerin uygunluk değeri atama ve uygunluk değerine göre çözüm sınıfına dahil edilmesi işlemine dayanır. Uygunluk değeri hesaplanmasında sınıflandırıcı metot olarak ROA kullanılmıştır. Oluşan alt kümeler, ROA'a girdi olarak verilmiştir. Sınıflandırma doğruluk değerleri ise her alt kümenin uygunluk değeri olarak atanır. Atanan değerler analiz edilir ve çözüm sınıfına ekleme işlemleri gerçekleştirilir. Alt küme değerlendirme işleminden sonra durdurma kriteri kontrol edilir. Kriter sağlanana kadar alt küme oluşturma ve değerlendirme işlemleri tekrar edilir. Kriter sağlandığında alt küme seçim işlemine geçilir. Bu çalışmada sonlandırma kriteri, nesil sayısı olarak belirlenmiştir. Son ana fonksiyon olan alt küme seçimi ise, çözüm sınıfında yer alan en yüksek uygunluk

değerine sahip alt küme seçim işlemini kapsar. Çıktı, nihai çözüm olan alt küme ve onun uygunluk değeridir.

Çalışmada, 3 analiz ile oluşturulmuş veri setleri ayrı ayrı TSOA ve ROA modelinin birleşimi ile öznitelik seçim işlemi uygulanmıştır. Her bir veri kümesi içerisinden, rastgele olarak seçilen öznitelikler ile sürüde ki canlı sayısı kadar farklı boyutlarda (öznitelik/sütun boyutu) alt veri kümeleri oluşturulmuştur. TSOA ile oluşturulan alt kümeler, ROA'ya girdi olarak verilmiştir. Sınıflandırma sonucu modellerin doğruluk değerleri uygunluk değeri olarak atanmıştır. Belirlenen uygunluk değerlerine göre gruplar oluşturulur. Her alt gruba lider olarak, sürüdeki en yüksek uygunluk değerine sahip alt veri kümesi, horoz olarak atanır. Tüm grup liderleri belirlendikten sonra uygunluk değeri en kötü alt veri kümeleri civciv olarak atanır. Kalan alt kümeler ise tavuk olarak belirlenir. Rastgele seçilen tavuklardan bazılarına yine rastgele seçilen civcivlerin annesi rolü verilir. Sürüdeki hiyerarşik düzen sağlandığında yiyecek aramak için konum değiştirme işlemleri gerçekleştirilir. Bu kısımda horozlar, tavuklar ve civcivlerin hareketlerinde gerçekleştirilen işlemler birbirinden farklıdır.

Horozlar için konum değiştirme işlemi Eş. 3.2'e göre gerçekleştirilir. Eş. 3.2, Eş. 3.1'e bağlı bir eşitliktir. En iyi uygunluk değerine sahip horozlar, yiyecek erişiminde önceliğe sahiptir. Bunun için uygunluk değerlerine göre daha geniş bir alanda yiyecek arayabilmesine imkan sağlanmıştır.

k horoz grubundan rastgele seçilen horozu temsil eder, $x_{i,j}^{t}$ t zamanında i. üyenin pozisyon bilgisidir. pfit uygunluk değeri ve ε=0,01 sıfıra bölme hatasını önlemek için eklenmiş sabit değerdir.

$$\epsilon = \begin{cases} 1 \text{ , if } \mathrm{pfit_i} \leq \mathrm{pfit_k} \text{ ,} \\ \exp\left(\dfrac{(\mathrm{pfit_k} - \mathrm{pfit_i})}{|\mathrm{pfit_i}| + \varepsilon}\right), \text{diğer durumlarda,} \end{cases} \quad k \in [1, N], k \neq i \tag{3.1}$$

$$x_{i,j}^{t+1} = x_{i,j}^{t} * \left(1 + \mathrm{Randn}(0, \epsilon)\right) \tag{3.2}$$

Tavuklarda yiyecek arama davranışı, genel olarak grup lideri horozu takip etme şeklindedir. Bazı durumlarda başka tavukların buldukları yiyecekleri yiyebilecekleri varsayılmaktadır. Uygunluk değeri yüksek olan tavuklar kendilerinden daha düşük uygunluk değerine sahip tavuklara göre yiyecek aramada daha avantajlı konumdadırlar.

r1 tavuğun bulunduğu grup lideri horozu temsil eder. r2 ise sürüden seçilmiş rastgele tavuk veya horozu temsil eder. Rand $\in (0,1)$ rasgele bir sayıdır.

$$S_1 = \exp\left(\frac{(\mathrm{pfit_i} - \mathrm{pfit_{r1}})}{\mathrm{abs}(\mathrm{pfit_i} + \varepsilon)}\right) \tag{3.3}$$

$$S_2 = \exp(\mathrm{pfit_{r2}} - \mathrm{pfit_i}) \tag{3.4}$$

$$x_{i,j}^{t+1} = x_{i,j}^{t} + S_1 * \mathrm{Rand} * \left(x_{r1,j}^{t} - x_{i,j}^{t}\right) + S_2 * \mathrm{Rand} * (x_{r2,j}^{t} - x_{i,j}^{t}) \tag{3.5}$$

İki tavuğun uygunluk değerleri arasındaki fark ne kadar büyükse, $S_2$ (bkz. Eş. 3.4) o kadar küçük ve pozisyonları arasındaki boşluk o kadar büyük olur. Böylece tavuklar, diğer tavukların bulduğu besini kolayca çalmazlar. $S_1$'in formül formunun (bkz. Eş. 3.3) $S_2$'ninkinden farklı olmasının nedeni, grupta yarışmaların mevcut olmasıdır. Basit olması için, tavukların horozun uygunluk değerine göre uygunluk değerleri, gruptaki tavuklar arasındaki yarışmalar olarak simüle edilir. $S_2 = 0$ varsayıldığı zaman tavuk kendi bölgesinde horozu takip ederek yiyecek arar. Tavukların hareketi Eş. 3.5'e göre gerçekleşir.

Civcivler yiyecek aramak için Eş. 3.6'da belirtildiği gibi annelerinin etrafında dolaşırlar. FL $\in (0,2)$ rastgele bir civcivin annesini ne kadar hızda takip edeceğini temsil eden parametredir ve mother $\in [1, N]$

$$x_{i,j}^{t+1} = x_{i,j}^{t} + \mathrm{FL} * (x_{\mathrm{mother},j}^{t} - x_{i,j}^{t}) \tag{3.6}$$

TSOA belirlenen G değeri kadar alt küme oluşturma ve değerlendirme işlemleri tekrar etmiştir. Çalışmada G değeri 32 olarak belirlenmiştir. Optimizasyon algoritmasının sonlandırma kriteri sağlandığında göre %100 sınıflandırma doğruluğuna en yakın doğruluk oranına sahip alt küme çözüm veri kümesi olarak seçilmiştir. TSOA ve ROA modelinin birleşimi ile öznitelik seçim işleminin başarısı seçilen veri kümesinin başarısına göre tartışılmıştır.

### 3.4. Makine Öğrenmesi

Makine öğrenmesi, bilgisayarları daha akıllı hale getirmeyi amaçlayan yapay zekanın bir alt dalıdır. Bilgisayarlara nasıl davranacaklarını açıkça öğretmeden, onlara insan müdahalesi

veya yardımı olmadan bir veri kümesi üzerinden otomatik olarak öğrenme yeteneği sağlar. Öğrenme işlemini verilerdeki kalıpları belirleyerek yapar. Öğrenme için veri kümesindeki mevcut örnek sayısı arttıkça başarımlarını uyarlamalı olarak iyileştirir.

Bu çalışmada, makine öğrenme yöntemi olarak TÖA kullanılmıştır. TÖA, Torbalama (bagging) ve Artırma (boosting) tekniklerini kapsar. Bu teknikler, birden fazla sınıflandırıcı metodu tek bir tahmin modelinde birleştiren meta yöntemlerdir. TÖA içerisinde kullanılacak sınıflandırıcı metod sayısı n ile ifade edilir. Bu çalışmada, hem Torbalama hem de Artırma tekniklerinde n değeri 10 olarak belirlenmiştir. İki teknik için de sınıflandırıcı model oluşturmada 3 farklı algoritma kullanılmıştır. Torbalama içerisinde; K en Yakın Komşu, Karar Ağacı ve Çok terimli Naive Bayes algoritmaları kullanılmıştır. Artırma yöntemi içinde de; Adaboost, XGBoost ve LightGBM algoritmaları kullanılmıştır. Tüm sınıflandırıcılarda, k-fold değeri 10 olarak belirlenerek çapraz doğrulama yapılmıştır.

### 3.4.1. Topluluk öğrenimi

Topluluk öğrenimi, klasik makine öğrenimi algoritmaları ile oluşan tekli sınıflandırıcı model kullanımının aksine birden fazla sınıflandırıcı model oluşturur. Değerlendirme işlemi, tüm sınıflandırıcı modellerden gelen sonuçların yorumlanıp, sunulması mantığına dayanır (Breiman, 1996; Zhu, 2020; An, 2020).

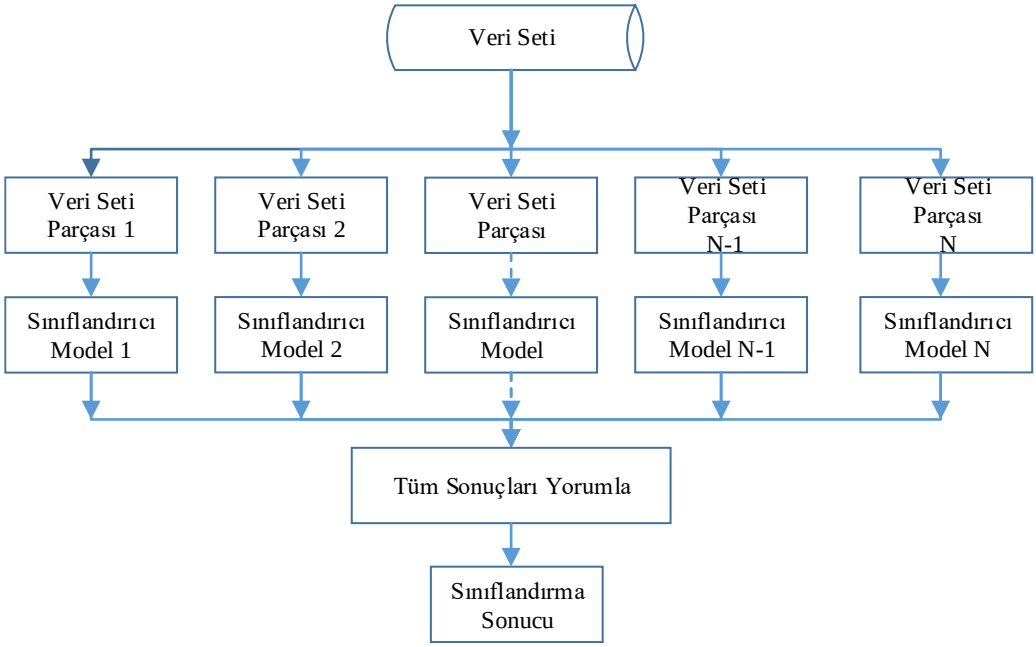
Topluluk öğrenme algoritmalarından Torbalama tekniği, (Breiman, 1996); izinsiz giriş tespiti, spam sınıflandırması, kredi puanlama vb. çeşitli gerçek problem senaryolarında uygulanan popüler topluluk öğrenme yaklaşımıdır (Agarwal, 2020). Bu teknik ile veri kümesi parçalara ayrılır ve her parça ayrı bir eğitim kümesi olarak temel sınıflandırıcılar ile modellenir. Test tüm modeller üzerinden gerçekleşir. Sınıflandırma sonucu, modellerden toplanan sınıflandırma sonuçlarının analiz edilmesi ile elde edilir. Sınıflandırma sonucu sayısal değer üzerinde sınıflandırma gerçekleşiyor ise Eş. 3.7, kategorik değerler üzerinde gerçekleşiyor ise Eş. 3.8'e göre gerçekleştirilir. Sayısal değer üzerinde sınıflandırma gerçekleşiyor ise çoklu sınıflandırma modellerinden gelen sayısal çıktı değerlerinin ortalaması alınarak sınıflandırma sonucu üretilir. Kategorik bir sonucu sınıflandırılırken ise, her sınıflandırıcı modellerden gelen kategorik değerler arasında en çok sınıflandırılan kategori sonuç olarak kabul edilir (Breiman, 1996).

$n \in [1, N]$ ve $M_n$: Ana veri kümesinden rastgele parça ile oluşturulan veri kümesini kullanılarak oluşturulan model.

$$M(x) = \frac{1}{N} + \sum_{n=1}^{N} (M_n(X)) \tag{3.7}$$

$$M(x) = \text{argmax}_y \sum_{n=1}^{N} (M_n(X) = y) \tag{3.8}$$

Torbalama tekniğinin çalışma prensibi Şekil 3.4.'te gösterilmiştir. Ana veri kümesi belirlenen parametre adedince rastgele parçalanır. Her veri kümesi parçası sınıflandırıcı modellerler eğitilir. Tüm modellerden toplanan sınıflandırma sonucu Eş. 3.7 veya Eş. 3.8'e göre yorumlanır.

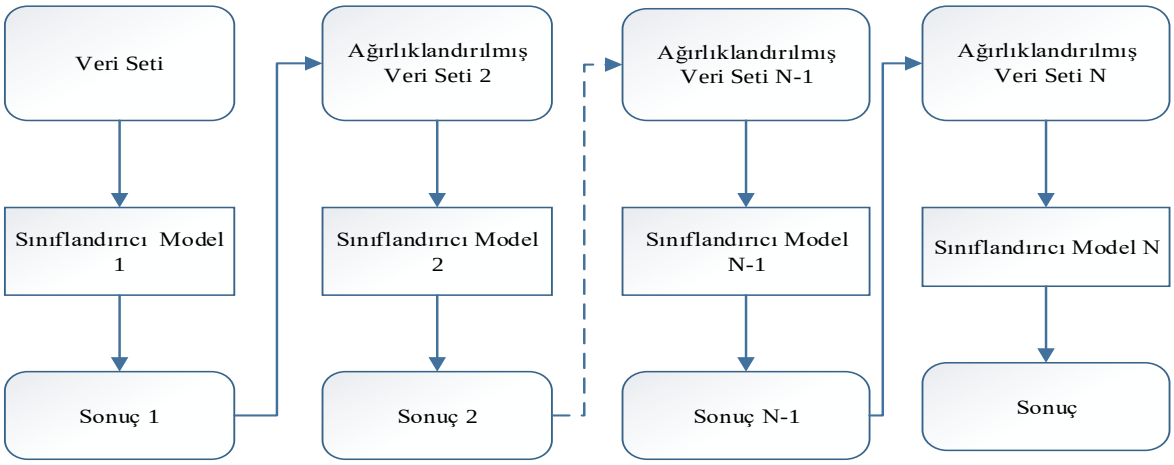



Şekil 3.4. Torbalama tekniğinin çalışma prensibi

Artırma (Boosting) tekniği, herhangi bir öğrenme algoritmasının performansını artırmak için geliştirilmiş bir TÖA yaklaşımıdır.

Şekil 3.5'te gösterildiği gibi zayıf öğrenme modelinin hatasını azaltmak için her iterasyonda, veri kümesi üzerinde ağırlıklandırma yapılır. Bu şekilde zayıf modellerin güçlendirilmesi amaçlanır. Güçlendirme, belirli bir zayıf öğrenme algoritmasını eğitim verileri üzerinden

çeşitli dağıtımlarda tekrar tekrar çalıştırarak ve ardından zayıf öğrenci tarafından üretilen sınıflandırıcıları tek bir bileşik sınıflandırıcıda birleştirerek çalışır (Quinlan, 1996).




Şekil 3.5. Artırma tekniğinin çalışma prensibi

### 3.4.2. Rastgele Orman Algoritması

Rastgele Orman Algoritması (Random Forest Algorithm) (ROA), Torbalama tekniğinin genişletilerek kullanıma sunulduğu bir TÖA'sıdır (Breiman, 1996). Sınıflandırma sürecinde birden fazla karar ağacı üretir ve karar verirken yine tüm ağaçlarla ürettiği modellerden gelen cevapları yorumlayarak sunar. Torbalama tekniği ile arasındaki fark, alt veri kümesi oluşturmada rastgele seçimin birleştirilmiş olmasıdır. Tek bir alt veri kümesinde en iyi özniteliği aramak yerine rastgele oluşturulmuş alt veri kümelerinde en iyiyi aramaya odaklanır. En büyük avantajı tekli sınıflandırıcılardaki aşırı öğrenme problemlerinin çok sayıda sınıflandırıcı modellerle önüne geçmesidir. Algoritmanın genel yapısının formüle edilmiş tanımı Eş. 3.9'da gösterilmiştir.

$$M_n(X, D_n) = E_s\,[M_n(X, S, D_n)] \tag{3.9}$$

ROA, yapısal olarak birçok regresyon ağacından oluşmaktadır. Eş. 3.9'da $E_s$, rastgele parametreye ilişkin beklentiyi gösterir. Bu beklenti, X'e ve $D_n$ veri kümesine bağlıdır. S değişkeni, rastgele olarak bölünecek koordinatın ve konumunun nasıl yapıldığını belirlemek için kullanılır. S değişkeni, X ve eğitim kümesi $D_n$'den bağımsızdır.

### 3.4.3. K En Yakın Komşu

K en yakın komşu (KEYK), gözetimli öğrenme kullanan en temel sınıflandırma yöntemlerden biridir. KEYK basitliği, etkinliği ve sağlamlığı nedeniyle yaygın olarak kullanılmaktadır. Verilerin sınıf dağılımı hakkında çok az bilginin olduğu sınıflandırma problemlerinde sıklıkla kullanılır. Olasılık hesaplamalarında parametrik tahmin yapılamasının zor oluğu durumlarda diskriminant analizi gerçekleştirmek için ortaya atılmıştır (Patrick ve Fischer, 1970). KEYK, iki örnek arasındaki farkı veya benzerliği ölçen bir mesafe ölçüm fonksiyonu aracılığıyla sınıflandırma işlemini gerçekleştirir. Örnekler arası mesafenin nasıl ölçüldüğü KEYK algoritmasının performansı için kritik öneme sahip noktalardan biridir. Genellikle Öklid mesafe fonksiyonu (Eş. 3.10) kullanılır ya da (Minkowski, Manhattan, Mahalanobis, Chebyschev, Dilca, vb.) bir başka mesafe ölçüm fonksiyonu da kullanılabilir.

$$d(x, y) = \sqrt{\sum_{i=0}^{n} (x_i - y_i)^2} \tag{3.10}$$

KEYK algoritmasında, komşu sayısı (k) parametresinin değerine dayalı olarak sınıflandırma yapılmaktadır. Sınıflandırma, doğru k parametresinin seçimine oldukça duyarlıdır. Test örneğinin sınıfını öngörmek için komşuların oylamasına başvurulur. Oylama için iki yaklaşım yaygındır.

Çoğunluk oyu (Majority voting): Bu yaklaşımda, tüm oylar eşittir. En çok oyu olan sınıfa atama yapılır.

Ağırlıklı oylama (Weighted voting) : Bu yaklaşımda, daha yakın komşular daha yüksek oy alır. En çok kullanılan ağırlık değeri atama, her bir komşunun ağırlığının, $1/d$ ya da $1/d^2$ şeklinde alınmasıdır. (d, komşuya olan mesafedir)

### 3.4.4. Karar Ağacı

Karar ağacı, gözetimli öğrenme kullanarak hem sayısal hem de kategorik verileri işleyebilen bir sınıflandırma algoritmasıdır. Karar ağacı öğrenmesi sırasında, öğrenilen bilgi bir ağaç

üzerinde modellenir. Karar ağacında her bir öznitelik bir düğüm tarafından temsil edilir. En üst yapı kök, en son yapı yaprak olarak adlandırılır. Düğümleri birbirine bağlayan yapılar ise dal olarak adlandırılır. Ağacın yaprakları seviyesinde sınıf etiketleri ve bu yapraklara giden ve başlangıçtan çıkan dallar ile de öznitelikler üzerindeki işlemler ifade edilmektedir. Tek bir kök düğümü ile başlanarak dallanmalar ile ağaç yapısı oluşturulur. Bir karar ağacındaki her düğümün yalnızca bir ebeveyn düğümü ve iki veya daha fazla alt düğümü vardır. Karar ağacı öğrenmesinde, öğrenme için kullanılan ve çok sayıda örnek içeren bir veri kümesi, bir dizi karar/kurallar uygulanarak daha küçük kümelere bölünür. Bu işlem, özyineli (recursive) olarak tekrarlanır ve tekrarlama işlemi sınıflama üzerinde bir etkisi kalmayana kadar sürer. (Friedl ve Brodley, 1997).

Karar ağaçları parametrik değildir ve girdi verilerinin dağılımlarına ilişkin varsayımlar gerektirmez. Ek olarak, öznitelikler ve sınıflar arasındaki doğrusal olmayan ilişkilerden etkilenmez, eksik değerlere izin verirler (K., 1998).

Bu çalışmada karar ağacı algoritması için kullanılan parametreler: max_depth=50, random_state=0

### 3.4.5. Çok Terimli Naive Bayes

Çok Terimli Naive Bayes (Naive Bayes Multinominal) (ÇTNB) sınıflandırıcı, bir Naive Bayes sınıflandırıcı varyantıdır (Xu, 2018). Hesaplama açısından çok verimli ve uygulaması kolay olan ÇTNB, metin sınıflandırma problemlerinde sıklıkla kullanılan bir öğrenme algoritmasıdır. ÇTNB, belirli bir terimin görünme sayısını veya frekansı temsil ettiği bir öznitelik vektörünü dikkate alır. Düşük maliyeti, büyük verilerde etkili sınıflandırma yeteneği ve kolay uygulanabilirliği açısından tercih edilir.

### 3.4.6. Adaboost

Adaboost, Artırma yöntemi ile zayıf öğrenme durumunu güçlü öğrenme durumuna geliştirmeye yönelik hazırlanan bir TÖA'sıdır (Schapire, 1999). Büyük veri kümelerine ihtiyaç duymayan bu algoritma, içeriğinde optimizasyon işlemlerini barındırır. Ayrıca performansı artırmak için farklı birçok makine öğrenme algoritması birlikte kullanılabilir.

Aşırı öğrenme problemine diğer öğrenme algoritmalarına göre daha az bağımlıdır. Bireysel modeller zayıf olabilir, ancak nihai model güçlü bir modele yakınsamaktadır. Model oluşturmada Bayes sınıflandırıcısını kullanır.

Bir eğitim örneği yanlış sınıflandırılırsa, bu eğitim örneğinin ağırlığı artırılır. Eşit olmayan yeni ağırlıklar kullanılarak ikinci bir sınıflandırıcı oluşturulur. Bu prosedür model sayısı kadar tekrar edilir. Her sınıflandırıcıya bir puan atanır. Son sınıflandırıcı, her aşamadaki sınıflandırıcıların doğrusal kombinasyonu olarak tanımlanır.

Bu çalışmada artırma yöntemi için kullanılan parametreler: max_depth=50, learning_rate=0.1, n_estimators=100, random_state=0

### 3.4.7. XGBoost

XGBoost, Extreme Grandient Boosting teriminin kısaltılmış halidir. Temeli Karar Ağacı ve Gradyan Artırma algoritmalarına dayanan bir TÖA'sıdır. Farklı derinlikte Karar Ağacı oluşturma ve optimize etmesi sayesinde hem sınıflandırma hem de regresyon problemlerinin çözümünde kullanılır. XGBoost gerçekleştirdiği optimizasyon teknikleri ile yerel bir makinede mevcut popüler çözümlerden on kat daha hızlı çalışır ve dağıtılmış / bellek sınırlı ayarda milyarlarca örneğe ölçeklenebilir. Paralel ve dağıtılmış bilgi işlem, öğrenmeyi hızlandırarak daha hızlı model keşfi gerçekleştirir. Genel olarak, iç düğümler öznitelik değerlerini belirtir ve yaprak düğüm puanları kararı gösterir. (Chen ve Guestrin, 2016).

Bu çalışmada XGBBoost yöntemi için kullanılan parametreler: learning_rate=0.1, max_depth=50, n_estimators=100, random_state=0

### 3.4.8. LightGBM

LightGBM, 2017 yılında geliştirilmiş ağaç tabanlı öğrenme algoritmalarını kullanan bir Gradyan Artırma algoritmasıdır (Ke ve diğerleri, 2017). Sınıflandırma, regresyon ve sıralama problemlerin çözümünde kullanılır. Seviye odaklı (level-wise or depth-wise) veya yaprak odaklı (leaf-wise) iki strateji kullanılabilir. Seviye odaklı stratejide ağaç büyürken ağacın dengesi korunur. Yaprak odaklı stratejide ise kaybı azaltan yapraklardan bölünme

işlemi devam eder. LightGBM bu özellikleri sayesinde diğer arttırma algoritmalarından ayrılmaktadır.

Üretilen model, yaprak odaklı strateji ile hızlı öğrenir ve daha az hata oranına sahip olur. Ancak veri sayısının az olduğu durumlarda bu strateji ile üretilen modellerde aşırı öğrenme gerçekleşebilir. Bu nedenle bu strateji, büyük verilerde kullanılmak için daha uygundur.

Bu çalışmada 3.4.8. LightGBM yöntemi için kullanılan parametreler: learning_rate=0.1, max_depth=50, n_estimators=100, n_jobs=-1, random_state=None

### 3.5. Performans Değerlendirme Ölçütleri

Makine öğrenme algoritmalarında model performansının değerlendirmesinde doğruluk, duyarlılık, kesinlik ve F-ölçütü sık kullanılan metriklerdir. Bu metrikler sınıflandırma sonucunda elde edilen hata matrisi kullanılarak çözümlenmektedir. ’de ikili sınıflandırmada kullanılan hata matrisi gösterilmektedir. Hata matrisinde, gerçek ve tahmin sınıfları için pozitif ve negatif etiketler bulunmaktadır. Gerçek sınıfta pozitif etikete sahip verilerin; tahmin sınıfında pozitif etikete sahip olması “Doğru Pozitif ” , tahmin sınıfında negatif etikete sahip olması “Yanlış Negatif” ile ifade edilir. Gerçek sınıfta negatif etikete sahip verilerin; tahmin sınıfında pozitif etikete sahip olması “Yanlış Pozitif ” , tahmin sınıfında negatif etikete sahip olması “Doğru Negatif” ile ifade edilir.

Doğruluk: Modelin doğru öngördüğü toplam örnek sayısının (gerçek durumu pozitifken öngörü sonucu da pozitif çıkan örneklerin sayısı + gerçek durumu negatifken öngörü sonucu da negatif çıkan örneklerin sayısı), veri kümesindeki toplam örnek sayısına oranıdır. (Eş. 3.11).

$$\text{Doğruluk (Accuracy)} = \frac{(\text{DP} + \text{DN})}{(\text{DP} + \text{DN} + \text{YP} + \text{YN})} \tag{3.11}$$

Kesinlik: Modelin doğru olarak öngördüğü pozitif örnek sayısının, öngörülen toplam pozitif örnek sayısına (gerçek durumu pozitifken öngörü sonucu da pozitif çıkan örneklerin sayısı + gerçek durumu negatifken öngörü sonucu pozitif çıkan örneklerin sayısı) oranıdır. (Eş. 3.12).

Çizelge 3.2. Hata Matrisi

| | | Tahmin Edilen Sınıf | |
|---|---|---|---|
| | | Pozitif | Negatif |
| Gerçek Sınıf | Pozitif | Doğru Pozitif (DP) | Yanlış Negatif (YN) |
| | Negatif | Yanlış Pozitif (YP) | Doğru Negatif (DN) |

$$\text{Kesinlik (Presicion)} = \frac{\text{DP}}{(\text{DP} + \text{YP})} \tag{3.12}$$

Duyarlılık: Modelin doğru öngördüğü pozitif örnek sayısının, toplam gerçek pozitif örnek sayısına (gerçek durumu pozitifken öngörü sonucu da pozitif çıkan örneklerin sayısı + gerçek durumu pozitifken öngörü sonucu negatif çıkan örneklerin sayısı) oranıdır. (Eş. 3.13).

$$\text{Duyarlılık (Recall )} = \frac{\text{DP}}{(\text{DP} + \text{YN})} \tag{3.13}$$

F-Ölçütü: Kesinlik ve duyarlılık değerlerinin harmonik ortalaması alınarak hesaplanır.(Eş. 3.14).

$$\text{F} - \text{Ölçütü (F} - \text{measure )} = \frac{2 * \text{Kesinlik} * \text{Duyarlılık}}{(\text{Kesinlik} + \text{Duyarlılık})} \tag{3.14}$$

### 3.6. Geliştirme Ortamı

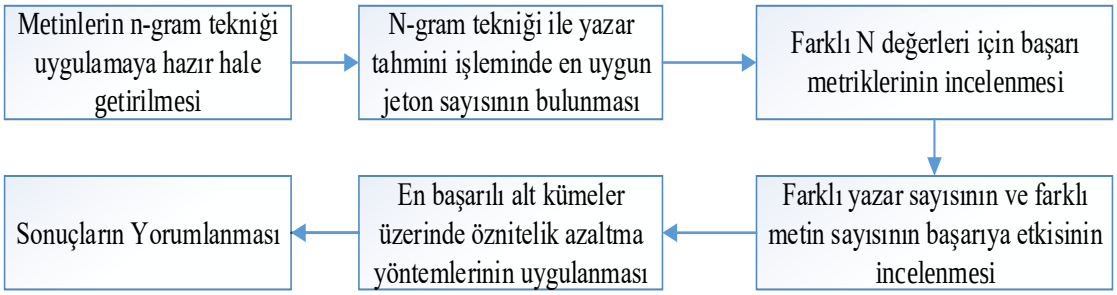
Bu çalışmada, tüm uygulamalar Python programlama dili ile gerçekleştirilmiştir. Tüm işlemler, 2.2 GHz İntel Core i7 işlemci ve 16,0 GB belleğe sahip bir bilgisayarda geliştirilmiştir.

# 4. METODOLOJİ VE DENEYSEL SONUÇLAR

Bu çalışma kapsamında, metinler üzerinden yazar tahmini için n-gram ve sözcüksel olmak üzere iki farklı analiz ile öznitelik çıkarımları gerçekleştirilmiştir. Analiz1; ngram tekniği ile öznitelik çıkarım işlemini, Analiz2; sözcüksel, yapısal ve sözdizimsel özellikleri içeren öznitelik çıkarımını kapsamaktadır. Karma Analiz ise iki analiz ile elde edilmiş özniteliklerin birleştirilmesi ile oluşturulmuştur.

n-gram tekniğinde iki farklı kaydırma metodu bulunmaktadır. Bu çalışma kapsamında her karakter başlangıç kabul edilerek gerçekleşen kaydırma metodu kullanılmıştır. Metin sınıflandırma işlemlerinde n-gram tekniği uygulanırken önemli parametrelerden biri n değeri, bir diğeri de jeton değeridir. Bu parametrelerin önemi doğrultusunda en başarılı n değerini elde etmek için 3 farklı n değerini kullanılarak alt kümeler oluşturulmuştur. Bir diğer önemli parametre olan jeton sayısını belirlemek için ise farklı senaryolar hazırlanmıştır. Çalışmada, en sık kullanılan jetonlar ile sınıflandırma işlemi gerçekleştirilmiştir. Senaryolar üzerinde gerçekleştirilen testlerle jeton sayısının en verimli olanını seçmek hedeflenmiştir.

N değeri ve jeton sayısının belirlenmesi için izlenen süreç Şekil 4.1.'de gösterilmiştir. n-gram tekniğini kullanmak için metinler üzerinde noktalama işaretleri, sayısal karakterler ve boşluk karakteri çıkarılmıştır. İlk aşamada n değeri 2 seçilerek hazırlanan 66 senaryo ile jeton değeri belirlenmiştir (Bkz. Çizelge 4.1). Belirlenen jeton değeri kullanılarak 54 farklı senaryo analiz edilerek en uygun n değeri belirlenmiştir (Bkz. Çizelge 4.2). Belirlenen jeton sayısı ve n değeri ile farklı sayıda yazar ve metin ile modeller oluşturulmuştur. En başarılı model üzerinde öznitelik seçme yöntemleri uygulanmıştır. Uygulanan yöntemlerle elde edilen en verimli modeller, yazar tahmini işleminde kullanılmıştır.




Şekil 4.1. Analiz1-n-gram tekniğinde izlenen süreç

En uygun jeton değerini belirlemek için hazırlanan senaryolarda, 5 ve 10 olarak 2 farklı yazar sayısı belirlenmiştir. Her yazar için kullanılan metin sayısı 100, 200, 500, 1000 ve 2000 olarak 5 farklı şekilde hazırlanmıştır. Derlemde 3000 metni olan 10 adet yazar olmadığından sadece 5 adet yazar için 3000 metin ile senaryo eklenmiştir. Yazar sayısı ve metin sayısı ile oluşturulan her kombinasyon için 6 farklı jeton değeri (50, 100, 200, 300, 400 ve 500) kullanılmıştır. Her senaryoda, yazarlar derlemden rastgele olarak seçilmiştir. Hazırlanan senaryolar için ROA ile eğitim ve test işlemi gerçekleştirilmiştir. Her senaryo için model ortalama doğruluk oranları Çizelge 4.1'de verilmiştir.

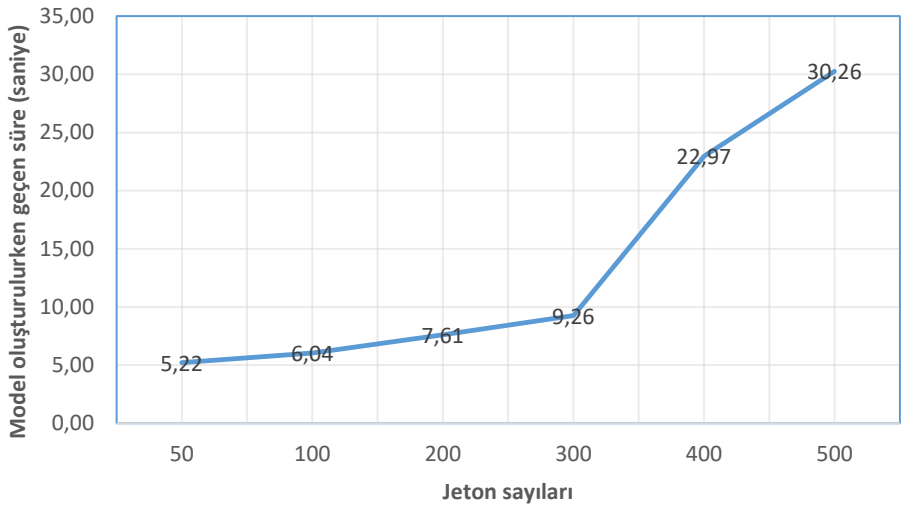
Çizelge 4.1 incelendiğinde genel olarak yazar sayısının 5 olduğu durumların, 10 olduğu durumlara göre daha yüksek doğruluk oranlarına sahip olduğu gözlemlenmiştir. Sınıflandırılacak yazar sayısı, iki katına çıkmış olmasına rağmen metin sayısı 100 seçildiğinde jeton sayısının 50'den yüksek olduğu her durum için doğruluk oranı birbirine yakındır. Yazar sayısının sabit kalıp, her yazara ait kullanılan metin sayısının artırılması genel olarak başarıyı artırmıştır. Her iki yazar grubu içinde, metin sayısının 200'den 500' e çıkarılması doğruluk oranının düşmesine yol açmıştır. Bunun her işlemde rastgele seçilen yazarlar ve onlara ait rastgele seçilen metinlerden kaynaklanabileceği düşünülmektedir. Genel olarak jeton değerleri arttığında doğruluk oranının arttığı gözlemlenmiştir.

Tüm jeton değerleri incelendiğinde, aynı yazar sayısı ve metin sayısına sahip durumları için 300, 400 ve 500 jeton değerlerinin kullanımında doğruluk oranının artışının çok yüksek olmadığı gözlemlenmektedir. Bu durumlara örnek olarak yazar sayısının 5, metin sayısının 3000 olduğu jeton sayısının 300, 400 ve 500 olduğu senaryolar verilebilir. 3 farklı jeton ile gerçekleştirilen bu durumda doğruluk oranı 300 jeton kullanımından 500'e çıkarıldığında %0,12 kadar yükselmiştir.

Çizelge 4.1. En uygun jeton değerini belirlemek için n değerinin 2 olarak sabit tutulduğu, farklı sayıda yazar ve farklı sayıda metinler ile hazırlanan senaryolarda 6 farklı jeton değeri için ROA kullanılarak hazırlanan modellerin ortalama doğruluk oranları (%)

| Yazar sayısı | Her bir yazar için metin sayısı | Jeton sayısı | | | | | |
|---|---|---|---|---|---|---|---|
| | | 50 | 100 | 200 | 300 | 400 | 500 |
| 5 | 100 | 61,6 | 61,6 | 63,2 | 65,4 | 64,2 | 65 |
| | 200 | 82,4 | 86,8 | 88,2 | 89,1 | 89,1 | 89,4 |
| | 500 | 80,72 | 84,52 | 86,32 | 87,48 | 87,84 | 87,48 |
| | 1000 | 83,94 | 88,32 | 90,98 | 91,62 | 91,56 | 91,92 |
| | 2000 | 88,77 | 91,35 | 93,26 | 93,38 | 93,46 | 94,08 |
| | 3000 | 93,43 | 95 | 96 | 96,06 | 96,14 | 96,18 |
| 10 | 100 | 58 | 62,9 | 64,9 | 65,6 | 65,5 | 66,9 |
| | 200 | 75,65 | 81,85 | 84,15 | 83,65 | 84,25 | 84,55 |
| | 500 | 72,64 | 80,72 | 82,78 | 83,28 | 83,66 | 83,9 |
| | 1000 | 77,61 | 82,73 | 85,83 | 86,6 | 86,93 | 87,33 |
| | 2000 | 76,92 | 82,04 | 83,96 | 85,78 | 86,27 | 86,86 |

Şekil 4.2.'de yazar sayısının 5, metin sayısının 3000 olduğu durumda farklı jeton kullanımları ile oluşan senaryolar için hazırlanan modellerin oluşturulma süreleri grafikte verilmiştir. Şekil 4.2. incelendiğinde doğruluk oranı artışının az olmasına rağmen 3 farklı jeton değeri ile gerçekleştirilen modellerin oluşturulmasında geçen süre 3,26 katı artış göstermiştir. Jeton sayısının 50 ile 300 olması arasında neredeyse 2 katı süre geçmesine rağmen doğruluk oranı %93,43'den %96,06'ya yükselmiştir. Jeton sayısının 50'den 500'e çıkarılmasında ise süre 5,79 kat artmış doğruluk oranı ise %93,43'den %96,18'ya yükselmiştir.




Şekil 4.2. Yazar sayısının 5 olduğu, her yazara ait 3000 metin ve 6 farklı jeton sayıları ile oluşturulan senaryolarda model geliştirme süreleri

Çizelge 4.1'de yer alan tüm senaryolar incelendiğinde doğruluk oranı ve model oluşturma süreleri birlikte incelendiğinde en verimli jeton sayısı değerinin 300 olduğu gözlemlenmiştir. Jeton sayısının 300 olarak belirlenmesinden sonra en uygun n değerini belirleme işlemi gerçekleştirilmiştir. n değerini belirlemek için 18 farklı durum hazırlanmıştır. Bu durumlar hazırlanırken jeton sayısı için sadece 300 değeri kullanılmıştır. 4 farklı yazar sayısı ve her yazar sayısı grubu için farklı sayılarda metinler kullanılmıştır. 18 farklı durum, n değerinin 2, 3 ve 4 olarak belirlenmesiyle toplamda 54 farklı senaryo oluşturulmuştur. Her senaryo için seçilen yazarlar ve o yazarlara ait metinler rastgele olarak seçilmiştir. 54 senaryo ayrı ayrı ROA ile sınıflandırılmış ve test edilmiştir. Her senaryo için oluşturulan modellerin doğruluk oranları Çizelge 4.2'te verilmiştir.

Çizelge 4.2'de modellerin doğruluk oranları incelendiğinde, en yüksek başarılı sınıflandırma %97,41 ile yazar sayısının 5, her yazar için kullanılan metin sayısının 3000 ve n değerinin 4 olduğu senaryo ile elde edilmiştir.

n değerlerine göre senaryoların ortalama doğruluk oranları;

- n=2 için %80,82
- n=3 için %81,86
- n=4 için %82,44

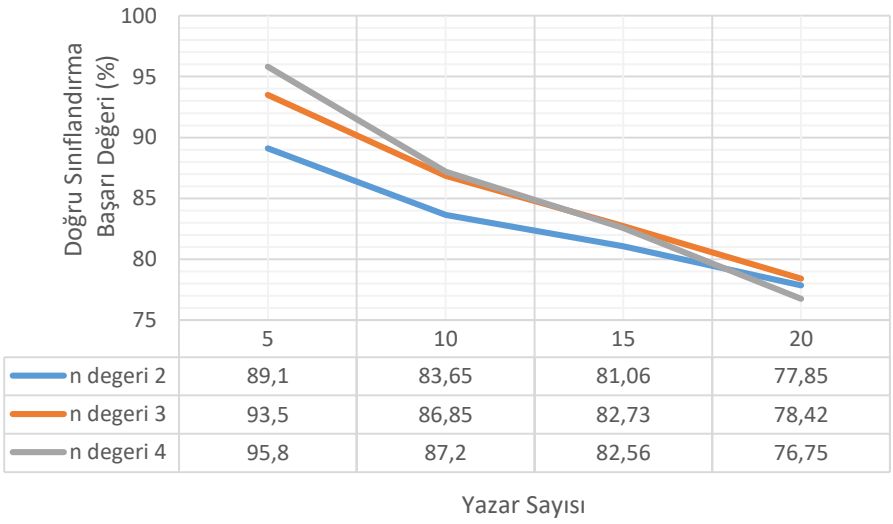
olarak elde edilmiştir.

Çizelge 4.2. n-gram tekniğinde en uygun n değerinin belirlenmesi için jeton sayısının 300 olarak kabul edildiği ve yazar sayısı, her yazar için kullanılacak metin sayısı, n değerlerinin farklı değerleri ile hazırlanan tüm senaryolar için ROA kullanılarak oluşturulan modellerinin doğruluk oranları (%)

| Yazar sayısı | Her bir yazar için metin sayısı | n değeri | | |
|---|---|---|---|---|
| | | 2 | 3 | 4 |
| 5 | 100 | 65,4 | 66 | 67,6 |
| | 200 | 89,1 | 93,5 | 95,8 |
| | 500 | 87,48 | 91,32 | 94,96 |
| | 1000 | 91,62 | 93,26 | 94,54 |
| | 2000 | 93,38 | 94,09 | 95,56 |
| | 3000 | 96,06 | 96,25 | 97,41 |
| 10 | 100 | 65,6 | 67,6 | 66,6 |
| | 200 | 83,65 | 86,85 | 87,2 |
| | 500 | 83,28 | 84,28 | 84,8 |
| | 1000 | 86,6 | 87,61 | 86,79 |
| | 2000 | 85,78 | 84,53 | 85,33 |
| 15 | 100 | 68,66 | 69,6 | 70,13 |
| | 200 | 81,06 | 82,73 | 82,56 |
| | 500 | 78,97 | 79,61 | 79,18 |
| | 1000 | 75,7 | 74,3 | 74,28 |
| 20 | 100 | 70,1 | 69,85 | 70,4 |
| | 200 | 77,85 | 78,42 | 76,75 |
| | 500 | 74,51 | 73,68 | 74,2 |

Bu çalışmada, metin sayısının 100 olduğu her senaryo (yazar sayısının 5, 10, 15 ve 20 olması durumu ve n değerinin 2, 3 ve 4 olduğu tüm senaryolar) için doğruluk oranları %65,4 ile %70,4 arasında birbirine yakın değerlere sahiptir. Metin sayısı 100 olduğunda en yüksek doğruluk oranı, yazar sayısının 20 olduğu durumda elde edilmiştir. Şekil 4.3.'de her n değeri için metin sayısının 200 olarak sabit tutulduğu senaryolarda, yazar sayısı değişiminin doğruluk oranlarına etkisi gösterilmiştir.

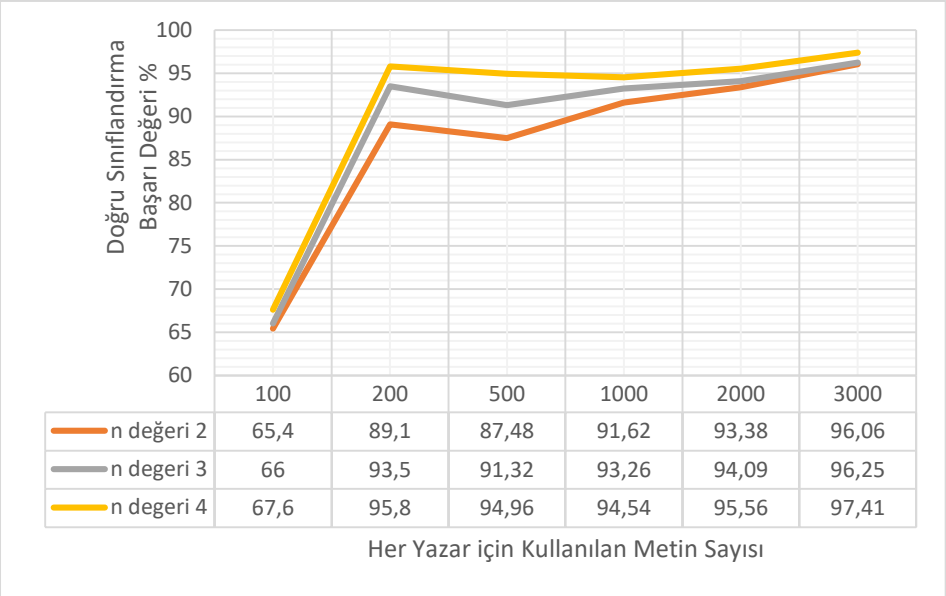
Çizelge 4.2 ve Şekil 4.3. incelendiğinde metin sayısının sabit olup yazar sayısının arttığı durumlarda genel olarak doğruluk oranı düşmüştür. Sınıf sayısının arttığı durumlarda doğruluk oranının düşmesi genel bir sınıflandırma problemidir. Bu probleme çözüm önerisi olarak her sınıfa ait örnek sayısının artırılması düşünülmüştür. Yazar sayısındaki artış ile en yüksek doğruluk oranı %95,8'den %77,85'e düşmüştür. Yazar sayısındaki 4 farklı durumun 2 sinde n değerinin 4, diğer ikisinde n değerinin 3 olduğu durumda en yüksek doğruluk oranı elde edilmiştir. n değerinin 2 olması bu senaryolarda en düşük doğruluk oranına sahip olarak sıralamada geriye düşürmüştür.




| | 5 | 10 | 15 | 20 |
|---|---|---|---|---|
| n degeri 2 | 89,1 | 83,65 | 81,06 | 77,85 |
| n degeri 3 | 93,5 | 86,85 | 82,73 | 78,42 |
| n degeri 4 | 95,8 | 87,2 | 82,56 | 76,75 |

Şekil 4.3. En uygun n değeri hesaplamasında oluşturulan her yazara ait 200 adet metnin kullanıldığı, farklı yazar ve farklı n değeri ile oluşturulan modellerin, sınıflandırma doğruluğunun yazar sayısına göre değişimini

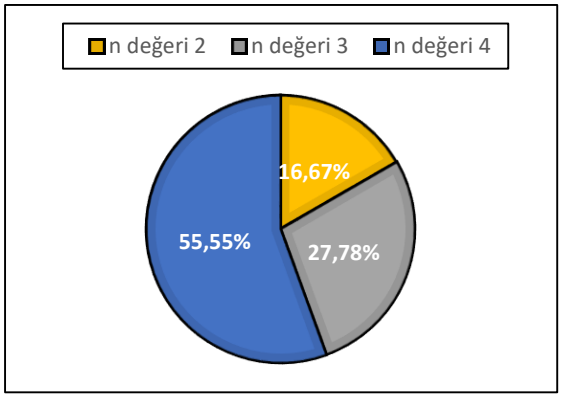
Yazarlara ait örnek sayısının artması (metin sayısının artması) sınıflandırma doğruluğunu genel olarak artırmıştır. Şekil 4.4.'da yazar sayısının 5 ile sabit tutulduğu durumda metin sayısındaki ve n değerindeki değişimin sınıflandırma doğruluğuna etkisi gösterilmektedir.




Şekil 4.4. En uygun n değeri hesaplamak için oluşturulan modellerin yazar sayısı 5 olarak sabit tutulduğunda metin sayıları değişiminin ve değişen n değerlerinin doğruluk oranı üzerindeki etkisi

Şekil 4.4. incelendiğinde sınıflandırma doğruluğu metin sayısının artışı ile birlikte yükselmektedir. Sadece metin sayısının 500 olduğu durumda genel olarak tüm n değerleri için genel bir düşüş gözlemlenmiştir. Bunun bilinen bir nedeni bulunmamaktadır. Her senaryo için seçilen metin ve yazarların rastgele seçilmesi sınıflandırma doğruluğunda bu tarz sonuçlara yol açabileceği düşünülmektedir.

Yazar sayısının 5 olduğu her senaryo için doğruluk oranları, n değerinin 4 olduğu durumda en yüksektir. Yazar sayısının 5 değeri dışındaki her senaryoda, sınıflandırma doğruluğunu en yüksek veren bir n değeri bulunmamaktadır. 18 senaryo üzerinde n değerine göre en yüksek doğruluk oranlarının dağılımları Şekil 4.5.'da gösterilmiştir.



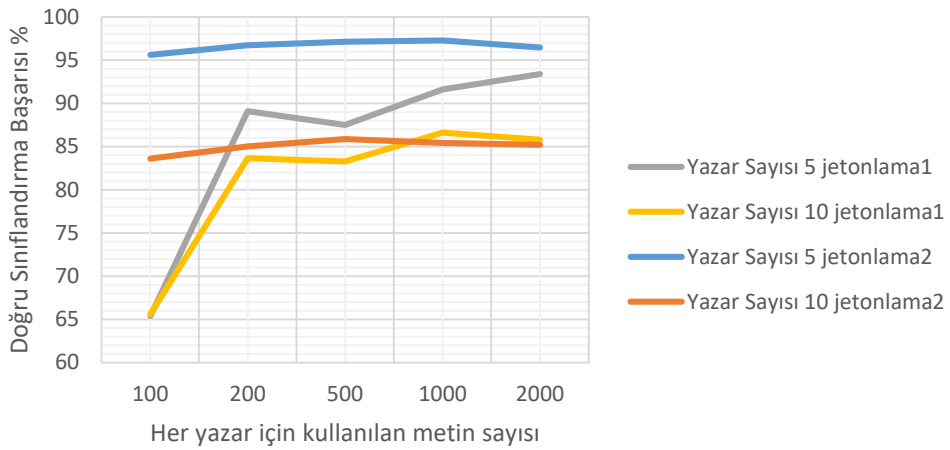

Şekil 4.5. Farklı yazar ve metin sayıları ile oluşturulan 18 senaryo üzerinde en yüksek sınıflandırma doğruluğunun n değeri üzerindeki dağılımları

18 senaryo içerisinde, 10 senaryoda en yüksek doğruluk oranlarına n değerinin 4 olduğu durumda ulaşılmıştır. Çizelge 4.2 incelendiğinde %97,41 ile en yüksek doğruluk yine n değerinin 4 olduğu durumda elde edilmiştir. Her n için 18 senaryo genelinde ortalama doğruluk oranları incelendiğinde %82,44 ile en yüksek ortalama sınıflandırma doğruluğuna yine n değerinin 4 olduğu durumda ulaşılmıştır. Şekil 4.5.'te gösterildiği gibi hazırlanan senaryoların %55,55'inde n değerinin 4 olduğu durumda en yüksek doğruluk oranına ulaşılmıştır. Bu analiz doğrultusunda çalışmanın geri kalan kısmında n değerinin 4 olarak kullanılmasına karar verilmiştir.

n değerinin 4 olarak seçilmesinin ardından yazar sayısı ve metin sayıları üzerinde analizler gerçekleştirilmiştir. En az örnek miktarını kullanarak en verimli yazar belirleme yöntemini belirlemek için 5 ve 10 yazar olarak iki küme gruplandırıldı. Her grupta iki farklı jeton üretme ve örnek sayısının değişiminin ROA kullanılarak gerçekleştirilen sınıflandırma doğruluğu üzerindeki etkisi incelenmiştir. Farklı jeton tanımlama ile amaç, sınırlı miktarda materyale sahip yazarları, farklı boyutlarda materyali hazır olanlara ölçeklendirilebilmektir. Bu şekilde verinin en az miktarı kullanarak, en verimli tanımlama yöntemine odaklanarak yazar tanıma gerçekleştirmektir. Bu amaç doğrultusunda iki jeton üretme tekniği kullanılmıştır. Bunlar;

- Jetonlama1: Sadece sınıflandırmada kullanılacak metinlerin kullanılması ile üretilen 300 jeton
- Jetonlama2: Derlemdeki tüm metinler kullanılarak üretilen 300 jeton

Yazar sayısının 5 ve 10 değeri, her yazar grubu için 5 farklı metin sayısı ve her durum için iki farklı jetonlama tekniği ile ROA kullanılarak modeller geliştirilmiştir. Modellerin doğruluk oranları Şekil 4.6.'de gösterilmiştir.




Şekil 4.6. Yazar sayısı 5 ve 10 olduğu durumda sınıflandırmada kullanılacak metin sayısı değişiminin ve farklı jeton değerlerinin sınıflandırma doğruluğu üzerindeki etkisi

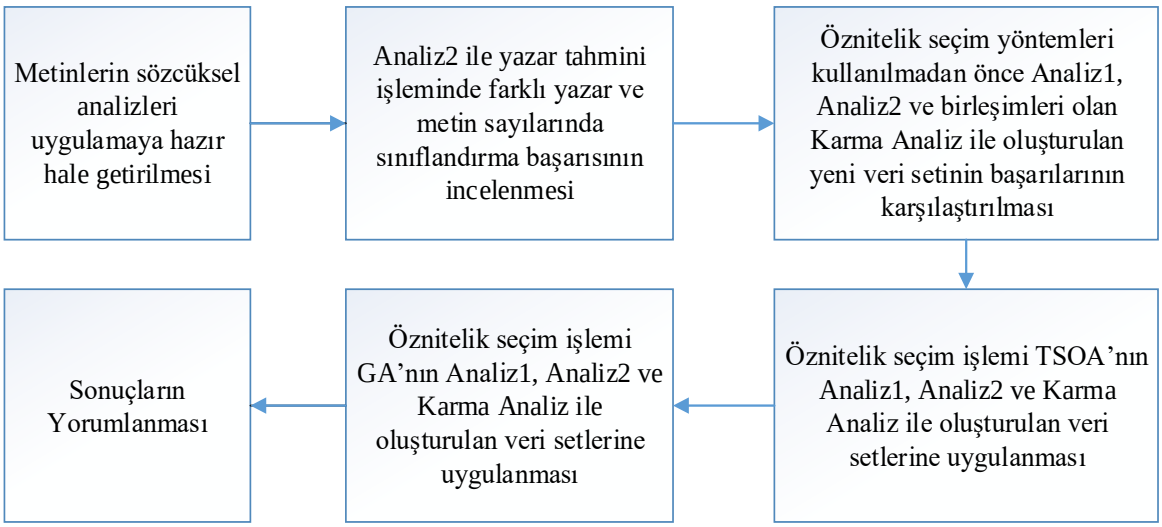
Oluşturulan 20 senaryo incelendiğinde, jetonlama2 kullanımında metin sayısının değişiminin doğruluk üzerinde yüksek oranda bir etkisi olmadığı tespit edilmiştir. Metin sayısından etkilenmese de sınıflandırılacak yazar sayısındaki değişim, doğrudan sınıflandırma doğruluğu üzerinde değişime neden olmuştur. Yazar sayısının aynı olduğu durumlarda metin sayısının 10 kat artırılmasına rağmen sınıflandırma doğruluğu üzerinde en fazla %1,68 artış sağlanmıştır. En yüksek doğruluk oranına sahip olan 5 yazarlı kümenin, doğruluk oranlarına 10 yazarlı küme ile ulaşmak için metin sayısında bir artış yapmanın etkisinin olmayacağı düşünülmektedir.

Jetonlama1 ile gerçekleştirilen sınıflandırma işleminde ise metin sayılarının değişiminin sınıflandırma doğruluğu üzerinde etkisi görülebilmektedir. Özellikle yazar sayısının 5 olduğu durumda metin sayısının 100'den 200'e yükseltilmesinde, doğruluk oranı %65,4'den %89,1'e kadar artmıştır.

En yüksek doğruluk oranlarına yazar sayısının 5 ve jetonlama2 seçildiğinde erişilebilmektedir. Burada amaç, en az veri miktarını kullanarak en verimli yazar tanımlama yöntemini belirlemektir. Şekil 4.6.'de belirtilen değerler incelendiğinde en verimli olan yöntemin, derlemdeki tüm metinler kullanılarak üretilen jetonlama2 olduğu sonucuna

ulaşılmıştır. Jetonlama2 de yer alan, tahmin edilecek yazar sayısındaki artışın başarı üzerindeki negatif etkisini azaltmaya yönelik öznitelikler üzerinde seçim işlemi gerçekleştirilmiştir. Analiz1 için jeton sayısı, jetonlama tekniği ve n değeri parametreleri belirlenmiştir.

Analiz1 için gerçekleştirilen süreçlerin ardından çalışmanın ilerleyişi Şekil 4.7.'de gösterilmiştir.




Şekil 4.7. Yazar tahmini problemi çözümünde Analiz2, Analiz1 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde gerçekleştirilecek işlemlerde izlenecek süreç

Öznitelik seçim işleminden önce Analiz1, Analiz2 ve Karma Analizin sınıflandırma doğruluklarını tartışmak üzere 18 farklı durum hazırlanmıştır. Hazırlanan durumlarda, 4 farklı yazar grubu (5, 10, 15, 20) ve onlara ait değişken sayıda metinler bulunmaktadır. Her durum üzerinde sırasıyla Analiz1, Analiz2 ve Karma Analiz gerçekleştirilmiştir. Analizlere ait özet bilgiler;

- Analiz1: n-gram yöntemi ile oluşturulan veri kümesinde n değeri4 ve jeton miktarı 300 kullanılmıştır. Öznitelik sayısı, yazar bilgisi hariç 300'dür.
- Analiz2: Sözcüksel, yapısal ve sözdizilimsel özniteliklerin yer aldığı analizde yazar bilgisi haricinde toplamda 40 öznitelik bulunmaktadır.

- Karma Analiz: Öznitelik seçim işlemi yapılmadan önce gerçekleştirilen iki analiz (Analiz1 ve Analiz2) yönteminin birleştirilmesi ile oluşturulmuştur. Sınıf bilgisi haricinde toplam öznitelik sayısı 340'dır.

Analizler sonucu oluşturulan veri setleri ROA ile sınıflandırılmış ve modellerin doğruluk değerleri oranları Çizelge 4.3'te gösterilmiştir. Toplam 54 adet senaryo birbiri ile karşılaştırılmak üzere sunulmuştur. Senaryoların fazla olması, farklı analizlerle oluşturulan veri setleri arasındaki başarı sınıflandırma doğruluklarının değişiminin daha objektif olarak tartışılmasını sağlamıştır. Çizelge 4.3'te Analiz2'e ait modellerinin, Analiz1'e ait olan modellere göre doğru sınıflandırma başarısının doğruluğunun daha yüksek olduğu gözlemlenmektedir. Özellikle her sınıfa ait örnek sayısının az olduğu durumlarda Analiz1'e ait modellerin başarısı doğruluğu düşük iken Analiz2'ye ait modellerin başarı değerleridoğruluk oranları yüksektir. Yazar sayısının 5 ve her yazara ait metin sayısının 3000 olduğu durumda Analiz1'e ait modellerin daha başarılı doğru sonuçlar ürettiği gözlemlenmiştir. Genel olarak tüm durumlar için Karma Analiz'e ait modellerin başarısı doğruluğu en yüksektir.

Karma Analiz ile oluşturulan modellerin, yazar sayısı ve metin sayısındaki değişikliklerle hazırlanmış 18 durum için ortalama doğruluk oranı %94,1 olarak tespit edilmiştir. Analiz2 ile bu oran %89,85, Analiz1 için %85,12 oranında kalmıştır. Karma Analiz'e ait modellerin en düşük doğruluk oranı %87,17 ile diğer analizlerle oluşturulan modellerin ortalama doğruluk değerine oranına dae yakındır. Hem en yüksek doğruluk oranı %98,92 ile hem en düşük doğru oranı tartışıldığında en verimli veri kümesiseti oluşturma yönteminin Karma Analiz olduğu gözlemlenmiştir.

Tüm sonuçlar incelendiğinde yazar sayısının sabit olduğu metin sayısının artırıldığı durumlarda genel olarak başarı doğruluk oranında çok büyük değişimler gözlemlenmemiştir. Bu durum az örnek sayısının olduğu durumlarda yazar tahmini probleminin çözümündede kullanılabilirliğine olan inancı artırmıştır. Yazar tahmini probleminde kullanılabilirliği artırmak amacıyla, başarıyı sınıflandırma doğruluğunu artırıp model oluşturma ve test işlemlerinin süresini azaltmaya yönelik öznitelik seçim işlemleri uygulanmıştır.

Çizelge 4.3. Yazar sayısı ve metin sayısının farklı kombinasyonları ile hazırlanmış toplam 18 durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulmuş veri setlerinin ROA ile gerçekleştirilen modellerin doğruluk oranları (%)

| Yazar sayısı | Her bir yazar için metin Sayısı | Analiz1 ile oluşturulan veri kümesi | Analiz2 ile oluşturulan veri kümesi | Karma Analiz ile oluşturulan veri kümesi |
|---|---|---|---|---|
| 5 | 100 | 95,6 | 95,6 | 97,6 |
| | 200 | 96,7 | 97,2 | 98 |
| | 500 | 97,12 | 97,9 | 98,92 |
| | 1000 | 97,28 | 97,4 | 98,28 |
| | 2000 | 96,46 | 96,8 | 98,23 |
| | 3000 | 96,46 | 96,82 | 98,233 |
| 10 | 100 | 83,6 | 91,2 | 92,23 |
| | 200 | 85 | 92,95 | 93,35 |
| | 500 | 85,86 | 94,84 | 94,46 |
| | 1000 | 85,42 | 93,74 | 93,42 |
| | 2000 | 85,2 | 93,37 | 93,68 |
| 15 | 100 | 70,13 | 77,26 | 92,46 |
| | 200 | 82,56 | 82,5 | 92,86 |
| | 500 | 79,18 | 85,05 | 94,32 |
| | 1000 | 74,28 | 86,28 | 88,14 |
| 20 | 100 | 70,4 | 76,2 | 88,8 |
| | 200 | 76,75 | 80,12 | 93,75 |
| | 500 | 74,2 | 82,14 | 87,17 |

Öznitelik seçim işlemlerinde GA ve ROA birleşimi ve TSOA ve ROA birleşimi olmak üzere 2 farklı yöntem kullanılmıştır. İlk olarak TSOA ve ROA birleşimi için Analiz1, Analiz2 ve Karma Analiz ile veri setleri oluşturulmuştur. Yazar sayısı 10 ile sabit tutulmuş 5 farklı boyutta veri kullanılmıştır. Öznitelik seçim işleminin sınıflandırma doğruluğuna etkisini detaylı incelemek için kullanılan tüm veri setlerine ait 4 adet bilgi verilmiştir. Bu bilgiler;

- Seçim işleminden önce elde edilen doğruluk oranı
- Seçim işleminden sonra elde edilen doğruluk oranı
- Seçilen öznitelik sayıları
- Model oluşturma süresi

Öznitelik seçim işleminde ilk olarak TSOA algoritması uygulanmıştır. Her analize ait uygulama sonucu Çizelge 4.4, Çizelge 4.5 ve Çizelge 4.6'da sunulmuştur.

- Analiz1 ile oluşturulan veri kümeleri üzerindeki değişim Çizelge 4.4
- Analiz1 ile oluşturulan veri kümeleri üzerindeki değişim Çizelge 4.5
- İki analizin birleştirilmesi ile hazırlanmış Karma Analiz ile oluşturulan veri kümeleri üzerindeki değişim Çizelge 4.6

Çizelge 4.4'e göre Analiz1 üzerinde TSOA kullanımı ile gerçekleştirilen işlem sonrasında 5 durum içinde öznitelik sayısında yüksek azalış gözlemlenmektedir. Sınıflandırma doğruluk oranları öznitelik azaltma işleminden önce ve sonrası için en büyük fark olan %1,36 ile metin sayısının 500 olduğu durumda yaşanmıştır. 5 farklı boyutta gerçekleştirilen karşılaştırmanın 3'ü için öznitelik seçim işleminde doğruluk oranı yükselmiştir. Tüm durumlar için model oluşturma sürelerinde öznitelik seçim işlemi sonrasında düşüş yaşanmıştır.

Çizelge 4.5' e göre Analiz2 ile oluşturulan veri kümeleri üzerinde TSOA ile gerçekleştirilen 5 işlemden sadece metin sayısının 500 olduğu durumda doğruluk oranında %0,42'lik bir düşüş yaşanmıştır. Öznitelik seçim işlemi sonrasında ilk durumdaki öznitelik sayısında en az %30 oranında azalma gerçekleşmiştir. Model oluşturma süresi, öznitelik seçim işlemi sonrasında daha kısa sürede gerçekleşmiştir.

Çizelge 4.4. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz1 ile oluşturulan veri kümesi üzerinde TSOA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranları, model oluşturma süreleri ve 300 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri gösterilmiştir

| Her bir yazar için metin sayısı | TSOA'dan önce model doğruluk oranı (%) | TSOA'dan sonra model doğruluk oranı (%) | TSOA'dan önce model oluşturma süresi (sn) | TSOA'dan sonra model oluşturma süresi (sn) | TSOA'dan sonra oluşturulan veri kümesi öznitelik sayısı |
|---|---|---|---|---|---|
| 100 | 83,6 | 83,7 | 0,92 | 0,7 | 143 |
| 200 | 85 | 85,65 | 1,65 | 1,15 | 160 |
| 500 | 85,86 | 84,5 | 4,45 | 3,47 | 163 |
| 1000 | 85,42 | 84,97 | 15,45 | 7,93 | 173 |
| 2000 | 85,2 | 85,37 | 26,77 | 16,95 | 171 |

Hem Analiz1 hem de Analiz2 ile oluşturulan veri setleri için ayrı ayrı 10 yazara ait 5 farklı durum için öznitelik seçim işlemi uygulanarak sınıflandırma işlemi gerçekleştirilmiştir. Sonuçlar Çizelge 4.4 ve Çizelge 4.5 gösterilmiştir. Karma Analiz ile oluşturulan veri setleri; n-gram tekniğinden 300, sözcüksel, yapısal ve sözdizilimsel özellikleri içeren veri kümesinden 40 öznitelik ile toplamda 340 özniteliğe sahiptir. Karma Analiz ile hazırlanan tüm veri kümeleri üzerinde TSOA kullanılarak öznitelik seçim işlemi gerçekleştirilmiştir. Sonuçlar Çizelge 4.6'te yer almaktadır.

Çizelge 4.5. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz2 ile oluşturulan veri kümesi üzerinde TSOA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranları, model oluşturma süreleri ve 40 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri gösterilmiştir

| Her bir yazar için metin sayısı | TSOA'dan önce model doğruluk oranı (%) | TSOA'dan sonra model doğruluk oranı (%) | TSOA'dan önce model oluşturma süresi (sn) | TSOA'dan sonra model oluşturma süresi (sn) | TSOA'dan sonra oluşturulan veri kümesi öznitelik sayısı |
|---|---|---|---|---|---|
| 100 | 91,2 | 91,6 | 0,41 | 0,28 | 23 |
| 200 | 92,95 | 92,8 | 0,84 | 0,61 | 23 |
| 500 | 94,84 | 94,42 | 1,68 | 1,36 | 21 |
| 1000 | 93,74 | 93,82 | 4 | 3,23 | 20 |
| 2000 | 93,37 | 93,68 | 9,54 | 7,34 | 28 |

Çizelge 4.6'ya göre 10 yazara ait 5 farklı durumun sadece metin sayısının 100 olduğu en küçük boyuttaki veri kümesi ile gerçekleştirilen işlemde doğruluk oranı, öznitelik seçim işlemi sonrasında düşmüştür. Öznitelik sayısı 340 iken 5 durum için neredeyse yarı yarıya düşüş sağlanmıştır. Öznitelik sayısının azalması ile birlikte tüm durumlar için model oluşturma süresi de azalmıştır.

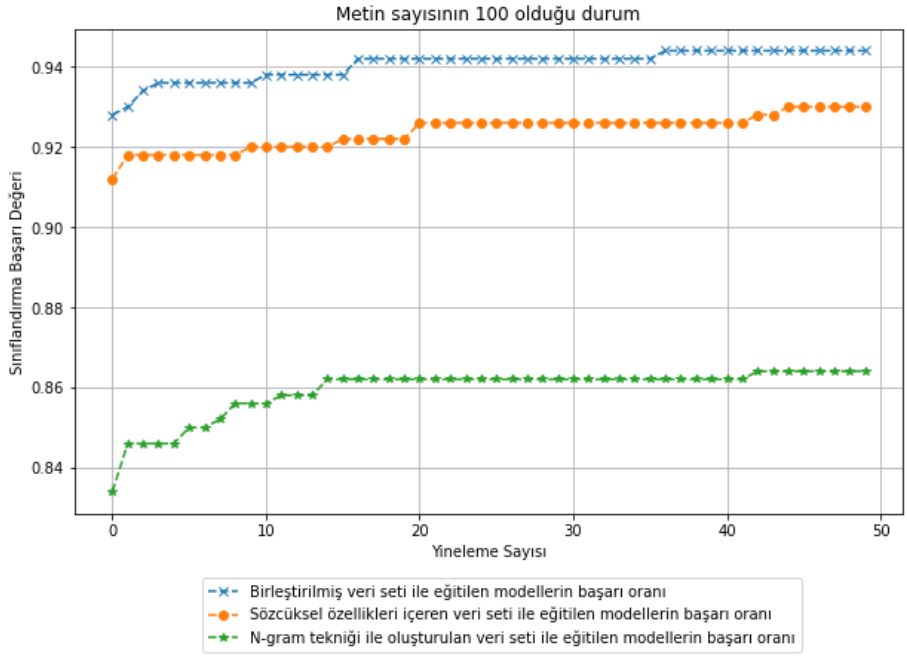
TSOA ile gerçekleştirilen uygulama sonrasında elde edilen Çizelge 4.4, Çizelge 4.5 ve Çizelge 4.6 incelendiğinde daha az veri ile daha doğru yazar tanıma işleminin yapılabileceğini gözlemlenmiştir. En iyi sonuç, Karma Analiz ile oluşturulan veri kümesi üzerinde gerçekleştirilen öznitelik azaltma işleminde her yazar için 500 metin içeren durumda elde edilmiştir.

TSOA ile gerçekleştirilen öznitelik seçim işleminden sonra, ilk veri kümeleri üzerinde ikinci öznitelik seçim yöntemi olan GA ile işlemler gerçekleştirilmiştir. Bu sayede iki optimizasyon tekniğinin sınıflandırma doğruluğuna etkisi de incelenmiştir. TSOA'da olduğu gibi yazar sayısının 10 olduğu ve 5 farklı boyutta hazırlanan 3 tip veri kümesi GA'ya girdi

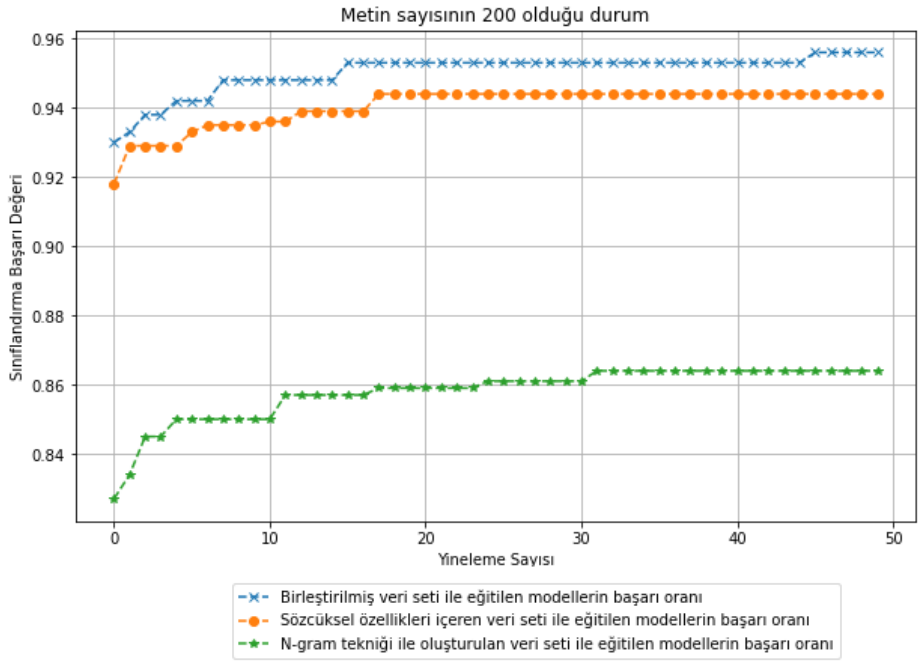
olarak verilmiştir. GA'da her yinelemede oluşan doğruluk oranı üzerindeki değişimler Şekil 4.8., Şekil 4.9., Şekil 4.10., Şekil 4.11. ve Şekil 4.12. gösterilmiştir.

Çizelge 4.6. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Karma Analiz ile oluşturulan veri kümesi üzerinde TSOA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranları, model oluşturma süreleri ve 340 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri gösterilmiştir

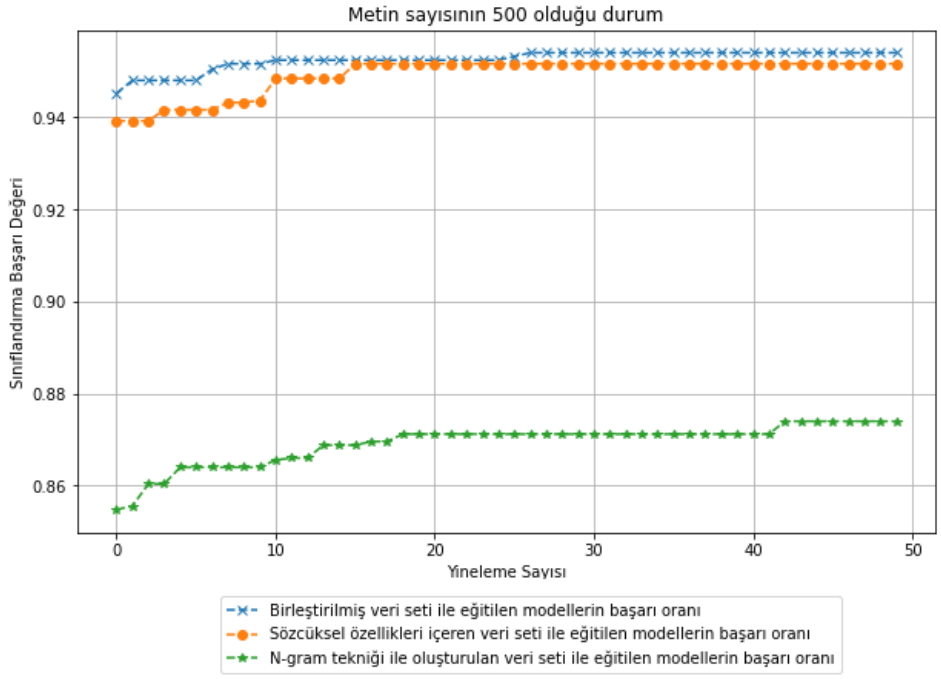
| Her bir yazar için metin sayısı | TSOA'dan önce model doğruluk oranı (%) | TSOA'dan sonra model doğruluk oranı (%) | TSOA'dan önce model oluşturma süresi (sn) | TSOA'dan sonra model oluşturma süresi (sn) | TSOA'dan sonra oluşturulan veri kümesi öznitelik sayısı |
|---|---|---|---|---|---|
| 100 | 92,23 | 91,7 | 0,64 | 0,52 | 157 |
| 200 | 93,35 | 94,5 | 1,9 | 1,06 | 181 |
| 500 | 94,46 | 95,06 | 3,58 | 2,93 | 162 |
| 1000 | 93,42 | 94,49 | 8,87 | 6,83 | 178 |
| 2000 | 93,68 | 94,9 | 23,62 | 16,03 | 175 |



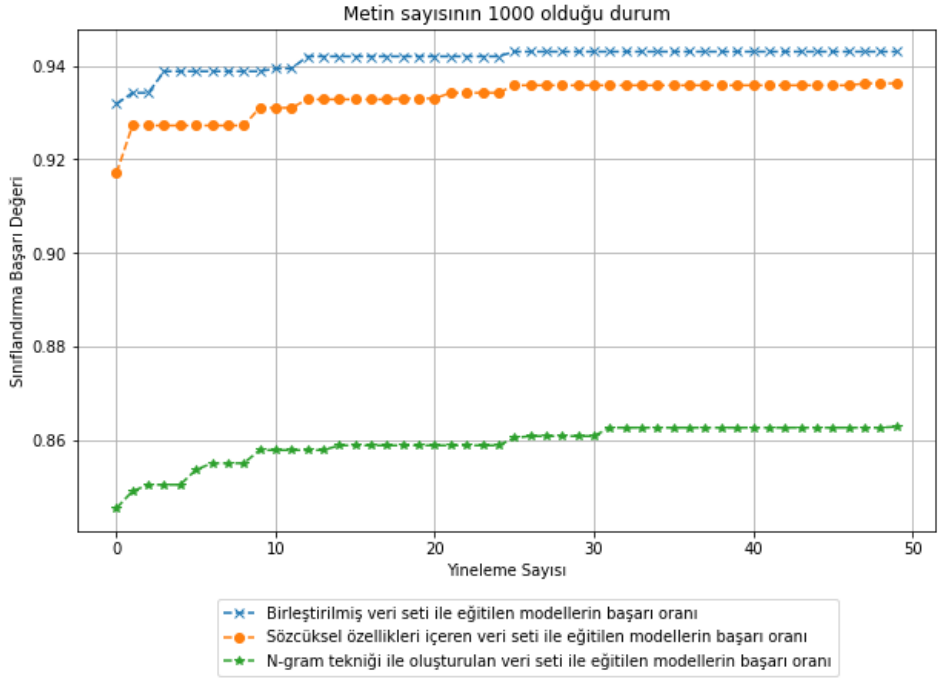

Şekil 4.8. Yazar sayısının 10 ve metin sayısının 100 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri



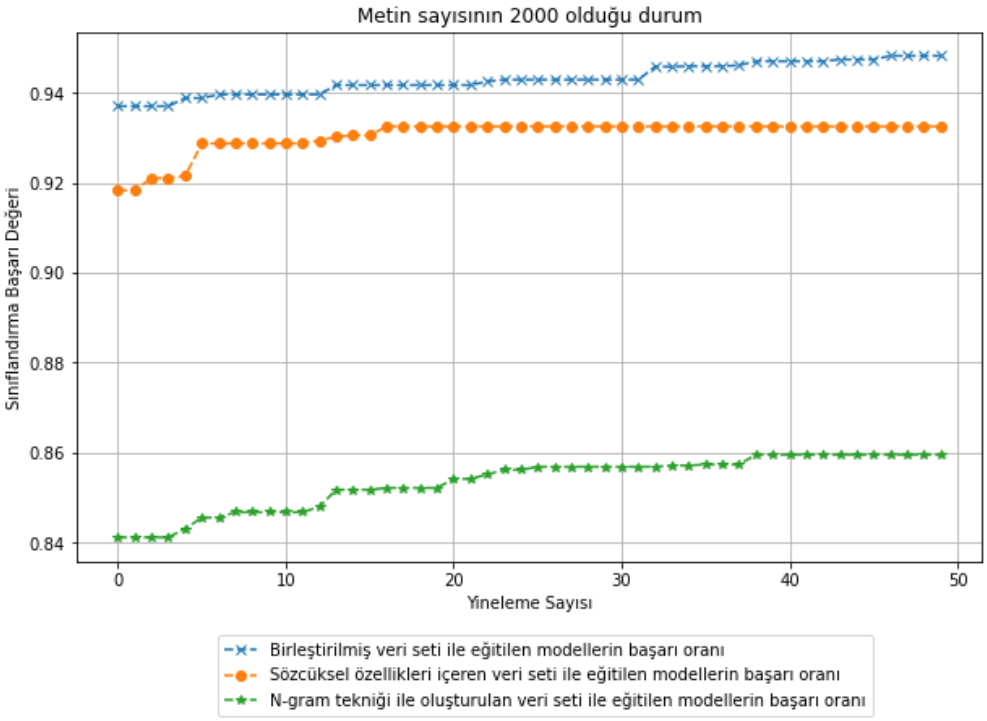

Şekil 4.9. Yazar sayısının 10 ve metin sayısının 200 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri




Şekil 4.10. Yazar sayısının 10 ve metin sayısının 500 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri




Şekil 4.11. Yazar sayısının 10 ve metin sayısının 1000 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri




Şekil 4.12. Yazar sayısının 10 ve metin sayısının 2000 olduğu durum için Analiz1, Analiz2 ve Karma Analiz ile oluşturulan veri kümeleri üzerinde GA kullanılarak gerçekleştirilen öznitelik seçim işleminde, oluşturulan popülasyondaki her yinelemedeki en yüksek doğruluk oranlarının değişimleri

Şekil 4.8., Şekil 4.9., Şekil 4.10., Şekil 4.11. ve Şekil 4.12.'de genel olarak doğruluk oranlarının değişimlerinin yineleme sayısının 40 olduğu durumdan sonra sabit kaldığı gözlemlenmiştir. En yüksek sınıflandırma doğruluğu birleştirilmiş veri kümesi yani Karma Analiz ile elde edilmiştir. Yineleme sayısının artışı genel olarak sınıflandırma doğruluğu üzerinde pozitif etkisi olmuştur.

GA kullanılarak gerçekleştirilen öznitelik seçim işleminden sonra için Analiz1, Analiz2 ve Karma Analiz ile hazırlanan veri kümelerinden alınan doğruluk oranlarının değişimleri, model oluşturma süreleri ve yeni öznitelik sayıları Çizelge 4.7 ve Çizelge 4.8'te gösterilmiştir.

Analiz1 ile oluşturulan veri kümeleri üzerinde GA kullanımı ile gerçekleştirilen işlem sonrasında 5 durum içinde öznitelik sayısında yüksek azalış gözlemlenmektedir. Bu durum TSOA'dan alınan sonuçlara benzerdir. Sınıflandırma doğruluğu için, öznitelik seçim işlemi öncesi ve sonrası için en büyük fark %0,46 ile metin sayısının 500 olduğu durumda yaşanmıştır. 5 farklı boyutta gerçekleştirilen karşılaştırmanın 2'si için öznitelik seçim

işleminde doğruluk oranı yükselmiştir. Tüm durumlar için model oluşturma sürelerinde öznitelik seçim işlemi sonrasında düşüş yaşanmıştır.

Çizelge 4.7. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz1 ile oluşturulan veri kümesi üzerinde GA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranı, model oluşturma süreleri ve 300 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri

| Her bir yazar için metin sayısı | GA'dan önce model doğruluk oranı (%) | GA'dan sonra model doğruluk oranı (%) | GA'dan önce model oluşturma süresi (sn) | GA'dan sonra model oluşturma süresi (sn) | GA'dan sonra oluşturulan veri kümesi öznitelik sayısı |
|---|---|---|---|---|---|
| 100 | 83,6 | 83,1 | 0,92 | 0,44 | 170 |
| 200 | 85 | 83,4 | 1,65 | 0,9 | 138 |
| 500 | 85,86 | 86,32 | 4,45 | 2,56 | 169 |
| 1000 | 85,42 | 85,28 | 15,45 | 5,68 | 160 |
| 2000 | 85,2 | 85,35 | 26,77 | 13,32 | 162 |

Analiz1 ile oluşturulan veri kümeleri üzerinde GA ve TSOA yöntemleri kullanılarak öznitelik kümelerinde seçim ile genel olarak doğruluk oranını artmıştır. Çizelge 4.4 ve Çizelge 4.7 incelendiğinde en yüksek sınıflandırma doğruluğu %86,32 ile metin sayısının 500 olduğu ve öznitelik seçim yöntemi olarak GA kullanıldığı durumda elde edilmiştir. Model oluşturma süreleri karşılaştırıldığında tüm durumlar için en düşük süreler GA uygulandıktan sonra oluşan veri kümelerinde elde edilmiştir.

Çizelge 4.8'e göre Analiz2 ile oluşturulan veri kümeleri ile GA kullanımının sonucunda %0,8'e kadar doğruluk oranında artış sağlanmıştır. Öznitelik seçim işleminde TSOA ve GA karşılaştırıldığında metin sayısının farklı olması ile oluşan 5 durum için:

- Doğruluk oranında pozitif yükseliş; TSOA ile 3 durumda, GA ile 4 durumda gerçekleşmiştir.

- Doğruluk oranında en yüksek artışlar; TSOA ile %0,4 iken GA ile %0,8'dir.
- Doğruluk oranında pozitif artıştaki en düşük artış miktarı TSOA ile %0,08 iken GA ile bu oran %0,39'dur.
- Tüm durumlar için ortalama öznitelik sayısı TSOA ile 23, GA ile 23,4'tür. TSOA, GA'dan daha az öznitelik kullanılarak tahmin yapılabileceğini göstermiştir.
- Model oluştururken geçen süre bazında değerlendirmede ortalama 3,294 saniye olan işlemler TSOA ile 2,564 saniyeye, GA ile 2,598 saniye düşürülmüştür.
- Sınıflandırma işleminde en yüksek doğruluk oranlarına metin sayısının 500 olduğu durumda TSOA %94,42 ve GA %95,26 ile ulaşılmıştır.

Çizelge 4.8. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Analiz2 ile oluşturulan veri kümesi üzerinde GA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranı, model oluşturma süreleri ve 40 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri

| Her bir yazar için metin sayısı | GA'dan önce model doğruluk oranı (%) | GA'dan sonra model doğruluk oranı (%) | GA'dan önce model oluşturma süresi (sn) | GA'dan sonra model oluşturma süresi (sn) | GA'dan sonra oluşturulan veri kümesi öznitelik sayısı |
|---|---|---|---|---|---|
| 100 | 91,2 | 92 | 0,41 | 0,41 | 24 |
| 200 | 92,95 | 92,8 | 0,84 | 0,49 | 23 |
| 500 | 94,84 | 95,26 | 1,68 | 1,31 | 20 |
| 1000 | 93,74 | 94,13 | 4 | 3,36 | 26 |
| 2000 | 93,37 | 93,89 | 9,54 | 7,42 | 24 |

Tüm bu değerlendirmeler ışığında öznitelik sayıları ve sürelerde çok büyük fark bulunmamaktadır. Genel olarak karşılaştırma doğruluk oranına göre yapılmıştır. Sonuç olarak, GA'nın TSOA'dan daha başarılı sonuçlar ürettiği gözlenmiştir.

Çizelge 4.9. Yazar Sayısının 10 olduğu durumda 5 farklı boyut üzerinde Karma Analiz ile oluşturulan veri kümesi üzerinde GA uygulanmadan öncesi ve sonrası için ROA ile gerçekleştirilen sınıflandırma sonucunda doğruluk oranı, model oluşturma süreleri ve 340 olan öznitelik sayılarının uygulama sonrasındaki yeni değerleri

| Her bir yazar için metin sayısı | GA'dan önce model doğruluk oranı (%) | GA'dan sonra model doğruluk oranı (%) | GA'dan önce model oluşturma süresi (sn) | GA'dan sonra model oluşturma süresi (sn) | GA'dan sonra oluşturulan veri kümesi öznitelik sayısı |
|---|---|---|---|---|---|
| 100 | 92,23 | 93,2 | 0,64 | 0,39 | 173 |
| 200 | 93,35 | 94,45 | 1,9 | 0,81 | 167 |
| 500 | 94,46 | 94,48 | 3,58 | 2,29 | 180 |
| 1000 | 93,42 | 93,99 | 8,87 | 5,68 | 191 |
| 2000 | 93,68 | 94,9 | 23,62 | 12,56 | 193 |

Çizelge 4.9 ve Çizelge 4.6'a göre öznitelik seçim işleminde TSOA ve GA karşılaştırıldığında metin sayısının farklı olması ile oluşan 5 durum için:

- Doğruluk oranında pozitif yükseliş TSOA ile 4 durumda, GA ile 5 durumda (tüm durumlarda) gerçekleşmiştir.
- Doğruluk oranında en yüksek artış TSOA ve GA için %1,22'dir.
- Doğruluk oranında pozitif artıştaki en düşük artış miktarı TSOA ile %0,6 iken GA ile bu miktar %0,02'dir.
- Tüm durumlar için ortalama öznitelik sayısı TSOA ile 170,6, GA ile 180,8'dir. TSOA, GA algoritmasından daha az öznitelik kullanılarak yazar tahmini yapılabileceğini göstermiştir.
- Model oluştururken geçen süre bazında değerlendirmede ortalama 7,722 saniye; TSOA ile 5,474 saniyeye, GA ile 4,346 saniyeye düşürülmüştür.
- Sınıflandırma işleminde en yüksek doğruluk oranı; TSOA ile metin sayısının 500 olduğu durumda %95,06, GA ile metin sayısının 2000 olduğu durumda %94,9'dur.

- Doğruluk oranında artış miktarları tüm durumlardaki artışın ortalamasına göre incelendiğinde TSOA ile %0,702, GA ile %0,776 yükseliş yaşanmıştır.

Tüm bu değerlendirmeler ışığında hem model oluşturma süreleri hem de doğruluk oranında alınan artışa göre GA'nın, TSOA'dan daha başarılı sonuçlar ürettiği sonucuna ulaşılmıştır. Uygulanan tüm analiz ve yöntemler sonucunda en başarılı öznitelik kümelerine Karma Analiz ile ulaşılmıştır.

TÖA ile sınıflandırma işlemi için analizlerde elde edilmiş en başarılı veri kümesi kullanılması planlanmıştır. Bu plana göre Karma Analiz ile oluşturulan veri kümeleri üzerinde GA ile alınmış en yüksek sınıflandırma doğruluğu, yazar sayısının 10 metin sayısının 2000 olduğu durumda gerçekleşmiştir (Bkz. Çizelge 4.9). Bu veri kümesi ile Torbalama ve Artırma algoritmaları uygulanmıştır.

Çizelge 4.10. Torbalama Yöntemi ile yazar sayısının 10 metin sayısının 2000 olduğu (Karma Analiz kullanılarak özniteliklerin çıkarılıp GA ile öznitelik seçim işlemi uygulandıktan sonra oluşturulan) veri kümesi üzerinde gerçekleştirilen sınıflandırma sonucu başarım metrikleri

| Torbalama Algoritması içerisinde kullanılan sınıflandırıcı metot | Doğruluk (ort) (%) | Kesinlik (ort) (%) | Duyarlılık (ort) (%) | F-ölçütü (ort) (%) |
|---|---|---|---|---|
| KEYK | 95,22 | 95,3 | 95,3 | 95,2 |
| Karar Ağacı | 95,74 | 95,8 | 95,7 | 95,8 |
| ÇTNB | 88,2 | 88,7 | 88,2 | 87,9 |

Çizelge 4.10 incelendiğinde en yüksek doğruluk oranı Torbalama içerisinde sınıflandırıcı metot olarak Karar Ağacı kullanıldığında erişilmiştir. Çizelge 4.11'te Artırma Algoritması olarak XGBoost kullanıldığında en yüksek başarı oranlarına ulaşılmıştır. İki farklı tarzda TÖA ile alınan tüm sonuçlar incelendiğinde, en yüksek %95,74 doğruluk oranı ile Torbalama Algoritmasında sınıflandırıcı metot olarak Karar Ağacı kullanıldığında ulaşılmıştır. En yüksek başarı Torbalama metodu ile alındığı gibi en düşük başarı da yine bu metot ile alınmıştır.

Çizelge 4.11. Artırma Yöntemi ile yazar sayısının 10 metin sayısının 2000 olduğu (Karma Analiz kullanılarak özniteliklerin çıkarılıp GA ile öznitelik seçim işlemi uygulandıktan sonra oluşturulan) veri kümesi üzerinde gerçekleştirilen sınıflandırma sonucu başarım metrikleri

| Artırma Algoritmaları | Doğruluk (ort) (%) | Kesinlik (ort) (%) | Duyarlılık (ort) (%) | F-ölçütü (ort) (%) |
|---|---|---|---|---|
| XGBoost | 95,3 | 95,2 | 95,3 | 95,2 |
| LightGBM | 93,95 | 93,9 | 93,9 | 93,9 |
| Adaboost | 89,62 | 90 | 89,6 | 89,5 |

Torbalama Algoritmaları ile gerçekleştirilen üç sınıflandırmanın ortalama doğruluk oranları %93,053, artırma yönteminde ise üç sınıflandırma yönteminin ortalama doğruluk oranı %92,9'dir. İki farklı TÖA sınıflandırma ortalama doğruluk oranlarında %0,096 kadar fark vardır. Bu sonuçlar ışığında bu çalışmada önerilen analiz ve yöntemlerin metinden yazar tanımada yüksek başarı sağladığı sonucuna ulaşılmıştır.

# 5. SONUÇ VE ÖNERİLER

Bilgiye erişimin çok kolay olduğu ama bilginin doğruluğundan emin olunmasının ise bir o kadar zor olduğu bir dönemde yaşamaktayız. Bilginin doğruluğuna ulaşmak için çeşitli problemler ileri sürülmüştür. Bunlardan en önemlisi bilgiyi sunan kişinin kim olduğunu bulmaktır. Özellikle dijital olarak hazırlanan metinlerin artması, metni oluşturan kişinin bilgisine olan ihtiyacı da artırmıştır. Metin yazarı tahmini ile doğru bilgiye ulaşarak infobezite etkisi azaltılabilir, eser hırsızlığı, intihar notu, terörist bildirileri gibi suçların aydınlatılmasında kullanılabilir. Bu çalışmada, bahsedilen problemlerin çözümü temelinde yazar tahmini için kullanılabilecek bir yöntem sunuyoruz.

Yazar tahmini için geniş bir Türkçe derlemin olmaması nedeniyle 4 farklı haber kaynağından toplamda 54 adet yazar ve değişken sayıda ve uzunlukta toplam 46837 adet köşe yazısına sahip bir derlem oluşturulmuştur. Çalışma, derlem üzerinde yapılan iki analiz ve onların birleşimleri ile toplamda 3 yönden tartışılmaya sunulmuştur. İlk analiz metin üzerinden n-gram tekniği ile gerçekleştirilmiştir ve Analiz1 olarak isimlendirilmiştir. Çalışmanın her aşamasında yazar sayısında ve metin sayısında farklılıklar oluşturularak çeşitli senaryolar oluşturulmuştur. Tüm senaryolarda sınıflandırıcı metot olarak ROA kullanılmıştır. Analiz1 için kullanılan parametrelerin en uygun değerleri için toplamda 140 adet senaryo üzerinde işlem gerçekleştirilmiştir. Analiz1, Analiz2 ve Karma Analiz karşılaştırılması için 18 adet durum ile toplamda 54 adet senaryo tartışılmıştır. Tüm senaryolar arasında en yüksek doğruluk oranı %98,92 ile 5 yazar ve her yazar için 500 metinden Karma Analiz ile oluşturulan veri kümesine ait modelden elde edilmiştir. Sınıflandırma doğruluğunun artırılması, model oluşturma ve test sürelerinin azaltılmasına yönelik öznitelik seçim işlemleri gerçekleştirilmiştir. Öznitelik seçim yöntemi olarak TSOA ve GA kullanılmıştır. Her iki algoritmada da uygunluk değeri hesaplamasında ROA ile oluşturulan modellerin başarı metrikleri kullanılmıştır. 10 adet yazar ve onlara ait 5 farklı boyuttaki metin sayıları ile oluşturulan modellere ait ortalama doğruluk oranları Çizelge 5.1'de verilmiştir. Metodoloji bölümünün özetini içeren bu tablo ile yazar tanıma probleminde en uygun analizin, Karma Analiz olduğu sonucuna ulaşılmıştır. Öznitelik seçim yöntemlerinden ise doğruluk oranlarının karşılaştırmasına göre en uygun olanının GA olduğu sonucu çıkarılmıştır.

Çizelge 5.1. Yazar sayısının 10 olduğu ve metin sayısının 5 farklı boyutta (100, 200, 500, 1000 ve 2000) olduğu durumlar için öznitelik seçim işlemi öncesi ve sonrası ROA ile oluşturulan modellerin 5 durum için ortalama doğruluk oranları

| Analiz | Öznitelik seçim işleminden önce ortalama doğruluk oranı (%) | TSOA'dan sonra ortalama doğruluk oranı (%) | GA'dan sonra ortalama doğruluk oranı (%) |
|---|---|---|---|
| Analiz1 | 85,016 | 84,838 | 84,69 |
| Analiz2 | 93,22 | 93,264 | 93,616 |
| Karma | 93,428 | 94,13 | 94,204 |

Çizelge 5.1'de sunulan en yüksek %94,204 doğruluk oranına ulaşılan veri kümesi üzerinde 2 farklı TÖA ile toplamda 6 farklı algoritma ile sınıflandırma yapılmıştır. Sınıflandırma sonuçlarına göre yazar tahmini işleminde en yüksek doğruluk oranı Torbalama içerisinde sınıflandırıcı metot olarak Karar Ağacının kullanılması ile elde edilmiştir. Elde edilen en yüksek doğruluk oranı %95,74'tür.

Türkçe alanında yapılan çalışmalar ile bu çalışmada önerilen yöntemlerden elde edilen doğruluk oranları Çizelge 5.2'de karşılaştırılmıştır. Tüm çalışmalar için ortak önemli nokta tahmin edilecek yazar sayısı ve elde edilen doğruluk oranıdır. Bu amaçla her çalışmada kullanılan yazar sayısı boyutu için GA uygulandıktan sonra Karma Analiz'e ait özniteliklerine göre aynı boyutta veri kümeleri oluşturulmuştur. Oluşturulan veri kümeleri için ROA ile sınıflandırma işlemi gerçekleştirilmiştir. Çizelge 5.2'de çalışmada kullanılan yazar sayısı, metin sayısı ve elde edilen ortalama doğruluk oranları gösterilmektedir.

Çizelge 5.2 incelendiğinde en yüksek doğruluk oranları bu çalışmada önerilen yöntem ile elde edilmiştir. Tahmin edilecek yazar sayısının artışı, genel olarak başarı üzerinde negatif etki oluşturmuştur. Sınıf sayısının arttığı durumlarda başarının düşmesi genel bir sınıflandırma problemidir. Bu çalışmada sınıf sayısının 7 kat artması (yazar sayısının 5'ten 40 değerine ulaşması) doğruluk oranında %10,26'lık bir değişime sebep olmuştur. Tüm durumlar incelendiğinde en yüksek sınıflandırma doğrulukları yazar sayısının 5 olduğu durumda elde edilmiştir.

Çizelge 5.2. Bu çalışmada önerilen yöntem ve diğer Türkçe derlemler ile yapılan çalışmaların karşılaştırılması

| Çalışmalar | Yazar Sayısı | Veri Kümesi Büyüklüğü | Doğruluk oranı (ort.) (%) |
|---|---|---|---|
| (Ekinci ve Takci, 2012) | 5 | Her yazar için 50 e-posta mesajı (toplamda 250 e-posta metni) | 83 |
| Bu Çalışma | 5 | Her yazar için 500 metin (toplamda 2 500 metin) | 99,26 |
| (Türkoğlu ve diğerleri., 2007) | 18 | Her yazar için 35 metin (toplamda 630 metin) | 72,4 |
| | 9 | Her yazar için 35 metin (toplamda 315 metin) | 82,9 |
| Bu Çalışma | 18 | Her yazar için 200 metin (toplamda 3 600 metin) | 94,8 |
| | 9 | Her yazar için 500 metin (toplamda 4 500 metin) | 96,3 |
| (Taş ve Görür, 2007) | 20 | Her yazar için 25 metin (toplamda 500 metin) | 80 |
| Bu Çalışma | 20 | Her yazar için 200 metin (toplamda 3 600 metin) | 93,85 |
| (Agun ve diğerleri., 2017) | 25 | 2015-2017 yılları arasında yazılmış Türkçe gazete makaleleri. Metin sayısı belirtilmemiş | 73,22 |
| Bu Çalışma | 25 | Her yazar için 100 metin (toplamda 2 500metin) | 92,13 |
| (Aydemir, 2019) | 40 | Her yazar için 10 metin (toplamda 400 metin) | 72 |
| Bu Çalışma | 40 | Toplamda 3269 metin (her yazar için minimum 20 metin kullanılmıştır) | 89 |

Her çalışmanın kendi tahmin edilecek yazar sayısı boyutu ile karşılaştırıldığında tüm durumlar için bu çalışmada önerilen yöntemler ile daha yüksek doğruluk değerleri elde edilmiştir. Yazar sayısı boyutlarına göre bu çalışmanın doğruluk oranındaki artışları şöyledir;

- Yazar sayısının 5 olduğu durumda %16,26
- Yazar sayısının 9 olduğu durumda %13,4
- Yazar sayısının 18 olduğu durumda %22,4
- Yazar sayısının 20 olduğu durumda %13,85
- Yazar sayısının 25 olduğu durumda %18,91
- Yazar sayısının 40 olduğu durumda %17

Metinden yazar tahmini problemini konu alan Türkçe çalışma sayısı fazla olmadığı için detaylı analiz veya karşılaştırma yapmak zordur. Bu nedenle bu çalışmada detaylı analizlere ve karşılaştırmalara yer verilmiştir. Bu çalışmada gerçekleştirilen detaylı analizler sonucunda diğer Türkçe çalışmalara göre tahmin edilecek yazar sayısının fazlalığa rağmen dikkate değer bir başarı elde edilmiştir. Tüm süreçlerde toplamda 8816 sınıflandırma işlemi gerçekleştirilmiştir. Ayrıca TSOA-ROA ve GA-ROA yöntemleri farklı sınıflandırma problemlerinin çözümünde de kullanılabileceği için öznitelik seçim işleminde kullanımının etkisi de incelenmiştir.

Çeşitli büyüklükteki veri kümeleriyle çalışmak, yazarlık sorununu farklı alanlara uygulanabilecek farklı senaryolara göre ölçeklendirme yeteneği kazandırır. Yazar kimliğinin sadece akademi veya edebiyatta önemli bir faktör olmadığını, diğer alanlara da genişletilebilir. Bu konuyla ilgili gelecekteki araştırmalar, kamuya açık literatürün ötesinde uygulanabilir ve özellikle SMS mesajlarından, web makalelerinden ve e-postalardan kaynaklanan sınırlı metin örneklerine odaklanabilir.

# KAYNAKLAR


Agun, H. V., Yilmazel, S., and Yilmazel, O. (2017). *Effects of language processing in Turkish authorship attribution*. 2017 IEEE International Conference on Big Data (Big Data), 1876–1881.

Akın, A. A., and Akın, M. D. (2007). Zemberek, an open source NLP framework for Turkic Languages. *Structure*, *10*, 1–5.

Al-Sarem, M., Saeed, F., Alsaeedi, A., Boulila, W., and Al-Hadhrami, T. (2020). Ensemble methods for instance-based Arabic language authorship attribution. *IEEE Access*, *8*, 17331–17345.

Alshaher, H., and Xu, J. (2020). *A New Term Weight Scheme and Ensemble Technique for Authorship Identification*. ICCDA 2020: Proceedings of the 2020 the 4th International Conference on Compute and Data Analysis. 123–130.

Aslantürk, O., Sezer, E. A., Sever, H., and Raghavan, V. (2010). *Application of cascading rough set-based classifiers on authorship attribution.* Proceedings - 2010 IEEE International Conference on Granular Computing, GrC 2010, 656–660.

Aydemir, E. (2019). Türkçe Köşe Yazılarında Yapay Sinir Ağlarıyla Yazar ve Gazete Tahmin Etme. *DÜMF Mühendislik Dergisi*, *10*(1), 45–56.

Bay, Y., and Çelebi, E. (2016). *Feature Selection for Enhanced Author Identification of Turkish Text*. Springer, Cham, 371–379.

Boran, T., Martinaj, M., and Hossain, M. S. (2020). Authorship identification on limited samplings. *Computers and Security*, 97, 101943.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2), 123–140.

Chen, T., and Guestrin, C. (2016). *XGBoost*. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.

Coban, O., and Karabey, I. (2017). *Music genre classification with word and document vectors*. 2017 25th Signal Processing and Communications Applications Conference (SIU), 1–4.

De Morgan, S. E., and De Morgan, A. (1882). *Memoir of Augustus de Morgan by his wife Sophia Elizabeth de Morgan with selections from his letters.* London: Longmans, Green, and Co.

Diederich, J., Kindermann, J., Leopold, E., and Paass, G. (2003). Authorship attribution with support vector machines. *Applied Intelligence*, 19(1–2), 109–123.

Digamberrao, K. S., and Prasad, R. S. (2018). Author Identification using Sequential Minimal Optimization with rule-based Decision Tree on Indian Literature in Marathi. *Procedia Computer Science*, 132, 1086–1101.

Doğan, N. (2016). İstem Sözcükleri ve Türkçe. *The Journal of Academic Social Science Studies, 1*(42), 251.

Ekinci, E., and Takci, H. (2012). *Using authorship analysis techniques in forensic analysis of electronic mails*. 2012 20th Signal Processing and Communications Applications Conference (SIU), 1–4.

Fatih Amasyali, M., and Diri, B. (2006). Automatic Turkish text categorization in terms of author, genre and gender. *Natural Language Processing and Information Systems. NLDB 2006. Lecture Notes in Computer Science*, Berlin:Springer, 221–226.

Friedl, M. A., and Brodley, C. E. (1997). Decision tree classification of land cover from remotely sensed data. *Remote Sensing of Environment*, 61(3), 399–409.

Hossny, A. H., Mitchell, L., Lothian, N., and Osborne, G. (2020). Feature selection methods for event detection in Twitter: a text mining approach. *Social Network Analysis and Mining*, 10(1), 1-15.

Hou, R., and Huang, C. R. (2020). Robust stylometric analysis and author attribution based on tones and rimes. *Natural Language Engineering*, 26(1), 49–71.

İlhan, N., ve Sağalltıcı, D. (2020). Twitter'da Duygu Analizi. *Harran Üniversitesi Mühendislik Dergisi*, 5(2), 146–156.

Juola, P. (2005). A Controlled-corpus Experiment in Authorship Identification by Cross-entropy. *Literary and Linguistic Computing*, 20, 59–67.

Juola, Patrick, and Baayen, R. H. (2005). A controlled-corpus experiment in authorship identification by cross-entropy. *Literary and Linguistics Computing*, 20, 59–67.

Murthy, S. K. (1998). Automatic Construction of Decision Trees from Data: A Multi-disciplinary Survey. *Data Mining and Knowledge Discovery*, 2(4), 345–389

Kalgutkar, V., Kaur, R., Gonzalez, H., Stakhanova, N., and Matyukhina, A. (2019). Code authorship attribution: Methods and challenges. *ACM Computing Surveys*, 52(1), 1-36.

Kaşıkçı, T., and Gökçen, H. (2013). Metin Madenciliği İle E-Ticaret Sitelerinin Belirlenmesi. *Bilişim Teknolojileri Dergisi*, 7(1), 25-32.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T. Y. (2017). *LightGBM: A highly efficient gradient boosting decision tree.* In Advances in Neural Information Processing Systems, 2017-Decem(Nips), 3147–3155.

Keselj, V., Peng, F., Cercone, N., and Thomas, C. (2003). *N-gram-based author profiles for authorship attribution.* In Proceedings of Pacific Association for Computational Linguistics 2003, 255–264.

Koppel, M., Akiva, N., and Mughaz, D. (2006). New methods for attribution of rabbinic literature. *Hebrew Linguistics: A Journal for Hebrew Descriptive, Computational and Applied Linguistics*, 57, 5–18.

Kukushkina, O. V., Polikarpov, A. A., and Khmelev, D. V. (2001). Using Literal and Grammatical Statistics for Authorship Attribution. *Problems of Information Transmission*, 37(2), 172–184.

Li, W. J., Wang, K., Stolfo, S. J., and Herzog, B. (2005). *Fileprints: Identifying file types by n-gram analysis.* Proceedings from the 6th Annual IEEE System, Man and Cybernetics Information Assurance Workshop, SMC 2005, 64–71.

Liang, S., Fang, Z., Sun, G., Liu, Y., Qu, G., and Zhang, Y. (2020). Sidelobe Reductions of Antenna Arrays via an Improved Chicken Swarm Optimization Approach. *IEEE Access*, 8, 37664–37683.

Liang, X., Kou, D., and Wen, L. (2020). An Improved Chicken Swarm Optimization Algorithm and its Application in Robot Path Planning. *IEEE Access, 8*, 49543–49550.

Loper, E., and Bird, S. (2002). NLTK: *the Natural Language Toolkit*. Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics, 1, 63–70.

Mendenhall, T. C. (1887). The Characteristic Curves of Composition. *Science*, 9(214), 237–249.

Mosteller, F., and Wallace, D. L. (1964). *Inference and Disputed Authorship: The Federalist*. Reading, MA: Addison-Wesley Publishing Company, (Reprinted 2008. Stanford: Center for the Study of Language and Information.)

Osamy, W., El-Sawy, A. A., and Salim, A. (2020). CSOCA: Chicken Swarm Optimization Based Clustering Algorithm for Wireless Sensor Networks. *IEEE Access*, 8, 60676–60688.

Otoom, A. F., Abdullah, E. E., Jaafer, S., Hamdallh, A., and Amer, D. (2014). *Towards author identification of Arabic text articles*. 2014 5th International Conference on Information and Communication Systems (ICICS*)*, 1–4.

Ouamour, S., and Sayoud, H. (2013). *Authorship Attribution of Short Historical Arabic Texts Based on Lexical Features.* 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 144–147.

Patrick, E. A., and Fischer, F. P. (1970). A generalized k-nearest neighbor rule. *Information and Control*, 16(2), 128–152.

Patton, J. M., and Can, F. (2004). A stylometric analysis of Yaşar Kemal's İnce Memed tetralogy. *Computers and the Humanities*, 38(4), 457–467.

Polat, H., ve Körpe, M. (2018). TBMM Genel Kurul Tutanaklarından Yakın Anlamlı Kavramların Çıkarılması. *Bilişim Teknolojileri Dergisi*, 11(3).

Quinlan, J. R. (1996). *Bagging, boosting, and C4.5.* Proceedings of the National Conference on Artificial Intelligence, 1, 725–730.

Ren, G., Yang, R., Yang, R., Zhang, P., Yang, X., Xu, C., Xu, B., Zhang, H., Lu, Y., and Cai, Y. (2019). A parameter estimation method for fractional-order nonlinear systems based on improved whale optimization algorithm. *Modern Physics Letters B*, 33(7).

Sarwar, R., Porthaveepong, T., Rutherford, A., Rakthanmanon, T., and Nutanong, S. (2020). StyloThai: A scalable framework for stylometric authorship identification of Thai documents. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 19(3).

Savoy, J. (2015). Comparative evaluation of term selection functions for authorship attribution. *Digital Scholarship in the Humanities*, 30(2), 246–261.

Schapire, R. E. (1999). *A brief introduction to boosting.* IJCAI International Joint Conference on Artificial Intelligence, 2(5), 1401–1406.

Şimşek İşliyen, F. (2020). Dijital Çağda Bilginin Değişen Nïteliği̇ ve İnfobezite: Z Kuşağı Üzerine Bir Odak Grup Çalışması. *Selçuk İletişim*, 13(1), 246–272.

Soler-Company, J., and Wanner, L. (2018). On the role of syntactic dependencies and discourse relations for author and gender identification. *Pattern Recognition Letters*, 105, 87–95.

Taş, T., and Görür, A. K. (2007). Author Identification for Turkish Texts. *Çankaya Üniversitesi Fen-Edebiyat Fakültesi, Journal of Arts and Sciences*, 7, 151–161.

Tian, F., Zhang, R., Lewandowski, J., Chao, K.-M., Li, L., and Dong, B. (2017). Deadlock-free migration for virtual machine consolidation using Chicken Swarm Optimization algorithm. *Journal of Intelligent ve Fuzzy Systems*, 32(2), 1389–1400.

Türkoğlu, F., Diri, B., and Amasyalı, M. F. (2007). Author Attribution of Turkish Texts by Feature Mining. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issue,s* 1086–1093. Springer Berlin Heidelberg.

Vijayakumar, B., and Fuad, M. M. M. (2019). A new method to identify short-text authors using combinations of machine learning and natural language processing techniques. *Procedia Computer Science*, 159, 428–436.

Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. *Journal of Information Science*, 44(1), 48–59.

Yıldırım, E., Çetin, F., G., E., and T., T. (2015). The Impact of NLP on Turkish Sentiment Analysis. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendislik Dergisi,* 43–51.

Yüksel, A. S., ve Tan, F. G. (2018). Metin Madenciliği Teknikleri ile Sosyal Ağlarda Bilgi Keşfi. *Mühendislik Bilimleri ve Tasarım Dergisi*, 6(2), 324–333.

Yule, G. U. (1944). The statistical study of literary vocabulary. *Cambridge [England: University Press*.

Zdonak, I. (2020). The Role of Word and N-gram Frequency Analysis in Inference of the Content of Scientific Publication. *Scientific Papers of Silesian University of Technology. Organization and Management Series*, 142, 21–31.

Zheng, R., Li, J., Chen, H., and Huang, Z. (2006). A framework for authorship identification of online messages: Writing-style features and classification techniques. *Journal of the American Society for Information Science and Technology*, 57(3), 378–393.

# ÖZGEÇMİŞ

## Kişisel Bilgiler

Soyadı, adı : GÜLLÜ, Merve

Uyruğu : T.C.

## Eğitim

| Derece | Eğitim Birimi | Mezuniyet Tarihi |
|---|---|---|
| Yüksek lisans | Gazi Üniversitesi / Bilgisayar Mühendisliği | Devam ediyor |
| Lisans | Gazi Üniversitesi / Bilgisayar Mühendisliği | 2015 |
| Lise | Ankara Nefise Andiçen Lisesi | 2010 |

## İş Deneyimi

| Yıl | Yer | Görev |
|---|---|---|
| 2019-Halen | Gazi Üniversitesi | Araştırma Görevlisi |
| 2017-2019 | T.C. Tarım ve Orman Bakanlığı | Bilgisayar Mühendisi |
| 2016-2017 | Proyapı Mühendislik Müşavirlik | Bilgisayar Mühendisi |

## Yabancı Dil

İngilizce

## Yayınlar

Ceyhan, E.B. ,Güllü, M., Ulucan, C. (2018). Analysis of Blood Groups from Fingerprint Patterns of Turkish Citizens. *Journal of engineering and technology*, 2(1), 29-38.

Duygu, B., Güllü, M., Coşkun, A. (2019). *Karar Ağaçlar ile Otistik Spektrum Bozukluğu Tanısı Koyma.* 3rd International Symposium on Innovative Approaches in Scientific Studies, 445-460.

Güllü, M., Ceyhan, E.B.,Ulucan, C. (2016). *A New Approach: Predicting Height of a Person from Joint Ratio of Fingers.* Proceedings of the Fifth International Conference on Network, Communication and Computing - ICNCC ’xx16.

Güllü, M., Polat H., (2019). *Classification of Chronic Kidney Disease by Machine Learning Methods.* 3. Uluslararas GAP Matematik-Mühendislik-Fen ve Sağlık Bilimleri Kongresi, 120-129.

Güllü M., Polat H., Çetin A.(2020). *Author Identification with Chicken Swarm Optimization Algorithm and Adaboost Approaches*. 2020 5th International Conference on Computer Science and Engineering (UBMK).

**Hobiler**

Doğa yürüyüşü, kitap okumak

*GAZİ GELECEKTİR...*