İKİLİ ARKA PLAN DESTEKLİ YAZAR - BAĞIMSIZ YAZARLIK DOĞRULAMA SİSTEMİ

BINARY BACKGROUND ASSISTED AUTHOR-INDEPENDENT AUTHORSHIP VERIFICATION SYSTEM

PELİN CANBAY

PROF. DR EBRU SEZER

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Egitim-Öğretim ve Sınav Yönetmeliğinin

Bilgisayar Mühendisliği Anabilim Dalı için Öngördüğü

DOKTORA TEZİ olarak hazırlanmıştır.

2020

Evladıma…

ÖZET

İKİLİ ARKA PLAN DESTEKLİ YAZAR - BAĞIMSIZ YAZARLIK DOĞRULAMA SİSTEMİ

Pelin CANBAY

Doktora, Bilgisayar Mühendisliği Bölümü

Tez Danışmanı: Prof. Dr. Ebru SEZER

Eş Danışman: Prof. Dr. Hayri SEVER

Temmuz 2020, 192 sayfa

Yazar doğrulama, yazarı bilinmeyen bir metnin şüpheli bir yazara ait olup olmadığını bulmaya çalışan bir karar problemidir. Bu problem, uzun yıllardır ilgi gören yazar analizi çalışmalarının en temel problemlerinden biridir. Yazar doğrulama probleminin zorluğu, şüpheli yazara ait olduğu bilinen doküman sayısının az olmasından kaynaklanmaktadır. Güncel ve ilgi gören bir konu olan yazar doğrulama problemine literatürde genellikle yazar bağımlı üslup karşılaştırması çözüm olarak sunulmuştur. Fakat şüpheli yazara ait olduğu bilinen metin miktarı azaldıkça doğru üslubun çıkarılması da zorlaşmaktadır. Bu tez çalışmasında, şüpheli yazara ait olduğu bilinen metin miktarının çok az olduğu durumda bile yazar doğrulamasının nasıl yapılması gerektiği üzerine çalışmalar yapılmıştır. Tez kapsamında, ele alınan bu durumun çözümü için ikili arka plan destekli yazar –bağımsız bir sistem önerilmiştir. Önerilen sistem mevcut çözümlere göre daha yüksek genelleştirme kabiliyetine sahiptir. Önerilen bu sistem ayrıca dil bağımsız bir sistemdir. Farklı dillerin kolaylıkla entegre edilebildiği bu sistem, tez kapsamında Türkçe

ve İngilizce dilleri ile test edilmiş ve yazar analizi alanında kullanılan açık bir İngilizce veri kümesinde şimdiye kadarki en başarılı yazar doğrulama sonucu elde edilmiştir. Tez kapsamında ek olarak bir Türkçe Blog Külliyatı da oluşturulmuş ve bu külliyat Türkçe odaklı yazar analizi çalışmalarının arttırılması amacı ile araştırmacılarla paylaşılmıştır. Türkçe Blog Külliyatı kullanıldığında da başarılı sonuçlar üreten önerilen sistem, Türkçe dili öncelikli, yüksek ayırt ediciliğe sahip özniteliklerin belirlenmesinde de faydalı olacak sonuçlar üretmiştir.

Anahtar Kelimeler: Yazar doğrulama, Yazar analizi, metin-doküman analizi, Türkçe Blog veri kümesi, makine öğrenmesi

ABSTRACT

BINARY BACKGROUND ASSISTED AUTHOR-INDEPENDENT AUTHORSHIP VERIFICATION SYSTEM

Pelin CANBAY

Doctor of Philosophy, Department of Computer Engineering

Supervisor: Prof. Dr. Ebru SEZER

Co- Supervisor: Prof. Dr. Hayri SEVER

July 2020, 192 pages

Authorship verification is a decision problem that tries to find out whether the text of an unknown author belongs to a suspicious author. The problem is the fundamental one of the studies in authorship analysis that has been interested in many years. The challenges of the authorship verification problems have arisen from the small number of known documents of the suspicious author. In the literature, author-dependent style comparison has generally presented as a solution to the authorship verification problem, which is a current and interesting subject. However, as the amount of known text of the suspicious author decreases, it becomes harder to extract accurate style. In this thesis, studies have been carried out on how authorship verification should be done even when the amount of known texts from the suspicious author is very low. Within the scope of this thesis, a binary background assisted author-independent system is proposed for the solution of the situation we handled. The proposed system has more generalization ability than the

current solutions. Besides, the proposed system is language independent. The system, in which different languages can be easily integrated, has been tested within the scope of the thesis with Turkish and English languages, and the most accurate authorship verification result has been obtained in a public English dataset used in the field of authorship analysis. In this thesis, also, a Turkish Blog dataset has been collected and shared with the researchers in order to increase the Turkish-based authorship analysis studies. The proposed system, which also produced successful results when using the Turkish Blog dataset, has produced results that will be useful even in determining the high discriminative features with the priority of the Turkish language.

Keywords: Authorship verification, authorship analysis, text-document analysis, Turkish Blog dataset, machine learning

TEŞEKKÜR

Bu tez çalışmamda, akademik ve sosyal hayatımda büyük emekleri olan kıymetli danışmanlarım Prof. Dr. Ebru Sezer ve Prof. Dr. Hayri Sever'e, tez izleme komitemde bulunan Prof. Dr. Erdoğan Doğdu ve Doç Dr Lale Özkahya hocalarıma, bu zorlu yolculukta benden desteğini esirgemeyen eşim Yavuz CANBAY'a, bir gülümsemesiyle tüm olumsuzlukları unutturan evladım Cihan Mert CANBAY'a ve bu günlere gelmemde büyük emeği olan anneme ve babama çok teşekkür ederim.

İçindekiler

ŞEKİLLER DİZİNİ

TABLOLAR DİZİNİ

SİMGELER VE KISALTMALAR

Simgeler

A	Yazarlar kümesi
D	Dokümanlar kümesi
V_B	Bileşim Vektörü
x^2	ki-kare
a_n^m	n numaralı yazarın m numaralı dokümanı
f_x	x numaralı öznitelik frekansı
M_n^m veya Mn_m	n numaralı yazarın m numaralı temsil modeli

Kısaltmalar

AV	Authorship Verification (Yazar Doğrulama)
ATV	Alan Temsil Vektörü
BBM	Binary Background Model (İkili Arka Plan Modeli)
tf-idf	Terim Frekansı – Ters Doküman Frekansı
tf-icf	Terim Frekansı – Ters Sınıf Frekansı
bow	BagOfWords (Kelime çantası)
SVM	Support Vector Machines (Destek Vektör Makineleri)
RBF	Radial Basis Function (Radyal Temel Fonksiyonu)
ANN	Artificial Neural Networks (Yapay Sinir Ağları)
LR	Logistic Regression (Lojistik Regresyon)
FS	Feature Set (Öznitelik Kümesi)
DS	DataSet (Veri Kümesi)

1. GİRİŞ

Bu bölümde metin analizi, yazar analizi ve yazar doğrulama hakkında genel bilgilere yer verilmiş, tezde ele alınan problemin tanımı, tezin amacı, özgün değeri ve organizasyonu alt başlıklar halinde sunulmuştur.

1.1. Genel Tanım

Metinler, yazılı ifadeler olarak insanlar arası iletişimi sağlayan en eski araçlardan biridir. Bir metin oluşturulduğu dilin semantik (anlam), sözdizimi (sentaks veya sıra), sözlük bilgisi, imla, noktalama ve morfoloji (ek-kök vb. yapı bilgisi) gibi tüm dilsel alanlarını kullanır**.** Bu alanların hepsi kendine ait kullanım kurallarına sahiptir fakat bu kuralların tekil veya birlikte kullanımları arasındaki seçenekler metnin yazarı tarafından belirlenir. Bir metin, yazarı tarafından alınan belirli seçimlerin sonucunda oluşmuş nihai bir üründür, dolayısı ile her metin kendi üreticisinin parmak izini taşır.

Yazma işi, araba sürmek, yürümek veya telefonla konuşmak gibi davranışsal (kavramsal veya soft) bir biyometridir [1, 2]. Bu özelliği ile metin formundaki veriler, anlamsal içerik barındırmanın yanında üreticisinin üslubunun bir temsilini de barındırmaktadır. Metinlerin analiz edilmesi ile yazarına ait üslupsal özelliklerin çıkarılması ve çıkarılan bu özelliklerin işlenerek metnin yazarı ile ilgili bilgi elde edilmesi, metin analizi çalışmalarının başlıca hedeflerinden biridir.

Günümüz dünyasında elektronik veri üretiminin sürekli ve katlanarak artıyor olması, insanların paylaşım yapabileceği ortamların ve bu ortamlara erişim imkanlarının artmış olması ile doğrudan bağlantılıdır. Üretilen verinin çok büyük bir bölümü metin verisidir. Elektronik metinlerin bolluğu (elektronik postalar, blog yazıları, sosyal medya yazışmaları, forum mesajları, kullanıcı yorumları, kaynak kodlar…) farklı alanlardaki metin analizinin potansiyelini ortaya çıkarmıştır. Elektronik ortamda var olan metinler başta insan hayatı olmak üzere birçok alanda yüksek fayda ve gelişim sağlıyor olsa da bu metinlerin tam anlamıyla güvenli ve güvenilir bir kaynaktan üretildiğinin ispatı pek mümkün değildir. İnternet dünyası takma isimle veya isimsiz olarak yazılmış metinler ile doludur. Özellikle adli ve finansal konular söz konusu olduğunda ele alınan metinlerin bilinmeyen yazarı ile ilgili bilgilerin, bu metinlerin analiz edilmesi ile elde edilebilmesi büyük önem taşımaktadır. Örneğin bir ürün ile

ilgili yapılan birkaç taraflı yorumun aynı kişi tarafından yapılıp yapılmadığı veya gönderilen birkaç elektronik tehdit postasının aynı kişi tarafından atılıp atılmadığı bilgisine ulaşabilmek, bir kişi veya kurumun sosyal hayatını sağlıklı sürdürebilmesi açısından büyük bir öneme sahiptir. Bu önem göz önüne alındığında, bir metin üzerinden yazarı ile ilgili bilgilerin çıkarılabilmesine olan ihtiyacın gün geçtikçe daha da arttığı görülebilmektedir.

Metin formundaki verilerin analiz edilmesi ile yapılan bilgi çıkarımı çalışmaları yıllar boyunca ilgi odağı olmuştur. Özellikle metinlerden yazarı/yazarları ile ilgili anlamlı bilgilerin çıkarılmaya çalışılması yaklaşık 200 yıllık bir geçmişe sahiptir [3, 4]. Yazar Analizi (Authorship Analysis) veya Yazar Niteleme (Authorship Attribution) adı altında yürütülen bu çalışmalar günümüzde hala popülerliğini korumaktadır. Bu tez çalışmasında ele alınan Yazar Doğrulama problemi, Yazar Analizi çalışmalarından biri olup, bu alanda ele alınan diğer problemlerin de en temel durumunu temsil etmektedir. Yazar doğrulama probleminin, yazar analizi çalışmalarındaki yeri ve alandaki diğer problemler ile benzerlik ve farklılıkları Bölüm 2'de detaylandırılmıştır.

1.2. Problem Tanımı

Bir dokümanın, az sayıda dokümanı bilinen bir yazar tarafından yazılıp yazılmadığının belirlenebilmesi zorlu bir problemdir. Literatürde Yazar Doğrulama olarak bilinen bu problem, adli bilişim çalışmalarının da bir alt dalı olup yapısı gereği gerçek dünya problemi olarak ele alınmaktadır. Şüpheli bir yazarın, sorgulanan bir dokümanın yazarı olup olmadığı sorusunun cevabı, arka planda o yazara ait yüzlerce doküman varsa kolaylıkla verilebiliyorken, bu dokümanların sayısının azalması verilecek cevabı hayli zorlaştırmaktadır. Klasik yaklaşımda yazar doğrulama problemi, az bir miktar (en fazla 10) dokümanı bilinen bir yazarın, harici bir dokümanın da yazarı olup olmadığının doğrulanabilmesi olarak ele alınmaktadır. Yani, bir yazara ait olduğu bilinen az miktarda dokümanın, o yazarın üslubunu temsil edebilmesi ve sorgulanan başka bir dokümanın da söz konusu yazar tarafından yazılıp yazılmadığını doğrulayabilmesi beklenmektedir. Gerçek dünya problemlerinde de sorgulanan bir metnin şüpheli yazarı ile ilgili arka planda çok sayıda doküman yoktur. Klasik yaklaşımda, arka planda söz konusu yazara ait, miktarı az da olsa bir grup doküman bulunmaktadır. Fakat bu durumu sağlamak gerçek dünyada her zaman mümkün değildir. Bu sebeple yazar doğrulama probleminin en zorlu seviyesi olan, "bir yazara ait sadece nispeten kısa bir dokümanın bilinmesi durumunda başka dokümanlarının yazarlığının doğrulanabilmesi" noktasında bu tez

çalışmasında çözümler sunulmuştur. Yani, ele aldığımız problemde, arka planda şüpheli yazar tarafından yazılmış sadece kısa bir doküman vardır ve harici bir dokümanın da bu yazar tarafından yazılıp yazılmadığının doğrulanması istenmektedir. Klasik yazar doğrulama problemlerinin aksine bu tez çalışmasında yazar bağımsız çözümler üzerine durulmuştur. Öyle ki, klasik yazar doğrulama çalışmaları arka planda var olan dokümanlardan şüpheli yazara ait bir profil çıkarıp sorgulanan dokümanın bu profili temsil etme derecesinde değerlendirme yaparken, bu tez çalışmasında aynı yazara ve farklı yazarlara ait dokümanların arasındaki benzerlikler veya farklılıklar üzerinden, yazar profili olmadan kullanılabilecek çözümler sunulmuştur. Çalışmanın genelinde yazar doğrulama probleminin sadece nispeten kısa bir doküman kullanılarak çözülebilmesi üzerine odaklanılmış, bu odak doğrultusunda farklı çözüm yaklaşımları sunulmuştur.

1.3. Tezin Amaç ve Hedefleri

Bir yazara ait en ayırt edici yazma üslubunun, o yazara ait tek bir dokümandan hatta mümkün olabildiği en kısa metinden çıkarabilmek ve böylece harici metinlerin yazar doğrulamasını yapabilmek bu tezin ana motivasyonudur. Bu kapsamda, tezin başlıca amaç ve hedefleri aşağıda listelenmiştir;

- Yazar analizi çalışmalarına katkı sağlayacak yöntem ve araçları geliştirmek, belirlemek ve uygulamak,
- Türkçe yazar analizi çalışmalarında kullanılabilecek en etkili yöntem ve araçları geliştirmek, belirlemek ve uygulamak,
- Klasik yazar doğrulama problemine daha fayda temelli bir yaklaşım kazandırmak,
- Klasik yazar doğrulama problemini daha güncel, daha zorlu ve daha bilgilendirici bir açıdan ele almak,
- Türkçe yazar analizi çalışmalarının gelişmesi ve yaygınlaşması için güncel ve iyi tanımlı bir veri kümesi oluşturmak,
- Türkçe dili ile yazılmış dokümanlardan, bir yazarı temsil edecek en anlamlı öznitelik kümelerini belirlemek,
- İngilizce dili ile yazılmış dokümanlardan, bir yazarı temsil edecek en anlamlı öznitelik kümelerini belirlemek,
- Bir yazarı temsil edecek en anlamlı dil bağımsız öznitelik kümelerini belirlemek,

- Seçili öznitelik kümelerinin literatürde var olan ve yeni geliştirilen öznitelik seçme algoritmaları kullanılarak yazar temsil derecelerini ölçmek,
- En yüksek yazar temsil derecesine sahip öznitelikleri güncel yazar doğrulama probleminin çözümünde kullanmak,
- Oluşturulan veri kümesinin Türkçe yazar tanıma ve Türkçe yazar doğrulama problemlerindeki temel başarı değerlerini belirlemek,
- Ele alınan yazar doğrulama probleminin çözümü için dil bağımsız yazar doğrulama sistemi geliştirmek,
- Ele alınan yazar doğrulama probleminin çözümü için konu ve tür bağımsız yazar doğrulama sistemi geliştirmek,
- Ele alınan yazar doğrulama probleminin çözümü için yazar bağımsız yazar doğrulama sistemi geliştirmek,
- Geliştirilen başarılı model ve sistemleri, oluşturulan Türkçe veri kümesi ve açık bir İngilizce veri kümesi üzerinde uygulamak, test etmek ve elde edilen sonuçları yayınlamak.

1.4. Tezin Özgün Değeri

Yazar analizi çalışmaları 200 yıllık bir geçmişe sahip olmasına rağmen güncel teknolojik gelişmeler ile birlikte elektronik metinlerin yazar analizi çalışmaları son yıllarda popüler hale gelmiştir. Yazar analizi alanında içerisinde Türkçe'nin de bulunduğu birçok farklı dilde çalışma literatürde mevcuttur, fakat özellikle yazar doğrulama probleminin çözümüne yönelik Türkçe dili üzerinde yapılmış bir çalışma bulunmamaktadır. Bu tez çalışmasının en önemli özgün değeri Türkçe dili ile yapılmış, şu ana kadarki ilk ve tek yazar doğrulama çalışması olmasıdır. Uluslararası çalışmalarda birçok yazar doğrulama yöntemi önerilmiş olmasına rağmen bu tez çalışmasında önerilen yazar doğrulama modellerini kullanan çalışma bulunmamaktadır. Önerilen modelin uygulanmasında İngilizce veri kümesi ile elde edilen başarının bu alanda günümüze kadar elde edilmiş en yüksek sonuç olması da bu tezin özgün değerleri arasındadır. Bu tez çalışmasının çok önemli bir diğer özgün değeri ise, Türkçe üzerine yapılacak yazar analizi çalışmalarının arttırılması ve geliştirilmesi amacı ile bu çalışmalara özgü iyi tanımlı bir külliyat oluşturulmuş olmasıdır. İsteyen her araştırmacıyla paylaşılabilecek olan bu külliyat uluslararası çalışmalarda da kullanılabilecek bir kaynak olacaktır. Böylece yazar analizi çalışmalarında Türkçe üzerinde değerlendirilen yöntemlerin ve bulguların artması

beklenmektedir. Bu çalışmaların sonucunda üretilecek olan akademik çıktılar da ülkemiz literatürüne büyük katkı sağlayacaktır. Bu tezde Yazar doğrulama problemine önerilen çözüm yöntemlerinin dil bağımsız oluşu, dolayısı ile farklı dillere de uygulanabilirliği bir başka özgün değerimizdir.

Klasik yazar doğrulama çalışmaları ele alınan yazarın dokümanlarına yani bilinen yazara göre çözümler sunarken bu tez çalışmasının literatürdeki diğer çalışmalardan en önemli farkı probleme yazar bağımsız bir çözüm sunabilmesidir. Bu çalışmada klasik yazar doğrulama çalışmalarının da kolaylıkla değerlendirilebileceği, üstelik daha faydalı ve daha etkili bir yaklaşım olan iki dokümanın yazarlığının doğrulanması üzerinde durulmuş, problem bu bakış açısıyla ele alınarak daha verimli çözümler sunulmuştur. Ayrıca ele alınan, bilinen ve sorgulanan dokümanların boyutunun olabildiğince kısa olması da çalışmaya ek bir ayrıcalık ve özgünlük katmıştır.

Yazar doğrulama çalışmalarının kapsamı gereği ele alınacak konular, yazar analizi çalışmalarının diğer önemli kolları olan yazar niteliklendirme ve profil çıkarımı çalışmalarını da kapsamaktadır. Bu sebeple yapılan tez çalışması, yazar analizi alanında hem Türkçe hem de başka dilleri kapsayan geniş bir literatür taraması olarak da yardımcı bir kaynak görevi görecektir.

1.5. Tez Organizasyonu

Devam eden kesimler şu şekilde tasarlanmıştır:

- İkinci bölümde, Yazar Doğrulama probleminin kapsama uzayı olan Yazar Analizi çalışmalarının alan bilgisi sunulmaktadır. Bu alanda yapılan çalışmaların büyük oranda birbirine benzemesinden dolayı ele alınan alt problemler sıklıkla birbiri ile karıştırılmaktadır. Bu tez çalışmasında ele alınan yazar doğrulama probleminin, alan içerisinde ele alınan diğer problemler ile karıştırılmaması amacı ile Yazar Analizi alanında ele alınan en temel 3 ana başlık, diğer problemlere olan benzerlikleri ve farklılıkları ile açıklanmış, temsili görseller kullanılarak tanımlamalar güçlendirilmiştir.
- Üçüncü bölümde, tezin genelinde ele almış olduğumuz yazar doğrulama problemine, belirli bir alana özgü "alan temsil modeli" üretimi ile yazar doğrulama yaklaşımı çözüm yöntemi olarak sunulmaktadır. Söz konusu yaklaşım ile belirli bir alanda üretilen yazılara ait bir temsilin o alandaki yazar doğrulama başarısı ölçülmeye çalışılmıştır.

Sunulan yazar doğrulama yöntemi bu çalışmaya özgü olup köşe yazılarının kullanıldığı bir alan modeli ile deneyler yapılmış ve sonuçlar tartışılmıştır.

- Dördüncü bölümde, yazar doğrulama problemlerinde dikkate alınan benzerlik eşik değerinin ne olması gerektiği sorusunun cevabı, yazar modelleri oluşturularak bulunmaya çalışılmıştır. Oluşturulan yazar modelleri, yazarları temsil eden bir doküman birleşimi gibi ele alınıp söz konusu yazara ait dokümanlara olan benzerlik değerlerine göre bir eşik belirlenmeye çalışılmıştır. Ek olarak bir yazarı temsil etmede kullanılan başarılı öznitelik kümelerinin birbirine göre karşılaştırması da yapılmıştır. Köşe yazılarının veri kümesi olarak kullanıldığı bu çalışmada elde edilen deneysel sonuçlar ve yapılan çıkarımlar sözlü bildiri olarak literatüre eklenmiştir.
- Beşinci bölümde, yazar doğrulama probleminin çözümünde kullanılması en uygun veri kümesinin hangi türden olması gerektiği üzerine araştırmalar yapılmıştır. Yazar doğrulama problemi, adli bilişim çalışmalarının bir alt dalı ve gerçek bir dünya problemi olduğundan çözümünde kullanılacak veri kümesinin tür seçimi önemlidir. Kullanılacak en uygun yazı türünün belirlenmesi amacı ile literatürde Türkçe ve başka dillerde hem genel yazar analizi çalışmalarında hem de özel olarak yazar doğrulama çalışmalarında kullanılan veri kümeleri değerlendirilmiş, uygunluk durumları sebepleri ile birlikte açıklanmıştır. Bu araştırma sonucunda yazar doğrulama probleminin çözümünde kullanılmasına karar verilen en uygun yazı türü belirlenmiştir ve daha sonraki çalışmalar bu türde oluşturulan bir külliyat kullanılarak yürütülmüştür.
- Altıncı bölümde, bir önceki bölümde külliyatı oluşturulmasına karar verilen Türkçe Blog veri kümesinin yazar doğrulama probleminin çözümü için önerdiğimiz evrensel bir arka plan modeli üretimi için uygunluğunun ön deneyleri yapılmıştır. Bu deneylerde amaç ele aldığımız veri kümesinin, ele aldığımız özniteliklere göre doğal dağılımını ve davranışını görmektir. Öyle ki; ele aldığımız verilerin önerdiğimiz modele uygun dağılıma sahip olup olmadığı ile ilgili bilgi çıkarımı yapılabilsin. Bu amaç doğrultusunda, ele alınan veriler kullanılarak farklı sayılarda kümeler oluşturulmuş ve verilerin oluşturulan bu kümelere dağılımı incelenmiştir. Bu incelemeler sonucunda ele aldığımız veri kümesi ile önereceğimiz modelin başarılı olup olamayacağı değerlendirilmiştir.
- Yedinci bölümde, yazar analizi çalışmalarının genel bir problemi olan anlamlı metin boyu seçimi üzerinde durulmuştur. Yapılan çalışmanın amacı yazar doğrulama problemlerinde temel adımlardan biri olacak anlamlı metin boyunun belirlenmesidir.

Çalışmada yapılan deneyler ile yazar tanımlamadaki en anlamlı ayırt edici metin boyunun ne olması gerektiği bulunmaya çalışılmıştır. Bulunan anlamlı metin boyunun yazar doğrulama çalışmalarında da başarılı sonuçlar vereceği varsayımı ile çalışmalar sürdürülmüştür. Türkçe Blog yazılarının veri kümesi olarak kullanıldığı bu çalışmada elde edilen deneysel sonuçlar ve yapılan çıkarımlar sözlü bildiri olarak literatüre eklenmiştir.

- Sekizinci bölümde, yazar analizi çalışmalarında elde edilecek başarılı sonuçları etkileyen en önemli parametrelerden öznitelik seçimi ve her yazara ait veri boyutunun etkileri üzerine çalışmalar yapılmıştır. Ele alınan veri kümelerinin dengesiz oluşu yapılan çalışmalardaki güvenilirliği sorgulatmaktadır. Bu çalışmada, veri kümesinin yazarlara ait veriler bakımından standartlaştırılması ve bu işlemlerin yazar tanımlamadaki etkisi ele alınmıştır. Kullanılan veri kümesi hem doğal hali ile hem de standartlaştırılmış hali ile yazar tanımlama işleminde kullanılmış, avantaj ve dezavantajları elde edilen sonuçlar üzerinden değerlendirilmiştir. Türkçe Blog yazılarının veri kümesi olarak kullanıldığı bu çalışmada elde edilen deneysel sonuçlar ve yapılan çıkarımlar sözlü bildiri olarak literatüre eklenmiştir.
- Dokuzuncu bölümde, derlemiş olduğumuz Türkçe Blog Yazıları külliyatı kullanılarak yazar doğrulama probleminin çözümüne yönelik uygulamalar gerçekleştirilmiştir. Bu uygulamalar, yazar doğrulama problemi özelinde önerdiğimiz çözüme yönelik uygulamalar olup, çözümde kullanılması gereken özniteliklerin ve sınıflandırma algoritmalarının seçimine yönelik karşılaştırmalı deneyleri içermektedir. Yazar doğrulama probleminin çözümüne yönelik önermiş olduğumuz yaklaşımın ve kullanmamız gereken araçların belirlendiği bu çalışma, yazar doğrulama problemine evrensel bir çözüm sunmaya çalıştığımız bu tez çalışmasının önemli bir adımı olarak değerlendirilmektedir.
- Onuncu bölümde, Yazar Analizi çalışmaları Adli bilişim kapsamında ele alınmış, bu alandaki çalışmaların genelleştirme yeteneğini ve kullanılan üslupsal özniteliklerin temsil ediciliğini arttırmak için yeni bir yöntem önerilmiştir. Üslupsal özniteliklerin nitel ve nicel özelliklerine göre Derin Sinir Ağları Kombinasyonu (C-DSA) kullanmaya dayalı önerilen bu yöntemin Yazar Doğrulama performansını değerlendirmek için iki farklı umumi İngilizce veri kümesi kullanılmıştır. Elde edilen sonuçlar, önerilen yaklaşımın, probleme özgü sunulan çözümlerin genelleştirme yeteneğini ve güvenilirliğini büyük ölçüde arttırdığını ve hatta tekil Derin Sinir Ağları'ndan daha

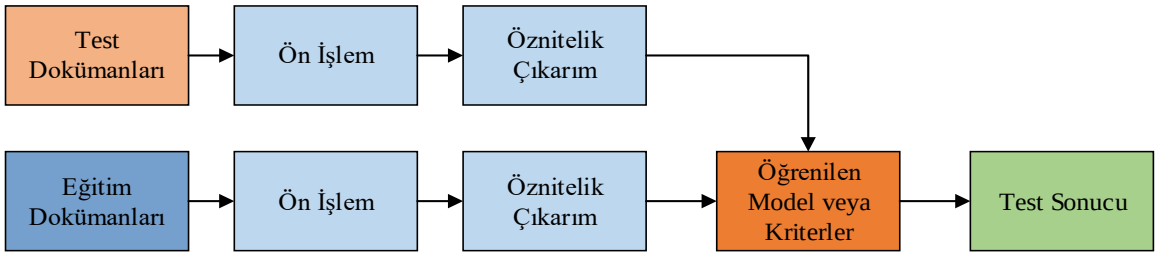
yüksek doğrulukta sonuçlar ürettiğini göstermiştir. Bu çalışmadan elde edilen sonuçlar, bulgular ve çıkarımlar, uluslararası bir dergide yayınlanmak üzere kabul edilmiştir.

- On birinci bölümde, tezin genelinde ele almış olduğumuz güncel yazar doğrulama probleminin çözümüne yönelik yazar bağımsız bir sistem önerilmiştir. Yapılan tez çalışmasının kalbi olan bu bölümde üretilen sistem, içerdiği model itibari ile bilgilendirici bir sistemdir. Önerilen sistem hem Türkçe hem de İngilizce dokümanlar ile test edilmiş, her iki dilde yazılmış dokümanlar için de başarılı sonuçlar üretmiştir. Özellikle İngilizce dilinin test edilmesi için kullanılan veri kümesi ile bildiğimize göre şu ana kadarki en yüksek başarılı sonuç elde edilmiştir. Önerilen sistemin konu ve tür bağımsız oluşu da bu sistemin birçok alanda kolaylıkla uygulanabilmesini sağlamaktadır. Önerilen sistem modelinin bir ön işlem olarak kullandığı doküman normalleştirilmesi ile de hem yazar doğrulama problemlerindeki dengesiz veri sorunu hem de dokümanlardaki zaman – konu bağı kırılarak daha faydalı bir yaklaşım geliştirilmiş ve etkileri gözlemlenmiştir. Bu çalışmada elde edilen sonuçlar, bulgular ve çıkarımlar bu tez çalışmasının ana yayını olarak uluslararası bir dergide yayınlanmak üzere gönderilmiş ve revizyon aşamasındadır.
- On ikinci bölümde, tezin ana çalışması ve çıktısı olan, on birinci bölümde ayrıntıları verilen sistem ile elde edilen sonuçlar değerlendirilmiş ve yorumlanmıştır.

2. YAZAR ANALİZİ ALAN BİLGİSİ

Bu bölümde, Yazar doğrulama probleminin kapsama uzayı olan Yazar Analizi çalışmalarının alan bilgisi sunulmaktadır. Bu alanda yapılan çalışmaların büyük oranda birbirine benzemesi, ele alınan alt problemlerin sıklıkla birbiri ile karıştırılmasına sebep olmaktadır. Bu tez çalışmasında ele alınan yazar doğrulama probleminin, alan içerisinde ele alınan diğer problemler ile karıştırılmaması amacı ile Yazar Analizi alanında ele alınan en temel 3 ana başlık, diğer problemlere olan benzerlikleri ve farklılıkları ile açıklanmış, temsili görseller kullanılarak tanımlamalar güçlendirilmiştir.

Yazarlık üslubunu analiz etme işlemi, üslubun ayırt edicilik niteliklerinin değerlendirilmesi noktasında ölçeklenebilir bir değer olarak ele alınmaktadır [5]. 1800'lü yıllarda ihtilaflı bir yazarlığın belirlenmesinde kelime uzunluklarının frekanslarının ölçülmesi ile başladığı düşünülen üslup analizi çalışmaları, faklı araç ve metotların kullanımı ile günümüze kadar gelmiştir. Bir metnin karakteristiğinin analiz edilmesi ile o metnin yazarı veya yazarları ile ilgili bilgi çıkarma işlemleri yazar analizi olarak adlandırılmaktadır. Temeli stilistik/üslupsal (stylometric) özelliklere dayanan bu sonuç çıkarma işlemlerinde [5, 6] çoğunlukla istatistik ve makine öğrenmesi yöntemleri kullanılmaktadır [7]. Elektronik metinlerden yazarının üslupsal özelliklerini temsil eden ham bilgilerin (özniteliklerin) çıkarılması ve bu bilgilerin istatistiksel yöntemler ile analiz edilmesinden oluşan yazar analizi çalışmalarının genel bir akışı Şekil 2.1'de gösterilmektedir. Çıkarılan özniteliklerin bir yazarın üslubunu temsil etmede ne kadar başarılı olduğu, kullanılan istatistiksel yöntemlerde elde edilen başarılar ile belirlenebilmektedir.



Şekil 2.1. Yazar analizi çalışmalarının genel akışı

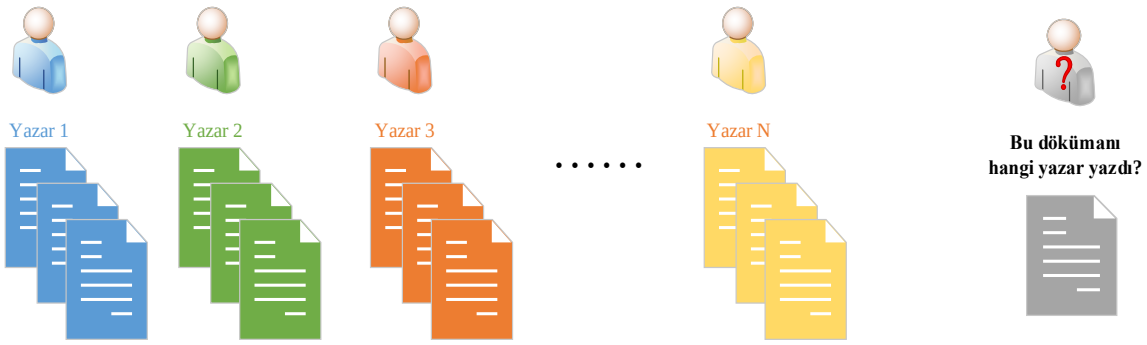
Yazar analizi çalışmaları ele alınan problemlerin yapısına göre alt dallara ayrılmaktadır. Her ne kadar farklı problemlerin çözümüne odaklanmış olsalar da, yazar analizi çalışmaları kullanılan yöntemler bakımından birbirine benzemektedir. Tüm yazar analizi çalışmalarının ortak noktası; ele alınan metinler üzerinden yazarın/yazarların üslupsal özelliklerinin çıkarımı ve bu

özelliklerin bilinen verilere göre yorumlandırılmasıdır. Çalışmalarda ele alınan problemlere özgü yaklaşımlar bu özelliklerin değerlendirilmesi aşamasında çeşitlilik göstermektedir. Elbette ki her problemin çözümünde aynı öznitelik kümeleri kullanılmamaktadır fakat probleme özgü öznitelik kümesinin veya kümelerinin belirlenme işlemi problemlerin büyük bir çoğunluğunda benzerdir. Akademik metinlerde intihal tespiti, yazarı ihtilaflı metinlerin yazarının tespiti, tüketici yorumlarından profil çıkarımı, tehdit/şantaj mektuplarının yazarına ait sosyo-demografik bilgililerin elde edilmesi ve terörist grupların gizli web yazışmalarının tespiti gibi birçok önemli gerçek dünya probleminin çözümü için günümüzde hala yazar analizi çalışmalarına ihtiyaç duyulmaktadır. Yazar analizi çalışmalarında farklı konu ve metinler üzerinden birçok problem ele alınmış olsa da bu alanda en çok ilgi gören 3 temel problem aşağıda verilmektedir. Bu 3 temel problem, tanımları ve karşılaştırmaları alt başlıklar halinde verilmiştir.

- Yazar Tanımlama (Yazar Atfetme-Niteleme)
- Yazar Profil Çıkarımı
- **Yazar Doğrulama (Klasik ve Güncel)**

2.1. Yazar Tanımlama (Yazar Atfetme-Niteleme)

Yazar Tanımlama problemi, temelde çok sınıflı bir sınıflama problemidir. Sorgulanan bir metnin, yazıları bilinen bir grup yazar arasından birine atfedilmesine veya doğru yazarın bulunmasına dayanır [8, 9]. Bu problemlerin çözümünde arka planda birden fazla yazar ve o yazarlara ait üslupsal özelliklerin belirlenebileceği dokümanlar bulunur [3, 10]. Günümüz teknolojisi göz önünde bulundurulduğunda, arka plandaki yazar sayısı dengesiz bir şekilde artmakta fakat her yazara ait doküman sayısı eşit oranda artmamaktadır [11]. Bu durum yapılan çalışmaları zorlaştırıyor olsa da gelişen çözüm yöntemleri de bu çalışmalara olan ilgiyi arttırmaktadır. Yazar tanımlama problemlerinde, sorgulanan dokümanın yazarının arka planda dokümanları bilinen yazarlardan biri olduğu garantilenmiştir. Yani, elimizde bir aday yazar kümesi vardır ve sorgulanan dokümanın yazarının bu aday yazarlardan biri olduğu kesindir. Bu durum aday yazar kümesinin boyutunun da çözümde önemli bir rol oynadığını göstermektedir. A kümesi n tane yazarın bulunduğu bir küme olsun; $A = \{a_1, a_2, a_3, a_4, \dots a_n\}$ ve D kümesi her yazara ait doküman kümelerinin bulunduğu bir küme olsun; $D = \{D_1, D_2, D_3, D_4, \dots D_n\}$, D_i kümesi a_i yazarına ait dokümanların kümesi olup $a_i \in A$'dir. Temsili bir yazar tanımlama probleminin yapısı Şekil 2.2'de gösterilmektedir.

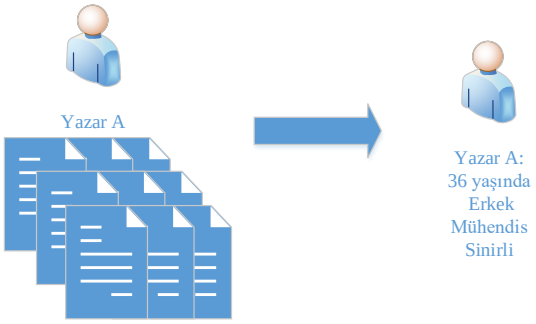


Şekil 2.2. Yazar tanımlama probleminin örnek bir temsili

Yazarı sorgulanan doküman d_s olarak adlandırılırsa, bu çalışmalardaki ana amaç d_s dokümanının en yakın olduğu veya en çok benzerlik gösterdiği D_i kümesini bulmaktır. Böylece $d_s \approx D_i$ koşulunda d_s dokümanı i. yazar tarafından yazılmıştır bilgisine ulaşılmaktadır.

2.2. Yazar Profil Çıkarımı

Yazar Profil Çıkarımı probleminde, belirli bir yazara ait dokümanlar bulunur ve eldeki dokümanlar kullanılarak o yazara ait demografik, sosyal veya kültürel özellikler belirlenmeye çalışılır [12–14]. Genellikle hangi üslupsal özelliğin hangi oranda sorgulanan yazara ait özelliği temsil ettiği, arka planda ya bir uzman görüşü alınarak veya o özellikteki yazarların dokümanlarından gerekli ön bilginin çıkarılması ile elde edilmiştir. Yazar profil çıkarımı çalışmalarında, incelemede tek bir yazar ve sadece o yazara ait dokümanlar bulunmaktadır, yani yazar bazlı sonuçlar üretilmektedir. Temsili bir yazar profil çıkarımı probleminin yapısı Şekil 2.3'te göstertilmektedir.



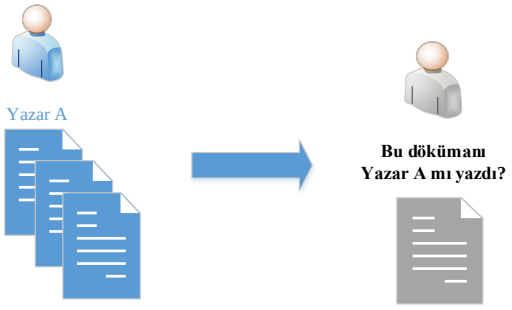
Şekil 2.3. Yazar profil çıkarımı probleminin örnek bir temsili

Yaş, cinsiyet, yabancı dil, kişisel özellik (sosyal, asosyal, eğlenceli, sinirli, obsesif…), eğitim ve çalışma alanı gibi özelliklerin belirlenmeye çalışıldığı yazar profil çıkarımı çalışmalarında

genellikle bu özelliklerde kişilerin dokümanlarının kullanıldığı sınıflar mevcuttur ve çalışmalar, sorgulanan dokümanların hangi sınıflara daha yakın veya daha benzer olduğunu tespit etmeye çalışır.

2.3. Klasik Yazar Doğrulama

Sorgulanan bir dokümanın şüpheli bir yazara ait olup olmadığının, şüpheli yazara ait olduğu bilinen başka dokümanlar kullanılarak doğrulanabilmesi Yazar Doğrulama işlemi olarak adlandırılmaktadır. Sorgulanan dokümanın ait olabileceği diğer yazarlar kümesinin sonsuz boyutu göz önüne alındığında Yazar doğrulama işlemi zorlu bir probleme dönüşmektedir. Yazar Doğrulama problemi, adli bilişim çalışmalarının bir alt dalı olarak ele alınıp özellikle intihal tespitinde kullanılan ve adli bilişim alanında sıkça karşılaşılan bir problemdir [3, 10, 15]. Yazar doğrulama, yazar analizi problemlerinin neredeyse hepsini kapsayacak niteliktedir, çünkü bu problemde, sadece bir yazara ait az miktarda doküman kullanılarak, sorgulanan harici bir dokümanın bu yazara ait olup olmadığının cevabını bulunmaya çalışılır [4]. Sorgulanan doküman, ya az miktarda yazısına sahip olduğumuz bir yazar tarafından yazılmıştır veya yeryüzündeki milyonlarca insandan biri tarafından [9, 16–18]. Yani sorgulanan dokümanın doğrudan atanabileceği garantilenmiş bir aday yazar kümesi bulunmamaktadır. Yapısı gereği tek sınıflı bir sınıflandırma problemi olan bu problemin çözümü farklı problemlere benzetilerek çözülmeye çalışılmaktadır. Temsili bir yazar doğrulama probleminin yapısı Şekil 2.4'te göstertilmektedir.



Şekil 2.4. Klasik yazar doğrulama probleminin örnek bir temsili

Klasik yazar doğrulama probleminin çözüm yaklaşımlar, yukarıda da anlatıldığı üzere bir dokümanın şüpheli bir yazara ait olup olmadığı sorusunun cevabını bulmaya yöneliktir. Bu problem üzerine yapılan çalışmalar genellikle bir yazara ait dokümanların (az bir miktar) arasındaki benzerliği dikkate alarak sorgulanan dokümanın da bilinen dokümanlar ile yakın

benzerlik gösterip göstermediği üzerinden çözümler sunmaktadır [14]. Çoğunlukla şüpheli yazarın bilinen dokümanları arasındaki ortalama benzerlik değeri eşik değer olarak alınır ve sorgulanan dokümanın bilinen dokümanlara olan ortalama benzerlik eşiğinin altında veya üstünde olması durumu yorumlanır. Eğer sorgulanan dokümanın bilinen dokümanlara olan benzerliği belirlenen eşik değerin altında ise "sorgulanan doküman şüpheli yazar tarafından yazılmamıştır" denir. Tersi durumda, yani sorgulanan dokümanın bilinen dokümanlara olan ortalama benzerlik değeri belirlenen eşik değerden yüksek çıkar ise "sorgulanan doküman şüpheli yazar tarafından yazılmıştır" denir [14, 19].

2.4. Güncel Yazar Bağımsız Yazarlık Doğrulama

Klasik yazar doğrulama probleminde, şüpheli yazara ait bir miktar doküman bulunması bu problemi yazar tanımlama problemlerinin temeli olarak ele alınmasını sağlamaktadır [20]. Bir dokümanın belirli bir yazara ait olup olmadığının sorgulanması standart yazar tanımlama problemlerinin göre daha zorlu bir probleme dönüşmesini sağlamaktadır. Klasik yazar doğrulama problemlerinin, yazar tanımlama problemlerine benzerliği göz önüne alındığında birçok çalışma yazar doğrulama problemini yazar tanımlama problemi gibi çözme yaklaşımında bulunarak kavram karmaşası oluşmasında da sebebiyet vermiştir [21]. İki temel yaklaşımın arasındaki tek fark aday yazar kümesidir. Yazar tanımlama çalışmalarında birçok şüpheli yazar varken ve sorgulanan dokümanın bu yazarlardan biri olduğu garantilenmişken, klasik yazar doğrulama probleminde tek bir şüpheli yazar ele alınır ve sorgulanan dokümanın bu yazara ait olmaması durumunda sayısız adaya ait olabilme ihtimali vardır.

Güncel yazar doğrulama problemi, yazarı bilinmeyen (anonim) iki dokümanın aynı yazar tarafından yazılıp yazılmadığının doğrulanması problemidir [22]. Bu yaklaşımda, klasik yazar doğrulama problemlerinden farklı olarak yazar bağımsız bir çözüm gerekmektedir. Bu yaklaşıma göre arka planda şüpheli yazara ait birkaç doküman yoktur ve amaç sorgulanan iki dokümanın aynı yazar tarafından yazılıp yazılmadığı bilgisini çıkarabilmektir. Klasik yazar doğrulama bakış açısı ile güncel yazar doğrulama problemi, şüpheli yazara ait tek bir dokümanın, sorgulanan bir dokümanı da aynı yazarın yazıp yazmadığını doğrulayabilmesi üzerinedir. Güncel yazar doğrulama yaklaşımında yazar bağımlı çözümlerin yerine daha genel bir çözüm üretilmesi zorunluluğu vardır. Tek bir doküman kullanılarak şüpheli yazara ait bir üslup analizinin yapılmasının zorluğu aşikardır. Bu çalışmalarda, aynı yazarlara ait dokümanlar ile farklı yazarlara ait dokümanların karşılaştırılması gibi daha genel bir ayırt edicilik tespit

edilmelidir. Literatürde güncel yazar doğrulama bakış açısı ile seçili öznitelikler kullanılarak iki doküman arası benzerlik eşik değeri belirleme çalışması yapılmıştır [23]. Bu çalışma başarılı sonuçlar üretmiş olsa da her yazarın dokümanları arasındaki benzerliğin aynı oranda ele alınmaması gerekliliği, bazı yazarların dokümanları arasında daha ayırt edici benzerliklerin aranması gerekliliği Bölüm 4'te yapılan deneysel çalışmalar sonucunda görülmüştür. Temsili bir yazar bağımsız yazar doğrulama probleminin yapısı Şekil 2.5'te göstertilmektedir.

Şekil 2.5. Güncel, yazar bağımsız yazarlık doğrulama probleminin örnek bir temsili

Güncel, yazar bağımsız yazarlık doğrulama probleminde şüpheli yazar veya yazarlar arka planda kalmaktadır. Bu yaklaşımda önemli olan belirli bir yazara veya bir grup yazara ait bir çözüm üretmek değil, genel olarak yazma üslubunda yazarları en derinden ayırt edebilecek bir yapıda çözüm sunabilmektir. Bu genel yapıyı elde edebilmek tahmin edileceği üzere zaten zorlu olan klasik yazar doğrulama problemini daha da zorlu hale getirmiştir. Bu amaç doğrultusunda bu tez çalışmasında birçok farklı yöntem ve uygulama gerçekleştirilmiş, en uygun ve en başarılı sonuç elde edilene kadar birçok farklı algoritma ve model denenmiştir. Yapılan her çalışma belirlenen amaca ulaşmak adına bir adımın sonucunu üretirken bağımsız alt yapılar olarak farklı başlıklarda, alt bölümler halinde ele alınmıştır.

2.5. Diğer Yazar Analizi Çalışmaları

Yazar analizi çalışmaları günümüze kadar yukarıda bahsedilen üç ana başlık üzerinde yoğunlukla yürütülmekteyken, günümüz teknolojik ve sosyal gelişmeler ışığında birçok farklı çalışmaları da içerisinde barındırmıştır. Yapılan alan yazın özeti (survey) araştırmalarının çoğu yukarıda bahsedilen üç başlığı ele almışken [3, 7, 10], yapılan en güncel alan yazın özet çalışması [5] üslupsal analiz çalışmalarını 5 ana başlıkta toplamıştır. Bu başlıklar aşağıdaki gibidir:

- Yazar Tanımlama

- Yazar Profil Çıkarımı
- Yazar Doğrulama
- Üslup Ölçekleme (Stylochronometry)
- Ters Üslup Ölçümü (Adversarial Stylometry)

İlk üç ana problem yukarıda detaylandırılmış, son iki problem daha seyrek çalışmalarda görülmesine rağmen ele alınış biçimleri sebebi ile farklı dallanmalar oluşturmuştur. Üslup ölçekleme temel olarak yazma üslubunun zaman içerisinde değişimi üzerine yapılan çalışmaları kapsamaktadır. Özellikle kullanılan dilin zaman içerisinde değişiminin büyük etken olarak ele alındığı bu çalışmalar, bu değişimin yazarın üslubuna etkisini araştırmaktadır. Türkçe dili üzerine yapılan, iki yazarın üslubunun zaman içerisinde değişiminin kullanılan kelime uzunluklarına etkisini gözlemleyen çalışma [24], güncel üslupta kullanılan kelimelerin daha uzun olduğunun tespitini içermektedir. Zamanla yazarın üslubunun değişiminin doğal olarak kaçınılmaz olduğunu gösteren bu çalışmaların [25, 26] yazarlara ait en ayırt edici üslupların belirlenmesi konusunda literatüre önemli bir katkı sağlayacağı aşikardır.

Ters üslup ölçümü olarak Türkçeye çevirebildiğimiz çalışmalar, bir yazarın, yazar tanımlama çalışmalarında tespit edilmemek için kasıtlı olarak üslubunu değiştirmesi üzerine yapılan tespit çalışmalarıdır [27]. Üç farklı yapıda ters ölçme yapılabilmektedir; imitasyon, çeviri ve perdeleme. İmitasyon işlemi bir yazarın üslubunu başka bir yazara benzeterek metin oluşturma işlemlerini kapsamaktayken, çeviri işlemi, dil çeviri uygulamaları kullanılarak bir metni önce başka bir dile sonra tekrar orijinal diline çevirme işlemidir. Perdeleme çalışmalarında ise yazar tanımlama çalışmalarında yazarları en iyi ayırt edebildiği bilinen kelime veya özniteliklerin bir yazar tarafından farklı değerlerde kullanımı ile yapılan çalışmalardır ki bu çalışmalar da yazarın yazma üslubunu perdelemek amacı ile yapılmaktadır [28].

3. ALAN MODELLEME İLE YAZAR DOĞRULAMA

Bu bölümde, tezin genelinde ele almış olduğumuz yazar doğrulama problemine, belirli bir alana özgü "alan temsil modeli" üretimi ile yazar doğrulama yaklaşımı çözüm yöntemi olarak sunulmaktadır. Söz konusu yaklaşım ile belirli bir alanda üretilen yazılara ait bir temsilin o alandaki yazar doğrulama başarısı ölçülmeye çalışılmıştır. Sunulan yazar doğrulama yöntemi bu çalışmaya özgü olup köşe yazılarının kullanıldığı bir alan modeli ile deneyler yapılmış ve sonuçlar tartışılmıştır.

3.1. Konu Kapsam ve Literatür

Suçluların veya terörist grupların sanal ortamın gizliliğinden faydalanmasıyla dijital dokümanlardaki yazar doğrulama problemi giderek daha önemli bir hal almaktadır. Elektronik ortamda dijital dokümanlar, eğer yazarı bilinirse şüphelileri suçlamak için delil niteliğindedir. Özellikle adli vakalarda bir dokümanın yazarını belirleyebilmek için arka planda o yazara ait çok sayıda doküman yoktur. Bir metnin bir yazar tarafından yazılıp yazılmadığının doğrulanabilmesi için az miktarda dokumanın yeterli olması gerekmektedir. Klasik yazar doğrulama çalışmalarında bu miktar 3 ile 10 arasında olmaktadır. Fakat bu miktar bile çoğu gerçek dünya probleminde tedarik edilemeyecek kadar çoktur. Söz konusu zorluk göz önüne alındığında problemin en temel noktaya yani bir yazara ait sadece bir dokümanın bulunduğu duruma indirgenmesi gerekliliği görülmektedir. Bu gereklilik göz önünde bulundurularak yapılan bu çalışmada, verilen sadece iki dokümanın aynı yazar tarafından yazılıp yazılmadığı bilgisine ulaşılmaya çalışılmıştır. Bu yaklaşım ile ele alınan temel problem, yazarı bilinen bir dokuman kullanılarak başka bir dokümanın aynı kişi tarafından yazılıp yazılmadığının doğrulanabilmesi noktasında çözülmeye çalışılmıştır.

Ele alınan tek dokümanlı yazar doğrulama probleminin çözümü için bu çalışmada "Alana-özgü Model Üretimi" çözüm yöntemi olarak sunulmaktadır. Söz konusu alan ile ilgili, o alanı temsil eden bir model üretilip verilen bir dokümanın alan doğrulamasının yapılabilmesi öngörülmektedir. Çalışmanın temel amacı ise üretilen model ile yazar doğrulama yapmaktır. Arka planda etiketli hiçbir veri olmadan, üretilen model kullanılarak, verilen iki dokümanın aynı yazar tarafından yazılıp yazılmadığının cevabı, yazar doğrulama probleminin çözümünü ve bu çalışmanın amacını oluşturmaktadır.

Belirtilen amaç doğrultusunda ilk olarak belirli bir alan için genel yapıyı temsil eden bir model oluşturmak gerekmektedir. Örneğin blog yazılarını temsil etmesi için üretilecek bir model girilen bir metnin blog yazısı olup olmadığını olasılıklar çerçevesinde belirleyebilecektir. Böylece, özellikle örgütsel mesaj veya bildirilerin kolaylıkla entegre edilebildiği bu yapıların tespiti kolaylaşacaktır. Daha sonra üretilen model veya modeller kullanılarak yazar doğrulama sistemi oluşturması gerekmektedir. Yazar doğrulama işlemi, adli bilişimin bir alt dalı olmakla birlikte yazar analizi problemlerinden en zorlu olanıdır. Yazar analizi işlemleri temelde çok sınıflı bir sınıflandırma problemi olup arka planda söz konusu yazar veya yazarlar ile ilgili birçok dokümandan bilgi elde edilmiştir. Fakat yazar doğrulamada söz konusu yazar ile ilgili çok az miktarda doküman ele alınır ve harici bir metnin söz konusu yazar tarafından yazılıp yazılmadığı sorgulanır. Temel yazar analizi işlemlerinin aksine yazar doğrulama problemi bir tek sınıflı sınıflandırma problemi olarak ele alınır. Bu çalışmada üretilecek model kullanılarak başarılı bir yazar doğrulama sistemi yapılması hedeflenmiştir.

Ulusal ve Uluslararası literatürde Türkçe dili için yapılan bir yazar doğrulama çalışması bulunmamaktadır. Uluslararası literatürde yazar doğrulama için birçok çalışma yapılmış olmasına rağmen önerdiğimiz metodu kullanan çalışma bulunmamaktadır. Uluslararası literatürde yapılan çalışmalar, kullanılan yöntemler ve veri kümeleri aşağıda detaylandırılmıştır.

Yazar doğrulama problemini tek sınıflı sınıflandırma problemi olarak ele alan çalışmalarda yazar doğrulama için sahtekarlar (impostors) [23, 29, 30] ve maske sıyırma (unmasking) [31–34] yöntemleri yoğunlukla kullanılmaktadır. Sahtekarlar yönteminde, sorgulanan dokümanın konusu ve alanı aynı olan başka yazarlara ait dokümanlar toplanarak iki sınıflı bir yapı elde edilmeye çalışılmış ve sınıflandırma algoritmaları kullanılarak yazar doğrulama gerçekleştirilmiştir. Bu yöntemde elde iki sınıf bulunmaktadır; A yazarına ait yazılar ve A yazarına ait olmayan yazılar. Sahtekarlar yöntemindeki en önemli problem, B yazarına ait bir yazının A yazarına ait olmayan yazılar sınıfında olmamasıdır yani sahtekarlık ile üretilen dokümanlara benzemeyen her yazı A yazarına aittir sonucu üretilecektir. Literatürde sahtekarlar yöntemi farklı şekillerde kullanan çalışmalar da bulunmaktadır [30]. Maske sıyırma yönteminde ise bir yazarın dokümanları arasındaki benzerliği bulmak için çıkarılan özniteliklerden en ayırt edici olanlar aşamalı olarak öznitelik kümesinden çıkarılmaktadır. Böylece aynı yazarın dokümanları arasında en yüksek benzerlik değerini üreten öznitelikler ve

kaç aşama ile elde edildiği bulunmuştur [15]. Elde edilen öznitelik kümesi yazar doğrulamada kullanılarak başarım değerlendirilmektedir.

Uluslararası literatürde yazar doğrulama çalışmalarında veri kümesi olarak; romanlar [10, 35], blog yazıları [23], elektronik postalar [17, 36], kısa mesajlar [36, 37], farklı konu ve türlerde metinler [18], internet forum mesajları [38], dijital kanıt soruşturmaları [39], intihal belgeleri [40], kaynak kodlar [41, 42] vb. kullanılmaktadır. Ulusal literatürde de yazar niteleme çalışmalarında kullanılan öznitelikler bu çalışmada dikkate alınacaktır. Başlangıç olarak köşe yazılarının alan olarak ele alındığı yazar çözümleme çalışmasında [43] en ayırt edici olduğu tespit edilen özniteliklerin sonuçları değerlendirilecektir. Daha sonra aşamalı olarak Türkçe üzerine yapılan yazar niteleme alanındaki çalışmaların öznitelik kümeleri denenerek yazar doğrulamadaki başarımları değerlendirilecektir.

3.2. Kullanılan veri kümesi

Bu çalışmada ele alınacak metinsel alan Köşe Yazıları alanı olacaktır. Oluşturulacak modelin başarısının ilk denemeleri az sayıda köşe yazarına ait yazılar üzerinden gerçekleştirilecektir. Elde edilen sonuçlar üzerinde yazar sayısı ve ele alınan metinsel alan sayısı arttırılarak çalışmanın başarısının değerlendirilebilmesi öngörülmektedir. İlk aşamada rastgele seçilmiş 5 köşe yazarına ait rastgele seçilmiş 12'şer köşe yazısı, ele alınan köşe yazıları alanının temsili için kullanılacaktır. Seçilen yazarlar ve bu yazarların aktif olarak yazılarını yayınladığı gazete listeleri Tablo 3.1'de verilmiştir.

Tablo 3.1. Köşe yazıları alan temsilinde ele alınan yazarlar ve yazıların yayınlandığı gazeteler listesi

Yazar	**Gazete**
Ahmet Hakan	Hürriyet
Bekir Coşkun	Sözcü
Mehmet Tezkan	Milliyet
Nihal Bengisu Karaca	Habertürk
Serpil Çevikcan	Milliyet

Yukarıda listesi verilen yazarların 2016 yılına ait rastgele seçilmiş 12 yazısı köşe yazıları alanının temsili için kullanılmıştır. Yazarların seçiminde de bir kasıt ve kısıt yoktur, yazarlar ve yazıları rastgele seçilmiş olup 2016 yılındaki her ay için rastgele bir yazı alınmaya

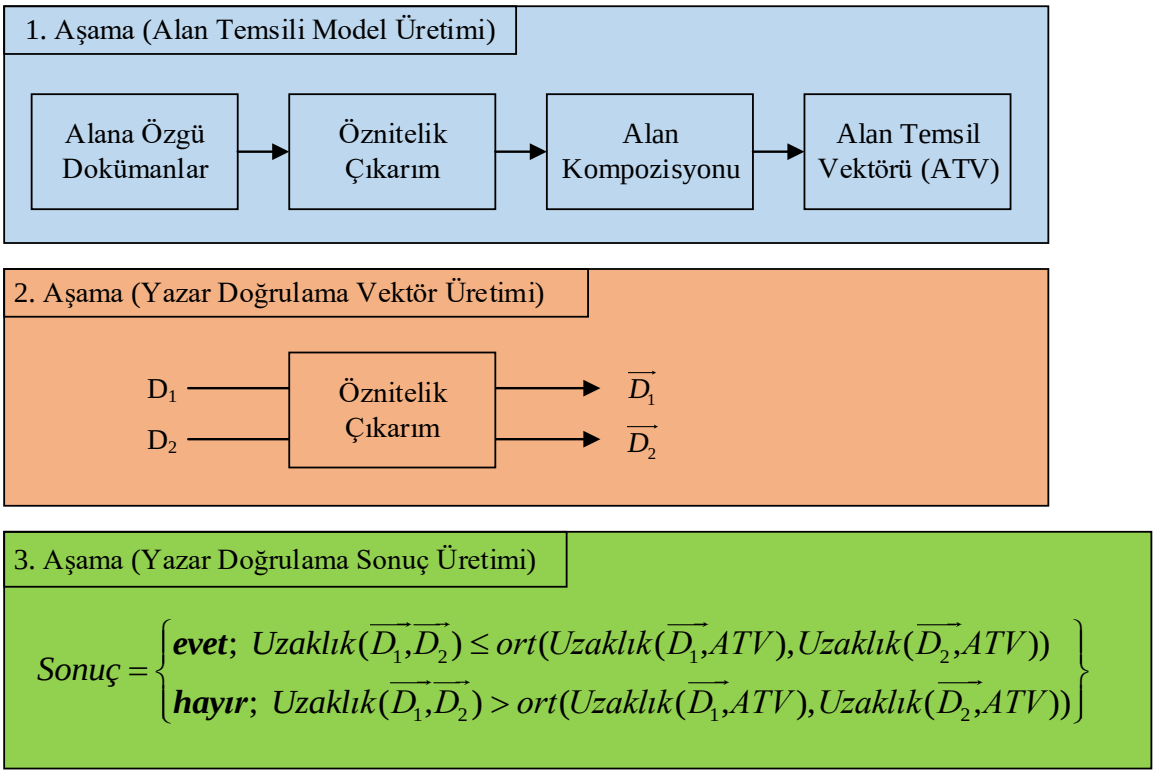
çalışılmıştır. Böylece dokümanlar arası zamana bağlı benzerlik çıkarımının önüne geçmek amaçlanmıştır. Çalışmanın devamında ele alınan verilere ait olduğu yazarların isimleri ile değil rastgele olarak atanan A1, A2, A3, A4 ve A5 yazarı adlandırmaları ile değinilecektir.

3.3. Materyal ve Metot

Bu çalışmanın iki temel aşaması bulunmakta ve her aşama kendi içinde iş parçalarına bölünerek artımlı bir ilerleme göstermesi öngörülmektedir. Çalışmanın ilk aşaması belirlenen bir alan için bu alanı temsil eden bir Türkiye modeli oluşturmaktır. Bu aşamanın 2 önemli iş parçacığı bulunmaktadır. İlki veri kümesi olarak o alanı en iyi temsil eden verilerin yeterli miktarda toplanmasıdır. Bu iş parçacığının başarılı bir şekilde yapılıp yapılmadığını belirleyebilecek bir araç veya algoritma literatürde olmadığından her güncelleme için tüm sistem test edilip sonuçlara göre veri kümelerinin başarımı değerlendirilecektir. İkinci iş parçacığı ise elde edilen veri kümesinin en ayırt edici öznitelikleri kullanılarak bu verileri ortak bir yapı haline getirmektir. Üretilecek bu ortak yapı verilen iki dokümanın yazar doğrulamasında karşılaştırmalı olarak kullanılacağından dokümanlar ile aynı yapıda olması gerekmektedir. Karşılaştırma işlemi dokümanların vektörel temsilleri ile yapılacağından üretilecek Alan Temsil Modeli'nin de ele alınan dokümanlar ile aynı yapıda bir vektör haline dönüştürülmesi gerekmektedir. Söz konusu alanı temsil etmesi için üretilen model/vektör, çalışmanın devamında Alan Temsil Vektörü (ATV) olarak adlandırılmıştır. Bu iş parçacığının başarımı söz konusu özniteliklerin eldeki veri kümesi için en ayırt edici nitelikte olmasına bağlıdır. Birçok başarılı öznitelik çıkarma algoritmalarına ek olarak yine başarımı yükseltmek için literatürde kullanılan öznitelikler uygulanıp sonuçlar farklı öznitelik grupları için sonraki çalışmalarda değerlendirilecektir.

Çalışmanın ikinci aşaması üretilen modellerin yazar doğrulama amacı ile kullanımıdır. Bu aşama öncelikle birinci aşama için bir test görevi görmektedir. Üretilen model yazar doğrulamada ne kadar başarılı sonuç verirse modelin o kadar başarılı bir temsil olduğu söylenebilecektir. Yazarlarının doğrulanması amacı ile sisteme girdi olarak verilecek iki dokümanın, ilk olarak ATV'yi oluşturan dokümanlara yapıldığı gibi bir öznitelik çıkarımı işlemi uygulanmaktadır. Bu işlem sonrası elde edilen doküman temsili vektörler ve ATV, yazar doğrulama işlemi için karşılaştırmalı olarak üçüncü aşamada değerlendirilmektedir. Kurgulanan aşamaları içeren sistemin akışı Şekil 3.1'de gösterilmektedir. Çalışmanın

devamında Şekil 3.1'de gösterilen aşama 2 ve aşama 3 bir bütün olarak değerlendirilecek olup, anlatımın daha sadeleştirilmesi amacı ile şekilde ayrı iş parçacıklarında gösterilmektedirler.



Şekil 3.1. Alana-özgü model üretimi ile yazar doğrulama sisteminin akışı

Köşe yazıları alanında bir Türkiye modeli üretebilmek için öncelikle bu alanda ayırt ediciliği en yüksek özniteliklerin belirlenmesi gerekmektedir. Çalışmanın bu aşamasında 2014 yılında O. Aslantürk tarafından yapılan yazar çözümleme konulu doktora çalışması sonucunda elde edilen ve köşe yazarları arasında en ayırt edici olduğu belirlenen öznitelikler kullanılmıştır. Söz konusu çalışmada yazarlar arası en ayırt ediciliği yüksek özelliklerin Tablo 3.2'de verili özniteliklerin kullanım sıklıkları (frekansları) olduğu belirlenmiştir.

Tablo 3.2. Köşe yazarları alanında ayırt ediciliği en yüksek öznitelikler

Nokta (.)	Soru işareti (?)	Ters Yan Çizgi (\)	Parantez ((,))
Virgül (,)	Yan çizgi (/)	İki nokta üst üste (:)	Ampersand (&)
Alt tire (_)	Tek tırnak(')	Noktalı Virgül (;)	
Tire (-)	Çift tırnak (")	Ünlem işareti (!)	

K kümesi köşe yazıları alanının temsili olmak üzere bu çalışma için küçük bir temsil olacak şekilde 5 yazar içermektedir K = {A_1, A_2, A_3, A_4, A_5}. D_i kümesi A_i yazarına ait dokümanların

bulunduğu küme olmak üzere her $D_i = \{d_1, d_2, d_3, \ldots, d_{12}\}$ şeklinde 12 elemana sahiptir. Seçili öznitelikler kullanılarak ele alınan bu 60 doküman vektörel hale getirilmiştir. Bu işlem, ele alınan özniteliklerin dokümanda geçme frekansları elde edilerek yapılmıştır. İşlem sonucunda her doküman $d_i = < f_1, f_2, f_3, \ldots f_{14} >$ şeklinde bir vektör ile temsil edilmektedir. Daha sonra bu vektörler normalizasyon işleminden geçirilmiştir. Normalize etme işlemi vektörlerin her bir özniteliği için aldığı değerin dağılımdaki yerini bozmadan 0-1 aralığına indirgenmesi ile yapılmıştır. Her öznitelik için normalize edilmiş değer o özniteliğin diğer dokümanlardaki en yüksek ve en düşük değerleri göz önünde bulundurularak Denklem (1)'de olduğu gibi hesaplanmıştır. f_x, x numaralı öznitelik olmak üzere, $d_i(f_{min})$ ve $d_i(f_{maks})$ sırasıyla i. doküman vektöründe bulunan en düşük ve en yüksek değerdeki özniteliğin değeridir.

$$f_x = \frac{f_x - d_i(f_{min})}{d_i(f_{maks}) - d_i(f_{min})} \tag{1}$$

Normalize edilmiş doküman vektörleri birleştirilerek bir köşe yazıları alan temsil vektörü (ATV) elde edilmiştir. Birleştirme işlemi için her vektörün ilgili özniteliğinin tüm vektörlerdeki değerinin toplanıp, toplam değerin vektör sayısına bölümü ile ATV vektörünün ilgili özniteliğinin değeri elde edilmiştir. Birleştirme işlemi ile elde edilen vektörün her özniteliğinin değeri Denklem (2)'de görüldüğü gibi hesaplanmıştır. Köşe yazıları alanını temsil edecek vektörü oluşturan her öznitelik değeri hesaplandıktan sonra elde edilen vektör, köşe yazıları alanının temel bir temsili olarak ele alınmıştır.

$$\mathrm{ATV}(f_x) = \frac{1}{N}\sum_{k=1}^{N} d_k(f_x) \tag{2}$$

f_x = x. özniteliğin değeri
N = doküman sayısı
d_k = k. dokümanın vektörü

ATV elde edildikten sonra veri kümesinde bulunan her doküman diğer tüm dokümanlar ve ATV ile karşılaştırılması ile yazar doğrulama yapılmıştır. Bu aşamada bahsedilen karşılaştırma işlemi uzaklık tabanlı bir karşılaştırmadır. Üretilen alan temsili model kullanılarak model - doküman ve doküman – doküman karşılaştırması yapılmıştır. Yapılan karşılaştırmada, karşılaştırma ölçeği olarak Denklem (3)'te verilen Kosinüs Benzerliği formülü kullanılmıştır. Kosinüs benzerliği kullanılarak girilen iki vektörün n boyutlu vektörel uzayda arasındaki uzaklık ölçülebilmektedir. Karşılaştırma yapılan doküman vektörleri ve üretilen ATV aynı

yapıda ve aynı boyutta vektörler olduğundan bu vektörler arasındaki mesafenin ölçümü için, hem dokümanlar arası mesafe doküman çiftleri olarak hesaplanmış hem de ele alınan her dokümanın üretilen ATV ile aralarındaki mesafe ölçülebilmiştir.

$$Cos\theta = \frac{\mathrm{a}^{\rightarrow}.\mathrm{b}^{\rightarrow}}{||\mathrm{a}^{\rightarrow}||\ ||\mathrm{b}^{\rightarrow}||} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2}\ .\sqrt{\sum_1^n b_i^2}} \quad (3)$$

$\mathrm{a}^{\rightarrow}, \mathrm{b}^{\rightarrow},$ *iki doküman vektörü olmak üzere,*

$a^{\rightarrow}.b^{\rightarrow}$ *bu vektörlerin noktasal çarpımını temsil etmektedir.*

Köşe yazıları alanını temsil eden vektör elde edildikten sonra yazar doğrulama problemindeki başarısı literatürde var olmayan bir yöntem kullanılarak değerlendirilmiştir. Bu çalışmada yazar doğrulama problemi, verilen iki dokümanın yazarının doğrulanabilmesi olarak ele alınmıştır. Bu doğrultuda her yazar için ele alınan iki doküman arasındaki benzerlik ile bu iki dokümanın üretilen alan modeline olan ortalama benzerliği karşılaştırılmıştır. Ele aldığımız probleme çözüm önerimiz şu şekildedir;

Ele alınan iki doküman arasındaki benzerlik değeri, bu dokümanların alan modeline olan benzerliğinden daha yüksek ise ele alınan dokümanlar aynı yazar tarafından yazılmıştır. Eğer söz konusu dokümanlar arasındaki benzerlik değeri bu dokümanların alan modeline olan ortalama benzerliğinden daha düşük ise ele alınan dokümanlar farklı yazarlar tarafından yazılmıştır.

Verilen yaklaşım kullanılarak önerilen çözümün başarım ölçümü için, bilgi getirimi çalışmalarında sıkça kullanılan bir istatistiksel ölçüm aracı olan f-ölçümü kullanılmıştır. F-ölçümü hesaplaması, elde edilen sonuçların kesinlik (precision) Denklem (4), hassasiyet (recall) Denklem (5) ve doğruluk (accuracy) Denklem (6) hesaplamaları üzerinden ele alınmaktadır. Bu hesaplamalar Tablo 3.3'te verilen hata matrisi değerleri kullanılarak elde edilir. F-ölçümü Denklem (7)'de verilmiştir.

Tablo 3.3. Hata matrisi

		Tahmin Edilen Değer	
		Pozitif	Negatif
Gerçek Değer	Doğru	DP	DN
	Yanlış	YP	YN

$$Kesinlik = \frac{DP}{DP + YP} \tag{4}$$

$$Hassasiyet = \frac{DP}{DP + YN} \tag{5}$$

$$Doğruluk = \frac{DP + DN}{DP + DN + YP + YN} \tag{6}$$

$$F - ölçümü = 2 * \frac{Kesinlik * Hassasiyet}{Kesinlik + Hassasiyet} \tag{7}$$

3.4. Sonuçlar

Girilen iki dokümanın yazar doğrulamasının yapılabilmesi için bu çalışmada üretilen model ile veri kümesindeki tüm dokümanlar çiftler halinde karşılaştırılmıştır. Dokümanlar arası karşılaştırmalar yapılırken öncelikle tüm dokümanların birbiri arası benzerlikleri ölçülmüştür. Her doküman çifti için sadece bir benzerlik değeri elde edilebildiğinden dolayı 60 doküman için toplamda 60*60/2 olmak üzere 1800 benzerlik karşılaştırması yapılmıştır. Doküman – doküman arası 1800 benzerlik ölçülürken, her dokümanın üretilen modele olan benzerliği de hesaplanmıştır. Toplamda 60 doküman – model benzerliği elde edilmiştir. İki dokümanlı yazar doğrulama problemine getirdiğimiz çözüm önerisi denkleminde verilen kurallar, elde edilen benzerlik sonuçlarına uygulanmıştır. Doküman – doküman benzerlik oranlarının karşılaştırıldığı matrisin boyutu büyük olduğundan tablo olarak verilememektedir. Ele alınan 60 dokümanın üretilen köşe yazıları alan modeline ortalama, en yüksek, en düşük benzerlik değerleri ve tüm değerler arası standart sapması Tablo 3.4'de verilmiştir.

Tablo 3.4. Üretilen köşe yazıları modelinin veri kümesindeki dokümanlara benzerliği bakımından özellikleri

Model Özellikleri	Değerler
Ortalama benzerlik değeri	0,676
En düşük benzerlik değeri	0.53
En yüksek benzerlik değeri	0,84
Standart Sapma	0,083

Yukarıda belirtilen sonuçlara bakarak ele alınan veri kümesi küçük bir örneği temsil etmiş olmasına rağmen üretilmiş model, üretildiği alanı kısmen temsil edebilecek benzerlik aralığına sahiptir. Bu değerler doğrultusunda + - %15'lik bir değer aralığında alana özgü dokümanların tespitinin yapılabileceği öngörülmektedir.

Yazar doğrulama problemine çözüm önerisi olarak sunulan hipotezin elde edilen benzerlik sonuçlarına uygulanması ile Tablo 3.5'deki hata matrisi elde edilmiştir. Toplamda hesaplanan 1800 doküman – doküman benzerlik değerinin 360 tanesi aynı yazara ait doküman çiftlerinin benzerliklerinden elde edilmekteyken kalan 1440 benzerlik değeri farklı yazarlara ait doküman çiftlerinin benzerlik değeridir.

Tablo 3.5. Kullanılan veri kümesindeki dokümanlardan yazar doğrulama yaklaşımında elde edilen sonuçlar

	Pozitif	Negatif
DOĞRU	354	866
YANLIŞ	6	574

Tablo 3.5'de gösterilen sonuçlar, önerilen hipotezin eldeki veri kümesine uygulanması sonucu elde edilen sonuçlardır. Bu sonuçlar değerlendirildiğinde, toplamda 360 tane olan aynı yazar tarafından yazılmış doküman çiftinin önerilen hipotez kullanılarak 354 tanesine doğru olarak ulaşılabilmektedir. 360 doğru cevaptan sadece 6 taneyi yanlış olarak belirlemek yani %2,16 gibi bir hata payı bu ölçekte bir veri kümesi için oldukça düşüktür. Fakat önermiş olduğumuz hipotez kullanılarak oluşturulan çalışma, toplamda 1440 tane olan farklı yazarlar tarafından yazılmış doküman çiftlerinden 574 tanesine "farklı yazarlar tarafından yazılmıştır" diyebilmiştir. Geri kalan 866 tane farklı yazarlar tarafından yazılmış doküman çifti önermiş olduğumuz çözüm yöntemi kullanılarak hatalı sınıflandırılmıştır. %60,13 değerinde bir yanlış

tahmin bu ölçekte bir veri kümesi için oldukça fazladır. Verilen hata matrisi kullanılarak hesaplanan gerekli ölçekler aşağıda verilmiştir.

- **Precision** = 0,983
- **Recall** = 0,381
- **Accuracy** = 0,677
- **F-ölçeği** = 0,549

3.5. Tartışma

Bu çalışmanın öncelikli özgün değeri, Türkçe dili üzerine ve Türkiye'deki uygulamaları devam eden alanlar gözetilerek alan tabanlı metinsel bir model üretilmesidir. Ulusal literatürde bu konu ile ilgili çalışma bulunmadığından hem akademik alana önemli bir katkı sağlanacak hem de bu alanda yapılacak çalışmalar için bir temel yapı oluşturması öngörülmektedir. Çalışmanın bir başka özgün değeri ise üretilecek modeller kullanılarak hem alan doğrulama hem de yazar doğrulama yapılabilmesidir. Yazar doğrulama çalışmaları uluslararası literatürde ele alınmış olmasına rağmen önerdiğimiz alan bağımlı model tabanlı yazar doğrulama yöntemi bu çalışmaya özgündür. Çalışmanın kapsamı gereği ele alınacak konular yazar niteliklendirme çalışmalarını kapsamaktadır. Bu sebeple çalışma, yazar niteliklendirme alanında geniş bir literatür taraması olarak da yardımcı bir kaynak görevi görecektir.

Ele alınan problemin çözümüne yönelik geliştirilen hipotez doğrultusunda yapılan deneylerden elde edilen sonuçlar ile ele alınan hipotezin aynı yazara ait dokümanları ayırt etmede başarılı olduğu fakat farklı yazarlara ait doküman çiftlerini ayırt etmede yeterli başarıyı gösteremediği görülmektedir. Bu çalışmadan elde edilen sonuçlarda da görüleceği üzere sonuçların daha memnun edici seviyelere çıkılabilmesi için aşamalı olarak veri kümesinin, üretilen modelin ve öznitelik setinin güncellenerek farklı sonuçların üretilip yorumlanabilmesi gerekmektedir.

4. YAZAR MODELLEME İLE YAZAR DOĞRULAMA

Bu bölümde, "Yazar doğrulama problemlerinde dikkate alınması gereken benzerlik eşik değerinin ne olması gerekir?" sorusunun cevabı, yazar modelleri oluşturularak bulunmaya çalışılmıştır. Oluşturulan yazar modelleri, yazarları temsil eden bir doküman gibi ele alınıp söz konusu yazara ait dokümanlara olan benzerlik değerlerine göre bir eşik belirlenmeye çalışılmıştır. Ek olarak, bir yazarı temsil etmede kullanılan başarılı öznitelik kümelerinin birbirine göre karşılaştırması da yapılmıştır. Köşe yazılarının veri kümesi olarak kullanıldığı bu çalışmada elde edilen deneysel sonuçlar ve yapılan çıkarımlar sözlü bildiri olarak literatüre eklenmiştir.

4.1. Konu Kapsam ve Literatür

Bir dokümanın incelenmesi ile o dokümanın yazarı ile ilgili bilgi çıkarma çalışmaları yıllardan beri ilgi gören zorlu bir çalışma alanıdır. Yazar Analizi [10] veya Yazar Tanımlama [7] başlıkları ile ele alınan bu çalışmaların 19.yy'dan beri hesaplamalı yöntemlerin kullanımı ile yaygınlığı artmıştır [3]. Günümüz teknolojisi de göz önünde bulundurulduğunda metinsel verilerin üretiminin sürekli olarak katlanarak artıyor olması göz ardı edilemeyecek bir birikim oluşturmaktadır. Bu birikim söz konusu alanlara özgü yapılan ve yapılacak çalışmaların popülerliğini arttırmaktadır. Bir takım kuruluşların da alana özgü veri yayını yapması [44, 45] ve yapılan çalışmaları desteklemesi bu alana olan ilgiyi daha da arttırmaktadır. Yazar analizi çalışmaları temelde 3 ana dala ayrılmaktadır; Yazar Tanımlama [9], Yazar Profil Çıkarımı [46, 47] ve Yazar Doğrulama [18, 36]. Bazı kaynaklarda aynı alan olarak ele alınmış olsa da Yazar Tanımlama çalışmaları, yazar analizi alanında ele alınan problemlerin başında gelmektedir. Yazar tanımlama problemlerinde sorgulanan bir dokümanın belirli bir grup aday yazar arasından birine ataması yapılmaya çalışılır. Bu çalışmalarda belirli bir grup yazar ve bu yazarlara ait olduğu bilinen birçok doküman vardır. Sorgulanan doküman hangi yazarın dokümanlarına daha fazla benzerlik gösterirse kullanılan algoritmaya göre, o yazarın çalışması olarak atanır. Yazar profil çıkarımı çalışmalarında amaç sorgulanan bir dokümanın yazarı ile ilgili yaş, cinsiyet, eğitim durumu, psikolojik durum gibi sosyo-demografik bilgilerin çıkarılmasıdır.

Yazar analizi çalışmalarının en zorlu problemlerinden biri Yazar Doğrulamadır. Yazar doğrulama problemlerinde arka planda tek bir yazar ve o yazara ait az sayıda doğrulamada

kullanılabilecek doküman vardır. Yazar analizi ve adli bilişim çalışmalarının ortak noktası olan bu problem diğer problemlerin en derin katmanı olarak ele aldığından yazar analizi çalışmalarının temel problemi olarak değerlendirilmektedir [20]. Yazar doğrulama probleminin zorluğu ele alınan veri kümesinin çok az (en fazla 10 doküman) olmasından kaynaklanmaktadır. Bu çalışmada yazar doğrulama problemine yazar modelleme yaklaşımı ile bir çözüm sunulmaya çalışılmıştır. Bir yazara ait birkaç dokümanın bileşimi kullanılarak o yazarı temsil edebilecek bir yapı, bir profil oluşturulmuştur. Elde edilen bu yapı ile söz konusu yazara ait dokümanlar arasındaki benzerlik değerleri ölçülmüştür. Ölçülen değerlerin, bir yazara ait dokümanların o yazara ait bir profile ne kadar yakınlık gösterebileceği bilgisini vermesi beklenmektedir. Ölçümler sonucu elde edilebilecek bu bilginin yazar doğrulama çalışmalarında ele alınan dokümanlar arası benzerlik değerinin nasıl yorumlanması gerektiğine yardımcı olacaktır.

Yazar analizi problemlerinin temel bir problemi olarak değerlendirilmesi sebebiyle yazar doğrulama problemi için sunulacak başarılı bir çözüm yaklaşımının bu alandaki farklı problemlerin çözümüne de katkı sağlayacağı aşikardır. Yapısı gereği yazar analizi çalışmalarında başarılı sonuçlar veren yaklaşımların da yazar doğrulama probleminin çözümüne katkı sağlaması beklenmektedir. Dolayısı ile yazar doğrulama probleminin çözümü için sunulması beklenen bir yaklaşımın bu alandaki diğer başarılı yaklaşımları da kapsam dahiline alması gerekmektedir. Bu bakış açısı ile yapılan çalışmada yazar analizi ile ilgili yapılan tüm başarılı çalışmalar kapsam dahilindedir. Yazar tanımlama alanında Türkçe metinler kullanılarak yapılan başarılı çalışmalar bulunmaktadır. Farklı özniteliklerin, farklı veri kümelerinin ve farklı yöntemlerin kullanıldığı çalışmalara [48] ek olarak bu alandaki en ayırt edici özniteliklerin ve sınıflandırma yöntemlerinin bulunmaya çalışıldığı çalışmalar da yapılmıştır [49]. Köşe yazılarının ele alındığı bir doktora tezi olan yazar tanımlama çalışmasında [43] belirlenen en ayırt edici öznitelik kümesi bu çalışmada ilk öznitelik kümesi olarak kullanılmaktadır.

Yazar doğrulama probleminde az sayıda veri ile çözüm sunabilmek gerektiğinden bu probleme çoğunlukla yazar analizi alanındaki başka problemlere benzetilerek çözümler üretilmeye çalışılmıştır. Yapısı gereği tek sınıflı bir sınıflandırma problemi [10, 15] olan yazar doğrulama probleminde ele alınan dokümanları metin bloklarına bölerek sınıf içeriğini arttıran farklı çözüm yaklaşımları da bulunmaktadır [3]. Harici yazarların dokümanlarını kullanarak tek sınıflı

sınıflandırma problemini çok sınıflı veya iki sınıflı sınıflandırma problemine dönüştüren çalışmalarda da başarılı sonuçlar elde edilmiştir [23, 29].

Bu çalışmada yazar doğrulama problemlerinde sorgulanan ve karşılaştırılan dokümanlar arası benzerlik aralığının karar kriteri olarak hangi değerlerde olması gerektiği ve elde edilen benzerlik değerinin nasıl yorumlanması gerektiği üzerine deneyler yapılmıştır. Bir grup yazara ait dokümanlar kullanılarak söz konusu yazarların yazma üslubunu temsil eden yazar modelleri oluşturulmuştur. Oluşturulan bu modellerin oluşturulduğu dokümanlar ile arasındaki benzerlikler hesaplanarak bir dokümanı bir yazara atayabilmek için gerekli olan benzerlik aralığı ne olmalıdır sorusunun cevabı bulunmaya çalışılmıştır.

4.2. Kullanılan Veri Kümesi

Yazar doğrulama çalışmalarında sorgulanan bir dokümanın şüpheli bir yazara ait olup olmadığına karar verilmeye çalışılır. Bu karar verme aşamasında dokümanlar arası benzerlik bakımından göz önünde bulundurulması gereken eşik değerin belirlenmesi çok önemlidir. Bu çalışmada, yazar doğrulama çalışmalarında dikkate alınması gereken eşik değerin ne olması gerektiğini belirlemek için deneyler yapılmıştır. Yapılan deneylerde veri kümesi olarak köşe yazıları kullanılmıştır. Güncel siyaset alanında yazan rastgele seçilmiş 12 köşe yazarının yine rastgele olarak seçilmiş 100 köşe yazısı bu çalışmanın veri kümesi olarak kullanılmıştır. Söz konusu yazılar 2012 – 2014 yılları arasında yayınlanmış yazılardır. Çalışmada kullanılacak öznitelik seti 5 gruba ayrılmış dolayısı ile her yazar için 5 farklı yazar modeli üretilmiştir. İlk grupta kullanılan öznitelikler bir önceki bölümde de bahsedildiği gibi yazar tanımlama çalışmalarında Türkçe için en ayırt edici olduğu bir doktora tezi çalışması ile tespit edilen öznitelikleri kapsamaktadır. Ele alınan 2. grup öznitelik kümesinde 1. grup özniteliklerine ek olarak Türkçe'de bulunan argo ve yansıma kelimeleri gibi semantik bakımından farklı özelliklerdeki kelimelerin frekansları bulunmaktadır. Üçüncü öznitelik kümesinde ise 2. kümedeki özniteliklere ek olarak devrik cümle, edilgen cümle gibi farklı cümle yapılarının frekansları eklenmiştir. Dördüncü öznitelik kümesinde 3. kümeye ek olarak cümle sayısı, kelime sayısı gibi dokümanın yapısal olarak sayısal özellikleri bulunmaktadır. Kullanılan son veri kümesinde ise 4. kümeye ek olarak bağlaç sayısı, fiil sayısı gibi sözcük türlerinin frekansları bulunmaktadır. Öznitelik çıkarımı işlemleri için yukarıda ve bir önceki bölümde bahsedilen doktora çalışması kapsamında geliştirilen bir araç kullanılmıştır. Kullanılan öznitelik kümeleri ve içerdiği öznitelikler Tablo 4.1'de ayrıntılı olarak verilmiştir.

Tablo 4.1. Yazar modellemede kullanılan öznitelik kümeleri

5. Set						
4. Set						
3. Set						
2. Set						
1. Set						
Nokta (.)	Parantez ((,))	Tire (-)	Argo	Devrik	Paragraf	Bağlaç
Çift tırnak (“)	Yan çizgi (/)	Virgül (,)	Osmanlıca	Edilgen	Ayrık Kelime	Edat
Soru işareti (?)	Noktalı Virgül (;)		Zaman etiketi	tek kelimelik	Cümle	Fiil
Ters Yan Çizgi (\)	İki nokta üst üste (:)		Özel İsim		Kelime	Mastar
Ünlem işareti (!)	Ampersand (&)		Yansıma		Sayı	İsim
Alt tire (_)	Tek tırnak(‘)		Kısaltma			Sıfat

Tablo 4.1’de görüldüğü üzere kullanılan öznitelik kümeleri artırımlı bir şekilde sıralanmaktadır. Her set ile ayrı ayrı yazar modelleri oluşturularak kullanılan özniteliklerin etkisi ölçülmek istenmektedir.

4.3. Materyal ve Metot

Bir dokümanın bir yazar tarafından yazıldığının iddia edilebilmesi için o yazara ait dokümanlar ile sorgulanan dokümanın bazı ortak özellikler barındırması gerekmektedir. Yazar doğrulama problemlerinde, ele alınan bir dokümanın şüpheli bir yazara atanabilmesi için belirli karar kriterlerini sağlaması gerekmektedir. Bu kriterler çoğunlukla yazarın üslupsal yazı izinin bulunması ile ilgilidir. Yazar doğrulama probleminde karar aşamasında dikkate alınması gereken üslupsal benzerlik aralığının belirlenmeye çalışıldığı bu çalışmada yazar modelleri kullanılmaktadır. Öncelikle, 12 köşe yazarının rastgele seçilmiş 100 yazısının belirlenen öznitelik setleri kullanılarak vektörel dönüşümleri gerçekleştirilmiştir. A kümesi kullanılan veri kümesinde bulunan yazarların olduğu küme olmak üzere; $A = \{A_1, A_2, A_3, \dots, A_{12}\}$, 12

elemandan oluşmaktadır. $A_i \in A$ olmak üzere her $A_i = \{d_1, d_2, d_3, \dots d_{100}\}$ şeklinde 100 dokümana sahiptir. 5 farklı öznitelik seti kullanıldığından her yazara ait her dokümanın 5 farklı vektörel temsili oluşturulmuştur. Seçili özniteliklerin kullanım sıklığı bilgisi (frekansı) kullanılarak elde edilen doküman vektörleri kullanılan her set için belirlenen boyutta bir vektöre dönüştürülmüştür. D kümesi, kullandığımız veri kümesindeki dokümanların kümesi olmak üzere, her $d_i \in D$ için 1. öznitelik seti kullanılarak oluşturulan d_i vektörü $<f_1, f_2, f_3, \dots f_{14}>$ ile, 2. öznitelik seti kullanılarak oluşturulan d_i vektörü $<f_1, f_2, f_3, \dots f_{20}>$ ile, 3. öznitelik seti kullanılarak oluşturulan d_i vektörü $<f_1, f_2, f_3, \dots f_{23}>$ ile, 4. öznitelik seti kullanılarak oluşturulan d_i vektörü $<f_1, f_2, f_3, \dots f_{28}>$ ile ve 5. öznitelik seti kullanılarak oluşturulan d_i vektörü $<f_1, f_2, f_3, \dots f_{34}>$ ile temsil edilmektedir. Kullanılan 1, 2, 3, 4 ve 5 numaralı setler ile oluşturulan doküman vektörlerinin boyutu sırasıyla; 14, 20, 23, 28 ve 34 olmaktadır. Üretilen her doküman vektörünün eşit standartta değerlendirilebilmesi için, yani dokümanlar arası boyut farkının dokümanların karşılaştırılmasındaki etkisini ortadan kaldırmak için tüm doküman vektörleri Denklem (1)'de verilen formül kullanılarak normalize edilmiştir.

Dokümanların vektörel dönüşümleri gerçekleştikten sonra, belirtilen öznitelik setleri kullanılarak her yazar için 5 farklı model oluşturulmaktadır. Bir yazar için üretilecek her model, o yazara ait olan tüm dokümanların seçili öznitelik setinden üretilmiş vektörleri kullanılarak oluşturulmaktadır. Bu aşamada; M_1, 1 numaralı yazarın modelini temsil ederken M_{1_1}, 1 numaralı yazarın 1 numaralı öznitelik seti kullanılarak oluşturulmuş modelini, M_{1_2}, 1 numaralı yazarın 2 numaralı öznitelik seti kullanılarak oluşturulmuş modelini, M_{1_3}, 1 numaralı yazarın 3 numaralı öznitelik seti kullanılarak oluşturulmuş modelini, M_{1_4}, 1 numaralı yazarın 4 numaralı öznitelik seti kullanılarak oluşturulmuş modelini ve M_{1_5}, 1 numaralı yazarın 5 numaralı öznitelik seti kullanılarak oluşturulmuş modelini temsil etmektedir. Örneğin 1 numaralı yazar için üretilen 1 numaralı modelin denklemi Denklem(8)'de verilmiştir. N, söz konusu yazara ait doküman sayısı olmak üzere; d_k, k numaralı doküman vektörünü temsil etmektedir. Her f_x değeri belirtilen modelin üretildiği öznitelik setindeki özniteliği temsil etmektedir.

$$M_{1\_1}(f_x) = \frac{1}{N}\sum_{k=1}^{N} d_k(f_x) \tag{8}$$

$A_i \in A$ olmak üzere; her $A_i = \{M_{i\_1}, M_{i\_2}, M_{i\_3}, M_{i\_4}, M_{i\_5}\}$ olacak şekilde 5 farklı modele sahiptir. Yani, her $A_i \in A$ için hem bir Model kümesi (M_i), hem de her modelin oluşturulduğu

5 farklı doküman vektör kümesi vardır. Her yazara ait 100 doküman ele alındığından, 5 farklı model ve bu beş modelin oluşturulduğu 100'er vektörlü 5 küme bulunmaktadır. Örneğin 1 numaralı yazar için bulunan veriler aşağıda sıralanmıştır.

- A_1 yazarı, M_{1_1}, M_{1_2}, M_{1_3}, M_{1_4} ve M_{1_5} olmak üzere 5 modele sahiptir.
- A_1 yazarı, sahibi olduğu 100 dokümanın 5 farklı yapıda vektörel haline sahiptir.

Kullanılan veri kümesinde bulunan 12 yazar için aynı işlemler yapılmış ve her birinin yukarıda belirtilen verileri elde edilmiştir. Elde edilen bu veriler kullanılarak yazar doğrulama probleminin çözümü için bir yaklaşım geliştirmek hedeflenmiştir. Bu aşamada çalışmanın hedefi olan yaklaşım aşağıdaki gibi ele alınmaktadır.

Bir yazarı temsil eden bir model ile o yazara ait dokümanlar arasındaki ortalama benzerlik değeri, sorgulanan bir dokümanın yazarını belirlemede eşik olarak kabul edilebilir bir değerdir. Dolayısı ile yazar doğrulama problemlerinde bu eşik değer göz önüne alınarak bir karar mekanizması oluşturulmalıdır.

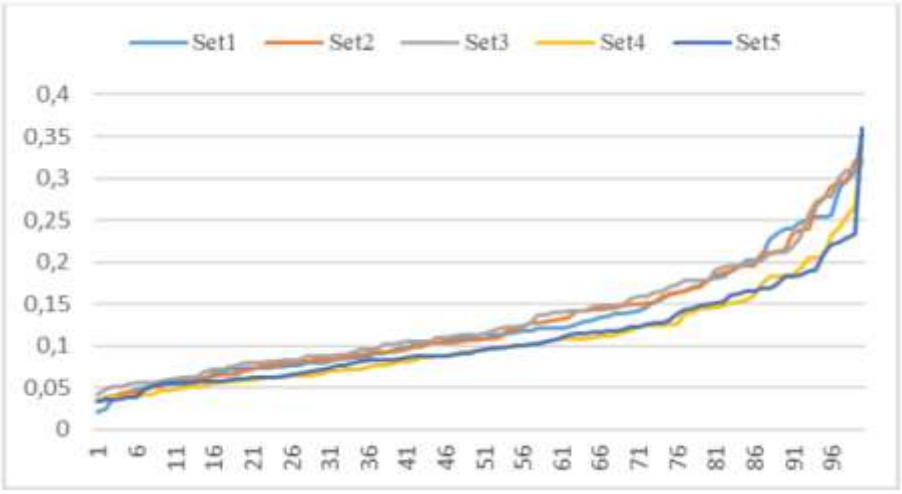
Söz konusu eşik değeri belirlemek için, her yazara ait üretilen her modelin üretildiği dokümanlara olan uzaklığı ölçülmüştür. Her model üretildiği dokümanlar ile aynı vektörel uzayda olduğundan aradaki uzaklığın ölçümü için Denklem (9)'da verilen kosinüs uzaklığı kullanılmıştır.

$$\text{Uzaklık}\,(a^{\leftarrow}, b^{\leftarrow}) = 1 - \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2} \cdot \sqrt{\sum_1^n b_i^2}} \qquad (9)$$

Verilen formül kullanılarak, üretilen her modelin üretildiği dokümanlar ile arasındaki uzaklık değerleri ölçülmüştür. Bu ölçümlerde alınan değerler 0-1 aralığında olup, karşılaştırılan veriler arası uzaklık değeri arttıkça yakınlık (benzerlik) değeri düşüyor demektir. En düşük uzaklık değeri en yüksek benzerliğe sahip, en yüksek uzaklık değeri de en düşük benzerliğe sahiptir. Verilen bilgiler doğrultusunda her yazar için üretilen 5 modelin, üretildiği 100 doküman vektörüne olan uzaklıkları Denklem (9) kullanılarak ölçülmüş ve elde edilen sonuçlar doğrultusunda bir eşik değer aralığı tespit edilmiştir.

4.4. Sonuçlar

12 yazar, her yazara ait 100 dokümanın 5 farklı vektörel temsili ve her yazara ait 5 farklı yazar temsil modelinin üretildiği bu çalışmada vektörler arası mesafenin anlamlandırılması üzerine deneyler yapılmıştır. Yazar doğrulama problemlerinde değerlendirilmek üzere dokümanlar arası anlamlı eşik değerin belirlenmeye çalışıldığı bu çalışmada, oluşturulan her yazar modeli o yazara ait ele alınan 100 doküman ile karşılaştırılmıştır. Böylece bir yazar modelinin, ait olduğu yazarın dokümanlarına ne kadar benzediği test edilmek istenmiştir. Öyle ki; bir yazarı temsil eden bir modelin o yazarın dokümanlarına olan benzerliğinin düşük çıkması, yazar doğrulama problemlerinde benzerlik aralığının bir karar kriteri olarak alınmasının pek de uygun olmayacağını gösterecektir. Diğer taraftan bir yazara ait modelin bile o yazarın dokümanlarına olan benzerliği, başka yazarların dokümanlarına olan benzerlikten düşük olması üretilen modelin temsil özelliklerini barındırmadığının bir göstergesi olacaktır. Bu beklentiler karşısında öncelikle veri kümesindeki her yazar için, üretilen 5 modelin üretildiği dokümanlara olan uzaklıklarına bakılmıştır. Şekil 4.1'de veri kümesindeki yazarlardan biri için elde edilen 5 modelin o yazara ait 100 dokümana olan uzaklıklarının gösterimi verilmektedir.

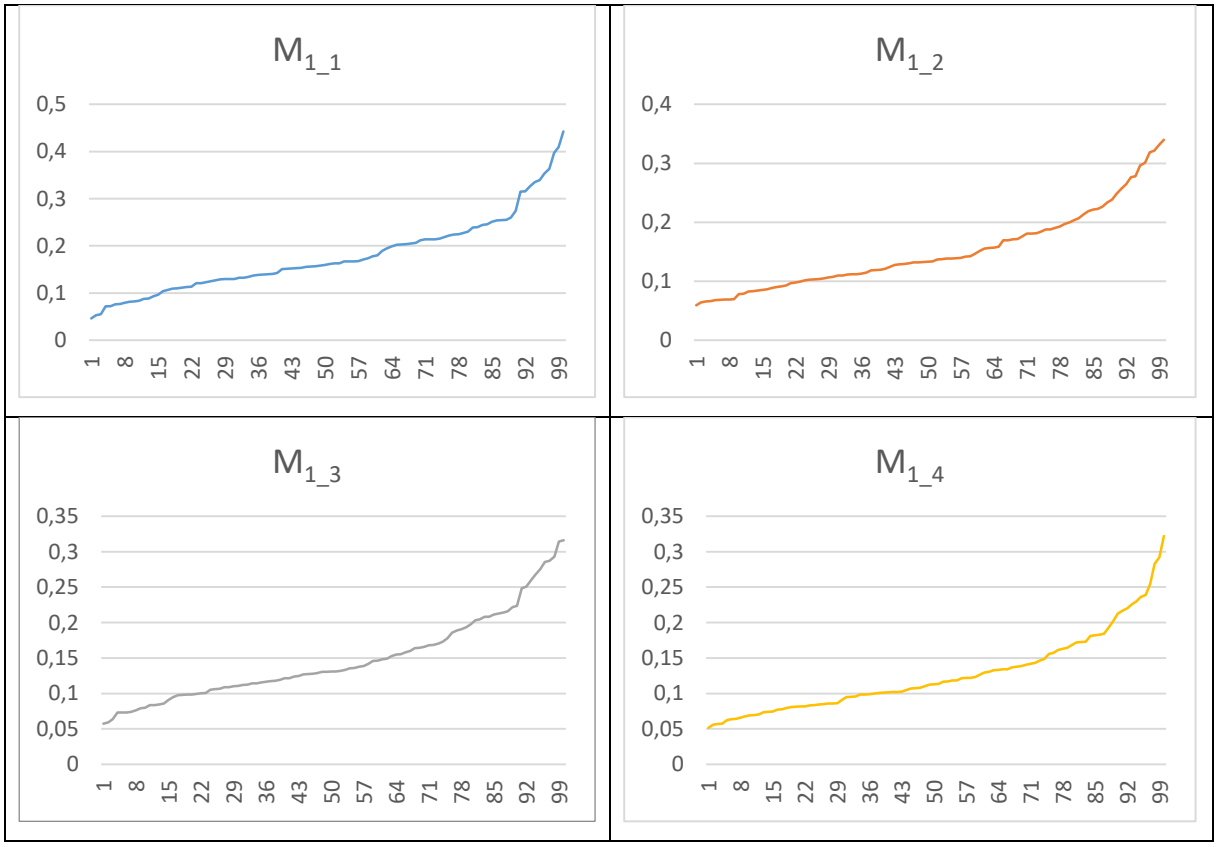


Şekil 4.1. Bir yazara ait üretilen modellerin yazarın dokümanlarına olan uzaklıkları

Şekil 4.1'de yatay eksen dokümanları, düşey eksen de modellerin uzaklık değerlerini göstermektedir. Görüldüğü üzere, üretilen modeller seçili özniteliklere göre ait oldukları yazarı temsil etmede başarılı bir duruş sergilemektedir. Üretilen modellerin söz konusu yazarın dokümanlarına olan uzaklığı 0,27 ile 0,03 aralığında yani, benzerliği %97 ile %63 aralığında yoğunlaşmıştır. Üretilen modeller arası başarım göz önünde bulundurulduğunda, modeller arası bariz ayırt edici farklar bulunmamakla birlikte ortalamada en başarılı sonuç 4. öznitelik setinin

kullanıldığı model ile elde edilmektedir. Bu sonuç, değerlendirmeye alınan öznitelik setlerinden yazarları en iyi temsil eden öznitelik setinin 4 numaralı set olduğunu göstermektedir. Model oluşturma aşamasında kullanılan öznitelik kümelerinden 5 numaralı öznitelik kümesi de ilk üç kümeye oranla daha iyi sonuçlar üretmiştir. Fakat yapılan ayrıntılı incelemeler sonucu kullanılan öznitelik çıkarma aracının bu set özelinde kullanılan özniteliklerin değerinde bir kısım hatalar barındırdığı tespit edilmiştir. Bu sebeple devam eden çalışmalarda, çalışmanın güvenilirliğini tehlikeye atmamak adına bu set deneylerden çıkarılmıştır.

Yapılan karşılaştırmalar sonucu elde edilen değerler, karar aşamasında kullanılmak için ayrıntılı olarak incelenmiştir. Bu incelemeler her yazar için ayrı ayrı ele alınıp anlamlı çıktılar elde edilmeye çalışılmıştır. Bu amaç doğrultusunda A_1 yazarı için üretilen her modelin A_1 yazarına ait dokümanlara olan uzaklıkları Şekil 4.2'de ayrı ayrı görülmektedir.

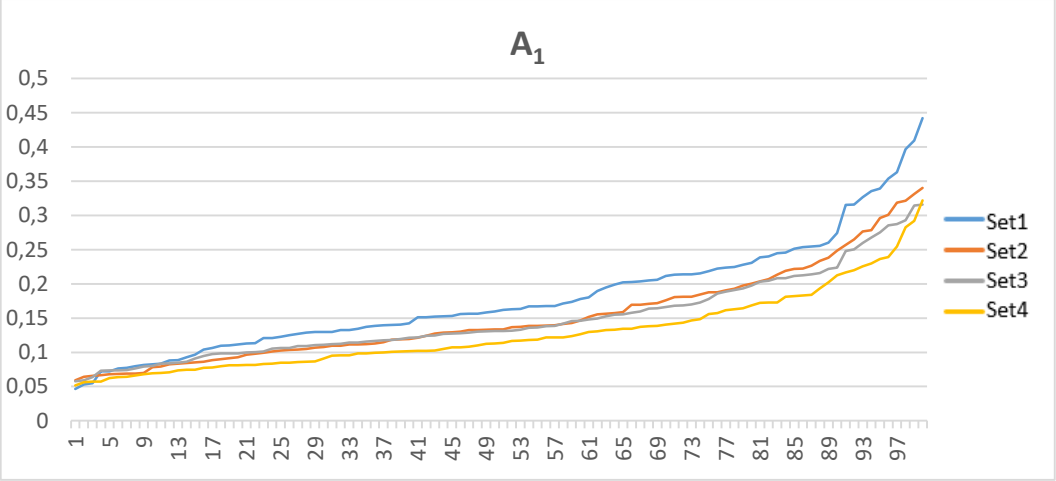


Şekil 4.2. A(1) yazarını temsilen üretilen modellerin yazara ait dokümanlara olan uzaklık grafikleri

A_1 yazarına ait modeller Şekil 4.2'deki gibi ayrı ayrı ele alındığında üretilen her modelin nispeten bir temsil olabileceği görülmektedir. Bu aşamada beklenen, üretilen modellerin yazara ait dokümanların büyük bir kısmına yüksek değerlerde benzerlik göstermesidir. Bazı olağan

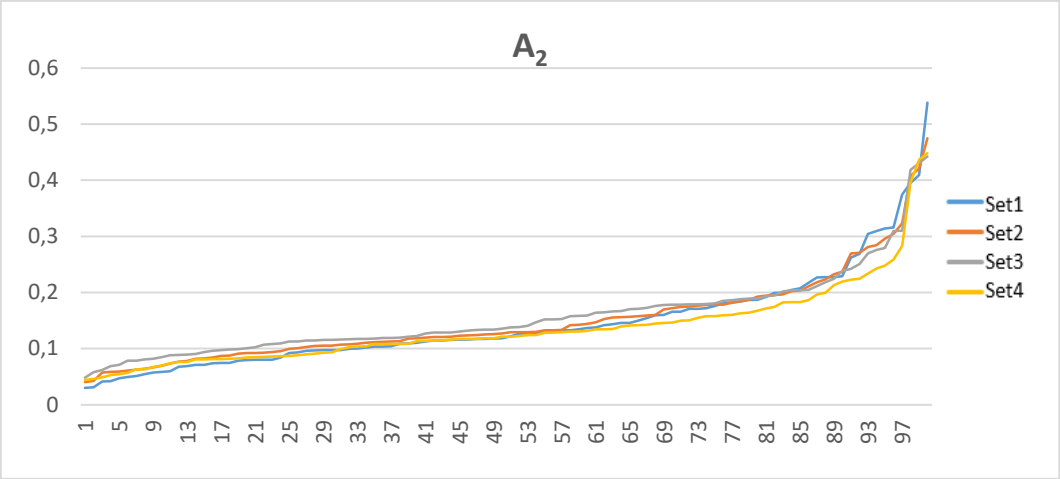
dışı durumlarda yazarın üslubunda zamana veya konuya bağlı değişimler olabilmektedir. Bu değişimler bazı istisnalar olarak üretilen modele genel doküman benzerliğinden nispeten daha düşük bir benzerlik gösterecektir. Fakat bu değişimler ile üretilen modele olan genel benzerlik değerinin yüksek değerlerde kayması üretilen modelin güvenilirliğini etkileyecektir.

Üretilen modeller yazarlara ait benzerlik değeri bakımından beklenen özellikleri barındırıyor olmasının yanında, kullanılan öznitelik setleri bakımından da bir karşılaştırma yapabilme olanağı sunmaktadır. Kullanılan veri kümesi ve değerlendirilen öznitelik setleri çerçevesinde standart bir kural veya karardan bahsedebilmek için veri kümesindeki her yazara ait model karşılaştırması tek tek ele alınmış ve yapılan değerlendirmeler doğrultusunda bazı çıkarımlara ulaşılmıştır. A_1 yazarını temsilen üretilen modellerin karşılaştırmasının gösterildiği grafik Şekil 4.3'te verilmektedir.



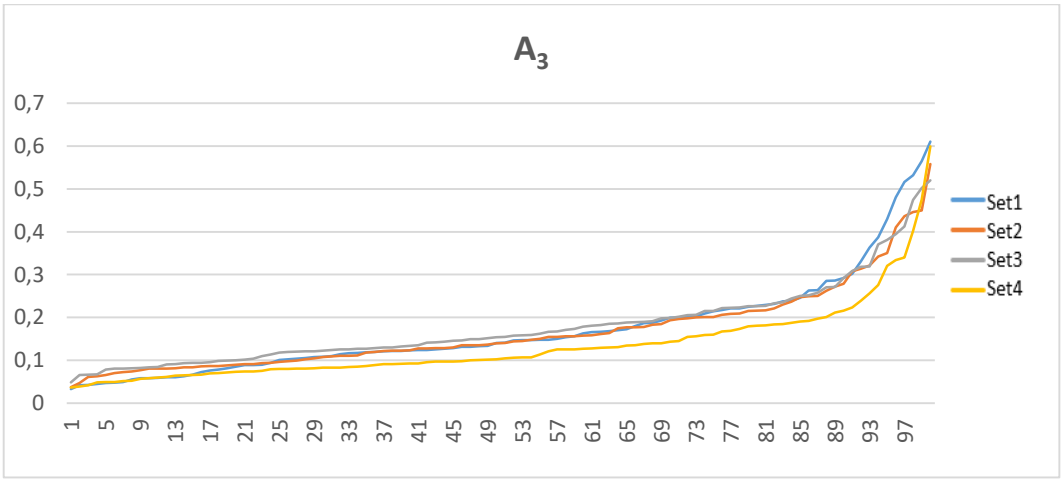
Şekil 4.3. A(1) yazarını temsilen üretilen modellerin karşılaştırması

Şekil 4.3'te görüldüğü üzere, A_1 yazarına ait üretilen modellerden en başarılısı 4 numaralı öznitelik seti kullanılarak üretilen model olmuştur. Ortalamada en düşük benzerlik değerine sahip model 1 numaralı öznitelik seti ile üretilen model olurken, kullanılan öznitelik kümesi arttırılırken elde edilen başarının da arttığı gözlemlenmektedir. A_2 yazarına ait üretilen modellerin karşılaştırması Şekil 4.4'te gösterilmektedir.



Şekil 4.4. A(2) yazarını temsilen üretilen modellerin karşılaştırması

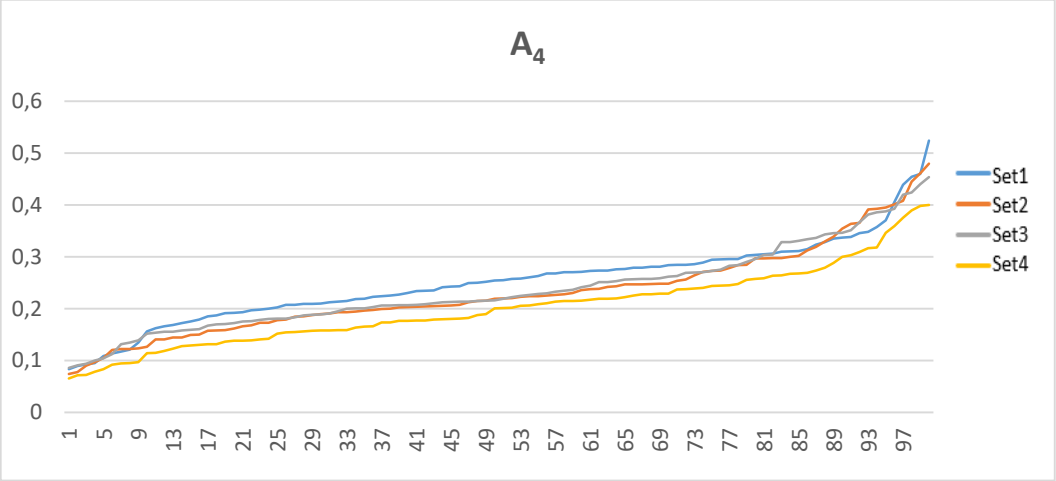
Şekil 4.4'te görüldüğü üzere, A_2 yazarını temsil eden modeller arasında en başarılı model, A_1 yazarında olduğu gibi 4 numaralı öznitelik setinin kullanıldığı model olmuştur. Ortalamada en düşük benzerlik değerine sahip model 3 numaralı öznitelik seti kullanılarak üretilen model olurken, A_2 yazarının temsillerinde ilk üç set ile üretilen modellerde elde edilen ortalama başarılar birbirine çok yakındır. Dolayısı ile A_1 yazarına ait modellerdeki gibi öznitelik artırımına bağlı bir başarı artırımından bahsedilememektedir. A_3 yazarına ait modellerin karşılaştırması Şekil 4.5'te gösterilmektedir.



Şekil 4.5. A(3) yazarını temsilen üretilen modellerin karşılaştırması

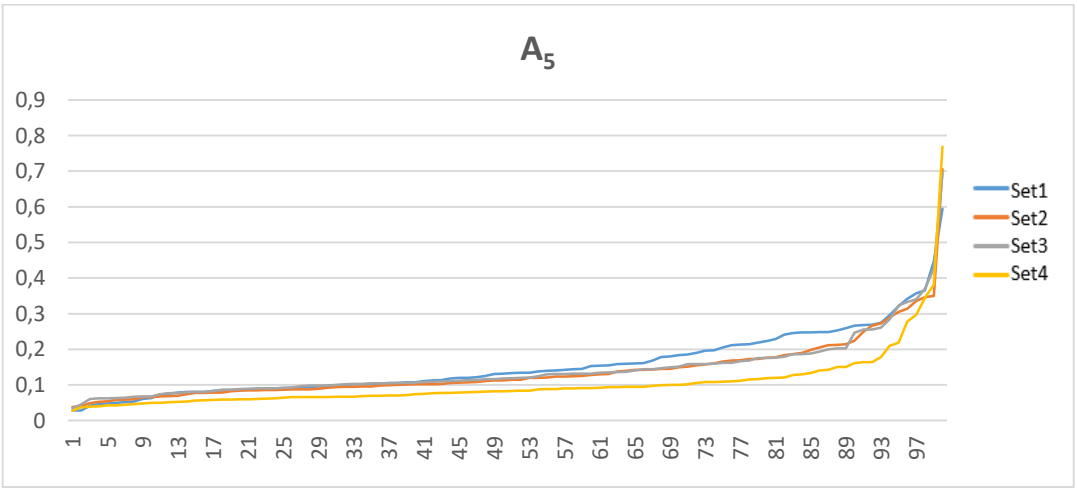
A_3 yazarını temsilen üretilen modellerde de A_2 yazarını temsilen üretilen modellere benzer bir örüntü görülmektedir. En başarılı temsil 4 numaralı öznitelik kümesi kullanılarak üretilen model olurken, diğer üç model ortalamada yakın sonuçlar üretmiştir. A_2 yazarından farklı

olarak A_3 yazarına ait dokümanlar arasındaki benzerliğin daha az olduğu görünmektedir. A_4 yazarına ait modellerin karşılaştırması Şekil 4.6'da gösterilmektedir.



Şekil 4.6. A(4) yazarını temsilen üretilen modellerin karşılaştırması

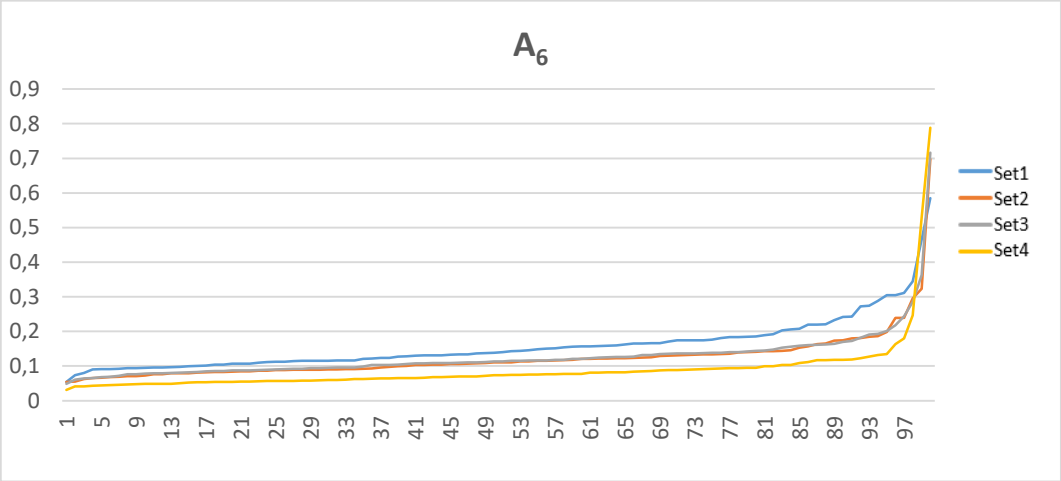
4 numaralı öznitelik seti kullanılarak üretilen model A_4 yazarını temsilen üretilen modeller arasında da en başarılı sonuçları üretmiştir. Şekil 4.6'da görüldüğü üzere, ortalamada en düşük başarı 1 numaralı öznitelik setinin kullanıldığı temsil modelinden elde edilirken, 2 ve 3 numaralı öznitelik setlerinin kullanıldığı modeller bu yazar özelinde birbirine çok yakın sonuçlar üretmiştir. A_5 yazarına ait modellerin karşılaştırması Şekil 4.7'de gösterilmektedir.



Şekil 4.7. A(5) yazarını temsilen üretilen modellerin karşılaştırması

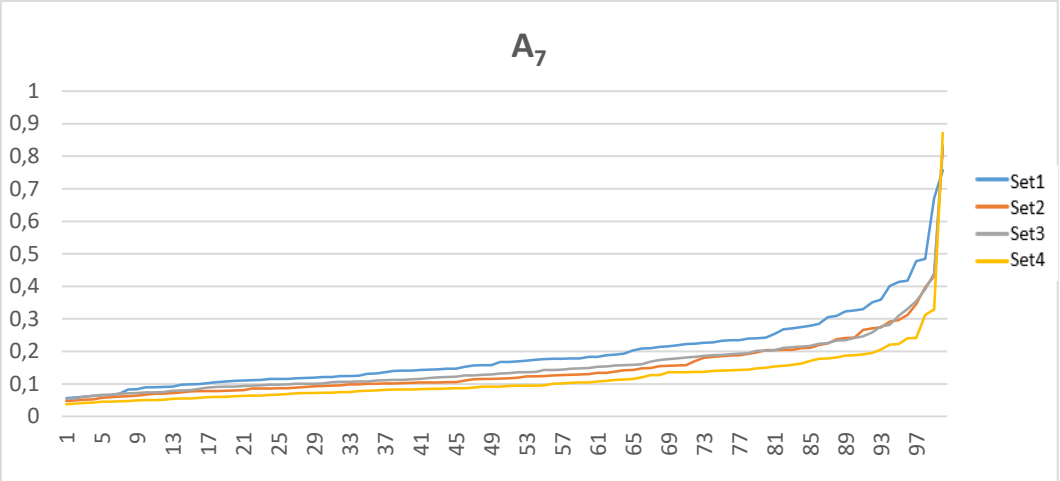
A_5 yazarına ait dokümanların üretilen modellere olan benzerlikleri göz önüne alındığında, 4 numaralı öznitelik kümesi ile üretilen modelin bu yazar özelinde de en yüksek benzerlik değerini ürettiği Şekil 4.7'de görülmektedir. Ortalamada en düşük benzerlik değeri Set1 ile üretilen model ile elde edilirken diğer iki modelin birbirine çok yakın benzerlik değerinde

olduğu görülmektedir. Yukarıda bahsedilmiş olan, bazı istisna dokümanların zamana veya konuya bağlı olarak yazarın belirgin üslubundan uzaklaşabilmeleri durumu A_5 yazarı özelinde görülebilmektedir. Ele alınan 100 doküman arasından birinin 4 numaralı öznitelik seti ile üretilen modele uzaklığı 0,7 üzerinde bulunmaktadır. Bahsi edilen durum üretilen tüm temsil modelleri tarafından ayırt edilebilmiştir. A_6 yazarına ait modellerin karşılaştırması Şekil 4.8'de gösterilmektedir.



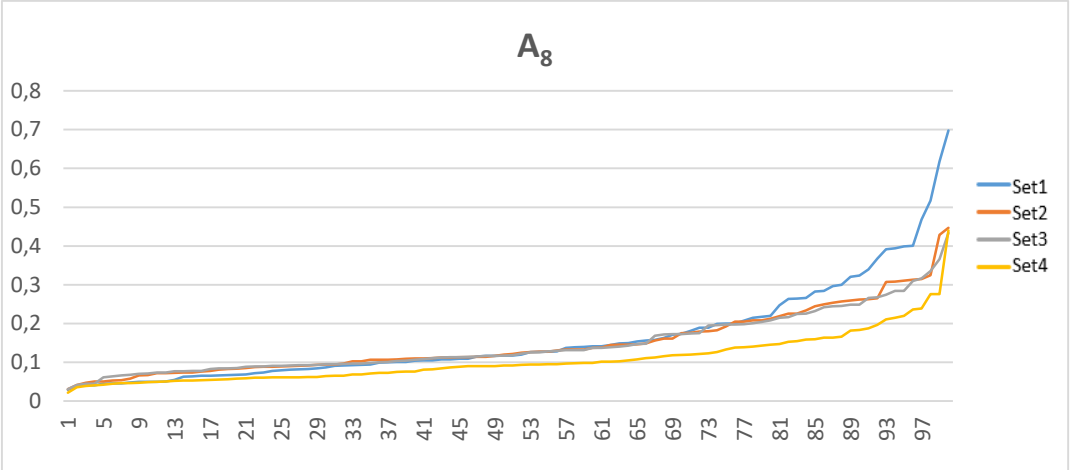
Şekil 4.8. A(6) yazarını temsilen üretilen modellerin karşılaştırması

Şekil 4.8 incelendiğinde A_5 yazarında olduğu gibi A_6 yazarına ait dokümanlarda da birkaç istisna dokümanı olduğu görülebilmektedir. A_6 yazarı özelinde bulunan istisna dokümanlar A_5 yazarındaki istisnalara göre modellere daha aykırı durmaktadır. Üretilen dört yazar temsil modeli de genel olarak yazara ait dokümanlara 0,7 – 0,95 aralığında benzerlik göstererek başarılı bir temsil sergilerken önceki beş yazarda olduğu gibi ortalamada en yüksek başarı 4 numaralı model kullanılarak elde edilmiştir. En düşük başarı bu yazar özelinde bir numaralı temsil modelinden elde edilirken 2 ve 3 numaralı öznitelik setleri ile üretilen modeller birbirine çok yakın sonuçlar üretmiştir. A_7 yazarına ait modellerin karşılaştırması Şekil 4.9'da gösterilmektedir.



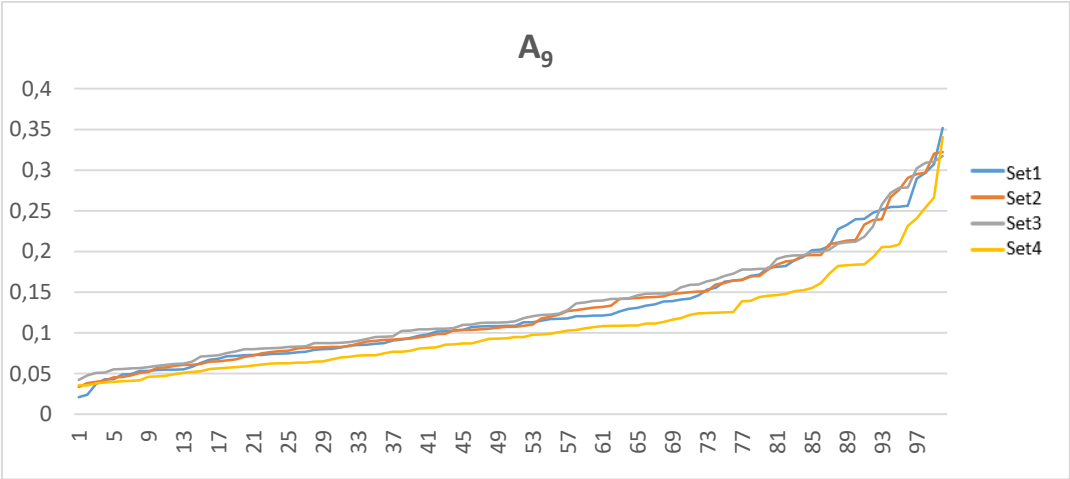
Şekil 4.9. A(7) yazarını temsilen üretilen modellerin karşılaştırması

A_7 yazarını temsilen üretilen modeller ile A_6 yazarını temsilen üretilen modeller birbirine benzer örüntü sergilemektedir. Şekil 4.9'da görüldüğü üzere, bu yazar özelinde ele alınan dokümanlardaki istisnalar neredeyse başka bir yazar tarafından yazılmış kadar modellerden aykırı durmaktadır. Bu sayı 3/100 gibi düşük bir oranda olsa da (100 dokümandan 3 doküman aykırı) özellikle adli bilişim konularında göz ardı edilemeyecek bir sapmadır ve bir yazarın üslubunun sadece kullanılan bu öznitelikler kapsamında net bir değerlendirmeden geçemeyeceğinin bir gösterimidir. Yazar analizi çalışmalarında da, her ne kadar üslupsal yazı izi tespitinden söz ediyor olsak da, parmak izi veya retina örüntüsü gibi kesin sonuçların çıkarımı bu çalışmalarda söz konusu değildir. Yapılan çıkarımlar elde edilen olasılıklar çerçevesinde geliştirilmektedir ki A_7 yazarını temsilen üretilen modellerin davranışı buna en güzel örneklerden biridir. Bu yazarda da 4 numaralı öznitelik seti kullanılarak üretilen model en başarılı sonuçları üretirken, 1 numaralı set kullanılarak en düşük başarılı sonuçlar elde edilmiştir. A_8 yazarına ait modellerin karşılaştırması Şekil 4.10'da gösterilmektedir.



Şekil 4.10. A(8) yazarını temsilen üretilen modellerin karşılaştırması

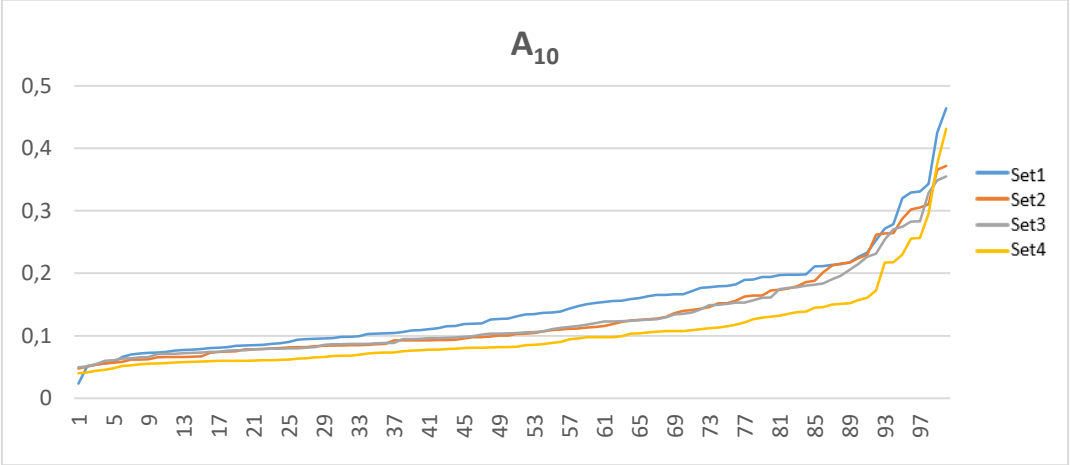
Şekil 4.10'da görüleceği üzere 4 numaralı öznitelik seti ile üretilen yazar temsil modeli en başarılı temsil olmaktadır. 1 numaralı öznitelik seti ile üretilen model en düşük başarıya sahip temsil olurken diğer üç temsile göre de A_8 yazarı özelinde, istisna olarak değerlendirdiğimiz dokümanlarda temsil ediciliği daha da azalmıştır. A_9 yazarına ait modellerin karşılaştırması Şekil 4.11'de gösterilmektedir.



Şekil 4.11. A(9) yazarını temsilen üretilen modellerin karşılaştırması

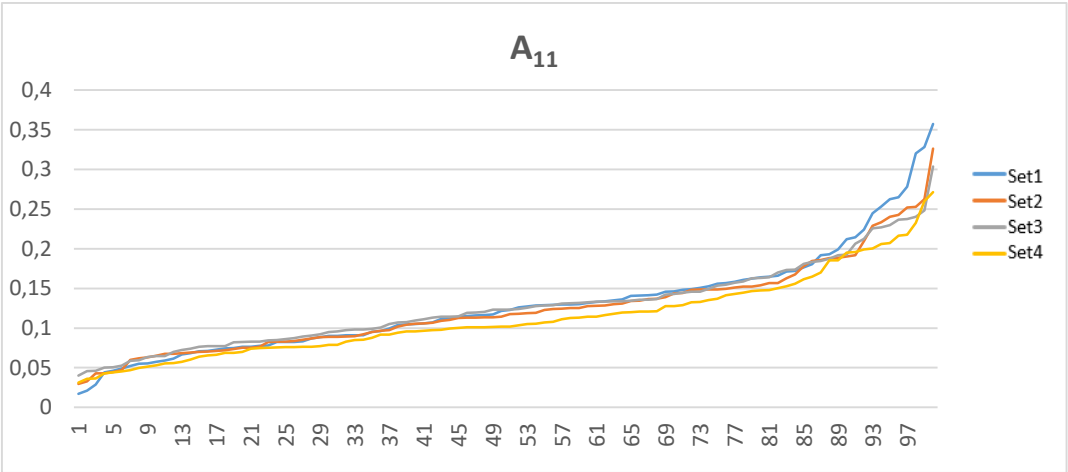
Üretilen tüm modellerin yüksek başarı sergilediği en uygun temsiller A_9 yazarına ait dokümanlar kullanılarak elde edilen temsiller olmuştur. Şekil 4.11'de görüldüğü üzere, üretilen modeller arasında ortalamada en yüksek başarı 4 numaralı öznitelik seti kullanılarak üretilen model ile elde edilmiştir. Üretilen diğer üç model ile birbirine yakın sonuçlar elde edilmiştir. A_9 yazarı özelinde ele alınan dokümanlar arası istisna olarak değerlendirebileceğimiz doküman olmaması, yani tüm dokümanların makul oranlarda (0,65 – 0,97) üretilen modellere benzerlik

göstermesi, bu yazar profilinin ideale yakın bir temsil için kullanılabileceğini göstermektedir. A_{10} yazarına ait modellerin karşılaştırması Şekil 4.12'de gösterilmektedir.



Şekil 4.12. A(10) yazarını temsilen üretilen modellerin karşılaştırması

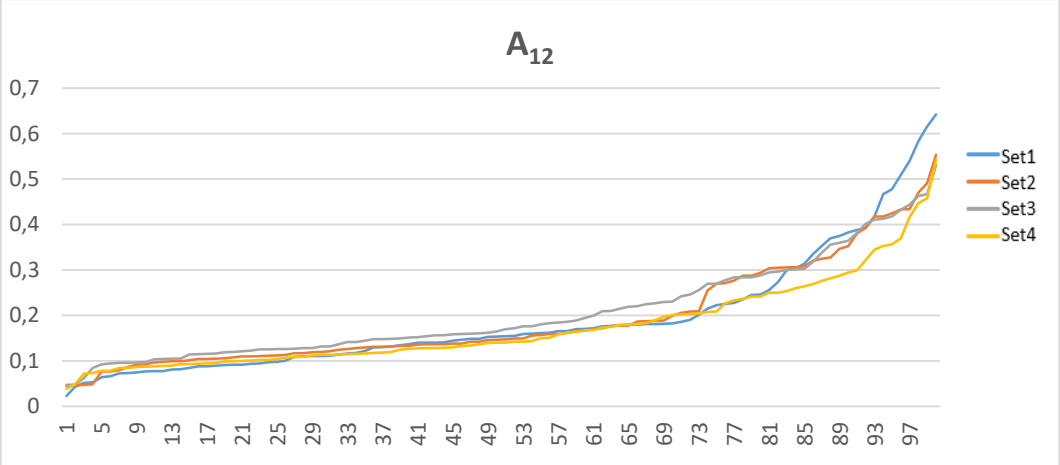
Kullanılan tüm öznitelik kümeleri ile üretilen modeller A_{10} yazarına ait dokümanlara 0,5 üzerinde benzerlik gösterirken ortalamada en yüksek benzerlik değerine sahip temsil 4 numaralı öznitelik seti ile üretilen model kullanılarak elde edilmektedir. Şekil 4.12'de görülebileceği üzere, ortalamada en düşük başarı 1 numaralı öznitelik kümesi kullanılarak üretilen model ile elde edilirken diğer iki model birbirine yakın benzerlik örüntüleri sergilemektedir. A_{11} yazarına ait modellerin karşılaştırması Şekil 4.13'te gösterilmektedir.



Şekil 4.13. A(11) yazarını temsilen üretilen modellerin karşılaştırması

A_{11} yazarını temsilen üretilen modeller birbirine yakın ve makul değerlerde sonuçlar üretmiş olsa da 4 numaralı öznitelik kümesi kullanılarak üretilen model Şekil 4.13'te görüldüğü üzere nispeten daha başarılı sonuçlar vermiştir. Bu yazar özelinde de 1 numaralı öznitelik seti ile

üretilen modelin nispeten genele aykırı dokümanlarda diğer modeller kadar kapsayıcı olamadığı görülebilmektedir. A_{12} yazarına ait modellerin karşılaştırması Şekil 4.14'te gösterilmektedir.



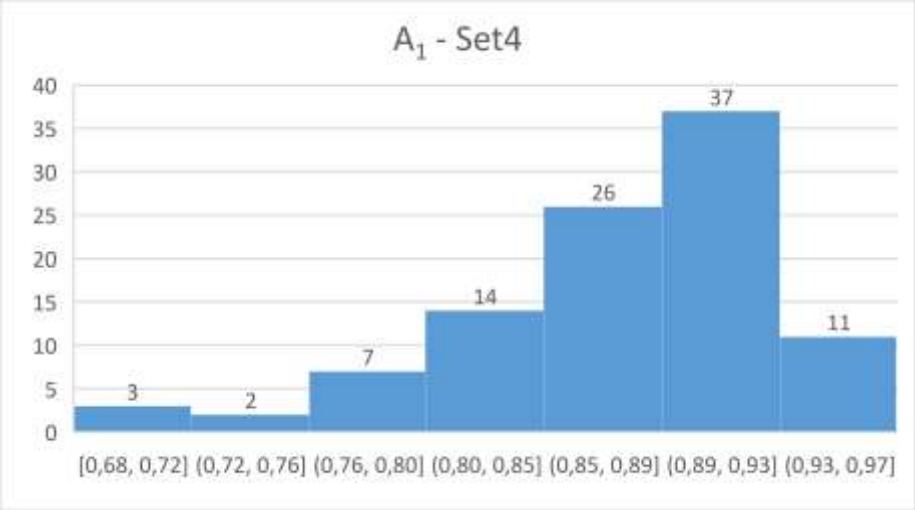
Şekil 4.14. A(12) yazarını temsilen üretilen modellerin karşılaştırması

Şekil 4.14'te görüldüğü üzere, A_{12} yazarını temsilen üretilen modellerin hepsi ortalamada birbirine yakın sonuçlar üretmiş olmasına rağmen 4 numaralı öznitelik seti kullanılarak üretilen model ortalamada yine en başarılı temsil olmaktadır. Yazara ait dokümanlar modellerden uzaklaştıkça en genelleyici modelin 4 numaralı set ile üretilen model, en az kapsayıcı modelin de 1 numaralı öznitelik seti kullanılarak üretilen model olduğu bu yazar özelinde de tekrar görülebilmektedir.

Değerlendirdiğimiz veri kümesinde bulunan 12 yazar için 4 farklı model oluşturup oluşturulan bu modellerin her bir yazar için temsil derecelerini yukarıdaki sonuçlar doğrultusunda gözlemleyebiliriz. Yazarları temsilen üretilen modellerde 4 numaralı öznitelik seti kullanılarak üretilen model istisnasız tüm yazarlarda ortalamada en iyi temsil olmuştur. Bu sonuç, 4 numaralı öznitelik seti özelinde bulunan özniteliklerin (paragraf frekansı, ayrık kelime frekansı, cümle frekansı, kelime ve sayı frekansı) yazar temsilinde önemli bir etkisi olduğunu göstermektedir.

Bir yazarı temsil etmede en iyi sonuçları veren öznitelik setinin yazar doğrulama çalışmalarında da en iyi sonuçları üreteceği hipotezinden yola çıkarak bu çalışmada seçili öznitelik kümelerinden ortalamada en iyi sonuçları üreten 4 numaralı öznitelik kümesinin yazar doğrulama çalışmalarında da en iyi sonuçları üreteceği çıkarımı yapılabilmektedir.

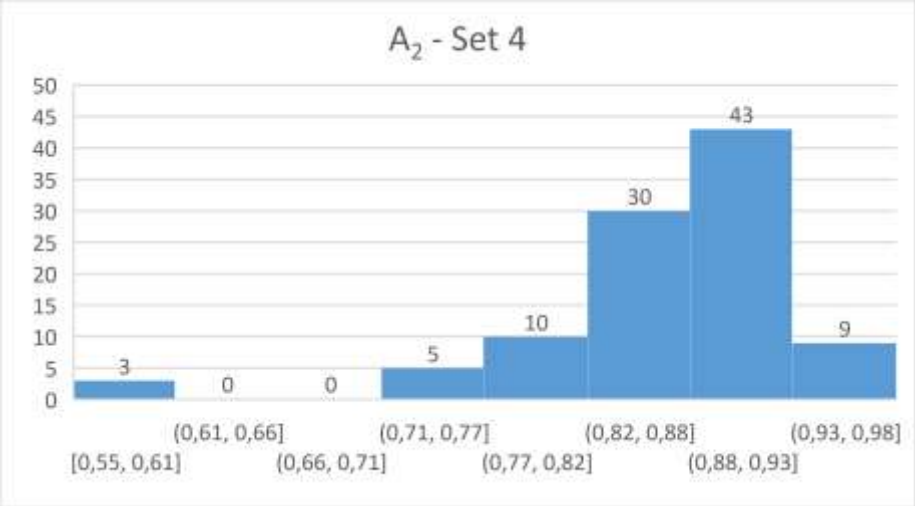
Üretilen modeller arasında en başarılı modelin bulunmuş olması bu çalışmanın ilk çıktılarından biridir. Bu çalışma kapsamında asıl amaç yazar doğrulama problemlerinin çözümünde dikkate alınması gereken benzerlik aralığının belirlenmesi olduğundan, çalışmanın devamında 4 numaralı öznitelik seti kullanılarak elde edilen modellerin yazar dokümanlarına olan benzerlik aralığı dikkate alınarak çıkarımlar yapılacaktır. Aşağıda bazı yazarları temsilen, 4 numaralı öznitelik seti kullanılarak üretilen modelin söz konusu yazarın dokümanlarına olan benzerlik değerleri grafikler halinde verilmiştir. Verilen grafiklerde yatay eksende benzeme oranları verilirken, düşey eksende belirtilen benzerlik aralığında söz konusu yazara ait kaç dokümanın bulunduğu gösterilmektedir. A_1 yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.15'te gösterilmektedir.



Şekil 4.15. A(1) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

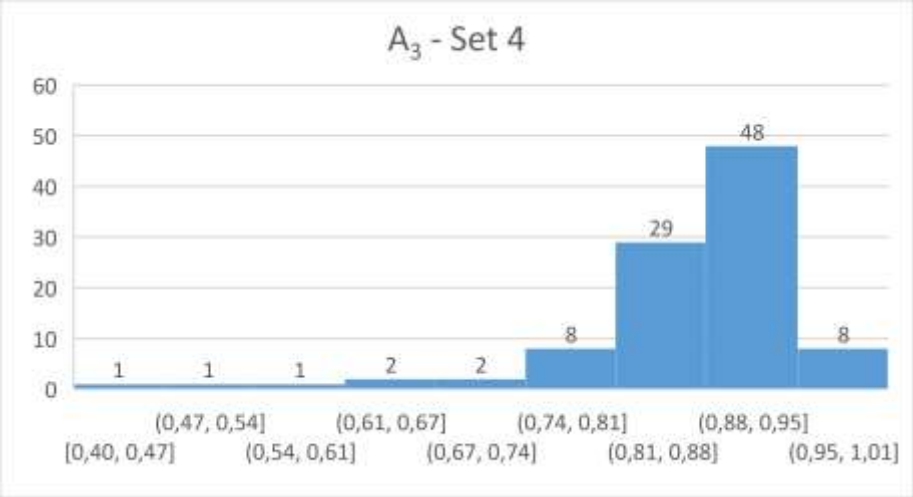
Üretilen modeller arasında en başarılı sonuçları üreten 4 numaralı öznitelik setinin kullanıldığı modelin A_1 yazarının dokümanlarına olan benzerlik değer aralıkları Şekil 4.15'te görüldüğü gibi, normal ve beklenen bir dağılım göstermektedir. Şöyle ki; yazara ait dokümanların büyük bir kısmı benzerlik değerinin yüksek olduğu aralıklarda toplanmıştır. %75 ve altı benzerlik aralığında toplamda 5 doküman bulunurken, en az benzerlik değeri %69'dur. Bu yazarı temsilen üretilen model kullanılarak, harici bir dokümanın benzerlik değeri ölçülerek eğer bu değer %68'in altındaysa A_1 yazarı tarafından yazılmış olma ihtimalinin çok düşük olduğu söylenebilir. Yine A_1 yazarı özelinde dokümanların %88'i 0,8 üzeri benzerlik göstermektedir. Bu durum harici bir dokümanın A_1 yazarını temsilen üretilen modele olan benzerliğinin bu değerlerde olmasının yüksek oranda A_1 yazarı tarafından yazılmış olduğu çıkarımına olanak

sağlamaktadır. Genel bir standart belirleyebilmek için daha farklı örüntüler de değerlendirmeye alınmalıdır. A_2 yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.16'da gösterilmektedir.



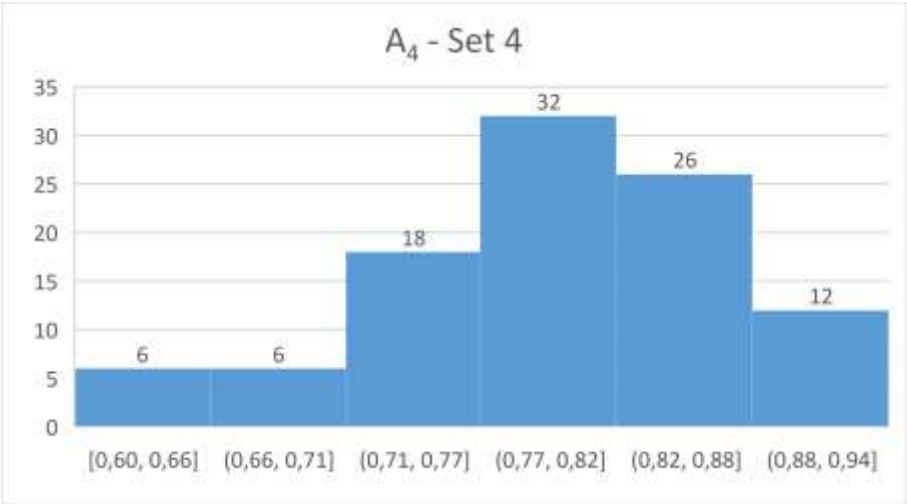
Şekil 4.16. A(2) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

A_2 yazarı özelinde, üretilen temsil modeli kapsamında 3 aykırı doküman bulunduğu Şekil 4.16'da göze çarpmaktadır. Aykırı dokümanlar hem söz konusu yazar üslup sürekliliğinde hem de kullanılan temsil modelinin başarısında farklı değerlendirmelerde bulunulmasını gerektirmektedir. A_2 yazarı özelinde inceleme yapacak olursak, yazara ait dokümanların %97'si 0,7 üzeri benzerlik gösterirken %3'lük bu aykırı değer 0,6 – 0,55 arasında benzerlik göstermektedir. Bu durum sorgulanan harici bir dokümanın üretilen modele benzerliğinin çok düşük de olabileceği, anlamlı bir benzerlikten söz edebilmek için sorgulamada birkaç harici doküman kullanmak gerekliliğini göstermektedir. Ele alınan dokümanların çok büyük bir kısmı üretilen temsil modeline 0,7 üzeri benzerlik gösterdiğinden, sorgulanan harici bir dokümanın üretilen modele benzerliği 0,7 üzerinde ise A_2 yazarı tarafından yazılmış olma ihtimali çok yüksektir. A_3 yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.17'de gösterilmektedir.



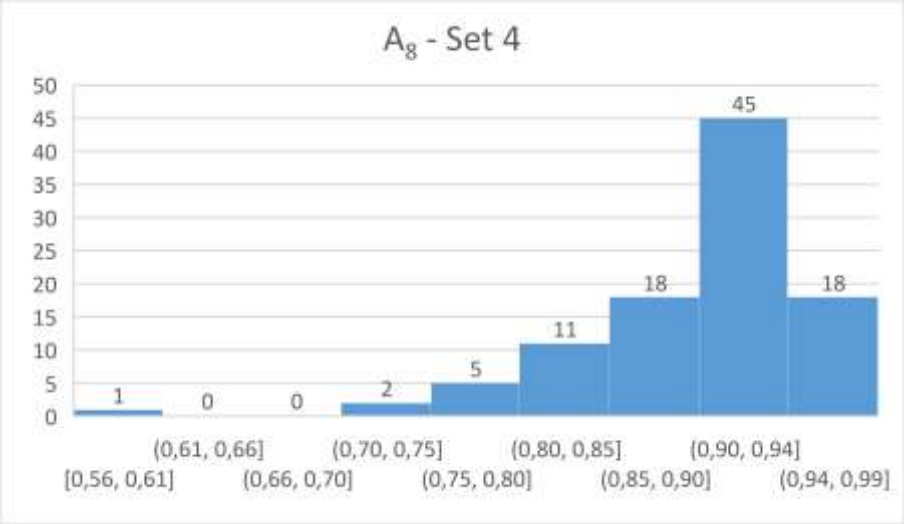
Şekil 4.17. A(3) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

Üretilen temsil modeli kapsamında A_3 yazarına ait dokümanlar içerisinde de zaman veya konu bağlı istisna dokümanlar olduğu Şekil 4.17'de görülmektedir. Sayı olarak az olmalarına karşın değer olarak üretilen modele çok düşük benzerlik göstermeleri bu dokümanların kapsama alınabileceği daha iyi modeller üretilmesi gerekliliğini göstermektedir. Diğer taraftan yazara ait dokümanların %93'ü üretilen temsil modeline 0,74 üzeri benzerlik göstermektedir. Bu sonuç, sorgulanan harici bir dokümanın temsil modeline 0,74 ve üzeri benzerlik göstermesi durumunda A_3 yazarı tarafından yazılmış olma ihtimalinin çok yüksek olduğunu göstermektedir. A_4 yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.18'de gösterilmektedir.



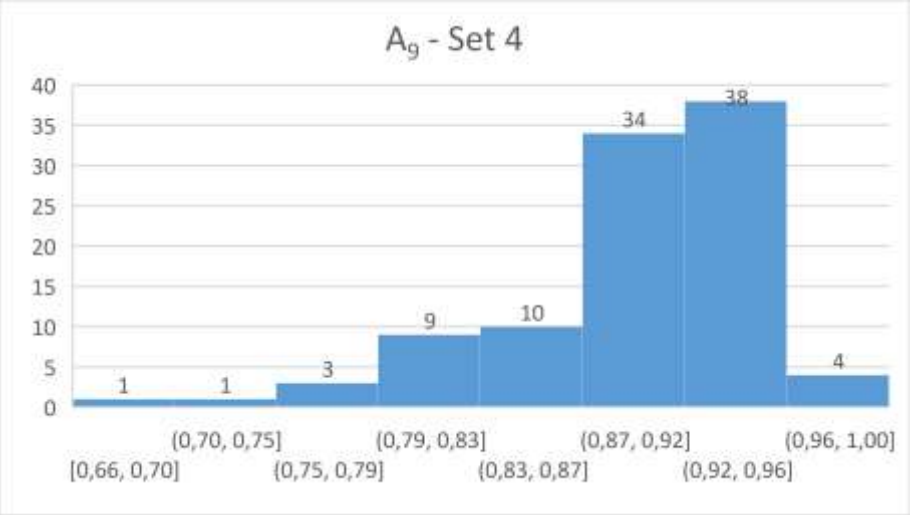
Şekil 4.18. A(4) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

A_4 yazarını temsilen üretilen model, yazara ait dokümanların tamamını kapsayıcı bir örüntü sergilediği Şekil 4.18'de görülebilmektedir. Üretilen model kapsayıcı olmasına rağmen dokümanların %12'si 0,6 - 0,7 aralığında benzerlik oranına sahiptir. Bu durum aslında yazarlara ait dokümanlar arasında da bir benzerlik kıyasının yapılması gerekliliğini göstermektedir. Şöyle ki; "Bir yazara ait her dokümanın o yazarın üslubunu aynı oranda temsil etmesi beklenebilir mi?" sorusu karşımıza çıkmaktadır. A_4 yazarı özelinde değerlendirecek olursak, harici bir dokümanın söz konusu temsil modeline olan benzerlik değeri 0,7 üzerinde ise yüksek oranda bu yazar tarafından yazılmıştır denebilir. A_8 yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.19'da gösterilmektedir.



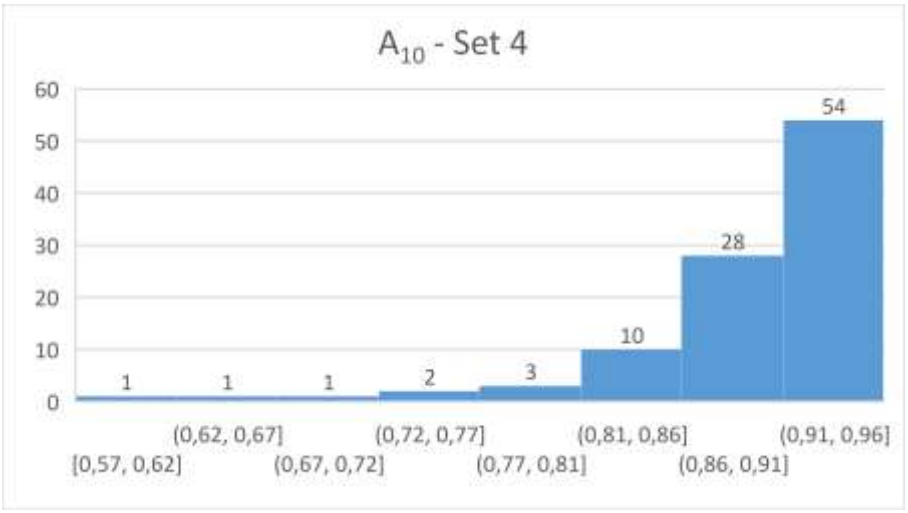
Şekil 4.19. A(8) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

İstisna olarak değerlendirebileceğimiz doküman sayısı Şekil 4.19'da görüleceği üzere, A_8 yazarı özelinde sadece 1 tanedir. Sadece 1 tane olmasına rağmen üretilen temsil modeline benzerlik değeri oldukça düşük olan bu dokümanın içerik olarak bu yazar özelinde ayrıca ele alınması gerekebilir. A_8 yazarı özelinde üretilen modelin dağılımı istisna dışında oldukça başarılıdır. Dokümanların %60'dan fazlası üretilen modele 0,9 üzeri benzerlik göstermektedir. Yine A_8 yazarı özelinde değerlendirme yapmak gerekirse, harici bir dokümanın temsil modele 0,8 üzerinde benzerlik göstermesi yüksek olasılık ile söz konusu yazar tarafından yazılmış bir doküman olduğunun göstergesidir denebilir. A_9 yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.20'de gösterilmektedir.



Şekil 4.20. A(9) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

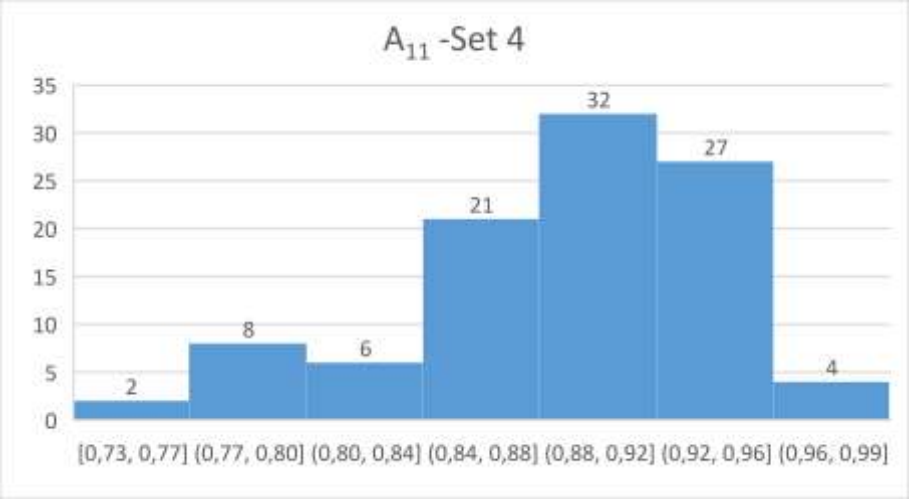
Üretilen temsil modellerinden en başarılı olanlardan bir A_9 yazarını temsilen üretilen modeldir. Şekil 4.20'de görüldüğü üzere üretilen model ile başarılı bir dağılım yakalanmıştır. Şöyle ki; yazara ait dokümanların %95'i üretilen modele %80 üstü benzerlik göstermektedir. Değerlendirmedeki iki doküman 0,66 – 0,75 aralığında olmasına rağmen istisna olarak değerlendirilemeyecek bir aralıkta olduğundan, üretilen modelin kapsayıcı olduğu söylenebilmektedir. A_{10} yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.21'de gösterilmektedir.



Şekil 4.21. A(10) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

Üretilen modelin temsil ettiği yazara ait dokümanlara olan benzerlik dağılımının nasıl olması gerektiğini en iyi gösteren grafik, Şekil 4.21'de gösterilen A_{10} yazarını temsilen üretilen modelin benzeme oranları grafiğidir. Şöyle ki; en çok doküman en yüksek benzerlik aralığında

bulunmaktadır ve benzerlik aralığı arttıkça doküman sayısı da benzer oranda artmıştır. En düşük benzerlik değerine sahip doküman istisna olarak ele alınamasa bile beklentinin altındadır. A_{11} yazarını temsilen üretilen modelin, yazarın dokümanlarına olan benzeme oranlarının gösterildiği grafik Şekil 4.22'de gösterilmektedir.



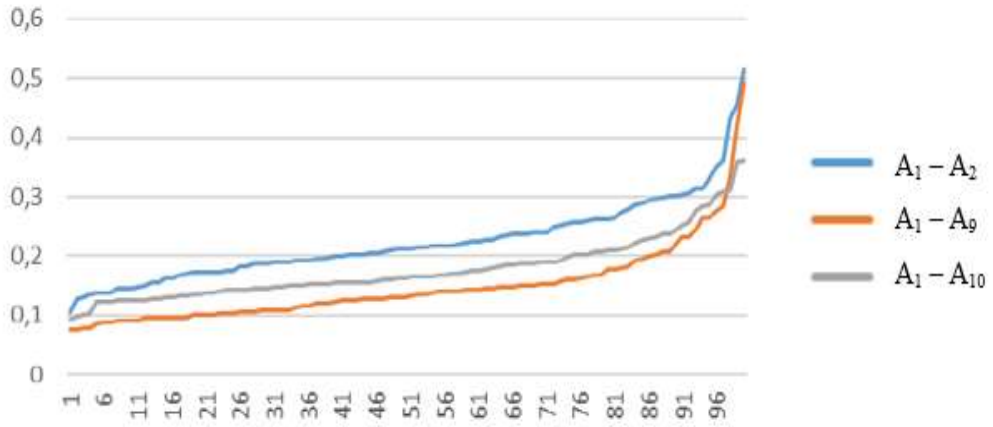
Şekil 4.22. A(11) yazarını temsil modelinin yazarın dokümanlarına benzeme oranları

A_{11} yazarını temsilen üretilen modelin yazara ait dokümanları kapsayıcı bir yapıda olduğu ve ayrıca başarılı bir temsil örüntüsü oluşturduğu Şekil 4.22'de görülmektedir. En düşük benzerlik değerinin bile 0,73 üzeri olması hem söz konusu yazarın belirgin, oturmuş bir üsluba sahip olduğunun hem de üretilen model kullanılarak verilecek kararların yüksek doğruluk ile başarılı olacağının göstergesi olmaktadır.

Genel bir değerlendirmede bulunulması gerekirse, yazar temsil modellerinin söz konusu yazara ait dokümanlara benzerliği bakımından başarılı sonuçlar ürettiği söylenebilir. Ayrıntılı incelemelerde, söz konusu yazarlara ait dokümanların büyük bir kısmının modele yüksek oranda benzerlik gösterirken çok az bir kısmının modelden uzaklaştığı görülmektedir.

Bu çalışmada üretilen yazar modelleri ile bu modellerin üretildiği dokümanlar arası yakınlık büyük oranda yüksek çıkmaktadır. Bu ulaşılmak istenen yani beklenen bir durumdur. Fakat çalışmanın sağlaması gereken önemli bir kriter daha vardır, bu da; bir yazarı temsilen üretilen modelin söz konusu yazarı diğer yazarlardan ayırmada ne kadar başarılı olduğudur. Kısacası, yazarları temsilen üretilen modellerin, başarılı bir şekilde yazarları birbirinden de ayırt edebilmesi gerekmektedir. Üretilen yazar modellerinin yazarları temsil etmede ne kadar başarılı olduğu gibi, yazarları birbirinden ayırt etmede ne kadar iyi sonuç vereceği de yazar doğrulama çalışmaları için büyük önem taşımaktadır.

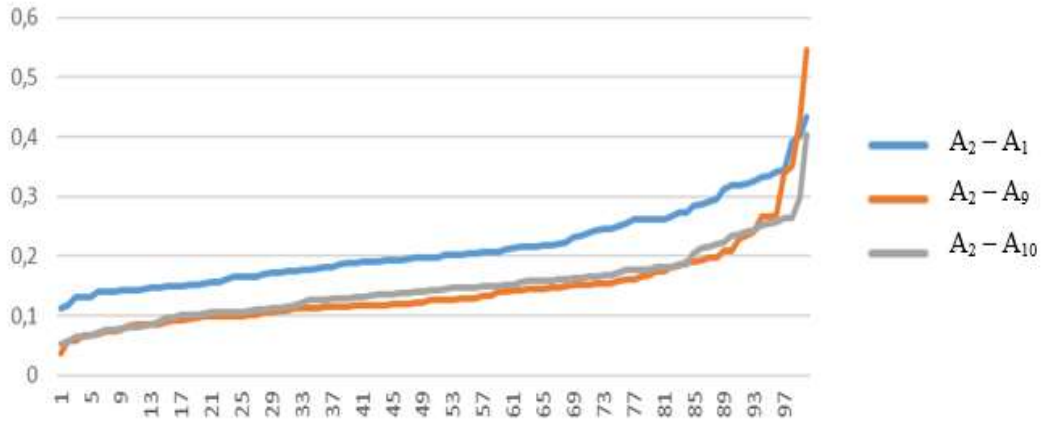
Üretilen modellerin yazar temsilinde yazarlar arası doküman benzerliği bakımından ne kadar ayırt edici olduğu bilgisine ulaşabilmek için veri kümesinde bulunan, farklı davranışlara sahip temsil modelleri olarak A_1, A_2, A_9 ve A_{10} yazarlarını temsilen, 4 numaralı öznitelik seti kullanılarak üretilmiş modeller karşılaştırılmıştır. Seçili dört yazar temsil modeli için, her modelin ait olduğu yazara ait dokümanlar kullanılarak üretilen diğer modellerin bu dokümanlara olan benzerlik değerleri ölçülmüştür. Yapılan karşılaştırmaların grafiksel görünümü ve açıklamaları aşağıdadır. Şekil 4.23'te A_1 yazarına ait dokümanların A_2, A_9 ve A_{10} yazarlarının temsil modellerine olan uzaklıkları gösterilmektedir.



Şekil 4.23. A(1) yazarına ait dokümanların A(2), A(9) ve A(10) yazar temsil modellerine uzaklığı

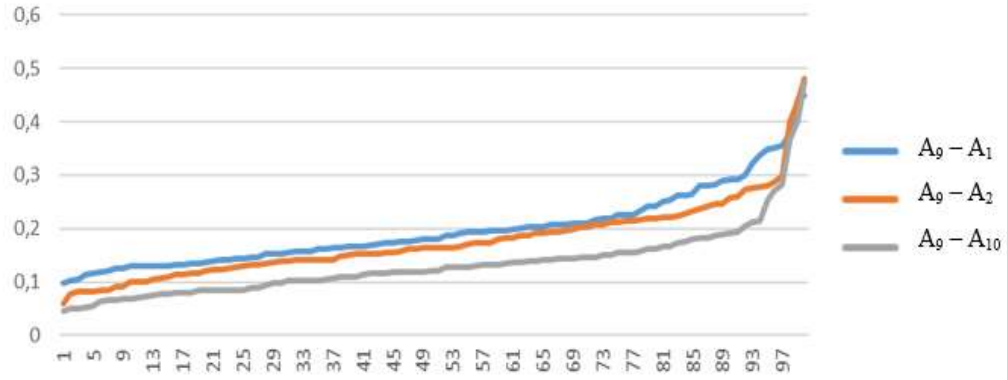
Bir yazara ait dokümanların, başka yazarları temsilen üretilen modele olan uzaklığın düşük yani benzerlik değerlerinin (yakınlığının) yüksek olması, üretilen modellerin yazarları ayırt etmedeki başarısının düşük olduğunu gösterir. Şekil 4.23'te görüldüğü gibi, A_2, A_9 ve A_{10} yazarına ait temsil modeller neredeyse A_1 yazarına ait bir model gibi yazara ait dokümanlara yüksek denebilecek değer aralıklarında yakınlık göstermektedir. A_2 yazarını temsilen üretilen modelin A_1 yazarına ait dokümanlara uzaklığı 0,1 – 0,5 aralığındadır, yani; A_2 yazarını temsil eden modelin A_1 yazarına ait dokümanlara benzeme aralığı %90 - %50 aralığındadır. A_9 yazarını temsilen üretilen modelin ise A_1 yazarına ait dokümanlara uzaklığı 0,08 – 0,5 aralığındadır, yani; A_9 yazarını temsil eden modelin A_1 yazarına ait dokümanlara benzeme aralığı %92 - %50 aralığındadır. A_1 yazarının kendi dokümanlarından üretilen model ile elde edilen benzerlik oranı %97 - %68 aralığındayken A_9 yazarını temsilen üretilen modelin dokümanlara olan benzerlik değeri oldukça yüksek görünmektedir. A_{10} yazarını temsilen üretilen modelin A_1 yazarına ait dokümanlara uzaklığı 0,1 – 0,37 aralığındadır, yani; A_{10}

yazarını temsil eden modelin A_1 yazarına ait dokümanlara benzeme aralığı %90 - %63 aralığındadır. Bu karşılaştırmadan yazarların, ele alınan öznitelikler doğrultusunda birbirine benzer üslup ile yazdıkları çıkarılabilmektedir. Şekil 4.24'te A_2 yazarına ait dokümanların A_1, A_9 ve A_{10} yazarlarının temsil modellerine olan uzaklıkları gösterilmektedir.



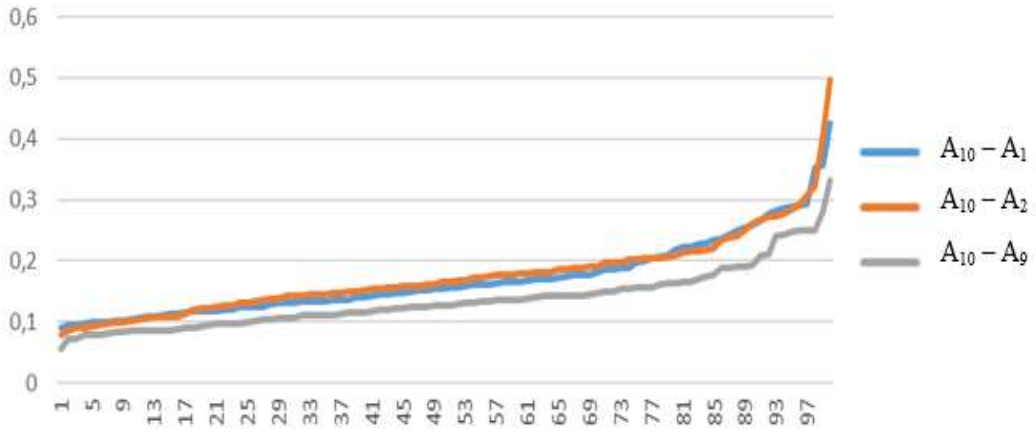
Şekil 4.24. A(2) yazarına ait dokümanların A(1), A(9) ve A(10) yazar temsil modellerine uzaklığı

A_2 yazarına ait dokümanların, A_1, A_9 ve A_{10} yazar temsil modellerine uzaklığı da Şekil 4.24'te görüldüğü üzere beklenenden yüksektir. A_1 yazarını temsilen üretilen modelin A_2 yazarına ait dokümanlara uzaklığı 0,12 – 0,44 aralığındadır, yani; A_1 yazarını temsil eden modelin A_2 yazarına ait dokümanlara benzeme aralığı %88 - %56 aralığındadır. A_9 yazarını temsilen üretilen modelin ise A_2 yazarına ait dokümanlara uzaklığı 0,04 – 0,55 aralığındadır, yani; A_9 yazarını temsil eden modelin A_2 yazarına ait dokümanlara benzeme aralığı %96 - %45 aralığındadır. A_{10} yazarını temsilen üretilen modelin A_2 yazarına ait dokümanlara uzaklığı 0,06 – 0,40 aralığındadır, yani; A_{10} yazarını temsil eden modelin A_2 yazarına ait dokümanlara benzeme aralığı %94 - %60 aralığındadır. A_2 yazarının kendi dokümanlarından üretilen model ile elde edilen benzerlik oranı %98 - %55 aralığındayken A_9 ve A_{10} yazarlarını temsilen üretilen modellerin A_2 yazarının dokümanlarına olan benzerlik değerleri oldukça yüksek görünmektedir. Şekil 4.25'te A_9 yazarına ait dokümanların A_1, A_2 ve A_{10} yazarlarının temsil modellerine olan uzaklıkları gösterilmektedir.



Şekil 4.25. A(9) yazarına ait dokümanların A(1), A(2) ve A(10) yazar temsil modellerine uzaklığı

Yazar temsili bakımından başarılı öznitelikler seçilerek oluşturulan A_1, A_2 ve A_{10} yazar temsil modellerinin A_9 yazarına ait dokümanlara olan uzaklık değerleri Şekil 4.25'te görüldüğü üzere yine beklenenin üstündedir. A_1 yazarını temsilen üretilen modelin A_9 yazarına ait dokümanlara uzaklığı 0,1 – 0,45 aralığındadır, yani; A_1 yazarını temsil eden modelin A_9 yazarına ait dokümanlara benzeme aralığı %90 - %55 aralığındadır. A_2 yazarını temsilen üretilen modelin ise A_9 yazarına ait dokümanlara uzaklığı 0,06 – 0,49 aralığındadır, yani; A_2 yazarını temsil eden modelin A_9 yazarına ait dokümanlara benzeme aralığı %94 - %51 aralığındadır. A_{10} yazarını temsilen üretilen modelin A_9 yazarına ait dokümanlara uzaklığı 0,05 – 0,48 aralığındadır, yani; A_{10} yazarını temsil eden modelin A_9 yazarına ait dokümanlara benzeme aralığı %95 - %52 aralığındadır. A_9 yazarının kendi dokümanlarından üretilen model ile elde edilen benzerlik oranı %99 - %66 aralığındayken A_1, A_2 ve A_{10} yazarlarını temsilen üretilen modellerin A_9 yazarının dokümanlarına olan benzerlik değerleri oldukça yüksek görünmektedir. Şekil 4.26'da A_{10} yazarına ait dokümanların A_1, A_2 ve A_9 yazarlarının temsil modellerine olan uzaklıkları gösterilmektedir.



Şekil 4.26. A(10) yazarına ait dokümanların A(1), A(2) ve A(9) yazar temsil modellerine uzaklığı

Şekil 4.26'da görüldüğü üzere A_1, A_2 ve A_9 yazar temsil modellerinin A_{10} yazarına ait dokümanlara olan uzaklık değerleri beklenenin üstündedir ve yazarları birbirinden ayırt edebilecek nitelikte değildir. A_1 yazarını temsilen üretilen modelin A_{10} yazarına ait dokümanlara uzaklığı 0,1 – 0,44 aralığındadır, yani; A_1 yazarını temsil eden modelin A_{10} yazarına ait dokümanlara benzeme aralığı %90 - %56 aralığındadır. A_2 yazarını temsilen üretilen modelin ise A_{10} yazarına ait dokümanlara uzaklığı 0,08 – 0,5 aralığındadır, yani; A_2 yazarını temsil eden modelin A_{10} yazarına ait dokümanlara benzeme aralığı %92 - %50 aralığındadır. A_9 yazarını temsilen üretilen modelin A_{10} yazarına ait dokümanlara uzaklığı 0,06 – 0,34 aralığındadır, yani; A_9 yazarını temsil eden modelin A_{10} yazarına ait dokümanlara benzeme aralığı %94 - %66 aralığındadır. A_{10} yazarının kendi dokümanlarından üretilen model ile elde edilen benzerlik oranı %96 - %57 aralığındayken A_1, A_2 ve A_9 yazarlarını temsilen üretilen modellerin A_{10} yazarının dokümanlarına olan benzerlik değerleri, A_{10} yazarını temsilen üretilen modele olan benzerlik değeriyle neredeyse başa baş gitmektedir.

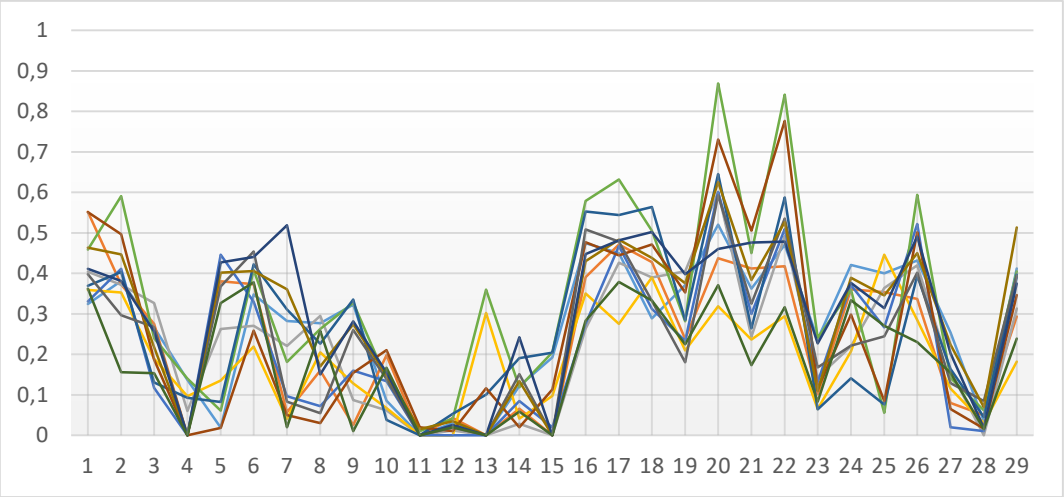
4.5. Tartışma

Yazar doğrulama problemlerinde ele alınan iki doküman arasındaki benzerlik veya yakınlık için anlamlı bir eşik değerin belirlenmeye çalışıldığı bu çalışmada yazar modelleme yaklaşımı çözüm önerisi olarak sunulmuş ve bu amaç doğrultusunda bir takım deneyler yapılmıştır. Deneyler aşamalı olarak gerçekleştirilmiş ve her aşama belirlenen hedefe ulaşmaya yardımcı bir araç olarak çalışmayı şekillendirmiştir. Bu çalışmada ilk olarak bir yazarı temsil etmede kullanılabilecek başarılı bir model üretimi için uygulamalar gerçekleştirilmiştir. Bu uygulamalar farklı öznitelik kümeleri kullanılarak bir yazara ait dokümanların bir araya

getirilmesi ve en iyi temsili üreten öznitelik kümesinin seçilmesini içermektedir. Uygulamalarda kullanılan öznitelik setleri, Türkçe yazar analizi çalışmalarında kullanılan özniteliklerin kategorilerine göre bir araya getirilmesi ile belirlenmiştir. Kullanılan öznitelik kümeleri ile yapılan karşılaştırmalı deneyler sonucunda bir yazarı temsil eden en başarılı öznitelik kümesi belirlenmiş ve çalışmanın sonraki aşamalarında bu öznitelik kümesi kullanılmıştır.

Seçili öznitelik kümesi ile üretilen yazar modelleri, temsil ettikleri yazarların dokümanlarına olan benzerlikleri açısından bakıldığında söz konusu yazarları temsil etmede başarılı sonuçlar vermektedir. Bu aşamada üretilen temsil modellerinin karşılaştırıldıkları dokümanlara karşı genel benzerlik değerleri dikkate alınmıştır. Çünkü yazarlara ait çok nadir bazı dokümanlar (yaklaşık % 1-3 oranında), zaman veya konu farklılığından kaynaklı olabilecek aykırı bir duruş sergilemektedir. Bu bakış açısı ile elde edilen sonuçlar anlamlıdır. Bu anlamlı sonuçların ikinci aşama olarak yorumlanması noktasında her yazar özelinde üretilen başarılı modelin yazar dokümanlarına olan benzerlik oranları ayrıntılı bir biçimde incelenmiştir. Yapılan deneyler sonucunda iki doküman arasındaki yakınlık değeri %100 ile %75 arasında ise "yüksek benzerlik", %75 ile %50 arasında ise "orta benzerlik", %50 ve altında ise "düşük benzerlik" olarak ele alınabileceği görülmüştür.

Çalışmada hedeflenen yazar doğrulama başarısının elde edilebilmesi noktasında üretilen temsil modellerinin yazarları temsilde başarılı sonuçlar ürettiği gibi yazarları birbirinden ayırt etmede de başarılı ve anlamlı sonuçlar üretebilmesi gerekmektedir. Çalışmanın üçüncü aşaması olarak üretilen modellerin yazarlar arası ayırt edicilik başarısını ölçmek amacı ile bir yazara ait dokümanların başka yazarlar için üretilen temsili modellere olan benzerlikleri karşılaştırılmıştır. Bu aşamada yapılan karşılaştırmalar sonucu görülmüştür ki; yazar temsili için üretilen yazar modelleri yüksek oranda birbirine de benzemektedir, yani; bir yazar dokümanları kullanılarak üretilen model başka yazarların dokümanlarına da yüksek oranda yakınlık gösterebilmektedir. Bu sonuçlar kullanılan veri kümesindeki yazarların, kullanılan öznitelik bakımından birbirine yüksek oranda benziyor olması ihtimalini ortaya çıkarmaktadır. Bu ihtimalin hangi oranlarda olduğu ve kullanılan öznitelikler özelinde ele alınan yazarların üslupsal davranışlarını karşılaştırabilmek amacı ile söz konusu yazarların, değerlendirilen öznitelikleri kullanım sıklıkları karşılaştırılmıştır. Şekil 4.27'de veri kümesindeki yazarları temsil modellerinin seçili özniteliklerde aldığı değerler gösterilmektedir.



Şekil 4.27. 12 yazara ait modellerin seçili özniteliklerdeki dağılımları

Şekil 4.27'de görüleceği üzere üretilen yazar modelleri seçili özniteliklerde yakın değerlere yani benzer örüntülere sahiptir. Bu durum veri kümesinde ele alınan yazarların seçili özniteliklere göre üslup ve yapı olarak çok yakın yazılar yazdığını göstermektedir. Bu sonuçlar doğrultusunda üretilen yazar temsil modellerinin temsil ettiği yazara ait dokümanlarını belirlemede başarılı sonuçlar üretmiş olsa da yazarlar arası ayırt edicilikte gerekli başarıyı gösterememiştir. Bu sonuçların en önemli sebeplerinden biri ele alınan veri kümesindeki yazarların işlev, amaç ve yapı olarak benzer yazılar yazmış olmalarından kaynaklanmaktadır. Kullanılan veri kümesi aynı tür ve konuda yazan köşe yazarlarının yazılarını içermektedir. Köşe yazarları profesyonel yazarlar olduğundan gerek noktalamalar bakımından gerekse metnin yapısal özellikleri bakımından tabiri caizse kuralına uygun yazılar üretmişlerdir. Dolayısı ile ele aldığımız öznitelikler bakımından birbirlerine benzer örüntüler elde edilmiş olması bu açıdan anlamlıdır. Elde edilen sonuçlar doğrultusunda devam eden yazar doğrulama çalışmalarında farklı bir veri kümesi ile çalışılması gerekliliği ortaya çıkmıştır. Diğer taraftan ele alınan öznitelik kümeleri de bu veri kümesi için yeteri kadar ayırt edicilik özelliğinde görülmemektedir. Yukarıda da bahsedildiği gibi kurallara bağlı olarak kullanılabilen özniteliklerden ziyade, yazar üslubunu daha ön plana çıkarabilecek daha bağımsız özniteliklerin de kullanılması gerekliliği de bu çalışmada ulaşılan çıkarımlardan biridir. Bu sebepler ışığında sonraki çalışmalarda farklı veri kümeleri, farklı öznitelik kümeleri ve farklı çözüm önerileri ile yazar doğrulama probleminin çözümüne odaklanılacaktır. Bu çalışmadan elde edilen sonuçlar Sinyal İşleme ve İletişim Uygulamaların Kurultayı'nda sözlü bildiri olarak sunulmuştur [50].

5. YAZAR DOĞRULAMA PROBLEMİNİN ÇÖZÜMÜNDE KULLANILACAK UYGUN YAZI TÜRÜ BELİRLEME

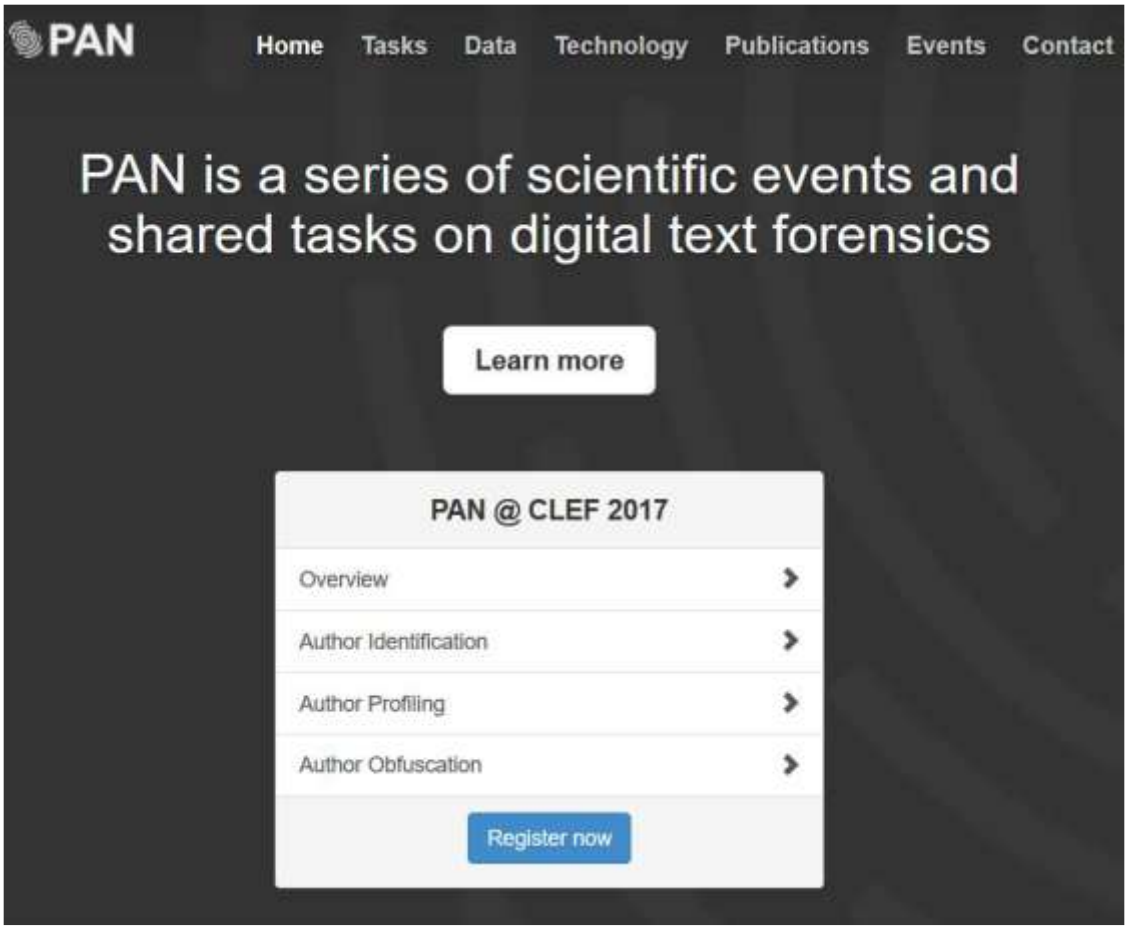
Bu bölümde, yazar doğrulama probleminin çözümünde kullanılması en uygun veri kümesinin hangi türden olması gerektiği üzerine araştırmalar yapılmıştır. Yazar doğrulama problemi, adli bilişim çalışmalarının bir alt dalı ve gerçek bir dünya problemi olduğundan çözümünde kullanılacak veri kümesinin tür seçimi önemlidir. Kullanılacak en uygun yazı türünün belirlenmesi amacı ile literatürde Türkçe ve başka dillerde hem genel yazar analizi çalışmalarında hem de özel olarak yazar doğrulama çalışmalarında kullanılan veri kümeleri değerlendirilmiş, uygunluk durumları sebepleri ile birlikte açıklanmıştır. Bu araştırma sonucunda yazar doğrulama probleminin çözümünde kullanılmasına karar verilen en uygun yazı türü belirlenmiştir ve daha sonraki çalışmalar bu türde oluşturulan bir külliyat kullanılarak yürütülmüştür.

5.1. Türkçe ve Başka Dillerde Yazar Analizi Çalışmalarında Kullanılan Yazı Türü Alan Özeti

Dijital ortamlarda üretilen metinlerin çeşitliliği, bu metinler ile yapılabilecek analizleri de çeşitlendirmektedir. Dijital metinler; kaynak kodları, elektronik postalar, haber grubu mesajları, micro-mesajlar, web forum mesajları, sohbet kayıtları ve blog yazıları gibi birçok farklı tür ve özellikte olabilmektedir.

Günümüze kadar ne yazık ki (bu tez kapsamında yapılan çalışmalar hariç) Türkçe metinler kullanılarak yapılmış yazar doğrulama çalışması bulunmamaktadır. Türkçe metinler üzerinden birçok yazar analizi çalışması yapılmıştır ve bu çalışmalar çoğunlukla yazar tanımlama üzerine yoğunlaşmıştır. Genellikle veri kümesi olarak köşe yazıları kullanılmış ve farklı öznitelik kümeleri ile farklı sınıflama algoritmaları yazar tanımlama yapılmıştır [48]. 12.000 köşe yazısının veri kümesi olarak kullanıldığı bir doktora tezinde yine yazar tanımlamada en başarılı öznitelik kümesinin tespiti üzerine deneyler yapılmıştır [43, 51]. Adli bilişim bakımından elektronik postaların yazarının belirlenmesi amacı ile elektronik postalar ile aynı karakteristiğe sahip haber grubu mesajlarını da veri kümesi olarak kullanan çalışma bulunmaktadır [52]. Oyun platformlarındaki sohbet yazışmalarının veri kümesi olarak kullanıldığı yazar analizi çalışmalarına [53, 54] ek olarak, popüler bir romanın da sayısal üslup araştırmasını ele alan çalışma bulunmaktadır [6].

Yazar doğrulama ve yazar tanımlama çalışmaları uluslararası literatürde birçok defa farklı veri kümeleri kullanılarak yapılmıştır. Özellikle yazar analizi alanına özgü veri yayını yapan PAN organizasyonu [9, 45, 55] bu çalışmalara olan ilgiyi arttırmıştır. Dijital metinlerin adli bilişiminde bir dizi bilimsel olay ve paylaşımlı uygulamayı ele alan PAN organizasyonunun (https://pan.webis.de/) yayınladığı veriler bakımından başka dillerdeki yazar analizi çalışmaları için önemi büyüktür. Şekil 5.1'de PAN organizasyonunun 2017 yılında ele aldığı uygulamalar görünmektedir.



Şekil 5.1. PAN organizasyonunun 2017 yılı giriş sayfası

Şekil 5.1'de web sitesi giriş sayfası verilen PAN organizasyonu 2009 yılından beri yazar analizi çalışmaları kapsamında faaliyette olan yarışma standardında bir platformdur. Yazar analizi genelinde; Yazar Tanıma, Yazar Doğrulama, Yazar Profili Çıkarma, Yazar Gizleme gibi alanlarda her yıl veri seti (derlem - corpus) yayını yaparak bu veri kümelerinden elde edilen başarılı sonuçların karşılaştırması yapılır. PAN organizasyonu tarafından 2013, 2014 ve 2015 yıllarında yazar doğrulama için de veri yayını yapılmıştır. Yapılan veri yayınlarından İngilizce dili için olan veri kümesinin içeriği ve özellikleri aşağıda detaylandırılmıştır.

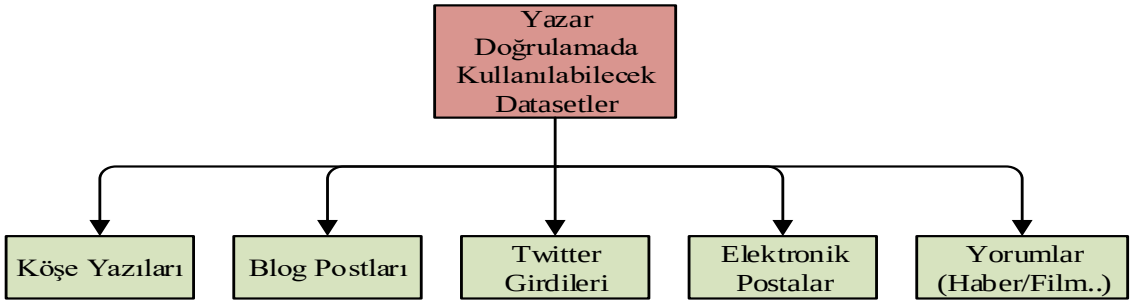
- 2013 eğitim
    - 10 İngilizce klasör, her klasörde [2 6] arası X yazarına ait bilinen dokuman ve 1 bilinmeyen doküman bulunuyor
- 2013 test
    - 30 İngilizce klasör, her klasörde [2 6] arası X yazarına ait bilinen dokuman ve 1 bilinmeyen doküman bulunuyor
- 2014 eğitim
    - 200 tane İngilizce denemeler klasörü, her klasörde [1 6] arası X yazarına ait bilinen dokuman ve 1 bilinmeyen doküman bulunuyor
    - 100 tane İngilizce roman klasörü, her klasörde 1 bilinen 1 bilinmeyen doküman bulunuyor
- 2014 test
    - 200 tane İngilizce denemeler klasörü, her klasörde [1 6] arası X yazarına ait bilinen dokuman ve 1 bilinmeyen doküman bulunuyor
    - 200 tane İngilizce roman klasörü, her klasörde 1 bilinen 1 bilinmeyen doküman bulunuyor
- 2015 eğitim
    - 100 tane İngilizce klasör, her klasörde 1 bilinen 1 bilinmeyen doküman bulunuyor.
    - 500 tane İngilizce klasör, her klasörde 1 bilinen 1 bilinmeyen doküman bulunuyor.

Yukarıda PAN organizasyonu tarafından yayınlanan veri kümelerinden İngilizce dili özelinde yayınlanan yazar doğrulama veri kümesinin ayrıntıları verilmiştir. Bu dilin yanı sıra; İngilizcelere ek olarak Yunanca, İspanyolca ve Flemenkçe veri setleri de bu organizasyon tarafından yayınlanmaktadır.

PAN organizasyonu tarafından yayınlanan veri kümelerine ek olarak blog yazıları [23], elektronik postalar [17, 36], online mesajlar [16], sosyal medya yazışmaları [36], farklı tür ve konu yazıları [18, 32], uç grupların web forum mesajları [38] vb. veri kümeleri kullanılarak farklı alanlarda da yazar tanımlama sorununa başka diller özelinde çözümler sunulmuştur.

5.2. Yazar Doğrulamada Kullanılabilecek Yazı Türü Değerlendirmeleri

Ulusal ve uluslararası alanda yapılmış olan yazar analizi çalışmalarında kullanılan veri kümeleri incelenerek yazar doğrulama probleminin çözümünde kullanılabilecek en uygun yazı türleri seçilmiştir. Şekil 5.2'de bu türlerin bulunduğu veri kümeleri gösterilmektedir.

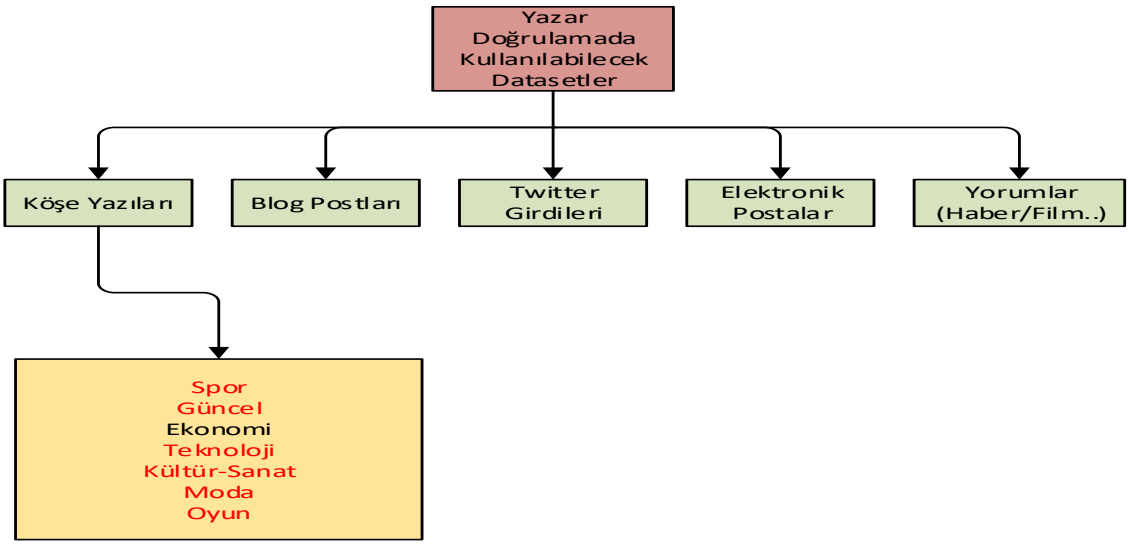


Şekil 5.2. Yazar doğrulamada değerlendirmeye alınacak veri kümeleri

Bu tez çalışmasında kullanılacak yazı türünü ve ele alınacak veri kümesini belirlemek için Şekil 5.2'de belirtilen veri kümeleri tek tek değerlendirilmiştir. Yazar doğrulama problemine uygunluğuna göre önceliğe alınan bu yazıların değerlendirilmesinde, kolaylıkla elde edilebilmesi, tek dokümanın boyut olarak uygun olması, konu çeşitliliğinin sağlanabilmesi yazarın bireysel üslubunu krallardan bağımsız sergileyebilmesi gibi, bazı kriterler ön planda tutularak değerlendirme yapılmıştır. Bu veri kümelerinin ele alınan problem özelindeki avantaj ve dezavantajları karşılaştırılarak en uygun olanın seçimi yapılmıştır. Yapılan değerlendirmeler aşağıdadır.

5.2.1. Köşe Yazıları

Günümüze kadar yapılan Türkçe yazar analizi çalışmalarının çoğunluğunda köşe yazıları kullanılmıştır. Bu durumun sebebi bu yazıların kolay ve istenen miktarda elde edilebilmeleri olabilir. Bu durum yazar analizi çalışmalarını etkileyecek bir durum değildir, fakat köşe yazarlarının, yazma mevzunda profesyonel kişiler olması ve ayrıca yazıların yayınlandığı ortam veya yayın aracının da kısıtlarını barındıracağı göz önüne alınmalıdır. Yazarların profesyonel kişiler olması, yazar analizinde sıklıkla kullanılan birçok özniteliğin aynı şekilde, kurallarına uygun olarak, tabiri caizse kitabına göre kullanılacağı, dolayısı ile analiz aşamasında bir ayırt etme işleminde başarısız olunacağı dikkate alınmalıdır. Diğer taraftan, köşe yazıları konu bağımlı yazılardır. Şekil 5.3'te köşe yazılarının bağlı olduğu kategoriler gösterilmektedir.

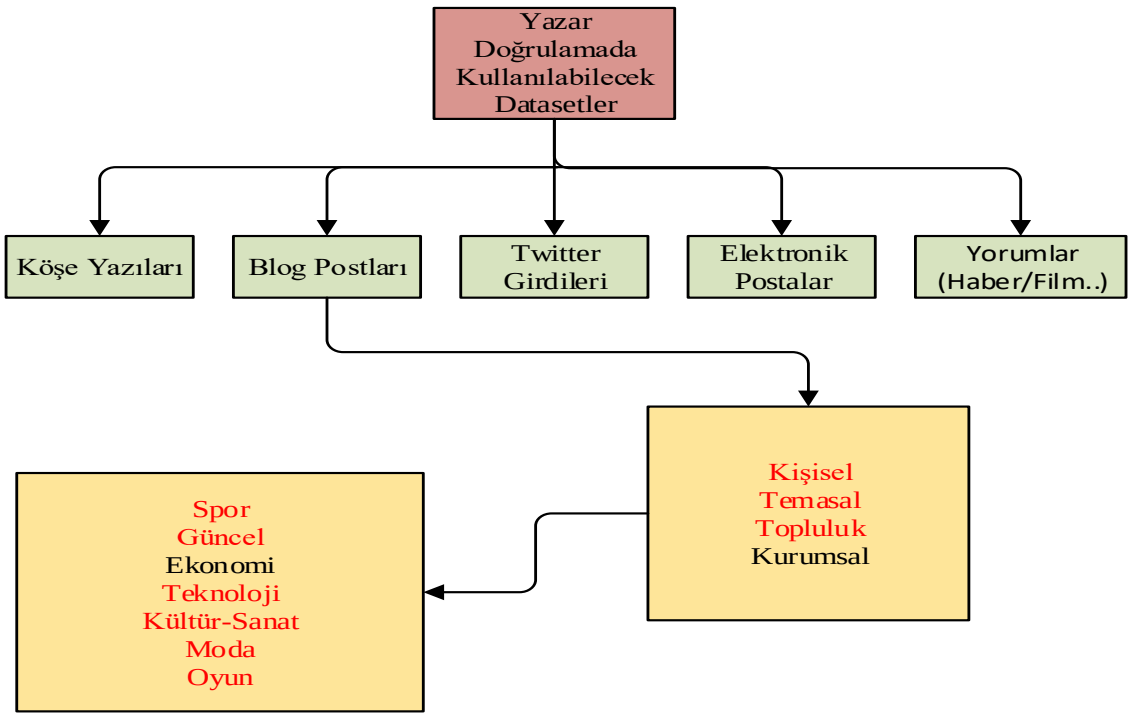


Şekil 5.3. Köşe yazılarının bağlı olduğu bazı kategoriler

Yazar doğrulama çalışmaları, yazar analizi çalışmalarının en zorlusu olarak ele alınmaktadır. Bu alanda ele alınan dokümanların konu ve tür bilgisi gibi dokümanları ayırt etmede yardımcı bilgileri içermemesi bu çalışmaların güvenilirliği açısından önemlidir. Şekil 5.3'te görüldüğü gibi köşe yazıları konu/kategori bağımlı yazılardır, yani; bir yazarın hem ekonomi hem de oyun kategorilerinde yazıyor olması çok düşük ihtimaller dahilinde olduğundan bir köşe yazarına ait konu bağımsız yazılar elde etmek pek mümkün değildir. Dolayısı ile köşe yazarlarının konu bağımsız üslup çıkarımı da yine bu yazılar kullanılarak pek mümkün gözükmemektedir. Bir önceki çalışmalarımızda da görülebileceği üzere köşe yazıları kullanılarak yapılabilecek bir yazar doğrulama çalışmasından beklenen sonuçları almak oldukça zordur. Bu sebeple köşe yazıları kullanılarak yapılabilecek en uygun çalışma belki de aynı kategorilerde yazan yazarlar arası yazar tanımlama çalışması olacaktır.

5.2.2. Blog Postları

İnternet üzerinden kişilerin düzenli olarak, duygularını, düşüncelerini, görüşlerini veya yorumlarını yazılı olarak paylaşımlarda bulunduğu, çoğunlukla kişisel olan web sayfaları blog olarak adlandırılmaktadır. Köşe yazıları kadar doğrudan bulunması kolay olmasa da, blog yazıları da umumi olarak paylaşıldığından elde edilmesi, toplanması mümkündür. Günümüze kadar Türkçe blog yazıları kullanılarak bir yazar analizi çalışma yapılmamış olmasına rağmen başka dillerde yapılan yazar analizi çalışmalarında önemli bir yere sahiptir. Şekil 5.4'te blog yazılarının özellikleri gösterilmektedir.



Şekil 5.4. Blog yazılarının türleri ve özellikleri

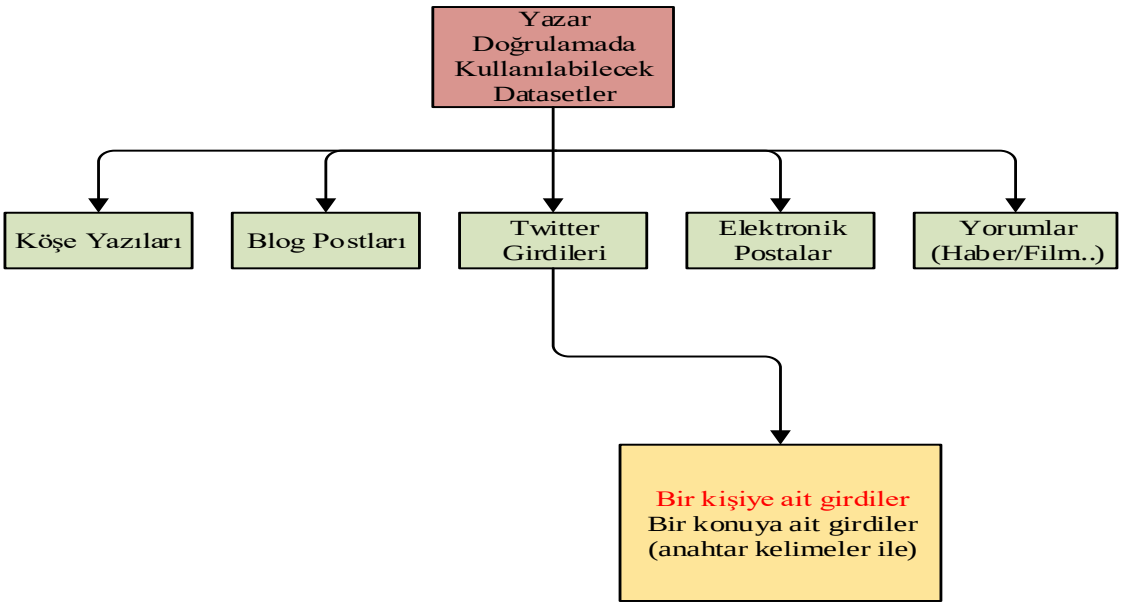
Yaptığımız araştırmalar sonucu 4 farklı tipte blog olduğunu gördük. En yaygın blog tipi kişisel bloglardır. Kişisel bloglar, bir kişinin bir olay, konu, ürün veya durum ile ilgili kişisel görüşlerini paylaştığı bloglardır. Blog yazarlarına blogger, bloggerların paylaşımlarına da post denmektedir. Kişisel bloglarda yazarlar gündelik yaşamlarını, okudukları kitapları, izledikleri filmeleri, kullandıkları ürünleri veya gördükleri bir olayı vb. gibi istedikleri her tür içeriği, konuyu takipçi adı verilen okuyucuları ile paylaşabilmektedirler. Bu içeriklerin oluşturulmasında konu ve dil bilgisi kısıtlaması olmaması, blog yazılarının başka dillerin yazar analizi çalışmalarındaki popülerliğinin en önemli sebebidir. Bu kısıtlamalar olmaksızın yazarın daha kişisel üslubunun ön plana çıkacağı görülmektedir. Diğer taraftan nispeten daha az bir miktarda da olsa, bazı kişisel bloglar da konu bağımlı olabilmektedir. Sadece spor veya moda ile ilgili görüş ve düşüncelerin paylaşıldığı kişisel bloglar da bulunmaktadır, fakat buradaki konu kısıtlaması kişilerin kendi istekleri doğrultusunda konulmuştur ve kişisel üslup daha ön plandadır. Bu özellikleri ile kişisel blog yazıları yazar üslubunu birçok durumdan bağımsız çıkarmaya daha uygundur.

Kişisel bloglar dışında farklı tip bloglar da mevcuttur. Temasal olarak adlandırdığımız blog türleri örneğin, birkaç kişinin sabit bir kategoride paylaşımlarda bulunduğu blog türüdür. Şöyle ki; sürekli teknoloji alanındaki gelişmelerin ele alındığı ve ikiden fazla kişi tarafından yazıların

paylaşıldığı birçok blog ile karşılaşmak mümkündür. Temasal bloglarda genellikle ana fikir Şekil 5.4'te gösterilen alt kategorilerden biri kapsamında yazılar paylaşılmaktadır. Topluluk blogları, ortak bir noktada bir araya gelen bir grup insanın, bu ortak nokta üzerine paylaşım yaptıkları bloglardır. Örneğin hayvan haklarının korunması veya java programlama dilinin yaygınlaşması üzerine paylaşımların bulunduğu bu gibi topluluk bloglarında paylaşımda bulunan birçok yazar vardır. Topluluk blogları da temasal bloglar gibi, Şekil 35'de gösterilen alt kategoriler kapsamında olabilmektedir. Temasal ve topluluk blogları hem birçok yazar tarafından kullanıldığından hem de konu bağımlı olduklarından yazar doğrulama çalışmaları için pek uygun görünmemekle birlikte, başka yazar analizi çalışmalarında kullanımı ile değişik çalışmaların yapılabileceği düşünülmektedir. Bu tip blog yazıları kullanılarak yapılan bir yazar analizi çalışmasına, yapılan araştırmalar boyunca rastlanmamıştır. Son olarak Kurumsal olarak da açılabilen bloglar, bir kuruma ait bilgilendirici veya duyuru içeren paylaşımların yapılabildiği bloglardır. Genellikle bu tip bloglar bir takım söz konusu kurum çalışanları tarafından yapılan paylaşımları içermektedir. Bu tip blog yazıları da yazar analizi çalışmaları için uygun görülmemekle birlikte, kurumsal bilgi çıkarımı gibi çalışmalarda kullanımı ile verimli olabileceği düşünülmektedir.

5.2.3. Twitter Girdileri

Günümüz sosyal medya uygulamalarından en popüleri, microblog teknolojisi olarak da ele alınan Twitter'dır. Twitter, kullanıcılarının 280 karakter ile sınırlı, tweet adı verilen paylaşımlarda bulunabildiği bir sosyal networktür. Türkçe ve başka dillerde, twitter verileri kullanılarak yapılmış birçok metin analizi ve yazar analizi çalışması bulunmaktadır. Bu çalışmaların çoğu umumi bir veri kümesini değil, bazı araçlar kullanarak elde ettikleri verileri kullanmışlardır. Köşe yazıları veya blog yazıları elde etmek kadar kolay olmasa da kullanıma açık bazı araçlar ile twitter üzerinden veri yani paylaşılan metinleri toplamak mümkündür. Twitter uygulamasında girdiler bireysel olmasına rağmen hashtag adı verilen (# işareti ile başlayan) sözcük veya söz ile toplu girdilere de ulaşılması mümkündür. Şekil 5.5'te twitter girdilerinden elde edilebilecek yazılar gösterilmektedir.



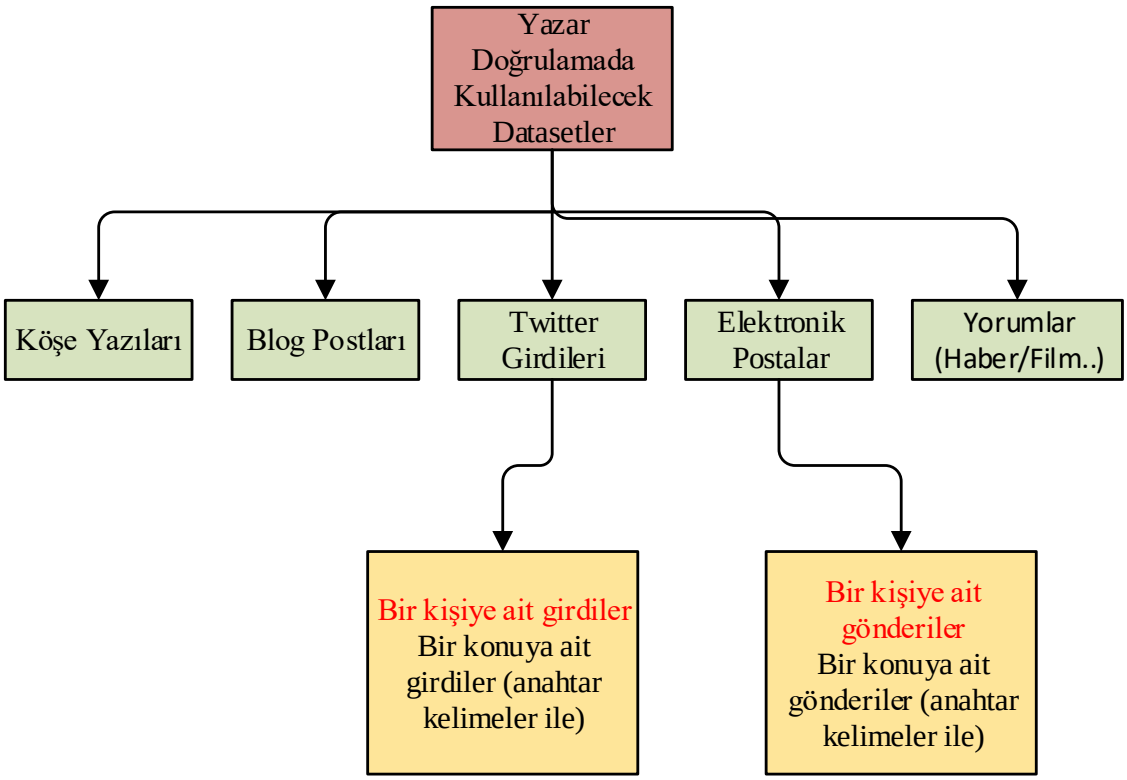
Şekil 5.5. Twitter uygulaması üzerinden elde edilebilecek yazılar

Twitter üzerinden yapılan metin analizi çalışmaları, konu çıkarımı veya metin kategorize etme işlemleri üzerine yoğunlaşmıştır. Bu durumun sebebi yukarıda bahsedilen hashtagler anahtar kelime olarak kullanılarak belli bir konu üzerine birçok kişi tarafından yazılmış yazılara ulaşılabilmesidir. Diğer taraftan yazar analizinde de twitter verileri kullanılmaktadır, fakat burada bir kişiye ait birçok girdinin (tweet'in) birlikte kullanımı söz konusudur. Şöyle ki; twitter uygulamasında kullanılan her tweet için belirlenen karakter kısıtı, bu girdilerin tek başına yazar üslubunu belirlemede yetersiz olmasına sebep olmaktadır. Bir kişiye ait girdilerin toplu olarak alınıp o kişiye ait üslupsal analiz yapılabilmesi ve daha sonraki girdiler için doğrulama yapılması olasılık dahilindedir. Bu tez çalışmasında ele alınan yazar doğrulama problemi, birebir karşılaştırma sonucu yazar doğrulaması yapabilmek üzerine olduğundan ve tek bir twitter girdisinin üslup karşılaştırmasında yeterli olamayacağından, twitter verilerinin kullanımı uygun görülmemiştir.

5.2.4. Elektronik Postalar

Elektronik postalar, yazı türü alanında toplanması pek mümkün olmayan hatta paylaşılması durumunda mahremiyet ihlaline sebebiyet verebilecek yazı türleridir. Bu sebeple Türkçe dili üzerine yapılan bir çalışmada, elektronik postaların yazarının belirlenmesi amacı ile elektronik postalar ile aynı karakteristiğe sahip haber grubu mesajlarını veri kümesi olarak kullanılmıştır [52]. Diğer taraftan, İngilizce yazar analizi çalışmalarında kullanılan, elektronik postaları içeren, umumi bir veri kümesi bulunmaktadır. Enron adında bir şirkette yapılan yolsuzluğun

ortaya çıkarılması üzerine yasal soruşturmalar sırasında ele geçen kurum içi elektronik postaların umumi olarak duyurulması sayesinde Enron email veri kümesi oluşmuştur [56]. Yaklaşık 150 Enron şirketi çalışanı tarafından toplamda yaklaşık 500.000 elektronik posta içeren bu veri kümesinin farklı versyonları hala güncellenerek araştırmacıların kullanımına sunulmaktadır (https://www.cs.cmu.edu/~enron/). Elektronik postalar kapsamındaki yazılar da iki tipte olabilmektedir. Şekil 5.6'da elektronik postalardan elde edilebilecek yazılar gösterilmektedir.



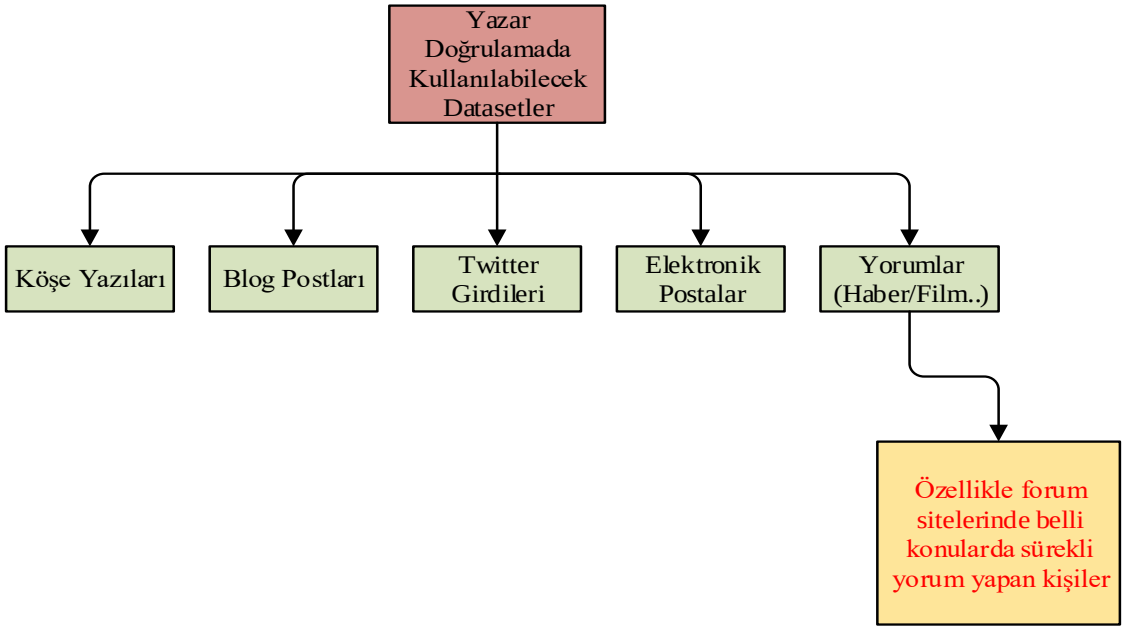
Şekil 5.6. Elektronik postalardan çıkarılabilecek yazılar

Elektronik postalardan da Şekil 5.6'da görülebileceği üzere, twitter verilerinde olduğu gibi bir kişiye ait gönderiler veya belirli bir konu kapsamındaki gönderiler süzülebilir. Farklı olarak elektronik posta verileri kısıtlı bir boyutta değildir ki bu özellik yazar analizi çalışmaları için istenen bir özelliktir. Gönderilen iletilerin ya da üretilen postaların boyutu da yazarların ayırt edilmesinde önemli bir parametre olmaktadır. Fakat yukarıda da bahsedildiği gibi elektronik postaların toplanması, kullanılması ve paylaşılması mahremiyet ihlallerine sebebiyet vermektedir. İngilizce dili özelinde veya dil bağımsız yapılan çalışmalarda Enron veri kümesi kullanılırken, biz bu tez çalışmasında öncelikle Türkçe dili üzerine başarılı bir yazar doğrulama

çözümü sunmak istediğimiz için bu aşamada veri kümesi temini bizim için pek mümkün olmamaktadır.

5.2.5. Yorum Mesajları

Yorum mesajları, özellikle film, haber, dizi, ürün veya uygulama gibi kullanıcıların değerlendirmelerinin önemli olduğu alanlarda veya ürünlerde, web siteleri aracılığıyla kullanıcılardan alınan geri bildirimlerdir. Yorum mesajları da twitter girdileri gibi kısa yazılardır. Dolayısı ile tek bir girdi ile üslupsal analiz yapılabilmesi oldukça zordur. Şekil 5.7'de ürün yorumlarından elde edilebilecek yazılar gösterilmektedir.



Şekil 5.7. Yorum mesajlarından elde edilebilecek yazılar

Yazar analizi çalışmaları kapsamında kullanılacak veriler yazarlara ait veriler olmalı. Kullanıcı yorumlarının alındığı web sitelerinden yorumlanan ürün veya uygulama için verilerin toplanması, kişisel olarak kullanıcıların yaptığı yorumların toplanmasından daha uygundur. Bu gibi sitelerden yönetici izni olmadan söz konusu kişisel yorumların toplanması nispeten zorlu bir işlemdir. Şekil 5.7'de belirtildiği gibi belli veya farkı konularda ama aktif olarak sürekli yorum yapan kişilerin tespit edilmesi ve bu kişilerin yorumlarının toplanması yazar analizi çalışmalarında kullanılabilir. Diğer taraftan bu verilerin yazar analizinde ne kadar başarılı olabileceğine dair bir ön çalışma bulunmadığından, ne kadar veri toplanması gerektiği, yazar

sayısının ne olması gerektiği gibi soruların da cevaplanması için ek çalışmalara ihtiyaç olacaktır.

5.3. Sonuç

Yazar doğrulama probleminin, yazar analizi problemleri arasında en zorlu problem olduğu önceki bölümlerde gerekçeleri ile belirtilmiştir. Bu zorlu probleme başarılı bir çözüm sunabilmek için öncelikle uygun veri kümesinin belirlenmesi gerekliliği bir önceki bölümde yapılan çalışmalar sonucu açık bir şekilde görülebilmektedir. Bu gereklilik göz önüne alınarak, yapılacak tez çalışmasına en uygun veri kümesinin belirlenmesi üzerine bu bölümde olası veri kümeleri üzerine araştırmalar yapılmıştır.

Hem Türkçe hem başka dillerde yazar analizinde kullanılan veri kümeleri incelenmiş, ele aldığımız probleme en uygun veri kümesi olarak da blog yazı tiplerinden oluşan bir veri kümesi kullanılmasının uygun olacağına karar verilmiştir. Literatürde İngilizce blog yazıları kullanılarak yapılmış başarılı bir yazar doğrulama çalışmasının var olması ve bu çalışmada kullanılan veri kümesinin umumi olması alınan kararda etkili olmuştur. Her ne kadar bu tez çalışmasının öncelikli amacı Türkçe metinler üzerine başarılı bir yazar doğrulama modeli geliştirmek olsa da, geliştirilen modelin başka dillerdeki başarısının da başka dillerde oluşturulmuş umumi veriler kullanılarak test edilecek olması planlanmaktadır. Bu planlamalar ve araştırmalar doğrultusunda, bu tez çalışması kapsamında Türkçe bir Blog yazıları Külliyatı oluşturulmasına karar verilmiştir. Sonraki çalışmalarda oluşturulan Türkçe Blog külliyatının özellikleri ve bu külliyat kullanılarak yapılan uygulamalardan elde edilen sonuçlar sunulmaktadır.

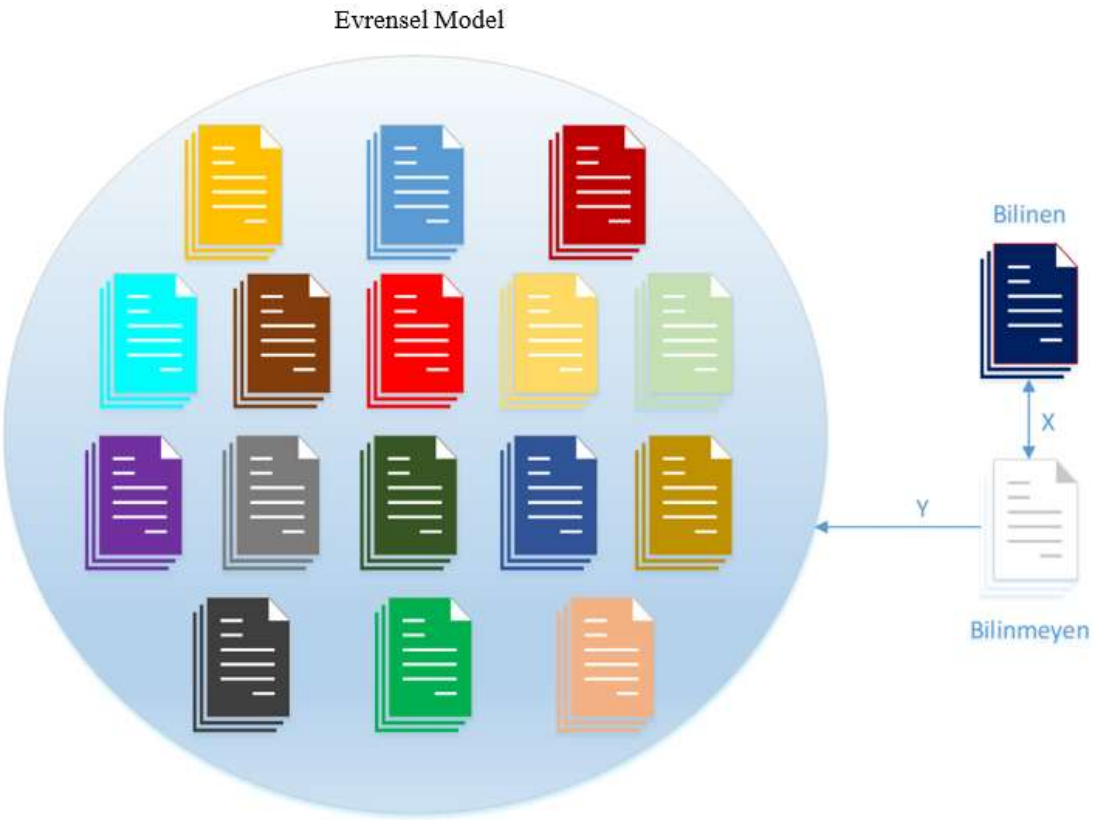
6. TOPLANAN VERİLERİN EVRENSEL MODEL OLUŞTURMAYA UYGUNLUĞUNUN ÖN DENEYİ

Bu bölümde, bir önceki bölümde külliyatı oluşturulmasına karar verilen Türkçe Blog veri kümesinin yazar doğrulama probleminin çözümü için önerdiğimiz evrensel bir arka plan modeli üretimi için uygunluğunun ön deneyleri yapılmıştır. Bu deneylerde amaç ele aldığımız veri kümesinin, ele aldığımız özniteliklere göre doğal dağılımını ve davranışını görmektir. Öyle ki; ele aldığımız verilerin önerdiğimiz modele uygun dağılıma sahip olup olmadığı ile ilgili bilgi çıkarımı yapılabilsin. Bu amaç doğrultusunda, ele alınan veriler kullanılarak farklı sayılarda kümeler oluşturulmuş ve verilerin oluşturulan bu kümelere dağılımı incelenmiştir. Bu incelemeler sonucunda ele aldığımız veri kümesi ile önereceğimiz modelin başarılı olup olamayacağı değerlendirilmiştir.

6.1. Yazar Doğrulama Probleminin Çözümü için Planlanan Evrensel Model

Bilgi teknolojilerinin, özellikle web tabanlı teknolojilerin hızlı ve kontrolsüz gelişimi, suç örgütleri veya terör organizasyonları için iletişim kurulabilecek, yada dışı ürün dağıtımı yapabilecek (korsan yazılım veya çalıntı ürün gibi) anonim bir ortam sağlamaktadır [57]. Suçlular sanal ortamda online mesajlaşmada kimliklerini gizleme eğilimindedirler. Dolayısı ile adli bilişim uzmanlarının suçlular ile mücadelelerinde bu online mesajların anonimliği büyük bir zorluk olarak önlerine çıkmaktadır [57].

Bir metnin karakteristiğinin üslupsal olarak analiz edilmesi ile o metnin yazarı ile ilgili çıkarımlara ulaşmak yazar analizi olarak adlandırılmaktadır. Yukarıdaki bölümlerde bahsedildiği gibi yazar analizi çalışmaları temelde üç başlık altında ele alınmıştır; yazar tanımlama, yazar profil çıkarımı ve yazar doğrulama [3]. Yazar doğrulama problemini ele aldığımız bu çalışmada, bu problemi diğer problemlerden ayıran en önemli etmen olan arka plan eksikliği üzerine evrensel bir model önerilmesi planlanmaktadır. Önerilmesi planlanan modelin gösterimi Şekil 6.1'de gösterilmektedir.



Şekil 6.1. Önerilmesi planlanan Evrensel Model gösterimi

Şekil 6.1'de gösterilen, önermeyi planladığımız evrensel model, birçok farklı türden doküman içermektedir. Bu aşamada; çok sayıda farklı yazardan alınan çok sayıda dokümanın, farklı öznitelikler kullanılması ile bir araya getirilerek evrensel bir modeli temsil etmesi beklenmektedir. Şekil 6.1'de, evrensel model dışında mavi renk ile gösterilen dokümanlar, yazar doğrulama problemi kapsamında ele alınan şüpheli yazara ait olduğu bilinen dokümanları temsil etmektedir. Yine Şekil 6.1'de, evrensel model dışında beyaz renk ile gösterilen doküman ise, şüpheli yazara ait olup olmadığı belirlenmek istenen, yazarı bilinmeyen yani sorgulanan dokümanı temsil etmektedir. Önerilmesi planlanan modelin, yazar doğrulama probleminin çözümüne yönelik yaklaşımı aşağıdaki gibidir.

*Eğer sorgulanan yazara ait olduğu bilinen dokümanlar ile yazarı bilinmeyen doküman arasındaki uzaklık, yazarı bilinmeyen doküman ile evrensel model arasındaki uzaklıktan **fazla** ise; yazarı bilinmeyen doküman şüpheli yazar tarafından **yazılmamıştır** denir. Eğer sorgulanan yazara ait olduğu bilinen dokümanlar ile yazarı bilinmeyen doküman arasındaki uzaklık, yazarı bilinmeyen doküman ile evrensel model arasındaki uzaklıktan **az** ise; yazarı bilinmeyen doküman şüpheli yazar tarafından **yazılmıştır** denir.*

Önerilmesi planlanan modelin, yazar doğrulama probleminin çözümüne yönelik yaklaşımının matematiksel ifadesi aşağıdaki gibidir.

Eğer,

X <= Y; sorgulanan doküman şüpheli yazara aittir

Eğer,

X > Y; sorgulanan doküman şüpheli yazara ait değildir

Önerilmesi planlanan modelin karar kuralları belirlendikten sonra bir önceki bölümde külliyat oluşturulması hedeflenen blog veri kümesinin bir kısmı ile toplanacak verilerin doğal davranışının ortak bir model oluşturmaya elverişli olup olmadığı bir takım kümeleme işlemleri ile ölçülmüştür. Ölçme işlemlerinin ayrıntıları ve elde edilen sonuçların yorumları bir sonraki bölümde bulunmaktadır.

6.2. Blog Verilerin Evrensel Model Oluşturmaya Uygunluğunun Gözlemlenmesi

Belirli bir alanda iyi tanımlı bir külliyat oluşturmak zaman alıcı bir süreçtir. Bir önceki bölümde karar verildiği üzere, bu tez çalışması kapsamında blog yazılarını içeren bir külliyat oluşturulmaktadır. Yine bir önceki süreçte ayrıntıları verildiği üzere, blog yazıları çok çeşitli ve çok yazarlı olabilmektedir. Bu yazılar arasında bir yazara ait olduğundan emin olduğumuz yazıları elde edebilmek için her yazının özenle incelenmesi gerekmektedir. Dolayısı ile amaçlanan külliyat oluşumu uzun bir zaman alacaktır. Bu süreçte, önermeyi planladığımız modelin anlamlı olup olmayacağını, toplanan bir kısım blog verisi ile bir takım testler yaparak görmek istemiş bulunmaktayız. Bu aşamada gözlemlemek istediğimiz durum; toplanan verilerin doğasında bir model barındırıp barındırmadığıdır.

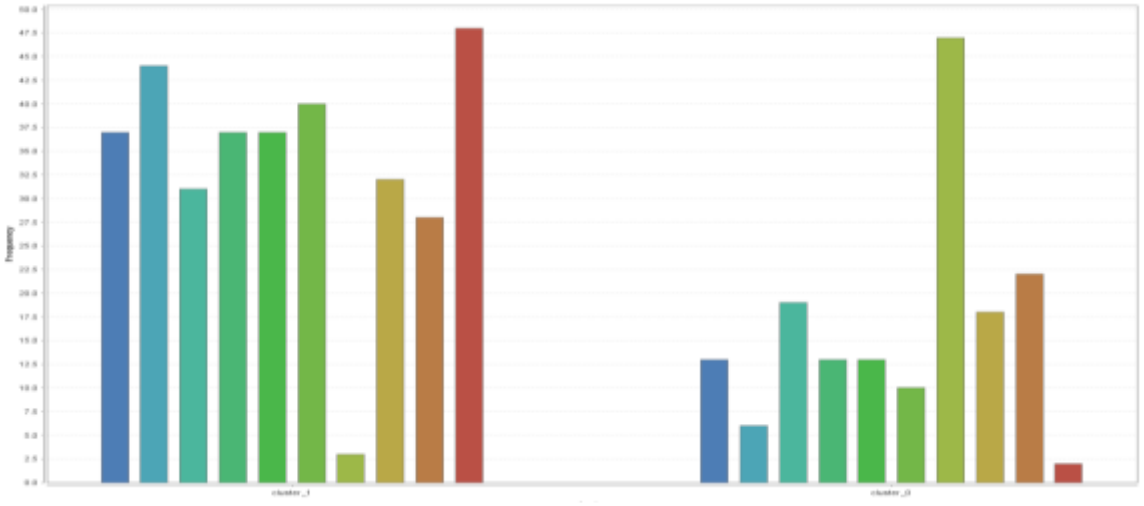
Mevcut durumda toplamış olduğumuz 10 yazara ait 50'şer blog yazısının bulunduğu veri kümesi kullanılarak farklı sayılarda kümeleme sonuçlarının karşılaştırılmasına ve verinin doğasında farklı kümeler barındırıp barındırmadığına bakılmasına karar verilmiştir.

Bu bölümde yapılacak gözlemler için; kelime çantası (BagOfWords - bow) adı verilen özniteliklerin kullanım sıklıkları, önceki bölümlerde başarılı sonuçlar üreten noktalamalar özniteliklerinin kullanım sıklıkları ve bu iki kümenin birleşiminden oluşan karma öznitelik kümesi kullanılmıştır. Bu üç farklı öznitelik kümesi kullanılarak vektörel temsilleri oluşturulan dokümanların (10 yazardan 50'şer doküman olmak üzere toplam 500 dokümanın) 2'li, 5'li, 10'lu ve 20'li kümeleri, üç öznitelik kümesi için de üretilmiştir. Kümeleme işlemlerinde k-means kümeleme algoritması [58] kullanılmıştır. Her bir doküman n tane özniteliği içeren n-boyutlu bir reel vektör olmak üzere $\{x_1, x_2, x_3, \dots, x_n\}$ haline dönüştürülmektedir. k oluşturulacak küme sayısı olmak üzere; tüm dokümanlar $D = \{D_1, D_2, \dots, D_k\}$ k tane kümeye ayrılır. μ_i, i. kümenin merkez noktasını veya ortalamasını temsil etmek üzere, k-means algoritmasına göre kümeleme işlemi için Denklem (10) ve Denklem (11)'de verilen işlemler kullanılmaktadır.

$$\mu_i = \frac{1}{|D_j|} \sum_{x_i \in D_j} x_i \tag{10}$$

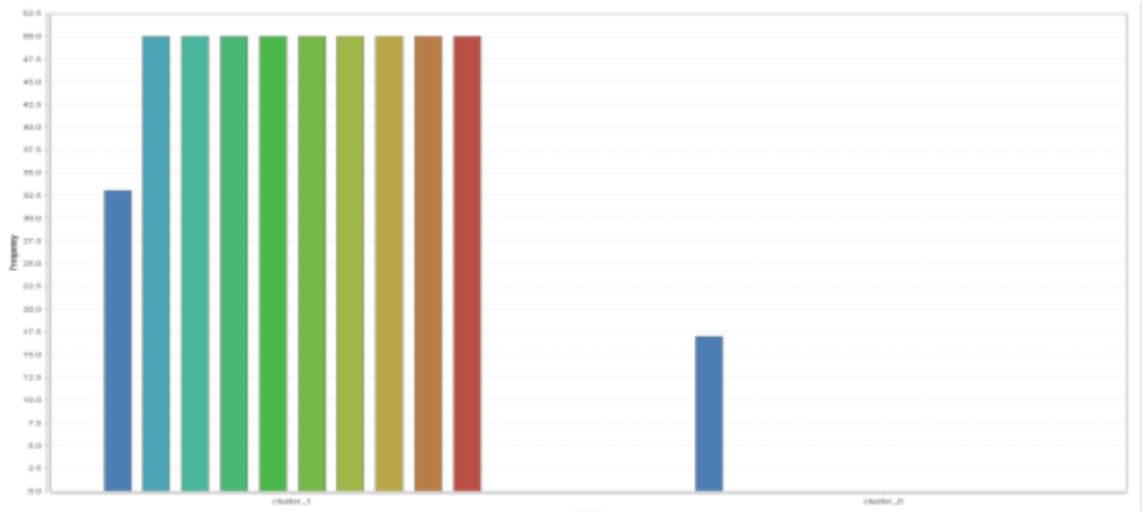
$$\underset{D}{arg\,min} \sum_{j=1}^{k} \sum_{x_i \in D_j} ||x_i - \mu_j||^2 \tag{11}$$

Kullanılan öznitelik kümelerinden Bow kümesi içerdiği öznitelik sayısı bakımından boyut fazlalığı oluşturduğundan bu aşamada Bilgi kazanımı öznitelik ağırlıklandırma algoritması (algoritmanın ayrıntıları ileriki bölümlerde verilecektir) kullanılmıştır. Kullanılan ağırlıklandırma algoritmasına göre en ayırt edici 100 öznitelik testlerde kullanılmıştır (en ayırt edici 500 ve 1000 öznitelik de testlerde kullanılmış fakat anlamlı bir fark görülmediğinden 100 öznitelik tercih edilmiştir). Şekil 6.2'de ele alınan on yazarın dokümanlarının noktalamalar öznitelik kümesi kullanılarak iki kümeye dağılımı görülmektedir.



Şekil 6.2. Noktalamalar öznitelik kümesine göre verilerin 2 kümeye dağılımı

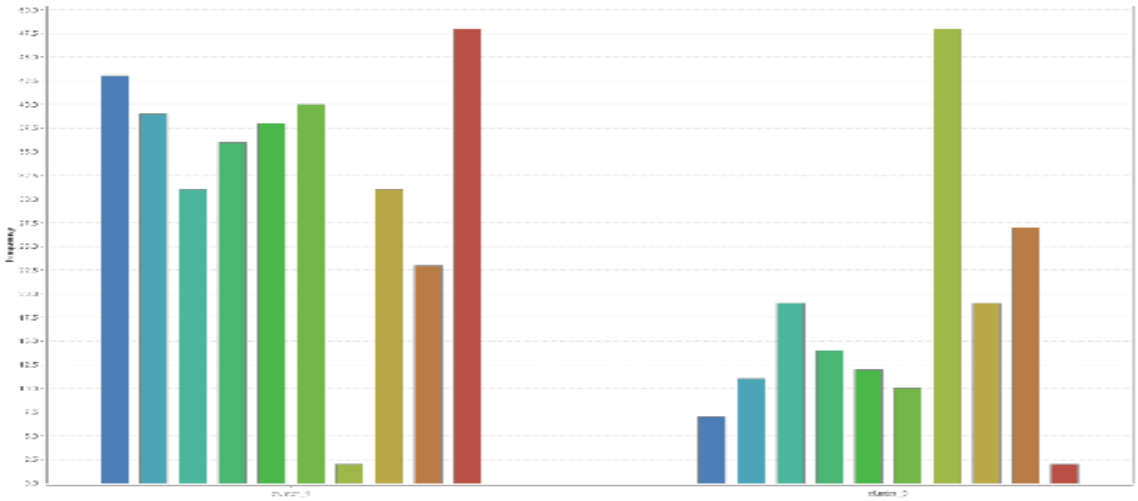
Şekil 6.2'de verilen grafik, 10 yazara ait dokümanların k-means kümeleme algoritması kullanılarak iki kümeye dağılımının grafiğidir. Verilen grafikte her renk bir yazara ait dokümanları temsil ederken, yatay eksen kümeleri, düşey eksen de doküman sayılarını temsil etmektedir. Bu dağılımda da 4. Bölümde karşılaşılan aykırı yazılar benzeri bir dağılım görülmektedir. Oluşturulan iki kümede de her yazardan dokümanlar bulunmakta fakat ilk küme daha kapsayıcı görülmektedir. İkinci küme için her yazardan az miktarda doküman barındırdığından daha yazar bağımlı istisnalardan bahsedilebilir. Şekil 6.3'te veri kümesinin bow öznitelik kümesi kullanılarak iki kümeye dağılımı görülmektedir.



Şekil 6.3. Bow öznitelik kümesine göre verilerin 2 kümeye dağılımı

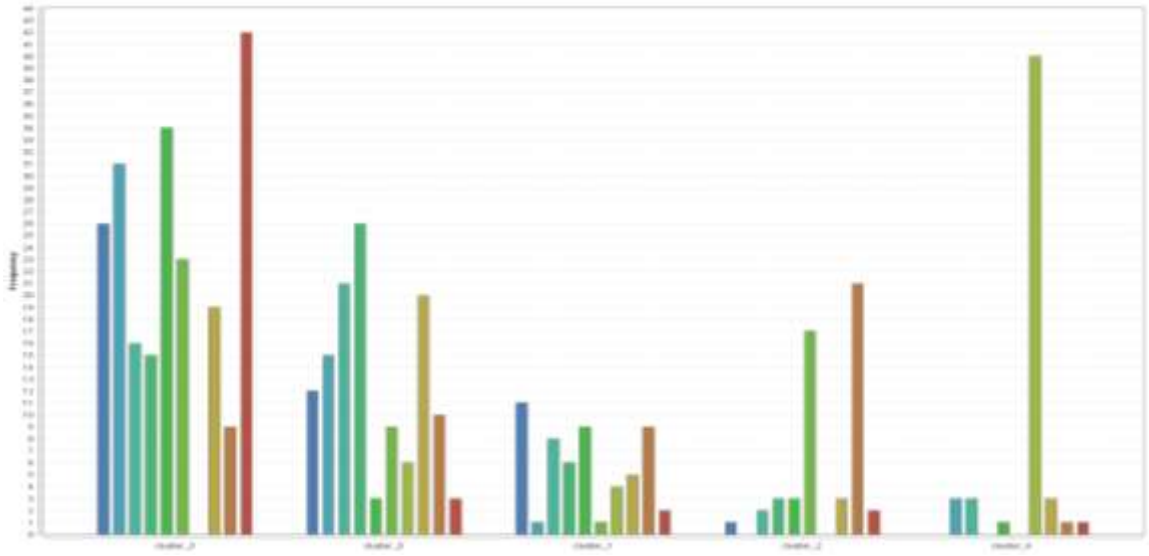
Bow öznitelik değerleri ile yapılan kümeleme sonuçları Şekil 6.3'te görüldüğü üzere, önermeyi planladığımız yaklaşım bakımından daha umut verici bir dağılım sergilemektedir. Şekilde

sadece ilk yazara ait bir takım dokümanlar ikinci kümeyi oluşturmaktayken diğer tüm dokümanlar birinci kümede bulunmaktadır. Bu durum bow öznitelik değerleri kullanılarak önerilmesi planlanan modelde başarılı sonuçlar elde edilebileceğine dair olumlu bir görseldir. Diğer taraftan bu tespit yeterli değildir, daha fazla küme oluşturularak verilerin davranışının daha ayrıntılı ele alınması gereklidir. Bunun yanında noktalamalar öznitelik kümesinin bilgilendirici özellikler barındırdığı hem literatürdeki çalışmalar ile hem de önceki bölümlerde yapılan çalışmalara ile görülmüştür. Dolayısı ile bow ve noktalamalar öznitelik kümelerinin birlikte kullanımında verilerin nasıl bir dağılıma sahip olacağının da incelenmesi gerekmektedir. Şekil 6.4'te, bu iki öznitelik kümesinin birleşiminden oluşan karma adını verdiğimiz öznitelik kümesinin kullanımı ile verilerin iki kümeye dağılımı görülmektedir.



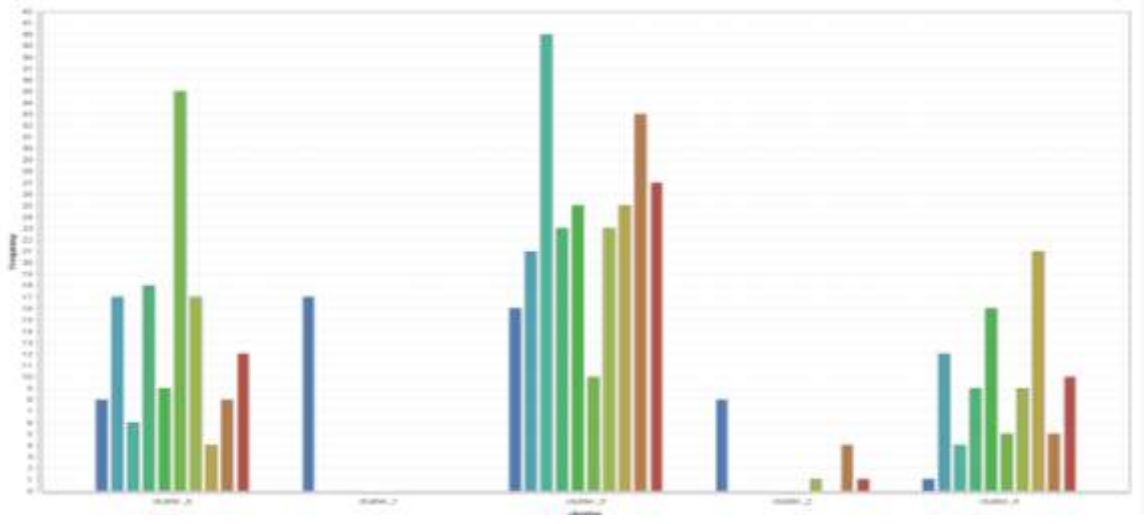
Şekil 6.4. Karma öznitelik kümesine göre verilerin 2 kümeye dağılımı

Karma öznitelik kümesi kullanılarak elde edilen küme dağılımları Şekil 6.4'te görüldüğü üzere noktalamalar öznitelik kümesinin dağılımına benzer bir dağılım sergilemektedir. Üretilen her iki kümede de veri kümesinde bulunan her yazara ait doküman bulunmaktadır. Bu dağılımda, ikinci kümenin aykırı yazı üslubunu temsil ettiğini varsayarsak bile bazı yazarların birçok dokümanı bu üsluba yatkın görülmektedir. Daha sağlam çıkarımlarda bulunabilmek için veri kümesinin farklı küme sayılarındaki dağılımlarının da incelenmesi gerekmektedir. Şekil 6.5'te Noktalamalar öznitelik kümesi ile veri kümesinin k-means algoritması ile 5 kümeye dağılımı gösterilmektedir.



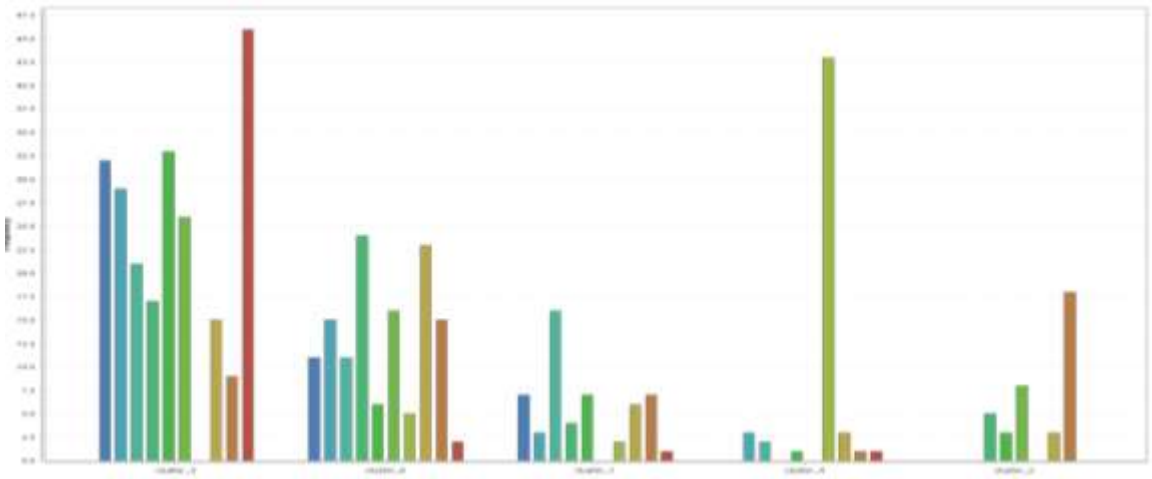
Şekil 6.5. Noktalamalar öznitelik kümesine göre verilerin 5 kümeye dağılımı

Noktalamalar öznitelik kümesine göre verilerin 5 kümeye dağılımı, Şekil 6.5'te görüldüğü üzere, 2 kümeye dağılımda olduğu gibi daha homojen bir dağılım sergilemektedir. Yani üretilen birinci ve beşinci küme dışındaki kümelerde birçok yazardan farklı sayıda doküman bulunmaktadır. Birinci küme daha kapsayıcı bir küme olarak 9 yazara ait çok sayıda dokümanı barındırırken, beş numaralı kümeye bakarak üretilen kümelerin daha yazar temsili kümeler olma eğiliminde olduğu çıkarımı yapılabilir. Bu çıkarımı daha sağlam yapabilmek için daha fazla küme sayılarında verinin dağılımı incelenecektir. Şekil 6.6'da bow veri kümesi kullanılarak eldeki veri kümesinin 5 kümeye dağılımı görülmektedir.



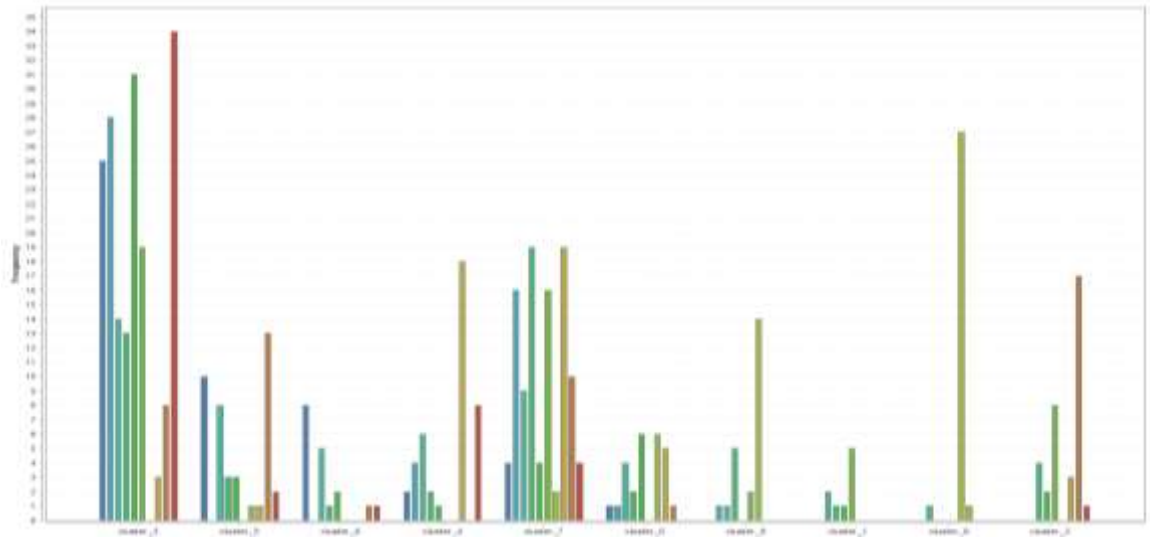
Şekil 6.6. Bow öznitelik kümesine göre verilerin 5 kümeye dağılımı

Şekil 6.6'daki veri dağılımı incelendiğinde üretilen üç kümenin de tüm yazarlara ait bir kısım dokümanı barındırdığı görülmektedir. Bu durum, önerilmesi planlanan evrensel modelin, tek bir temsilinin değil birden çok temsilinin olabileceğini göstermektedir. Yani; aslında tek bir evrensel model değil birden çok evrensel model olabilir. Bu çıkarım dokümanların içerdikleri konulara bağlı da olabilmektedir. Şöyle ki; evrensel modeller yazar üslubuna göre şekillenmiş olsaydı her kümede bir takım yazarlar baskın olurdu fakat her kümede her yazara ait dokümanlar var ise bu durum içerik ile alakalı bir bağlantı sonucu oluşmaktadır sonucu çıkarılabilir. Şekil 6.7'de karma öznitelik kümesi kullanılarak üretilen verilerin 5 kümeye dağılımı görülmektedir.



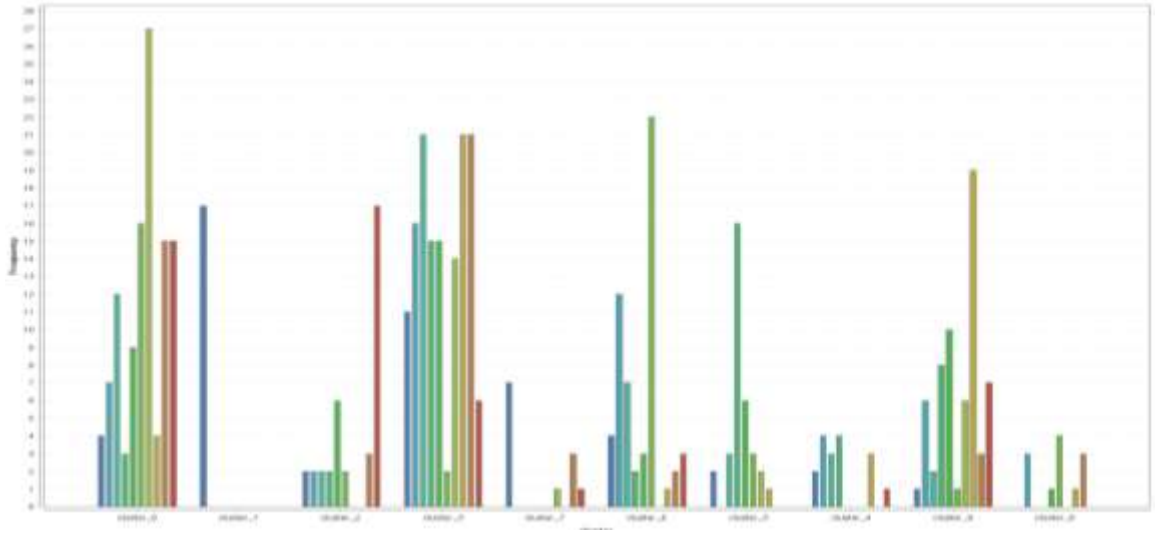
Şekil 6.7. Karma öznitelik kümesine göre verilerin 5 kümeye dağılımı

Kullanılan veri kümesinin karma öznitelikler ile beş kümeye ayrılması ile elde edilen dağılım, Şekil 6.7'de görüldüğü üzere küme sayısı arttıkça hem noktalamalar hem de bow öznitelik kümesi ile elde edilen sonuçların bir ortalaması şeklini almaktadır. Bu dağılım özelinde de tam olarak üretilen kümelerin ortak bir modeli temsil edebileceği veya yazar üslubuna dayalı kümeler üretilebileceği çıkarımı yapılamamaktadır. Bu sonuçların yetersizliği karşısında, üretilen kümelerin yazar üslubu dağılımını daha net görebilmek için yazar sayısı kadar kümeler üretilmesine karar verilmiştir. Şekil 6.8'de noktalamalar öznitelik kümesi kullanılarak veri kümesinin 10 kümeye dağılımı gösterilmektedir.



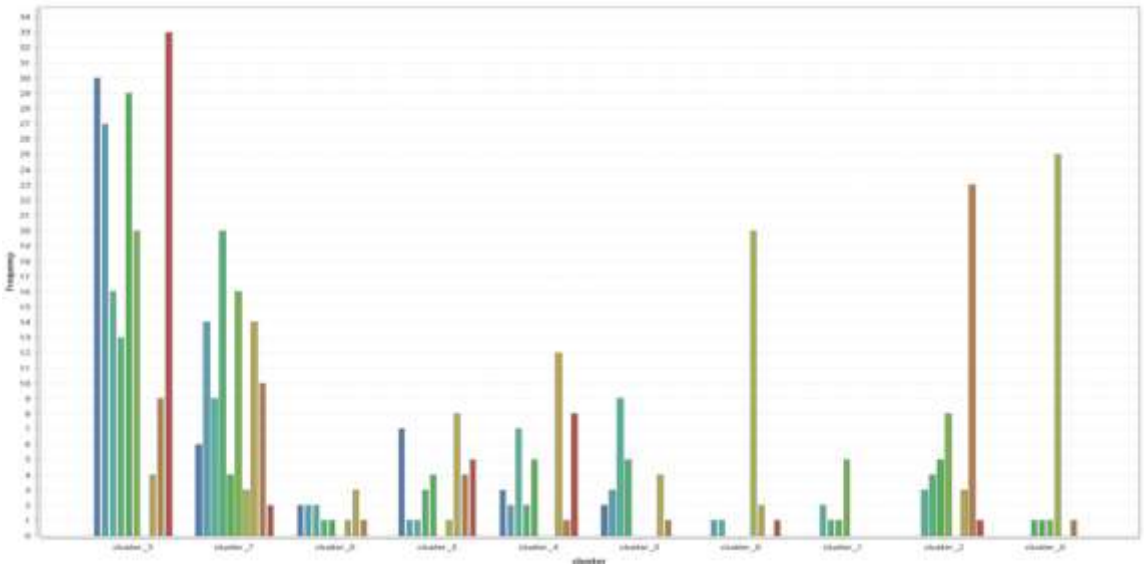
Şekil 6.8. Noktalamalar öznitelik kümesine göre verilerin 10 kümeye dağılımı

Önceki 2 ve 5 kümeye dağılımlarına göre, Şekil 6.8'de görüldüğü üzere noktalamalar öznitelikleri kullanılarak kümelenen veriler yine homojen bir dağılım gösteriyor olmasına rağmen bazı kümelerde bazı yazarların daha baskın olduğu söylenebilir. Bu oran yazarların kümeler halinde ayrılabileceği çıkarımının yapılabileceği bir oran değildir. Dolayısı ile bu noktada sadece bazı yazarların bu öznitelikler özelinde ayırt edilebileceği çıkarımı yapılabilir. Diğer taraftan birinci küme içeriği incelendiğinde yazarlara ait dokümanların çoğunun tek bir yerde kümelenme eğiliminin hala var olması evrensel modelin nispeten etkili olabileceğinin bir göstergesidir. Şekil 6.9'da bow öznitelik kümesi kullanılarak üretilen verilerin 10 kümeye dağılımı görülmektedir.



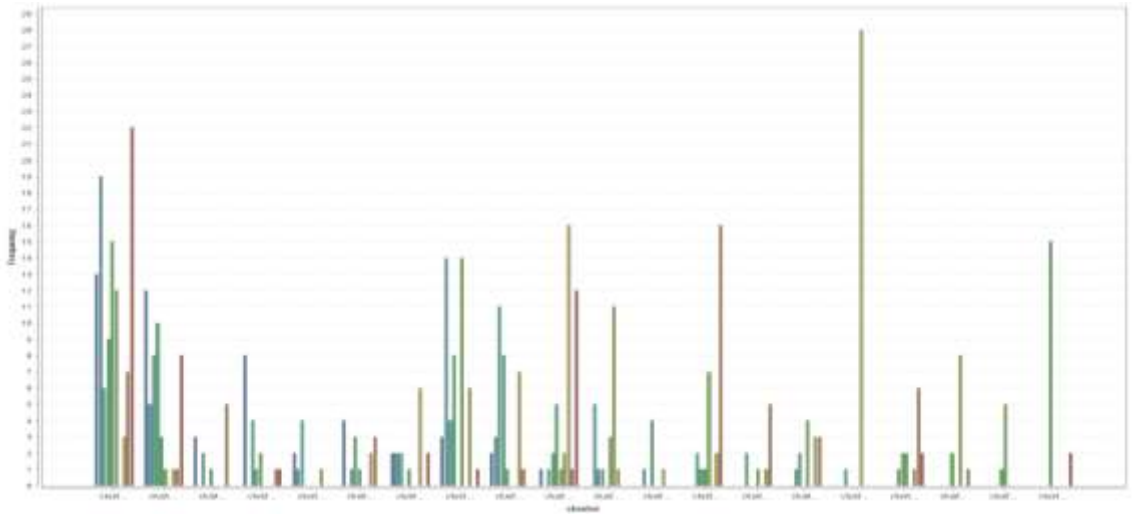
Şekil 6.9. Bow öznitelik kümesine göre verilerin 10 kümeye dağılımı

Bow özniteliklerinin kullanımı ile elde edilen verilerin on kümeye dağılımı Şekil 6.9'da görüldüğü üzere evrensel model mantığına uygun bir düzeydedir. Şöyle ki; üretilen kümelerin çoğunda faklı yazarlara ait dokümanlar bir arada bulunmaktadır. Noktalamalar dağılımında olduğu gibi burada da bazı kümeler bazı yazarları daha çok temsil etme eğiliminde olsalar da genel izlenim yazarlar arası ortak noktaların tespit edilebileceği üzerinedir. Şekil 6.10'da karma öznitelik kümesi kullanılarak üretilen verilerin 10 kümeye dağılımı görülmektedir.



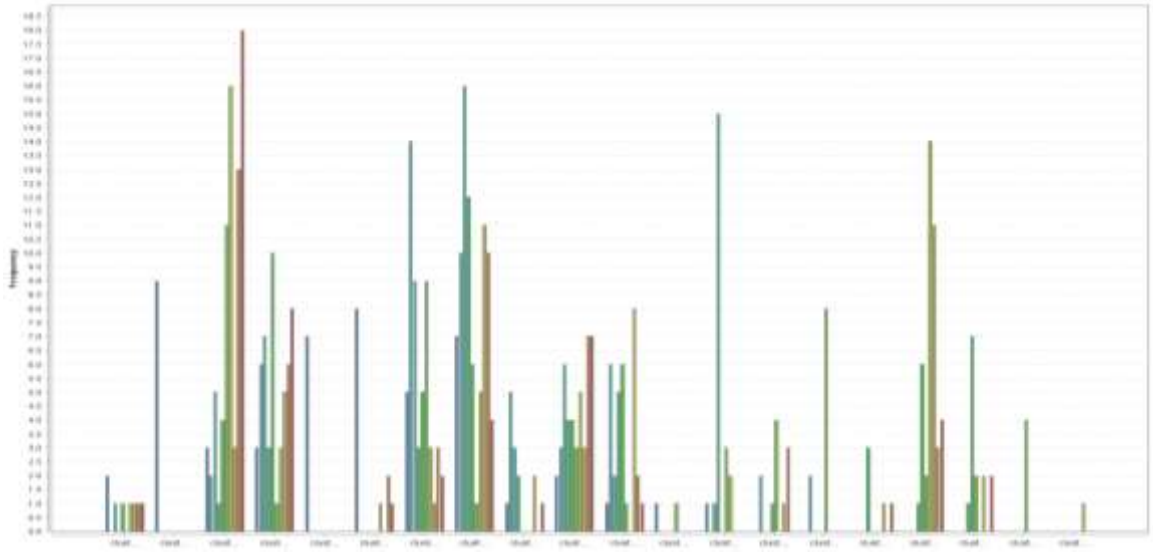
Şekil 6.10. Karma öznitelik kümesine göre verilerin 10 kümeye dağılımı

Kullandığımız veri kümesindeki yazar sayısı kadar küme üreterek verilerin kümelerdeki dağılımını karşılaştırmak için, Şekil 6.10'da olduğu gibi dokümanların kümelere paylaşımları görselleştirilmiştir. Karma öznitelik kümesinin kullanımıyla da elde edilen sonuçlar ne kümelerin tamamen yazar bazlı dağılıma sahip olduğunu ne de tamamen ortak üslup bazlı bir dağılıma sahip olduğunu göstermektedir. Birinci ve ikinci kümeler dikkate alındığında ortak bir veya birkaç model üretilebileceği sonucu çıkarılabilmekteyken, son küme dikkate alındığında daha yazar bağımlı modeller üretilebileceği sonucu çıkarılabilmektedir. Bu noktada; küme sayısını yazar sayısından yüksek tutarak belki asıl küme dağılımının şekli görüntülenebilir. Yani; bazı kümeler tamamen belli yazarların farklı üsluplarını temsil ederken bazıları belli oranlarda tüm yazarların ortak üslubunu temsil eder gibi bir çıkarım yapılabilir. Bu beklenti ile ele aldığımız veri kümesi yine belirlenen üç öznitelik kümesi kullanımı ile 20 kümeye bölünmüştür. Şekil 6.11'de noktalamalar öznitelik kümesi kullanılarak üretilen verilerin 20 kümeye dağılımı görülmektedir.



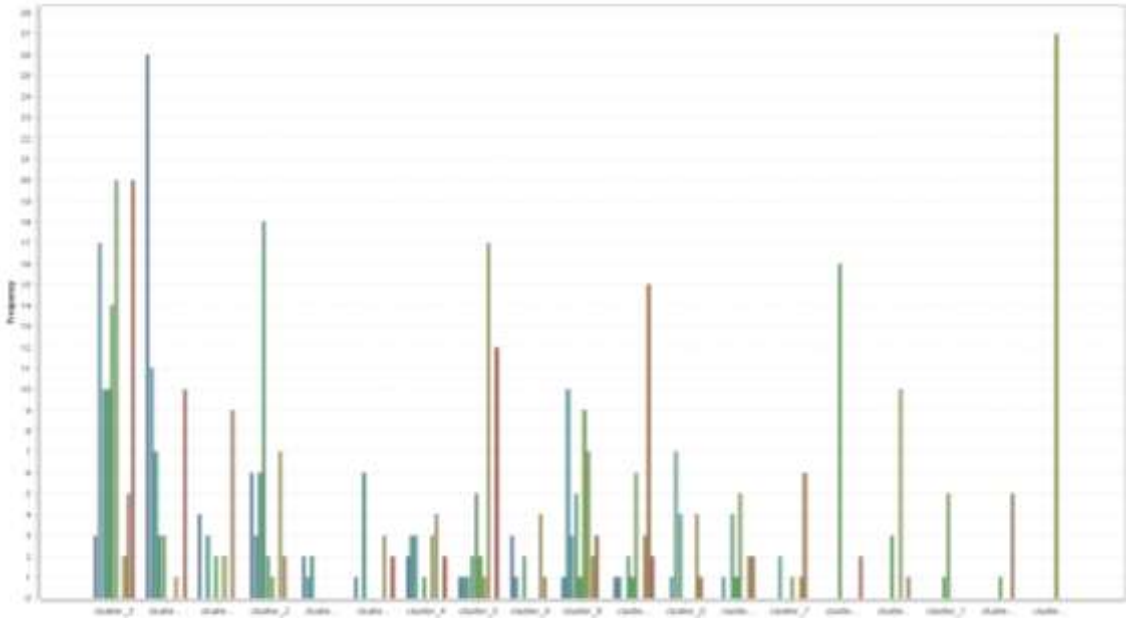
Şekil 6.11. Noktalamalar öznitelik kümesine göre verilerin 10 kümeye dağılımı

Noktalamalar öznitelik kümesi ile üretilen veriler kullanılarak on yazar verilerinin 20 kümeye dağılımı Şekil 6.11'de görüldüğü üzere, önceki noktalamalar dağılımına benzer olarak daha yazar bazlı bir kümeleme dağılımı sergilemektedir. Bu grafik özelinde daha belirli olarak bazı çıkarımlar yapılabilmektedir. Şöyle ki; yazarlar arası kullanılan ortak bir üsluptan bahsedilebilir ve bazı yazarların üslupları bu ortak üslup ile temsil edilebilir niteliktedir. Diğer taraftan yazarların büyük bir kısmı bu ortak üsluba yatkın yazılar üretmiş olsalar da yazılarının büyük bir çoğunluğu daha özel bir üslubun temsilidir. Şekil 6.12'de bow öznitelik kümesi kullanılarak üretilen verilerin 20 kümeye dağılımı görülmektedir.



Şekil 6.12. Bow öznitelik kümesine göre verilerin 20 kümeye dağılımı

Şekil 6.12’de görülen dağılım, bow öznitelik kümesi kullanılarak elde edilen, daha ortak üslubun yaygın olduğu, önceki dağılımlara benzer bir davranış sergilemektedir. Noktalamalar öznitelik kümesi ile elde edilen verilerin 20 kümeye dağılımında yapmış olduğumuz çıkarımlar kısmen burada da geçerlidir. Şöyle bir farkla ki; noktalamalar öznitelik kümesi kullanılarak üretilen verilerin dağılımı daha yazar bağımlı bir duruş sergilerken bow öznitelik kümesi kullanılarak üretilen verilerin dağılımı daha ortak üslup bağımlı bir duruş sergilemektedir. Şekil 6.13’te karma öznitelik kümesi kullanılarak üretilen verilerin 20 kümeye dağılımı görülmektedir.



Şekil 6.13. Karma öznitelik kümesine göre verilerin 20 kümeye dağılımı

Karma öznitelik kümesi kullanılarak üretilen verilerin 20 kümeye dağılımında, Şekil 6.13’te görüldüğü gibi küme temsillerinde bazı yazarlar baskın görünse de genel davranış ortak modellerin daha baskın olduğunu göstermektedir. Bu dağılımda en dikkat çekici nokta son kümenin sadece bir yazara ait dokümanlardan oluşmasıdır. Tek bir yazara ait küme sayısının 20 kümede bir olması ve bu kümenin söz konusu yazar dokümanların yaklaşık yarısını içermesi, sağlam bir çıkarım yapabilmek için yeterli değildir. Diğer taraftan önceki iki 20 kümelik dağılımlarda yapmış olduğumuz çıkarımlar nispeten bu dağılım için de geçerlidir.

6.3. Çıkarımlar

Yazar doğrulama problemine evrensel bir model önermeyi planladığımız bu tez çalışmasında, planlanan önermenin küçük çaplı bir ön deneyini uygulayarak ne kadar başarılı bir model

üretebileceğimizi nispeten gözlemleyebilmek için bu bölümde, toplanan verilere bir takım kümeleme işlemleri yapılmıştır. 10 blog yazarına ait 50'şer yazıl olmak üzere toplamda 500 dokümanın dağılımının ayrıntılı olarak ele alındığı ve yorumlandığı bu çalışmada, kullanılan verinin kendi doğasında ortak bir model barındırıp barındırmadığı gözlemlenmek istenmiştir. Noktalamalar, kelime çantası (bow) ve bu iki öznitelik kümesinin birlikte kullanımı ile elde edilen karma öznitelik kümesi kullanılarak, ele alınan dokümanların 3 farklı türde vektörel temsillerinin üretilip karşılaştırılmıştır. Her gösterim için 2, 5, 10 ve 20 olmak üzere 4 farklı küme sayısında kümeleme işlemleri yapılmış ve elde edilen kümelerin yazar etiketli doküman dağılımları görsel temsilleri ile yorumlanmıştır. Yapılan tüm değerlendirmeler sonucunda, ele alınan verinin, seçili öznitelik kümeleri doğrultusunda, doğasında ortak bir model barındırdığını söylemenin pek mümkün olmadığı görülmüştür. Şöyle ki; veri içerisinde neredeyse tüm yazarlara ait bir kısım dokümanları barındıran birkaç ortak nokta çıkarılması mümkündür fakat bu oran tüm yazarları genel anlamda temsil edebilecek bir tek modelin var olabileceği çıkarımının yapılabileceği kadar yüksek değildir. Yapılan çalışmalar doğrultusunda, daha eğitilmiş bir modelin bu ortak noktaların tespit ve yönetiminde kullanımının daha uygun olacağı çıkarımına varılmış ve sonraki çalışmalarda makine öğrenmesi çalışmalarında kullanılan sınıflama algoritmalarının kullanımının ön planda tutulması gerektiği kararı alınmıştır.

7. YAZAR TANIMLAMA İLE ANLAMLI METİN BOYU SEÇİMİ

Bu bölümde, yazar analizi çalışmalarının genel bir problemi olan anlamlı metin boyu seçimi üzerinde durulmuştur. Yapılan çalışmanın amacı yazar doğrulama problemlerinde temel adımlardan biri olacak anlamlı metin boyunun belirlenmesidir. Çalışmada yapılan deneyler ile yazar tanımlamadaki en anlamlı ayırt edici metin boyunun ne olması gerektiği bulunmaya çalışılmıştır. Bulunacak anlamlı metin boyunun yazar doğrulama çalışmalarında da başarılı sonuçlar vereceği varsayımı ile çalışmalar sürdürülmüştür. Türkçe Blog yazılarının veri kümesi olarak kullanıldığı bu çalışmada elde edilen deneysel sonuçlar ve yapılan çıkarımlar sözlü bildiri olarak literatüre eklenmiştir.

7.1. Konu Kapsam ve Literatür

İnternet kullanımının ve veri paylaşımının popülerliği göz önünde bulundurulduğunda, elektronik ortamlardaki metinsel veriler, metin analizinin farklı çeşitlerde yapılabilme potansiyelini ortaya koymuştur. Metin formunda verilerin analiz edilmesi ile yapılan bilgi çıkarma çalışmaları yıllardan beri ilgi odağı olmuştur. Bir metin oluşturulurken anlamı, söz dizimsel özellikleri, kelime bilgileri, noktalamalar, ekler ve köklerin kullanımı gibi tüm dilsel alanlar kullanılır. Tüm bu dilsel alanlar belirlenmiş kuralları çerçevesinde kullanılır. Fakat bu kuralların kullanımına ve aralarındaki bağlantılara metni yazan kişi karar verir. Bir metin bir yazar tarafından yapılmış belirli seçimlerin son ürünüdür, dolayısı ile her metin kendi yaratıcısının parmak izini taşır. Bu çıkarımdan yola çıkarak metinsel ve dilsel özniteliklerin kullanımı ile farklı bakış açılarında birçok başarılı yazar tanımlama çalışması yapılmıştır.

Bir metnin yazarı ile ilgili bilgi çıkarmak amacı ile karakteristiğini analiz eden çalışmalar Yazar Tanımlama veya Yazar Analizi adı altında ele alınmaktadır [7]. 19. Yy'dan beri istatistiksel ve hesaplamalı yöntemlerin kullanımı ile bu bilgi çıkarma çalışmaları metinsel öznitelikler üzerinden yapılmaktadır [3]. Günümüzde Yazar Analizi çalışmaları 3 ana dala yoğunlaşmaktadır; Yazar Tanımlama, Yazar Profil Çıkarımı ve Yazar Doğrulama. Yazar tanımlama temelde çok sınıflı bir sınıflandırma problemdir [55]. Sorgulanan bir doküman ve bu dokümanın yazarının da aralarında olduğu bir grup yazar ele alınır. Amaç sorgulanan dokümanın yazarının bulunabilmesidir. Bu problemin zorluğu metin analizinde, yazarlar arası en ayırt edici öznitelik kümesinin belirlenmesidir. Yazar profil çıkarımı [10] metinlerden

yazarları ile ilgili demografik veya psikolojik bilgiler çıkarabilmeyi amaçlamaktadır. Yazar doğrulama problemlerinde [3] ise amaç sorgulanan bir dokümanın şüphelenilen bir yazara ait olup olmadığının tespit edilmesidir.

Anonim olarak elektronik metinlerin kolaylıkla üretilip yayınlanabilmesi ile yeni bir tehlike doğmuştur. Anonim metinler ile kişisel hayatın ihlaline sebebiyet verebilen bu gibi tehlikelere çözüm üretebilmek için yazar tanımlama çalışmalarına olan ihtiyaç artmaktadır. Bu çalışmada tüm yazar tanımlama problemlerine yazar doğrulama bakış açısı ile temel bir çözüm önerisi sunmak amaçlanmıştır. Tüm yazar tanımlama problemleri bir dizi yazar doğrulama problemine indirgenebileceğinden [23], yazar doğrulama problemlerinin çözümü için sunulacak başarılı bir katkının yazar doğrulama problemlerinde de başarılı bir katkı olarak ele alınacağı barizdir. Bu yönü ile bu çalışmada yapılan deneylerde yazar tanımlamadaki en anlamlı ayırt edici metin boyunun ne olması gerektiği bulunmaya çalışılmıştır. Bulunacak anlamlı metin boyunun yazar doğrulama çalışmalarında da başarılı sonuçlar vereceği varsayımı ele alınmıştır.

Çalışmada incelenen literatür, ulusal yani Türkçe üzerine yapılan çalışmalar ve uluslararası yani başka diller üzerinden yapılan çalışmalar olarak iki koldan ele alınmıştır. Türkçe metinler üzerinden birçok yazar analizi çalışması yapılmıştır ve bu çalışmalar çoğunlukla yazar tanımlama üzerine yoğunlaşmıştır. Genellikle veri kümesi olarak köşe yazıları kullanılmış ve farklı öznitelik kümeleri ile farklı sınıflama algoritmaları kullanılarak başarılı sonuçlar elde edilmiştir [48]. 12.000 köşe yazısının veri kümesi olarak kullanıldığı bir doktora tezinde 1135 deney yapılarak en başarılı öznitelik kümesi olarak bazı noktalama işaretleri belirlenmiştir [43, 51]. Adli bilişim bakımından elektronik postaların yazarının belirlenmesi amacı ile elektronik postalar ile aynı karakteristiğe sahip haber grubu mesajlarını da veri kümesi olarak kullanan çalışma bulunmaktadır [52]. 5 yazar ve her yazara ait 250 mesajın veri kümesi olarak kullanıldığı söz konusu çalışmada ise 49 metinsel öznitelik ile %80 üzeri sınıflama başarısı elde edilmiştir. Yazar tanımlama çalışmaları uluslararası literatürde birçok defa farklı veri kümeleri kullanılarak yapılmıştır. Alana özgü veri yayını yapan PAN organizasyonu [9, 45, 55] bu çalışmalara olan ilgiyi arttırmıştır. Pan organizasyonu tarafından yayınlanan veri kümelerine ek olarak blog yazıları [23], elektronik postalar [17, 36], online mesajlar [16], sosyal medya yazışmaları [36], farklı tür ve konu yazıları [18, 32], uç grupların web forum mesajları [38] vb. veri kümeleri kullanılarak farklı alanlarda da yazar tanımlama sorununa çözüm sunulmuştur. Bazı yazar doğrulama problemleri de yazar tanımlama problemlerine dönüştürülerek çözülmeye çalışılmıştır. Özellikle sahtekarlık (imposter) yöntemleri ile yazar doğrulama

problemleri çok aday problemine dönüştürülerek başarılı sonuçlar elde edilmeye çalışılmıştır [29].

Bir metinden yazarı ile ilgili bilgi çıkarma çalışmalarında birçok öznitelik kümesi ve birçok yöntem kullanılmış olmasına rağmen, standart bir öznitelik kümesi ve yöntem henüz net olarak belirlenememiştir. Sürekli olarak farklı çeşitlerde verilerin elektronik ortamlarda üretiliyor olması gerçeği de bu işlemleri daha da zorlaştırmaktadır. Yazar tanımlama çalışmaları adli bilişim, bilgisayar bilimleri ve dil bilimlerinin ortak bir dalı olması sebebiyle popülerliğini uzun süre korumaktadır. Bu çalışmada yazar tanımlama çalışmaları için ele alınan metinlerin en düşük ayırt edici ve memnuniyet verici metin boyunun ne olması gerektiği sorusu üzerine odaklanılmıştır. Bu sorunun çözümü için Türkçe blog yazıları toplanmış ve bu yazılar üzerinden farklı metin boyutları ve farklı öznitelik kümeleri ele alınarak deneyler yapılmıştır.

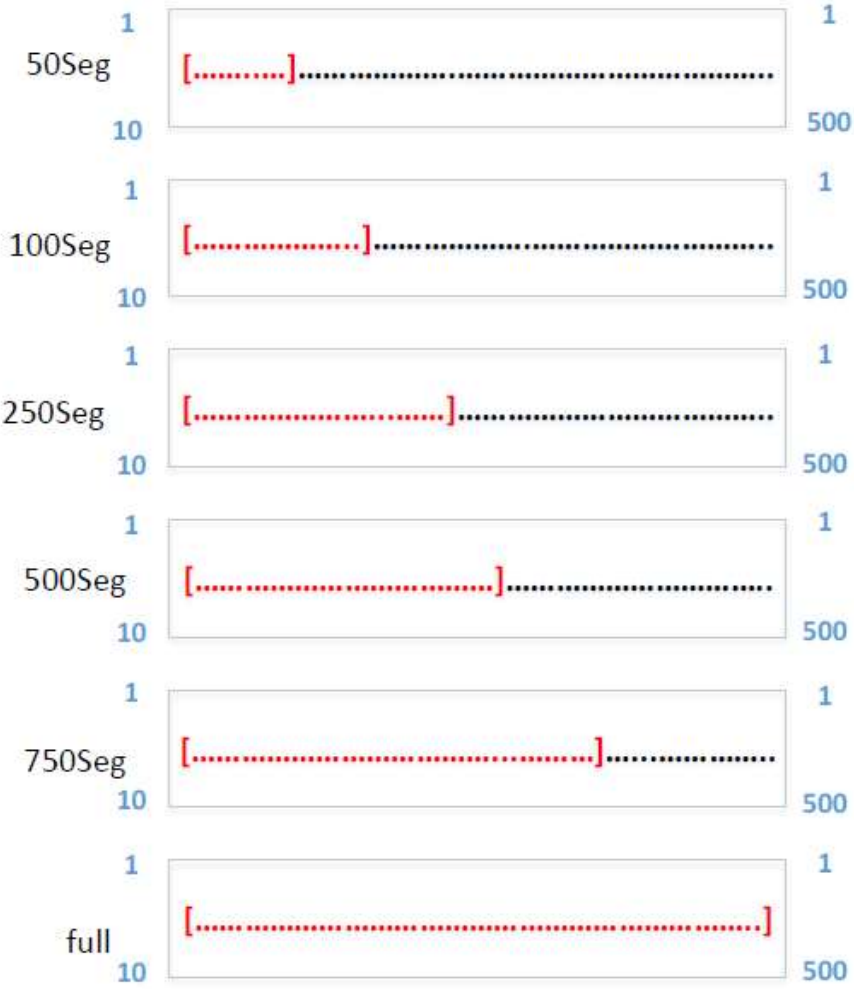
7.2. Kullanılan Veri Kümesi

Yazar doğrulama çalışmalarında sorgulanan metnin yazarı ile ilgili bir arka plan bilgisi olmadığından ele alınan metinlerin boyu problemin başarısı açısından büyük önem taşımaktadır. Ele alınan metinlerin boyutlarına göre yazar analizi çalışmalarında uygulanan yöntemler de değişmektedir. Örneğin sosyal medya mesajları veya kısa mesajların yazar analizinin yapılacağı bir çalışma ile romanların yazar analizinin yapılacağı çalışmalarda aynı uygulamaların, yöntemlerin ve özelliklerin bire bir kullanımı anlamsız olmaktadır. Bu çalışmanın amacı yazar doğrulama problemlerinde temel adımlardan biri olacak anlamlı metin boyunun belirlenmesidir. Bu amaç doğrultusunda hem farklı boylarda veriler içerebilen hem de konu bağımlı olmayan metinlerden oluşan blog yazıları bu çalışma için en uygun veri kümesi olarak tercih edilmiştir. Ayrıca blog yazıları hem yazar doğrulama çalışmalarında önemli bir veri kümesi hem de daha çok kişisel yazılar özelliği barındırdığından yazar analizi problemlerinde ele alınabilecek en gerçek dünya veri kümesi olarak, yapmış olduğumuz önceki çalışmalarda karara bağlanmıştır.. Blog yazılarının farklı boyutlarda olabilmesi, üretilecek çözümün farklı boylardaki verilerin bulunduğu veri kümelerine de uygulanabileceğini de göstermektedir. Bu ayrıcalıklar göz önünde bulundurularak Türkçe blog yazıları bu çalışmada ilk kez toplanmış ve Türkçe yazar analizi çalışmasında uygulanmıştır. Blog yazıları toplanmadan önce bazı kriterler belirlenmiştir. Aşağıda sıralı kriterlere göre blog yazıları seçilmiştir.

- Sürekli sabit bir konu üzerine yazılmamış,

- Reklam veya promosyon yazıları içermeyen,
- En az son 3 yıl içinde yazılmış
- Yazarın bir konu hakkındaki kişisel görüşlerini belirttiği yazılar

Yukarıda belirtilen kriterler doğrultusunda 10 blog yazarı ve her yazarın son 50 yazısı toplanmıştır. Toplanan veri kümesi 3 farklı grupta ele alınmıştır. 1. grup veri kümesi 10 yazardan rastgele 5 tanesinin seçilmesi ile elde edilmiş ve DS1 olarak adlandırılmıştır. Kalan 5 yazara ait yazılar DS2 adı ile ikinci grup veri kümesi olarak kullanılmıştır. 3. grup veri kümesi ise tüm yazarların bir arada değerlendirildiği veri kümesidir. Yazar odaklı yazar analizi çalışmasının tepkisi ölçülmek istendiğinden bu gruplara bölme işlemi yapılmıştır. Ayrıca bu çalışmanın amacı yazar analizi çalışmalarındaki en anlamlı ayırt edici minimum metin boyunu bulmak olduğundan, ele alınan veri kümeleri farklı metinsel büyüklüklerde ele alınmıştır. Her veri kümesi 6 farklı metin boyu ile değerlendirmeye alınmıştır. Her yazara ait 50, 100, 250, 500, 750 kelimelik segmentler (metin parçaları) ve metnin tamamı olmak üzere toplamda 6 farklı boyda metinler elde edilmiş. Metin boylarının ele alınan metinden seçilme gösterimi Şekil 7.1'de verilmiştir.



Şekil 7.1. Metinlerden belirli segmentlerde parçaların seçimi

Metin boyları, Şekil 7.1'de görüldüğü üzere, her metnin başından itibaren sayılan kelime sayısı ile belirlenmiştir. Şekilde gösterilen 1-10 değerleri yazar sayısını, 1-500 değerleri doküman sayısını gösterir ki; her segment alma işleminin tüm dokümanlara uygulandığı belirtilmek istenmiştir. Her 3 veri kümesi de bu 6 farklı boyda ele alındığından toplamda 18 veri kümesi deneylerde değerlendirilmiştir.

7.3. Materyal ve Metot

Farklı öznitelik kümeleri bu çalışmada ele alınmış ve elde edilen yazar tanımlama başarıları karşılaştırılmıştır. Türkçe üzerine yapılan yazar analizi çalışmalarında bazı noktalama işaretleri (temel noktalamalar ve ileri noktalamalar) en başarılı ayırt edici öznitelik kümesi olarak belirlendiğinden [43] bu çalışmada kullanılan ilk veri kümesi belirlenen bu noktalama işaretleri olmaktadır. İkinci öznitelik kümesi olarak bag of word (BoW) olarak adlandırılan kelime

çantası öznitelik kümesi kullanılmaktadır. Veri kümesinde bulunan her kelime lematizasyon (lemmatization) olarak adlandırılan kelime gövdeleme işleminden geçmektedir. Yani her kelime çekim eklerinden ayrılıp sadece gövde olarak bulunmaktadır. Bu veri kümesinde ele alınana öznitelikler içerisinden, İngilizcesi stop words olan ve dilimize etkisiz kelimeler olarak çevrilebilen, dillerde ayırt edicilik bakımından pek katkı sağlamadığı düşünülen kelimeler birçok çalışmada olduğu gibi bu çalışmada da bow veri kümesinden çıkarılmıştır. Üçüncü öznitelik kümesi olarak da kullanılan ilk iki öznitelik kümesinin birleşimi alınmış ve bu veri kümesi karma veri kümesi olarak adlandırılmıştır. Sonuç olarak 3 farklı veri kümesi, 6 farklı metin boyu ve 3 farklı öznitelik kullanımı ile toplamda 54 farklı özelliklerde veri kümesi bulunmaktadır. Bu veri kümeleri, içerdikleri segment boyu ve öznitelikleri Tablo 7.1'de gösterilmektedir.

Tablo 7.1. Anlamlı doküman boyu belirlemede kullanılan veri kümeleri ve özellikleri

Veri Kümesi	Bow Öznitelik kümesi ile	Noktalamalar Öznitelik kümesi ile	Karma Öznitelik kümesi ile
DB1	DB1_Bow_50Seg	DB1_Punct_50Seg	DB1_Karma_50Seg
	DB1_Bow_100Seg	DB1_Punct_100Seg	DB1_Karma_100Seg
	DB1_Bow_250Seg	DB1_Punct_250Seg	DB1_Karma_250Seg
	DB1_Bow_500Seg	DB1_Punct_500Seg	DB1_Karma_500Seg
	DB1_Bow_750Seg	DB1_Punct_750Seg	DB1_Karma_750Seg
	DB1_Bow_full	DB1_Punct_full	DB1_Karma_full
DB2	DB2_Bow_50Seg	DB2_Punct_50Seg	DB2_Karma_50Seg
	DB2_Bow_100Seg	DB2_Punct_100Seg	DB2_Karma_100Seg
	DB2_Bow_250Seg	DB2_Punct_250Seg	DB2_Karma_250Seg
	DB2_Bow_500Seg	DB2_Punct_500Seg	DB2_Karma_500Seg
	DB2_Bow_750Seg	DB2_Punct_750Seg	DB2_Karma_750Seg
	DB2_Bow_full	DB2_Punct_full	DB2_Karma_full
DB3	DB3_Bow_50Seg	DB3_Punct_50Seg	DB3_Karma_50Seg
	DB3_Bow_100Seg	DB3_Punct_100Seg	DB3_Karma_100Seg
	DB3_Bow_250Seg	DB3_Punct_250Seg	DB3_Karma_250Seg
	DB3_Bow_500Seg	DB3_Punct_500Seg	DB3_Karma_500Seg
	DB3_Bow_750Seg	DB3_Punct_750Seg	DB3_Karma_750Seg
	DB3_Bow_full	DB3_Punct_full	DB3_Karma_full

Yazar tanımlama çalışmalarında belirlenecek başarılı minimum metin boyunun yazar doğrulama çalışmalarında da başarılı sonuçlar vereceği varsayımından yola çıkarak, bu çalışmada yazar tanımlama uygulamalarında anlamlı sonuç verecek en kısa metin boyu tespit edilmeye çalışılmıştır. Bu amaç doğrultusunda iki önemli makine öğrenimi algoritması kullanılmıştır. Bu algoritmalar ve özellikleri aşağıdaki bölümlerde kısaca belirtilmiştir.

7.3.1. Destek Vektör Makinaları (Support Vector Machines – SVM)

Makine öğrenimi algoritmaları arasında güçlü teorik alt yapısı ve ürettiği güvenilir sonuçlar ile SVM algoritması son yılların yaygın olarak kullanılan algoritması olarak ele alınmaktadır [59, 60]. Birçok yazar analizi çalışmasında da başarılı sonuçlar üretmesi sebebiyle SVM algoritması

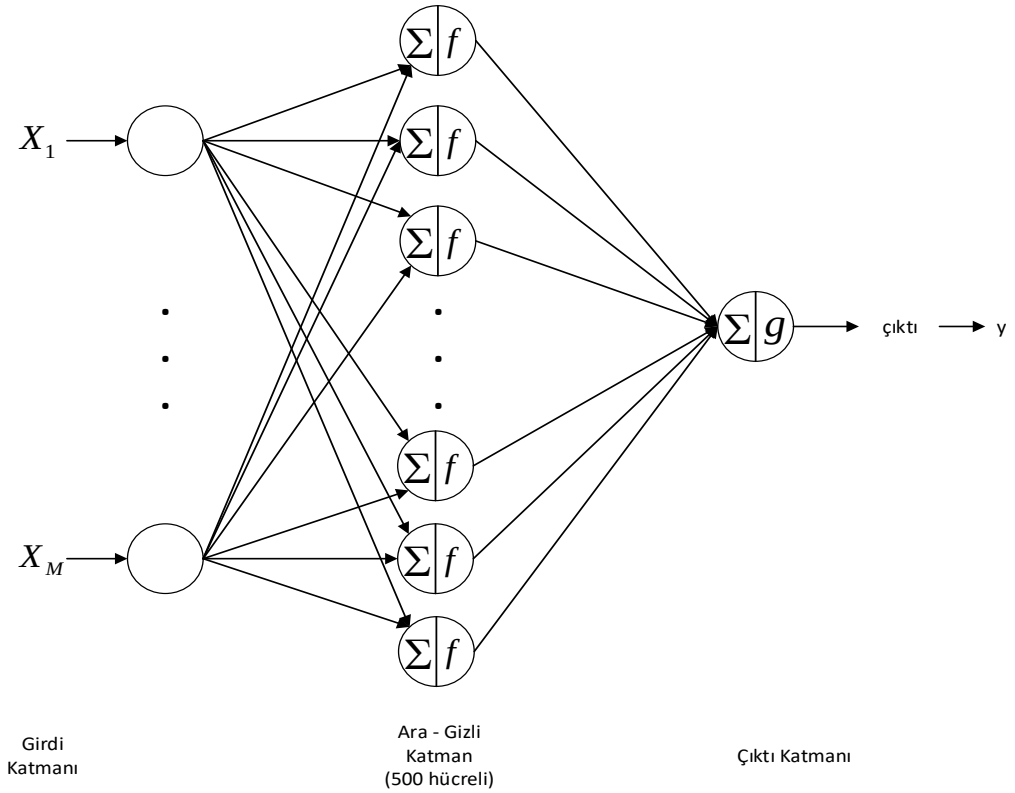
bu alanda da önemini korumakta ve popülerliğini sürdürmektedir [23, 61, 62]. Özellikle verilerin doğrusal olarak ayrımlarını belirlemede kullanılan algoritma hiper düzlemi en uygun bölecek doğruyu bulmaya çalışır. Doğrusal olarak ayrılabilen verilerin en uygun ayırt edici düzlemini bulabilmek amacı ile ilk zamanlarda geliştirilmiş olsa da farklı çekirdek fonksiyonlarının kullanımı ile aralarında doğrusal ilişkili olmayan veriler üzerinde de başarıyla kullanılabilmektedir. Özellikle RBF çekirdek fonksiyonu ile yazar analizi çalışmalarında çok sınıflı sınıflandırıcı olarak SVM ile sınıf ayrımları elde edilebilmektedir. RBF çekirdek fonksiyonu Denklem (12)'de gösterildiği gibi hesaplanmaktadır.

$$RBF_k\left(x_i, x_j\right) = \exp\left(-\gamma . \left|\left|x - y\right|\right|^2\right) \tag{12}$$

Bu çalışmada SVM algoritması RBF çekirdek fonksiyonu ile yazar tanımlama amacıyla sınıflama başarısını elde etmek için kullanılmaktadır. SVM çeşidi olarak libSVM ile SVC (Spport Vector Classifier) yöntemi kullanılmıştır. Girdi parametreleri gamma = 0.0, c =0.0, cahce size = 80 ve epilson = 0,001 olarak verilmiştir. Her deneyde 10 katlı- çapraz doğrulama kullanılmıştır.

7.3.2. Yapay Sinir Ağları (Artificial Neural Network - ANN)

Yapay sinir ağları, yapay zeka çalışmalarının temelini oluşturan, insan beyninin çalışma prensibine dayalı bilgisayar programlarıdır [63]. İçerisinde birçok farklı tasarım barındıran bu hesaplamalı yaklaşımlar veri madenciliğinden, sınıflamaya, örüntü tanımadan tahmin sistemlerine, doğal dil işlemeye kadar birçok alanda başarıyla kullanılmaktadır [64]. Bu çalışmada yapay sinir ağları sınıflandırıcı olarak yazar tanımlama amacı ile kullanılmaktadır. Girdi olarak verilen öznitelikler 500 sinir hücreli ara katmandan geçerek karar sinir hücresine iletilmektedir. Ara katmanda farklı sayılarda sinir hücreleri kullanılmış ve en uygun sayı hem performans hem de başarı olarak en iyi sonucu verdiğinden 500 olarak seçilmiştir. Girdi parametreleri training cycle = 100, learning rate = 0.3 olarak verilmiştir. Seçilen parametrelere göre temsili bir yapay sinir ağı Şekil 7.2'de gösterilmektedir.



Şekil 7.2. Yapay sinir ağları uygulamasının şekilsel gösterimi

7.3.3. Bilgi Kazanımı (Information Gain)

Seçili öznitelikler bakımından sınıflandırma performanslarının anlamlı olabilmesi için özellikle bow öznitelik kümesinin boyunun azaltılması gerekmektedir. Bu amaç doğrultusunda Bilgi Kazanımı (Information Gain) öznitelik seçme algoritması kullanılmıştır [65]. Denklem (13)'te verilen formül ile, Bilgi kazanımı algoritmasında her özniteliğin bilgi kazanım metriğine göre değeri hesaplanmıştır. Verilen formülde t_k değişkeni k. özniteliği, c_i değişkeni ise i. sınıfı temsil etmektedir.

$$BK(t_k, c_i) = \sum_{c_i \in c} \sum_{t_i \in t} P(t,c).\log \frac{P(t,c)}{P(t).P(c)} \qquad (13)$$

Bilgi kazanımı öznitelik seçme algoritmasının kullanımı ile SVM sınıflandırıcılarında bow öznitelik setinin veri sayısının yarısı kadar en yüksek ağırlığa sahip öznitelik seçilmiş ve hem bow hem karma öznitelik kümelerinde bu seçili özniteliklerin frekansları kullanılmıştır. Yapay sinir ağlarının (ANN) sınıflandırıcı olarak kullanıldığı deneylerde ise yapılan ön deneyler ile bilgi kazanımı algoritması kullanılarak seçili en yüksek değere sahip 1000 özniteliğin kullanımında en anlamlı sonuçlar elde edilmiştir. Dolayısı ile ANN kullanılan deneylerde hem

bow öznitelik setinde hem de karma öznitelik setinde en yüksek ağırlığa sahip 1000 öznitelik kullanılmaktadır.

7.4. Sonuçlar

Ele alınan DS1, DS2 ve DS3 veri kümeleri, 50, 100, 250, 500, 750 ve tam metin olmak üzere 6 farklı metin boyunda, noktalamalar, bow ve karma olmak üzere 3 farklı öznitelik kümesi ve SVM ile ANN olmak üzere 2 farklı sınıflandırma algoritması kullanılarak yazar tanımlama çalışmasında kullanılmıştır. Toplamda 108 farklı kombinasyonlar ile deneyler yapılmıştır.

Faklı kombinasyonlar sonucu elde edilen veri kümelerinin sınıflama performansları ilk olarak SVM ile hesaplanmıştır. Çalışmanın kapsamında da belirtildiği gibi elde edilen sınıflama performanslarının her sınıf bazında dikkatlice ele alınması amacı ile sınıflandırma sonuçları; en iyi sınıf performansı, en kötü sınıf performansı ve ortalama performans olarak her deney 3 farklı sonuç ile değerlendirilmiştir. SVM algoritması kullanılarak elde edilen deneysel sonuçlar Tablo 7.2'de verilmiştir.

Tablo 7.2. SVM ile farklı metin boylarından elde edilen sınıflamaların doğruluk sonuçları

		SVM								
		Noktalama			Bow			Karma		
Metin boyu	Veri kümesi	maks	min	ort	maks	min	ort	maks	min	ort
50	DS1	68,00	24,00	43,60	84,00	0,00	37,20	82,00	20,00	37,60
100	DS1	72,00	30,00	49,20	94,00	2,00	38,00	88,00	16,00	40,80
250	DS1	72,00	42,00	56,00	88,00	0,00	44,80	84,00	22,00	43,60
500	DS1	62,00	42,00	54,40	82,00	0,00	46,40	98,00	14,00	34,00
750	DS1	62,00	44,00	54,00	82,00	0,00	47,20	92,00	12,00	32,00
Tam metin	DS1	62,00	40,00	53,60	50,00	26,00	38,00	50,00	26,00	38,00
50	DS2	86,00	12,00	47,20	74,00	16,00	36,80	78,00	8,00	38,00
100	DS2	82,00	38,00	63,20	80,00	30,00	53,20	76,00	20,00	48,80
250	DS2	88,00	52,00	71,20	92,00	30,00	55,60	90,00	24,00	52,40
500	DS2	88,00	68,00	75,60	96,00	30,00	62,40	98,00	8,00	51,20
750	DS2	86,00	60,00	75,20	94,00	30,00	65,20	90,00	10,00	49,60
Tam metin	DS2	86,00	66,00	74,80	60,00	24,00	39,20	60,00	24,00	39,20
50	DS3	82,00	8,00	28,40	78,00	2,00	26,40	58,00	6,00	30,80
100	DS3	80,00	10,00	36,80	62,00	2,00	34,40	62,00	10,00	32,80
250	DS3	80,00	22,00	47,20	86,00	0,00	45,00	96,00	4,00	30,20
500	DS3	88,00	20,00	48,80	92,00	0,00	46,20	98,00	4,00	29,20
750	DS3	88,00	10,00	47,80	94,00	0,00	46,60	76,00	2,00	27,40
Tam metin	DS3	86,00	10,00	47,60	48,00	0,00	24,60	48,00	0,00	24,60

SVM sınıflandırma ile elde edilen deneysel sonuçlar doğrultusunda DS2 veri kümesi ile DS1 ve DS2 veri kümesinden genel olarak daha başarılı sonuçlar elde edilmiştir. Ortalama sınıflandırma başarısı bakımından bakacak olursak, noktalamalar öznitelik kümesi bow ve karma öznitelik kümelerinden daha yüksek doğruluk sonucunu üretmiştir. Fakat bireysel sınıflama başarıları göz önüne alındığında bow ve karma öznitelik kümeleri ile daha yüksek doğruluklar elde edilmiştir. Yine bireysel sınıflandırma başarısı açısından bakıldığında en yüksek sınıflandırma başarısı 500 kelime boyutlu metinlerde karma öznitelik kümesinin kullanımı ile %98 olarak elde edilmiştir. Ortalama sınıflandırma başarısı olarak en iyi sonuç DS2 veri kümesinde 500 ve üzeri kelime boyutlarında noktalamalar öznitelik kümesi ile elde edilmiştir.

Yazar tanımlama çalışması olarak sınıflandırıcı performansını çalışmanın etki alanında tutmamak adına SVM algoritması kullanılarak yapılan tüm deneyler aynı özelliklerde bir de yapay sinir ağları (ANN) kullanılarak tekrarlanmıştır. Noktalamalar öznitelik kümesinin doğrudan girdi katmanına verildiği ANN deneylerinde, bow öznitelik kümesi için en yüksek ağırlığa sahip 1000 öznitelik girdi katmanına verilmişken aynı 1000 özniteliğe ek olarak noktalamalar da karma öznitelik kümesi olarak girdi katmanına verilmiştir. Tüm ANN deneylerinde ara katmanda 500 sinir hücresi bulunmaktadır. SVM algoritmasının kullanımında olduğu gibi ANN kullanılan deneylerde de bireysel sınıf başarılarının değerlendirileceğinden, her deneyde en yüksek ve en düşük sınıf performansı ve veri kümesindeki tüm sınıfların ortalama performansları değerlendirilmiştir. ANN sınıflandırma algoritması kullanılarak ile elde edilen sonuçlar Tablo 7.3'te gösterilmektedir.

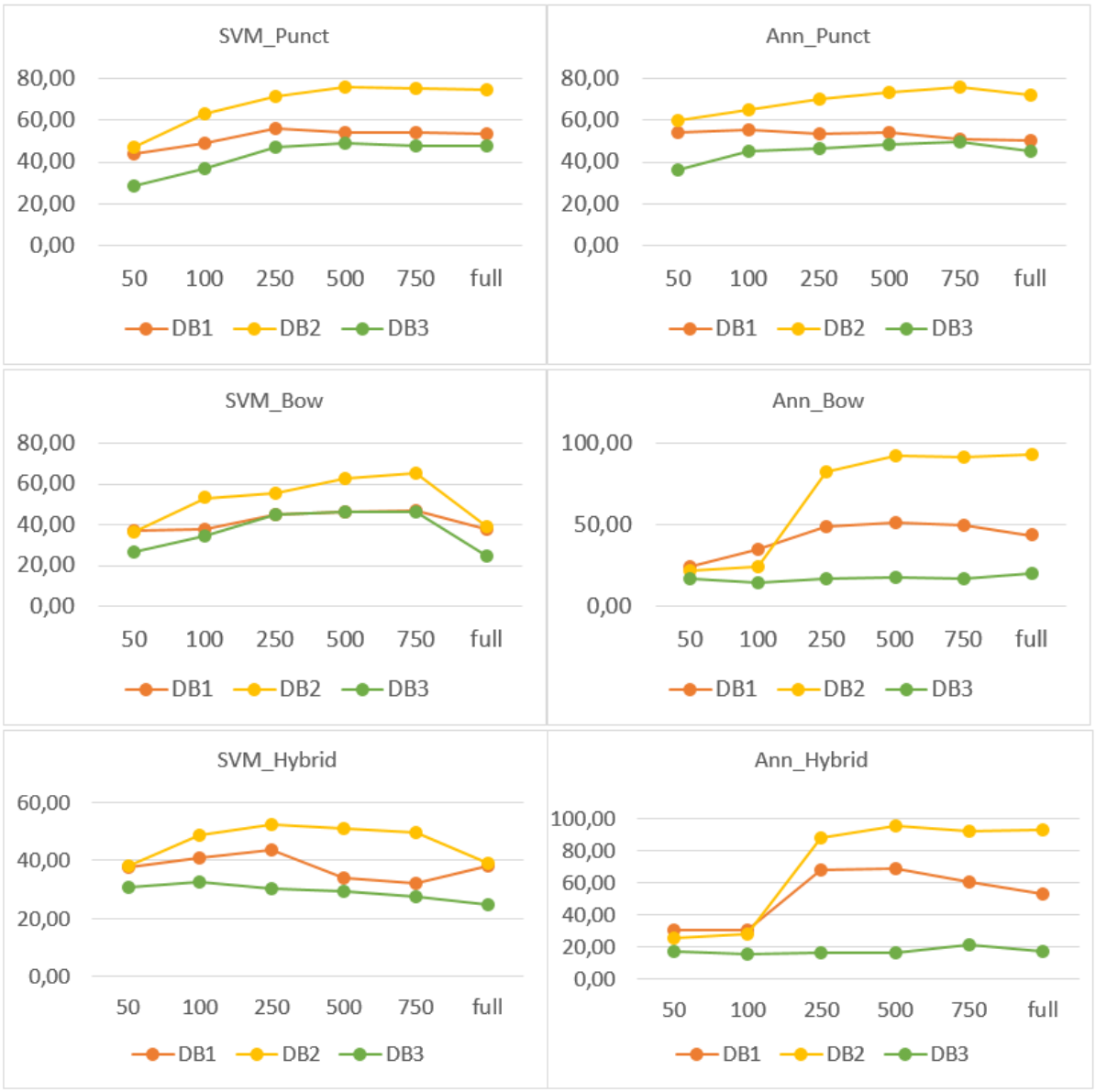
Tablo 7.3. ANN ile farklı metin boylarından elde edilen sınıflamaların doğruluk sonuçları

		ANN								
		Noktalama			**Bow**			**Karma**		
Metin boyu	**Veri kümesi**	**maks**	**min**	**ort**	**maks**	**min**	**ort**	**maks**	**min**	**ort**
50	DS1	62,00	40,00	54,40	68,00	0,00	24,40	60,00	12,00	30,80
100	DS1	68,00	44,00	55,60	60,00	10,00	34,80	58,00	12,00	30,80
250	DS1	80,00	30,00	53,20	76,00	24,00	49,20	76,00	56,00	67,60
500	DS1	66,00	28,00	54,00	80,00	38,00	51,20	84,00	60,00	68,80
750	DS1	68,00	28,00	51,20	82,00	28,00	50,00	78,00	54,00	60,40
Tam metin	DS1	66,00	26,00	50,40	72,00	16,00	43,60	68,00	36,00	52,80
50	DS2	74,00	46,00	59,60	56,00	0,00	22,00	52,00	0,00	25,60
100	DS2	78,00	42,00	64,80	68,00	8,00	24,00	42,00	18,00	28,40
250	DS2	84,00	56,00	70,40	86,00	78,00	82,40	94,00	82,00	88,00
500	DS2	94,00	56,00	73,60	96,00	78,00	92,00	100,00	84,00	95,60
750	DS2	94,00	62,00	75,60	98,00	78,00	91,60	100,00	80,00	92,40
Tam metin	DS2	92,00	60,00	72,00	98,00	82,00	93,20	100,00	82,00	92,80
50	DS3	64,00	10,00	36,20	50,00	2,00	16,80	60,00	0,00	17,40
100	DS3	80,00	22,00	45,00	56,00	0,00	14,80	68,00	0,00	15,80
250	DS3	80,00	14,00	46,20	58,00	0,00	17,40	80,00	0,00	16,60
500	DS3	90,00	18,00	48,60	74,00	0,00	17,60	72,00	0,00	16,60
750	DS3	92,00	16,00	49,80	52,00	0,00	17,00	78,00	4,00	21,20
Tam metin	DS3	94,00	16,00	45,20	64,00	8,00	20,60	44,00	0,00	17,20

Deneyler üzerinden yapay sinir ağlarının kullanımı ile yapılan sınıflandırma sonuçları dikkate alındığında DS2 veri kümesi ile genel olarak DS1 ve DS3 veri kümelerinden daha yüksek doğrulukta sonuçlar elde edilmiştir. Ortalama sınıflama başarısı dikkate alındığında karma öznitelik kümesi ile bow ve noktalamalar öznitelik kümelerinde 250 ve üzeri metin boylarında daha yüksek doğrulukta sonuçlar elde edilmiştir. Fakat bireysel sınıflama başarıları açısından bow ve karma öznitelik kümeleri daha iyi sonuçlar üretmektedir. DS3 veri kümesinde ise noktalamalar öznitelik kümesi ile bow ve karma öznitelik kümelerinden daha yüksek başarı elde edilmiştir. Genel olarak ANN ile yapılan deneyler göz önünde bulundurulduğunda 250

kelime ve üzeri metin boylarında bow ve karma öznitelik kümeleri kullanılarak en iyi sonuçlar elde edilmiştir. En yüksek doğruluk oranları ise DS2 veri kümesinde 500 kelime boyundaki metinlerin sınıflandırılmasında karma öznitelik kümesi ile elde edilmiştir.

SVM sınıflama algoritması ve ANN sınıflama yapısı kullanılarak, veri kümesinden elde edilen sınıflama işlemlerinin doğruluk sonuçlarının grafiksel gösterimi, sonuçların daha anlaşılır olması bakımından, karşılaştırmalı olarak Şekil 7.3'te verilmektedir.



Şekil 7.3. Kullanılan sınıflama algoritmalarından elde edilen karşılaştırmalı sonuçlar

7.5. Tartışma

Bu bölümde, yazar doğrulama çalışmalarında ele alınması gereken en düşük doküman boyunun ne olması gerektiği sorusunun cevabı için bir takım deneyler yapılmıştır. Bu deneyler yazar tanımlama problemi üzerinden yürütülmüştür. Bu durumun sebebi öncelikle bir yazarı temsil edebilecek en anlamlı metin boyunun bulunması gerekliliğidir. Çalışmanın başlangıcında ele aldığımız varsayım doğrultusunda; bir yazarın üslubunu barındırabilen, bu üslubu temsil edebilen en kısa metin parçası, yazar doğrulama çalışmalarında da ele alınabilecek en kısa metin parçası olmalıdır.

Farklı metin boylarının değerlendirildiği yazar tanımlama deneylerine göre, yazar tanımlama ve yazar doğrulama çalışmalarında dikkate alınması gereken en düşük ve anlamlı metin boyu 500 kelime olarak görünmektedir. 250 kelimelik metinlerden de elde edilen başarılar nispeten iyi olmasına rağmen 500 kelimelik metinler kadar güvenilir sonuçlar üretmemiştir. Diğer taraftan 750 kelimelik metinlerden de elde edilen başarılar birçok deneyde 500 kelimelik yapılardan daha iyi sonuçlar üretmesine rağmen, amacımız en düşük anlamlı boyutu belirlemek olduğundan 500 kelimelik yapıların yeterli olacağı kararlaştırılmıştır. Bir başka önemli çıkarım ise yapılan deneyler sonucunda görülmektedir ki bazı veri kümeleri ve öznitelik kümelerinin kullanımında metin boyunun artması elde edilen başarının düşmesine sebep olabilmektedir. Bu sonuç doğrultusunda nispeten daha uzun metinlerin ayrıca incelenmesi gerekliliği ortaya çıkmaktadır. Elde edilen sonuçlar doğrultusunda görülmektedir ki çok aşırı olmamak şartı ile metin boyu arttıkça noktalamalar, bow ve karma öznitelik kümelerinin kullanımında elde edilen başarı da artmaktadır. Bunun yanında yapılan deneyler göstermektedir ki 500 kelime boyunu aştıktan sonra elde edilen başarı metin boyunun artması ile çok büyük farklar oluşturmamaktadır. Yapılan deneylerde ayrıca görülmektedir ki ele alınan öznitelik kümesinin başarısı da en az metin boyunun başarısı kadar önemlidir. Bu çalışma ile elde edilen sonuçlar ışığında ileriki çalışmalarda belirlenen anlamlı metin boyu üzerinde yazar doğrulama yapılması planlanmaktadır. Yazar tanımlama ve yazar doğrulama çalışmaları için anlamlı minimum metin boyunun belirlenmeye çalışıldığı bu çalışma Uluslararası Dijital Adli Bilimler ve Güvenlik Sempozyumu'nda, Türkçe Metinler Üzerinde Yazar Tanımlama için Anlamlı Metin Boyunun Kararı adı ile sözlü bildiri olarak sunulmuştur [66].

8. ÖZNİTELİK SEÇİMİ VE VERİ KÜMESİ DENGELEME

Bu bölümde, yazar analizi çalışmalarında elde edilecek başarılı sonuçları etkileyen, bir veri kümesinde gözlemlenmesi gereken en önemli parametrelerden öznitelik seçimi ve her yazara ait veri boyutunun etkileri üzerine çalışmalar yapılmıştır. Ele alınan veri kümelerinin dengesiz oluşu yapılan çalışmaların güvenilirliğini olumsuz bakımdan etkilemektedir. Bu çalışmada, veri kümesinin yazarlara ait veriler bakımından standartlaştırılması ve bu işlemlerin yazar tanımlamadaki etkisi ele alınmıştır. Kullanılan veri kümesi hem doğal hali ile hem de standartlaştırılmış hali ile yazar tanımlama işleminde kullanılmış, avantaj ve dezavantajları elde edilen sonuçlar üzerinden değerlendirilmiştir.

8.1. Konu Kapsam ve Literatür

Metinler yazar/yazarlarının davranışsal parmak izini taşıyan en önemli verilerden biridir. Günümüz teknolojilerini kullanarak birçok çalışma bu parmak izlerinin karakteristik özelliklerini çıkararak söz konusu metnin yazarını tespit etmeyi amaçlamıştır. Yazarı bilinmeyen bir dokümanın karakteristiğinin analiz edilmesi ile o dokümanın yazarı ile ilgili bilgi çıkarma işlemleri Yazar Niteleme (Authorship Attribution) [10] veya Yazar Analizi (Authorship Analysis) [7] adı altında ele alınmaktadır. 19. yy'den beri istatistiksel ve hesaplamalı yöntemlerin kullanıldığı bu çalışmalar farklı türlerde ve farklı kişiler tarafından sürekli olarak metinsel veriler üretilmesiyle günümüzde hala popülerliğini korumaktadır. Yazar niteleme çalışmaları temelde 3 ana kola ayrılmaktadır [10]. Bu alanda yaygın olarak ele alınan çalışmalar Yazar Tanımlama (Authorship Identification) çalışmalarıdır. Yazar tanımlamada genellikle bir grup aday yazara ait birçok doküman işlenerek o yazarlara ait sınıflar elde edilmeye çalışılır. Daha sonra yazarı bilinmeyen bir dokümanın bu aday yazarlardan hangisine ait olduğu sorgulanır. Yazar Tanımlama çalışmalarında sorgulanan doküman yazarının aday yazarlardan biri olduğu garantilenir, bu sebeple Yazar Tanımlama çalışmaları bir kapalı küme (closed set) problemi olarak ele alınmaktadır. Yazar Profil Çıkarımı (Author Profiling) [46, 47, 67] yazar niteleme çalışmalarının ana kollarından biri olup, bir yazarın dokümanları üzerinden o yazara ait yaş, cinsiyet, psikolojik durum gibi sosyo-demografik bilgilerin çıkarıldığı çalışmalardır. Yazar Doğrulama (Authorship Verification), yazar niteleme çalışmalarının en zorlu koludur, yazarı bilinmeyen bir dokümanın ele alınan bir yazara ait olup olmadığı sorgulanır [23]. Yazar Doğrulama probleminde sorgulanan dokümanın yazarı ile ilgili arka plan bilgisi yoktur ve ele alınan sadece 1 yazar olduğundan bu problem hem tek sınıflı bir

sınıflandırma problemi hem de açık küme (open set) problemi olarak ele alınmaktadır. Yazar doğrulama çalışmaları temelde, yazarı bilinen bir doküman ile yazarı bilinmeyen bir dokümanı karşılaştırıp, iki dokümanın aynı yazar tarafından yazılıp yazılmadığı bilgisine ulaşmaya çalışır. Çalışmaların zorluğu kullanılan veri setinin sınırlı olmasından da kaynaklanmaktadır [50].

Metin formundaki verilerin analiz edilmesiyle metinlerden yazarı ile ilgili bilgi çıkarmayı amaçlayan yazar niteleme çalışmaları son yıllarda yaygın olarak yürütülmektedir. Bu alanda ele alınan veri kümeleri standart bir yapıda olmayıp farklı yazarlardan farklı boyut ve sayılardaki metinler ile niteleme işlemleri yapılmaktadır. Bu durumda üretilen modeller, verisi daha fazla olan yazar tipini daha iyi niteleyebilme eğiliminde olmaktadır. Bu çalışmada yazar niteleme çalışmaları sonucu üretilecek modellerin güvenilirliğini arttırmak için yazar-doküman uzayı standartlaştırılarak yazar niteleme yapılmıştır. Ele alınan veri kümesi hem doğal hali ile hem de standartlaştırılmış hali ile yazar tanımlama işleminde kullanılmış, avantaj ve dez avantajları elde edilen sonuçlar üzerinden değerlendirilmiştir. Her yazarın aynı oranda eğitime katılması ile elde edilen yazar tanımlama sonuçlarının başarısının alana anlamlı bir bakış açısı katması beklenmektedir. Yazar niteleme problemlerinin çözümüne yönelik veri kümesi normalizasyonu için nasıl bir yol izlenmesi gerektiği, bu çalışmada deneysel çalışmalar ile gösterilmektedir. Deneysel çalışmalar Türkçe metinler üzerinde gerçekleştirilmiş, her aşamada makine öğrenmesi yöntemleri kullanılarak yazar niteleme yapılmış ve elde edilen sonuçlar değerlendirilmiştir.

Yazar niteleme çalışmalarında iki temel parametre elde edilen sonuçların başarısını belirlemekte önemli rol oynamaktadır, bunlar; veri kümesindeki yazar sayısı ve her yazar için ele alınan verinin boyutudur. Yapılan çalışmalarda yazar sayısının artması veya yazarlara ait veri boyutunun azalması elde edilecek başarının düşük çıkmasına sebep olduğu görülmüştür [68]. Bu çalışmada yazar niteleme çalışmalarında kullanılacak veri kümesinin bir ön işlem olarak nasıl normalize edilmesi gerektiği ve hangi kıstasları sağlaması gerektiği üzerine çalışmalar yapılmıştır. Hem yazar - doküman uzayının hem de ele alınan dokümanların boyutunun nasıl normalize edilmesi gerektiği yapılan deneyler ile gösterilmiştir. Her veri kümesi için farklı algoritmalar farklı sonuçlar üretecek olsa da önerilen adımlar her veri kümesi için standart olması beklenmektedir.

Yazar niteleme çalışmaları, doküman yazar ataması içeren tüm problemleri kapsar. Bu sebeple bu alanda farklı veri kümeleri kullanılarak farklı problemler farklı algoritmalar ile çokça ele alınmaktadır. Tarihi 19.yy'a dayanan bu çalışmalar, temelde yazarların üslup veya stilistik

özelliklerinin dokümanlardan çıkarılarak sayısallaştırılması ile gerçekleştirilmektedir. 2009 ve 2014 yılında Yazar Niteleme çalışmaları ile ilgili yapılan araştırma makaleleri [3, 7], hesaplamalı yöntemlerin ele alındığı çalışma [10], literatürde var olan problemleri ve kullanılan yöntemleri tanıtmaktadır. Kullanılan veri kümesine göre yapılan çalışmalar çeşitlenmektedir. Aykırı grupların tespit edilmesi amacıyla web forumlardaki yazışmalara uygulanan yazar analizi çalışmasında [38], İngilizce ve Arapça veri kümelerinden birçok farklı öznitelik kümesi çıkarılarak uygulamaya katılmış ve elde edilen sonuçlar karşılaştırmalı olarak değerlendirilmiştir. Türkçe metinler üzerine yapılan çok oyunculu oyun platformlarında birden fazla kimlik kullananların tespitinin yapıldığı çalışmalarda [53, 54] oyun ortamlarındaki sohbet yazışmaları kullanılarak anlamlı sonuçlar elde edilmiştir. Sosyal medya ortamlarındaki anonim yazarların adli bilişim süreçlerinde tespiti için yapılan yazar niteleme çalışması [69], dokümandan yazar cinsiyetinin tespiti için yapılan çalışma [70] ve elektronik postaların yazar tespiti için yapılan çalışma [71] ile yazar niteleme alanındaki problem çeşitliliği gözler önüne serilmektedir. Yazarların üslup veya stilistik özelliklerinin sayısallaştırılması ile ele alınan yazar niteleme çalışmaları genel kapsamı ile birçok araştırmanın konusu olmuştur [5]. Türkçe metinler üzerinde sayısal üslup araştırmalarını inceleyen çalışma [6], orijinali Türkçe olan bir romanın çevirilerinin aslına olan sadakatini de ölçmektedir.

Genellikle yapılan çalışmalar dokümanların yapısını bozmadan yazar-doküman uzayına evirip çıkarılan özniteliklerin indirgenmesi ve veri madenciliği algoritmaları ile bilgi çıkarımı üzerinedir. Böylesi durumlarda çok sayıda ve nispeten daha uzun yazılar içeren dokümanların yazarına ait elde edilecek bilgi çıkarımı daha fazla olacaktır. Örneğin yazar doğrulama çalışmalarında ele alınan doküman sayısının az veya dokümanların içeriğinin kısa oluşu bu çalışmaların güvenilirliğini sarsmaktadır. Bu sebeple yazar niteleme çalışmalarında dokümanların boyutunun belirli sınırlarda olmasını gerektirmektedir. Yazar doğrulamada ele alınacak doküman boyutunun sahip olması gereken alt sınırı Türkçe metinler için belirleyen çalışma [66] ve İngilizce için yazarlara ait veri boyutunun güvenilir minimum değerinin 10.000 kelime olması gerektiğini belirten çalışma [72], yazar niteleme alanında ele alınan veri kümesinin belirli kıstasları sağlaması gerekliliğini göstermektedir. Söz konusu gereklilik göz önünde bulundurularak bu çalışmada Türkçe yazar niteleme çalışmalarındaki veri kümesi normalizasyonunun nasıl yapılması gerektiği deneysel çalışmalar ile her aşamada yazar tanımlama yapılarak elde edilen sonuçların değerlendirilmesi ile gösterilmiştir.

8.2. Kullanılan Veri Kümesi

Yazar niteleme alanında yapılan ulusal çalışmalarda çoğunlukla, kolayca ve fazlaca elde edilebildiği için köşe yazıları kullanılmaktadır. Köşe yazıları her ne kadar yazar nitelemede kullanılsa da içerik bakımından konu bağımlı yazılardır. Farklı dillerde yapılan yazar niteleme çalışmalarında farklı veri kümeleri yayınlanmaktadır, fakat Türkçe dilinde yazar niteleme çalışmalarında kullanılabilecek iyi tanımlanmış bir veri kümesi bulunmamaktadır. Bu eksiklik doğrultusunda, bu çalışmanın başlangıcında iyi tanımlı bir veri kümesi toplanmıştır. Güncel blog yazılarından toplanan bu veri kümesinin özellikleri Tablo 8.1'de gösterilmektedir.

Tablo 8.1. Yazar niteleme çalışmaları için toplanan veri kümesinin özellikleri

120 blog yazarı
Her yazara ait farklı boyutlarda 20 – 100 arası toplamda **6430** doküman
2015 Ocak – 2018 Ekim arası yayınlanmış yazılar
Her yazardan toplamda en az **10.000** kelime
Eşit oranda cinsiyet dağılımı (%50 Kadın, %50 Erkek)
Farklı içerikler
Veri kalitesi kontrolü için Doküman Benzerliği (Kosinüs Uzaklığı ile) Karşılaştırması

Toplanan veri kümesinin birçok çalışmada sağlıklı sonuçlar verebilmesi için yukarıda belirtilen standartlar özellikle konulmuş ve veriler bu standartlara göre toplanmıştır. Bu durum yazarlar arası doküman sayısı ve doküman boyutundaki farklılığın önüne geçmemektedir. Bu sebeple yazar niteleme çalışması için bir takım ön işlemler uygulanmıştır. Uygulanan ön işlemler sırasıyla;

- Her yazara ait dokümanların birleştirilmesi
- Birleştirilen dokümanların eşit sayıda kelime (token "surrounded by spaces") içerek şekilde doküman parçalarına ayrılması
- Her yazara ait eşit sayıda doküman parçalarının olduğu yazar doküman uzayının belirlenmesi

Yapılan standartlaştırma işlemlerinin yazar tanımlamaya etkisini gösterebilmek adına her aşamada veri kümesi 120 sınıflı olarak sınıflandırılmıştır. Her sınıflandırma işlemi için 5 katlı

çapraz doğrulama kullanılmış ve veri kümesinin %20'si test için ayrılmıştır. Sınıflandırma işlemlerinin yapıldığı durumlar sırasıyla;

- Veri kümesine herhangi bir değişiklik yapılmadığı durum,
- Her yazara ait eşit uzunlukta dokümanların olduğu durum
- Her yazara ait hem eşit uzunlukta hem de eşit sayıda dokümanların olduğu durum

Veri kümesinin sınıflandırılması amacıyla metinlerin karakter n-gram ve token n-gram öznitelik kümeleri çıkarılmıştır. Yapılan çalışmada tüm sınıflandırma işlemleri 120 sınıflı bir sınıflandırma olduğu için elde edilen başarının, mevcut yazar tanımlama çalışmalarından nispeten düşük çıkması beklenen bir sonuçtur. Kullanılan öznitelikler ve bu özniteliklere göre elde edilen sınıflandırma sonuçları aşağıda detaylandırılmıştır.

8.2.1. N-gramlar ve Özellikleri

Literatürde n-gram kavramı farklı yapı ve özelliklerdeki veriler için kullanılmaktadır [18]. Örneğin; özellikle konu çıkarımı ile ilgili çalışmalarda kelime n-gramlar kullanılırken, yazar niteleme çalışmalarında sıklıkla karakter n-gram ve token n-gramlar kullanılmaktadır. Bu sebeple bu çalışmada karakter n-gramlar ve token n-gramlar öznitelik kümesi olarak dikkate alınmıştır. Daha ayrıntılı açıklama için, ele alınan n-gramların farklılıkları ve çıkarım özellikleri Tablo 8.2'de örneklenmiştir.

Tablo 8.2. Yazar niteleme çalışmalarında kullanılan n-gram çeşitleri ve örnekleri

Kelime N gram	**Karakter N gram**	**Token N gram (token n prefix)**
Bugün çok güzel bir gün.	Bugün çok güzel bir gün.	Bugün çok güzel bir gün.
* Kelime 3 gram (trigram); -Bugün çok güzel -çok güzel bir -güzel bir gün * Kelime 2 gram (bigram); -Bugün çok -çok güzel -güzel bir -bir gün * Kelime 1 gram (unigram); -bugün -çok -güzel -bir -gün	* Karakter 2 gram (bigram); -Bu-ug-gü-ün-nç -ço-ok-kg-gü-üz -ze-el-lb-bi-… * Karakter 3 gram (trigram); -Bug-ugü-gün-ünç -nço-çok-okg-kgü-güz -üze-zel-elb-… * Karakter 4 gram (four_gram); -Bugü-ugün-günç-ünço -nçok-çokg-okgü-kgüz -güze-üzel-zelb-…	* Token 2 gram (bigram); -bu-ço-gü -bi-gü * Token 3 gram (trigram); -Bug-çok-güz -bir-gün * Token 4 gram (four_gram); -Bugü-çok-güze -bir-gün

Yazar analizi çalışmalarında token n-gramlar için token n-prefix, kelime k-ön ek gibi farklı adlandırmalar bulunmaktadır. Bu çalışmada söz konusu öznitelik kümesi token n-gram olarak kullanılmaktadır. Kelime kök temsili olarak bu öznitelik kümesi Türkçe ve İngilizce çalışmalarda kullanımı bulunmaktadır [18].

8.3. Materyal ve Metot

Yapısı gereği çok sınıflı sınıflandırma problemleri olan yazar tanımlama problemlerinin çözümü için sınıflandırıcı makine öğrenmesi algoritmalarından faydalanılmaktadır. Bu sınıflandırıcıların kullanımı ile veri kümesinde bulunan yazarlara ait bir kısım veriler ile o yazarlara ait örüntüler öğrenilmeye çalışılır. Sınıflandırıcı kullanılarak üretilen eğitilmiş modeller yardımı ile sorgulanan dokümanların öğrenilen yazarlardan birine atanması gerçekleştirilir. Bu çalışmada literatürde yazar tanımlama amacıyla sıklıkla kullanılan 3 farklı

sınıflama algoritması, yani makine öğrenmesi yöntemi kullanılmıştır. Bu yöntemler ve tanımları aşağıda verilmektedir.

8.3.1. Lojistik Regresyon (Logistic Regression) Algortiması

Doğrusal regresyon analizinin geliştirilmiş yöntemi olan lojistik regresyon sınıflandırma çalışmalarında, girilen verilerin hangi sınıfa ait olabileceği olasılığını lojistik fonksiyon kullanarak hesaplamaktadır [73–75]. Bu algoritma temelde En Yakın Komşuluk (Maximum Likelihood Estimetion- MLE) algoritmasını kullanarak sınıflandırma yapar. Lojistik regresyon olasılığa dayalı başarılı hesaplamaları sebebiyle yazar analizi çalışmalarında da sıklıkla tercih edilen yöntemlerden biridir [11, 76]. Belirtilen sebepler doğrultusunda bu çalışmada da lojistik regresyon algoritması sınıflama başarısı hesaplama amacı ile kullanılmaktadır.

8.3.2. Rastgele Orman (Random Forest) Algoritması

Rastgele Orman veya Rassal orman olarak Türkçeye çevrilebilen Random Forest algortiması, farklı karar ağaçlarını optimum çözümü bulana kadar deneyen ve hem tahmin hem de sınıflama çalışmalarında başarılı sonuçlar üretebilen bir algoritmadır [77, 78]. Karar ağaçları arasından parametre ayarı yapmadan probleme en uygun dallanmayı belirleyebildiğinden kullanım kolaylığı sağlamaktadır. Karar ağaçları genellikle öznitelik seçme yöntemi olarak Bilgi Kazanım Oranı'nı (Information Gain Ratio) [79] kullanırken, Rastgele Orman algoritması Gini Index [80] yöntemini kullanır. Yazar analizi çalışmalarında da Rastgele Orman algoritması sıklıkla kullanılmakta ve başarılı sonuçlar elde edilmektedir [81]. Bu çalışmada da sınıflandırma performansı elde etmede Rastgele Orman Algoritması da kullanılmış ve elde edilen sonuçlar diğer algoritmalardan elde edilen sonuçlar ile karşılaştırılmıştır.

8.3.3. Naive Bayes Algoritması

Olasılık tabanlı hesaplamalar ile sınıflandırıcı olarak kullanılan Naive Bayes algoritması, hem tek başına hem de Logistic Regresyon ile karşılaştırmalı olarak birçok çalışmada kullanılmaktadır [82, 83]. Metin madenciliği ve yazar analizi çalışmalarında da çokça kullanılan yöntemlerden biri olan Naive Bayes sınıflandırma algoritması [84] bu çalışmada da sınıflandırma performansı elde etmede kullanılmıştır.

8.4. Kullanılan Öznitelikler ve Öznitelik Seçme Algoritmaları

Kullanılan öznitelik setleri, öznitelik setlerine göre veri kümesinin sayısal özellikleri, sınıflandırma algoritmaları ve elde edilen sonuçlar Tablo 8.3'te gösterilmektedir.

Tablo 8.3. Seçili öznitelik kümelerinin toplanan veri kümesindeki dağılımları

Öznitelik Yapısı	**Veri Kümesi Temsili**	**Sınıflandırma Sonuçları**
Karakter 3-gram	Sözlükteki kelime sayısı = 34.224 Toplam kelime sayısı = 16.314.891 Veri Boyutu (34.224 X 6430)	Logistic Regression = 0.00855 Random Forest = 0.18195 Naive Bayes = 0.09642
Karakter 4-gram	Sözlükteki kelime sayısı = 238.126 Toplam kelime sayısı = 16.308.461 Veri Boyutu (238.126 X 6430)	Logistic Regression = 0.01088 Random Forest = 0.24339 Naive Bayes = 0.18040
Token 4-gram + 3 karakterli ve 2 karakterli tokenler	Sözlükteki kelime sayısı = 27.269 Toplam kelime sayısı = 2.591.587 Veri Boyutu (27.269 X 6430)	Logistic Regression = 0.00777 Random Forest = 0.23872 Naive Bayes = **0.40450**
Token 5-gram + 4 karakterli ve 3 karakterli tokenler	Sözlükteki kelime sayısı = 49.922 Toplam kelime sayısı = 2.373.917 Veri Boyutu (49.922 X 6430)	Logistic Regression = 0.04121 Random Forest = 0.23872 Naive Bayes = 0.26516

Yukarıdaki tablo dikkate alındığında, kullanılan öznitelikler, veri kümesinin öznitelikler tarafından temsil ediciliği ve sınıflandırma sonuçları karşılıklı olarak ele alındığında kullanıma en uygun öznitelik setinin Token 4-gram + 3 karakterli ve 2 karakterli tokenler olduğu çıkarılmıştır. Benzer şekilde yukarıdaki tablo incelendiğinde, bu veriler ve öznitelik kümeleri kullanıldığında en anlamlı sınıflandırma algoritmalarının Random Forest ile Naive Bayes olduğu görülmektedir.

Veri kümesinin doğal halinin özellikleri elde edildikten sonra veri kümesinin normalizasyon işlemlerine geçilebilmesi için bir ön işlem daha gerekmektedir. Bu ön işlem yazar niteleme çalışmalarında sıklıkla kullanılan öznitelik ağırlıklandırma işlemidir. Ağırlıklarına göre seçilen özniteliklerin sınıflandırma aşamasında kullanılması ve diğer özniteliklerin çıkarılması ile öznitelik indirgemesi yapılmaktadır. Öznitelik indirgemesi yapılması sonucu hem sınıflandırılacak verinin boyutu küçüleceğinden sınıflandırma maliyeti düşürülecek hem de sınıflama başarısına etkisi olmayan veya başarıyı düşüren öznitelikler sınıflandırma aşamasında kullanılmayacaktır. Bu amaç doğrultusunda makine öğrenmesi yöntemlerinde sıklıkla

kullanılan, Denklem (11) ile hesaplanan Information Gain [65, 85], Denklem (14) ile hesaplanan Chi_Square [65, 86], Gini Index [87–89] ve Gain Ratio [90] algoritmaları, metin analizi işlemlerinde kullanılan, Denkelm (12) ile hesaplanan tf-idf (term frequency – inverse document frequency) [91, 92] algoritması ve Denklem (14) ile hesaplanan tf-idf'e benzer şekilde sınıflar arası benzerliği dikkate alan tf-icf (term frequency – inverse class frequency) [92–94] algoritmaları kullanılmıştır.

$$tf - idf(w_i, d_j) = tf_{w_i} \times \log\left(\frac{N}{d_j(w_i)}\right) \tag{14}$$

Denklem (14), metin analizi işlemlerinde sıklıkla kullanılan tf_idf öznitelik seçme algoritmasının matematiksel ifadesidir. Tf_idf öznitelik ağırlıklandırma yöntemi, özniteliklerin dokümanlar arası ayırt edicilik değerini çıkarmaya çalışır. Bu gösterimde tf değişkeni terim frekansını temsil ederken $tf_{w(i)}$ değişkeni i. özniteliğin tüm dokümanlardaki frekansını belirtmektedir. N değişkeni veri kümesinde bulunan doküman sayısını temsil ederken, $d_j(w_i)$ değişkeni ise i. özniteliği içeren doküman sayısını temsil etmektedir.

$$tf - icf(w_i, c_j) = \sum_{d \in c_i} tf_{w_i, c_j} \times log\left(\frac{N}{c_j(w_i)}\right) \tag{15}$$

Denklem (15), tf_idf öznitelik ağırlıklandırma algoritması ile benzer amaca sahiptir. Tf_idf algoritması ile veri kümesindeki özniteliklerin dokümanlar arası ayırt ediciliği hesaplanmaya çalışılırken, tf_icf öznitelik ağırlıklandırma algoritması ile verilen özniteliklerin sınıflar arası ayırt edicilik değerleri hesaplanmaktadır. Verilen denklemde tf değişkeni terim frekansını temsil ederken $tf_{w(i)c(j)}$ değişkeni i. özniteliğin j. sınıftaki frekansını temsil etmektedir. N değişkeni veri kümesinde bulunan sınıf sayısını temsil ederken $c_j(w_i)$ değişkeni ise i. özniteliğin bulunduğu sınıf sayısını belirtmektedir.

$$x^2(w_i, c_j) = \sum_i \sum_j \frac{\left(A_{w_i c_j} - E_{w_i c_j}\right)^2}{E_{w_i c_j}} \tag{16}$$

Denklem (16), ki-kare, Chi_Square veya x^2 olarak bilinen öznitelik ağırlıklandırma algoritmasının matematiksel ifadesidir. Gözlenen ve beklenen değerlerin arasındaki farkın anlamlılığının ölçülmesi temeline dayanır. Verilen denklemde w_i değişkeni i. özniteliği, c_j değişkeni j. sınıfı temsil etmektedir. $A_{w(i)c(j)}$ değişkeni i. değişkenin j. sınıftaki ölçülen

frekansını temsil etmektedir. $E_{w(i)c(j)}$ değişkeni i. değişkenin j. sınıftaki beklenen frekansını temsil etmektedir.

8.5. Sonuçlar

Kullanılan veri kümesine göre en anlamlı öznitelik seti olarak tespit edilen token 4-gram özniteliklerinin, belirlenen öznitelik ağırlıklandırma algoritmalarında kullanılıp, en yüksek ağırlıklı 500, 1000 ve 5000 özniteliğin seçilmesi ile yapılan Naive Bayes sınıflandırma işlemleri sonucu Tablo 8.4'te gösterilmektedir.

Tablo 8.4. Seçili ağırlıklandırma yöntemlerinin sınıflandırma başarısı sonuçları

Yöntem / Boyut	**Information Gain**	**Gain Ratio**	**Chi Square (x^2)**	**Gini Index**	**TF-IDF**	**TF-ICF**
500	45,02	15,35	40,05	**44,32**	39,95	29,83
1000	45,09	15,82	44,14	**47,22**	43,11	33,87
5000	41,74	27,22	43,5	**44,51**	41,21	41,03

Kullanmış olduğumuz veri kümesi için en ayırt edici öznitelik setinin Gini Index kullanılarak seçilen 1000 özniteliğin frekanslarının olduğu Tablo 8.4'te görülmektedir. Bu aşamadan sonra yapılacak sınıflandırmalar için seçili özniteliklerin kullanılması uygun görülmüştür.

Veri kümesinin doğal halinin özellikleri elde edildikten sonra veri kümesinin normalizasyon işlemlerine geçilmiştir. Bu aşamada yazarlara ait farklı boyut ve sayıdaki dokümanlar bir araya getirilerek eşit uzunluklu doküman parçaları elde edilmiştir. Öznitelik ağırlıklandırma çalışmalarında en başarılı sonucu 1000 kelimelik öznitelikler verdiği için doküman parçalarının 1000 kelimelik yapıda olmalarına karar verilmiştir. Yazarlara ait dokümanların birleştirilerek 1000 kelimelik dokümanlar halinde ayrılması sonucunda veri kümesinde 120 yazara ait toplamda 2531 doküman elde edilmiştir. Doküman boyutları normalize edildikten sonra öznitelik indirgeme işlemi uygulanmadan elde edilen sınıflandırma sonucu aşağıdadır.

- Standartlaştırılmış Doküman boyutlarının Öznitelik İndirgeme İşlemi Öncesi Sınıflandırma Sonuçları (2.531 X 27.269);
    - Logistic Regression algoritması ile sınıflandırma başarısı (accuracy) = 0.89349
    - Naive Bayes algoritması ile sınıflandırma başarısı (accuracy) = 0.44970
    - Random Forest algoritması ile sınıflandırma başarısı (accuracy) = 0.35700

Yukarıdaki sonuçlar ele alındığında; doküman boyutlarındaki normalizasyon işleminin sınıflandırma başarısını arttırdığı görünmektedir. Bu sonuç, yazarların doküman bazlı temsil ediciliğini ortadan kaldırmış, bir yazara ait her dokümanın eşit oranda temsil ediciliğe katılmasını sağlamıştır. Doküman birleştirme işlemi rastgele yapıldığından dokümanların zaman bağımlılığı da ortadan kalkmış ve elde edilen sonucun daha anlamlı olması sağlanmıştır.

Doküman boyutları normalize edildikten sonra öznitelik indirgeme işlemi yapılmıştır. Her yazarın farklı sayılarda fakat aynı uzunlukta dokümanlarının bulunduğu uzayın, seçili özniteliklere göre öznitelik indirgeme işlemi sonrası elde edilen sınıflandırma başarısı aşağıda gösterilmektedir.

- Standartlaştırılmış Doküman Boyutlarının Öznitelik İndirgeme İşlemi Sonrası Sınıflandırma Sonuçları (2.531 X 1000);
    - Logistic Regression algoritması ile sınıflandırma başarısı (accuracy) = 0.88165
    - Naive Bayes algoritması ile sınıflandırma başarısı (accuracy) = 0.87771
    - Random Forest algoritması ile sınıflandırma başarısı (accuracy) = 0.44575

Öznitelik indirgeme işlemi sonrası elde edilen sınıflandırma başarısında kullanılan algoritmaya göre artışlar veya azalışlar olmaktadır. Öznitelik indirgeme işlemi ile sınıflandırma maliyeti büyük ölçüde azalmaktadır. Bunun sonucunda bireysel faydası düşük öznitelikler sınıflandırma işleminden çıkarılırken, bu özniteliklerin toplu etkisi de sonuca yansımaktadır. Öznitelik indirgeme işlemi bir fayda – zarar dengesi oluşturmaktadır, bu sebeple indirgeme işlemi kararı kullanıcıya bırakılmaktadır. Bu çalışmada veri kümesini en minimum boyutta anlamlı olarak nasıl temsil edilebileceği gösterilmek istendiğinden öznitelik indirgeme işlemi yapılması tercih edilmiştir.

Doküman boyutları normalize edildikten sonra veri kümesindeki her yazara ait doküman sayısındaki dengesizliği de ortadan kaldırmak için her yazara ait eşit sayıda (Bu çalışma için 10 doküman kullanılmaktadır.) doküman parçası alınarak tekrar sınıflama yapılmıştır. Elde edilen sonuçlar aşağıdadır.

- Doküman Sayısı Dengeleme Sonrası Sınıflandırma Sonuçları (1200 X 1000);
    - Logistic Regression algoritması ile sınıflandırma başarısı (accuracy) = 0.825
    - Naive Bayes algoritması ile sınıflandırma başarısı (accuracy) = 0.6958
    - Random Forest algoritması ile sınıflandırma başarısı (accuracy) = 0.3333

Veri kümesindeki doküman sayısının yarısından fazlasının sınıflandırmaya katılmamış olmasına rağmen elde edilen sınıflandırma sonucu tatmin edici ve anlamlıdır. Sınıflandırmalar 120 yazarın eşit oranda verisi alınarak yapıldığından elde edilen modeller güvenilirdir. 120 yazarlı bir doküman uzayının adil, güvenilir ve başarılı bir şekilde sınıflandırılabilmesi, yapılan çalışmaların farklı veri kümelerinde, farklı yazarlar kullanıldığında da anlamlı sonuçlar vereceğinin bir göstergesidir.

8.6. Tartışma

Metin analizi çalışmalarından biri olan Yazar Niteleme çalışmaları uzun yıllardır popülerliğini korumaktadır. Uluslararası çalışmalar karşısında Türkçe üzerine yapılan çalışmalar, veri kümesi eksikliğinden dolayı yetersiz kalmıştır. Bu çalışmada söz konusu yetersizlik göz önüne alınarak öncelikle yazar analizi çalışmalarında kullanılmak üzere iyi tanımlı bir veri kümesi toplanmıştır. Toplanan veri kümesi doğal yapısı korunarak ve daha sonra normalize edilerek Yazar Niteleme işlemleri sonuçları karşılaştırılmıştır. Veri kümesi 120 sınıflı olmasından dolayı elde edilen sonuçlar çok başarılı olmamasına rağmen anlamlıdır. Çalışmada ele alınan Veri Kümesinin Normalize Edilmesi yaklaşımı, literatürde ele alınan veri kümelerinin tekrar değerlendirilmesi gerekliliğini ortaya koymuştur. Veri kümesi normalizasyonu ile her yazar sınıfı eşit sınırlar çerçevesinde değerlendirilmiş böylece elde edilen sonucun daha anlamlı olması sağlanmıştır. İleriki çalışmalarda, toplanan veri kümesi ve elde edilen başarılı çıktılar kullanılarak Yazar nitelemenin alt problemlerine çözüm bulma çalışmaları yapılacaktır.

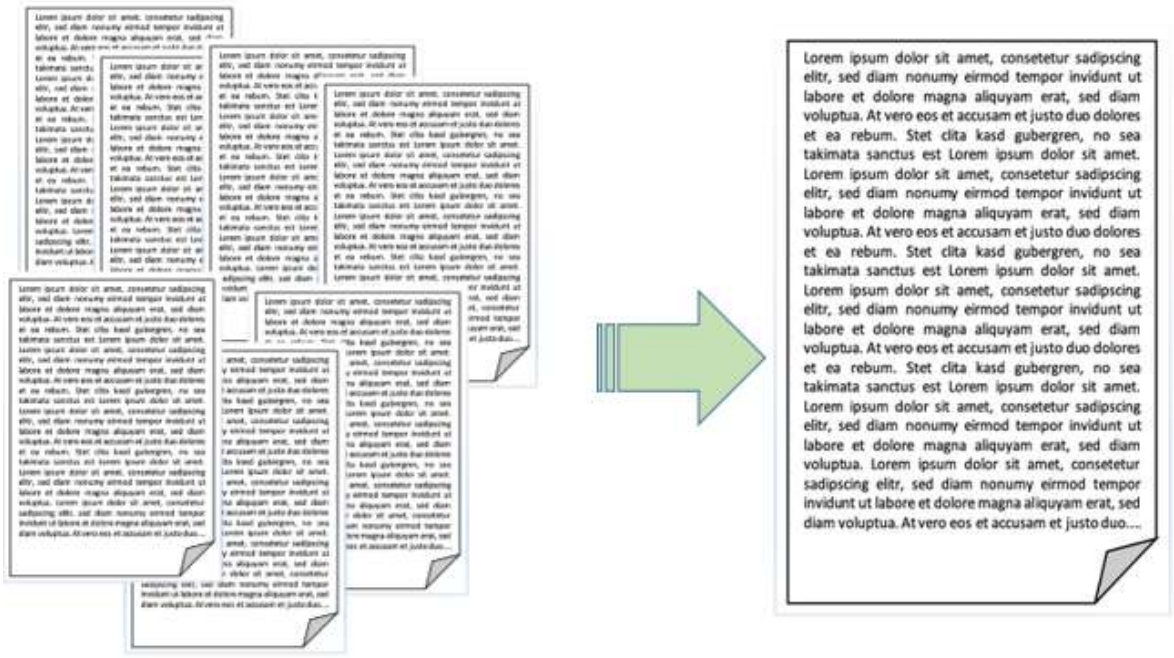
9. TÜRKÇE METİNLER İÇİN YAZAR DOĞRULAMA YAKLAŞIMI, ÖZNİTELİK KÜMESİ VE SINIFLANDIRMA ALGORİTMASI SEÇİMİ

Bu bölümde, derlemiş olduğumuz Türkçe Blog Yazıları külliyatı kullanılarak yazar doğrulama probleminin çözümüne yönelik uygulamalar gerçekleştirilmiştir. Bu uygulamalar, yazar doğrulama problemi özelinde önerdiğimiz çözüme yönelik uygulamalar olup, çözümde kullanılması gereken özniteliklerin ve sınıflandırma algoritmalarının seçimine yönelik karşılaştırmalı deneyleri içermektedir. Yazar doğrulama probleminin çözümüne yönelik önermiş olduğumuz yaklaşımın ve kullanmamız gereken araçların belirlendiği bu çalışma, yazar doğrulama problemine evrensel bir çözüm sunmaya çalıştığımız bu tez çalışmasının önemli bir adımı olarak değerlendirilmektedir.

Önceki bölümlerde yapılan çalışmalar, yazar doğrulama problemine yaklaşımımızı şekillendirmek için ele aldığımız çalışmaları içermekteydi. Önceki bölümlerden elde edilen çıktılar doğrultusunda yazar doğrulama probleminin çözümüne bir yaklaşım geliştirdiğimiz bu çalışmada öncelikle Türkçe metinlerin yazarlık doğrulaması üzerine deneyler gerçekleştirdik. Geliştirmiş olduğumuz yaklaşım birkaç adımdan oluşmaktadır. Bu adımların oluşmasının sebebi önceki çalışmalarda elde ettiğimiz anlamlı sonuçların uygulanabilmesidir. Bu adımlar ve içerikleri aşağıdaki alt bölümlerde gösterilmektedir.

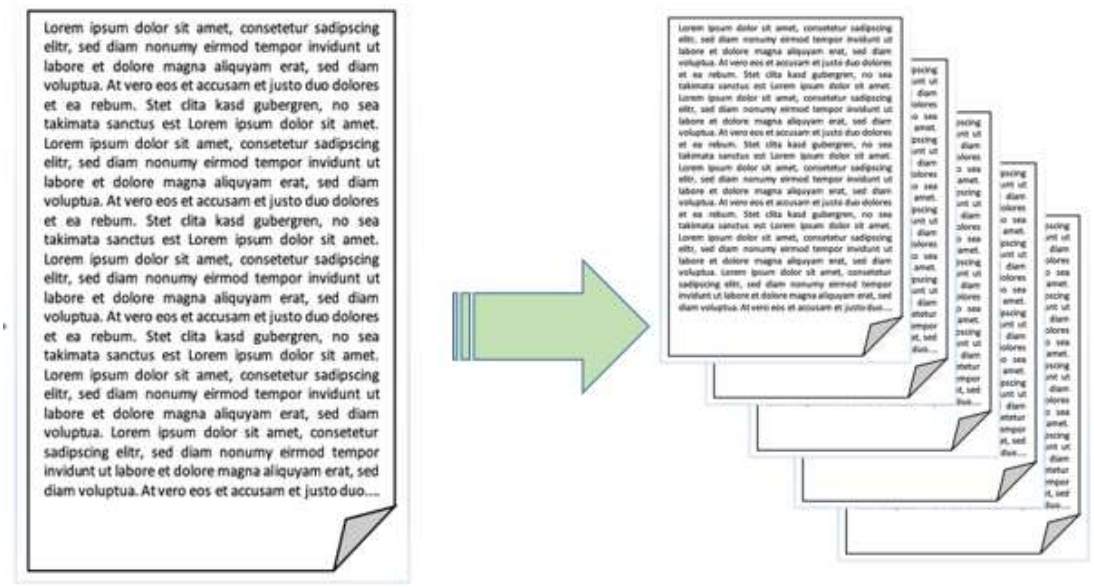
9.1. Veri Kümesi Ön İşleme Adımları

Önceki çalışmalarda elde ettiğimiz sonuçlar doğrultusunda, öncelikle değerlendirilecek veri kümesinin bir takım ön işlemlerden geçmesi gerektiği görülmüştür. Bu işlemlerden ilki bir yazara ait farklı zaman ve konularda yazılmış dokümanların bir araya getirilmesi ile tek bir konu veya zaman bağımlı yazılardan bağımsız bir temsil etmektir. Bu amaç doğrultusunda yapılan doküman birleştirme işlemi Şekil 9.1'de gösterilmektedir.



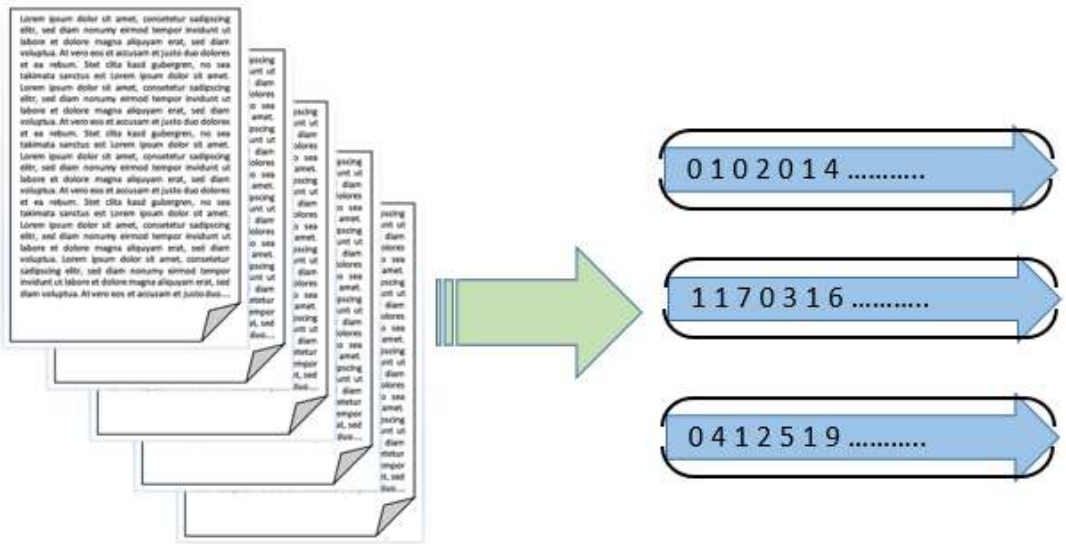
Şekil 9.1. Veri kümesi ön işleme adımları (Adım 1)

Veri kümesinin ön işlem adımları Şekil 9.1’de görüldüğü gibi öncelikle doküman birleştirme üzerinedir. Bu aşamada birleştirilen dokümanlar tek bir yazara ait toplamda belli bir standardı sağlayacak şekilde bir araya getirilmektedir. Şöyle ki; bu aşamada kullanacağımız verinin dengeli olmasını istediğimizden her yazara ait toplamda 10.000 kelimeye ulaşana kadar, rastgele seçilmiş dokümanlar arka arkaya eklenmiştir. Böylece, elimizde her yazara ait 10.000 kelimelik bir temsil dokümanı bulunmaktadır. Birçok yazar analizi çalışmasında bu doküman profil dokümanı olarak ele alınmıştır. Şekil 9.2’de veri kümesinde yapılması gereken ön işlemlerden ikincisi görülmektedir.



Şekil 9.2. Veri kümesi ön işleme adımları (Adım 2)

Her yazara ait rastgele seçili dokümanların bir araya getirilmesi ile elde edilen 10.000 kelime uzunluğundaki profil dokümanı, Şekil 9.2’deki temsili gösterimde olduğu gibi 2. adımda n eşit parçaya bölünerek n tane doküman elde edilmektedir. Bu aşamadaki n değişkeninin değeri yazara ait ele alınacak doküman sayısını belirlemektedir. Bu tez çalışması kapsamında 500 ve 1000 kelimelik dokümanlar üzerine yoğunlaşıldığından n değer 10 ve 20 olarak farklı deneyler kapsamında ileriki çalışmalarda ele alınmıştır. Şekil 9.3’te veri kümesinde yapılması gereken ön işlemlerden üçüncüsü görülmektedir.

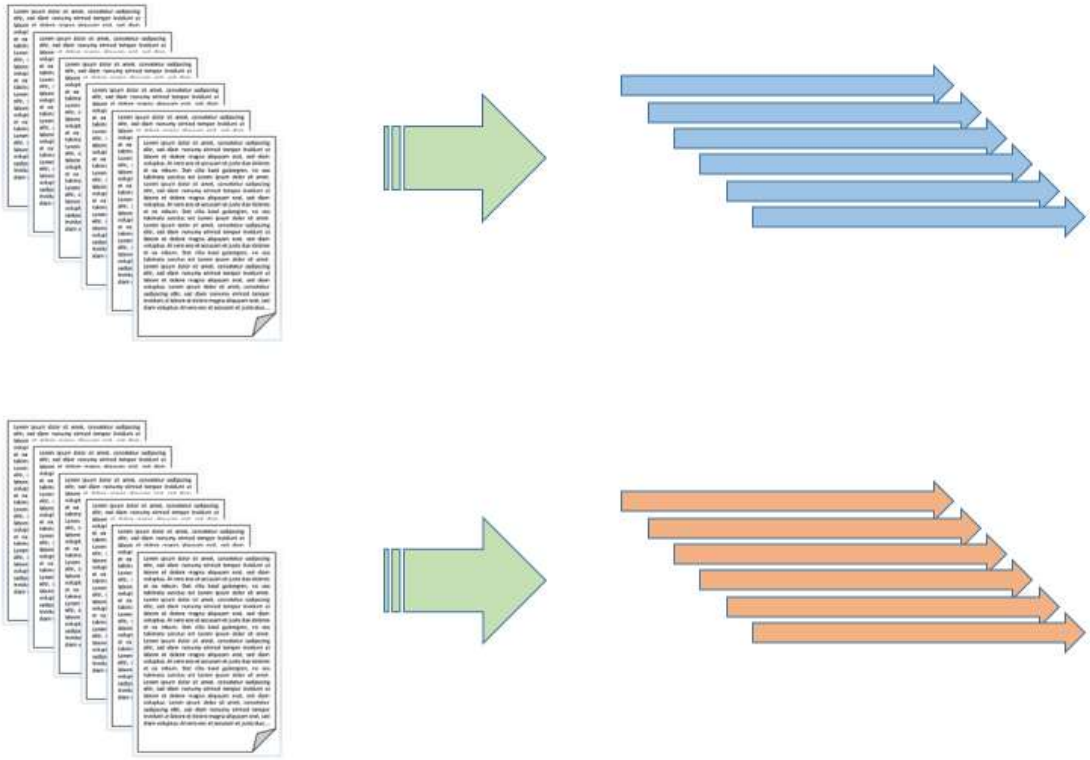


Şekil 9.3. Veri kümesi ön işleme adımları (Adım 3)

Veri kümesi üzerinde yapılacak son işlem, Şekil 9.3'te görüldüğü gibi, üretilen eşit uzunluktaki dokümanların seçili öznitelikler kullanılarak vektörel temsillerine dönüştürülmesidir. Bu aşamada her yazara ait üretilen dokümanların belirlenen öznitelik kümesine göre vektörleri oluşturulmaktadır. Oluşturulan öznitelik vektörleri de, tüm dokümanlar için aynı öznitelik kümesi kullanıldığından hepsi eşit sayıda boyutta olmaktadır. Bu çalışmada genel olarak kelime çantası (bow) öznitelik kümesi kullanılmıştır. Çalışmanın devamında bow öznitelik kümesi farklı ağırlıklandırma yöntemleri ile ele alınmış olmasına rağmen vektorel olarak dokümanları temsil eden vektörlerin boyutu sabittir.

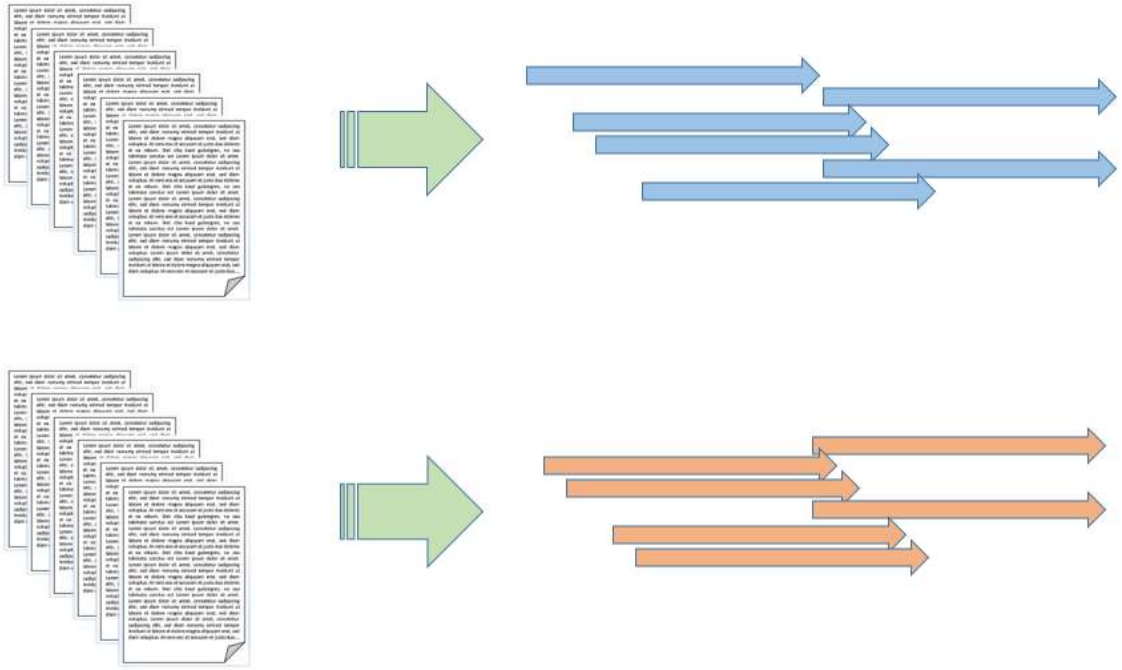
9.2. Yazar Doğrulama Örneklerinin Üretilmesi ve Etiketlenmesi

Yazar doğrulama problemine yönelik çalışmalar, daha önce de bahsedildiği gibi, iki türlü ele alınmaktadır. Sorgulanan bir dokümanın belirli bir yazara ait olup olmadığının belirlenmeye çalışıldığı yazar doğrulama çalışmaları ve iki dokümanın aynı yazar tarafından yazılıp yazılmadığının belirlenmeye çalışıldığı yazar doğrulama çalışmaları. Biz bu tez kapsamında, yukarıda bahsedilen ikinci durum için, başarılı bir yaklaşım sunabilmek adına birçok ön çalışma gerçekleştirdik. Bu ön çalışmaların çıktısı olarak bu bölümde önerdiğimiz yaklaşımı geliştirmiş bulunmaktayız. Bu yaklaşım içerisinde ele aldığımız probleme özgü örneklerin üretimi de bu çalışma içerisinde gerçekleşmesi gereken aşamalardan biridir. Söz konusu problem kapsamında ele alınacak veri kümesinde aynı yazara ait ve farklı yazarlara ait olmak üzere doküman çiftleri bulunmalıdır. Ele aldığımız probleme özgü örneklerin nasıl üretildiği temsili olarak aşağıdaki şekillerde gösterilmektedir. Şekil 9.4'te rastgele seçilmiş iki yazara ait verilerin temsili gösterimi bulunmaktadır.



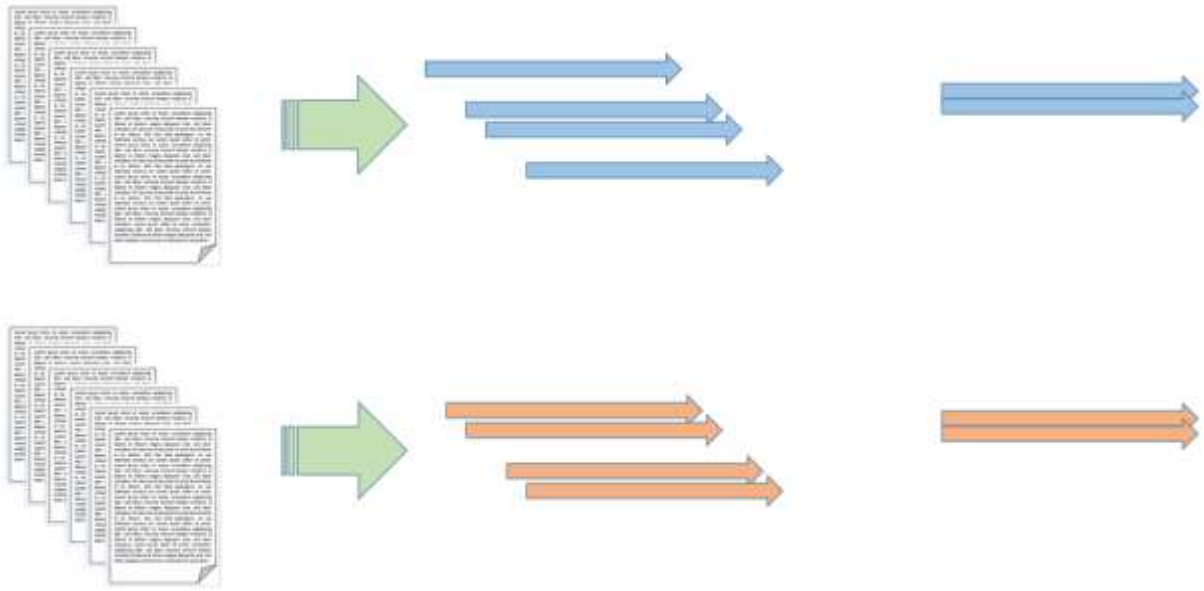
Şekil 9.4. Veri kümelerinden rastgele seçilmiş iki yazara ait verilerin temsili gösterimi

Yazar doğrulama bakış açısı ile bu çalışmada değerlendireceğimiz veri kümesinde, aynı yazar tarafından üretilmiş ve farklı yazarlar tarafından üretilmiş doküman çiftlerinin bulunması gerekmektedir. Bu gereklilik doğrultusunda öncelikle veri kümesinde bulunan yazarlardan Şekil 9.4'te görüldüğü gibi rastgele ikililer seçilmiştir. Seçilen bu ikililerin dokümanlarını temsilen üretilen vektörler de temsilen görülmektedir. Bu aşamada üretilmesi gereken örnekler için seçili yazarlardan üretilmiş rastgele doküman çiftleri alınmaktadır. Bu aşamanın temsili Şekil 9.5'te görülmektedir.



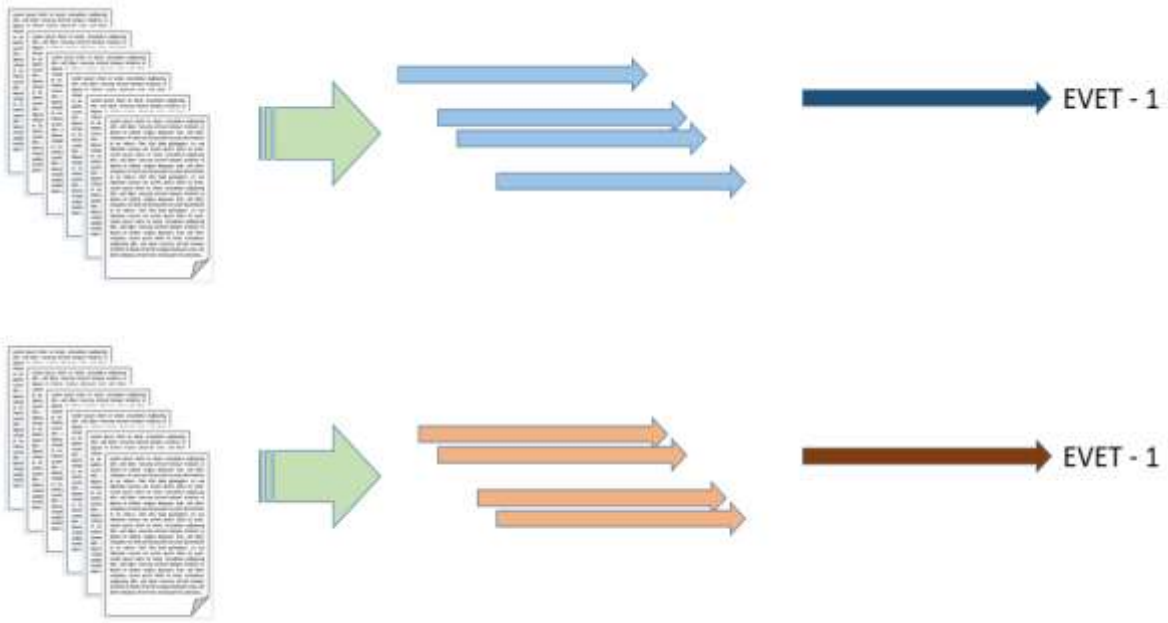
Şekil 9.5. Yazarlara ait rastgele doküman çifti seçiminin temsili gösterimi

Sadece iki yazara ait görselin bulunduğu Şekil 9.5'teki işlem tüm yazarlara uygulanmaktadır. Bu aşamada her yazardan rastgele olacak şekilde (bu noktada ard arda olmayan dokümanların seçimine özen gösterilmiştir) doküman çiftleri seçilmiştir. Seçili bu doküman çiftlerinin yani dokümanları temsil eden vektör çiftlerinin makine öğrenimi, veri madenciliği veya yapay zeka teknikleri kullanılarak ele alınabilmeleri için tek bir örnek haline getirilmeleri gerekmektedir. Dolayısı ile bir birleştirme yöntemi kullanılarak seçili vektör çiftlerinin birleştirilip tek bir örnek haline getirilmesi gerekmektedir. Şekil 9.6'da doküman çiftlerinin birleşiminin temsili gösterimi bulunmaktadır.



Şekil 9.6. Seçili doküman çiftlerinin birleşiminin temsili gösterimi

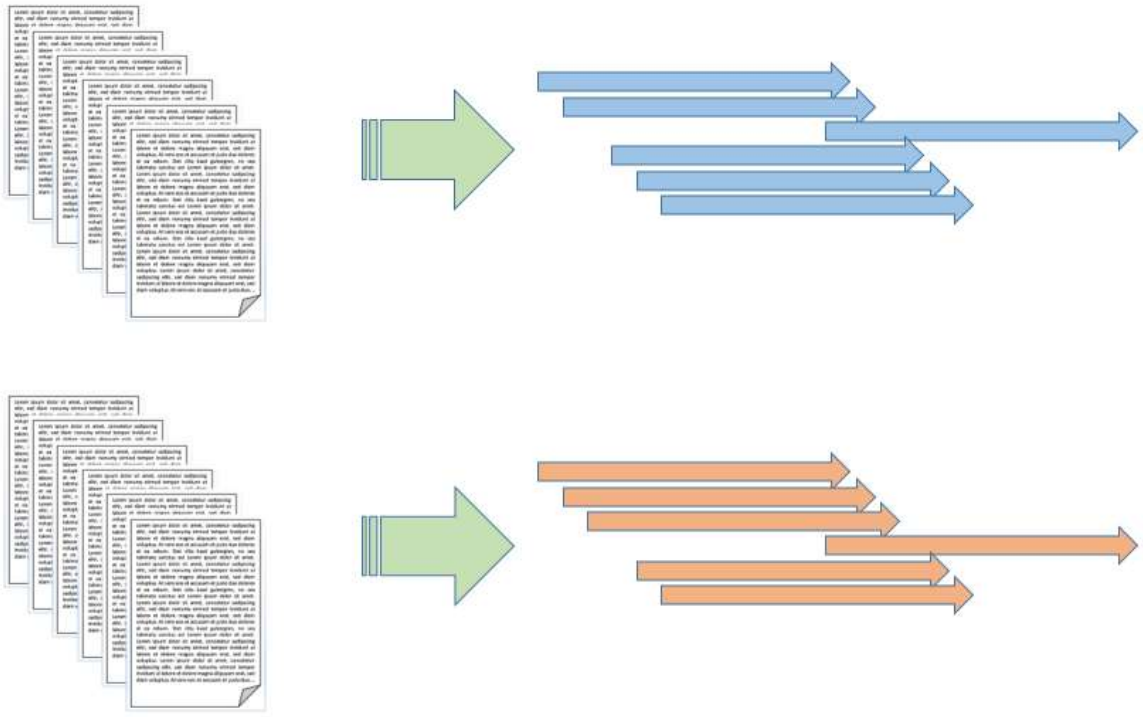
İki dokümanın yazar doğrulamasının önerdiğimiz yaklaşımda yapılabilmesi için tek bir temsile dönüştürülmesi gerekmektedir. Şekil 9.6'da gösterildiği gibi, veri kümesinde bulunan her yazara ait rastgele seçilmiş doküman çiftlerinin birleştirilmesi ile ele aldığımız probleme uygun pozitif örnekler üretilebilmektedir. Bu aşamada farklı birleştirme işlemleri kullanılabilmektedir. Devam eden bölümlerde hangi birleştirme işlemlerinin ele alındığı deneysel çalışmalar ile ayrıntılı olarak verilmektedir. Belirli bir birleştirme işlemi kullanılarak tek bir vektöre dönüştürülen doküman temsili vektör çiftlerinin etiketlenme işlemi yapılması gerekmektedir. Şekil 9.7'de aynı yazarlara ait doküman çiftlerinin birleştirilerek etiketlenme adımı temsili olarak gösterilmektedir.



Şekil 9.7. Aynı yazar tarafından yazılan doküman çiftlerinin birleşiminin etiketlenmesi

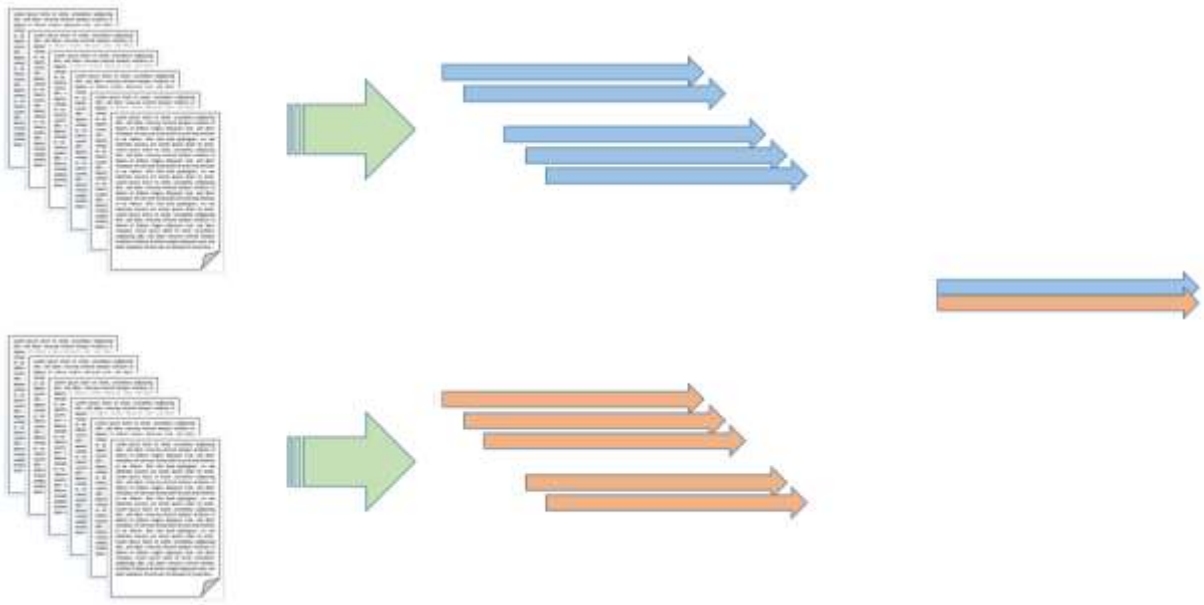
İki dokümanın yazar doğrulaması problemine yönelik üretilen çözümlerin girilen örneklere göre iki cevabı bulunmaktadır. Sorgulanan iki dokümanın aynı yazar tarafından yazılıp yazılmadığının iki cevabı vardır; "evet" ve "hayır". Eğer sorgulanan doküman çifti aynı yazar tarafından üretilmiş ise, geliştirilen çözümün bu örneğe "evet" yani "bu iki doküman aynı yazar tarafından yazılmıştır" cevabını verebilmesi beklenir. Yazar doğrulama probleminin çözümüne yönelik önerdiğimiz yaklaşım içerisinde eğitilmiş bir model barındıracağından, modelin eğitilmesi için kullanacağımız örneklerin "1" yani "evet" ve "0" yani "hayır" etiketlerine sahip örnekleri barındırması gerekir. Bu aşamada aynı yazarlar tarafından yazılmış doküman çiftlerinin birleşimiyle elde ettiğimiz vektörü "1" olarak yani evet olarak Şekil 9.7'de görüldüğü gibi etiketlenmiştir.

Yazar doğrulama probleminin çözümünde kullanılmak üzere probleme özgü veri kümesi oluşturmak amacıyla yukarıda anlatıldığı gibi pozitif örnekler üretilmiştir. Benzer şekilde negatif örneklere de, yani farklı yazarlar tarafından yazılmış doküman çiftlerinin birleşim vektörüne, ihtiyaç vardır. "evet" örneklerinin üretimine benzer şekilde "hayır" örneklerinin üretimi için de ele aldığımız veri kümesinden temsili olarak iki yazar ait dokümanlardan rastgele birer doküman seçilmektedir. Şekil 9.8'de farklı yazarlara ait doküman çiftlerinin seçiminin temsili gösterimi bulunmaktadır.



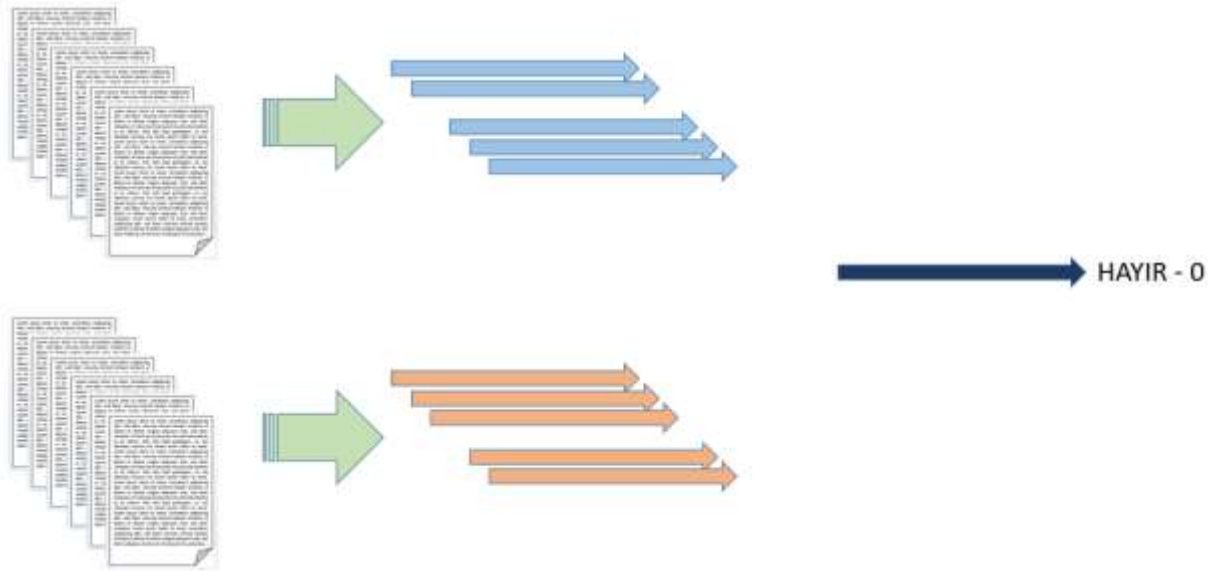
Şekil 9.8. Farklı yazarlara ait rastgele dokümanların seçimi

Farklı yazarlar tarafından yazılmış doküman çiftlerinin örneklerini oluşturmak için Şekil 9.8'de görüldüğü gibi veri kümesinde bulunan her yazara ait dokümanlar rastgele olarak (bu aşamada bir yazara ait dokümanların sürekli olarak aynı başka bir yazara ait dokümanlar ile eşleştirilmemesi için özen gösterilmiştir.) başka yazarlara ait dokümanlar ile eşleştirilmiştir. Bu aşamada bir yazara ait dokümanların mümkün olduğu kadar veri kümesindeki farklı yazarlar ile eşleştirilmesi için geliştirilen algoritma bir sonraki ana bölümde ayrıntılı olarak ele alınmıştır. Şekil 9.9'da farklı yazarlara ait doküman çiftlerinin bileşim işleminin temsili gösterimi bulunmaktadır.



Şekil 9.9. Farklı yazarlara ait doküman çiftlerinin birleşiminin temsili gösterimi

Aynı yazara ait doküman bileşiminde olduğu gibi, farklı yazarlar ait dokümanları temsil eden vektörler de belirli bir birleştirme işlemi kullanılarak tek bir vektöre dönüştürülmektedir. Birleştirme işlemi sonucu oluşan vektörün etiketlenmesi görseli Şekil 9.10'da verilmektedir.



Şekil 9.10. Farklı yazarlar tarafından yazılmış doküman çiftlerinin birleşiminin etiketlenmesi

Birleştirme işlemi sonucu "1" ile etiketlenen "evet" örneklerinde olduğu gibi, aynı birleştirme işleminin farklı yazarlara ait doküman çiftlerine kullanılması ile üretilen birleşim vektörü "0" olarak yani "hayır" olarak etiketlenmiştir. Ön işlemler sonrası elde ettiğimiz veri kümesinde bulunan tüm yazarlar ve tüm dokümanlar için bu işlemler yapıldıktan sonra yeni bir veri kümesi

oluşturulmuş olmaktadır. Şekil 9.11'de bu yeni veri kümesinin içeriği temsili olarak gösterilmektedir.

Şekil 9.11. Yazar doğrulama yaklaşımında kullanılacak vektör uzayının temsili gösterimi

Aynı yazarlara ait ve farklı yazarlara ait doküman çiftlerinin birleştirilmesi ile Şekil 9.11'de görüldüğü gibi, yeni oluşturduğumuz veri kümesinde 0 ve 1 etiketlerine sahip vektörel yapılar mevcuttur. Yani yeni veri kümemiz iki sınıflı bir küme haline gelmiş ve makine öğrenme algoritmaları, veri madenciliği yaklaşımları, yapa zeka teknikleri gibi farklı sınıflandırma algoritmaları kullanılarak iki sınıflı bir model üretilebilecek bir yapıdadır.

9.3. Vektörel Birleşimde Ele Aldığımız İşlemler

Değerlendirilecek iki dokümanın ortak bir temsilini elde etmek amacı ile bu çalışmada bazı vektörel birleştirme işlemleri uygulanmıştır. Bu işlemler girdi olarak aldığı iki vektörün ortak bir temsilini üretebilmesi beklenen işlemlerdir. Çok karmaşık hesaplamalar içermeyen bu işlemler ele alınan vektörlerin her elemanını işleyip ortak bir temsil çıkaracak niteliktedir. Şöyle ki; A_1, A yazarına bir numaralı dokümanın vektörü olsun. $A_1(i)$, A yazarına ait bir numaralı doküman vektörünün i. özniteliğinin değeri olmak üzere; i değeri 0 ile doküman vektörlerinin boyutu arasında değer alır. Bu çalışmada kullanılan doküman birleştirme yöntemleri aşağıda detaylandırılmıştır.

Denklem (17) ile gösterilen ilk işlem, birleştirme işlemi olarak toplama işleminin kullanıldığı formüldür. Birleştirilecek iki vektörün eş indexleri toplanarak yeni oluşacak birleşim vektörünün aynı indexteki değerini oluşturmaktadır.

$$V_B(i) = \sum (A_1(i), A_2(i)) \tag{17}$$

V_B iki vektörün birleşim vektörünü temsil etmek üzere, $V_B(i)$ birleşim vektörünün i. özniteliğinin değerini temsil etmektedir. Denklem (18) ile gösterilen ikinci birleştirme işlemi

fark alma işlemidir. Burada birleştirilecek iki vektörün eş indexlerinin farkı alınarak oluşacak yeni vektörün aynı indexinin değeri elde edilmektedir.

$$V_B(i) = A_1(i) - A_2(i) \tag{18}$$

Denklem (19) ile gösterilen üçüncü birleştirme işlemi ortalama alma işlemidir. Burada birleştirilecek iki vektörün eş indexlerinin aritmetik ortalaması alınarak oluşacak yeni vektörün aynı indexinin değeri elde edilmektedir.

$$V_B(i) = \frac{1}{2}\sum(A_1(i), A_2(i)) \tag{19}$$

Denklem (20) ile gösterilen dördüncü birleştirme işlemi çarpım işlemidir. Burada birleştirilecek iki vektörün eş indexlerinin çarpımı alınarak oluşacak yeni vektörün aynı indexinin değeri elde edilmektedir.

$$V_B(i) = \prod(A_1(i), A_2(i)) \tag{20}$$

Denklem (21) ile gösterilen beşinci birleştirme işlemi çarpımın karekökünün alınması işlemidir. Burada birleştirilecek iki vektörün eş indexlerinin çarpımlarının karekökü alınarak oluşacak yeni vektörün aynı indexinin değeri elde edilmektedir.

$$V_B(i) = \left(\prod(A_1(i), A_2(i))\right)^{\frac{1}{2}} \tag{21}$$

Yukarıdaki işlemler veri kümesinde eşleştirme için belirlediğimiz "0" ve "1" etiketli tüm doküman çiftlerine uygulanmış, elde edilen birleşim vektörleri doküman çiftlerinin etiket bilgisine göre etiketlenmiştir. Kullanılan beş farklı birleştirme işlemi ile aynı veri kümesinden 5 farklı veri kümesi elde edilmektedir.

9.4. Değerlendirilen Öznitelik Kümeleri

Önceki bölümlerde yapılan hazırlık çalışmaları sonucunda kelime k-ön kök (token k-prefix) adı verilen özniteliklerin sık kullanım (frekans) değerlerinin Türkçe metinlerin yazar tanımlamasında anlamlı sonuçlar verdiği çıkarımı elde edilmiştir. Ayrıca, yine önceki bölümlerde yapılan hazırlık çalışmalarında Gini index ağırlıklandırması kullanılarak ağırlıklarına göre seçilen verilerin yazar tanımlamada ele alınan diğer ağırlıklandırma yöntemlerinden daha ayırt edici sonuçlar ürettiği çıkarımı elde edilmiştir. Elde edilen çıkarımlar

doğrultusunda bu bölümde, eldeki veri kümesi kullanılarak farklı öznitelik kümeleri kullanılarak hem yazar tanımlama hem yazar doğrulama yapılmıştır.

Bu bölümde, bir önceki bölümde de kullanılan, bu tez çalışması kapsamında derlenmiş Türkçe Blog Külliyatı veri kümesi olarak kullanılmıştır. Kullanılan veri kümesinde 120 yazara ait blog metinleri bulunmaktadır. Veri kümesinin daha ayrıntılı özellikleri bir sonraki bölümde verilmektedir. Bu çalışmada veri kümesinde dokümanlar arası birleştirme işlemi yapılmadan önce, yani her yazara ait eşit sayıda ve eşit boyutta n tane dokümanın bulunduğu durumda veri kümesi yazar aşağıda verilen öznitelikler ile 120 sınıflı ve 100 sınıflı olarak sınıflandırma işlemine tabi tutulmuştur. Daha sonra yukarıda açıklanan birleştirme işlemleri ile 120 sınıflı veri kümesi iki sınıflı yapıya dönüştürülüp yazar doğrulama çalışması bakımından sınıflandırma başarısı ölçülmüştür.

Bu çalışmada gerçekleştirilen tüm sınıflandırma işlemlerinde veri kümesinin %70'i ele alınan sınıflandırıcı modelin eğitiminde, %30'u ise modelin testinde kullanılmıştır. Başarı ölçümü olarak da formülü önceki bölümlerde verilen doğruluk (accuracy) başarı ölçeği kullanılmıştır. Bu çalışmada kullanılan öznitelikler ve bu özniteliklerden elde edilen doğruluk değerleri aşağıda alt başlıklar halinde sunulmaktadır.

9.4.1. Özniteliklerin Kullanım Sıklıkları

Kelime k-ön kök özniteliklerinin kullanıldığı bu çalışmada k değişkeni olarak, bir önceki çalışmalarda kullanımı ile başarılı sonuçlar elde edilmesi sebebi ile 4 kullanılmıştır. Veri kümesi ön işleme adımlarından sonra veri kümesinde bulunan her dokümandaki kelimelerin 4-ön-kök verileri elde edilmiştir. Yine önceki çalışmalardan elde edilen sonuçlar doğrultusunda, sayıca fazla olan bu öznitelikler, kullanımları ile üretilen temsil vektörlerinin seyrek bir vektör olmasına sebep olacaktır. Dolayısı ile bu öznitelikler önceki çalışmalarda kullanımı başarılı bulunan Gini index ağırlıklandırma algoritması ile ağırlıklandırılmış ve en anlamlı bulunan 1000 öznitelik değerlendirmeye alınmıştır. Veri kümesinde bulunan bu 1000 özniteliğin her doküman için kullanım sıklıkları hesaplanmış ve elde edilen değerler ile öncelikle doküman temsili vektörler üretilmiştir. Yazar etiketleri barındıran bu vektörler 120 sınıflı ve 100 sınıflı olarak sınıflandırma işlemine tabi tutulmuştur. Daha sonra yine bu temsil vektörlerine yukarıda bahsedilen birleştirme işlemleri uygulanmış ve elde edilen veri kümesi 2 sınıflı sınıflandırma işlemine tabi tutulmuştur. Bu aşamada ikili sınıflandırma işlemi 100 sınıflı veriye uygulanmış ve bu sınıflandırma işleminde hiç bulunmayan diğer 20 yazara ait üretilen yazar doğrulama

örneklerinden elde edilen sonuçlar da validasyon sonucu olarak değerlendirmeye alınmıştır. Belirlenen özniteliklerin kullanım sıklıkları kullanılarak yapılan sınıflandırma sonuçları Tablo 9.1'de gösterilmektedir.

Tablo 9.1. Belirlenen özniteliklerin kullanım sıklıklarından elde edilen sınıflandırma sonuçları

Özniteliklerin Kullanım Sıklıkları							
	Çoklu Sınıflandırma		İkili Sınıflandırma				
Sınıflandırma Yöntemi	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,825	0,825	0,37	0,305	0,38	0,63	0,555
AdaBoost	0,0708	0,095	0,6	1	0,6	0,64	0,65
Random Forest	0,3333	0,315	0,41	0,735	0,49	0,61	0,545
SVM	0,6875	0,58	0,44	0,7	0,525	0,62	0,475
NaiveBayes	0,6958	0,61	0,35	0,585	0,35	0,655	0,655

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,525	0,545	0,515	0,735	0,61
AdaBoost	0,595	1	0,595	0,65	0,65
Random Forest	0,48	0,765	0,545	0,555	0,585
SVM	0,585	0,745	0,67	0,765	0,5
NaiveBayes	0,445	0,68	0,445	0,665	0,66

Çoklu Sınıflandırma ve İkili Sınıflandırma alt başlıkları ile verilen değerler Tablo 9.1'de gösterildiği gibi kullanılan sınıflama yöntemlerine göre büyük değişiklikler göstermektedir. İkili sınıflama alt başlığında verilen yazar doğrulama sonuçları veri kümesinde bulunan, rastgele seçilmiş 100 yazarın verileri kullanılarak yapılan ikili sınıflama sonuçlarıdır. Ayrıca validasyon alt başlığında verilen sonuçlar ikili sınıflamada hiç verisi bulunmayan 20 yazara ait verilerin yazar doğrulaması sonuçlarıdır. Çoklu sınıflandırma ve ikili sınıflandırma başlığı altında verilen tüm sonuçlar test kümesinden elde edilen sonuçlardır. Bu ve bundan sonraki tüm tablolarda verilen değerler, elde edildikleri işlemlerin 5 defa tekrarlanması ile çıkan sonuçların ortalaması alınarak bulunmuştur.

9.4.2. Özniteliklerin Normalleştirilmiş Kullanım Sıklıkları

Kelime 4-ön kök özniteliklerinin kullanıldığı bu çalışmada bir önceki öznitelik değerlerinin, öznitelik bazında, yani öznitelik-doküman matrisinin sütun bazında değerlerinin normalleştirilmesi ile elde edilen değerler kullanılmıştır. Normalleştirme işlemi olarak önceki bölümlerde bahsedilen min-max normalleştirme yöntemi kullanılmıştır. Böylece dokümanların vektörel temsillerindeki öznitelik değerleri 0-1 aralığında olmaktadır. Bu uygulama hem

sınıflandırma yöntemlerinin daha etkili çalışmasına hem de verinin daha düzgün bir dağılımla temsil edilebilmesine katkı sağlayacaktır. Belirlenen özniteliklerin normalleştirilmiş kullanım sıklıkları kullanılarak yapılan sınıflandırma sonuçları Tablo 9.2'de gösterilmektedir.

Tablo 9.2. Belirlenen özniteliklerin normalleştirilmiş kullanım sıklıklarından elde edilen sınıflandırma sonuçları

Özniteliklerin Normalleştirilmiş Kullanım Sıklıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,7916	0,78	0,4	0,39	0,44	0,635	0,595
AdaBoost	0	0,065	0,5	0,8	0,5	0,61	0,625
Random Forest	0,3	0,315	0,465	0,575	0,42	0,53	0,555
SVM	0,6791	0,7	0,475	0,475	0,475	0,475	0,475
NaiveBayes	0,3666	0,32	0,36	0,415	0,36	0,655	0,645

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,6	0,595	0,6	0,695	0,58
AdaBoost	0,585	0,77	0,585	0,655	0,665
Random Forest	0,555	0,64	0,535	0,585	0,605
SVM	0,5	0,5	0,5	0,5	0,5
NaiveBayes	0,465	0,53	0,465	0,675	0,665

Tablo 9.2'de görüldüğü gibi, özniteliklerin kullanım sıklıklarının normalleştirilmesi ile bazı sınıflandırmalardan elde edilen sonuçlar yükselirken bazıları düşmektedir.

9.4.3. Kullanılan Özniteliklerin Ağırlıkları

Yukarıda da belirtildiği gibi bu çalışmada kullanılacak özniteliklerin belirlenmesinde öznitelik ağırlıklandırma yöntemlerinden biri olan ve önceki çalışmalarda başarılı çıktılar üreten Gini İndex ağırlıklandırma yöntemi kullanılmıştır. Tablo 9.3'te, doküman temsili vektörlerin üretiminde seçili özniteliklerin ağırlıklarının kullanımı ile elde edilen sınıflandırma sonuçları gösterilmektedir.

Tablo 9.3. Belirlenen özniteliklerin ağırlıkları kullanılarak elde edilen sınıflandırma sonuçları

Öznitelikler Ağırlıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0	0	0,475	0,475	0,475	0,475	0,475
AdaBoost	0,0041	0,005	0,595	0,575	0,595	0,615	0,615
Random Forest	0,2291	0,175	0,46	0,465	0,46	0,505	0,49
SVM	0,4916	0,53	0,475	0,475	0,475	0,475	0,475
NaiveBayes	0,3333	0,25	0,42	0,415	0,42	0,63	0,63

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,5	0,5	0,5	0,5	0,5
AdaBoost	0,635	0,615	0,635	0,62	0,62
Random Forest	0,545	0,575	0,6	0,605	0,57
SVM	0,5	0,5	0,5	0,5	0,5
NaiveBayes	0,455	0,445	0,455	0,65	0,65

Seçili özniteliklerin ağırlıkları kullanılarak oluşturulan temsil vektörleri ile elde edilen sınıflandırma sonuçları Tablo 9.3'te görüldüğü gibi, özniteliklerin kullanım sıklıkları ile elde edilen doğruluk sonuçlarından daha az başarılı sonuçlar üretmiştir.

9.4.4. Kullanılan Özniteliklerin Normalleştirilmiş Ağırlıkları

Gini index kullanılarak belirlenen özniteliklerin ağırlıklarının da hem sınıflama başarısını yükseltecek olması hem de kullanılan verilerin ağırlık bakımından daha normal bir dağılımda temsil ediciliğinin arttırılması amacı ile bir önceki deneylerde kullanılan ağırlıklar normalleştirilmiştir. Buradaki normalleştirme işleminde de min-max normalleştirmesi kullanılmıştır. Tablo 9.4'te, doküman temsili vektörlerin üretiminde, seçili özniteliklerin normalleştirilmiş ağırlıklarının kullanımı ile elde edilen sınıflandırma sonuçları gösterilmektedir.

Tablo 9.4. Belirlenen özniteliklerin normalleştirilmiş ağırlıkları kullanılarak elde edilen sınıflandırma sonuçları

Özniteliklerin Normalleştirilmiş Ağırlıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,7416	0,74	0,29	0,225	0,32	0,615	0,555
AdaBoost	0,0041	0,005	0,595	0,575	0,595	0,615	0,615
Random Forest	0,2041	0,215	0,525	0,465	0,465	0,56	0,515
SVM	0,6833	0,685	0,475	0,475	0,475	0,475	0,475
NaiveBayes	0,3333	0,25	0,42	0,415	0,42	0,63	0,63

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,53	0,49	0,52	0,67	0,64
AdaBoost	0,635	0,615	0,635	0,62	0,62
Random Forest	0,555	0,495	0,625	0,555	0,57
SVM	0,5	0,5	0,5	0,5	0,5
NaiveBayes	0,455	0,445	0,455	0,65	0,65

Normalleştirme işlemi uygulanarak elde edilen sonuçlar birçok yöntemde değişiklik göstermemiş olup bazı sınıflandırma sonuçlarında bir miktar iyileşmelere sebep olmuştur.

9.4.5. Özniteliklerin Kullanım Sıklıkları ile Ağırlıkları

Seçili özniteliklerin kullanım sıklıkları ile ağırlıkları çarpımından elde edilen değerler kullanılarak bu deney kapsamında doküman temsil vektörleri elde edilmiştir. Elde edilen vektörler ile gerçekleştirilen sınıflandırma sonuçları Tablo 9.5'te gösterilmektedir.

Tablo 9.5. Belirlenen özniteliklerin kullanım sıklıkları ile ağırlıkları çarpımı kullanılarak elde edilen sınıflandırma sonuçları

Öznitelikler Kullanım Sıklıkları X Ağırlıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,3166	0,325	0,485	0,445	0,485	0,515	0,475
AdaBoost	0,0875	0,085	0,6	1	0,6	0,64	0,65
Random Forest	0,3208	0,3	0,44	0,705	0,49	0,575	0,54
SVM	0,4666	0,51	0,475	0,475	0,475	0,475	0,475
NaiveBayes	0,5916	0,56	0,35	0,565	0,35	0,655	0,68

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,525	0,495	0,5	0,55	0,5
AdaBoost	0,595	1	0,595	0,65	0,65
Random Forest	0,495	0,755	0,52	0,64	0,62
SVM	0,5	0,5	0,5	0,5	0,5
NaiveBayes	0,45	0,665	0,45	0,66	0,64

9.4.6. Özniteliklerin Kullanım Sıklıkları ile Normalleştirilmiş Ağırlıkları

Seçili özniteliklerin kullanım sıklıkları ile normalleştirilmiş ağırlıkları çarpımından elde edilen değerler kullanılarak bu deney kapsamında doküman temsil vektörleri elde edilmiştir. Elde edilen vektörler ile gerçekleştirilen sınıflandırma sonuçları Tablo 9.6'da gösterilmektedir.

Tablo 9.6. Belirlenen özniteliklerin kullanım sıklıkları ile normalleştirilmiş ağırlıkları çarpımı kullanılarak elde edilen sınıflandırma sonuçları

Özniteliklerin Kullanım Sıklıkları X Normalleştirilmiş Ağırlıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,8166	0,84	0,34	0,34	0,365	0,675	0,68
AdaBoost	0,0541	0,08	0,6	1	0,6	0,64	0,65
Random Forest	0,3291	0,265	0,48	0,885	0,415	0,53	0,565
SVM	0,7458	0,765	0,55	0,605	0,485	0,53	0,555
NaiveBayes	0,6208	0,56	0,35	0,565	0,35	0,655	0,68

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,535	0,59	0,56	0,725	0,715
AdaBoost	0,595	1	0,595	0,65	0,65
Random Forest	0,545	0,92	0,595	0,655	0,62
SVM	0,585	0,66	0,52	0,585	0,645
NaiveBayes	0,45	0,665	0,45	0,66	0,64

9.4.7. Özniteliklerin Normalleştirilmiş Kullanım Sıklıkları ile Ağırlıkları

Seçili özniteliklerin normalleştirilmiş kullanım sıklıkları ile ağırlıkları çarpımından elde edilen değerler kullanılarak bu deney kapsamında doküman temsil vektörleri elde edilmiştir. Elde edilen vektörler ile gerçekleştirilen sınıflandırma sonuçları Tablo 9.7'de gösterilmektedir.

Tablo 9.7. Belirlenen özniteliklerin normalleştirilmiş kullanım sıklıkları ile ağırlıkları çarpımı kullanılarak elde edilen sınıflandırma sonuçları

Özniteliklerin Normalleştirilmiş Kullanım Sıklıkları X Ağırlıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0	0	0,475	0,475	0,475	0,475	0,475
AdaBoost	0	0,065	0,6	0,665	0,6	0,6	0,57
Random Forest	0,3083	0,335	0,4	0,515	0,41	0,47	0,52
SVM	0,0125	0,01	0,475	0,475	0,475	0,475	0,475
NaiveBayes	0,3416	0,325	0,355	0,445	0,355	0,655	0,68

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,5	0,5	0,5	0,5	0,5
AdaBoost	0,615	0,765	0,615	0,625	0,575
Random Forest	0,55	0,625	0,575	0,545	0,535
SVM	0,5	0,5	0,5	0,5	0,5
NaiveBayes	0,465	0,535	0,465	0,67	0,625

9.4.8. Özniteliklerin Normalleştirilmiş Kullanım Sıklıkları ile Normalleştirilmiş Ağırlıkları

Seçili özniteliklerin normalleştirilmiş kullanım sıklıkları ile normalleştirilmiş ağırlıkları çarpımından elde edilen değerler kullanılarak bu deney kapsamında doküman temsil vektörleri elde edilmiştir. Elde edilen vektörler ile gerçekleştirilen sınıflandırma sonuçları Tablo 9.8'de gösterilmektedir.

Tablo 9.8. Belirlenen özniteliklerin normalleştirilmiş kullanım sıklıkları ile normalleştirilmiş ağırlıkları çarpımı kullanılarak elde edilen sınıflandırma sonuçları

Özniteliklerin Normalleştirilmiş Kullanım Sıklıkları X Normalleştirilmiş Ağırlıkları							
Sınıflandırma Yöntemi	Çoklu Sınıflandırma		İkili Sınıflandırma				
	120 sınıf	100 sınıf	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,5583	0,55	0,485	0,415	0,485	0,545	0,48
AdaBoost	0	0,065	0,5	0,8	0,5	0,61	0,635
Random Forest	0,2916	0,305	0,41	0,615	0,435	0,485	0,565
SVM	0,4541	0,495	0,475	0,475	0,475	0,475	0,475
NaiveBayes	0,3416	0,33	0,355	0,445	0,355	0,655	0,68

Sınıflandırma Yöntemi	Validasyon				
	toplam	fark	ortalama	çarpkok	çarpım
Logistic Regression	0,55	0,45	0,5	0,61	0,51
AdaBoost	0,585	0,77	0,585	0,655	0,615
Random Forest	0,52	0,73	0,57	0,585	0,58
SVM	0,5	0,5	0,5	0,5	0,5
NaiveBayes	0,465	0,54	0,465	0,67	0,63

9.4.9. Kelime Bulutu

Öznitelik kullanım sıklıklarının ön planda tutulduğu bu bölümde 4-ön-kök adı verilen, 4 karaktere kadar olan tüm kelimeler öznitelik kümesine eklenmiştir. Ayrıca 4'ten fazla karaktere sahip kelimelerin ilk 4 karakteri, ön kök olarak kelimeleri temsilen yine bu öznitelik kümesine eklenmiştir. Birçok doğal dil işleme ve yazar analizi çalışmasında olduğu gibi bu çok fazla sayıda olan özniteliklerin belli bir orana indirilmesi gerekmektedir. Bu işlem için yaz öznitelik indirgeme işlemleri yapılır veya öznitelik seçme işlemi yapılır. Bu tez kapsamında kullanılan özniteliklerin bireysel olarak ağırlığı – önemi de dikkate alındığından öznitelik seçme işlemi kullanılmıştır. Bu amaç ile kullanılan gini index öznitelik ağırlıklandırma yöntemi ile veri kümesinde bulunan en yüksek ağırlıklı 1000 öznitelik seçilmiştir. Seçili bu özniteliklerin kullanım sıklıklarına göre boyutlandırıldığı kelime bulutu Şekil 9.12'de gösterilmektedir.

Şekil 9.12. Seçili özniteliklerin kelime bulutu gösterimi

Kullanılan veri kümesinde seçili öznitelikler içinde en çok kullanılan öznitelikler; anla, insa(n), düşü(n), konu, zama(n), … gibi kelimeler olmuştur.

9.5. Çıkarımlar

Yazar Doğrulama probleminin çözümü için önerilen yöntem içerisinde, “token 4-gram” olarak da bilinen kelime 4-ön kök öznitelikleri kullanılarak elde edilen en iyi sonuçlar, dokümanların fark, çarpım ve çarpımlarının karekökleri alınarak birleştirilmesi ile elde edilmiştir.

Problemin çözümüne hizmet eden en anlamlı öznitelik setleri; özniteliklerin frekansları ve özniteliklerin frekanslarının ağırlıkları ile çarpımıdır. Elde edilen sonuçların başarısı anlamlı fakat yetersizdir.

İleriki çalışmalarda, seçili veri kümesi ve model için başarıyı yükseltecek ek öznitelikler kullanılması, elde edilen en başarılı yapının Literatürdeki farklı dillerde yayınlanan veri kümeleri için sonuçlarının alınması ve yayın yazımı planlanmaktadır.

10. YAZAR ANALİZİNİN ADLİ BİLİŞİMİNDE ÜSLUPSAL ÖZNİTELİKLERİN DERİN KOMBİNASYONU

Adli bilişimde Yazar Analizi (YA), bir yazara ait yazılardan o yazar ile ilgili bilgi çıkarmayı amaçlayan bir süreçtir. Adli YA çalışmalarına tehdit ve şantaj içerikli iletilerle dijital medya kullanıcılarının güvenliğini tehlikeye atan anonim yazarların karakteristiğinin tespit edilmesi için ihtiyaç duyulmaktadır. İki kısa anonim yazının aynı yazar tarafından yazılıp yazılmadığının analizi için biz bu çalışmada farklı kategorideki üslupsal özellikleri farklı süreçlerde birleştiriyoruz. Önceki YA çalışmalarının çoğunda farklı kategorideki üslupsal özellikler sunulan çözümün başında bir ön işlem olarak birleştirilmektedir. Öğrenme süreci boyunca kategori tabanlı bir işlem yapılmamaktadır; tüm kategoriler eşit olarak değerlendirilmektedir. Fakat önerdiğimiz yaklaşım her öznitelik kategorisine kendi nitel ve nicel özelliklerine göre ayrı öğrenme süreçlerine sahiptir ve karar aşamasında bu süreçleri Derin Sinir Ağları Kombinasyonu (C-DSA) kullanarak birleştirmektedir. Önerdiğimiz yaklaşımın Yazar Doğrulama (YD) performansını değerlendirmek için bu çalışmada, kullanılan her üslupsal kategori için probleme özgü Derin Sinir Ağları (DSA) tasarlanmış ve uygulanmıştır. Çalışmanın deneyleri iki farklı umumi İngilizce veri kümesi üzerinde yürütülmüştür. Elde edilen sonuçlar önerilen yaklaşımın sunulan çözümlerin genelleştirme yeteneğini ve güvenilirliğini büyük ölçüde arttırdığını ve hatta tekil DSA'lardan daha yüksek doğrulukta sonuçlar ürettiğini göstermiştir.

10.1. Arka Plan Bilgisi

Dijital ortamlarda metinsel verilerin üretimi gün be gün sürekli ve üstel olarak artmaktadır. Bu ortamalarda anonim veri üretmenin kolaylığı, takma isimler kullanarak birilerini tehdit etmek veya birilerine şantaj yapmak gibi anonim siber suçların işlenmesini de kolaylaştırmaktadır. Bu sebepler dolayısı ile yazı üslubunun analiz edilmesi ile dijital dokümanlardan yazarları ile ilgili bilgi elde edebilmek büyük önem taşımaktadır. Bir yazara ait yazma üslubu hassas (bilişsel veya davranışsal) bir biyometridir [1, 2] ve üslupsal özniteliklerin yardımı ile bu biyometriler belirlenebilmektedir. Bu öznitelikler dokümanları yazarının üslupsal temsiline dönüştürmede kullanılmaktadır. Bunlar; sözcüksel, yapısal, alana özgü, sözdizimsel veya anlamsal özniteliklerdir [5]. Yaklaşık 200 yıldır bu öznitelikler birçok çalışmada yazarların kişisel karakteristiklerini elde etmek için kullanılmaktadır [4]. Bu çalışmada, bu özniteliklerin yazar

analizi yöntemlerinin genelleştirme yeteneğini arttırmak için nasıl kullanılması gerektiği üzerine araştırma yapılmıştır.

Yazarların yazma üslubunu elde etmek için birçok üslupsal teknik geliştirilmiştir. Araştırmacıların ilgilendiği problemler bakımından Yazar Analizi (YA) beş önemli alt bölüme ayrılmıştır: Yazar tanıma, yazar doğrulama, yazar profil çıkarımı, üslupsal ölçek ve ters üslupsal ölçek [5]. Bu alanda araştırmacılar çoğunlukla yazar tanıma ve yazar doğrulama çalışmalarına yoğunlaşmış olsalar da [37], bu alandaki problemler birbiriyle ilişkilidir ve bir alt alan için önerilecek optimal bir çözüm barındırdığı üslupsal altyapı sebebiyle diğer alanlarda da faydalı olacaktır. Yazar tanımlama çalışmaları sorgulanan bir dokümanı bir yazara atama üzerinedir. Yazar tanımlama çalışmalarının yazar doğrulama çalışmalarından farkı kullanılan aday yazar kümesidir. Yazar tanımlama çalışmalarında sorgulanan dokümanın yazarının aday yazarlar kümesinde olduğu garantilenmiştir, bu sebeple bu çalışmalar kapalı küme atama problemi olarak ele alınmaktadır [10]. Fakat yazar doğrulama çalışmalarında sorgulanan dokümana ait bir aday yazarlar kümesi veya dokümanın yazarına ait başka bir arka plan bilgisi bulunmamaktadır.

Yazar Doğrulama (YD) Dijital Metinlerin Adli Bilişimi ve YA çalışmalarının ortak alanlarından biridir [20, 95]. YD çalışmalarının amacı, verilen bir grup dokümanın yazarının aynı zamanda sorgulanan bir dokümanın da yazarı olup olmadığına karar vermektir (bir-e-çok) [19, 96]. Bu çalışmaların en zorlu yapısında ki bizim bu çalışmada da ele aldığımız yapı, YD verilen iki kısa anonim dokümanın aynı yazar tarafından yazılıp yazılmadığına karar vermeyi amaçlar (bir-e-bir) [22, 97]. Bu yapıda, sorgulanan bir metnin yazarı hem yazarı bilinen hem de yazarı bilinmeyen metinlere karşı doğrulanabilmektedir. Bu çalışmada, bir-e-bir YD problemi olarak, verilen iki dokümanın üslupsal farkının tek bir yazarı temsil edip etmediği üzerine çalışmalar yapılmıştır. Bu çalışmalar ile amacımız, bazı tehdit mesajlarının aynı kişi tarafından atılıp atılmadığı veya bir ürün için önyargılı yorumların aynı kişi tarafından yapılıp yapılmadığı gibi sorunların çözümüne katkıda bulunmaktır. YD zorlu bir problem olduğundan, birçok çalışmada farklı üslupsal kategorilerdeki öznitelikleri birlikte barındıran modellerin, tek bir kategorideki özellikleri barındıranlardan daha yüksek doğrulukta sonuçlar ürettiği gösterilmiştir [2, 18, 57]. Bu çalışmaların çoğunda farklı kategorideki öznitelikler öğrenme sürecinden önce yalın haliyle birleştirilmiştir [18, 98]. Farklı olarak biz bu çalışmada kullandığımız her üslupsal öznitelik kategorisi için bu kategorilerin nitel ve nicel özelliklerine göre ayrı öğrenme süreçleri geliştirdik ve bu süreçleri çözümün karar aşamasında bir araya

getirdik. Böylece, hem öğrenme modelinin genelleştirme yeteneği hem de kullanılan öznitelik kategorilerinin temsil ediciliği arttırılmış oldu.

YA problemleri sınıflandırma problemleridir ve ele alınan problemin yapısına göre tek sınıflı, iki sınıflı veya çok sınıflı olarak değerlendirilmektedir. YA problemlerinin çözümünde; ağaç-tabanlı öğrenmeler, lojistik regresyon, Destek Vektör Makineleri, Bayes sınıflandırıcılar gibi birçok denetimli öğrenme teknikleri kullanılmaktadır [5]. Denetimsiz öğrenme veya kümeleme bakış açısı ile de K-ortalamalar, Beklenti Maksimizasyonu gibi teknikler genellikle üslupsal benzerliklerine göre bir grup içindeki benzer dokümanların bulunmasında kullanılmaktadır [95, 99]. Ayrıca, bazı öznitelik kategorilerinin çok boyutlu olması sebebi ile Temel Bileşenler Analizi gibi öznitelik seçimi veya boyut indirgeme teknikleri de YA çalışmalarında sıklıkla kullanılmaktadır [100]. Bu çalışmada Yapay Sinir Ağları (YSA) kullanılarak denetimli öğrenme yaklaşımı uygulanmıştır. YSA'lar birden çok gizli katman ve farklı mimariler ile Derin Sinir Ağları (DSA) modeli olarak ele alınmaktadır. Sinir ağlarının genelleştirme yeteneğini arttırmak için birçok sinir ağının bileşimi birçok çalışmada çok umut verici sonuçlar üretmiştir [101]. Etkili bir sinir ağları birleşimi tasarlamak tek bir sinir ağı tasarlamaktan daha karmaşık olmasına rağmen bir birleşik tasarım bir tekli tasarıma göre daha güvenilir ve daha genelleştirilmiş bir çözüm sunmaktadır. Bu sebeple, AA problemlerinin çözümünde tek bir en iyi DSA mimarisi kullanmak yerine, bu çalışmada farklı DSA'ların birleşiminden oluşan yeni bir yaklaşımın (C-DSA) kullanılması önerilmiştir. Farklı üslupsal kategoriler için tasarlanan DSA mimarileri, önerilen C-DSA mimarisinde karar aşamasında birleştirilmektedir. Bu çalışmada, hem tekil DSA mimarilerinin hem de önerilen C-DSA mimarisinin YD performansları deneysel çalışmalar ile gösterilmiştir. Böylece, hem farklı kategorilerdeki özniteliklerin YA'da kullanımının etkisi hem de farklı çözüm modellerinin YA'da bir arada kullanımının etkisi uygulamalarla gösterilmiştir.

İki umumi İngilizce veri kümesi hem tekil DSA'ların hem de önerilen C-DSA yaklaşımının performans testinde kullanılmıştır. Deneysel çalışmaların sonuçları önerilen C-DSA yaklaşımının tekli DSA'lara göre daha güvenilir, daha genelleştirilmiş ve daha doğruluğu yüksek çözüm sunduğunu göstermektedir.

10.2. Problem Tanımı

Adli bilişimde YA uzun yıllardır intihal tespiti veya tartışmalı metinlerin yazarını belirleme gibi birçok sorunun çözümünde çalışılmaktadır [3]. Metinlerin dijitalleştirilmesiyle bu çalışmalar,

dijital metinlerin adli bilişiminde YA olarak gelişmiştir [102]. 2000 yılında Stamatatos ve ark. belirli bir kişinin sorgulanan metnin yazarı olduğu hipotezinin onaylanması (veya reddedilmesi) gerekliliğiyle ilgilenmiştir [103]. Çalışma, yazarlık doğrulama sorununun ele alındığı ilk çalışmadır. Bu durumda, bir yazar tarafından yazılan metinler ve bu özel yazar tarafından yazılıp yazılmadığı sorgulanan harici bir metin vardır. Öte yandan, 2014 yılında Koppel ve Winter [23] YD problemini bir yazarın bilinen bir belgesi olarak görece kısa bir doküman olarak ele aldılar. Sorgulanan belgenin yazarlığını doğrulamak için bilinen tek bir kısa belge varsa, sorun daha zor hale gelir. İki anonim metnin yazarlık doğrulaması (bire bir karşılaştırma) olarak kabul edilen sorun, birçok yazarlık analizi çalışmasının temelini oluşturmaktadır [2]. Bu tür doğrulama problemlerinin zorlukları arasında, bilinen ve sorgulanan belgelerin uzunluğu [23], en iyi ayırt edici öznitelik setlerinin seçimi [18] ve başarılı bir yanıt işlevi bulunması [104] bulunmaktadır.

YA'da yapılan çalışmalar, en iyi ayırt edici özniteliklerin tek bir üslupsal kategoriden olmadığını göstermiştir. Farklı kategorilerdeki üslupsal özniteliklerin birlikte kullanılması birçok önerilen yöntemin başarısını arttırmaktadır [2, 18, 57, 105]. Bu alanda yapılan çalışmaların çoğu ya çözümde tek bir üslup kategorisi kullanmıştır [106–108] ya da farklı öznitelik kategorileri için tek bir öğrenme modeli kullanmıştır [1, 18, 109, 110]. Bu çalışmada literatürden farklı olarak, üslupsal öznitelikleri öğrenme sürecinde nitel ve nicel özelliklerine göre değerlendiren yeni bir yaklaşım öneriyoruz. Kullanılan her üslup kategorisi için ayrı bir öğrenme süreci geliştiriyor ve bu aşamaları karar aşamasında bir C-DSA mimarisi yardımıyla birleştiriyoruz. Öğrenme modelinin sağlamlığını / güvenilirliğini ve kullanılan özniteliklerin temsil edilebilirliğini artırmak için her öznitelik kategorisi için farklı bir süreç kullanıyoruz. Önerilen yaklaşım, tek bir üslup kategorisini veya farklı öznitelik kategorileri için tek bir modeli kullanan yaklaşımdan daha genel bir çözüm sunmaktadır.

10.3. YD Örneklerinin Üretimi

10.3.1. Veri Kümesi Düzenlemeleri

Önerilen yaklaşımın performansını test etmek için iki popüler veri kümesi, İngilizce Blog Külliyatı [15] ve PAN-2015-İngilizce veri kümesi [111] kullanılmıştır. İngilizce Blog Külliyatı, 19320 blog yazarı tarafından üretilen gönderilerden oluşmaktadır. Bu veri kümesinden, 1000 yazardan 500 kelime içeren 20 adet rastgele metin alarak yeni bir veri kümesi elde ettik. Aynı yazardan alındığı anlamına gelen 20 pozitif belge çifti, her bir yazardan rastgele

alınmıştır. Benzer şekilde, farklı yazarlardan alındıkları anlamına gelen 20 negatif belge çifti, her yazar için diğer yazarların rastgele seçilmiş metinleri kullanılarak elde edilmiştir. Böylece, 20.000 pozitif ve 20.000 negatif örnek içeren 40.000 örnek ile yeni bir veri kümesi elde ettik.

PAN-2015-İngilizce veri kümesi, konuşmacı isimleri, karakterler vb. listesi hariç olmak üzere oyunlardan elde edilen bir dizi iletişim yazışmalarına sahiptir [111]. Bu veri kümesinde eğitim kümesinde 100 belge çifti ve test kümesinde 500 belge çifti vardır. Derin mimariler küçük bir eğitim kümesi ile iyi performans göstermeyeceğinden, eğitim ve test kümelerini deneylerde bir araya getirdik.

10.3.2. Üslupsal Öznitelikler

Literatürde, çeşitli metin analiz yöntemlerinde binden fazla üslupasal öznitelik kullanılmıştır. Bu öznitelikler sözcüksel, sözdizimsel, anlamsal, yapısal ve içeriğe özgü olarak kategorilere ayrılmaktadır [30, 34]. Yazar analizi çalışmaları arasında hangi kategorilerin veya özniteliklerin en iyi sonuçları verdiğine ilişkin ortak bir karar yoktur. Miktar ve kalite bakımından çok farklı olmalarına rağmen, çalışmaların çoğu elde edilecek başarıyı artırmak için bazı kategorileri birarada kullanmıştır [2, 18, 105].

Bu çalışmanın deneylerinde üç farklı öznitelik kategorisi kullanılmıştır. Sözcüksel öznitelikler (c1) olarak, beş karaktere kadar olan tüm kelimeleri ve daha uzun kelimeler için ilk beş karakteri alan kelime 5 öneklerinin frekanslarını kullandık. Noktalama frekansları (c2) sözdizimsel öznitelikler olarak kullanılmıştır ve yapısal öznitelik kategorisi (c3) için yapısal özellikler (kelime tabanlı paragraf uzunluğu, ortalama kelime uzunluğu, metin başına karakter sayısı, metin başına noktalama sayısı) kullanılmıştır. Bu öznitelik kümelerini iki farklı yapıda kullandık. İlk yapıda, öğrenme sürecinden önce, belgelerin vektör temsilini almak için tüm özellikleri birleştirdik (dc). Bu yapıda her belge, farklı üslup kategorilerinden farklı öznitelikler içeren tek bir vektörle temsil edilir. İkinci formda, her belge için ayrı ayrı üç farklı vektör temsili kullandık (dc1, dc2 ve dc3). Her temsil belirli bir öznitelik kategorisinin özelliklerini taşır ve öğrenme aşamasında ayrı bir süreci vardır. Bu dört vektörün temsili aşağıda gösterilmiştir.

$$d(c1) = [c1_1, c1_2, c1_3, \ldots, c1_n]$$
$$d(c2) = [\, c2_1, c2_2, c2_3, \ldots, c2_m]$$
$$d(c3) = [\, c3_1, c3_2, c3_3, \ldots, c3_k]$$
$$dc = [c1_1, c1_2, c1_3, \ldots, c1_n, c2_1, c2_2, c2_3, \ldots, c2_m, c3_1, c3_2, c3_3, \ldots, c3_k]$$

Yukarıdaki vektörel gösterimlerde c1, c2 ve c3 kullanılan özniteliklerin kategorilerini temsil ederken, $c2_m$ ise c2 kategorisindeki m. özniteliğin değerini temsil etmektedir.

10.3.3. Doküman Çiftlerinin Değerlendirilmesi

Belge çiftlerini tek bir gösterimde değerlendirebilmek için, vektör uzayındaki çiftlerin mutlak farkını kullandık. Bunu yaparak, değerlendirilen çiftlerin aynı sayıda boyutuna sahip yeni bir vektör temsili elde ettik. Bu gösterimi kullanarak, bir belge çiftinin mutlak farkının tek bir yazarın stilini temsil edip etmediğini sorguladık. Söz konusu belge çiftlerini, [23] çalışmasındaki denetimli öğrenme yönteminde uygulandığı şekilde, pozitif ve negatif AV örneklerinin tek bir temsilini üretmek için değerlendirdik. Her YD örneği aşağıdaki temsil kullanılarak belge çiftlerinden (X ve Y) elde edildi.

$$C(X, Y) = [\ \| X_1 - Y_1 \|, \| X_2 - Y_2 \|, \| X_3 - Y_3 \|, \ldots, \| X_n - Y_n \|\ ]$$

X ve Y iki belgenin öznitelik vektörleri olsun ve X_i ile Y_i ilgili özniteliklerin değeri olsun. X ve Y aynı yazarın çiftiyse, C vektörü pozitif, farklı yazarlara ait doküman çifti ise negatif olarak etiketlendi. Bu etiketleme işlemini, kullanılan her iki veri kümesindeki metin çiftlerine uygulayarak, bu veri kümeleri pozitif ve negatif AV örnekleri içeren iki sınıflı veri kümeleri haline dönüştürülmüştür.

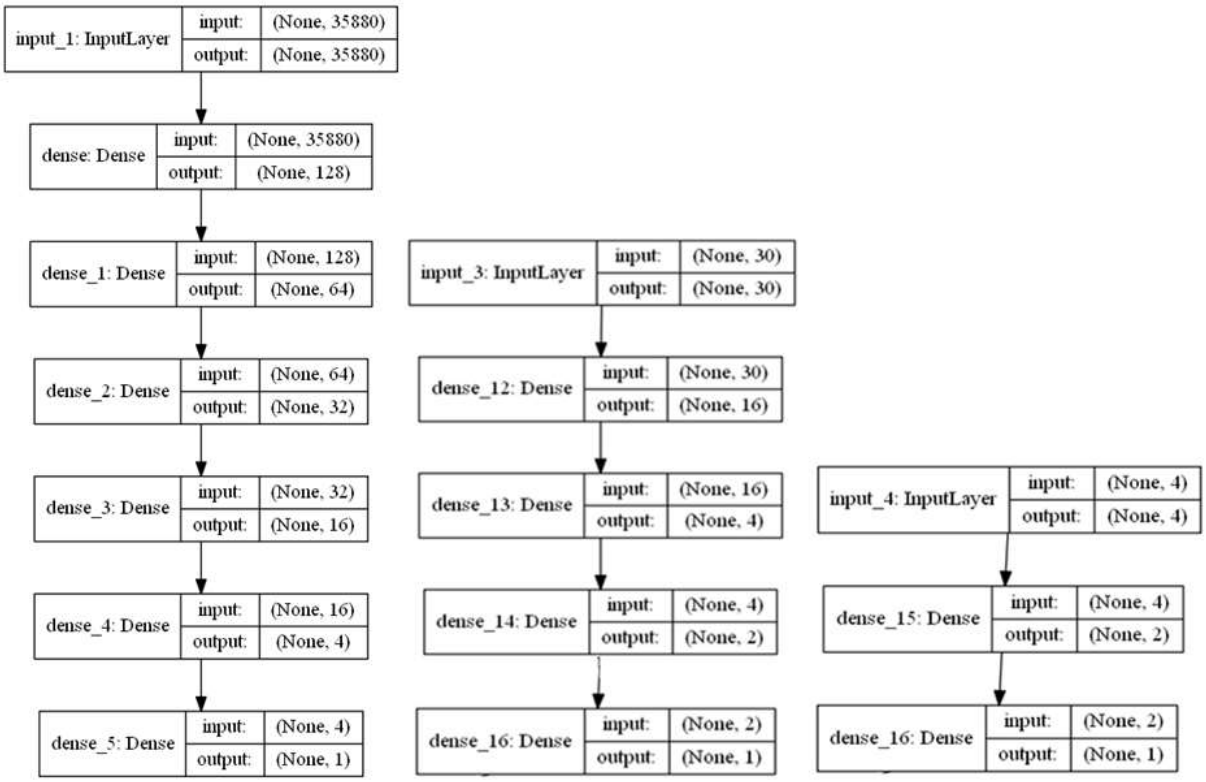
10.4. Üslupsal Özniteliklerin Derin Birleşimi

Bu çalışmada, iki anonim belgenin yazarlığını doğrulama probleminin çözümü için C-DSA kullanan yeni bir yaklaşım uyguladık. Her üslup kategorisi ve tüm kategorilerin birleştirilmiş formu için tek bir DSA tasarladık. Ardından, bir C-DSA mimarisinde, tek DSA'ları derin bir mimarinin karar aşamasında birleştirdik. Mimarilerin detayları aşağıda verilmiştir.

10.4.1. DSA Mimarileri

Hesaplamalı teknikler hızla gelişirken, derin mimarileri olan DSA'lar denetimli öğrenme yöntemleri için güçlü yapılar sunmaktadır [112]. Bu çalışmada ele alınan YD probleminin çözümüne uygun derin mimariler ile Sinir Ağları (DSA) ile araştırılmıştır. Kullanılan problem türüne ve veri setine göre, DSA yapıları içerdikleri katman ve sinir hücresi sayısı veya aktivasyon fonksiyonlarının türü gibi her bir katmanda kullanılan hiperparametreler açısından birçok şekil alabilir. Her soruna ve her veri setine uygun mimariyi üretmek karmaşık bir süreçtir. Özellikle üretilen yöntemin kullanılan veriyi ezberlemesini önlemek için, her bir

soruna ve her veri setine özgü uygun mimariyi ve parametreleri belirlemek gerekir. Önerilen DSA tasarımları Şekil 10.1'de gösterilmektedir.



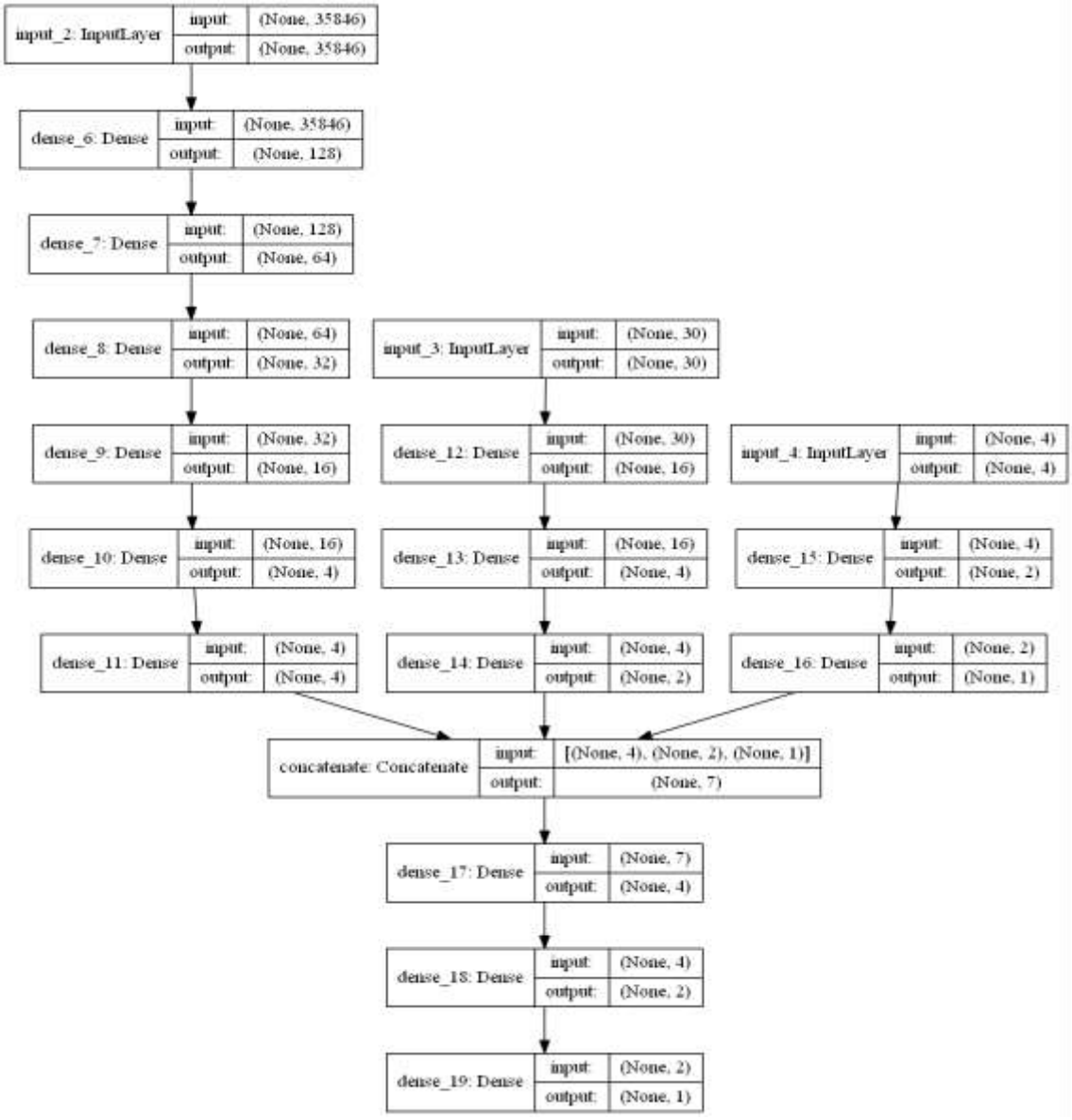
Şekil 10.1. Farklı üslupsal öznitelikler için tasarlanan DSA mimarileri

Şekil 10.1'de gösterilen DSA mimarileri, blog veri kümesi için üretilen YD modellerinin temsilidir. Şekil 10.l'in sol tarafında gösterilen ilk mimari, hem c1 (sözcüksel öznitelikler) hem de dc (tüm özelliklerin birleştirilmesi) örnekleri için kullanılmıştır. Şekil 10.l'in ortasında gösterilen ikinci mimari, c2 (sözdizimsel öznitelikler) örnekleri için kullanılmıştır. Şekil 10.l'in sağ tarafında gösterilen son mimari c3 (yapısal öznitelikler) örnekleri için kullanılmıştır. Blog veri kümesinin ds biçimindeki girdi örnekleri 35880 boyutlu vektörlerdir. Sonuncusu hariç tüm yoğun katmanlarda, selu (Scaled Exponential Linear Unit) aktivasyon fonksiyonu ve çekirdek başlatıcısı için lecun_normal dağılımı kullandık. Son yoğun tabakada, numunelerin sınıfını tahmin etmek için sigmoid aktivasyon fonksiyonunu kullandık. Tüm DSA mimarilerinde, katman sayısını artırdığımızda veya her katmandaki sinir hücresi sayısını artırdığımızda, yöntemin ezberleme eğilimi de paralel olarak artmaktaydı. Bu sebeple, Blog veri kümesi 40.000 örnek içermesine rağmen, daha karmaşık bir mimari oluşturmak gerekmemiştir. Diğer taraftan, PAN veri kümesi 600 örnek içermesine rağmen, aynı DSA mimarisinde dc formuyla başarılı

sonuçlar vermiştir. PAN örnekleri 5551 boyutlu vektörlerdir ve aynı mimarileri kullanan bir YD modelinin üretiminde de kullanılmıştır.

10.4.2. C-DSA Yaklaşımı

Farklı üslupsal öznitelik kategorilerinin temsil edilebilirliğini ve önerilen DSA'larin genelleme kabiliyetini arttırmak için, her katmanda uygun fonsiyonlara ve sinir hücrelerine sahip bir C-DSA mimarisi tasarladık. Önerilen C-DSA mimarisinin 3 fazı vardır; öğrenme veya boyut azaltma, birleştirme ve karar verme. YD için önerilen C-DSA mimarisi Şekil 10.2'de gösterilmiştir.



Şekil 10.2. YD problemi için tasarlanmış, önerilen C-DSA mimarisi

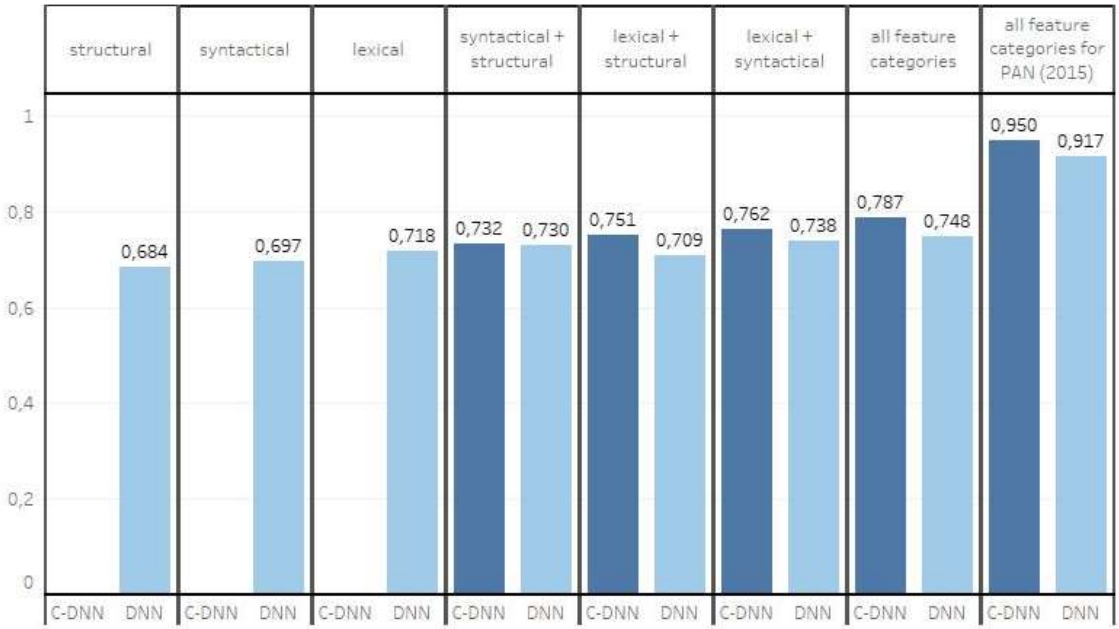
Şekil 10.2'de gösterilen C-DSA mimarisi, blog veri kümesi için tasarlanmış YD modelinin temsilidir. C-DNN mimarisinin ilk aşaması, bir örneğin üç farklı vektör gösteriminden alınan üç farklı girdi içerir. Temsiller kullanılan üslupsal özniteliklerin kategorilerine dayanmaktadır. Bu aşama, YD probleminin çözümü için her bir girdi kategorisinin kodlanmış veya indirgenmiş biçimlerini öğrenir. Niceliksel ve niteliksel özelliklerine göre kodlanan girdiler ikinci aşamada birleştirilir.

Sözcüksel öznitelikler YA çalışmalarında en çok kullanılan özniteliklerdir [19, 23], ve literatüre göre, bu öznitelikler tek başlarına kullanıldıklarında sözdizimsel özelliklerden daha etkilidir [18]. Bu nedenle, birinci kategorinin kodlanmış formu için 4 sinir hücresi kullandık. Öte yandan, YA çalışmalarında farklı sözdizimsel özniteliklerle de başarılı çalışmalar yürütülmüştür [107, 109]. Bu çalışmada kullandığımız sözdizimsel özellikler nicelik olarak sözcüksel özelliklerden daha kısadır ve literatürde daha az kullanılmaktadır; bu sebeple, ikinci kategorinin kodlanmış formu için 2 sinir hücresi kullandık. Üçüncü kategori sadece 4 özniteliğe sahiptir ve genellikle ana öznitelik kümesini desteklemek için kullanılır. Bu sebeple, son kategorinin kodlanmış formu için sadece 1 sinir hücresi kullandık. Birleştirme katmanına kadar olan tüm yoğun katmanlarda, DSA katmanlarının özellikleri korunmuştur.

Son karar aşamasında farklı kategorilerin birleşiminin gücünü arttırmak için mimariye iki tane daha yoğun katman eklenmiştir. Bu katmanlarda tanh (Hiperbolik tanjant) aktivasyon fonksiyonu ve çekirdek başlatıcısı olarak düzgün dağılım kullandık. Nihai karar sigmoid aktivasyon fonksiyonuna sahip son katmandan elde edilmektedir. PAN veri kümesinden elde edilen örnekler için de benzer ayarlamalar kullanılmıştır.

10.5. Deneysel Sonuçlar

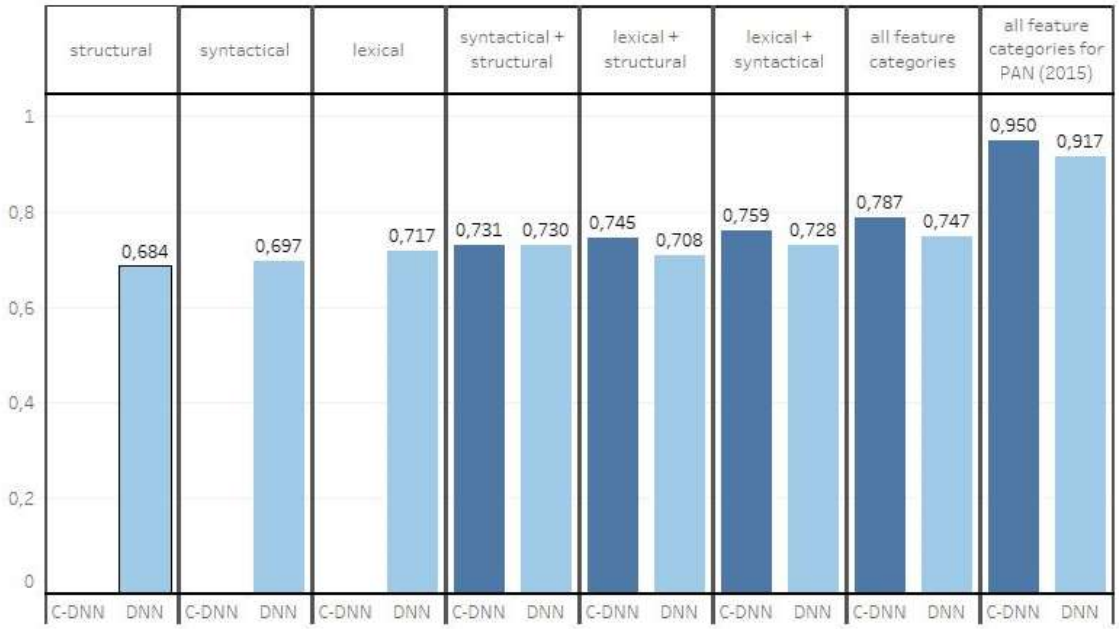
Deneysel çalışmalarda, önerilen DSA ve C-DSA yaklaşımları YD performanslarını değerlendirmek için önceden belirtilen iki veri seti kullanılarak test edilmiştir. Tüm deneyler Python 3.7'de Keras kullanılarak Tensorflow 2.0 üzerinde yapılmıştır. Örnekleri 100 büyüklüğünde gruplar halinde (batch size) gruplandırdık ve tüm deneylerde 50 devir (epoch) kullandık. Veri kümeleri pozitif ve negatif örnek sayısı açısından dengelendiğinden, değerlendirmelerde doğruluk ve f-ölçüsü kullanmayı tercih ettik. Ele alınan ölçeklere göre, blog ve PAN veri kümelerinin DSA ve C-DSA mimarilerinden elde edilen YD doğrulukları ve karşılaştırmaları Şekil 10.3'te gösterilmektedir.



Şekil 10.3. DSA ve C-DSA mimarilerinin YD performanslarının elde edilen doğruluk değerleri bakımından karşılaştırması

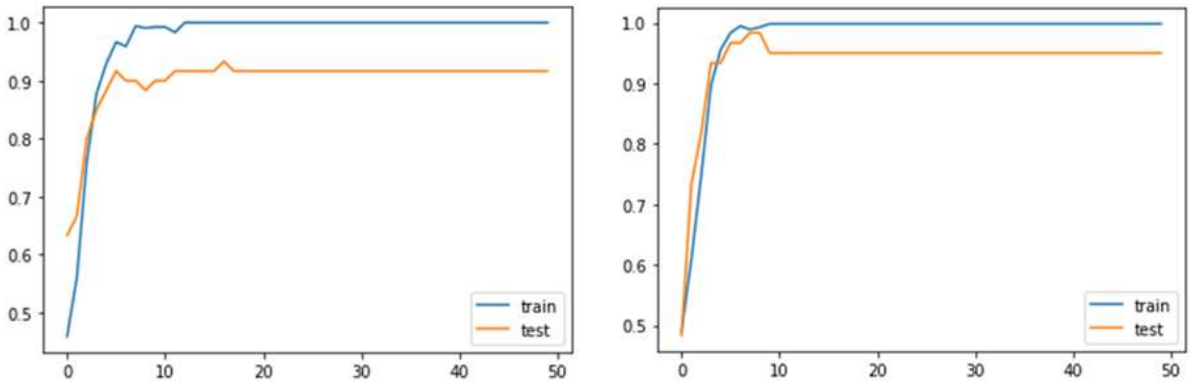
Şekil 10.3'te gösterilen sonuçlar, son sütun hariç, blog veri kümesinin deneylerinden elde edilmiştir. Tek bir kategorideki öznitelikler kullanılarak, elde edilen sonuçlar arasında büyük bir fark olmamakla birlikte, doğruluğu en yüksek sonuç sözcüksel özniteliklerden elde edilmiştir. Öte yandan, önerilen C-DSA mimarileri kullanılan üslup kategorisinin her kombinasyonunda DSA mimarilerinden daha yüksek doğrulukta sonuçlar üretmiştir. DSA ve C-DSA mimarileri ile tüm öznitelik kategorileri kullanılarak PAN veri kümesinden de son sütunda görüldüğü üzere, benzer sonuçlar elde edilmiştir.

Blog ve PAN veri kümelerinin DSA ve C-DSA mimarilerinden elde edilen YD f-ölçümleri ve karşılaştırmaları Şekil 10.4'te gösterilmektedir.



Şekil 10.4. DSA ve C-DSA mimarilerinin YD performanslarının elde edilen f-ölçeği değerleri bakımından karşılaştırması

Deneylerde kullanılan veri kümeleri dengeli kümeler olmasına rağmen, önerilen yaklaşımın sağlamlığını / güvenilirliğini göstermek için f-ölçümleri de hesaplanmıştır. Şekil 10.4'te gösterildiği gibi, f-ölçümleriyle elde edilen sonuçlar, doğruluk ölçümleriyle elde edilen sonuçlarla hemen hemen aynıdır. F- ölçümleri ile alınan sonuçlara göre, önerilen C-DSA mimarileri kullanılan üslup kategorisinin her kombinasyonunda DSA mimarilerinden daha sağlam sonuçlar üretmiştir. Sonuçları Şekil 10.3 ve Şekil 10.4'te verilen deneylerde, 10 kat çapraz doğrulama uygulanmıştır. Şekillerde, tüm sütunlar sonuncusu hariç blog veri kümesinin YD performanslarını gösterir. DSA mimarisi kullanılarak elde edilen doğruluk değerleri dikkate alındığında, tüm öznitelikleri birlikte değerlendiren deneylerden yüksek doğruluk elde edilmiştir. Diğer yandan, aynı özniteliklerden elde edilen doğruluğu arttırmanın yanı sıra, C-DSA mimarisi kullanılarak çözümün genelleştirilmesi de artmıştır. Şekillerin son sütunları, kullandığımız PAN veri kümesinin YD performanslarını göstermektedir. C-DSA mimarisinin genelleme yeteneğini göstermek için, Şekil 10.5'te PAN veri kümesinin eğitim ve test kümelerinin doğruluk performansları gösterilmektedir.



Şekil 10.5. DSA (solda) ve C-DSA (sağda) mimarilerinde tüm öznitelik kategorilerinin kullanımı ile PAN veri kümelerinden elde edilen doğruluk değerleri

Kullandığımız PAN veri kümesinin YD performansı, blog veri kümesiyle kullanılan aynı mimarilerle test edilmiştir. Önerilen mimarinin farklı bir veri kümesinde başarısını göstermek için DSA ve C-DSA mimarileri üzerindeki farklı kategorilerden tüm özniteliklerle PAN deneyleri gerçekleştirilmiştir. Şekil 5'te gösterildiği gibi, test ve eğitim doğrulukları arasındaki fark, C-DSA mimarisinde DSA'dan daha küçüktür. Bu fark, C-DSA yaklaşımının tek bir DSA yaklaşımından daha sağlam ve genelleştirilmiş bir çözüm sağladığını göstermektedir.

10.6. Sonuçlar ve İleriki Çalışmalar

Üslupsal öznitelikler, dokümanlardan bilgi elde etmek için kullanılan özniteliklerdir. Dijital metinlerin adli bilişiminde, dijital belgelerin yazarı hakkında bilgi edinmek önemlidir. Bu adli tabanlı yazarlık analizi çalışmalarının gerekliliği, isimsiz yazarların dijital ortamda artmasıyla artmıştır. Yazarlık Analizi çalışmalarının çoğu, farklı kategorilerdeki üslupsal öznitelikleri önerilen bir çözümde ön işleme olarak birleştirmekte ve bunları tek bir öğrenme sürecinde kullanmaktadır. Bu çalışmada, farklı süreçlerden farklı üslupsal öznitelikleri bir araya getiren Derin Sinir Ağları Kombinasyonu (C-DSA) yaklaşımı ilk kez tanıtılmış, uygulanmış ve başarıyla gerçekleştirilmiştir.

Önerilen yaklaşımın üç aşaması vardır; öğrenme, birleştirme ve karar verme. Öğrenme aşamasında, kullanılan her öznitelik kategorisi için nitel ve nicel özelliklerine göre uygun bir DSA mimarisi üretmeyi öneriyoruz. Bu aşama her kategorinin kodlanmış veya indirgenmiş biçimini öğrenir. Birleştirme aşamasında, farklı öznitelik kategorilerinin kodlanmış formları, bir örneğin tek bir temsilini elde etmek için birleştirilir. Karar verme aşamasında, incelenen örneklerin ele alınan probleme göre sınıfına karar verilir.

Verilen iki belgenin üslupsal farkının tek bir yazar tarzını temsil edip etmediğini test etmek için C-DSA yaklaşımını kullandık. Bir YD problemi olarak, bu araştırma yazarlık analizi çalışmalarının en zorlayıcılarından biri olmasına rağmen, C-DSA kullanarak tek mimarilere göre daha doğru ve sağlam sonuçlar elde ettik.

C-DSA umut verici bir tasarım alternatifidir, çünkü bir grup sınıflandırıcıyı birleştirerek elde edilen sonuçlar tek bir en iyi sınıflandırıcıdan daha iyi olma eğilimindedir. Deneysel sonuçlar, bir çözümün genelleme yeteneğinin ve kullanılan üslupsal özniteliklerin temsil edilebilirliğinin C-DSA yaklaşımı kullanılırken arttığını doğrulamaktadır.

Farklı üslupsal özniteliklerin sıra bilgilerini kullanarak, CNN ve LSTM gibi etkili derin öğrenme yöntemleri, YA problemlerinin bir çözümünde gelecekteki çalışmalar olarak C-DSA yaklaşımı ile test edilmesi planlanmaktadır.

11. İKİLİ ARKA PLAN DESTEKLİ YAZAR-BAĞIMSIZ YAZARLIK DOĞULAMA SİSTEMİ

Bu bölümde, tezin genelinde ele almış olduğumuz güncel yazar doğrulama probleminin çözümüne yönelik yazar bağımsız bir sistem önerilmiştir. Yapılan tez çalışmasının kalbi olan bu bölümde üretilen sistem, içerdiği model itibari ile bilgilendirici bir sistemdir. Önerilen sistem hem Türkçe hem de İngilizce dokümanlar ile test edilmiş, her iki dilde yazılmış dokümanlar için de başarılı sonuçlar elde edilmiştir. Özellikle İngilizce dilinin test edilmesi için kullanılan veri kümesi ile bildiğimize göre şu ana kadarki en yüksek başarılı sonuç elde edilmiştir. Önerilen sistemin konu ve tür bağımsız oluşu da bu sistemin birçok alanda kolaylıkla uygulanabilmesini sağlamaktadır. Önerilen sistem modelinin bir ön işlem olarak kullandığı doküman normalizasyonu ile de hem yazar doğrulama problemlerindeki dengesiz veri durumu hem de dokümanlardaki zaman – konu bağı kırılarak daha faydalı bir yaklaşım geliştirilmiş ve etkileri gözlemlenmiştir.

11.1. Yazar Bağımsız Yazarlık Doğrulama Sistemi

Yazarlık doğrulama, yazar analizi ve dijital metinlerin adli bilişimi konularının ortak alanıdır. Klasik yazarlık doğrulama problemi, sorgulanan bir dokümanın belirli bir yazar tarafından yazılıp yazılmadığının doğrulanması problemidir. Güncel çalışmalar bu problemi verilen iki dokümanın yazarının doğrulanması olarak ele almaktadır. Eğer çalışmalarda sorgulanan durum iki dokümanın aynı yazar tarafından yazılıp yazılmadığı ise, çalışmaların verilecek nihai karar için yazar bağımsız bir çözüm sunması gerekmektedir. Her iki durumda da çalışmaların ana zorlukları verilen dokümanların nasıl değerlendirileceği ile ilgili arka plan bilgisinin eksik oluşu ve uygun karar koşullarının belirlenmesidir. Bu zorlukların üstesinden gelebilmek için yapılan tez çalışmasında yazar bağımsız İkili Arka Plan Modeli (BBM – Binary Backrgound Model) önerilmiştir. Ele alınan iki dokümanın yazarlık doğrulaması için önerilen BBM, ikili örnekler barındıran, dengeli ve ölçeklenebilir bir arka plan sunar. Örnekler aynı yazardan veya farklı yazarlardan alınan doküman çiftlerinin birleştirilmesinden oluşmaktadır. Doküman çiftlerinin doğrulanmasında kullanılmak üzere yazar bağımsız bir arka plan sunmak için birçok yazardan örnekler alınmıştır. Önerilen model hem modelin geliştirilmesinde dokümanı bulunan yazarların dokümanları ile hem de modelin geliştirilme aşamasında görülmemiş yazarların dokümanları ile test edilmiştir. Yapılan deneyler göstermektedir ki, elde edilen sonuçlar

doğrultusunda önerilen model, yazar bağımsız bir yazarlık doğrulama sistemi üretmede başarılıdır. Modelin test edilmesinde, iyi tanımlı ve elle etiketlenmiş bir Türkçe Blog külliyatı, kullanıma açık bir İngilizce Blog külliyatı ve güncel yazar doğrulama problemlerinde kullanılmak üzere yayınlanmış yine kullanıma açık bir veri kümesi kullanılmıştır. İngilizce Blog külliyatı kullanılarak geliştirilen modelin değerlendirilmesinde, önerilen model hem eğitimdeki yazarların hem de model üretiminde görülmeyen yazarların test kümelerinde %90 üzerinde doğruluk sonucu vermiştir. Bu sonuçlar, bildiğimiz kadar ile bu veri kümesinde pozitif ve negatif örneklerinin kullanıldığı yazar doğrulama çalışmalarında elde edilmiş en iyi sonuçtur.

11.2. Konu Kapsam

Elektronik ortamlarda metinsel verilerin üretimi sürekli ve katlanarak artmaktadır. Bu ortamlarda anonim olarak kolaylıkla veri üretebilme imkanı, takma isimler kullanılarak kişilerin tehdit edilmesi gibi suçların işlenmesini kolaylaştırmıştır. Bu sebeple, dijital dokümanlardan yazarları ile ilgili bilgi edinimi büyük önem taşımaktadır. Ele alınan dokümanlar; kaynak kodları [41, 42, 113–115], elektronik postalar [17, 36, 52, 61, 95, 116, 117], mikro mesajlar [36, 118, 119], uç grupların web forum mesajları [38], elektronik sohbet kayıtları [53, 120], blog gönderileri [23, 66, 121] veya farklı metinler olabilmektedir. Dokümanlardan yazarları ile ilgili bilgi çıkarma işlemleri yazarların yazma üslubunun analizi ile yapılabilmektedir. Otomobil kullanma veya telefonla konuşma gibi bir yazarın yazma üslubu hassas veya davranışsal bir biyometri olarak ele alınmaktadır [1, 2] , ki bu durum ‘her doküman kendi yazarının parmak izini taşır’ fikrinin de çıkış noktasıdır. Bu bakış açısı ile yazar analizi çalışmaları, yazarın yazma üslubunu ölçmek için istatistiksel ve hesaplamalı yöntemlerin kullanımıyla birlikte yaklaşık 200 yıllık uzun bir geçmişe sahiptir [3, 4, 122].

Yazarların yazma üslubunu elde edebilmek için birçok stilistik - üslupsal teknik geliştirilmiştir. Araştırmacıların ilgilendiği problemlere göre yazar analizi çalışmaları birçok stilistik teknik kullanılarak 5 ana kategoriye ayrılmaktadır. Bunlar; yazar niteleme, yazar doğrulama, yazar profil çıkarımı, stilistik ölçümü ve ters stilistik ölçümüdür [5]. Yazar analizi alanında araştırmacılar çoğunlukla niteleme ve doğrulama kategorilerine ilgi göstermiş [37] olmalarına rağmen, bu alandaki tüm problemler birbiriyle ilişkilidir ve biri için önerilecek en iyi çözüm, stilistik altyapı dolayısıyla, diğerleri için de faydalı olacaktır. Yazar niteleme veya tanımlama çalışmaları, sorgulanan bir dokümanı bir yazara atamaya çalışır. Yazar tanımlama

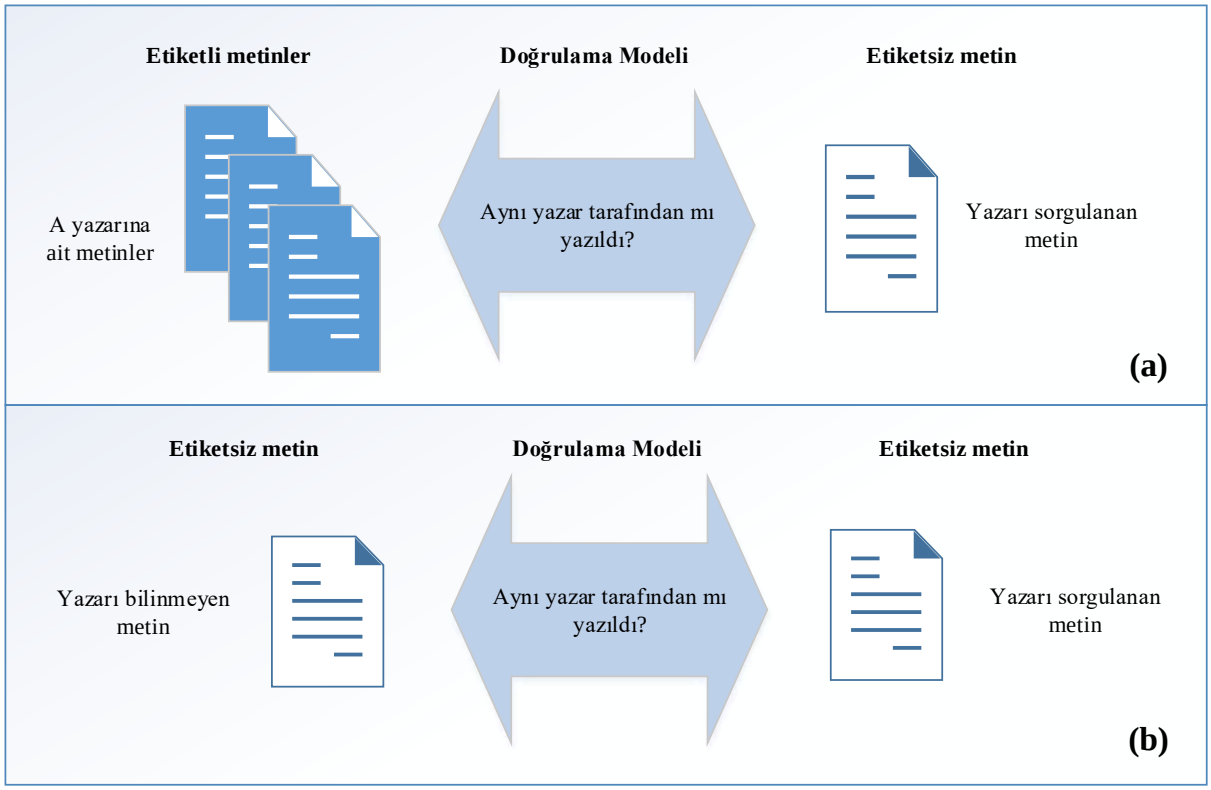
problemlerinin yazar doğrulama problemlerinden farkı sahip olduğu aday kümesidir. Yazar tanımlama çalışmalarında sorgulanan dokümanın yazarının sahip olunan aday yazarlar kümesinde olduğu garantilenmiştir. Dolayısıyla bu çalışmalar kapalı küme problemi olarak ele alınmaktadır [10, 123]. Diğer taraftan yazar doğrulama problemlerinde sorgulanan doküman ile ilgili bir arka plan bilgisi veya bir aday yazarlar kümesi yoktur.

Bu tez çalışmasında biz yazar analizi çalışmalarının en zorlu problemlerinden biri olarak değerlendirilen [20] yazar doğrulama (Authorship Verification – AV) problemine bir çözüm önerdik. Yazar doğrulama çalışmaları sorgulanan bir dokümanın belirli bir yazar tarafından yazılıp yazılmadığının kararını vermek için başarılı karar ölçütleri bulmaya çalışır [103]. Yazar doğrulama problemi, yazar analizi problemlerinin temel bir problemi [20] olduğundan bu alandaki diğer tüm problemler bir dizi yazar doğrulama problemine indirgenebilir [23] ve yazar doğrulama problemi çözülürse diğer problemler de çözülebilir. Fakat bu durumun tersi pek mümkün değildir. Yazar doğrulama problemi, sorgulanan dokümanın yazarını içeren bir aday seti barındırmadığından ki bu durumda herhangi biri sorgulanan dokümanın gerçek yazarı olabilir, çözümü kapalı küme problemlerinin çözümünden daha zorlu olan bir açık küme problemi olarak değerlendirilmektedir [14]. Yazar doğrulama probleminde sorgulanan doküman ile ilgili arka plan bilgisi olmadığından ve şüpheli olarak değerlendirilen tek bir yazar olduğundan bu problem ayrıca tek sınıflı bir sınıflandırma problemi olarak da değerlendirilmektedir [14, 15, 124]. Sorgulanan doküman ya belirtilen sınıfa atanır ya da aykırı sınıf veya negatif örnek olarak değerlendirilir [124]. Dolayısı ile yazar doğrulama çalışmalarındaki en büyük zorluk, belirtilen yazarın dokümanlarını başka yazarların dokümanlarından ayıracak evrensel bir eşik değer veya sınır belirlemektir [18]. Bu durum ile birlikte, başka yazarların dokümanlarını temsil eden olası negatif örneklerin büyüklüğü göz önünde bulundurulduğunda, en temsil edici veya kapsayıcı alt örneklem kümesinin seçimi de oldukça zorlu bir aşamadır [15]. Bu tez çalışmasında, biz bir yazara ait dokümanları karşılaştırmak için gereken negatif örnekleri, ele alınan veri kümesindeki diğer yazarların dokümanlarını kullanarak ürettik. İçerdiği örnekler bakımından dengeli bir ikili arka plan sunabilmek için biz her yazarın doküman sayısı yani pozitif örnek sayısı kadar negatif örnek ürettik.

Literatürdeki birçok çalışma yazar doğrulama problemini bir çeşit yazar tanımlama problemi olarak değerlendirdiğinden dolayı, yazar tanımlamada ele alınan tüm zorluklar ayrıca yazar doğrulama için de geçerlidir [3, 18]. Yazar tanımlama çalışmalarında 3 temel parametre elde

edilen sonuçların başarısını belirlemede önemli rol oynamaktadır. Bunlar; değerlendirilen veri kümesindeki yazar sayısı [125], değerlendirilen dokümanların metin boyutu [126, 127] ve verilerin yazarlar üzerindeki dağılımıdır [128]. Eğer aday yazar sayısı 20'nin altında ve her yazara ait en az 1000 kelime uzunluğunda eğitim için kullanılabilecek doküman varsa, bu problem çözülebilir bir problem haline gelir [5]. Aday yazar sayısının artması veya yazarlara ait veri miktarının azalması elde edilecek başarının azalmasına sebep olmaktadır [68]. Diğer taraftan eğer değerlendirilen yazarlara ait veriler eşit dağılmamış ise, değerlendirme aşamasında da yazarlar eşit olarak dikkate alınamayacaktır. Bu zorlukların üstesinden gelebilmek için ele alınan veri kümesindeki her yazar için eşit sayıda ve eşit uzunluktaki dokümanları veri kümesi normalizasyonu olarak bir ön işlem ile toparladık. Bu aşamada, her yazar için 10.000 kelime uzunluğunda bir dokümanı profil örneği olarak ürettik ve yazar verisi bakımından dengeli bir veri kümesi oluşturmak için bu profil örneğini eşit uzunlukta dokümanlara böldük. Yazarların veri büyüklüklerinin farklılığından kaynaklı zorluğa ek olarak, yazar analizi çalışmalarındaki zorluklardan biri de farklı doküman boyutlarının değerlendirilmesi olduğu için, önerdiğimiz modeli 100, 500 ve 1000 kelimelik doküman örnekleri ile ayrı ayrı olmak üzere 3 farklı örneklem boyutu ile de test ettik.

Bu tez çalışmasında, verilen iki dokümanın aynı yazar tarafından yazılıp yazılmadığı sorusunun cevabının hangi değerlendirmeler ile verilebileceği üzerine araştırmalar yaptık. Bazı çalışmalar bu soruyu belirlenen bir yazar üzerinden yazar tabanlı bir doğrulama problemi olarak değerlendirmekte ve birçok veya uzun bir dokümana sahip yazarların yazar tanımlama problemi olarak değerlendirmektedir. Bu bakış açısı ile yazar doğrulama çalışmaları birçok yanlış anlaşılmayı barındırabilecek hale gelmektedir [21]. Fakat iki dokümanın yazarının doğrulanması durumunda değerlendirilen soru, yazar bağımsız bir soru haline dönüşmektedir. Geleneksel yazar bağımlı yazarlık doğrulama ve yazar bağımsız yazarlık doğrulama problemlerinin temsilen gösterimi Şekil 11.1'de gösterilmektedir.



Şekil 11.1. Yazar bağımlı yazarlık doğrulama (a), yazar bağımsız yazarlık doğrulama (b)

Yazar bağımsız yazarlık doğrulama durumunda bilinen dokümanın yazarının kim olduğu önemli değildir, önemli olan iki dokümanın aynı yazar tarafından yazılıp yazılmadığı sorusunun cevaplanabilmesidir. Bu soru yazar analizi çalışmalarının en zorlu sorularından biri olduğu için açıktır ki bu soru için önerilecek başarılı bir çözüm yazar analizi çalışmalarının çoğu sorusu için de başarılı bir çözüm sunacaktır. Bizim önemle vurguladığımız ana araştırma sorusuna ek olarak, bu tez çalışmasında başka bazı araştırma konularını da değerlendirdik. Bir dil için başarılı sonuçlar veren öznitelik kümesinin başka bir dildeki etkisini değerlendirdik. Ayrıca, yazar analizi çalışmaları aday yazar sayısı ve veri boyutu bakımından zorluklar barındırdığından, önerilen model açısından yazar doğrulama çalışmalarına bu ölçülerin etkisini de değerlendirdik.

İki dokümanın yazarlığına karar verme aşamasında bir arka plan bilgisi sunan, yazar bağımsız bir yazarlık doğrulama modelini bu tez çalışmasında önerdik. Yazar doğrulama problemini dijital metinlerin adli bilişimi alanında gerçek bir dünya problemi olarak ele aldığımız için, değerlendirme aşamasında blog yazılarını kullandık, çünkü blog yazıları çoğunlukla amatör yazarlar tarafından yazılan, konu ve dil bilgisi kısıtlaması olmayan yazılardır. Önerdiğimiz model ayrıca diğer tür yazılara da kolaylıkla uygulanabilirdir. Dahası, model geliştirme

sürecinde dil bağımlı bir araç da kullanmadığımızdan önerdiğimiz model dil bağımsız bir modeldir. Türkçe ve İngilizce blog yazıları külliyatı model değerlendirme aşamasında kullanılmıştır. Bu çalışma için el ile birebir kontrol edilen ve etiketlenen iyi tanımlı bir Türkçe Blog külliyatı topladık. İlk defa bu çalışmada sunulan bu Türkçe külliyat Türkçe dili ile ilgili yapılacak yazar analizi çalışmalarında güvenle kullanılabilir. Bunlara ek olarak bu tez çalışması Türkçe dilinde yazılan metinleri yazar doğrulama perspektifinde kapsamlı olarak değerlendiren ilk çalışmadır. Önerdiğimiz modelin literatüre en önemli katkısı yazar bağımsızlığıdır. Önerilen model bir grup yazar kullanılarak bir defa üretildikten sonra, aynı tür ve dilde olmak şartı ile model üretme aşamasında dokümanı görülmeyen başka yazarların dokümanlarının doğrulanması için de başarılı bir şekilde test edilebilir. Deneyler sonucunda önerilen model hem model üretim aşamasında dokümanı görülmeyen hem de dokümanı görülen yazarların başka dokümanları ile test edildiğinde benzer başarılı sonuçlar vermiştir. Pozitif ve negatif örneklerin değerlendirildiği yazar doğrulama çalışmaları bakımından önerilen model, İngilizce blog külliyatında bildiğimiz kadar ile şimdiye kadarki en yüksek doğruluk sonucu olan %90 üzerinde doğruluk sonucu vermiştir. Önerilen model tür bağımlılığı eğilimi içermesine rağmen bu çalışma kapsamında, bir tür kullanılarak üretilen bir model başka bir türün veri kümesinde de karşılaştırılabilir sonuçlar üretmiştir. Belirtilen tüm bu katkılara ek olarak önerdiğimiz model ayrıca zamanla sürdürülebilirlik sağlayacak yeni örnekler eklenerek güncel olarak eğitilebilecek kapasitededir.

11.3. İlgili Çalışmalar

Yazar analizi alanında 200 yılı aşkın süredir en çok ele alınan problemler yazar niteleme veya yazar tanımlama problemleri olmuştur [3, 4, 122]. Bu problemler sorgulanan bir dokümanın verilen bir grup aday yazar içerisinde dokümanları kendine en benzer olan dokümanların yazarına atanması problemidir. 2000 yılında Stamatatos ve arkadaşları yaptıkları bir çalışmada verilen bir yazarın sorgulanan bir dokümanın yazarı olup olmadığı hipotezinin onaylanması veya reddedilmesi gerekliliği ile ilgilenmişlerdir. Bu çalışma yazar doğrulama probleminin ele alındığı ilk çalışmadır. Bu bakış açısında, belirtilen bir yazar tarafından yazılan örnek dokümanlar vardır ve sorgulanan, harici bir dokümanın belirtilen bu yazar tarafından yazılıp yazılmadığıdır. Benzer bakış açısı ile dijital metnin adli bilişimi ve üslup özellikleri üzerine bir dizi bilimsel olay ve paylaşımlı uygulama barındıran PAN organizasyonu, 2013, 2014 ve 2015 yıllarında yazar doğrulama çalışmalarına odaklanmışlardır [8, 9, 44, 45]. Bu organizasyon bilinen yazarlara ait örnek doküman veya dokümanların olduğu ve her yazar için o yazara ait

olup olmadığı sorgulanan bir dokümanın olduğu yazar doğrulama çalışmaları için açık veri kümesi paylaşmıştır. PAN organizasyonu tarafından 2015 yılında yayınlanan yazar doğrulama veri kümesi yazar bağımsız modeller ile çözülebilecek yapıda olduğundan bu çalışmada da üretilen modelin testinde kullanılmıştır. Sorgulanan bir dokümanın yazarlığını doğrulamak için arka planda bir yazara ait bilinen bir grup nispeten kısa veya tek bir uzun dokümanın var olduğu yaklaşımlar **küme doğrulama** olarak adlandırılabilir. Diğer taraftan, 2014 yılında Koppel ve Winter yazar doğrulama problemini belirtilen yazarın bilinen dokümanı olarak sadece nispeten kısa bir dokümanı ele almıştır [23]. Bu bakış açısı ile klasik yazar doğrulama problemi daha zorlu bir problem haline gelmiştir. Sorgulanan bir dokümanın yazarını doğrulamak için sadece nispeten kısa bir dokümanın bulunduğu yaklaşımlar **tekil doğrulama** olarak adlandırılabilir. Tekil doğrulama probleminin zorlukları bilinen ve sorgulanan dokümanların uzunluğu, en ayırt edici öznitelik kümelerinin seçimi ve başarılı bir cevap fonksiyonunun bulunmasıdır. Bu tez çalışmasında farklı uzunluklarda dokümanlar, farklı öznitelik kümeleri ve cevap fonksiyonu için ikili sınıflandırıcılar ele alınmıştır.

Yazar doğrulama problemi hem küme doğrulama hem de tekil doğrulama olarak iki farklı değerlendirme yöntemi ile ele alınmaktadır, bunlar içsel ve dışsal yöntemlerdir [8]. İçsel yöntemlerde doğrulama probleminin cevabına karar genellikle sadece belirtilen yazarın referans kümesi kullanılarak verilir. İçsel doğrulama yöntemleri genellikle tek sınıflı bir sınıflandırma problemi olarak değerlendirilir ve bu metodun zorluğu bilinen ve bilinmeyen dokümanlar arası belirlenecek sınırın veya evrensel eşik değerinin karakterize edilmesidir [18]. Dışsal doğrulama yöntemlerinde ise belirtilen yazarın bilinen dokümanlarına ek olarak başka kaynaklardan veya başka yazarlardan alınan doküman örnekleri aykırı bir sınıf veya negatif örnekleri temsil etmede kullanılmaktadır [23, 30]. Negatif örneklerin elde edilmesi ile tek sınıflı sınıflandırma problemi çok sınıflı veya iki sınıflı sınıflandırma problemine dönüşmektedir. İçsel modeller özellikle yazar başına bir grup bilinen dokümanın veya uzun bir dokümanın mevcut olduğu yazar doğrulama çalışmaları için daha uygundur. Bu uygunluk ile çoğunlukla içsel modeller küme doğrulama probleminde başarıyla uygulanmaktadır. Diğer taraftan eğer yazarların bilinen dokümanı olarak nispeten kısa bir dokümanın olduğu durumda problemin çözümünde içsel yöntemler yetersiz kalacaktır. Yani, küme doğrulama probleminin çözümü için uygulanan yaklaşımlar içsel ve dışsal yöntemler kullanılarak başarılı bir şekilde yürütülebilmektedir. Fakat tekli doğrulama probleminin çözümü için uygulanan yaklaşımlar içsel metotların kullanımı için uygun değildir, bu yaklaşımların daha fazla parametreyi

değerlendirebilmesi için dışsal yöntemlerin kullanılması daha uygundur. Bu tez çalışmasında tek ve nispeten kısa dokümanların yazarlık doğrulaması probleminin çözümüne başarılı bir model geliştirmeyi amaçladığımız için önerdiğimiz yazarlık doğrulama modelinde dışsal bir yöntem uyguladık. Her yazara negatif örnekler üretmek için, değerlendirilen veri kümesindeki diğer yazarların dokümanları kullanılarak bu tez çalışmasında dışsal yöntem uygulanmıştır. Bu çalışmada uygulanan ile karşılaştırılabilir bir yöntem, sorgulanan dokümanın bazı kelimeleri kullanılarak benzer tür ve konuda olan, arama motorlarına gönderilen sorgular sonucu toplanmış örneklerin negatif örnek olarak ele alındığı Sahtekarlar (Imposter) yöntemi kullanılmıştır [23]. İçerdiği web uygulamasının getirdiği hesaplama maliyetine ek olarak bu yöntem bazı kuşkular içermektedir. Bu kuşkular arama motoru kullanılarak elde edilen negatif örnekler (imposters) ile ilgilidir. Bu örneklerin yazarı bilinen dokümanın yazarı tarafından yazılmış başka dokümanlar olabilme ihtimali bu yönteme kuşku düşürmektedir [14]. Önerdiğimiz model negatif örnekleri değerlendirilen veri kümesindeki başka yazara ait olduğu bilinen diğer dokümanları kullanarak üretmektedir. Özellikle Türkçe külliyat ele alındığında bu külliyat el ile kontrol edilmiş ve etiketlenmiş olduğundan aynı yazara ait başka bir dokümanın negatif örnek üretmede kullanılmış olma riski bu çalışmada yoktur.

Yazar tanımlama ve yazar doğrulama çalışmalarında son yıllarda popülerlik dil bağımsız modellerin üretilmesi üzerinedir [18], özellikle de Hint-Avrupa dil ailesindeki diller için [104]. Dil bağımsızlık yazar analizi çalışmalarında çoğunlukla dil bağımlı doğal dil işleme araçlarının öznitelik çıkarımı aşamasında kullanılmamasına bağlıdır. Bu tez çalışmasında önceki yazar analizi çalışmalarında başarılı sonuçlar üreten ve herhangi bir dil bağımlı doğal dil işleme aracı gerektirmeyen farklı öznitelik kümeleri kullanılmıştır. Böylece önerilen model hem tüm Hint-Avrupa dil ailesi dillerinde hem de Türkçe'nin de içerisinde bulunduğu Ural-Altay dil ailesi dillerinde birebir uygulanabilir. Bu tez çalışmasında önerilen model Türkçe ve İngilizce ile yazılmış iki blog külliyatı ile üretilmiş, test edilmiş ve doğrulanmıştır.

11.4. Kullanılan Veri Kümeleri

Blog yazıları çoğunlukla amatör yazarlar tarafından yazılan, konu ve dil bilgisi kısıtlaması olmayan yazılardır. Bu yazılar yazar analizi çalışmalarının adli bilişimi alanında kullanılan gerçek dünya yazıları olarak değerlendirilebilir. Bu sebeple, önerdiğimiz yazar doğrulama modelinin test edilmesinde Türkçe ve İngilizce olmak üzere iki farklı dilden toplanmış blog

yazılarını kullandık. Aşağıdaki alt başlıklar kullanılan külliyatların ayrıntılı tanımını ve özelliklerini vermektedir.

11.4.1. Türkçe Blog Yazıları Külliyatı

Son zamanlarda birçok dilde yazar analizi çalışmaları yürütülmüştür. Türkçe dilinde, yazar analizi çalışmaları arasında çoğunlukla yazar tanımlama çalışmaları yürütülmüştür. Fakat Türkçe dili kullanılarak yürütülen kapsamlı bir yazar doğrulama çalışması literatürde bulunmamaktadır. Bu sebeple Türkçe dili ile yapılacak yazar analizi çalışmalarının sayısını arttırmak ve kapsamlı bir yazar doğrulama çalışması yürütebilmek için Türkçe Blog Yazıları Külliyatı'nı bu tez çalışmasının bir özgün değeri olarak derledik, topladık. Bu külliyat blogspot.com veya kişisel blog domainlerinden alınan toplamda 120 yazarın blog yazılarını içermektedir. Yazılar Ocak 2016 – Ekim 2018 yılları arasında yayınlanmış ve 2018 yılı içerisinde toplanmıştır. Türkçe blog yazıları külliyatı eşit sayıda kadın ve erkek yazarı barındırmakta ve her yazarın en az 10.000 kelimelik yazısı olduğunu garantilemektedir. Bu külliyat iyi tanımlı bir külliyattır, çünkü bir yazarın kendi yazısından emin olmak için el ile etiketlenmiştir. Blog yazıları ile ilgili bazı karmaşık durumlar söz konusu olabilmektedir. Örneğin; bir yazar başka bir yazarın yazısını veya kendi önceki yazılarından birini tekrar yayınlamış olabilir veya kendi kişisel yazısı yerine bir reklam içeriği yayınlamış olabilir. Bu gibi karmaşık durumların üstesinden gelebilmek için külliyattaki tüm dokümanları kosinüs benzerliği uygulayarak kontrol ettik. Bir yazara ait benzer yazıları toplamamak için, benzerlik oranı %97 ve üstü olan yazıları külliyattan çıkardık. Her yazara ait her yazıyı tek tek etiketledik. Türkçe blog külliyatının metinsel özellikleri Tablo 11.1'de verilmektedir. Bu çalışmada Türkçe blog külliyatını DS1 adı ile ilk veri kümesi olarak ele aldık. Bu külliyat en yakın zamanda tüm ilgililerin kullanabilmesine açık hale getirilecektir.

Tablo 11.1. Türkçe blog külliyatının özellikler listesi

Özellikler	Toplam değer	Yazar başına ortalama değer	Yazar başına standart sapma	En yüksek değer	En düşük değer	Yazar başına ortalama değer
Yazı sayısı	6430	53,58	18,70	100	23	
Paragraf sayısı	194129	1617,74	1113,18	7493	144	30,19
Cümle sayısı	252870	2107,25	1274,95	8932	607	39,32
Kelime sayısı	2646391	22053,26	12700,49	94109	10610	411,56
Karakter sayısı	16327751	136064,60	78206,11	557788	61113	2539,30
Noktalama sayısı	569192	4743,26	3395,29	22073	1621	88,52

11.4.2. İngilizce Blog Yazıları Külliyatı

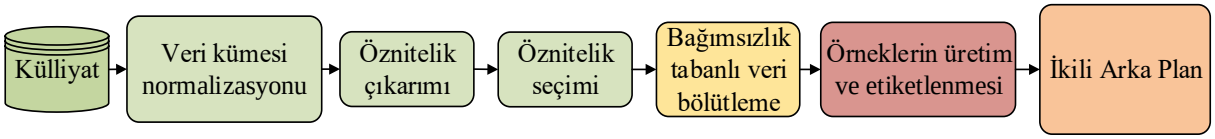
İngilizce blog yazıları külliyatı, blogger.com üzerinden, Ağustos 2004 tarihinde, 19.320 blogger'dan toplanmış yazıları barındırmaktadır [129]. Bu külliyat önerdiğimiz modeli sık sık karşılaştırdığımız Koppel ve Winter tarafından yürütülmüş olan [23] yazar doğrulama çalışmasında kullanılmıştır. Külliyat eşit sayıda kadın ve erkek yazarı barındırmaktadır. Yaygın kullanılan İngilizce kelimelerinden en az 200 defa kullanılmış bloglar toplanmış ve her toplanan bloğun aktif olduğu günden külliyatın toplandığı güne kadar yayınlanan tüm yazıları alınmıştır. Bu külliyatı DS2 ve DS3 adı ile iki farklı boyutta veri kümesi olarak bu tez çalışmasında kullandık. İlk olarak tüm blogları içerdikleri kelime sayısına göre büyükten küçüğe sıraladık. İlk 120 yazarı, Türkçe blog külliyatı ile aynı ölçeklerde değerlendirebilmek için DS2 veri kümesi olarak kullandık. Sıralı listedeki 1200 yazarı da ayrıca DS3 veri kümesi adı ile üçüncü veri kümesi olarak seçip kullandık. Bu tez çalışmasında önerdiğimiz modelin ölçeklenebilirliğini göstermek adına üçüncü veri kümesini diğer iki veri kümesinden içerdiği yazar bakımından 10 kat daha büyük olarak seçtik. Hem DS2 hem de DS3 veri kümelerindeki her yazarın Türkçe blog külliyatında olduğu gibi en az 10.000 kelime içerdiği kontrol edilmiştir.

11.5. İkili Arka Plan Destekli Yazar Doğrulama Sistemi

11.5.1. Yazar Bağımsız BBM Üretiminin Genel Şeması

Başka kaynaklardan toplanan örneklerin bir problemin çözümü için arka plan bilgisi olarak kullanılması, konuşma tanımlama ve konuşma doğrulama çalışmalarında başarılı bir şekilde kullanılan, kabul edilmiş ve iyi yerleşmiş bir yöntemdir [130, 131]. Bu yöntem ayrıca yazar

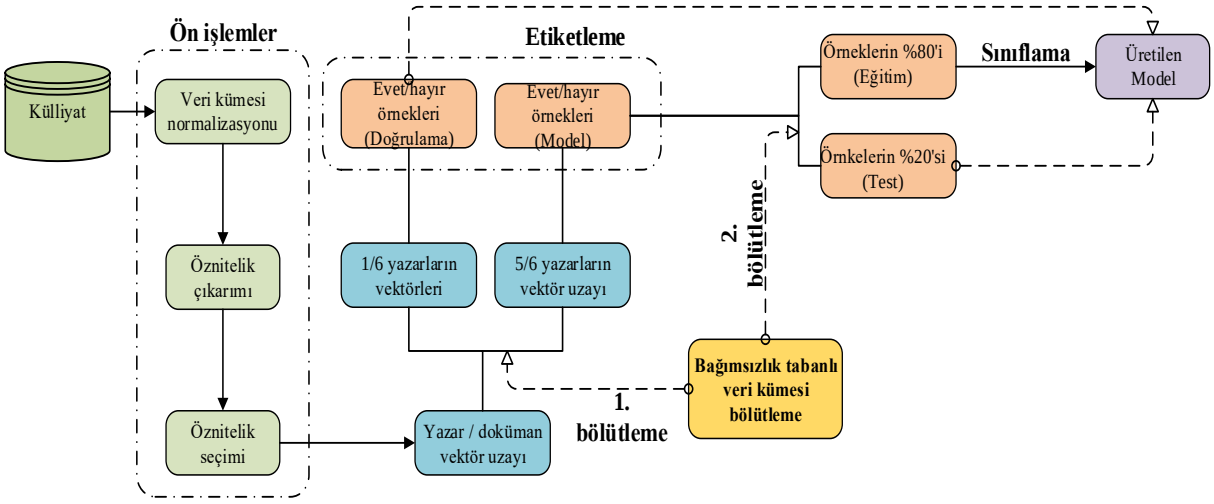
doğrulama çalışmalarında da farklı altyapılar ile kullanılmıştır [23, 132]. Yazar bağımsız bir yazar doğrulama modeli geliştirmek için bu tez çalışmasında, aynı yazara ait doküman çiftlerinin farklı yazarlara ait doküman çiftlerinden farkını temsil eden, dengeli ve ölçeklenebilir ikili bir arka plan önerdik. Arka plan örneklerini üretmek için bir takım birleştirme yöntemleri kullandık fakat üretilen sistemin esnekliğinin sağlanması adına arka plan bilgisi üretmek için başka yaklaşımlar da bu aşamada kullanılabilir. Şekil 11.2 arka plan üretmede kullandığımız metotların akışını göstermektedir.



Şekil 11.2. İkili arka plan üretiminde kullanılan yöntemlerin akışı

Önermiş olduğumuz yazar bağımsız yazar doğrulama sistemi 4 iyi tanımlı adımdan oluşmaktadır. İlk adım birer ön işlem olarak 3 alt adımdan oluşmaktadır. Yazar kümesi büyüklüğü ve veri büyüklüğü bakımından dengeli bir arka plan elde etmek için ilk olarak, ele aldığımız veri kümelerindeki her yazar için eşit olacak şekilde hibrit örnekleme ile veri kümelerini normalize ettik. Normalize etme işlemi sonrasında her yazar için eşit sayıda ve eşit boyutta dokümanlar elde edilmiştir. Daha sonra tüm bu doküman örneklerinden, dil bağımsız, Türkçe ve İngilizce yazar analizi çalışmalarında başarılı sonuçlar veren bazı öznitelik kümelerini çıkardık. Ön işlem adımlarında son olarak eldeki örneklere uygulamadaki hesaplama maliyetini ve zamanını azaltmak için tf-idf değerleri ile öznitelik seçme yöntemini uyguladık. Öznitelik seçme işlemi uygulandıktan sonra eldeki tüm örnekler yani dokümanlar seçilen özniteliklere göre öznitelik vektörlerine dönüşmüştür. İkinci adımda, ilk durumda her yazarın normalize edilmiş doküman örneklerini temsil eden öznitelik vektörleri kümesi bulunmaktadır. Bu öznitelik vektörleri kümesini yazar tabanlı olarak Model kümesi ve Doğrulama kümesi olmak üzere 2 gruba ayırarak veri bölütlemesi yaptık. Bir grup bağımsız yazara sahip olabilmek için, önerilen modelin geliştirilme aşamasında etkisi olmayacak olan yazarların dokümanlarını barındıran Doğrulama kümesini Model kümesinden ayırdık. Değerlendirilen veri kümelerindeki yazarların 1/6'i Doğrulama kümesi için rastgele seçilmiştir ve önerilen modelin tutarlılığını sağlamak için bu işlem model geliştirme sürecinde Doğrulama kümesinin seçiminden sürecin sonuna kadar 10 defa tekrarlanmıştır. Üçüncü adımda, elimizdeki öznitelik vektörlerini çiftler olarak birleştirdik ve birleştirilen vektör çiftlerinin yazarlıklarına bağlı olarak onları "evet" veya "hayır" olarak etiketledik. Birleştirilen iki

öznitelik vektörü eğer aynı yazardan elde edilmiş ise “evet” olarak, farklı yazarlardan elde edilmiş ise “hayır” olarak etiketlenmiştir. Üretilen “evet/hayır” vektörleri Model kümesi ve Doğrulama kümesi için ayrı ayrı üretilmiştir. Her iki kümede de yazarları aynı olan doküman çiftleri ve farklı olan doküman çiftlerinin birleşimini temsil eden öznitelik vektörleri bulunmaktadır. Önerilen modelin geliştirilmesinde rol oynayan dokümanların yazarlarına ait başka dokümanları, bağımlı yazarların doküman kümesi olarak ayırmak için ikinci veri bölütleme işlemini üçüncü adımda gerçekleştirdik. Model kümesindeki verileri örneklem bazında Eğitim kümesi ve Test kümesi olarak ikiye ayırdık. Model kümesindeki örneklerin 1/5’ini rastgele olarak Test kümesi için seçtik. Önerdiğimiz modelin tutarlılığını sağlamak için bu seçim işlemi her model üretiminde bu aşamadan sonuna kadar 5 defa tekrarlanmıştır. Önerilen modelin son aşaması dördüncü adımdır. Bu adımda bir takım ikili sınıflandırma algoritmaları Eğitim kümesi ile eğitilmiş bir model üretmek için kullanılmıştır. Sınıflandırma algoritmalarının sonucu olarak üretilen modeli hem bağımlı yazarların hem de bağımsız yazarların örnekleri ile test ettik. Önerilen modelin iş akışı Şekil 11.3’te gösterilmektedir. Devam eden bölümlerde yapılan işlem adımları detaylandırılmıştır.



Şekil 11.3. İkili arka plan destekli yazarlık doğrulama sisteminin iş akışı

11.5.2. BBM için Ön İşlemler

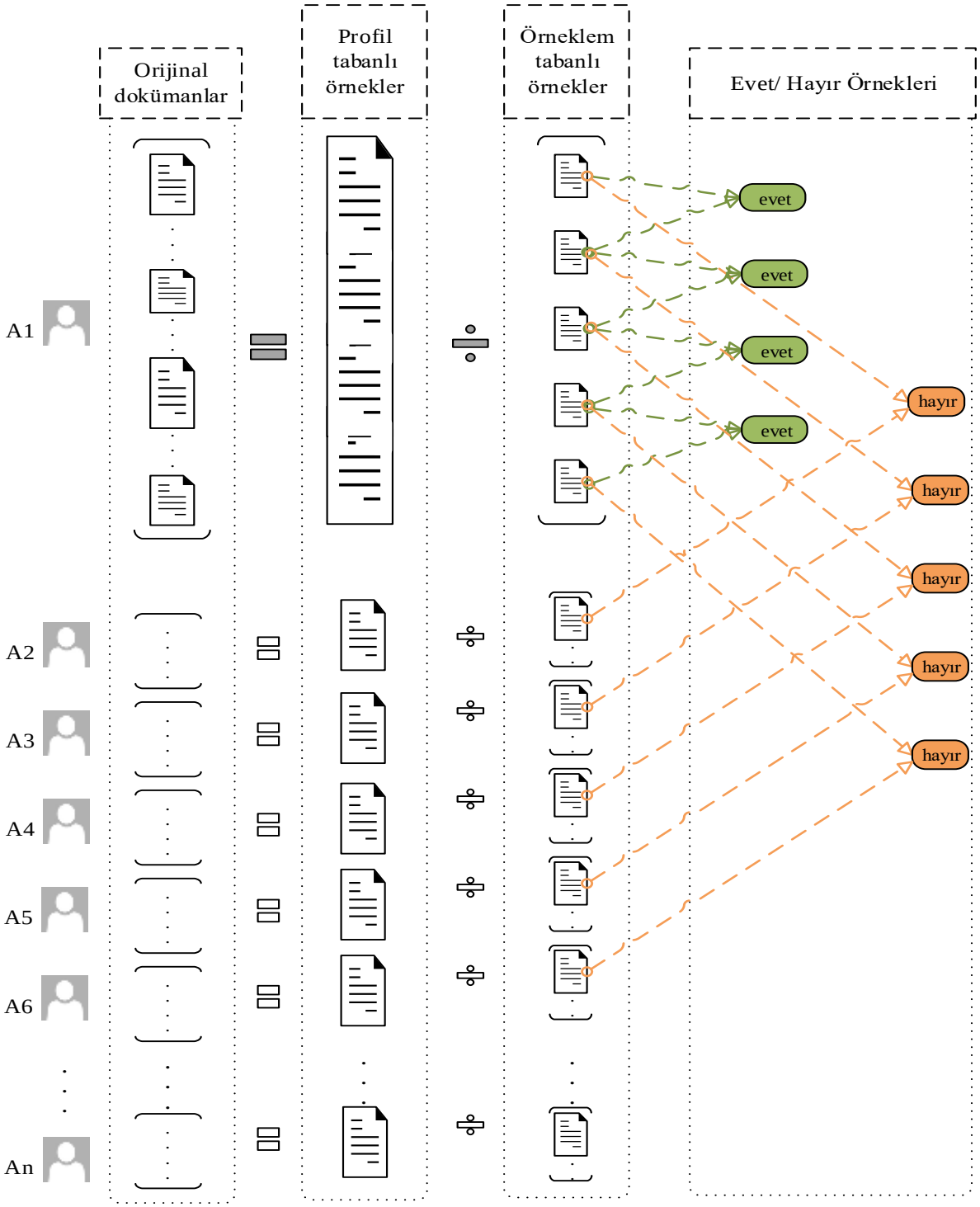
a) Örnekleme Tabanlı Veri Kümesi Normalizasyonu

Yazar niteleme çalışmalarında ele alınan üç çeşit örnekleme yaklaşımı bulunmaktadır; profil tabanlı, örneklem tabanlı ve hibrit. Profil tabanlı örnekleme yaklaşımında yazar başına kullanılabilecek tüm dokümanlar yazarın toplu yazma üslubunu temsil eden uzun, tek bir

doküman elde edebilmek için birleştirilir [3, 14]. Yazar başına yaklaşık 10.000 kelimelik bir veri, yazarın üslupsal kümesinin belirlenmesi için yeterli bir güven değeri olarak görülmektedir [72]. Bu sebeple profil tabanlı örneklerin nispeten uzun dokümanlar olması beklenmektedir. Örneklem tabanlı yaklaşımlarda ise ya bir yazara ait eldeki tüm örnekler ayrı ayrı değerlendirilir ve ya eğer bir yazara ait uzun bir doküman varsa değerlendirme için bu uzun doküman birçok parçaya ayrılır. Örneklem tabanlı yaklaşımlarda yazar başına elde bulunan eğitilecek doküman sayısı bakımından dengesiz sınıf problemleri oluşabilmektedir. Ayrıca elde bulunan dokümanların boyutlarının farklılığı da sınıf dengesizliği oluşturabilmektedir. Diğer taraftan profil tabanlı yaklaşımlarda dengesiz sınıf problemi sadece profil örneğinin uzunluğuna bağlıdır [3]. Örneklem tabanlı yaklaşımların ele alındığı durumlarda değerlendirilecek dokümanların boyutları bakımından normalize edilmesi gerekmektedir. Normalizasyon işlemi için bu yaklaşımlarda uzun dokümanlar eşit uzunlukta dokümanlara ayrılmaktadır. Örneğin 500 karakter içeren [34, 37], 500 kelime içeren [23, 33], 200,500 ve 1000 kelimelik metinler içeren [126], 500 ve 2000 kelime içeren [11] ve benzer oranlarda veri içeren dokümanlar literatürde değerlendirilmiştir.

Örneklem tabanlı ve profil tabanlı örnekleme yaklaşımlarının birlikte kullanımı ile Hibrit örnekleme yaklaşımı uygulanabilmektedir. Hibrit örnekleme yaklaşımının bir uygulaması, bir yazara ait eğitim için kullanılacak tüm dokümanlar örneklem tabanlı yaklaşımda olduğu gibi ayrı ayrı ele alınır, ve daha sonra her yazara ait örnekler tek bir profil vektörü temsili elde etmek için ortalama bir öznitelik vektörü olarak birleştirilir [133, 134]. Diğer bir hibrit örnekleme yaklaşımı uygulamasında ise bir önceki uygulamadaki işlem sırasının tersi uygulanmaktadır. Bu yaklaşımın uygulamasında öncelikle her yazar için tüm eğitim örnekleri birleştirilerek profil örneği üretilir ve daha sonra bu profil örneği eşit büyüklükte örneklemler elde edebilmek için eşit uzunluktaki dokümanlara bölünür [37, 135]. Yazar küme büyüklüğü ve veri büyüklüğü bakımından dengeli bir model üretebilmek için bu tez çalışmasında profil tabanlı örnekleme yaklaşımı ile örneklem tabanlı örnekleme yaklaşımının sıralı olarak bir arada kullanıldığı hibrit örnekleme yaklaşımı kullanılmıştır. Profil tabanlı örneklerin üretim aşamasında bir yazara ait orijinal dokümanların rastgele birleştirilmesine özen gösterdik. Yazarların yazma üslubu zaman içinde dokümandan dokümana bilinçli olarak veya bilinçsizce değişebilir [24]. Bu nedenle bir yazarın bilinçli veya bilinçsizce yapılan değişimleri içeren dokümanları arasındaki derin benzerliğin çıkarımı, o yazarın ayırt edici yazma üslubunun temsili için oldukça önemlidir [15]. Dokümanları arasındaki bilinçli veya bilinçsiz farklılıklara rağmen yazarın ayırt edici yazma

üslubunu elde edebilmek için dokümanları her yazar için profil tabanlı örnekler oluşturma aşamasında rastgele olarak bir araya getirdik ve daha sonra bu profil örneklerini eşit büyüklükteki örneklemlere böldük. Böylece bir yazarın dokümanları arasındaki zaman ve konu bağlamı elimine edilmiş oldu. Örneklem tabanlı örnekler elde edildikten sonra, her örnek için bir öznitelik vektörü ürettik. İki dokümanın yazarlık doğrulamasına verilecek cevabın fonksiyonunu üretebilmek için vektörleri çiftler bazında birleştirdik. Birleştirme sonucunda elde edilen vektörleri, aynı yazara ait örneklerin birleşiminden oluşuyorsa "evet" olarak, faklı yazarların örneklerinin birleşiminden oluşuyorsa "hayır" olarak etiketledik. Birleştirme işleminin ayrıntıları Bölüm 7.5.4.'te verilmektedir. Bir yazara ait orijinal dokümanların "evet/hayır" örneklerine dönüşmesi sırasında geçtiği aşamaları gösteren örnekleme işleminin iş akış diyagramı Şekil 11.4'te gösterilmektedir. Bu süreç ele alınan veri kümelerindeki her yazar için ayrı ayrı uygulanmıştır.



Şekil 11.4. Bir yazar için örnekleme aşamasının iş akışı

Yazarların orijinal dokümanlarını 10.000 kelime tamamlanıncaya kadar rastgele olarak bir araya getirdik ve böylece her yazar için 10.000 kelime uzunluğunda tek bir profil dokümanı üretmiş olduk. Örneklem tabanlı örnekler bakımından farklı büyüklüklerde dokümanları değerlendirebilmek için elimizdeki veri kümelerinin [DS1, DS2, DS3] her birin 3 farklı veri

kümesine dönüştürdük. Dönüştürülen ilk veri kümeleri profil tabanlı örneklerin 10 eşit parçaya bölünmesi ile elde edilen, 1.000 kelime uzunluğundaki doküman örneklerini barındıran ve DS1.1, DS2.1 ve DS3.1 olarak adlandırılan veri kümeleridir. İkinci dönüşüm işlemi yine her veri kümesine bu defa 500 kelime uzunluğundaki örneklemleri elde edebilmek için profil örneklerinin 20 eşit parçaya bölünmesi ile uygulanmış ve elde edilen veri kümeleri DS1.2, DS2.2 ve DS3.2 olarak adlandırılmıştır. Son dönüşüm işlemi 100 kelime uzunluğundaki doküman örneklerini içeren, DS1.3, DS2.3 ve DS3.3 olarak adlandırılan profil örneklerinin 100 eşit uzunluktaki dokümana bölünmesi ile elde edilen verilere uygulanmıştır. Veri kümesindeki tüm yazarlar için eşit büyüklükteki örnekler elde edildikten sonra bu örneklerin aynı yapıdaki öznitelik vektörlerini elde edebilmek için bazı öznitelik çıkarma ve seçme algoritmaları bu örneklere uygulanmıştır. Bu uygulamalar sonrasında elde edilen öznitelik vektörlerinin aynı yazardan alınan çiftlerin birleştirilip "evet" olarak etiketlenmesi ve farklı yazarlardan alınan çiftlerin birleştirilip "hayır" olarak etiketlenmesi ile "evet/hayır" örnekleri üretilmiş olmaktadır. Üretilen tüm veri kümelerinin listesi ve bu kümelerin barındırdığı örneklem sayıları Tablo 11.2'de gösterilmektedir.

Tablo 11.2. Üretilen veri kümelerinin listesi ve özellikleri

Veri kümesi	**Dil**	**Veri kümesindeki yazar sayısı**	**Yazar başına örneklem sayısı**	**Örneklem başına kelime sayısı**	**Toplam evet/hayır örnek sayısı**
DS1.1	Türkçe	120	10	1.000	2.400
DS1.2	Türkçe	120	20	500	4.800
DS1.3	Türkçe	120	100	100	24.000
DS2.1	İngilizce	120	10	1.000	2.400
DS2.2	İngilizce	120	20	500	4.800
DS2.3	İngilizce	120	100	100	24.000
DS3.1	İngilizce	1.200	10	1.000	24.000
DS3.2	İngilizce	1.200	20	500	48.000
DS3.3	İngilizce	1.200	100	100	240.000

b) Öznitelik Çıkarımı

Yazar analizi doğrudan bilinçaltı söz dizimsel deyimler gibi özniteliklerin, bir yazarın tekil yazma üslubunun tanımlanmasında yeterli görülen üslupsal analiz ile türetilmektedir [1, 5]. Bu tez çalışmasında öznitelik çıkarma işlemlerini ele alınan veri kümelerindeki her yazar için örneklem tabanlı örnekleri ürettikten sonra bu örnekler üzerine uyguladık. Bu işlemler

öncesinde tüm orijinal dokümanlardaki büyük harfler küçük harflere dönüştürülmüş ve 'utf-8' tabanlı tüm dil kodlaması hataları göz ardı edilmiştir. Bu çalışmada, Türkçe ve İngilizce dilleri kullanılarak yapılan yazar analizi çalışmalarında en iyi sonuçları veren öznitelik kümeleri kullanılmıştır.

Hint-Avrupa dil ailesi ile karşılaştırıldığında, kök kelimeye eklerin sondan eklenmesi ile geliştirilen sondan eklemeli dil yapısı gereği Türkçe çok farklı bir morfolojik ve söz dizilimsel yapıya sahiptir. İngilizce'de "we are not coming" olarak 4 kelime ile yazılan anlam Türkçe'de "gelmiyoruz" olarak tek bir kelime ile çevrilebilmektedir. Verilen örnekteki "come" kelimesi "gel" kök kelimesi olarak çevrilmekte iken verilen İngilizce cümledeki diğer kelimeler bu kök kelimesine ek olarak sondan eklenmektedir. Türkçenin sondan eklemeli bir dil olması ve yüzlerce farklı kelimenin tek bir kök kelimeden türetilebiliyor olmasından ötürü Türkçe dili kullanılarak yapılan yazar analizi çalışmalarında köklerin belirlenmesi önemli bir adımdır. Kelimelerin köklerinin çıkarılmasında dil bağımlı bir doğal dil işleme aracı kullanmamak için, Türkçe dili üzerinde bu alanda yapılan bazı çalışmalar Sabit Ön Kök (Fixed Prefix Stemming) yöntemini kullanmışlardır [136, 137]. Hint-Avrupa dil ailesinde kullanımı için de benzer şekilde bu öznitelik çıkarma yaklaşımı, bir dokümanda iki boşluk arasındaki yapının ilk k karakterinin elde edilmesi 'kelime k-ön ek' [token k-prefix] olarak adlandırılıp kullanılmaktadır [18]. Biz de bu tez çalışmasında k-sabit ön kök özniteliklerini, sağladığı dil bağımsızlığına ek olarak Türkçe dokümanlar için de önemini göz önünde bulundurup kullandık. Farklı k değerleri için farklı öznitelik kümeleri çıkardık. İlk üç ana öznitelik kümemiz sırası ile 3, 4 ve 5 k değerlerin ön kök özniteliklerinin frekanslarını içermektedir.

İngilizce dili kullanılarak yapılan yazar analiz çalışmalarında en çok kullanılan öznitelik kümesi, hem çıkarılması kolay hem dil bağımsız hem de gürültü toleransı olduğu için bir dokümandaki sıralı n karakterin değerlendirildiği karakter n-gramlardır [5, 68]. Özellikle karakter 4-gramlar en iyi sonucu vermektedir [11, 23], yazar başına ele alınan doküman sayısı azalmış olsa bile [37]. Yukarıda vurguladığımız sonuçlar doğrultusunda deneylerimizde kullandığımız dördüncü ana öznitelik kümesi karakter 4-gram özniteliklerinin frekansları olmuştur.

Söz dizimsel bilgilerin eklenmesi gibi daha çok çeşitli özniteliklerin öznitelik kümelerine eklenmesinin yazar analizi çalışmalarında olumlu bir etkiye sahip olduğu yapılan çalışmalar ile desteklenmektedir [68]. Bu bakımdan bu tez çalışmasında, yukarıda belirtmiş olduğumuz 4 ana öznitelik kümesine ek olarak bazı yardımcı öznitelikler de eldeki doküman örneklerinden

çıkarılarak ana öznitelik kümelerine eklenmiştir. Kullandığımız yardımcı öznitelikler her doküman örneği için; noktalama frekanslarını, kullanılan ortalama kelime uzunluğunu, paragraf frekansını ve toplam karakter sayısını içermektedir. Sonuç olarak bu tez çalışmasında kullandığımız öznitelik kümeleri, barındırdıkları öznitelikler ve bu kümelere verilen adlandırmalar Tablo 11.3'te gösterilmektedir.

Tablo 11.2. Kullanılan öznitelik kümeleri, içerikleri ve adlandırmaları

Öznitelik kümesi	**Açıklama**
FS1	kelime 3-ön-kök + 2 karakterli kelimeler + yardımcı öznitelikler
FS2	kelime 4-ön-kök + 2 ve 3 karakterli kelimeler + yardımcı öznitelikler
FS3	kelime 5-ön-kök + 2, 3 ve 4 karakterli kelimeler + yardımcı öznitelikler
FS4	karakter 4-gramlar + yardımcı öznitelikler

c) Öznitelik Seçimi

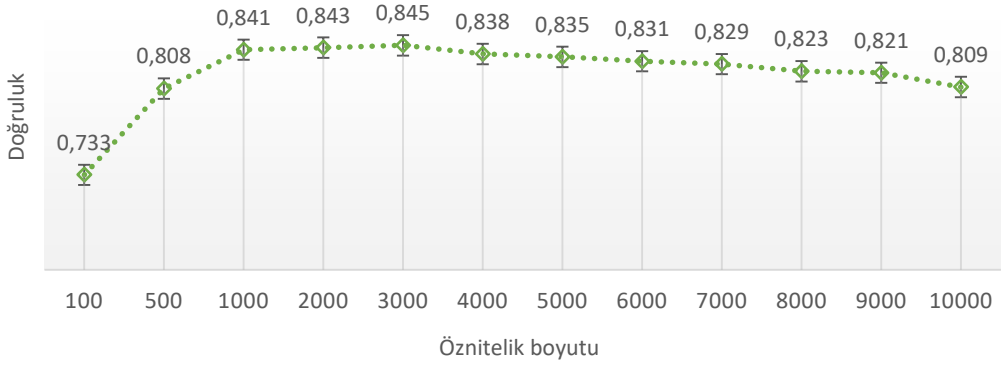
Yazarlara ait dokümanların temsili için çıkarılan öznitelik kümelerinin, özellikle de karakter tabanlı öznitelik kümesinin çok fazla bir büyüklüğe sahip olması örneklerin gösterimini oldukça büyük boyutta olmasına sebep olmuştur. Öznitelik boyutu büyüdükçe yapılan uygulamaların hesaplama maliyeti ve kullanılan bellek boyutu da büyük ölçüde artmaktadır. Bu özniteliklerin boyutunu düşürmek amacı ile hesaplama işlemlerinin faydası açısından bir öznitelik seçme algoritmasının kullanılması gerekmektedir [5]. İki dokümanın yazarlık doğrulaması amacı ile geliştirilen yazar bağımsız yazar doğrulama sisteminin geliştirilmesinde kullanılacak öznitelik seçme algoritmasının ele alınacak dokümanlar arasındaki tekil farklılık üzerinden öznitelik seçimi yapabilmesi gerekmektedir. Bu durumda tf-idf öznitelik seçme algoritmasının önerilen modelin amaçlarına en uygun algoritma olduğu belirlenmiştir. Örneklem tabanlı örnekler üretildikten sonra ele alınan tüm veri kümeleri için Denklem 22'de verilen formül, öznitelik kümelerindeki her özniteliğin tf-idf değerini belirlemek için kullanılmıştır.

$$tf - idf(i) = tf(i) \times log\left(\frac{N}{df(i)}\right) \tag{22}$$

Denklem (15), bu çalışmada kullanılan tf-idf öznitelik ağırlıklandırma yönteminin matematiksel temsilidir. Bu gösterimde tf değişkeni terim frekansını temsil ederken tf(i)

değişkeni i. özniteliğin tüm dokümanlardaki frekansını belirtmektedir. N değişkeni veri kümesinde bulunan doküman sayısını temsil ederken, df(i) değişkeni ise i. özniteliği içeren doküman sayısını temsil etmektedir.

Seçili öznitelikler bakımından en iyi sonucu verecek değeri bulmak için, önerdiğimiz modelin eğitiminde en yüksek tf-idf değerine sahip ilk 100'den ilk 10.000'e kadar olan öznitelikleri arttırarak önerilen modelin eğitim aşamasında elde ettiği doğruluk başarısını değerlendirdik. Farklı boyutlardaki seçili özniteliklerin eğitim aşamasında değerlendirilmesi sonrası elde edilen doğruluk sonuçları Şekil 11.5'te görülmektedir. DS1.2 veri kümesinde, SVM sınıflandırma algoritması ve kelime 4-ön kök öznitelik kümesi kullanımı ile yapılan deney sonucu 100 ve 10.000 en iyi tf-idf özniteliklerinin verdiği sonuçlar doğrultusunda, en yüksek tf-idf değerine sahip ilk 3000 özniteliğin kullanımı ile en yüksek doğruluk başarısı elde edilmiştir. Yapılan birçok deney sonucunda, en yüksek başarı çoğunlukla 3000 en yüksek tf-idf değerli öznitelik kümesi ile elde edildiğinden devam eden çalışmalarda yapılacak deneylerin bu bulguya göre en yüksek 3000 tf-idf değerli özniteliklerin seçimi ile yapılması uygun görülmüştür.



Şekil 11.5. tf-idf öznitelik seçimine göre farklı boyutlardaki özniteliklerin doğruluk sonuçları

11.5.3. BBM için Bağımsızlık Tabanlı Veri Kümesi Bölütleme

Veri kümesi bölütleme işlemi bu tez çalışmasının en önemli ve en gerekli adımlarından biridir. Önerdiğimiz modelin geliştirilme ve değerlendirilme aşamalarının sağlıklı yürütülebilmesi için ele aldığımız veri kümeleri süreç içerisinde iki defa bölütlenmektedir. İlk bölütleme işlemi veri kümesindeki yazarlar tarafında yapılmaktadır. Önerilen modelin geliştirilmesi aşamasında dokümanları ele alınmayan bağımsız yazarların dokümanları ile modelin test edilebilmesi için, veri kümesindeki yazarların 1/6'sı Doğrulama kümesi, kalan 5/6'sı ise Model kümesi olarak bölütlenmektedir. Seçim işlemi rastgele yürütülmektedir, bu sebeple model üretim ve test

aşamaları, Doğrulama ve Model kümeleri için rastgele yazarlar seçilerek 10 farklı iterasyon ile uygulanmaktadır. İlk bölütleme işleminden sonra elde edilen her iki küme içerisinde o kümelere ait "evet/hayır" örnekleri üretilmektedir. İkinci bölütleme işlemi, Model kümesi üzerinden örnekler bazında yapılmaktadır. Model kümesinde bulunan "evet/hayır" örneklerinin 1/5'i rastgele olarak Test kümesi için bölütlenmekte, geri kalan 4/5 örnek kümesi ise sınıflandırma algoritmasının eğitiminde kullanılmak üzere Eğitim kümesi olarak bölütlenmektedir.

Bu bölümün önemi, önerilen modelin yazar bağımsızlığını test edebilecek altyapının hazırlanmasıdır. Test kümesindeki örnekler, başka örnekleri önerilen modelin geliştirilmesinde rol oynayan yazarlardan elde edilmiştir ve bu sebeple bağımlı yazarların örneklerini temsil etmektedirler. Doğrulama kümesindeki örnekler ise, herhangi bir örneği önerilen modelin geliştirilmesinde rol oynamayan yazarlara ait örneklerden oluşmaktadır ve bu sebeple bağımsız yazarların örneklerini temsil etmektedirler. Sonuç olarak, görünen yani bağımlı yazarların örneklerinin doğruluk değerleri ile görünmeyen yani bağımsız yazarların örneklerinin doğruluk değerleri arasındaki farkı test edebilecek altyapı hazırlanmış olmaktadır. Önerilen modelin üretilmesi, Test ve Doğrulama kümeleri ile değerlendirilmesi süreçlerinin sözde kodu aşağıda gösterilmektedir.

```
10 tekrar yap {
    Veri kümesindeki yazarların 1/6'sını Doğrulama kümesi için, 5/6'sını Model kümesi için rastgele seç
    Seçilen her iki küme için evet/hayır örneklerini üret
    5 tekrar yap{
        Model kümesindeki örneklerin %80'ini Eğitim kümesi için, %20'sini Test kümesi için rastgele seç
        Eğitim kümesi örnekleri ile İkili Arka Plan Modeli'ni üret
        Üretilen model ile Doğrulama kümesi ve Test kümesinin doğruluk değerlerini çıkar
        }
}
```

Bu tez çalışmasında uygulamış olduğumuz veri bölütleme işleminin adli bilişim bakımından iki avantajı bulunmaktadır. Yapılan bölütleme işlemleri ışığında yazar doğrulama problemlerinin iki durumlu hali için çözüm altyapısı hazırlanmış olmaktadır. Eğer bir yazara ait bir grup doküman varsa ve harici bir dokümanın aynı yazar tarafından yazılıp yazılmadığı doğrulanmak

isteniyorsa, hazırlanan bu alt yapı ile bağımlı bir yazarın yazar doğrulaması gibi çözüm sunulabilmektedir. Diğer taraftan eğer elimizde yazarı bilinmeyen iki doküman varsa ve bu iki dokümanın aynı yazar tarafından yazılıp yazılmadığı doğrulanmak isteniyorsa, hazırlanan bu altyapı ile bağımsız bir yazarın yazar doğrulaması gibi bir çözüm sunulabilmektedir. Ayrıca yine adli bilişim perspektifinden bakacak olursak, bu altyapı ile farklı boyutlarda dokümanların da değerlendirilebilir olması ekstra bir avantaj sağlamaktadır.

11.5.4. BBM Örneklerinin Üretimi ve Etiketlenmesi

Hibrit örnekleme yöntemi ön işlem olarak uygulandıktan sonra ele aldığımız her veri kümesindeki her yazar için doküman örnekleri bulunmaktadır. Var olan doküman örneklerine öznitelik çıkarımı ve öznitelik seçme algoritmalarının uygulanması ile bu doküman örnekleri öznitelik vektörlerine dönüşmektedir. Bu tez çalışmasında önerdiğimiz model ele alınan vektör çiftlerinin birleştirilmesine dayanmaktadır. Aynı yazardan alınan vektörleri çiftler halinde birleştirip bu bileşim vektörünü "evet" olarak etiketledik. Benzer şekilde, farklı yazarlardan alınan bu öznitelik vektörlerini çiftler halinde birleştirdik ve bu bileşim vektörünü "hayır" olarak etiketledik. Birleştirme işlemi vektör çiftlerinde aynı sırada bulunan özniteliklere aynı işlemlerin yapılması ile yürütülür ve yapılan işlemin sonucuna göre de yeni bir bileşim vektörü üretilmiş olur. Bu tez çalışmasında birleştirme işlemi için 3 faklı işlem kullandık ve böylece 3 farklı model üretmiş olduk. Kullandığımız ilk işlem fark işlemidir. Birleştirilecek vektör çiftleri üzerindeki özniteliklerin mutlak farkından oluşturulan yeni vektörler BBM_fark olarak adlandırılmaktadır. Üretilen vektörün n. öznitelik değeri, birleştirilen vektörlerin n. öznitelik değerlerinin mutlak farkından elde edilmiştir. İkinci birleştirme işlemi, birleştirilecek vektör çiftlerinin çarpımını temel alır. BBM_carp olarak adlandırılan yöntemde birleşim vektörünün n. elemanı, birleştirilecek vektör çiflerinin n. elemanlarının çarpımı ile elde edilir. Üçüncü birleştirme yöntemimiz ise, birleştirilecek vektör çiftlerinin çarpımlarının karekökü alınarak elde edilen ve BBM_kok olarak adlandırdığımız yöntemdir. BBM_kok yönteminde birleştirilecek vektör çiftlerinin n. elemanların çarpımı alınır ve alınan bu çarpım değerinin karekökü alınarak birleşim vektörünün n. değeri elde edilmiş olur. Bu işlem geometrik ortalama alma işlemidir [138–140]. Birleştirme işlemleri ile oluşan "evet/hayır" örneklerinin, örneklem tabanlı örneklerden birleştirilmiş "evet/hayır" vektörleri olana kadarki tüm işlem aşamaları, tanımları ve matematiksel gösterimleri ile Tablo 11.4'te gösterilmektedir.

Tablo 11.3. Üretilen evet/hayır örneklerinin işlem tanımları ve matematiksel gösterimleri

Tanım	Matematiksel Gösterim
120 yazar bulunan bir veri kümesi	$A_B = \{A_1, A_2, A_3, \ldots\ldots, A_{120}\}$
Her yazar örneklem tabanlı 10 örneğe sahip	$A_1 = \{a_1^1, a_1^2, a_1^3, a_1^4, a_1^5, a_1^6, a_1^7, a_1^8, a_1^9, a_1^{10}\}$
Her örnek bir öznitelik vektörü ile temsil edilir	$a_1^1 = \langle f_1, f_2, f_3, \ldots\ldots, f_n \rangle$
A_1 yazarı için 10 evet örneğinin üretilmesi	**birleştir** $\{(a_1^1, a_1^2), (a_1^2, a_1^3), (a_1^3, a_1^4), \ldots\ldots, (a_1^{10}, a_1^1)\}$
A_1 yazarı için 10 hayır örneğinin üretilmesi	**birleştir** $\{(a_1^1, a_2^1), (a_1^2, a_3^2), (a_1^3, a_4^3), \ldots\ldots, (a_1^{10}, a_{11}^{10})\}$
BBM_fark fonksiyonu ile birleştirme	**fark**$(a_1^1, a_1^2) = \langle c_1, c_2, c_3, \ldots\ldots, c_n \rangle$, $c_i = \|\|a_1^1(f_i) - a_1^2(f_i)\|\|$
BBM_carp fonksiyonu ile birleştirme	**carp**$(a_1^1, a_1^2) = \langle c_1, c_2, c_3, \ldots\ldots, c_n \rangle$, $c_i = a_1^1(f_i) \times a_1^2(f_i)$
BBM_kok fonksiyonu ile birleştirme	**kok**$(a_1^1, a_1^2) = \langle c_1, c_2, c_3, \ldots\ldots, c_n \rangle$, $c_i = \left(\prod (\boldsymbol{a_1^1(f_i)}, \boldsymbol{a_1^2(f_i)})\right)^{\mathbf{1/2}}$

BBM_fark yöntemine benzer bir birleştirme işlemi Koppel ve Winter (2014) tarafından farklı bir senaryo ile bir taban uygulama olarak uygulanmıştır [23]. Yazarlar en iyi sonucu önerdikleri danışmansız öğrenme yöntemi ile elde etmişlerdir. Fakat uygulamış oldukları bu taban uygulamanın önerdikleri model ile olan rekabet gücüne de dikkat çekmişlerdir. Biz bu tez çalışmasının ilk uygulamalarında BBM_toplam ve BBM_ort olarak adlandırdığımız, adından da anlaşılacağı üzere vektörel birleşim sürecinde ele alınan vektörlerin özniteliklerinin toplamını ve aritmetik ortalamasını bileşim işlemi olarak kullanan iki yöntemi de değerlendirdik. Yaptığımız çalışmaların neredeyse hepsinde bu iki yöntem önermiş olduğumuz diğer üç yönteme kıyasla en kötü sonuçları vermiştir. Tez çalışması boyunca yapılan deneylerin çokluğu ve çeşitliliği göz önünde bulundurularak bu iki yöntem devam eden çalışmalardan çıkarılmıştır.

11.5.5. BBM Destekli Yazar Doğrulama Modelinin Hiper-parametreleri

Önermiş olduğumuz BBM destekli yazar doğrulama modelinin hiper-parametrelerinin önerilen modelin çıktısına doğrudan ve sezgisel olarak bazı etkileri vardır. Önerilen modelin hiper-parametreleri; arka planda kullanılan metinlerin boyutu, arka planda kullanılan yazarların sayısı, arka planda kullanılan öznitelik kümesi ve arka planda kullanılan öznitelik seçme algoritmasıdır. Önerdiğimiz model nispeten kısa dokümanların yazarlık doğrulaması için uygun bir modeldir. Önerilen model farklı boyutlardaki dokümanların da yazarlık doğrulamasında kullanılabildiği için, model üretiminde doküman boyutu parametresinin ele alınan problem

kümesine özgü olarak belirlenmesi gerekmektedir. Biz bu çalışmada, 100 kelime ve altındaki dokümanların yazarlık doğrulamasında kullanılmak üzere 100 kelimelik doküman çiftlerini önerilen modelin eğitiminde kullandık. Aynı şekilde 100 ile 500 kelime arasındaki dokümanların yazarlık doğrulaması için arka plan örnekleri 500 kelimelik dokümanlardan oluşan veri kümesi ile, 500 kelime üzeri dokümanların yazarlık doğrulaması için arka plan örnekleri 1000 kelimelik dokümanlardan oluşan veri kümesi ile eğitilmesini öneriyoruz. Arka planda kullanılan yazarların sayısı önerdiğimiz modelde sezgisel bir parametredir. Arka plan örneklerinde bulunan yazar sayısının sürekli artması, üretilen modelin başarısında sürekli bir artış sağlamamaktadır. Probleme özgü yazar sayısının bulunabilmesi için parametre optimizasyonu önerilmektedir. Arka planda kullanılacak öznitelik kümesinin seçimi önerilen modelin başarısını doğrudan etkilemektedir. Probleme ve kullanılan dile özgü olarak farklı öznitelik setlerinin farklı sonuçlar üreteceği aşikardır. Dolayısı ile kullanılacak öznitelik kümesi seçimi için ele alınan dil ve tür dikkate alınmalıdır. Arka planda kullanılan öznitelik seçme algoritması kullanımı ve seçimi tercihe bağlıdır. Bu çalışmada kullanılan veri kümesi çok boyutlu olduğundan ve devamındaki işlemlerin gerçekleştirilmesini büyük ölçüde etkilendiğinden öznitelik seçme algoritması kullanılmıştır fakat makul oranlarda olan öznitelik kümeleri için öznitelik seçme işlemi yapılmasına gerek yoktur. Ayrıca, kullanılacak öznitelik seçme algoritmasının boyutunun belirlenmesi de probleme özgüdür. Yine farklı değerler test edilerek probleme en uygun miktar belirlenmelidir.

11.6. Deneysel Sonuçlar ve Tartışma

Destek Vektör Makineleri (Support Vector Machine) ve Lojistik Regresyon (Logistic Regression) sınıflandırma algoritmaları, python 3.6 scikit kütüphanesindeki varsayılan parametreleri ile bu tez çalışmasında, önerilen modelin üretilmesinde kullanılacak sınıflandırma algoritmaları olarak ele alınmıştır. Bu iki yöntemin pratikteki karşılaştırılabilir performansları dolayısı ile yazar analizi çalışmalarında bu iki metot sıklıkla kullanılmaktadır [23, 36, 62]. Önerilen birleştirme yöntemlerinde olduğu gibi bu tez çalışmasının ilk uygulamalarında birçok sınıflandırma algoritması kullanılmıştır. Fakat yaptığımız çalışmaların neredeyse hepsinde SVM ve LR sınıflandırma algoritmaları diğer sınıflandırma algoritmaları ile kıyaslandığında en başarılı sonuçları ürettiğinden ve tez çalışması boyunca yapılan deneylerin çokluğu ile çeşitliliği göz önüne alındığında, devam eden çalışmalar bu iki yöntem üzerinden sürdürülmesine karar verilmiştir. Önerilen modelin başarısını değerlendirmek için ortalama doğruluk (accuracy) ölçeği kullanılmıştır. Tüm veri kümelerimizdeki Eğitim, Test ve

Doğrulama kümelerimiz örnekler bakımından dengeli olduğu için bu çalışmalardaki en uygun ölçeğin doğruluk ölçeği olduğuna karar verilmiştir.

Önceki bölümlerde bahsedildiği gibi bu tez çalışmasında önerilen model hem model geliştirilmesi aşamasında dokümanı bulunan yazarların başka dokümanları ile (Test kümesi) hem de model geliştirme aşamasında hiç dokümanı olmayan yazarların dokümanları ile (Doğrulama kümesi) test edilmiştir. Yapılan tüm deneyler, Windows 10 Enterprise Edition 64 bit İşletim Sistemi, 16gb ram, 2.00GHz Intel[R] Xeon[R] CPU E5-2620 özelliklerine sahip kişisel bir bilgisayar üzerinden yürütülmüştür. Tüm veri kümeleri için, seçili öznitelik kümeleri ve Lojistik Regresyon algoritması, BBM_fark, BBM_carp ve BBM_kok yöntemleri kullanılarak Eğitim, Test ve Doğrulama örnekleri için elde edilen ortalama doğruluk değerleri Tablo 11.5'te gösterilmektedir.

Tablo 11.4. Lojistik Regresyon kullanılarak üretilen İkili Arka Plan Modelinin ortalama doğruluk sonuçları

BBM Yöntem	Veri kümesi	Eğitim				Test				Doğrulama			
		FS1	FS2	FS3	FS4	FS1	FS2	FS3	FS4	FS1	FS2	FS3	FS4
BBM_fark	DS1.1	0,742	0,743	0,751	0,774	0,687	0,738	0,740	0,769	0,660	0,721	0,705	0,752
	DS1.2	0,706	0,713	0,700	0,717	0,671	0,693	0,692	0,703	0,645	0,655	0,673	0,698
	DS1.3	0,596	0,610	0,622	0,652	0,584	0,597	0,620	0,654	0,576	0,582	0,595	0,648
	DS2.1	0,999	0,993	1,000	1,000	0,695	0,640	0,721	0,736	0,667	0,621	0,727	0,735
	DS2.2	0,995	0,942	0,998	0,889	0,621	0,651	0,645	0,653	0,613	0,652	0,703	0,690
	DS2.3	0,791	0,804	0,794	0,740	0,651	0,642	0,642	0,582	0,652	0,641	0,655	0,623
	DS3.1	0,819	0,808	0,822	0,817	0,753	0,757	0,768	0,791	0,751	0,751	0,756	0,795
	DS3.2	0,757	0,750	0,756	0,776	0,718	0,716	0,727	0,749	0,719	0,715	0,710	0,748
	DS3.3	0,755	0,740	0,708	0,689	0,744	0,729	0,687	0,671	0,730	0,722	0,684	0,674
BBM_carp	DS1.1	0,899	0,847	0,822	0,894	0,797	0,764	0,764	0,833	0,815	0,787	0,783	0,829
	DS1.2	0,833	0,846	0,796	0,860	0,782	0,810	0,771	0,817	0,772	0,766	0,747	0,804
	DS1.3	0,794	0,807	0,749	0,828	0,789	0,796	0,743	0,806	0,763	0,771	0,718	0,794
	DS2.1	0,902	0,871	0,850	0,901	0,827	0,812	0,815	0,827	0,809	0,791	0,789	0,812
	DS2.2	0,851	0,846	0,823	0,840	0,807	0,796	0,788	0,801	0,786	0,771	0,759	0,784
	DS2.3	0,789	0,794	0,776	0,764	0,770	0,768	0,756	0,737	0,753	0,745	0,734	0,728
	DS3.1	0,800	0,766	0,750	0,778	0,801	0,766	0,752	0,778	0,791	0,759	0,754	0,777
	DS3.2	0,820	0,813	0,797	0,800	0,814	0,808	0,797	0,795	0,813	0,806	0,799	0,795
	DS3.3	0,760	0,774	0,771	0,788	0,759	0,773	0,771	0,784	0,756	0,768	0,768	0,784
BBM_kok	DS1.1	1,000	1,000	1,000	1,000	0,808	0,837	**0,847**	0,834	0,829	0,828	**0,843**	**0,840**
	DS1.2	0,983	0,998	0,999	1,000	0,825	**0,849**	**0,843**	0,805	**0,821**	0,818	**0,828**	0,810
	DS1.3	0,891	0,896	0,888	0,887	**0,857**	**0,850**	0,835	0,798	**0,826**	**0,823**	0,813	0,786
	DS2.1	1,000	1,000	1,000	1,000	0,805	**0,853**	**0,846**	0,791	0,805	**0,818**	**0,821**	0,811
	DS2.2	0,978	0,992	0,991	1,000	0,831	**0,841**	**0,845**	0,767	0,807	**0,814**	**0,812**	0,756
	DS2.3	0,826	0,846	0,839	0,832	**0,791**	**0,798**	**0,797**	0,732	**0,767**	**0,768**	**0,764**	0,722
	DS3.1	0,941	0,949	0,946	0,924	**0,901**	**0,902**	**0,910**	0,814	**0,890**	**0,895**	**0,903**	0,812
	DS3.2	0,911	0,926	0,927	0,888	**0,885**	**0,891**	**0,894**	0,840	**0,882**	**0,888**	**0,891**	0,844
	DS3.3	0,841	0,854	0,848	0,816	0,836	**0,847**	**0,840**	0,805	0,834	**0,842**	**0,841**	0,806

Ele alınan tüm veri kümeleri için en iyi sonuçlar LR sınıflandırma algoritması ile BBM_kok birleştirme yönteminin kullanımında ve özellikle kelime k-ön kök öznitelik setleri ile elde edilmiştir. Bu çalışmada elde ettiğimiz sonuçları yazar doğrulama çalışmaları arasında 500 kelimelik doküman çiftlerinin yazarlık doğrulamasını ele alan ve bu değerlendirmeler ile başarılı sonuçlar üreten Koppel ve Winter tarafından yapılan çalışma [23] ile karşılaştırdık. Karşılaştırdığımız çalışma benzerlik tabanlı bir Imposter metodu önermiş ve İngilizce Blog külliyatı kullanarak en başarılı sonucu %87,4 doğruluk ile skor eşik değeri ve karakter n-gram

öznitelik kümesi kullanarak elde etmiş. Bu Imposter metodu imposterler oluşturmak için web tabanlı uygulamalar kullandığından bir takım kuşkular barındırmaktadır. Ayrıca birebir aynı uygulama altyapısı kullanarak benzer sonuçları elde etmek de bu çalışma için pek mümkün görünmemektedir. Bu tez çalışmasında geliştirilen uygulama ile karşılaştıracak olursak, 500 kelime uzunluğundaki dokümanların kullanımında, Test kümesi için %89 üzeri, Doğrulama kümesi için ise yine %89 yakınlarında başarı, BBM_kok birleştirme yöntemi ve kelime k-ön kök öznitelik setleri kullanılarak elde edilmiştir. Bu çalışmada ele alınan altyapının iyi tanımlı olmasından kaynaklı olarak, üretilen negatif örneklerin aynı yazara ait olması ihtimali de bulunmamaktadır. Dahası, önerdiğimiz model aynı altyapı kullanıldığı sürece aynı sonuçları üretecektir. Önerdiğimiz modelin karşılaştırılan çalışmadan bir diğer farkı ise kullanılan öznitelik kümeleridir. Kelime k-ön kök öznitelik kümeleri karakter n-gram öznitelik kümesinden daha iyi sonuçlar vermiştir. Bu çalışmada en başarılı sonuçlar, İngilizce Blog külliyatından üretilmiş 1000 kelime uzunluğundaki dokümanların değerlendirildiği deneylerde kelime k-ön kök özniteliklerinin kullanımı ile Test kümesinde %90 üzeri ve Doğrulama kümesinde de neredeyse benzer oranlarda doğruluk olarak elde edilmiştir. Elde edilen bu sonuçlar başarılı bir yazar bağımsız yazarlık doğrulama modelinin bu altyapı kullanılarak elde edildiğini göstermektedir.

Öznitelik çıkarımı bölümünde kelime k-ön kök öznitelikler kümesinin Türkçe dilinin kullanımındaki öneminden bahsettiğimiz gibi bu öznitelik kümelerinin başarıya etkisi elde ettiğimiz test sonuçlarında görülebilmektedir. Türkçe dokümanların kullanıldığı deneylerde 100 kelime uzunluğundaki dokümanlarda bile Test kümesinde %84 üzeri, Doğrulama kümesinde ise %82 üzeri doğruluk başarısı elde edilmiştir. Benzer ölçekler ile kıyaslandığında, 120 yazarlı İngilizce Blog külliyatında 1000 kelimelik ve 500 kelimelik dokümanlarda Test kümesi için %84 üzeri, Doğrulama kümesi için ise %81 üzeri doğruluk başarısı elde edilmiştir. Fakat aynı ölçeklerle İngilizce Blog külliyatında 100 kelimelik dokümanların değerlendirildiği deneylerde elde edilen doğruluk başarısı Test kümesi için %79 üzeri, Doğrulama kümesi için ise %76 üzeri olarak elde edilmiştir. Öte yandan, İngilizce blog külliyatında ele aldığımız veri kümesini yazarlar ve örnekler bakımından 10 kat genişlettiğimiz durumlarda, 1000 kelimelik ve 500 kelimelik dokümanlar için elde ettiğimiz doğruluk oranı %90 yakınlarında olmaktadır. Hatta bu genişletilmiş veri kümesinde 100 kelime uzunluğundaki dokümanların değerlendirilmesinde bile elde ettiğimiz doğruluk oranı %84 yakınlarında olmaktadır. Bu genişletilmiş veri kümesi ile yapılan deneylerden elde edilen sonuçlar, üretilen arka plan

örneklerinin yazar ve örneklem bakımından boyutunun modele etkisindeki gücünü göstermektedir. BBM_kok birleştirme yönteminin Eğitim kümesi üzerindeki başarılı sonuçları sınıflandırma uygulaması olarak ele alınan modelin alt yapısal başarısını, özellikle 1000 kelime uzunluğundaki dokümanlar değerlendirildiğinde, açığa çıkarmaktadır. LR sınıflandırma algoritması kullanıldığında en düşük sonuçlar BBM_fark birleştirme yönteminin kullanımı ile elde edilmiştir.

Önerdiğimiz modelin bir diğer değerlendirme deneyleri SVM sınıflandırma algoritması kullanılarak gerçekleştirilmiştir. SVM sınıflandırma algoritması kullanılarak elde edilen ortalama doğruluk değerleri, seçili öznitelik kümeleri ve BBM_fark, BBM_carp ile BBM_kok yöntemlerinin kullanımındaki Eğitim, Test ve Doğrulama kümelerindeki örnekleri için Tablo 11.6'da verilmektedir.

Tablo 11.5. Destek Vektör Makineleri kullanılarak üretilen İkili Arka Plan Modelinin ortalama doğruluk sonuçları

BBM yöntem	Veri kümesi	Eğitim				Test				Doğrulama			
		FS1	FS2	FS3	FS4	FS1	FS2	FS3	FS4	FS1	FS2	FS3	FS4
BBM_fark	DS1.1	0,996	0,996	0,995	1,000	0,737	0,735	0,719	0,687	0,767	0,758	0,748	0,706
	DS1.2	0,919	0,915	0,894	0,999	0,687	0,680	0,670	0,729	0,705	0,677	0,680	0,723
	DS1.3	0,691	0,658	0,634	0,770	0,642	0,605	0,586	0,652	0,655	0,632	0,605	0,672
	DS2.1	0,989	0,992	0,992	0,998	0,749	0,759	0,763	0,776	0,788	0,781	0,770	0,769
	DS2.2	0,882	0,875	0,891	0,989	0,702	0,700	0,720	0,741	0,712	0,704	0,717	0,741
	DS2.3	0,621	0,632	0,649	0,701	0,574	0,591	0,613	0,628	0,595	0,598	0,614	0,641
	DS3.1	0,986	0,982	0,981	0,996	0,784	0,777	0,781	0,776	0,794	0,790	0,798	0,784
	DS3.2	0,885	0,889	0,885	0,975	0,750	0,756	0,757	0,763	0,768	0,771	0,773	0,777
	DS3.3	0,687	0,694	0,701	0,724	0,677	0,675	0,684	0,684	0,677	0,679	0,687	0,685
BBM_carp	DS1.1	1,000	1,000	1,000	1,000	0,490	0,490	0,490	0,490	0,500	0,500	0,500	0,500
	DS1.2	1,000	1,000	1,000	1,000	0,484	0,485	0,484	0,484	0,500	0,500	0,500	0,500
	DS1.3	0,750	0,754	0,791	0,971	0,585	0,549	0,534	0,542	0,569	0,545	0,534	0,540
	DS2.1	1,000	1,000	1,000	1,000	0,526	0,517	0,516	0,525	0,532	0,540	0,540	0,526
	DS2.2	1,000	1,000	1,000	1,000	0,521	0,546	0,532	0,544	0,507	0,532	0,520	0,536
	DS2.3	0,747	0,796	0,876	0,968	0,558	0,545	0,543	0,542	0,546	0,539	0,535	0,541
	DS3.1	1,000	1,000	1,000	1,000	0,545	0,543	0,545	0,544	0,532	0,533	0,528	0,531
	DS3.2	1,000	1,000	1,000	1,000	0,535	0,531	0,536	0,535	0,532	0,536	0,526	0,533
	DS3.3	0,725	0,734	0,790	0,766	0,632	0,642	0,684	0,655	0,633	0,663	0,675	0,656
BBM_kok	DS1.1	0,932	0,928	0,910	0,998	**0,812**	0,785	0,771	0,724	**0,791**	0,769	0,734	0,687
	DS1.2	0,880	0,864	0,846	0,951	**0,825**	**0,821**	0,791	0,780	**0,804**	0,775	0,758	0,754
	DS1.3	0,788	0,739	0,716	0,806	**0,788**	0,738	0,709	**0,793**	**0,760**	0,719	0,693	**0,780**
	DS2.1	0,926	0,909	0,897	0,971	**0,815**	0,795	0,767	0,742	**0,805**	0,777	0,744	0,700
	DS2.2	0,858	0,841	0,838	0,880	**0,802**	0,793	0,784	0,754	**0,778**	0,771	0,748	0,726
	DS2.3	0,723	0,732	0,729	0,724	0,721	**0,728**	**0,729**	0,716	0,706	0,707	**0,712**	0,702
	DS3.1	0,932	0,918	0,914	0,967	**0,876**	**0,863**	0,839	0,817	**0,874**	**0,859**	0,836	0,814
	DS3.2	0,881	0,872	0,855	0,895	**0,860**	**0,850**	0,833	0,828	**0,858**	**0,852**	0,833	0,828
	DS3.3	0,788	0,783	0,780	0,776	**0,782**	**0,781**	0,778	0,776	**0,787**	**0,782**	0,774	0,773

LR sınıflandırma algoritmasında elde edilen sonuçlar ile karşılaştırıldığında, SVM algoritması kullanıldığı durumlarda da en başarılı sonuçlar BBM_kok birleştirme yöntemlerinin kullanıldığı yöntemler ve çoğunlukla kelime k-ön kök öznitelik kümesi kullanımı ile elde edilmiştir. Türkçe külliyatın değerlendirildiği deneylerde sadece 100 kelimelik dokümanların ele alındığı durumda karakter 4-gram öznitelik kümesi az bir farkla kelime k-ön kök öznitelik kümelerinden daha iyi sonuçlar üretmiştir. Koppel ve Winter tarafından yapılan çalışmada, BBM_fark birleştirme yöntemine benzer bir yöntem karakter 4-gram öznitelik kümesi

kullanımı ile 500 kelime uzunluğundaki dokümanlara SVM kullanılarak bir taban uygulama olarak uygulanmıştır [23]. Yazarlar İngilizce Blog külliyatında bu taban uygulamaları ile %79,8 doğruluk elde etmişler. Önerdiğimiz model, ele alınan külliyatın 120 yazarlı veri kümesinde 500 kelimelik dokümanlarda hem Test kümesinde hem de Doğrulama kümelesinde aynı danışmanlı taban uygulama özelliklerinin farklı ön işlemler uygulanması ile %74,1 doğruluk verirken 1200 yazarlı veri kümesinde %76 üzeri doğruluk vermiştir. Diğer taraftan BBM_kok birleştirme yöntemi uygulandığında aynı veri kümelerinden elde edilen Test kümesi ve Doğruluk kümesinin doğruluk oranları artmaktadır. Bu tez çalışmasında önerilen modelin SVM kullanımı ile elde edilen en yüksek doğruluk değeri 1200 yazarlı İngilizce Blog külliyatındaki 1000 kelime uzunluğunda dokümanların değerlendirildiği ve FS1 öznitelik kümesinin kullanıldığı deneylerde hem Test kümesi hem de Doğruluk kümesi sonuçlarına göre %87 üzeri çıkmaktadır. Deneylerin tamamı göz önüne alındığında hem SVM hem de LR sınıflandırma algoritmalarının kullanımında en yüksek doğruluk değerleri DS3.1 veri kümesinin kelime k-ön kök öznitelik setleri ile değerlendirilmesinde elde edilmiştir. Türkçe veri kümesinin SVM ile değerlendirilmesinde en yüksek doğruluk oranı %80 yakınlarında, 1000 ve 500 kelimelik dokümanların değerlendirildiği deneylerde, Test ve Doğruluk kümelerinin sonuçlarıyla elde edilmiştir. SVM ile gerçekleştirilen deneylerin geneline bakıldığında yine en yüksek doğrulukları BBM_kok birleştirme yöntemi verirken LR sınıflandırma algoritmasından farklı olarak en düşük sonuçlar BBM_carp birleştirme yöntemi ile elde edilmiştir. Birleştirme yöntemleri karşılaştırıldığında en yüksek öznitelik değerleri BBM_carp birleştirme yöntemi ile üretilen değerlerde bulunmaktadır ve bu durum SVM ile sınıflandırma modelinin başarısını düşüren bir durum olabilmektedir.

LR sınıflandırma algoritmasının kullanımı ile karşılaştırıldığında, SVM sınıflandırma algoritmasının kullanımı ile tüm veri kümelerindeki en yüksek doğruluk değerlerinde düşüş meydana gelmiştir. Bu düşüşe ek olarak, SVM sınıflama algoritmasının hesaplama maliyeti LR sınıflandırma algoritmasının hesaplama maliyetinden çok daha fazladır. Özellikle diğer iki veri kümesinin 10 katı büyüklükte olan1200 yazarın ele alındığı DS3 veri kümesinde model üretmek için gerçekleştirilen her bir tekrar LR kullanımında yaklaşık 1 saatin altında sürerken SVM kullanımında bu süre en az 3 gün sürmektedir. DS1 ve DS2 veri kümeleri için 1000 kelime uzunluğundaki dokümanların kullanıldığı deneylerde hesaplama süresi hem SVM hem de LR algoritmalarının kullanımında 5 dakikanın altında olurken, bu süre 500 kelime uzunluğundaki dokümanların değerlendirilmesinde 10 dakikanın altında, 100 kelime uzunluğundaki

dokümanların değerlendirilmesinde ise 140 dakikanın altında olmaktadır. Model üretiminde gerçekleştirilen her iterasyonda aynı veri bölümlerine hem SVM hem de LR algoritması uygulandığından yapılan deneylerin hesaplama süresi uzun olmaktadır. Öte yandan LR sınıflandırma algoritması tek başına uygulandığında deneylerin hesaplama süresi büyük ölçüde azalmaktadır. Öznitelik çıkarımı ve seçimi aşamalarında da, sayıca fazla olduğu için özellikle karakter n-gram öznitelik kümesinin işlemleri diğer öznitelik kümelerinin işlemlerinden çok daha uzun sürmektedir. Karakter 4-gram öznitelik kümesinin kullanımında, yazar küme sayısı ve dokümanların veri miktarı arttıkça çıkarılan özniteliklerin sayısı da 300.000 yakınlarında olmaktadır. Kelime k-ön kök öznitelik kümeleri için en uzun öznitelik çıkarma ve seçme yöntemlerinin süresi DS3.3 veri kümesi için yaklaşık 2 gün sürmektedir. Diğer veri kümeleri için bu süre 2 saatin altındadır.

İngilizce blog veri kümesinin kullanımını içeren çalışmaların genel bir karşılaştırması [141] çalışmasında yapılmıştır. Çalışmada 2000 yazarın 4 sayfa olacak şekilde (yaklaşık 2000 kelimelik) dokümanları çıkarılmış. Benzer bir altyapı sağlamak için biz bu çalışmada her yazara ait 1000 kelimelik doküman çiftlerini kullandık. Önerdiğimiz modelin parametrelerini şöyle ayarladık; arka plan üretiminde 1000 yazar, tf-idf öznitelik ağırlıklandırma kullanılarak en yüksek ağırlığa sahip 3000 öznitelik, sınıflandırma için Logistic Regression algoritması ve çalışmada başarısı test edilmiş üç öznitelik kümesi (FS1, FS2 ve FS3). Çalışmanın performans ölçeği olarak, yazar doğrulama çalışmalarında sıklıkla kullanılan performans ölçekleri; doğruluk (accuracy), Alıcı İşlem Karakteristiğinin (Reciever Operating Characteristic - ROC) Eğri Altındaki Alanı (Area Under Curve - AUC) ve Denklem 23'te gösterilen c@1 metrikleri kullanılmıştır.

$$c@1 = \frac{1}{n}\left(n_c + \left(\frac{n_u n_c}{n}\right)\right) \tag{23}$$

n = problem sayısı

nc = doğru cevapların sayısı

nu = cevaplanamayan problem sayısı

Yukarıdaki denklemde eğer n_u değişkeni yani cavaplanamayan problem sayısı sıfıra eşit ise, c@1 ölçeği doğruluk ölçeği ile aynı sonucu üretecektir. Önerilen problemden ve [141] çalışmasından elde edilen sonuçlar Tablo 11.7'de gösterilmektedir. Tüm çalışmalarda n_u değeri sıfırdır. Tabloda verilen öğrenme tipinde; Türkçeye eğitilmiş olarak çevirmeyi uygun gördüğümüz istekli (eager) olarak belirtilen çalışmalar, verilen bir eğitim seti kullanarak genel

bir yazar doğrulama modeli üretme eğilimindeki çalışmaları temsil etmektedir. Türkçeye miskin olarak çevirmeyi uygun gördüğümüz tembel (lazy) olarak belirtilen çalışmalar ise her yazar doğrulama problemi için ayrı bir öğrenme durumu ele alır.

Tablo 11.6. İngilizce blog külliyatı kullanan çalışmalar ile karşılaştırma

AV yaklaşımı	AV yöntemi	Öğrenme Tipi	c@1	AUC	AUC*c@1
Sıkıştırma Modeli [141]	içsel	miskin	0,852	0,926	0,789
Profil Tabanlı Yöntem [14]	içsel	miskin	0,763	0,847	0,646
GLAD [142]	içsel	eğitilmiş	0,838	0,912	0,764
Sahtekarlar Yöntemi [23]	dışsal	miskin	0,744	0,792	0,589
BBM_kok (FS3)	dışsal	eğitilmiş	0,895	0,957	0,856
BBM_kok (FS2)	dışsal	eğitilmiş	0,892	0,954	0,850
BBM_kok (FS1)	dışsal	eğitilmiş	0,894	0,955	0,843

Yukarıda verilen yaklaşımlar ile kıyaslandığında, ele aldığımız problem özelinde yani yazar doğrulama problemini iki dokümanın yazarlığının doğrulanması olarak ele alan çalışmalar arasında, önerdiğimiz model İngilizce Blog külliyatında, ele alınan üç ölçekte de en yüksek başarılı sonuçları üretmiştir. Bu çalışmada üretilen model ile test edilen verilerin aynı türe (blog verisi) ait olmuş olması elde edilen yüksek başarının en önemli etmenidir. Karşılaştırılan diğer yöntemler ile kıyaslandığında önerilen birleştirme yönteminin de bu problem özelinde başarılı bir karar kriteri oluşturduğu görülmektedir.

Yazar doğrulama çalışmalarının karşılaştırılmasında tür bağımlılık önemli bir özelliktir. Bu sebeple, önerdiğimiz modelin farklı bir türe ait veriler içeren başka bir veri kümesi ile test edilmesi için, 2015 yılında PAN organizasyonu tarafından yayınlanan İngilizce yazar doğrulama veri kümesini kullandık. Bu veri kümesi oyun platformlarından elde edilen, kullanıcıların listesinin ve konuşucuların isimlerinin çıkarıldığı bir iletişim yazıları (dialog lines) topluluğunu barındırmaktadır. Veri kümesi karışık konularda veriler içermekte (cross-topic) ve ele aldığımız yazar doğrulama yapısı gibi her yazara ait bilinen sadece bir doküman bulunmaktadır. Veri kümesinde bulunan dokümanların boyu yaklaşık 500 kelime civarındadır. Veri kümesinin içerisinde eğitim kümesi ve test kümesi olmak üzere iki bölüm bulunmaktadır yani veri yayıncıları eğitim ve test kümelerini belirlemiştir. Eğitim kümesinde 100 örnek, test

kümesinde ise 500 örnek barındırmaktadır bu veri kümesi. Eğitim için verilen örnek sayısı, önerdiğimiz modele uygun bir şekilde arka plan oluşturmaya yetersiz olduğundan bu verinin testi için İngilizce Blog külliyatı, önerdiğimiz modelde arka plan oluşturmak için tekrar kullanılmıştır. PAN kümesindeki örneklerin boyu yaklaşık 500 kelimelik olduğundan biz de arka planda 500 kelimelik doküman çiftleri ürettik. Bu deneyde Logistic Regresyon sınıflandırma algoritmasını ön tanımlı parametreleri ile kullandık. Karakter n-gram'dan daha yüksek başarı elde ettiğimi kelime k-ön kök öznitelik kümelerini (FS1, FS2, FS3) bu deneylerde kullandık. Bu deneylerde farklı olarak bir öznitelik seçme algoritması kullanmadık, öznitelik kümesinde bulunan tüm öznitelikler temsil vektörlerde bulunmaktadır. PAN verilerinden eğitim kümesi, arka planda oluşturulacak modelin yazar sayısı hiper-parametresinin optimizasyonunda kullanılmıştır. Kullanılan her öznitelik kümesi için PAN'ın eğitim verisi 50 ile 1000 yazar arasında farklı yazar sayıları ile test edilmiş ve her öznitelik kümesi için eğitim kümesinden elde edilen en yüksek yazar sayısı PAN'ın test kümesinde yazar sayısı parametresi değeri olarak kullanılmıştır.

PAN organizasyonu, problemin çözümüne önerilen modellerin sıralamasını başarı sırasına göre (AUC*c@1) yayınlamaktadır. Tablo 11.8'de önerdiğimiz modelden elde edilen test başarıları, PAN yayını zamanında sıralanan ilk beş model ve başarısı ve benzer başka karşılaştırılabilir çalışma ve sonuçları sırasıyla verilmiştir.

Tablo 11.7. PAN 2015 İngilizce veri kümesini kullanan yazar doğrulama yaklaşımları ile karşılaştırma

AV yaklaşımı	AV yöntemi	Öğrenme Tipi	c@1	AUC	AUC*c@1	Cevapsız problem sayısı	Çalışma süresi
Geliştirilmiş dışsal AV [143]	dışsal	miskin	0,785	0,812	0,637	-	-
Genelleştirilmiş maske sıyırma [31]	içsel	eğitilmiş	0,76	-	-	-	-
Bagnall [144]	dışsal	miskin	0,757	0,811	0,614	3	21:44:03
Sıkıştırma modeli [141]	içsel	miskin	0,754	0,802	0,605	-	-
BBM_kok (FS1)	dışsal	eğitilmiş	0,708	0,764	0,540	0	00:00:36
Castro D. ve ark [145]	dışsal	miskin	0,694	0,750	0,520	0	02:07:20
BBM_kok (FS3)	dışsal	eğitilmiş	0,69	0,744	0,513	0	00:17:01
Gutierrez ve ark [146]	dışsal	miskin	0,694	0,739	0,513	39	00:37:06
Kocher ve Savoy [147]	dışsal	miskin	0,689	0,738	0,508	94	00:00:24
BBM_kok (FS2)	dışsal	eğitilmiş	0,68	0,739	0,502	0	00:09:02
PAN 15-ensemble [45]	-	-	0,596	0,786	0,468	0	-

PAN 2015 İngilizce veri kümesi eğitim kümesinde çok az sayıda örnek içerdiğinden zorlu bir veri kümesidir. İçerisinde bulunan dokümanların da kısa boyutlu olması bu veri kümesini daha da zorlaştırmaktadır. Söz konusu bu zorluklara ek olarak, PAN eğitim kümesindeki örneklerden hiçbir içerik bilgisi kullanmamamıza rağmen önerdiğimiz modeli kullanarak bu veri kümesinin testinde karşılaştırılabilir sonuçlar elde ettik. Dahası, önerdiğimiz modelde kullanılan BBM blog verileri kullanılarak üretilmiş olmasına rağmen diyalog yazılarının testinde makul sonuçlar üretmiştir. Bu sonuçlar önerdiğimiz BBM'in genelleştirme kabiliyetini ve önerdiğimiz birleştirme yöntemi ve öznitelik kümesinin ayırt edicilik gücünü göstermektedir. Yukarıdaki tabloda bulunan diğer çalışmalar ile karşılaştırıldığında sadece önerdiğimiz model dışsal ve eğitimli bir yaklaşım ile yazar doğrulama problemine çözüm sunmuş olmanın yanında, PAN'ın eğitim kümesinin herhangi bir içerik bilgisini kullanmayarak başarılı sonuçlar üretebilen tek çalışmadır.

Bu bölümde elde edilen sonuçlar, bu tez kapsamında çözülmesi planlanan problemin çözümüne yönelik elde ettiğimiz en önemli ve en anlamlı sonuçlardır. Önceki bölümlerde ele aldığımız problemin çözümüne yardımcı olacağı düşünülen deneyler yapılmış ve her bölümün sonuçları kendi içinde değerlendirilmiştir. Bu bölüm tez çalışmasının ana bölümü olduğundan bu bölümde yapılan deneysel çalışmaların sonuçlarının yorumlanması TARTIŞMA bölümüne bırakılmıştır.

12. TARTIŞMA

Yazar doğrulama, yazar analizi çalışmalarının en zorlu problemlerinden biridir. Bu tez çalışmasında yazar doğrulama problemini verilen iki dokümanın yazarlığının doğrulanması olarak değerlendirdik. Bu bakış açısı ile ele alınan yazar doğrulama problemi genel anlayıştan daha zorlu olan yazar bağımsız bir yazar doğrulama problemine dönüşmektedir. Bu problemin çözümünü sağlamaya yönelik bir altyapı sunmak için literatüre en uygun ön işlemleri uyguladık ve onları bir alt işlem kümesi olarak, önerdiğimiz modelde kullandık. Önerdiğimiz modelin işlem adımları iyi tanımlıdır ve başkaları tarafından kolaylıkla uygulanabilir. İlk işlem olarak orijinal dokümanlar yazar küme büyüklüğü ve veri büyüklüğü bakımından normalize edilmiştir. Bir yazar tarafından farklı zamanlarda yazılmış dokümanlar rastgele olarak birleştirilmiş ve bu yeni oluşum sırasıyla eşit sayıda ve eşit büyüklükte örneklere bölünmüştür. Bu işlem ile yazar küme sayısı ve veri büyüklüğü bakımından veri kümeleri normalize edilmiş ve dokümanlar arası zaman ve konu bağı kırılmış olmaktadır. Böylece yazarların zaman ve konu bağımsız yazma üslubu da ön plana çıkmış olmaktadır. Bir veri kümesi için tüm yazarlara ait örnekler üretildikten sonra bu örneklere öznitelik çıkarımı ve seçimi yöntemleri uygulanmıştır. Öznitelik çıkarımı için bu çalışmada, Türkçe ve İngilizce yazar analizi çalışmalarında en başarılı öznitelikler olarak seçilen özniteliklerin çıkarımı yapılmıştır. Örnekler arası en ayırt edici öznitelik kümesini bulmak için tf-idf öznitelik seçme yöntemi kullanılmıştır. İkinci işlem olarak, model geliştirme aşamasında görülen ve görülmeyen yazarların dokümanları arasındaki üretilen modelin vereceği cevap farkını görebilmek için bir veri bölütleme stratejisi kullanılmıştır. Bu işlem önerilen modelin yazar bağımsızlığını test etmek açısından büyük önem taşımaktadır. Veri kümeleri 3 ayrı kümeye bölünmektedir; model üretim örnekleri için Eğitim kümesi, görülen yazarların örnekleri için Test kümesi ve görülmeyen yazarların örnekleri için Doğrulama kümesi. Üçüncü işlem verilen doküman örneklerinin birleştirilmesi ile "evet/hayır" örneklerinin üretimini içermektedir. Örneklerin üretilmesinde 3 farklı birleştirme yöntemi kullanılmıştır. BBM_fark yöntemi dokümanlardaki özniteliklerin değerlerinin mutlak farkını temel alırken, BBM_carp bu özniteliklerin çarpımlarını, BBM_kok birleştirme yöntemi ise bu özniteliklerin çarpımlarının karekök değerlerini temel almaktadır. Son işlem önerilen modelin üretim ve test aşamalarını içermektedir. Bu çalışmada önerilen model SVM ve LR ile iki farklı sınıflandırma algoritması kullanarak önerilen model üretilmiştir.

Bu tez çalışmasında önerilen model dil bağımsız bir modeldir. Türkçe ve İngilizce Blog külliyatları aynı altyapılar ile test edilmiştir. Elde edilen sonuçlar önerilen modelin dil bağımsızlığını doğrulamaktadır. Ayrıca önerilen model ölçeklenebilir bir modeldir. Aynı dil için bir veri kümesini iki farklı büyüklükte test edilmesi ve her iki boyut için de önemli sonuçların elde edilmiş olması çalışmanın ölçeklenebilirliğini doğrulamaktadır. Bunların yanında önerilen model bilgilendiricidir. Birçok yazar analizi çalışmasının aksine arka planda kullanılan yazar küme sayısı ve veri büyüklüğü arttıkça elde edilen doğruluk oranı da önerilen model ile artmaktadır. Ek olarak önerilen model sürdürülebilirdir. Zamanla aktif olarak kullanılan kelimeler değişse bile arka plana güncel örneklerin eklenmesi ile önerilen modelin sürdürülebilirliği sağlanabilmektedir. Yapılan deneylerde iki dokümanın yazarlığının doğrulanmasında en başarılı sonuçları veren öznitelik kümesi kelime 4-ön kök ve en başarılı birleştirme yöntemi de BBM_kok olarak görülmektedir. Yazar doğrulama probleminin yazar analizi çalışmalarının temel problemi olduğu göz önüne alındığında, bu bulguların bu tür çalışmalara katkı sağlayacağı beklenmektedir.

Bu tez çalışmasındaki en temel çıkarımlardan biri ele alınan probleme uygun bir altyapı kurulması gerekliliğidir. Daha yüksek doğruluk sonuçları farklı öznitelik kümeleri, farklı sınıflandırma algoritmaları ve hatta farklı sınıflandırma parametreleri ile elde edilebilir. Bu çalışmada karşılaşılan en önemli zorluk, model üretiminde kullanılan yazar ve örneklem sayısı arttıkça işlem ve bellek bakımından hesaplama maliyetinin de artıyor olmasıdır. Bu artış düşünüldüğünde önerilen model bir noktadan sonra klasik bilgisayarlar ile üretilemeyecek hale gelecektir. Bu sebeple ileriki çalışmalar daha çok yazar ve örnekler kullanılarak üretilecek modellerin büyük veri platformlarında test edilmesi üzerine de olacaktır.

Yazar analizi problemlerinden en zorlu olan nispeten kısa iki dokümanın yazarlık doğrulamasının yapıldığı bu çalışmada, biz dengeli, ölçeklenebilir ve bilgilendirici bir arka plan ile desteklenen bir model önerdik. Bu arka planın kullanımı ile sistemin karar üretme aşamasında kullanılmak üzere bir BBM oluşturduk. Ayrıca bu çalışmada, önerilen sistemde doküman çiftlerinin değerlendirilebilmesi için yeni, basit ve etkili bir doküman birleştirme yöntemi önerdik.

Bu çalışmada önerdiğimiz model iki kamuya açık İngilizce veri kümesi ile; PAN – 2015 İngilizce yazar doğrulama veri kümesi ve İngilizce Blog külliyatı, ve bir yeni derlenmiş Türkçe veri kümesi ile test edilmiştir. Önerilen model blog külliyatları üzerinde, aynı veri kümesinden farklı yazarlar kullanılarak üretilen BBM ile en başarılı doğruluk sonucunu üretmiştir. Bu

deneyler göstermektedir ki; değerlendirilen tür ile ilgili bilgiye sahip olduğumuz, bu bilgiyi kullandığımız durumda önerilen model ile oldukça başarılı sonuçlar elde edilebilmektedir. Blog verilerini kullanarak üretmiş olduğumuz BBM ile PAN verilerinin testi de yapılmıştır. Bu deneyde PAN verilerindeki eğitim kümesi sadece arka planda kullanılacak yazar sayısını belirlemede kullanılmıştır. PAN verilerinden eğitim kümesinin hiçbir içeriksel bilgi kullanmamış olmamıza rağmen, PAN verilerinin test kümesinden makul sonuçlar elde ettik. Bu deneysel sonuçlar göstermektedir ki; önerilen model, değerlendirilen tür ile ilgili herhangi bir bilgiye sahip olmamasına rağmen yazar doğrulama performansı memnun edici seviyededir. Bu durumda, sadece üretilecek BBM'in ele alınan türe göre parametrelerinin optimize edilmesi gerekmektedir. İngilizce veri kümelerinden elde edilen sonuçların yanı sıra, Türkçe Blog külliyatı kullanılarak gerçekleştirilen deneylerden de elde edilen sonuçlar yüksek başarı göstermektedir.

Yazar doğrulama, yazar analizi çalışmalarında ele alınan sıcak ve popüler bir konudur. Araştırmacılar gün be gün bu konu üzerine yeni çözümler sunmaktadır. Bizim bu çalışmadaki önerimiz, iyi tanımlı bir BBM kullanmanın yazar doğrulama çalışmalarında iyi bir yönlendirici ve teşvik edici olacağıdır. Bu çalışmanın devamında, yazar doğrulama problemlerinin çözümü için kullanılacak BBM gibi yapıların evrensel bir model mi yoksa farklı türlerin birleşiminden oluşan tür bağımlı bir model mi olması gerektiği üzerine araştırmalar yapılacaktır.

KAYNAKLAR

[1] Neal T, Sundararajan K, Woodard D. Exploiting linguistic style as a cognitive biometric for continuous verification. In: *Proceedings - 2018 International Conference on Biometrics, ICB 2018*. IEEE, 2018, pp. 270–276.

[2] Pokhriyal N, Tayal K, Nwogu I, et al. Cognitive-Biometric Recognition from Language Usage: A Feasibility Study. *IEEE Trans Inf Forensics Secur* 2017; 12: 134–143.

[3] Stamatatos E. A survey of modern authorship attribution methods. *J Am Soc Inf Sci Technol* 2009; 60: 538–556.

[4] Juola P. Authorship attribution. *Found Trends Inf Retr* 2006; 1: 233–334.

[5] Neal T, Sundararajan K, Fatima A, et al. Surveying stylometry techniques and applications. *ACM Comput Surv* 2017; 50: 86.

[6] Can F, Çalışkan S. Türkçe Metinler Üzerine Yapılan Sayısal Üslup Araştırmalarını İnceleyen ve Benim Adım Kırmızı Çevirilerinin Aslına Olan Sadakatini Ölçen Bir Çalışma (A Survey of Stylometry Research on Turkish Texts and A Study on Quantification of Loyalty for Translation. *Turk Kutuph - Turkish Librariansh* 2018; 32: 251–286.

[7] Elmanarelbouanani S, Kassou I. Authorship Analysis Studies: A Survey. *Int J Comput Appl* 2014; 86: 22–29.

[8] Stamatatos E, Daelemans W, Verhoeven B, et al. Overview of the author identification task at PAN 2014. In: *CEUR Workshop Proceedings*, pp. 877–897.

[9] Juola P, Stamatatos E. Overview of the Author Identification Task. In: *CLEF 2013 Evaluation Labs and Workshop -- Working Notes Papers*. 2014.

[10] Koppel M, Schlier J, Argamon S. Computational methods in authorship attribution. *J Am Soc Inf Sci Technol* 2009; 60: 9–26.

[11] Koppel M, Schler J, Argamon S. Authorship attribution in the wild. *Lang Resour Eval*. Epub ahead of print 2011. DOI: 10.1007/s10579-009-9111-2.

[12] Fatima M, Anwar S, Naveed A, et al. Multilingual SMS-based author profiling: Data and methods. *Nat Lang Eng*. Epub ahead of print 2018. DOI: 10.1017/S1351324918000244.

[13] Fatima M, Hasan K, Anwar S, et al. Multilingual author profiling on Facebook. *Inf Process Manag*. Epub ahead of print 2017. DOI: 10.1016/j.ipm.2017.03.005.

[14] Potha N, Stamatatos E. A profile-based method for authorship verification. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, 2014, pp. 313–326.

[15] Koppel M, Schler J. Authorship verification as a one-class classification problem. In: *Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004*. ACM, 2004, pp. 489–495.

[16] Nirkhi S, Dharaskar R V., Thakare VM. Authorship Verification of Online Messages for Forensic Investigation. *Phys Procedia* 2016; 78: 640–645.

[17] Iqbal F, Khan LA, Fung BCM, et al. E-mail authorship verification for forensic investigation. In: *Proceedings of the ACM Symposium on Applied Computing*. ACM, 2010, pp. 1591–1598.

[18] Halvani O, Winter C, Pflug A. Authorship verification for different languages, genres and topics. *DFRWS 2016 EU - Proc 3rd Annu DFRWS Eur* 2016; 16: S33–S43.

[19] Stamatatos E. Authorship verification: a review of recent advances. *Res Comput Sci* 2016; 123: 9–25.

[20] Koppel M, Schler J, Argamon S, et al. The 'Fundamental Problem' of Authorship Attribution. *English Stud* 2012; 93: 284–291.

[21] Halvani O, Winter C, Graner L. Unary and Binary Classification Approaches and their Implications for Authorship Verification. *arXiv Prepr arXiv190100399*, https://arxiv.org/pdf/1901.00399.pdf (2018).

[22] Ding SHH, Fung BCM, Iqbal F, et al. Learning stylometric representations for authorship analysis. *IEEE Trans Cybern*. Epub ahead of print 2019. DOI: 10.1109/TCYB.2017.2766189.

[23] Koppel M, Winter Y. Determining if two documents are written by the same author. *J Am Soc Inf Sci Technol* 2014; 65: 178–187.

[24] Can F, Patton JM. Change of writing style with time. *Comput Hum* 2004; 38: 61–82.

[25] Klaussner C, Vogel C. Stylochronometry: Timeline Prediction in Stylometric Analysis. In: *Research and Development in Intelligent Systems XXXII*. 2015. Epub ahead of print 2015. DOI: 10.1007/978-3-319-25032-8_6.

[26] Stamou C. Stylochronometry: Stylistic development, sequence of composition, and relative dating. *Lit Linguist Comput*. Epub ahead of print 2008. DOI: 10.1093/llc/fqm029.

[27] Stuart LM, Tazhibayeva S, Wagoner AR, et al. On identifying authors with style. In: *Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013*. 2013. Epub ahead of print 2013. DOI: 10.1109/SMC.2013.520.

[28] Zoeten R. *Computational Stylometry in Adversarial Settings*. University of Amsterdam, https://esc.fnwi.uva.nl/thesis/centraal/files/f1650865434.pdf (2015, accessed 14 October 2019).

[29] Potha N, Stamatatos E. An improved impostors method for authorship verification. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, 2017, pp. 138–144.

[30] Seidman S. Authorship verification using the impostors method: Notebook for PAN at CLEF 2013. In: *CEUR Workshop Proceedings*. Citeseer, 2013.

[31] Bevendorff J, Stein B, Hagen M, et al. Generalizing Unmasking for Short Texts. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, pp. 654–659.

[32] Kestemont M, Luyckx K, Daelemans W, et al. Cross-Genre Authorship Verification

Using Unmasking. *English Stud* 2012; 93: 340–356.

[33] Koppel M, Schier J, Bonchek-Dokow E. Measuring differentiability: Unmasking pseudonymous authors. *J Mach Learn Res* 2007; 8: 1261–1276.

[34] Sanderson C, Guenter S. Short text authorship attribution via sequence kernels, Markov chains and author unmasking: An investigation. In: *COLING/ACL 2006 - EMNLP 2006: 2006 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*. Association for Computational Linguistics, 2006, pp. 482–491.

[35] Mayor C, Gutierrez J, Toledo A, et al. A single author style representation for the author verification task: Notebook for PAN at CLEF 2014. In: *CEUR Workshop Proceedings*. 2014, pp. 1079–1083.

[36] Brocardo ML, Traore I, Woungang I. Authorship verification of e-mail and tweet messages applied for continuous authentication. *J Comput Syst Sci* 2015; 81: 1429–1440.

[37] Brocardo ML, Traore I, Saad S, et al. Authorship verification for short messages using stylometry. In: *2013 International Conference on Computer, Information and Telecommunication Systems, CITS 2013*. IEEE, 2013, pp. 1–6.

[38] Abbasi A, Chen H. Applying authorship analysis to extremist-group Web forum messages. *IEEE Intell Syst* 2005; 20: 67–75.

[39] Chaski CE, D P. Who ' s At The Keyboard ? Authorship Attribution in Digital Evidence Investigations. *Int J* 2005; 4: 1–13.

[40] Stein B, Lipka N, Prettenhofer P. Intrinsic plagiarism analysis. *Lang Resour Eval* 2011; 45: 63–82.

[41] Burrows S, Uitdenbogerd AL, Turpin A. Comparing techniques for authorship attribution of source code. *Softw - Pract Exp* 2014; 44: 1–32.

[42] Frantzeskou G, Stamatatos E, Gritzalis S, et al. Effective identification of source code authors using byte-level information. In: *Proceedings - International Conference on Software Engineering*. ACM, 2006, pp. 893–896.

[43] Aslantürk O. *Tamgacı: Artırımsal ve Geri Beslemeli Türkçe Yazar Çözümleme*. Hacettepe Üniversitesi, Bilgisayar Mühendisliği Bölümü., 2014.

[44] Stamatatos E, Potthast M, Rangel F, et al. Overview of the PAN/CLEF 2015 evaluation lab. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, 2015, pp. 518–538.

[45] Stamatatos E, Daelemans W, Verhoeven B, et al. Overview of the author identification task at PAN 2015. In: *CEUR Workshop Proceedings*. 2015, pp. 1–21.

[46] Raghunadha Reddy T, Vishnu Vardhan B, Vijayapal Reddy P. A survey on Authorship Profiling techniques. *Int J Appl Eng Res* 2016; 11: 3092–3102.

[47] Rangel F, Rosso P, Koppel M, et al. Overview of the author profiling task at PAN 2013. *Noteb Pap CLEF* 2013; 23–26.

[48] Türkoğlu F, Diri B, Amasyali MF. Author attribution of turkish texts by feature mining. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial*

Intelligence and Lecture Notes in Bioinformatics). Berlin, Heidelberg: Springer, pp. 1086–1093.

[49] Fatih Amasyali M, Diri B. Automatic Turkish text categorization in terms of author, genre and gender. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp. 221–226.

[50] Canbay P, Sezer EA, Sever H. Authorship modelling approach for authorship verification on the Turkish texts. In: *26th IEEE Signal Processing and Communications Applications Conference, SIU 2018*. IEEE, 2018, pp. 1–4.

[51] Aslantürk O, Sezer EA, Sever H, et al. Application of cascading rough set-based classifiers on authorship attribution. In: *Proceedings - 2010 IEEE International Conference on Granular Computing, GrC 2010*. IEEE, 2010, pp. 656–660.

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

[53] Kuzu RS, Salah AA. Chat biometrics. *IET Biometrics* 2018; 7: 454–466.

[54] Kuzu RS, Balci K, Salah AA. Authorship recognition in a multiparty chat scenario. In: *Proceedings - 2016 4th International Workshop on Biometrics and Forensics, IWBF 2016*. IEEE, 2016, pp. 1–6.

[55] Argamon S, Juola P. Overview of the international authorship identification competition at PAN-2011. In: *CEUR Workshop Proceedings*. 2011.

[56] Klimt B, Yang Y. Introducing the Enron Corpus. *Mach Learn*.

[57] Zheng R, Li J, Chen H, et al. A framework for authorship identification of online messages: Writing-style features and classification techniques. *J Am Soc Inf Sci Technol*. Epub ahead of print 2006. DOI: 10.1002/asi.20316.

[58] Jain AK. Data clustering: 50 years beyond K-means. *Pattern Recognit Lett*. Epub ahead of print 2010. DOI: 10.1016/j.patrec.2009.09.011.

[59] Cortes C, Vapnik V. Support-vector networks. *Mach Learn* 1995; 20: 273–297.

[60] Wu X, Kumar V, Ross QJ, et al. Top 10 algorithms in data mining. *Knowl Inf Syst* 2008; 14: 1–37.

[61] Teng GF, Lai MS, Ma J Bin, et al. E-mail authorship mining based on SVM for computer forensic. In: *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*. IEEE, 2004, pp. 1204–1207.

[62] Diederich J, Kindermann J, Leopold E, et al. Authorship attribution with support vector machines. *Appl Intell* 2003; 19: 109–123.

[63] Nilsson NJ. *Principles of Artificial Intelligence*. Morgan Kaufmann, 1981. Epub ahead of print 1981. DOI: 10.1109/TPAMI.1981.4767059.

[64] Russell SJ, Norvig P. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited, 2016.

[65] Houvardas J, Stamatatos E. N-gram feature selection for authorship identification. In:

Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer, 2006, pp. 77–86.

[66] Canbay P, Sever H, Akcapinar Sezer E. Determining of discriminative blog size for authorship attribution on the Turkish texts. In: *6th International Symposium on Digital Forensic and Security, ISDFS 2018 - Proceeding*. IEEE, 2018, pp. 1–5.

[67] Rosso P. Author profiling and Plagiarism detection. In: *Communications in Computer and Information Science*. 2015. Epub ahead of print 2015. DOI: 10.1007/978-3-319-25485-2_6.

[68] Luyckx K, Daelemans W. The effect of author set size and data size in authorship attribution. *Lit Linguist Comput* 2011; 26: 35–55.

[69] Rocha A, Scheirer WJ, Forstall CW, et al. Authorship Attribution for Social Media Forensics. *IEEE Transactions on Information Forensics and Security*. Epub ahead of print 2017. DOI: 10.1109/TIFS.2016.2603960.

[70] Cheng N, Chandramouli R, Subbalakshmi KP. Author gender identification from text. *Digit Investig*. Epub ahead of print 2011. DOI: 10.1016/j.diin.2011.04.002.

[71] Chen X, Hao P, Chandramouli R, et al. Authorship similarity detection from email messages. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, 2011, pp. 375–386.

[72] Burrows J. All the way through: Testing for authorship in different frequency strata. *Lit Linguist Comput*. Epub ahead of print 2006. DOI: 10.1093/llc/fqi067.

[73] Kleinbaum DG, Dietz K, Gail M, et al. *Logistic regression*. Springer, 2002.

[74] Hosmer DW, Lemeshow S, Sturdivant RX. *Applied Logistic Regression: Third Edition*. 2013. Epub ahead of print 2013. DOI: 10.1002/9781118548387.

[75] Harrel Jr. FE. Regression Modeling Strategies - With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis. *R Softw*. Epub ahead of print 2015. DOI: 10.1007/978-3-319-19425-7_1.

[76] Madigan D, Genkin A, Lewis DD, et al. Bayesian multinomial logistic regression for author identification. In: *AIP Conference Proceedings*. AIP, 2005, pp. 509–516.

[77] Pavlov YL. *Random forests*. 2019.

[78] Breiman L. Randomforest2001. *Mach Learn*. Epub ahead of print 2001. DOI: 10.1017/CBO9781107415324.004.

[79] Quinlan JR. *C4. 5 programs for machine learning. Elsevier*. Elsevier, 2014.

[80] Breiman L, Friedman JH, Olshen RA, et al. *Classification and regression trees*. Routledge, 2017. Epub ahead of print 2017. DOI: 10.1201/9781315139470.

[81] Palomino-Garibay A, Camacho-González AT, Fierro-Villaneda RA, et al. A random forest approach for authorship profiling. In: *CEUR Workshop Proceedings*. 2015.

[82] Ng AY, Jordan MI. On discriminative vs. Generative classifiers: A comparison of logistic regression and naive bayes. In: *Advances in Neural Information Processing*

Systems. 2002, pp. 841–848.

[83] Rish I. IBM Research Report An empirical study of the naive Bayes classifier. In: *Science*. 2001, pp. 41–46.

[84] Altheneyan AS, Menai MEB. Naïve Bayes classifiers for authorship attribution of Arabic texts. *J King Saud Univ - Comput Inf Sci* 2014; 26: 473–484.

[85] Coyotl-Morales RM, Villaseñor-Pineda L, Montes-y-Gómez M, et al. Authorship attribution using word sequences. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, 2006, pp. 844–853.

[86] Lancaster HO, Seneta E. Chi-square distribution. *Encycl Biostat*; 2.

[87] Bellù LG, Liberati P. Inequality Analysis: The Gini Index. *EASYPol Modul 040*. Epub ahead of print 2003. DOI: 10.1016/0306-4573(92)90089-I.

[88] Abbasi A, Altmann J, Hossain L. Identifying the effects of co-authorship networks on the performance of scholars: A correlation and regression analysis of performance measures and social network analysis measures. *J Informetr*. Epub ahead of print 2011. DOI: 10.1016/j.joi.2011.05.007.

[89] Müller-Funk U. Data Analysis, Machine Learning and Applications. In: *Data Analysis, Machine Learning and Applications*. 2008. Epub ahead of print 2008. DOI: 10.1007/978-3-540-78246-9.

[90] Karegowda AG, Manjunath AS, Jayaram MA. Comparative Study of Attribute Selection Using Gain Ratio and Correlation Based Feature Selection. *Int J Inf Technol Knowl Manag* 2010; 2: 271–277.

[91] Aizawa A. An information-theoretic perspective of tf–idf measures. *Inf Process Manag* 2003; 39: 45–65.

[92] Domeniconi G, Moro G, Pasolini R, et al. A study on term weighting for text categorization: A novel supervised variant of tf.idf. In: *DATA 2015 - 4th International Conference on Data Management Technologies and Applications, Proceedings*. 2015, pp. 26–37.

[93] Chen K, Zhang Z, Long J, et al. Turning from TF-IDF to TF-IGM for term weighting in text classification. *Expert Syst Appl* 2016; 66: 1339–1351.

[94] Tutkan M, Ganiz MC, Akyokuş S. Helmholtz principle based supervised and unsupervised feature selection methods for text mining. *Inf Process Manag* 2016; 52: 885–910.

[95] Iqbal F, Binsalleeh H, Fung BCM, et al. Mining writeprints from anonymous e-mails for forensic investigation. *Digit Investig* 2010; 7: 56–64.

[96] Litvak M. Deep dive into authorship verification of email messages with convolutional neural network. In: *Communications in Computer and Information Science*. 2019. Epub ahead of print 2019. DOI: 10.1007/978-3-030-11680-4_14.

[97] Boenninghoff B, Nickel RM, Zeiler S, et al. Similarity Learning for Authorship Verification in Social Media. In: *ICASSP, IEEE International Conference on Acoustics,*

Speech and Signal Processing - Proceedings. 2019. Epub ahead of print 2019. DOI: 10.1109/ICASSP.2019.8683405.

[98] Varela P, Justino E, Britto A, et al. A computational approach for authorship attribution of literary texts using sintatic features. In: *Proceedings of the International Joint Conference on Neural Networks*. 2016. Epub ahead of print 2016. DOI: 10.1109/IJCNN.2016.7727835.

[99] Dunn J, Argamon S, Rasooli A, et al. Profile-based authorship analysis. *Digit Scholarsh Humanit*. Epub ahead of print 2016. DOI: 10.1093/llc/fqv019.

[100] Afroz S, Caliskan-Islam A, Stolerman A, et al. Doppelgänger finder: Taking stylometry to the underground. In: *Proceedings - IEEE Symposium on Security and Privacy*. 2014. Epub ahead of print 2014. DOI: 10.1109/SP.2014.21.

[101] Ahmad Z, Zhang J. Selective combination of multiple neural networks for improving model prediction in nonlinear systems modelling through forward selection and backward elimination. *Neurocomputing*. Epub ahead of print 2009. DOI: 10.1016/j.neucom.2008.02.005.

[102] Rosso P, Potthast M, Stein B, et al. Evolution of the PAN Lab on Digital Text Forensics. 2019. Epub ahead of print 2019. DOI: 10.1007/978-3-030-22948-1_19.

[103] Stamatatos E, Kokkinakis G, Fakotakis N. Automatic text categorization in terms of genre and author. *Comput Linguist* 2000; 26: 471–495.

[104] Adamovic S, Miskovic V, Milosavljevic M, et al. Automated language-independent authorship verification (for Indo-European languages). *J Assoc Inf Sci Technol* 2019; 70: 858–871.

[105] Abbasi A, Chen H. Applying authorship analysis to extremist-group Web forum messages. *IEEE Intelligent Systems*. Epub ahead of print 2005. DOI: 10.1109/MIS.2005.81.

[106] Shrestha P, Sierra S, González FA, et al. Convolutional neural networks for authorship attribution of short texts. In: *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference*. 2017. Epub ahead of print 2017. DOI: 10.18653/v1/e17-2106.

[107] Jafariakinabad, Fereshteh, Tarnpradab S, Hua KA. Syntactic Neural Model for Authorship Attributione. In: *The Thirty-Third International Flairs Conference*. 2020.

[108] Brocardo ML, Traore I, Saad S, et al. Authorship verification for short messages using stylometry. In: *2013 International Conference on Computer, Information and Telecommunication Systems, CITS 2013*. 2013. Epub ahead of print 2013. DOI: 10.1109/CITS.2013.6705711.

[109] Jafariakinabad F, Hua KA. Style-aware neural model with application in authorship attribution. In: *Proceedings - 18th IEEE International Conference on Machine Learning and Applications, ICMLA 2019*. 2019. Epub ahead of print 2019. DOI: 10.1109/ICMLA.2019.00061.

[110] Brocardo ML, Traore I, Woungang I, et al. Authorship verification using deep belief network systems. *Int J Commun Syst*. Epub ahead of print 2017. DOI: 10.1002/dac.3259.

[111] Stamatatos E, Daelemans W, Verhoeven B, et al. Overview of the author identification task at PAN 2015. In: *CEUR Workshop Proceedings*. 2015, pp. 1–8.

[112] Liu W, Wang Z, Liu X, et al. A survey of deep neural network architectures and their applications. *Neurocomputing*. Epub ahead of print 2017. DOI: 10.1016/j.neucom.2016.12.038.

[113] Dauber E, Caliskan A, Harang R, et al. Poster: Git blame who?: Stylistic authorship attribution of small, incomplete source code fragments. *Proc - Int Conf Softw Eng* 2018; 2019: 356–357.

[114] Caliskan-Islam A, Harang R, Liu A, et al. De-anonymizing Programmers via Code Stylometry. In: *USENIX sec*, pp. 255–270.

[115] Burrows S, Tahaghoghi SMM. Source code authorship attribution using n-grams. In: *ADCS 2007 - Proceedings of the Twelfth Australasian Document Computing Symposium*. Citeseer, 2007, pp. 32–39.

[116] Iqbal F, Hadjidj R, Fung BCM, et al. A novel approach of mining write-prints for authorship attribution in e-mail forensics. *DFRWS 2008 Annu Conf* 2008; 5: S42–S51.

[117] Schmid MR, Iqbal F, Fung BCM. E-mail authorship attribution using customized associative classification. *Proc Digit Forensic Res Conf DFRWS 2015 USA* 2015; 14: S116–S126.

[118] Altakrori MH, Iqbal F, Fung BCM, et al. Arabic authorship attribution: An extensive study on twitter posts. *ACM Trans Asian Low-Resource Lang Inf Process* 2018; 18: 5.

[119] Layton R, Watters P, Dazeley R. Authorship attribution for Twitter in 140 characters or less. In: *Proceedings - 2nd Cybercrime and Trustworthy Computing Workshop, CTC 2010*. IEEE, 2010, pp. 1–8.

[120] Kucukyilmaz T, Cambazoglu BB, Aykanat C, et al. Chat mining: Predicting user and message attributes in computer-mediated communication. *Inf Process Manag* 2008; 44: 1448–1466.

[121] Overdorf R, Greenstadt R. Blogs, Twitter Feeds, and Reddit Comments: Cross-domain Authorship Attribution. *Proc Priv Enhancing Technol* 2016; 2016: 155–171.

[122] Holmes DI. The Evolution of Stylometry in Humanities Scholarship. *Lit Linguist Comput* 1998; 13: 111–117.

[123] Stolerman A, Overdorf R, Afroz S, et al. Breaking the closed-world assumption in stylometric authorship attribution. In: *IFIP Advances in Information and Communication Technology*. Springer, 2014, pp. 185–205.

[124] Stein B, Lipka N, Zu Eissen SM. Meta analysis within authorship verification. In: *Proceedings - International Workshop on Database and Expert Systems Applications, DEXA*. IEEE, 2008, pp. 34–39.

[125] Koppel M, Schier J, Argamon S, et al. Authorship attribution with thousands of candidate authors. In: *Proceedings of the Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Citeseer, 2006, pp. 659–660.

[126] Hirst G, Feiguina O. Bigrams of syntactic labels for authorship discrimination of short texts. *Lit Linguist Comput* 2007; 22: 405–417.

[127] Marton Y, Wu N, Hellerstein L. On compression-based text classification. In: *European Conference on Information Retrieval*. Springer, 2005, pp. 300–314.

[128] Stamatatos E. Author identification: Using text sampling to handle the class imbalance problem. *Inf Process Manag* 2008; 44: 790–799.

[129] Schler J, Koppel M, Argamon S, et al. Effects of age and gender on blogging. In: *AAAI Spring Symposium - Technical Report*. 2006.

[130] Reynolds DA. Speaker identification and verification using Gaussian mixture speaker models. *Speech Commun* 1995; 17: 91–108.

[131] Reynolds DA, Quatieri TF, Dunn RB. Speaker verification using adapted Gaussian mixture models. *Digit Signal Process A Rev J*. Epub ahead of print 2000. DOI: 10.1006/dspr.1999.0361.

[132] Pacheco ML, Fernandes K, Porco A. Random forest with increased generalization: A universal background approach for authorship verification. In: *CEUR Workshop Proceedings*. 2015.

[133] Grieve J. Quantitative authorship attribution: An evaluation of techniques. *Lit Linguist Comput* 2007; 22: 251–270.

[134] Van Halteren H. Author Verification by Linguistic Profiling: An Exploration of the Parameter Space. *ACM Trans Speech Lang Process* 2007; 4: 1–17.

[135] Halvani O, Steinebach M, Zimmermann R. Authorship verification via k-nearest neighbor estimation: Notebook for PAN at CLEF 2013. *CEUR Workshop Proc*; 1179.

[136] Can F, Kocberber S, Balcik E, et al. Information retrieval on turkish texts. *J Am Soc Inf Sci Technol* 2008; 59: 407–421.

[137] Kılınç D, Özçift A, Bozyigit F, et al. TTC-3600: A new benchmark dataset for Turkish text categorization. *J Inf Sci* 2017; 43: 174–185.

[138] Ando T, Li CK, Mathias R. Geometric means. *Linear Algebra Appl*. Epub ahead of print 2004. DOI: 10.1016/j.laa.2003.11.019.

[139] Kirkwood TBL. Geometric Means and Measures of Dispersion. *Source: Biometrics*. Epub ahead of print 1979. DOI: http://www.jstor.org/stable/2530139.

[140] Arsigny V, Fillard P, Pennec X, et al. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM J Matrix Anal Appl*. Epub ahead of print 2006. DOI: 10.1137/050637996.

[141] Halvani O, Winter C, Graner L. On the usefulness of compression models for authorship verification. In: *ACM International Conference Proceeding Series*. 2017. Epub ahead of print 2017. DOI: 10.1145/3098954.3104050.

[142] Hürlimann M, Weck B, Van Den Berg E, et al. GLAD: Groningen lightweight authorship detection. In: *CEUR Workshop Proceedings*. 2015.

[143] Potha N, Stamatatos E. Improved algorithms for extrinsic author verification. *Knowl Inf*

Syst. Epub ahead of print 2019. DOI: 10.1007/s10115-019-01408-4.

[144] Bagnall D. Author identification using multi-headed recurrent neural networks. In: *CEUR Workshop Proceedings*. 2015.

[145] Castro D, Adame Y, Pelaez M, et al. Authorship verification, combining linguistic features and different similarity functions. In: *CEUR Workshop Proceedings*. 2015.

[146] Gutierrez J, Casillas J, Ledesma P, et al. Homotopy based classification for author verification task. In: *CEUR Workshop Proceedings*. 2015.

[147] Kocher M, Savoy J. UniNE at CLEF 2015: Author Identification. In: *CEUR Workshop Proceedings*. 2015.