Journal of Investigations on Engineering & Technology
Volume 4, Issue 2, 36-46, 2021
ISSN: 2687-3052

Araştırma Makalesi

Kısa Metinleri Yazıldıkları Dile Göre Sınıflandırma ve Farklı Öznitelik Seçim Yöntemlerinin Uygulanması

Murat ASLANYÜREK[1], Altan MESUT[2]

(Alınış / Received: 27.09.2021, Kabul / Accepted: 23.12.2021, Online Yayınlanma / Published Online: 31.12.2021)

Anahtar Kelimeler
Dil tanıma
Fasttext
Langdetect
Makine öğrenmesi

Öz: Özet: Bu çalışmada Wikipedia makale özetlerinden oluşan farklı boyutlardaki iki veri seti üzerinde dil tanımaya yönelik sınıflandırma işlemi yapılmıştır. A veri seti grubu 204 bayt ve daha kısa makale özetlerinden oluşurken, B veri seti grubu 204 ile 512 bayt arasındaki özetlerden oluşmaktadır. Çalışmadaki birinci hedef kısa metinlerin boyutlarına göre uygun makine öğrenmesi ve öznitelik seçme yönteminin belirlenmesidir. İkinci hedef ise en hızlı ve yüksek doğrulukla sınıflandırma yapan yöntemin tespit edilmesidir. Yapılan testler sonucunda öznitelik seçiminde SelectFromModel-Lojistik Regresyon kullanılması ile en yüksek doğruluk değerine ulaşılırken, makine öğrenmesi yöntemi olarak Naive Bayes Multinominal ve Naive Bayes Bernoilli farklı uzunluktaki veri setlerine göre birbirlerine üstünlük sağlamaktadır. Ayrıca çalışmada kullanılan tüm sınıflandırma yöntemleri ile yapılan testler sonucunda, her iki veri setinde diğer sınıflandırma yöntemlerine göre fasttext'in doğruluk bakımından, Kelime Tabanlı İstatistiksel Yöntem (KTİY)'nin ise hız bakımından üstünlük sağladığı anlaşılmıştır.

Classification of Short Texts According to the Language They Are Written in and Application of Different Attribute Selection Methods

Keywords
Language recognition
Fasttext
Langdetect
Machine learning

Abstract: In this study, a classification process for language recognition has been performed on two data sets of different sizes consisting of Wikipedia article abstracts. Dataset group A consists of article abstracts of 204 bytes and less, while dataset group B consists of abstracts of between 204 and 512 bytes. The first goal of the study is to determine the appropriate machine learning and attribute selection method according to the sizes of the short texts. The second goal is to determine the fastest and most accurate classification method. As a result of the tests performed; the highest accuracy value has been achieved by using SelectFromModel-Logistic Regression in feature selection, while as a machine learning method, Naive Bayes Multinominal and Naive Bayes Bernoilli have been superior to each other according to data sets of different lengths. In addition, as a result of the tests performed with all classification methods used in the study, it has been understood that fasttext is superior in terms of accuracy and Word-Based Statistical Method WBSM in terms of speed in both data sets compared to other classification methods.

[1] Kırklareli Üniversitesi, Pınarhisar MYO, Bilgisayar Programcılığı Programı, Kırklareli, Türkiye, m.aslanyurek@klu.edu.tr
[2] Trakya Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Edirne, Türkiye, altanmesut@trakya.edu.tr

Sorumlu yazar: Murat ASLANYÜREK, (m.aslanyurek@klu.edu.tr)

1. Giriş

Doğal dil işlemenin alt konularından olan Dil Tanıma (DT) problemi ile ilgili geçmişten günümüze birçok çalışma yapılmıştır. Bir belgenin yazıldığı dilinin bilinmesi ya da bir belge üzerindeki farklı dillere ait metinlerin dillerine göre gruplanması bir sınıflandırma problemidir. Metinlerin dillerine göre gruplanması onlar üzerinde yapılacak ön işlem adımlarını kolaylaştırmakla beraber, arama, sıkıştırma gibi işlemleri de kolaylaştıracaktır. Dil tanıma ya da metin sınıflandırma işlemleri denetimli ve denetimsiz öğrenme yöntemleri ile yapıldığı gibi, istatistiksel farklı yaklaşımlarla da yapılabilir. Dil tanıma problemlerinde ki en temel unsur, ayırt edici özelliklerin belirlenmesidir. Bu anlamda istatistiksel yöntemler ile yapılan DT problemleri için o dile ait sık geçen kelimelerin belirlenmesi önemlidir. DT işlemlerinde makine öğrenmesi yöntemleri kullanıldığında öznitelik seçimin etkili yapılması hem işlem süresini hem kısaltması hem de sınıflandırma başarısını arttırması beklenmektedir. Öznitelik seçimi, bir veri kümesindeki özniteliklerin, sınıflandırma başarısına en çok katkı sağlayacak olanların seçilmesi olarak tanımlanabilir. Böylelikle veri kümesini en iyi ifade edecek alt küme oluşturulması hedeflenir. Bu anlamda öznitelik seçimiyle alakalı literatürde birçok farklı yaklaşım ve teknik uygulanmaktadır. Gülşen ve ark., Türk kullanıcıların web günlük verilerine dayalı olarak cinsiyet tahmini yaptıkları sınıflandırma çalışmalarında öznitelik çıkarma tekniği olarak Bilgi kazanımı ve Ki-Kare tabanlı öznitelik seçim teknikleri kullanarak, Lojistik Regresyon (LR) sınıflandırıcısı ile yüksek başarımlı bir sınıflandırma yapmışlardır [1]. Yengi ve Omuca LR sınıflandırıcısının performansını arttırmak için öznitelik seçimi ve öznitelik azaltma yöntemlerini (wrapper, filter vb.) kullanmışlardır. Bu sayede LR sınıflandırıcının performansını DVM (Destek Vektör Makineleri) gibi güçlü bir sınıflandırıcının performansına eşdeğer hale getirmeyi başarmışlardır [2]. Parlar ve ark., Twitter verilerini kullanarak oluşturdukları bir veri kümesinde duygu analizi sınıflandırması için bir çalışma yapmışlardır. Bu çalışmada sınıflandırma başarısını arttırmak için farklı öznitelik seçim yöntemlerini kullanmışlardır. Kullandıkları Bilgi Kazanımı, Ki-Kare modeli, Karınca Kolonisi Optimizasyonu ve Sorgu Genişletme yöntemlerinden, en iyi sonucu veren yöntemlerin Karıca Kolonisi Optimizasyonu ve Sorgu Genişletme yöntemleri olduğunu tespit etmişlerdir [3]. Sel ve ark., yaptıkları çalışmada öznitelik seçim tekniği kullanmadan oluşan 600 öznitelik ve Karşılıklı Bilgi öznitelik seçim yöntemini kullanarak 20, 50, 100 ve 200 öznitelik kullanarak sınıflandırma yapmışlardır. 50 öznitelik kullanıldığı durumda en yüksek sınıflandırma doğruluğuna ulaştıkları çalışma sonucunda anlaşılmıştır [4]. Erdem ve Özgür, sınıflandırma başarısını arttırmak ve sınıflandırma süresini azaltmak için Genetik Algoritma kullanan bir öznitelik seçim yöntemi önermişlerdir. Önerdikleri yöntemi birçok sınıflandırıcı ile kullanarak yöntemin başarısını daha önce yapılan çalışmalar ile karşılaştırmışlardır. Karşılaştırmalar sonucunda önerdikleri yöntemin yüksek doğruluk ve düşük çalışma zamanı elde ederek doğrulamışlardır [5]. Akyol meme kanseri tanısı için yaptığı sınıflandırma çalışmasında özyinelemeli öznitelik seçim yöntemi kullanarak, elde ettiği öznitelik kümesini Rastgele Orman ve Lojistik Regresyon sınıflandırıcısında kullanmıştır. Sınıflandırma güvenilirliğini test etmek için kullandığı 5 kat çapraz doğrulama tekniği ile Rastgele Orman sınıflandırıcının doğruluğunu %98 olarak ölçmüştür [6]. Ataş ve ark., Alzheimer hastalığını erken teşhisi için yaptıkları çalışmada öznitelik seçimi olarak Değişken Komşuluk Arama yöntemini ve sınıflandırıcı olarak DVM kullanmışlardır. Alzhemier teşhisinde bu iki model kullanıldığında, benzer çalışmalara göre daha iyi sonuç verdiği testler sonucu anlaşılmıştır [7]. Kaya ve ark., dil tanıma üzerinde yaptıkları bir çalışmada UTF-8 değerlerini karşılaştırarak ikili örüntüler elde etmeye dayanan ve bunları kullanarak dili tanıyan yeni bir öznitelik seçme yöntemi önermişlerdir. Önerilen bu yöntem Almanca, Fransızca, İngilizce ve Türkçe dillerinde Yapay Sinir Ağları (YSA) kullanarak sınıflandırma başarısı test edilmiştir. Testler sonucunda %99 ve %89 doğruluk ile sınıflandırma yapmayı başarmışlardır [8].

Bu çalışmada 6 farklı dile ait farklı boyutlardaki veri setleri üzerinde istatistiksel yöntem, makine öğrenmesi, langdetect ve fasttext yöntemleri kullanılarak kısa metinler dillerine göre sınıflandırılmıştır. Makine öğrenmesi yöntemleri kullanılarak yapılan sınıflandırmalarda farklı öznitelik seçme yöntemleri uygulanmıştır.

2. Yöntemler

2.1. Sınıflandırma Yöntemleri

2.1.1. Naive Bayes Sınıflandırıcı

Daha önceden sınıf etiketleri bilinen veri seti kullanılarak öğrenme modeli oluşturma temeline dayanan bu model Bayes teoriminden esinlenerek geliştirilmiştir [9]. Bu yaklaşımda farklı hesaplama dağılımları kullanılabilmektedir. Genellikle belge sınıflandırmasında kullanılan ve multinom dağılımına dayanan model Multinominal Naive Bayes (M-NB) olarak bilinir. M-NB gibi çalışan ancak Bernoilli dağılımına dayanan model ise Bernoilli Naive Bayes (B-NB) olarak bilinir. Bu iki yöntem arasındaki diğer bir fark B-NB ikili terim özniteliklerinden faydalanırken, M-NB terim frekanslarını kullanmaktadır. Her bir sınıfla ilişkili olan sürekli değerlerin Gauss dağılımına göre dağıtılmasına dayanan model ise Gaussian Naive Bayes (G-NB) olarak bilinir.

2.1.2. Karar Ağaçları

Sınıflandırma ve regresyon çalışmalarında kullanılan ağaç tabanlı bir algoritmadır. Denetimli bir öğrenme metoduna dayanan bu yöntemde amaç, veri özelliklerinden çıkarılan basit karar kurallarını öğrenerek bir hedef değişkenin değerini tahmin eden bir model oluşturmaktır. Tahmin kuralının oluşturulması "böl ve yönet" yaklaşımı yinelemeli bir şekilde bölümlere ayırarak gerçekleşir [10].

2.1.3. K-En Yakın Komşu

Fix tarafından 1951'de geliştirilen [11] ve Cover ve Hart tarafından iyileştirilen bu modelde sınıf tahmini yapmak için bağımsız değişkenlerden oluşan vektöre en yakın olan komşuların uzaklık derecesine bakılır [12]. Hem sınıflandırma hem de regresyon için kullanılabilir [13].

2.1.4. Destek Vektör Makinesi

İlk fikir olarak 1960'lı yıllarda temeli atılan ve 1970' li yıllara kadar geliştirilerek devam eden Destek Vektör Makineleri (DVM) asıl başarısına 1990'lı yıllarda ulaşmıştır. Cortes ve Vapnik' in çalışmaları ile popüler hale gelen bu model sınıflandırma ve regresyon analizi için verileri analiz eden denetimli bir modeldir [14]. Eğitim ve sınıf verisi çok olduğu durumlarda genellikle model oluşturma süresi fazladır [15].

2.1.5. Langdetect

Python açık kaynak kitaplığı langdetect [16], başlangıçta dil tanımlaması için açık koynak kodlu bir Java kitaplığı olarak geliştirilmiştir. Karakter n-gram özelliklerine sahip saf bir Bayes algoritması kullanır.

2.1.6. Fasttext

Hızlı ve etkili sınıflandırma yöntemi olan fasttext, Facebook çalışma grubu tarafından geliştirilmiştir. Word2Vec yöntemine dayanan bu model de kelimeler n-gramlar şeklinde ifade edilir. Word2Vec gibi yöntemlerde her kelime bir vektöre dönüştürülürken, fastext ile n-gramlar vektöre dönüştürülür. Yapısında çok fazla kelime barındıran ve seyrek kelimelerin çok olduğu dillerde n-gramlar ile çalışmayan yöntemler başarısız olmaktadır. Fastext ile bu problem giderilmiştir [17].

2.1.7. Kelime Tabanlı İstatistiksel Yöntem

Paulsen ve Martino' nun yaptıkları patent çalışmasına göre bu yöntemde her dil için bir kelime tablosu tutulur. Kelime tablolarında ilgili dille ilgili en sık geçen etkisiz kelimeler (stopwords) kullanılır. Bir metnin ait olduğu dil belirlenirken metindeki her kelime tüm kelime tablolarında aranır ve hangi dilin kelime tablosunda bulunursa o kelime tablosuna ait sayaç değişkeninin değeri arttırılır. Böylece hangi dilin kelime tablosuna ait sayaç değeri daha yüksek ise metin o dile ait olacaktır [18].

Bu çalışmada dillere ait her bir kelime tablosu 256 kelimeden oluşmaktadır. 256 kelime olmasının sebebi daha sonraki yapılacak olan metin sıkıştırma çalışmasında dillere ait tabloların sıkıştırma için kullanılacak olmasıdır.

2.2. Kullanılan Öznitelik Seçme Yöntemleri

2.2.1. TF-IDF

TF-IDF (Terim Frekansı – Ters Doküman Frekansı), bir terimin/kelimenin dokümanlar içerisindeki önemini gösteren istatistiksel yöntemler ile hesaplanan bir ağırlık faktörüdür. TF ile metin dokümanları içerisindeki bir kelimenin tüm dokümanlarda geçme sıklığını verirken, IDF ise bir kelimenin hangi dokümanlarda geçtiği bilgisini verir [19]. Bu yöntem ile terim ağırlıklandırma işlemleri yapılarak önemli öznitelikler belirlenir. TF-IDF' ye ait matematiksel formül aşağıdaki gibidir.

$$W_{TF-IDF} = TF(t_i, d_k) * \log\left(\frac{D}{d(t_i)}\right) \quad (1)$$

D: Toplam doküman sayısı
d(ti): ti terimin geçtiği toplam doküman sayısı

2.2.2. Ki-Kare

Bu yöntem ile Ki-Kare istatistiksel kriteri kullanılarak hedef değişkenle en belirgin ilişkisi olan özniteliklerin belirlenmesi sağlanır. Ki-Kare yöntemine ait formüller aşağıdaki gibidir [20].

$$Chi2(t,C) = \sum_{t\in\{0,1\}} \sum_{C\in\{0,1\}} \frac{\left(N_{t,C} - E_{t,C}\right)^2}{E_{t,C}} \quad (2)$$

$$Chi2(t) = \sum_{i=1}^{M} P(C_i) * Chi2(t,C) \quad (3)$$

P(Ci): Sınıf olasılığı.
N: t teriminin C sınıfı için gözlenen frekansı.
E: t teriminin C sınıfı için beklenen frekansı.

2.2.3. SelectFromModel-Lojistik Regresyon Yöntemi

Python paketi olan SelectFromModel [21], belirli bir eşik fonksiyonu (threshold) parametresi ile karşılaştırılan parametrelerin her biri için değerler sağlayan bir girdi modeli değerlendiricisidir. Eşik değerinin altında belirlenen öznitelikler kaldırılarak öznitelik seçimi gerçekleştirilir [22]. Bu model ile Lojistik Regresyon (LR) gibi modellerden üretilen tahminler parametre olarak kullanılarak önemli öznitelikler belirlenebilir. Özellikler arasındaki ilişki açıklamak için kullanılan LR, sınıf etiketleri 0 ve 1 şeklinde verilen ve ikili sınıflandırma problemlerinde kullanılan istatistiksel bir modeldir [23]. LR analizinde amaç, bağımlı ve bağımsız değişkenler arasındaki ilişki veya ilişkileri, en az değişken ile en iyi uyuma sahip olacak şekilde tanımlamaktır [24].

2.3. Yöntemlerin Uygulanması

Yapılan tüm sınıflandırma ve öznitelik seçme yöntemleri python programlama dili kullanılarak uygulanmıştır. Kelime Tabanlı İstatistiksel Yöntem (KTİY) Paulsen ve Martino' nun patent çalışmasından esinlenerek python programlama dili ile kodlanmıştır. langdetect için langdetect paketi ile "detect()" metotu ve fasttext için fasttext paketi ile "fasttext.load_model()" ve "predict()" metotları kullanılmıştır.
Öznitelik çıkarma, seçme ve makine öğrenmesi yöntemleri sklearn paketindeki aşağıdaki metotlar kullanılarak yapılmıştır.

Öznitelik çıkarma ve vektöre dönüştürme → CountVectorizer(), TfidfVectorizer()
Öznitelik seçme → TfidfVectorizer(max_features), SelectKBest(chi2, max_features), SelectFromModel(estimator=LogisticRegression, max_features)
Makine öğrenmesi ile sınıflandırma → GaussianNB(), BernoulliNB(), MultinomialNB(), DecisionTreeClassifier(), RandomForestClassifier(), SVC().

Makine öğrenmesi ile sınıflandırma yöntemlerinin tamamı varsayılan parametreleri ile kullanılmıştır.

2.4. Veri Setleri

Wikipedia makale özetlerinden oluşan 6 farklı dile ait 2 farklı boyuttaki kısa metinlerden oluşan veri setleri hem test verisi hem de öğrenme verisi olarak kullanılmıştır. A veri seti grubunda kullanılan kısa metinlerin (makale özeti) boyutu 0,2 KB'tan küçük değişik uzunluktaki karakter sayısından oluşmaktadır. A veri seti grubu içerdiği özetler bakımından tamamen farklı A1 ve A2 veri setinden oluşmaktadır. Öğrenme (model kurma) için kullanılan A1 veri seti 6.000 kısa metinden ve toplam 765 KB dosya boyutundan oluşmaktadır. Kısa metinlerin dillerinin belirlenmesi (test) için kullanılan A2 veri seti ise 24.000 kısa metinden ve toplam 3 MB dosya boyutundan oluşmaktadır.
B veri seti grubunda kullanılan özetlerin boyutu ise 0,2 KB ile 0,5 KB arasında değişik uzunluktaki karakter sayısından oluşmaktadır. B veri seti grubu içerdiği özetler bakımından tamamen farklı B1 ve B2 veri setinden oluşmaktadır. Öğrenme için kullanılan B1 veri seti 6.000 kısa metinden ve toplam 1,98 MB dosya boyutundan oluşmaktadır. Kısa metinlerin dillerinin belirlenmesi için kullanılan B2 veri seti ise 24.000 kısa metinden ve toplam 7,92 MB dosya boyutundan oluşmaktadır.

2.5. Veri Önişlem Aşamaları

Metinler üzerinde makine öğrenmesi yöntemlerinin uygulanabilmesi için farklı önişlem aşamalarında geçmesi gerekmektedir. Özellikle metinler üzerinde doğrudan makine öğrenmesi yöntemleri uygulanamayacağından verilerin sayısallaştırılması gerekmektedir. Başka bir deyişle metinlerin vektör uzayında ifade edilmelidir.

Makine öğrenmesi için önişlem aşamaları genel hatları ile aşağıdaki adımlardan oluşmaktadır. Ancak bu yöntemlerin yapılacak çalışmaya bağlı olarak bazıları kullanılmayabilir.

- küçük/büyük harf dönüşümü,
- noktalama işaretlerinin kaldırılması,
- etkisiz kelimelerin silinmesi (remove stopwords),
- kelime köklerinin çıkarılması (stemming),
- öznitelik çıkarımı,
- öznitelik seçimi.

İstatistiksel yöntemlerde yapılacak önişlem aşamaları makine öğrenmesi yöntemleri gibi karmaşık olamamakla beraber, etkisiz kelimelerin kullanılması belirleyici öznitelikler olmaktadır. Bu anlamda Kelime Tabanlı İstatistiksel Yöntem (KTİY), langdetect ve fasttext kullanılarak yapılan sınıflandırma küçük/büyük harf dönüşümü ve noktalama işaretlerinin kaldırılması dışında herhangi bir önişlem adımı uygulanmamıştır.

2.6. Testlerin Değerlendirilmesi

Yöntemlerin sınıflandırma başarısını ölçmek için doğruluk, duyarlılık, kesinlik F1 skoru ve jaccard skoru değerleri, sklearn.metrics paketindeki ilgili fonksiyonlar kullanılarak elde edilmiştir. Yöntemlerin işlem süreleri "Intel Core i7 9750H 2.60 GHz" işlemci, "16 GB" hafıza ve "Windows 10 Pro 64-bit" işletim sistemi kullanan bir bilgisayar ile ölçülmüştür.

2.6.1. Doğruluk, Kesinlik, Duyarlılık, F1 Skoru ve Jaccard Skoru

Sınıflandırma performansını ölçmek için kullanılan Doğruluk, Kesinlik, Duyarlılık ve F1 Skoru dört temel değer üzerinden hesaplanırlar: Doğru Pozitif (DP), Doğru Negatif (DY), Yanlış Pozitif (YP) ve Yanlış Negatif (YN).

- **Doğruluk:** Doğru olarak tahmin edilen örneklerin sayısının, veri kümesinde bulunan tüm örneklerin sayısına bölünerek hesaplanır.

$$doğruluk = \frac{DP + DN}{DP + DN + YP + YN} \tag{4}$$

- **Kesinlik:** Doğru Pozitif olarak tahmin edilen örneklerin sayısının, tüm pozitif örneklerin sayısına bölünerek hesaplanır.

$$kesinlik = \frac{DP}{DP + YP} \tag{5}$$

- **Duyarlılık:** Doğru Pozitif olarak tahmin edilen örneklerin sayısının hem Doğru Pozitif hem de Yanlış Negatif örneklerin toplam sayısına bölünmesi ile hesaplanır:

$$duyarlılık = \frac{DP}{DP + YN} \tag{6}$$

- **F1 Skoru:** Duyarlılık ve Kesinlik değerlerinin harmonik ortalaması ile hesaplanır.

$$F1\ Skoru = 2 * \frac{Duyarlılık * Kesinlik}{Duyarlılık + Kesinlik} \tag{7}$$

- **Jaccard Skoru:** Jaccard indeksi veya Jaccard benzerlik katsayısı olarak da bilinen Jaccard skoru, kesişme boyutunun iki etiket kümesinin birleşiminin boyutuna bölünmesi olarak tanımlanır ve aşağıdaki gibi hesaplanabilir [25]. Başka bir ifade ile jaccard skoru Doğru Pozitif örneklerin sayısının, Doğru Negatif dışındaki (DP, YP, YN) tüm örneklerin sayısının toplamına bölünmesi ile hesaplanabilir.

$$Jaccard\ Skoru(d1, d2) = \frac{d1\ \cap \mathrm{d2}}{d1\ \cup\ d2} \tag{8}$$

$$Jaccard\ Skoru = \frac{DP}{DP + YP + YN} \tag{9}$$

3. Test Sonuçları

3.1. A Veri Seti Grubu Testleri

Tablo 1'de 6.000 özet içeren A1 eğitim veri setinin farklı öznitelik seçme teknikleri ve 10-Kat çapraz doğrulama kullanılarak makine öğrenmesi yöntemleri ile sınıflandırılması sonucu elde edilen performans değerleri verilmiştir. Yapılan 3 değerlendirmede de kullanılan 4.153 öznitelik sayısı SelectFromModel-LR tarafından belirlenen en önemli özniteliklerdir. Elde edilen performans değerleri incelendiğinde veri setinin uygunluğu yüksek doğruluk değerleri ile doğrulanmıştır. En yüksek doğruluk %99,92 değeri öznitelik seçimi olarak SelectFromModel-LR, sınıflandırma yöntemi olarak ise M-NB kullanıldığında elde edilmiştir. Ayrıca her üç öznitelik seçim tekniğinde M-NB en iyi sınıflandırmayı yapmıştır. Makine öğrenmesi yöntemleri için en uygun öznitelik seçim tekniği değerlendirildiğinde en uyumlu ikililerin M-NB – SelectFromModel, B-NB – SelectFromModel, G-NB – Ki-Kare, KA – Ki-Kare, RO – Ki-Kare, KNN – Ki-Kare, DVM – TF-IDF şeklinde olduğu anlaşılmaktadır. KNN sınıflandırma yöntemi ile beraber TF-IDF öznitelik seçim tekniği uygulandığında en düşük doğruluk değerine ulaşılmıştır.

Tablo 1. A1 veri setinin sınıflandırılmasında SelectFromModel-LR, Ki-Kare, TF-IDF ile öznitelik seçimi ve 10-Kat çapraz doğrulama tekniği uygulanarak elde edilen performans değerleri

SelectFromModel-LR							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,92%	99,67%	99,75%	98,45%	99,73%	94,57%	99,57%
F1 Skoru	99,91%	99,66%	99,75%	98,44%	99,74%	94,73%	99,57%
Duyarlılık	99,91%	99,65%	99,75%	98,42%	99,73%	94,55%	99,56%
Kesinlik	99,92%	99,67%	99,75%	98,48%	99,74%	95,65%	99,58%
Jaccard Skoru	99,83%	99,32%	99,50%	96,94%	99,48%	90,26%	99,14%
Ki-Kare							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,87%	99,60%	99,80%	98,63%	99,77%	95,35%	99,70%
F1 Skoru	99,87%	99,59%	99,80%	98,63%	99,77%	95,49%	99,70%
Duyarlılık	99,87%	99,58%	99,80%	98,61%	99,76%	95,35%	99,69%
Kesinlik	99,87%	99,61%	99,79%	98,67%	99,78%	96,17%	99,71%
Jaccard Skoru	99,73%	99,19%	99,60%	97,31%	99,54%	91,59%	99,40%
TF-IDF							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,87%	99,62%	98,45%	98,28%	99,72%	72,40%	99,85%
F1 Skoru	99,87%	99,61%	98,43%	98,27%	99,72%	74,44%	99,85%
Duyarlılık	99,87%	99,60%	98,43%	98,26%	99,72%	72,34%	99,85%
Kesinlik	99,86%	99,62%	98,45%	98,30%	99,72%	90,23%	99,85%
Jaccard Skoru	99,73%	99,23%	96,92%	96,62%	99,44%	62,56%	99,70%

Makine öğrenmesi yöntemlerinde kullanılan öznitelik sayısı hem sınıflandırma performansını hem de sınıflandırma süresini doğrudan etkilemektedir. Bu çalışmada yapılan sınıflandırma dil tanımaya yönelik olduğundan veri setine ait özniteliklerin tamamı kullanılmadan bile yüksek doğruluk değeri ile sınıflandırma yapılabileceği öngörülmektedir.

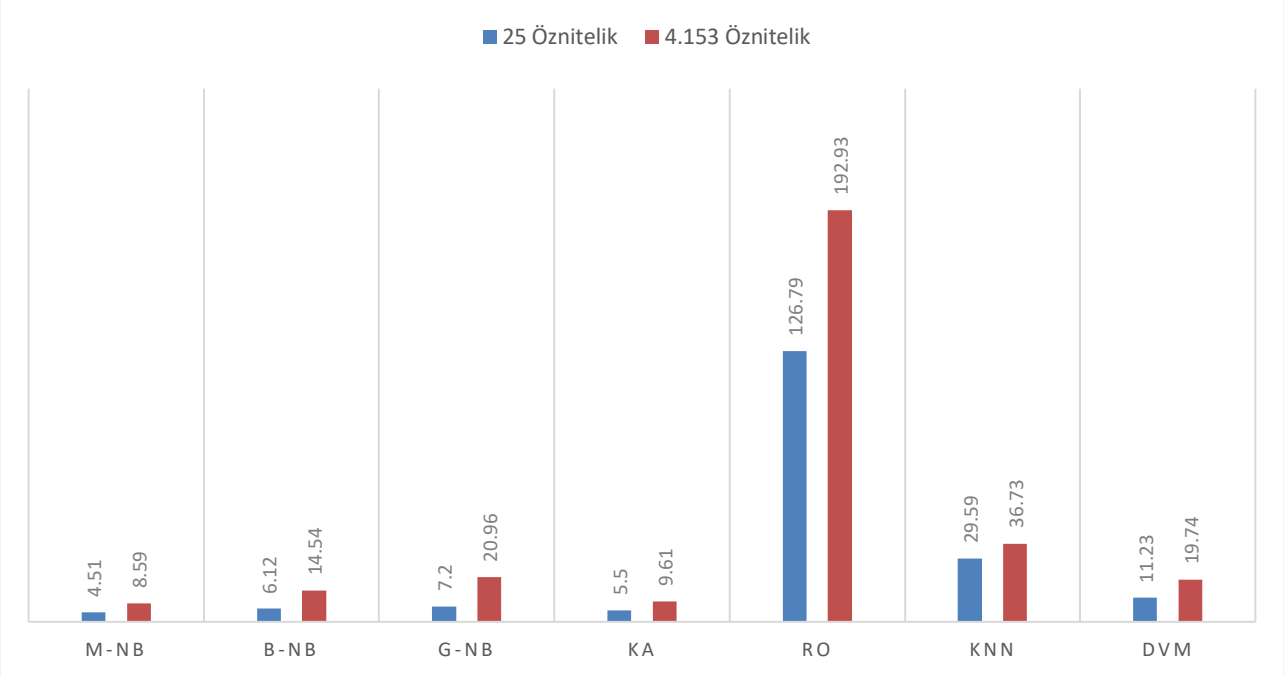
Tablo 2'de 24.000 özet içeren A2 test veri setinde bulunan özetlerin, A1 veri seti ile kurulan modele göre yazıldıkları dillerin tahmin edilmesine yönelik makine öğrenmesi ile farklı öznitelik sayısı kullanılarak edilen performans değerleri verilmiştir. Öznitelik seçim tekniği olarak SelectFromModel-LR ile oluşan hem 4.153 belirleyici öznitelik hem de bunlar arasındaki en belirleyici 25 öznitelik kullanılarak elde edilen performans değerlerine göre en yüksek doğruluk değeri 4.153 öznitelik sayısı ile M-NB kullanıldığında elde edilmiştir. En

düşük doğruluk değeri ise 4.153 öznitelik sayısı ile KNN olduğu görülmektedir. B-NB, KA, KNN ve DVM sınıflandırma yöntemlerinde 25 öznitelik kullanıldığında daha yüksek doğruluk değerleri elde edilmiştir. Metin sınıflandırma problemlerinin en önemli ve en zor aşaması olan en uygun özniteliklerin belirlenmesidir. Bu anlamda dil tanıma problemlerinde özniteliklerin sadece bir kısmı kullanılarak bile yüksek doğruluk değeri ile sınıflandırma yapılabileceği anlaşılmıştır.

Tablo 2. Öznitelik seçimi olarak SelectFromModel-LR, model kurmak için (öğrenme) A1 kullanılarak A2 veri setindeki özetlerin dillerinin belirlenmesinde edilen performans değerleri

SelectFromModel-LR- 4.153 Öznitelik							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,71%	91,45%	99,13%	96,67%	98,09%	86,80%	97,34%
F1 Skoru	99,71%	90,64%	99,12%	96,65%	98,07%	86,91%	97,30%
Duyarlılık	99,71%	93,63%	99,13%	96,79%	98,18%	92,22%	97,55%
Kesinlik	99,71%	91,45%	99,13%	96,67%	98,09%	86,80%	97,34%
Jaccard Skoru	99,42%	85,08%	98,26%	93,57%	96,27%	79,08%	94,89%
SelectFromModel-LR- 25 Öznitelik							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	97,90%	95,30%	95,67%	97,23%	97,98%	95,60%	97,35%
F1 Skoru	97,90%	95,21%	95,67%	97,24%	97,99%	95,63%	97,37%
Duyarlılık	97,99%	95,73%	95,90%	97,33%	98,05%	95,95%	97,54%
Kesinlik	97,90%	95,30%	95,67%	97,23%	97,98%	95,60%	97,35%
Jaccard Skoru	95,91%	91,09%	91,70%	94,66%	96,07%	91,66%	94,93%

Şekil 1'de makine öğrenmesi yöntemlerinin A1 veri seti ile oluşturulan modele göre A2 veri setine ait özetleri dillerine göre sınıflandırma süreleri saniye türünden verilmiştir. Sınıflandırma işlemi SelectFromModel-LR öznitelik seçimi ile elde edilen hem 4.153 hem de 25 öznitelik kullanılarak iki farklı şekilde yapılmıştır. Buna göre en hızlı sınıflandırma yönteminin M-NB, en yavaş sınıflandırma yönteminin ise RO olduğu görülmektedir. Ayrıca daha az öznitelik kullanılarak daha hızlı sınıflandırma yapılabileceği Şekil 1'den açıkça anlaşılmaktadır.



Şekil 1. A2 veri setinin makine öğrenmesi yöntemleri ile farklı öznitelik sayısındaki sınıflandırma süreleri (sn)

Tablo 3'te A2 veri setinin fasttext, KTİY ve langdetect ile sınıflandırılması sonucu elde edilen performans değerleri ve sınıflandırma süreleri verilmiştir. Buna göre %99,79 doğruluk değeri ile en iyi sınıflandırma fasttext yöntemi ile olurken, %96,79 doğruluk değeri ile en düşük sınıflandırmayı yapan yöntem KTİY olmuştur. Ancak KTİY ile yapılan sınıflandırmanın fasttext' e göre daha hızlı oluğu anlaşılmaktadır. fasttext yöntemi ile yapılan

sınıflandırmanın, Tablo 2' de verilen makine öğrenmesi yöntemlerinden hem süre hem de doğruluk oranı bakımından en iyi sınıflandırmayı yapan M-NB' den daha iyi olduğu görülmektedir.

Tablo 3. A2 veri setinin fasttext, KTİY ve langdetect ile sınıflandırma performans değerleri ve süreleri

	fasttext	**KTİY**	**langdetect**
Doğruluk	99,79%	96,99%	99,66%
F1 Skoru	99,79%	97,00%	99,66%
Duyarlılık	99,79%	97,13%	99,66%
Kesinlik	99,79%	96,99%	99,66%
Jaccard Skoru	99,58%	94,21%	99,33%
Süre (sn)	1,58	0,97	151,49

3.2. B Veri Seti Grubu Testleri

Tablo 4'te 6.000 özet içeren B1 eğitim veri setinin farklı öznitelik seçme teknikleri ve 10-Kat çapraz doğrulama kullanılarak makine öğrenmesi yöntemleri ile sınıflandırılması sonucu elde edilen performans değerleri verilmiştir. SelectFromModel-LR kullanılarak elde edilen öznitelik sayısı 10.100' dür. Ancak A veri seti grubu ile yapılacak karşılaştırmanın daha adil olması için en değerli 4.153 öznitelik kullanılmıştır. Elde edilen performans değerleri incelendiğinde veri setinin uygunluğu yüksek doğruluk değerleri ile doğrulanmıştır. En yüksek doğruluk değeri olan %99,99, öznitelik seçimi olarak SelectFromModel-LR, sınıflandırma yöntemi olarak ise B-NB kullanıldığında elde edilmiştir. Diğer öznitelik seçim teknikleri ile yapılan sınıflandırmalarda M-NB ile B-NB arasında anlamlı bir fark oluşmadığı görülmektedir. Kullanılan veri seti için makine öğrenmesi yöntemleri ile uyumlu öznitelik seçim teknikleri değerlendirildiğinde en uyumlu ikililerin B-NB – SelectFromModel-LR, G-NB – Ki-Kare, KA – SelectFromModel-LR, RO – Ki-Kare, KNN – Ki-Kare, DVM – (Ki-Kare ve TF-IDF) şeklinde olduğu anlaşılmaktadır. Sınıflandırma yöntemi olarak KNN ve öznitelik seçim tekniği olarak Ki-Kare kullanıldığında en düşük performans değeri elde edildiği ayrıca anlaşılmaktadır.

Tablo 4. B1 veri seti LR, Ki-Kare, TF-IDF ile öznitelik seçimi ve 10-Kat çapraz doğrulama tekniği uygulanarak elde edilen performans değerleri

SelectFromModel-LR							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,98%	99,99%	99,72%	98,67%	99,97%	99,52%	99,90%
F1 Skoru	99,98%	99,99%	99,72%	98,67%	99,97%	99,52%	99,90%
Duyarlılık	99,98%	99,99%	99,71%	98,66%	99,96%	99,51%	99,89%
Kesinlik	99,98%	99,99%	99,72%	98,69%	99,97%	99,53%	99,91%
Jaccard Skoru	99,97%	99,98%	99,44%	97,38%	99,93%	99,04%	99,80%
Ki-Kare							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,98%	99,98%	99,93%	98,62%	99,93%	95,70%	99,93%
F1 Skoru	99,98%	99,98%	99,93%	98,62%	99,93%	95,80%	99,93%
Duyarlılık	99,98%	99,98%	99,93%	98,61%	99,93%	95,70%	99,93%
Kesinlik	99,98%	99,98%	99,94%	98,65%	99,94%	96,37%	99,94%
Jaccard Skoru	99,97%	99,97%	99,87%	97,29%	99,87%	92,11%	99,87%
TF-IDF							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,98%	99,98%	98,80%	98,57%	99,95%	98,37%	99,93%
F1 Skoru	99,98%	99,98%	98,80%	98,56%	99,95%	98,36%	99,93%
Duyarlılık	99,98%	99,98%	98,80%	98,54%	99,95%	98,35%	99,93%
Kesinlik	99,98%	99,98%	98,81%	98,60%	99,95%	98,41%	99,94%
Jaccard Skoru	99,97%	99,97%	97,65%	97,18%	99,90%	96,79%	99,87%

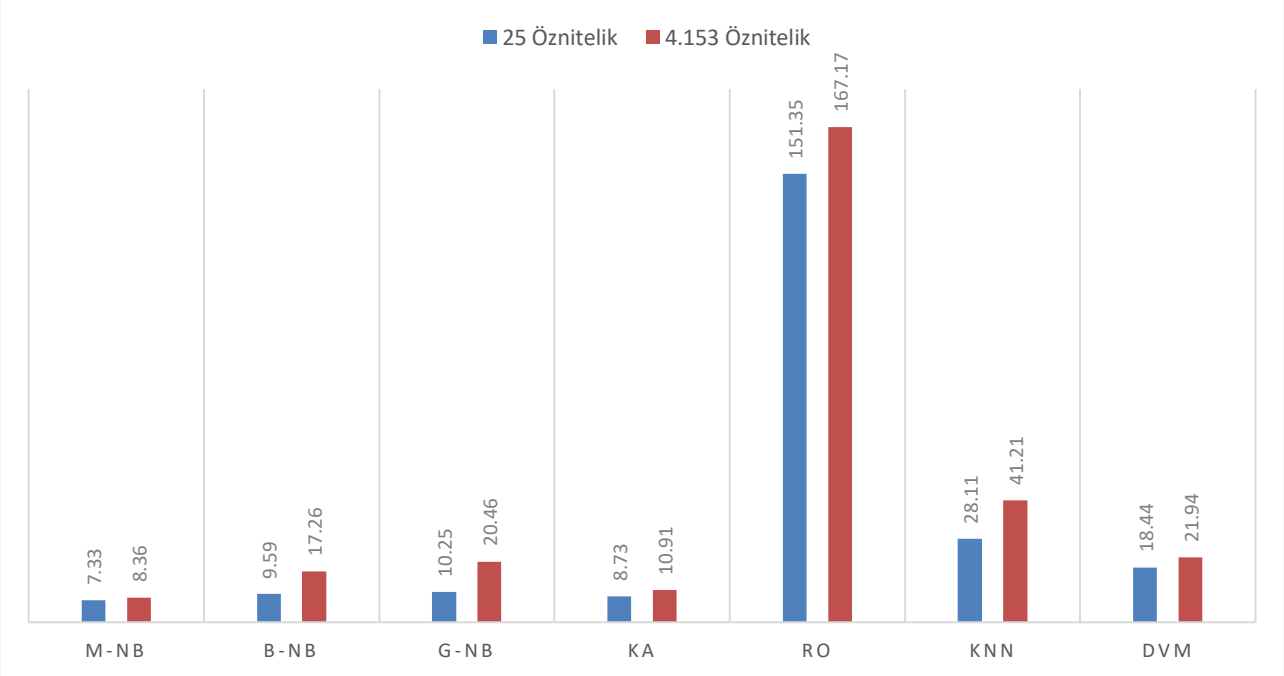
Tablo 5'te 24.000 özet içeren B2 test veri setinde bulunan özetlerin, B1 veri seti ile kurulan modele göre yazıldıkları dillerin tahmin edilmesine yönelik makine öğrenmesi ile farklı öznitelik sayısı kullanılarak edilen performans değerleri verilmiştir. Öznitelik seçim tekniği olarak SelectFromModel-LR ile oluşan hem 4.153 en

önemli öznitelik hem de bunlar arasındaki en önemli 25 öznitelik kullanılarak elde edilen performans değerlerine göre en yüksek doğruluk değeri 4.153 öznitelik sayısı ile B-NB kullanıldığında elde edilmiştir. Ayrıca en düşük doğruluk değeri 25 öznitelik sayısı ile KNN olduğu görülmektedir. KA en yüksek doğruluk değerini 25 öznitelik sayısında verirken diğer yöntemler 4.153 öznitelik sayısı ile daha yüksek doğruluk değerleri vermiştir. Bu veri setinde de sadece 25 öznitelik kullanılarak metinlerin dillerinin yüksek doğruluk değeri ile belirlenebileceği anlaşılmıştır.

Tablo 5. B1 veri seti ile kurulan modele göre B2 veri setinin sınıflandırılması

SelectFromModel-LR- 4.153 Öznitelik							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,97%	99,98%	99,77%	98,71%	99,98%	99,44%	99,93%
F1 Skoru	99,97%	99,98%	99,77%	98,71%	99,97%	99,44%	99,93%
Duyarlılık	99,97%	99,98%	99,77%	98,71%	99,98%	99,44%	99,93%
Kesinlik	99,97%	99,98%	99,77%	98,71%	99,98%	99,44%	99,93%
Jaccard Skoru	99,95%	99,96%	99,54%	97,46%	99,95%	98,88%	99,86%
SelectFromModel-LR- 25 Öznitelik							
	M-NB	**B-NB**	**G-NB**	**KA**	**RO**	**KNN**	**DVM**
Doğruluk	99,60%	99,64%	98,90%	98,85%	99,63%	98,10%	99,54%
F1 Skoru	99,60%	99,64%	98,89%	98,85%	99,63%	98,10%	99,54%
Duyarlılık	99,61%	99,64%	98,90%	98,85%	99,63%	98,18%	99,54%
Kesinlik	99,60%	99,64%	98,90%	98,85%	99,63%	98,10%	99,54%
Jaccard Skoru	99,21%	99,28%	97,82%	97,72%	99,27%	96,30%	99,08%

Şekil 2' de makine öğrenmesi yöntemlerinin B1 veri seti ile oluşturulan modele göre B2 veri setine ait özetleri dillerine göre sınıflandırma süreleri saniye türünden verilmiştir. Sınıflandırma işlemi SelectFromModel-LR öznitelik seçimi ile elde edilen hem 4.153 hem de 25 öznitelik kullanılarak iki farklı şekilde yapılmıştır. Bu veri setinde kullanılan özetlerin boyutu A veri seti grubundakilerden daha büyük olmasına rağmen Şekil1'deki gibi benzer sınıflandırma süreleri oluşması kullanılan öznitelik sayısı ile açıklanabilir.



Şekil 2. B2 veri setinin makine öğrenmesi yöntemleri ile farklı öznitelik sayısındaki sınıflandırma süreleri (sn)

Tablo 6'da B2 veri setinin fasttext, KTİY ve langdetect ile sınıflandırılması sonucu elde edilen performans değerleri ve sınıflandırma süreleri verilmiştir. Buna göre %99,99 doğruluk değeri ile en iyi sınıflandırma fasttext yöntemi ile olurken, %99,93 doğruluk değeri ile en düşük sınıflandırmayı yapan yöntem KTİY olmuştur. Bu veri setinde de KTİY ile yapılan sınıflandırmanın fasttext' e göre daha hızlı oluğu anlaşılmaktadır. fasttext yöntemi ile yapılan

sınıflandırmanın, Tablo 4'te verilen makine öğrenmesi yöntemlerinden en hızlı olan M-NB' den, doğruluk bakımından en iyi sınıflandırmayı yapan B-NB' den daha iyi olduğu görülmektedir.

Tablo 6. A2 veri setinin fasttext, KTİY ve langdetect ile sınıflandırma performans değerleri ve süreleri

	fasttext	**KTİY**	**langdetect**
Doğruluk	99,99%	99,93%	99,97%
F1 Skoru	99,99%	99,93%	99,97%
Duyarlılık	99,99%	99,93%	99,97%
Kesinlik	99,99%	99,93%	99,97%
Jaccard Skoru	99,98%	99,86%	99,94%
Süre (sn)	2,95	1,68	179,16

4. Sonuçlar

Bu çalışmada A1, A2, B1 ve B2 veri setleri kullanılarak dil tanımaya yönelik farklı sınıflandırma yöntemleri kullanılarak sınıflandırma çalışması yapılmıştır. Makine öğrenmesi ile yapılan sınıflandırmalarda A1 ve B1 veri setleri kullanılarak bir model oluşturulmuş ve kurulan modellerin uygunluğu test edilmiştir. Yapılan testler sonucunda 0,2 KB'tan küçük ve 0,2 KB – 0,5 KB arası kısa metinleri içeren sırası ile A1 ve B1 veri seti için farklı öznitelik seçim teknikleri uygulanarak ve 10-kat çapraz doğrulama yapılarak model geçerliliği doğrulanmıştır. Makine öğrenmesi yöntemlerinin sınıflandırma başarıları kullanılan kısa metinlerin büyüklüğüne, öznitelik seçim tekniği ve sayısına göre değiştiği görülmüştür. 0,2 KB ve daha küçük kısa metinler için makine öğrenmesi ile yapılacak sınıflandırmalarda M-NB ve B-NB ile SelectFromModel-LR, G-NB, KA, RO ve KNN ile Ki-Kare, DVM ile TF-IDF öznitelik seçim tekniğinin kullanılması ile daha yüksek doğruluk değerleri elde edilmiştir. Boyutu 0,2 KB ve 0,5 KB arasında değişen kısa metinlerde ise tüm yöntemler ile SelectFromModel-LR öznitelik seçim tekniği ile en yüksek doğruluk değerleri oluştuğundan bu boyuttaki kısa metinlerin sınıflandırılmasında SelectFromModel-LR seçim tekniğinin kullanılmasının daha uygun olduğu anlaşılmıştır. Yapılan testlere göre öznitelik sayısının fazla olması daha yüksek doğruluk ile sınıflandırma yapılacağı garantisini vermediğini göstermektedir. 0,2 KB'den küçük metinlerde KNN, B-NB, KA ve DVM, 0,2-0,5 KB arasındaki kısa metinlerde ise KA 25 öznitelik kullanıldığında 4.153 öznitelik kullanıldığı duruma göre daha yüksek doğruluk değeri ile sınıflandırma yapılabilmiştir. Makine öğrenmesi yöntemlerinde hem 4.153 hem de 25 öznitelik sayısı ile yapılan sınıflandırmalara göre 0,2 KB'den küçük metinlerde M-NB, 0,2-0,5 KB arası metinlerde ise B-NB kullanılmasının daha uygun olduğu anlaşılmıştır. Eşit sayıda özet içeren hem 0,2 KB hem de 0,2-0,5 KB arası veri seti gruplarında makine öğrenmesi yöntemlerinde eşit sayıda öznitelik kullanılarak yapılan test sonuçlarında göre benzer sınıflandırma süreleri oluşmuştur. Bu durum kullanılan öznitelik sayısının sınıflandırma süresinde belirleyici bir etken olduğunu açıklamaktadır. Makine öğrenmesi yöntemleri ile sadece 25 öznitelik kullanıldığında bile KTİY' nin hem A2 veri setini hem de B2 veri setini dillerine göre çok daha hızlı bir şekilde ayırdığı yapılan testler ile doğrulanmıştır. KTİY tüm yöntemler arasında en hızlı yöntem olurken 0,2 KB ve daha küçük boyuttaki kısa metinleri sınıflandırma başarısı, 0,2-0,5 KB arasındaki boyutlardan oluşan kısa metinleri sınıflandırma başarısından daha düşük olduğu görülmüştür. Diğer tüm sınıflandırma yöntemlerinin 02-05 KB boyuttaki kısa metinleri sınıflandırma başarıları daha yüksek olmuştur. Tüm kullanılan sınıflandırma yöntemleri arasında en yüksek doğruluk değeri ile sınıflandırma yapan yönteminin fasttext olduğu yapılan testler sonucunda anlaşılmıştır.

Kaynakça

[1] Gülşen, E., Gündüz, H., Cataltepe, Z., & Serinol, L. (2015, May). Big data feature selection and projection for gender prediction based on user web behaviour. *In 2015 23nd Signal Processing and Communications Applications Conference (SIU) (pp. 1545-1548). IEEE.*

[2] Yengi, Y., & Omurca, S. İ. (2015). Lojistik Regresyonun Özellik Azaltma Teknikleri ile Gen Dizilimlerinin Sınıflandırılmasındaki Başarısı. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, 8(1),* 1-12.

[3] Parlar, T., Saraç, E., & Özel, S. A. (2017, May). Comparison of feature selection methods for sentiment analysis on Turkish Twitter data. *In 2017 25th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.*

[4] Sel, İ., Karci, A., & Hanbay, D. (2019, September). Feature Selection for Text Classification Using Mutual Information. *In 2019 International Artificial Intelligence and Data Processing Symposium (IDAP) (pp. 1-4). IEEE.*

[5] Erdem, H., & Özgur, A. (2018). Feature selection and multiple classifier fusion using genetic algorithms in intrusion detection systems. *Journal of the Faculty of Engineering and Architecture of Gazi University 33:1*, 75-87.

[6] Akyol, K. (2018). Meme Kanseri Tanısı İçin Özniteliklerin Öneminin Değerlendirilmesi Üzerine Bir Çalışma. *Akademik Platform Mühendislik ve Fen Bilimleri Dergisi, 6(2),* 109-115.

[7] Ataş, P. K., Tufan, K., & Şevkli, A. Z. (2016, April). A variable neighborhood search based feature selection model for early prediction of the Alzhemier's disease. *In 2016 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT) (pp. 1-4). IEEE.*

[8] Kaya, Y., Ertuğrul, Ö. F., & Tekin, R. (2015). Doküman dili tanıma için ikili örüntüler tabanlı yeni bir yaklaşım. *Akademik Bilişim, Eskişehir.*

[9] Rish, I. (2001, August). An empirical study of the naive Bayes classifier. *In IJCAI 2001 workshop on empirical methods in artificial intelligence Vol. 3, No. 22,* pp. 41-46.

[10] Agrawal, R., Imielinski, T., & Swami, A. (1993). Database mining: A performance perspective. *IEEE transactions on knowledge and data engineering, 5(6),* 914-925*.*

[11] Fix, E. (1951). Discriminatory analysis: nonparametric discrimination, consistency properties. *USAF School of Aviation Medicine.*

[12] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification*. IEEE transactions on information theory, 13(1),* 21-27.

[13] Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician, 46(3),* 175-185.

[14] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning, 20(3),* 273-297.

[15] Tipping, M. E. (2000). The relevance vector machine. *In Advances in neural information processing systems (pp. 652-658).*

[16] Danilk M.M., (2013) https://pypi.org/project/langdetect/, Erişim (Tarihi: 11.06.2021).

[17] Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., & Mikolov, T. (2016). Fasttext. zip: Compressing text classification models*. arXiv preprint arXiv:1612.03651.*

[18] Paulsen Jr, R. C., & Martino, M. J. (2004). U.S. Patent No. 6,704,698. *Washington, DC: U.S. Patent and Trademark Office.*

[19] Kınık, D. (2020). *TF-IDF ve Doc2vec Tabanlı Metin Sınıflandırma Sisteminin Başarım Değerinin Ardışık Kelime Gurubu Tespiti İle Arttırılması.* (Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, İstanbul)

[20] Çekik R. (2020). *Kısa Metin Sınıflandırma İçin Öznitelik Seçimi.* (Fen Bilimleri Enstitüsü Doktora Tezi, Eskişehir).

[21] *https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectFromModel.html.* (Erişim Tarihi: 11.06.2021).

[22] Popov, N. V., Razmochaeva, N. V., & Klionskiy, D. M. (2020, June). Investigation of Algorithms for Converting Dimension of Feature Space in Retail Data Analysis Problems. *In 2020 9th Mediterranean Conference on Embedded Computing (MECO) (pp. 1-4). IEEE.*

[23] Önder, H., & Cebeci, Z. (2002). Lojistik regresyonlarda değişken seçimi. *Çukurova Üniv. Ziraat Fakültesi Dergisi, 17(2),* 105-114*.*

[24] Çokluk, Ö. (2010). Lojistik Regresyon Analizi: Kavram ve Uygulama. Educational Sciences: Theory & Practice, 10(3).

[25] Jaccard, P., (1901). Étude comparative de la distribuition florale dans une portion des Alpes et des Jura. *Bull Soc Vandoise Sci Nat* 37, 547-579.