Konya Mühendislik Bilimleri Dergisi., c. 8, s. 4, 839-852, 2020
Konya Journal of Engineering Sciences v. 8, n. 4, 839-852, 2020
ISSN: 2667-8055 (Elektronik)
DOI: 10.36306/konjes.650150

TWİTTER VERİLERİNDEN DOĞAL DİL İŞLEME VE MAKİNE ÖĞRENMESİ İLE HASTALIK TESPİTİ

[1,2]Ali ÖZTÜRK, [3]Üsame DURAK, [4]Fatma BADILLI

[1,3,4]KTO Karatay Üniversitesi, Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü, 42020, Konya/TÜRKİYE
[2]Havelsan A.Ş., 06510, Ankara/TÜRKİYE
[1,2] ali.ozturk@karatay.edu.tr, [3] usame.durak@karatay.edu.tr, [4] fatma.badilli@ogrenci.karatay.edu.tr

(Geliş/Received: 23.11.2019; Kabul/Accepted in Revised Form: 22.07.2020)

ÖZ: Bu çalışmada twitterdaki kullanıcıların yazmış oldukları mesajların hastalık konulu olup olmadığı ve hastalık türleri tespit edilmiştir. Bu amaçla gözetimli ve gözetimsiz makine öğrenmesi algoritmaları, TF-IDF ve BOW yöntemleri ile çıkarılan özellikler ile denenmiş ve karşılaştırmalar yapılmıştır. Veriler Python betikleri ile twitter üzerinden toplanmıştır. Algoritmaları uygulamak için Python için geliştirilmiş Scikit-Learn kütüphanesi kullanılmıştır. Gözetimsiz olarak verilerin kümelenmesinde %68.60'lık bir başarı elde edilirken, gözetimli algoritmalar ile yapılan sınıflandırmalarda %97.48'lik başarı oranına ulaşılmıştır.

Anahtar Kelimeler: *Twitter, Hastalık Tanıma, Doğal Dil İşleme, Makine Öğrenmesi*

Disease Detection From Twitter Data Using Natural Language Processing and Machine Learning

ABSTRACT: In this study, we determined whether the subject of the messages of the twitter users were about a disease and what kind of diseases they were. For this purpose, supervised and unsupervised machine learning algorithms were tested and compared using the features extracted via TF-IDF and BOW methods. Data were collected with Python scripts from Twitter. The Scikit-Learn library which was developed for Python was used to implement the algorithms. The clustering algorithms which are unsupervised methods achieved an accuracy level of %68.60, while the performance of the supervised classification algorithms reached to the accuracy level of %97.48.

Key Words: *Twitter, Disease Recognition, Natural Language Processing, Machine Learning*

GİRİŞ (INTRODUCTION)

Twitter gibi mikro blog platformları yeni nesil bilgi kaynakları oluşturmakta ve bu bilgi kaynakları anlık olarak işlenerek dünya üzerinde herhangi bir anda ve yerde neler olduğuna dair çeşitli fikirler verebilmektedir. Bunun için de mesajların sınıflandırılması gerekmektedir. Doğal dil işleme, yapay zekanın alt alanlarından birisidir. Doğal dil ile yazılmış bir metni anlamak, bir metne cevap olarak veya farklı amaçlar için bir metin üretmek temel hedefleridir. Doğal dil işleme yöntemleri, insanların el ile yapmak zorunda olduğu bazı metin sınıflandırmalarının otomatik olarak yapılabilmesine olanak vermekte ve örneğin anket değerlendirme gibi spesifik uygulamalarda çok kullanışlı olabilmektedir.

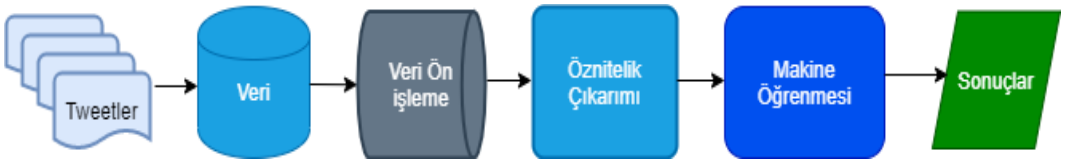
Literatürde, metinler üzerinde doğal dil işleme ile çıkarılan özelliklerin otomatik sınıflandırılmasının yapıldığı farklı çalışmalar bulunmaktadır. Ambert ve Cohen (2009), hastanelerde kullanılan serbest yazı klinik raporları üzerine doğal dil işleme yöntemleri uygulamışlardır. Bu raporlardan, hastaya ait hastalık ve tedavi durumu hakkında bir çıkarım yapabilen otomatik bir sistem

önermişlerdir. Önişleme kısmında Auto HP+ NegEx kullanarak vektörler oluşturmuşlar ve burada oluşan sıfır vektörleri için Zero-Vector Filtering (ZeroVF) kullanmışlar, ardından sınıflandırma işlemi için de Linear SVM (Support Vector Machine) algoritmasını uygulamışlardır. Sonuç olarak önerilen sistem, uzman kural tabanlı sistemlere karşı iyi bir performans göstermiştir. Achrekar ve diğ. (2011), twitter verileri üzerinde ARX (auto-regression with exogenous inputs) modelini k-katmanlı çapraz doğrulama yöntemi ile uygulayarak nezle vakalarındaki artışı tespit etmişlerdir. Çalışma sonucunda elde ettikleri sonuçlarla, hastalık kontrol ve önleme merkezlerinden topladıkları veriler arasındaki korelasyon katsayısını 0.9846 olarak bulmuşlardır. Morita ve diğ. (2013), yapmış olduğu araştırmada grip kelimesini içeren twitter mesajlarını veri seti olarak toplamışlar, ardından bu veriler arasında gerçekten grip ile ilgili olan verileri gözetimli makine öğrenmesi algoritmaları ile test etmişler ve sonuç olarak SVM algoritmasının en iyi performansa sahip olduğunu bulmuşlardır. Özellik çıkarım adımında ise BOW algoritmasını kullanmışlar ve tweetleri belli bir doğruluk oranıyla negatif ya da pozitif olarak ayırt etmişlerdir.

Dai ve Mikdash (2015), twitter verilerinden grip hastalığı ile ilgili olanları tespit etmek için yaptıkları çalışmada öncelikle doğal dil işleme yöntemleri ile gürültülü verileri eleyerek özellik vektörü çıkarımı yapmışlardır. Daha sonra kendilerinin tanımladığı özellikler ve makine öğrenmesi algoritmalarının otomatik ürettiği özelliklere Naive Bayes yöntemini uygulayarak sınıflandırma yapmışlardır. Yine aynı araştırmacılar bir başka çalışmada (Dai ve Mikdash 2016), gürültülü twitter verilerini eleyip bunları bölgelere ayırmak için uzaklık-tabanlı aykırılık tespit yöntemini uyguladıktan sonra, grip salgınını tespit etmek için bölge-tabanlı hipotez testini kullanmışlardır. Rudra ve diğ. (2017), iki farklı salgınla (Ebola ve MERS) ilgili twitter verilerini alt düzey sözlük özelliklerine ayırarak otomatik olarak sınıflandırmışlardır. Diğer bir çalışmalarında (Rudra ve diğ., 2018) ise sınıflandırılan veriler özetlenerek, salgından etkilenen ya da etkilenmesi muhtemel topluluklarla ve sağlık kuruluşlarıyla paylaşılabilecek hale getirilmiştir.

Sosyal medya uygulamaları, sağlık araştırmaları için değerli bir bilgi kaynağı olarak nitelendirilmektedir (Conway ve diğ., 2019). İnternet kullanıcıları, sosyal medya üzerinden sağlıkla ilgili konuları takip etmekte ve kendi kişisel sağlık bilgilerini paylaşmaktadır. Sosyal medya verilerinin doğal dil işleme ve makine öğrenmesi yöntemleri ile değerlendirilmesi, bilgi toplama, sağlık iletişimi, metinsel analiz, yeni çıkan bir hastalığın veya sağlık davranışlarındaki ani değişimlerin tarihsel sürecini anlama gibi alanlarda uygulamalara olanak sunmaktadır. Edo-Osagie ve diğ. (2020), yaptıkları geniş kapsamlı inceleme çalışmasında twitter verileri kullanarak yapılan sağlık bilgilerini toplama, vaka tespiti ve ilaç güvenliği gibi tıbbi uygulama alanlarını araştırmışlardır. Grip gibi enfeksiyon hastalıkları yanında akıl sağlığı gibi enfeksiyonla ilgili olmayan hastalıklarla ilgili çalışmalara da değinmişlerdir. Bu amaçla, makine öğrenmesi yöntemlerinin yanı sıra son zamanlarda derin öğrenme yöntemlerinin de kullanıldığını vurgulamışlardır. Sonuç olarak twitter verilerinin halk sağlığını ilgilendiren alanlarda etkin olarak kullanılabileceği, fakat yarı-öğreticili (semi-supervised) yöntemlerin daha yaygın kullanımının ve araştırmaların pratiğe uygulanmasının gerekliliği üzerinde durmuşlardır. Tavoshi ve diğ., (2020) İtalya'da aşı ile ilgili paylaşılan 693 adet twiter verisini kullanarak bunları aşı taraftarı, aşıya karşı ve çekimser olarak otomatik olarak sınıflandırmışlardır. Bunun için öncelikle twitter metinlerini özellik vektörlerine çevirmek için BOW(Bag of Words) tekniğini kullanmışlardır. Metindeki her ilgili kök kelimenin ağırlığını belirlemek için ise TF-IDF (Term Frequency-Inverse Document Frequency) yöntemini kullanmışlardır. Kullandıkları pek çok öğreticili makine öğrenme algoritmasını 10-katlı çapraz doğrulama yöntemi ile karşılaştırmışlardır. Bunlar içinde en iyi sonuç veren algoritmanın %64 doğrulukla Destek Vektör Makinaları (DVM) olduğunu tespit etmişlerdir.

Bu çalışmada takip ettiğimiz yöntem, twitter verilerini belirli ön işleme adımlarından geçirerek özellik çıkarımı yapmakta ve bu özellikler gözetimli ve gözetimsiz makine öğrenmesi algoritmalarına uygulanarak otomatik olarak sınıflandırılmaktadır. Twitter verilerinden hastalığın varlığı, var ise türü veya hastalığın yokluğu belirlenmiştir. Takip edilen yöntemin başlıca adımları Şekil 1'de verilmiştir.



Şekil 1. Takip edilen yöntemin başlıca adımları
Figure 1. *Main steps of the followed method*

VERİLERİN ELDE EDİLMESİ VE ÖZELLİK ÇIKARIMI (OBTAINING DATA AND FEATURE EXTRACTION)

Veriler, twitter üzerinde herkese açık şekilde paylaşımda bulunulan tweetler arasından toplanmıştır. Tweetler İngilizce olup 3 ana sınıf altında "Sağlıklı", "Alerji", "Nezle" şeklinde değerlendirilmiştir. Veriler araştırmacılar tarafından okunup tweet'i atan kişinin bu 3 sınıftan hangisine uygun olduğuna karar vermesi ile etiketlenmiştir. Python dili ile araştırmacılar tarafından kodlanan bir bot kullanılarak toplam 1032 tweet mesajı toplanmıştır. Toplanan tweetlerde "Alerji" sınıfına ait verileri toplamak için içerisinde "alergy", "high fever", "hyper sensitivity" veya "alergic reaction" ifadelerini içeren tweetler seçilirken, "Nezle" sınıfına ait verileri toplamak için ise "cold", "catorrh", "common cold", "sniffles", "coryza" veya "snuffles" ifadelerini içeren tweetler seçilmiştir. "Sağlıklı" sınıfına ait veriler ise rasgele tweetlerden seçilmiştir. Kelimeler seçilirken içerisinde eşanlamlı kelimeleri bulunduran TheSaurus adlı web sitesi kullanılmıştır. Ayrıca verilerdeki rasgeleliğin arttırılması açısından tweetler 10 farklı şehirden seçilmiştir. Bu şehirler şunlardır; "London", "New York", "Ankara", "İstanbul", "Tokyo", "Helsinki", "Paris", "Washington","Moscow" ve "Stockholm". Bu mesajlar, daha sonra virgül ile ayrılmış değerler (csv) biçiminde bir dosya haline getirilmiştir. Bu verilerin 425'i "Sağlıklı", 307'si "Alerji", 300 tanesi ise "Nezle" sınıfına ait tweetlerden oluşmaktadır.

Doğal dil işlemede, önişleme kısmı modelin anlaşılır ve güvenilir olmasında önemli bir rol oynamaktadır. Bu çalışmada uygulanan önişleme adımları Şekil 2'de verilmiştir. Metodoloji genel olarak anlatılmakla beraber, Şekil 2 üzerindeki işlem adımlarından URL temizlenmesi, Alfanümerik olmayan karakterlerin çıkarılması ve büyük küçük harflerin düzeltilmesi işlemleri, metinsel işlemlerde sıklıkla kullanılan Python Regex kütüphanesi aracılığıyla yapılmıştır. Kelimeler Python ile ayrılıp, köklerine ayırma (Stemming) ve etkisiz kelimelerin (Stopword) atılması adımında Python NLTK(Natural Language Tool Kit) kütüphanesi kullanılmıştır. Yanlış yazılmış olan kelimelerin düzeltilmesi için Levenshtein Distance ile Python Spellchecker modülü kullanılmış olup yanlış olduğu düşünülen kelimeler için gerekli düzeltmeler yapılmıştır. Yanlış olup olmadığı kararı ise kelimenin sözlükte olup olmadığı aracılığı ile kontrol edilerek hata en alt seviyeye indirgenmeye çalışılmıştır. Yapılan işlemlerin ardından ayrılıp düzenlenen metinler, birleştirilerek önişleme kısmı bitirilmiştir.

Öznitelik (özellik) çıkarımı bir veriyi diğer verilerden ayırt edebileceğimiz özellikleri elde ettiğimiz kısımdır. Ayrıca makine öğrenmesi algoritmaları sayılar üzerinde çalıştığı için elimizdeki metinlerin sayısal vektörler haline çevrilmesi gerekmektedir. Bunun için doğal dil işlemede farklı özellik çıkarımı algoritmaları bulunmaktadır (Manning, 1999). Bu makalede doğal dil işleme için kullanılan vektörizasyon metotlarından TF-IDF ve BOW algoritmaları kullanılmış ve karşılaştırılmıştır. Bu algoritmaların oluşturduğu vektörlerin boyutu için 1x1000 değeri seçilmiştir.

N-Gram verilen bir metinde kelimelerin n'erli atomik yapılar olduğu kabul edilir (Cavnar,1994). N değeri 1 iken her bir kelime atomik yapıdayken, N=2 için kelimelerin 2 şerli gruplar halinde oluşturdukları kelime grupları da atomik yapılar olarak ayrıca dikkate alınır. Bu makalede BOW ve TF-IDF algoritmaları için N-Gram(1,2) seçilmiş 1'li ve 2'li kelime grupları dikkate alınmıştır.

BOW algoritması en çok geçen kelimelerden oluşan bir kelime çantası oluşturur (Zhang,2010) ve her bir tweetin içinde bu kelimelerden var olanlar için 1, olmayanlar için 0'lardan oluşan bir vektör oluşturur. Tahmin edilebileceği üzere bu vektörlerin büyük bir çoğunluğu 0'lardan oluşur ve bu matrislere seyrek matris (sparse matrix) adı da verilir. Bu çalışmada, farklı sayılar denenebileceği gibi, parametre olarak 1000 değeri seçilerek 1000 uzunluğundaki vektörler oluşturulmuştur.

Şekil 2. Metin Önişleme Adımları
Figure 2. *Text preprocessing steps*

TF-IDF yöntemi, metinleri istatiksel olarak incelemek için kullanılan bir özellik çıkarım algoritmasıdır. Bu algoritma kullanılarak bir kullanıcının attığı tweetlerin incelenmesiyle yazılan tweetlerde hangi konulara ağırlık verildiği çıkarılabilir. TF ve IDF değerleri hesaplanırken önce TF hesaplanır. TF hesabı için metinde en fazla geçen terim sayısına ihtiyaç vardır. İstenilen terim sayısının metinde en fazla geçen terim sayısına oranı TF değerini verir (Salton,1988). IDF hesabı için ise kullanılacak olan doküman sayısına ihtiyaç vardır. Bu dokümanlar arasında istenilen terimin kaç kez geçtiğinin bilinmesi gerekmektedir. Toplam doküman sayısının terimi içeren doküman sayısına oranının logaritması alınarak hesaplanmaktadır. Burada dikkat edilmesi gereken nokta ise terimi içeren doküman sayısıdır. Hiç bir doküman terimi içermiyorsa 0'a bölünüp belirsiz bir durum oluşabileceği için terimi içeren doküman sayısına 1 eklenmesi alternatif bir çözümdür. Son olarak, bulunan TF ve IDF değerleri çarpılarak sonuca ulaşılır (Robertson, 2004).

GÖZETİMSİZ ÖĞRENME VE K-ORTALAMA ALGORİTMASI (UNSUPERVISED LEARNING AND K-MEANS ALGORITHM)

Gözetimsiz öğrenme, verilerin değerlerinin ("Sağlıklı", "Alerji", "Nezle") algoritmaya verilmeden algoritmanın sadece verilere bakarak bir kümeleme yapmasını, daha sonra da yaptığı kümelemenin gerçekte olması gereken sınıflarla ne kadar eşleştiğini inceleyerek başarı oranını ölçen makine öğrenmesi algoritmalarının genel öğrenme biçimidir. K-Ortalama algoritması en temel gözetimsiz öğrenme algoritmalarından birisidir. K, oluşturulacak sınıf sayısını temsil etmektedir. Veriler k farklı kümeden birisine dahil edilmeye çalışılır. Algoritma en başta uzayda k noktada rasgele olarak küme merkezleri oluşturur. Ardından, her veri için bu noktalardan olan uzaklıkları hesaplanır. Uzaklıklarına göre her veri bir kümeye dahil edilir. Daha sonra yeni küme merkezleri, kümelerin ağırlık merkezine göre veya farklı yöntemlere göre seçilir ve yeni küme merkezlerine göre kümelere dağılma işlemi tekrarlanır. Bu tekrarlanmalar, kümelenme kararlı bir hale gelene kadar veya belirlenen iterasyon sayısı bitene kadar devam eder. Kümelenmenin kararlı hale gelip gelmemesi durumu da uzaklık ölçütlerinin sonuçlarına göre değerlendirilebilir. (Hartigan,1979). Ancak K-Ortalama algoritmasının bazı temel sorunları bulunmaktadır. İlk sorun kaç farklı küme olacağının belirlenmesi işlemidir. Araştırmada her ne kadar 3 sınıf olduğunu bilsek de, 3 sınıf olması kümelenmenin 3 yerde olacağı anlamına gelmemektedir. Farklı kümelenmeler tek bir sınıfı da belirtiyor olabilir. Mesela 6 farklı kümelenme bulunabilir. 3 farklı sınıf için olası durumlardan birisi, 3 kümelenmenin 1.sınıfı belirtmesi, 2 kümelenmenin 2.sınıfı belirtmesi, ve son kümelenmenin ise 3.sınıfı belirtmesi durumudur ki bu örnekte de anlaşıldığı üzere sınıf sayımız belirli olsa dahi bu küme sayısı konusunda bizim farklı tercihler yapabileceğimiz anlamını taşımaktadır. Bu durumun kararını vermek için WCSS (Within Cluster Sum of Squares) yöntemi izlenmiştir. Bu yöntemi kısaca anlatmak gerekirse uzaklıkların kareler toplamını her küme için toplayıp küme n için bir değer ortaya çıkarır. İstenilen sonuç, bu değerin olabildiğince düşük bir değer olmasıdır. Fakat 0 olması için aslında n'in veri sayısı kadar çok olması gerekebilir. Bu yüzden burada dikkat edilmesi gereken nokta bizim için belirli sınıflar var ise onu kullanmamız veya n sayısındaki ufak bir değişiklik ile bize doğrulukta yüksek kazanç sağlayacak ara bir küme sayısı belirlemektir. Genel olarak böyle durumlarda, dirsek noktası denilen, türevi diğer bölgelere göre yüksek olan kısımları seçmek mantıklı olacaktır. Bu değeri hesaplamamızın altında yatan bir diğer sebep ise aynı kümeye ait bazı tweetlerin küme sayısının arttırılması ile kendi aralarında kümelenme olasılığının olmasıdır. Böyle bir durumda, örnek vermek gerekirse, nezle sınıfına ait olan tweetler 2 farklı yerde kümelenmiş ise, oluşan bu durumda küme sayısı

4 olsaydı WCSS değeri ile bu problemin varlığı hakkında yorum yapılabilir ve problem çözülebilirdi. WCSS değeri 1 numaralı eşitlikte görüldüğü gibi hesaplanmaktadır.

$$WCSS(k) = \sum_{j=1}^{k} \sum_{x_j \in \text{küme } j} \left\| x_i - \bar{x_j} \right\|^2 \quad (1)$$

Burada, $\bar{x_j}$ değeri j. kümedeki örneklem ortalamasıdır.

Noktaların uzayda birbirine olan uzaklıklarının ölçülmesi için Öklid uzaklığı kullanılmıştır (Aloise, 2009). K-Ortalama algoritmasının ikinci bir zayıf noktası ise başlangıç merkezlerinin rasgele olarak seçilmesidir. Başlangıçta rasgele seçilen k merkez, eğer istenmeyen noktalara giderse, ki bu ihtimal gayet kuvvetlidir, oluşacak kümelenmeler de buna göre olacağı için bu rasgelelik olumsuz sonuçlar doğurabilir. Bu sorunu çözmek amacıyla araştırmada başlangıç değeri için farklı değerler denenmiştir. Farklı değerler ile deneme işlemi ise random_state parametresi olarak belirtilen bir değişken ile kontrol edilmiştir. Bu sayede rasgelelik durumu kontrol altına alınıp deterministik hale getirilmiştir. Böylece veri uzayında bir tarama işlemi gerçekleştirilerek en iyi başlangıç noktalarını veren random_state değişkeni aranmıştır. Bu işlem gerçekleştirilirken kıstas olarak ise başarı oranı tercih edilmiştir.

GÖZETİMLİ ÖĞRENME VE KULLANILAN ALGORİTMALAR (SUPERVISED LEARNING AND THE ALGORITHMS USED)

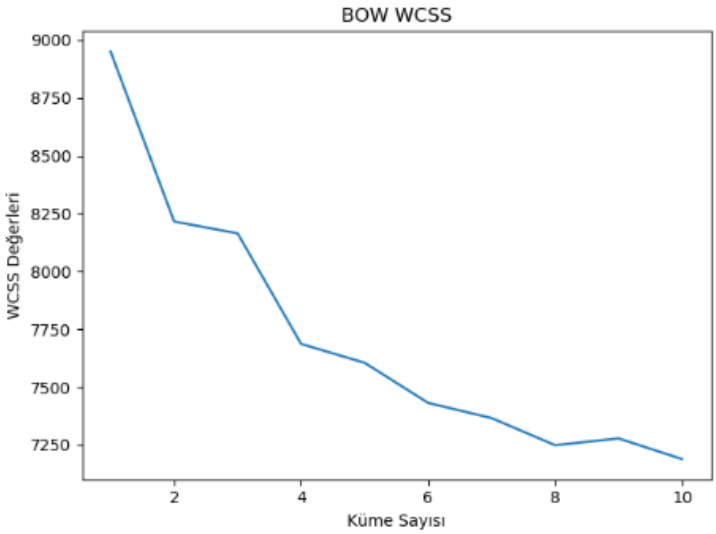
Gözetimli öğrenme, gözetimsiz öğrenmeden farklı olarak algoritmaya, veriye ait özelliklerin çıkış değerleri ("Sağlıklı", "Alerji", "Nezle") ile birlikte verilmesiyle gerçekleştirilir. Bu çalışmada, 7 farklı gözetimli öğrenme algoritması, TF-IDF ve BOW ile elde edilen özellik vektörleri üzerinde denenmiş ve sonuçlar elde edilmiştir. Kullanılan algoritmalar, k-En Yakın Komşu (kNN), Karar Ağacı (Decision Tree), Lojistic Regresyon (Logistik Regression), Naïve Bayes, Rasgele Orman (Random Forest), Destek Vektör Makinesi (Support Vector Machine), Topluluk Öğrenmesi (Ensemble Learning) algoritmalarıdır. Parametre optimizasyonu için ızgara arama (grid search) yöntemi kullanılmış (Lerman, 1980) ve belirtilen parametreler arasından en iyi olanları seçilmiştir. Güvenirliliği arttırmak ve aşırı öğrenmeyi (overfitting) engellemek için de 5-katmanlı çapraz doğrulama (cross-validation) yöntemi kullanılmıştır (Srivastava, 2014). Veri 5 parçaya bölünmüş, ardından 4 parça eğitim için 1 parça ise test için kullanılmıştır. Böylece, her algoritma 5 kez çalıştırılarak her veri, hem test için hem de eğitim için kullanılarak bunlar arasında en yüksek başarı oranı, en düşük başarı oranı, standart sapma ve ortalama başarı değerleri ölçülmüştür (Kohavi, 1995).

DENEYSEL SONUÇLAR (EXPERIMENTAL RESULTS)

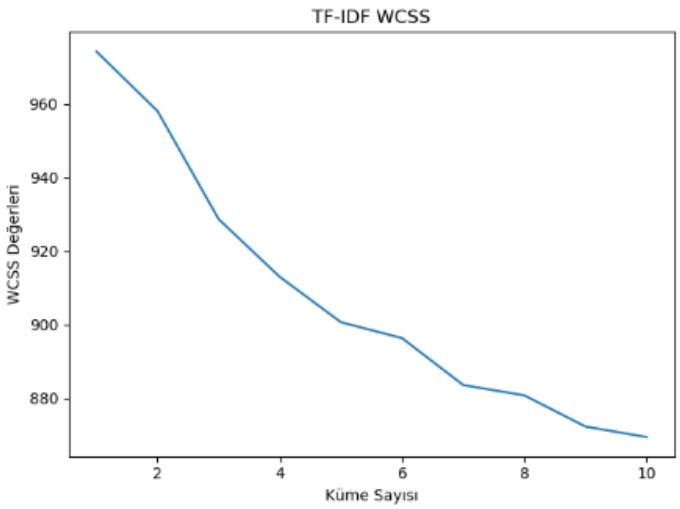
Bu bölümde araştırma boyunca yapılan deneylerin sonuçları raporlanmış ve yorumlanmıştır. Temel olarak gözetimli ve gözetimsiz algoritmalar ayrılırken, bunlar da kendi içerisinde kullandıkları öznitelik çıkarımı yöntemleri olan TF-IDF ve BOW algoritmalarına göre ayrılmışlardır.

Gözetimsiz Öğrenme Sonuçları (Results of Unsupervised Learning)

BOW ve TF-IDF yöntemleri için elde edilen farklı WCSS değerleri sırasıyla Şekil 3 ve Şekil 4'te görülmektedir. Şekil 3 ve Şekil 4'te görüldüğü üzere BOW ve TF-IDF için hesaplanan WCSS değerlerinde, küme sayısının artması ile yüksek bir kazanç elde edilememiştir. Bu durumdan dolayı her iki yöntem için de küme sayısını değiştirmek gerekli görülmemiştir.

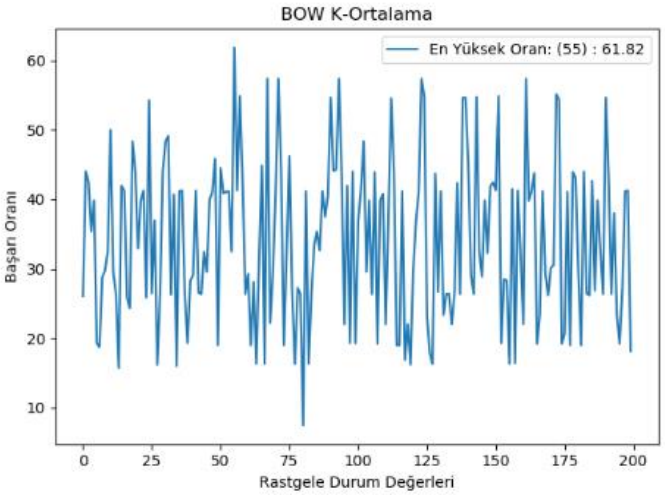


Şekil 3. BOW için WCSS değerleri
Figure 3. *WCSS values for BOW*



Şekil 4. TF-IDF için WCSS değerleri
Figure 4. *WCSS values for TF-IDF*

K-Ortalama algoritmasının başlangıç için belirlenen rasgele durum (random_state) parametresi için 0 ila 200 arasında ki tüm rasgele durum değerleri denenmiş ve sonuçlar BOW ve TF-IDF için ayrı ayrı ölçülmüştür.



Şekil 5. BOW İçin K-Ortalama rasgele durum değerleri

Figure 5. *K-Means random state values for BOW*

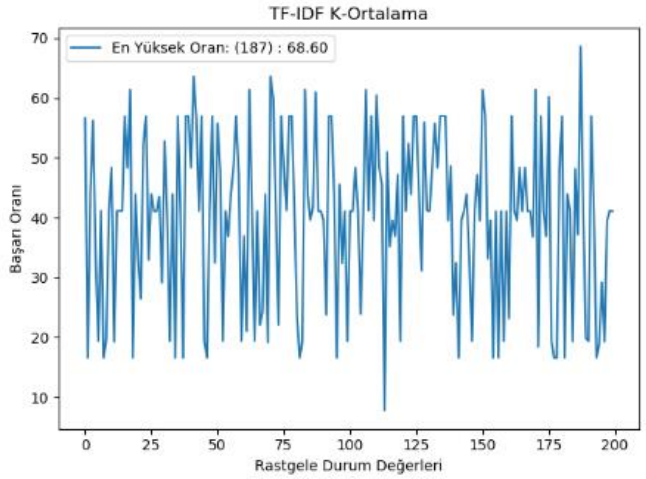
Şekil 5'de görüldüğü üzere en yüksek başarı oranı rasgele durum parametresinin 55 değeri ile 61.82 olarak ölçülmüştür. Bu durumda iken oluşan karmaşıklık matrisi Çizelge 1'de verildiği gibidir.

Çizelge 1. BOW K-Ortalama için karmaşıklık matrisi

Table 1. *Confusion matrix for BOW K-Means*

	Tahmin	**Tahmin**	**Tahmin**
	Sağlıklı	**Alerji**	**Nezle**
Sağlıklı	425	0	0
Alerji	167	139	1
Nezle	226	0	74

BOW kullanımında K-Ortalama algoritması için, alerji olarak tahmin edilen tweetlerde %100 oranında doğruluk elde edilirken, nezle diye tahmin edilen tweetlerin %98.66 olarak bulunması başarılı olduğu kısımlardır. Geliştirilmesi gereken kısım hastalığın olmadığı söylenen ancak nezle veya alerji bulunan durumlardır. Algoritmanın sağlıklı olarak sınıflandırdığı tweetler için hassasiyet (precision) değeri %51.95 olduğu Çizelge 1'den, $425/(425 + 167 + 226)$ şeklinde hesaplanabilir. Şekil 6'da görüldüğü üzere en yüksek doğruluk rasgele durum parametresinin 187 değeri ile 68.60 olarak ölçülmüştür. Bu durumdayken oluşan karmaşıklık matrisi Çizelge 2'de verildiği gibidir.



Şekil 6. TF-IDF İçin K-Ortalama Rasgele Durum Değerleri
Figure 6. *K-Means random state values for TF-IDF*

TF-IDF için kümeleme işlemi nispeten daha başarılı sayılabilecek durumdadır. Alerji olarak tahmin edilen tweetlerde %99.27 oranında doğruluk elde edilirken, nezle diye tahmin edilen tweetlerde bu oranın %99.32 olarak bulunması algoritmanın başarılı olduğu kısımlardır. BOW'a göre daha başarılı bir sonuç veren diğer kısım ise hastalık yok diye kümelenmiş olan kısımdır. Burada elde edilen başarı oranı %56.83'dür. Genel olarak bakıldığında TF-IDF ile özellik çıkarımının, algoritmanın daha başarılı çalışmasına yardım ettiği görülmektedir.

Çizelge 2. TF-IDF K-Ortalama İçin Karmaşıklık Matrisi
Table 1. *Confusion matrix for TF-IDF K-Means*

	Tahmin	**Tahmin**	**Tahmin**
	Sağlıklı	**Alerji**	**Nezle**
Sağlıklı	424	1	0
Alerji	170	136	1
Nezle	152	0	148

Gözetimli Öğrenme Algoritmaları İçin Sonuçlar (Results for Supervised Learning Algorithms)

Gözetimli öğrenme algoritmaları BOW ve TF-IDF metotları ile edilen özellik vektörleri ile denenmiştir. K-En Yakın Komşu (kNN) algoritması [{'n_neighbors':[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15], 'metric':['minkowski','euclidean'],'weights':['uniform','distance']}] parametreleri BOW ve TF-IDF için çalıştırılmış ve sonucunda BOW için en iyi parametreler n_neighbors=1, metric='minkowski', weights='distance' olarak bulunurken, TF-IDF için en iyi parametreler {'metric': 'minkowski', 'n_neighbors': 1, 'weights': 'uniform'} olarak bulunmuştur. Çapraz doğrulama katsayısı değeri de daha önceden söylendiği gibi 5 olarak seçilmiştir. Karar Ağacı için Criterion='Entropy', girilmiş tek parametredir. Lojistik Regresyon için herhangi bir parametre değiştirilmemiş, model Sklearn kütüphanesinin geçerli parametreleri ile oluşturulmuştur. Naive Bayes (Gaussian) için herhangi bir parametre değiştirilmemiş, model Scikit Learn kütüphanesinin geçerli parametreleri ile oluşturulmuştur. Rasgele Orman için parametreler n=7 (ağaç sayısı), max_depth=30 şeklinde belirlenmiştir. Destek Vektör Makinesi için [{'C':[1,2,3,4,5],'kernel':['linear','poly','rbf'], 'gamma': ['scale', 1,0.5,0.1,0.01,0.001]}] denenmiş ve en iyi parametreler olarak {'C': 1, 'gamma': 1, 'kernel': 'linear'} bulunmuştur. Topluluk Öğrenmesi algoritması için Rasgele Orman, Karar Ağacı ve Lojistik Regresyon algoritmaları arasında çoğunluk oylaması yapılmıştır. Algoritmalar bahsedilmeyen tüm parametreler için varsayılan değerleri ile çalıştırılmıştır. Kullanılan algoritmalara ait parametre havuzu ve seçilen değerler Çizelge 3'te verilmiştir.

Çizelge 3. Algoritmaların parametre seçimleri
Table 3. *Parameter selection of algorithms*

Algoritmalar	Parametre Havuzu	Seçilen Değerler (Sırasıyla)
kNN + BOW	[{'n_neighbors':[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15], 'metric':['minkowski','euclidean'],'weights':['uniform','distance']}]	1, "minkowski", "distance"
kNN + TF-IDF		1, "minkowski", "uniform"
Karar Ağacı + BOW ve TF-IDF	Criterion='Entropy',	'Entropy'
Lojistik Regresyon + BOW ve TF-IDF	Scikit-Learn varsayılan parametreleri	Varsayılan
Naive Bayes + BOW ve TF-IDF	Scikit-Learn varsayılan parametreleri	Varsayılan
Rasgele Orman + BOW ve TF-IDF	n (ağaç sayısı)=[1,2,3,4,…,15], max_depth=[1,2,3,…,30]	7, 30
DVM + BOW DVM + TF-IDF	[{'C':[1,2,3,4,5],'kernel':['linear','poly','rbf'], 'gamma': ['scale', 1,0.5,0.1,0.01,0.001]}]	1, 1, "linear"
Topluluk Öğrenmesi + BOW ve TF-IDF	Scikit-Learn varsayılan parametreleri	Varsayılan

BOW ve TF-IDF özellik çıkarım yöntemleri için gözetimli öğrenme algoritmalarının Ortalama Doğruluk Oranı (ODO), Doğruluk Standart Sapması(DSS), En Düşük Doğruluk Oranı (EDDO), En Yüksek Doğruluk Oranı (EYDO) değerleri Çizelge 4'te verilmiştir.

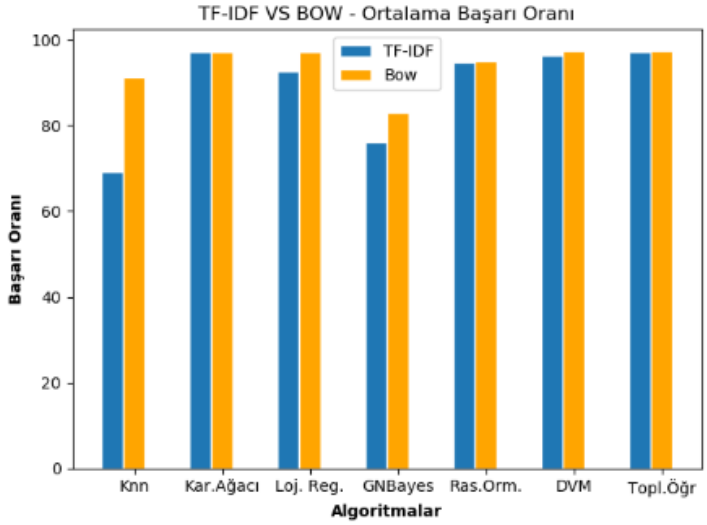
Çizelge 4. TF-IDF ve BOW ile gözetimli öğrenme algoritmaların doğruluk oranları
Table 4. *Accuracy rates of supervised learning algorithms with TF-IDF and BOW*

	BOW				TF-IDF			
Algoritmalar	**ODO**	**DSS**	**EDDO**	**EYDO**	**ODO**	**DSS**	**EDDO**	**EYDO**
kNN	%91.37	%2.25	%87.86	%94.68	%69.25	%1.22	%67.14	%70.53
Karar Ağacı	%97.19	%1.34	%95.65	%99.02	%97.19	%1.15	%95.65	%98.54
Lojistik Regresyon	%97.19	%1.19	%95.65	%99.02	%92.63	%2.28	%89.80	%96.13
Naive Bayes	%83.04	%1.93	%80.09	%85.43	%75.96	%2.67	%72.33	%79.22
Rasgele Orman	%95.06	%2.35	%92.27	%98.54	%94.67	%2.18	%92.23	%97.57
DVM	%97.48	%0.71	%96.60	%98.54	%96.31	%1.42	%94.66	%98.05
Topluluk Öğrenmesi	%97.48	%0.93	%96.60	%99.02	%97.09	%1.80	%94.20	%99.02

Gözetimli Öğrenme Algoritmalarının Karşılaştırılması (Comparison of Supervised Learning Algorithms)

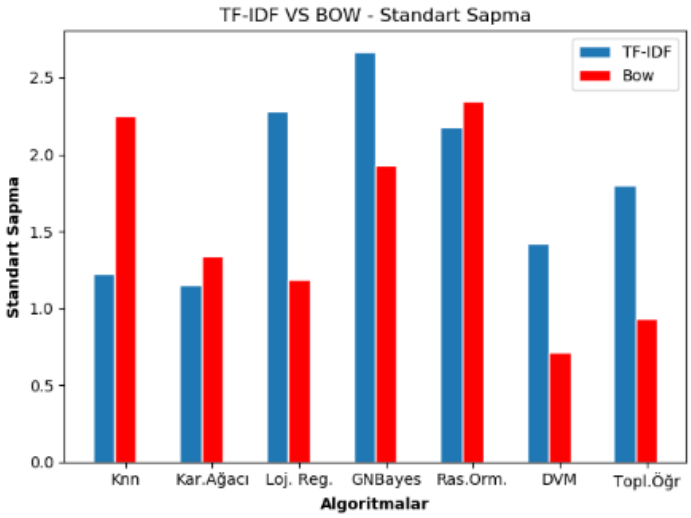
Gözetimli öğrenme algoritmalarını kendi aralarında karşılaştırmak gerekirse, bu 4 farklı ölçüme göre yapılabilir. Bunlar, en yüksek başarı oranı, en düşük başarı oranı, ortalama başarı oranı ve standart sapmadır. Ortalama başarı oranına göre gözetimli öğrenme algoritmalarının karşılaştırılması Şekil 7'de

görülmektedir. Bunların ne kadar tutarlı olduğunu ölçmek için de, Şekil 8'de standart sapmalar kullanılmıştır.



Şekil 7. TF-IDF ve BOW için algoritmaların ortalama doğrulukları
Figure 7. *Mean accuracies of the algorithms for TF-IDF and BOW*

Şekil 7 incelenerek başarı oranlarının BOW ile TF-IDF'ye göre genel olarak daha iyi sonuçlar verdiği gözlemlenebilir.



Şekil 8. TF-IDF ve BOW için algoritmaların standart sapmaları
Figure 8. *Standard deviations of the algorithms for TF-IDF and BOW*

Şekil 8'e bakıldığında TF-IDF veya BOW yöntemlerinin, ilgili algoritmanın standart sapması üzerinde etkili olduğu görülmektedir. Bu yüzden algoritmalar için her iki özellik çıkarımı da denenmiş ve başarı oranı eşit olan durumlarda standart sapmayı daha düşük olarak veren algoritma tercih edilmiştir.

Ayrıca TF-IDF ve BOW için en yüksek doğruluk oranı ve hassasiyet değerleri Çizelge 5 üzerinden incelenebilir ve algoritmaların sonuçları hakkında daha fazla bilgiye sahip olunabilir. Hassasiyet (Precision) değerinin anlamı bizim için algoritmanın tahmini ve gerçekte olan değeri arasındaki ilişkidir. Örnek vermek gerekirse Rasgele Orman algoritmasının TF-IDF ile kullanıldığı durumda başarı oranı %97.57 olarak ölçülmüştür. Ancak algoritmanın Sağlıklı dediği örnekleri incelersek, algoritmanın bunları doğru olarak sınıflandırıp Sağlıklı demiş olduğu durumlar %95.7'dir. Başka bir deyişle

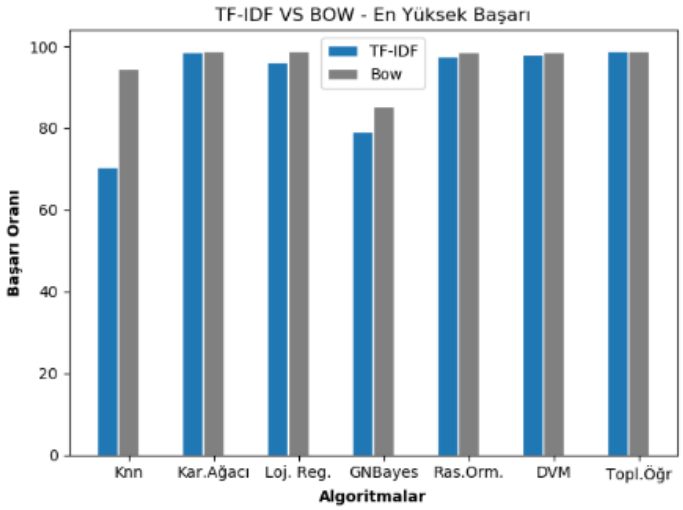
algoritmanın Sağlıklı dediği ancak Alerji veya Nezle Sınıfına ait %4.3'lük bir kısım vardır. Nezle için ise bu oran %100'dür bunun anlamı model bir veri için Nezle tahmininde bulunduysa bu oran %100 olarak tutmuştur ve Nezle için verilen tüm örnekleri doğru olarak sınıflandırmıştır.

Çizelge 5. TF-IDF ve BOW İle Gözetimli Öğrenme Algoritmaların En Yüksek Doğruluk Oranları
Table 5. Maximum *accuracy rates of supervised learning algorithms with TF-IDF and BOW*

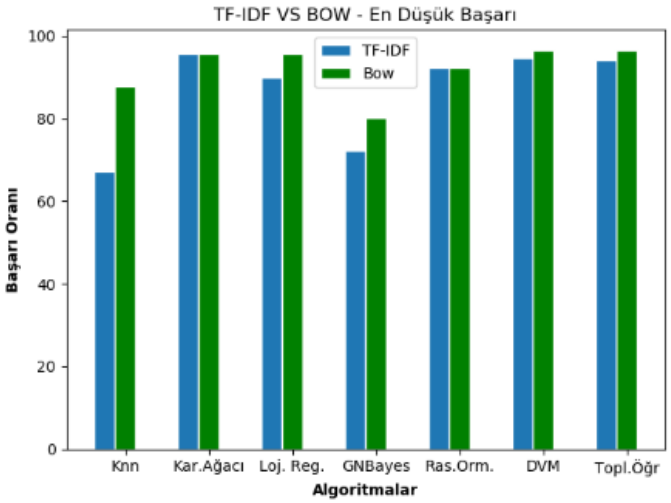
Algoritmalar	Başarı	Sağlıklı	Alerji	Nezle
kNN + BOW	%94.6	%89	%100	%100
kNN + TF-IDF	%70.53	%60.6	%97.6	%82
Karar Ağacı + BOW	%99.02	%98.9	%98.3	%100
Karar Ağacı + TF-IDF	%98.54	%98.9	%96.7	%100
Lojistik Regresyon + BOW	%99.02	%97.8	%100	%100
Lojistik Regresyon + TF-IDF	%96.13	%91.2	%100	%100
Naive Bayes + BOW	%85.43	%95.9	%83.6	%77
Naive Bayes + TF-IDF	%79.22	%83.9	%76.4	%78
Rasgele Orman + BOW	%98.54	%96.7	%100	%100
Rasgele Orman + TF-IDF	%97.57	%95.7	%98.3	%100
DVM + BOW	%98.54	%98.9	%96.7	%100
DVM + TF-IDF	%98.05	%95.7	%100	%100
Topluluk Öğrenmesi + BOW	%99.02	%97.8	%100	%100
Topluluk Öğrenmesi + TF-IDF	%99.02	%97.8	%100	%100

Standart sapmalar göz önünde bulundurularak tahmin edilebilse de, çapraz doğrulama ile yapılan her eğitim ve testin ardından ulaşılan en yüksek ve en düşük doğruluk durumlarını daha detaylı incelemek için Şekil 9 ve Şekil 10'a bakılabilir. Şekil 9 ve Şekil 10 incelendiğinde BOW'un TF-IDF'ye göre neredeyse tüm algoritmalarda daha iyi bir sonuca yardımcı olduğu görülmektedir. En yüksek ve en düşük değerlerinde daha iyi sonuçlar elde etmek için BOW ile özellik çıkarımı yapmanın TF-IDF'ye göre daha iyi sonuçlar verdiği ortadadır. Ancak bu durum veri kümesi ile de ilgilidir. Veri kümesinin bu şekilde tweetlerden oluştuğu sistemler için BOW'un TF-IDF'ye göre daha iyi olduğunu söylemek mümkündür.

Elde edilen sonuçlara göre, ortalama başarı değeri olarak en yüksek değerler Destek Vektör Makinesi ve Topluluk Öğrenmesi Algoritmaları ile BOW kullanılarak elde edilmiş olan %97.48'lik doğruluk oranlarıdır.



Şekil 9. TF-IDF ve BOW için algoritmaların en yüksek doğruluk oranları
Figure 9. *Maximum accuracy rates of the algorithms for TF-IDF and BOW*



Şekil 10. TF-IDF ve BOW için algoritmaların en düşük doğruluk oranları
Figure 9. *Minimum accuracy rates of the algorithms for TF-IDF and BOW*

Doğruluk oranları eşit olan algoritmalar arasında güvenirliği daha fazla olan algoritmayı seçmek gerekmektedir. Bundan dolayı Destek Vektör Makinesi ve Topluluk Öğrenmesinin BOW ile elde edilmiş sonuçları arasında standart sapmaya bakılmalı ve düşük olan seçilmelidir. Bu, daha güvenli bir modeldir. Destek Vektör Makinesine ait standart sapma değerinin %0.71, Topluluk Öğrenmesine ait standart sapma değerinin %0.93 olduğu görülmektedir. Karşılaştırmanın sonucu olarak BOW ile kullanılan DVM algoritması en başarılı olarak bulunan algoritmadır.

SONUÇ VE TARTIŞMA (RESULT and DISCUSSION)

Genel olarak gözetimli öğrenmenin gözetimsiz öğrenmeye göre çok daha başarılı sonuçlar gösterdiği görülmüştür. TF-IDF ve BOW için ise, gözetimsiz öğrenmede TF-IDF'nin daha iyi sonuçlar vermesine rağmen gözetimli öğrenmede her ne kadar benzer sonuçlar çıkarsalar da BOW'un biraz daha önde olduğu gözlenmiştir. Bununla beraber elde edilen yüksek başarı oranları, tweetler baz alınarak bölgesel veya küresel olarak yaygın hastalıklarda artış veya azalışların gözlenebilmesi ve fark edilmesinde kullanılabileceği gibi, bu hastalıklarla başka değişkenlerin aralarında korelasyon olup olmadığını inceleyen araştırmalara da temel olabilir. Özellikle dünyanın küresel salgın ile uğraştığı bu günlerde, hastalığın ne kadar yayıldığını ölçmek için sadece testlerdeki oranlar dikkate alınmaktadır. Ancak birçok ülkede aylar geçmesine rağmen kendi nüfuslarının sadece %1-2'lik kısımlarına test yapılabilmiştir. Yapılan testlerin pozitif olan her kişi için 2 defa yapıldığı da göz önünde bulundurulursa sayıların yanıltıcı olabileceği açıktır. Normal zamanlarda bu kadar yoğun hastane başvurusuna ve test kiti kullanımına ihtiyaç duyulmayacağından, ülkelerin bu yöntemi her durumda izlemesi aşırı maliyetli ve verimsiz olacaktır. Bu durumda alternatif çözümler aranması gerektiği kaçınılmazdır. Twitter veya diğer micro blog platformlarındaki verilerin hastalıkların yayılma hızının tespiti için de kullanılabileceğinden dolayı, bu çalışmanın bu gibi alanlarda da rahatlıkla kullanılabileceğini düşünmekteyiz. Yaptığımız çalışma, farklı hastalıklar ile ilgili tweetler için de kolay biçimde adapte edilebilir olduğundan dolayı sadece 3 farklı sınıf için sınırlı değildir. İstendiği takdirde farklı hastalık durumları için eklemeler yapılabilir. Bu esneklik sayesinde birçok farklı çalışma buradaki adımları yineleyebilir ve farklı durumlar için sonuçlar gözlemlenebilir.

KAYNAKLAR (REFERENCES)

Aloise, D., Deshpande, A., Hansen, P., Popat, P., 2009, "NP-hardness of Euclidean sum-of-square clustering", *Machine learning*, Cilt 75, Sayı 2, ss. 245-248.

Ambert, K. H., Cohen, A.M., 2009, "A System for Classifying Disease Comorbidity Status from Medical Discharge Summaries Using Automated Hotspot and Negated Concept Detection", *Journal of the American Medical Informatics Association*, Cilt 16, Sayı 4, ss. 590–595.

Acherkar, H., Gandhe, A., Lazarus, R., Yu, S., Liu, B., 2011, "Predicting Flu Trends using Twitter Data", *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Shanghai, China, 702-706.

Cavnar, W. B., Trenkle, J. M., 1994, "N-gram-based text categorization.", *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, Las Vegas, Nevada, A.B.D., 161-175.

Conmay, M., Hu, M., Chapman W.W., 2019, "Recent Advances in Using Natural Language Processing to Address Public Health Research Questions Using Social Media and ConsumerGenerated Data", Yearbook of Medical Informatics, Cilt 28, Sayı 1, ss. 208-217.

Dai, X., Bikdash, M., 2015, "Hybrid Classification for Tweets Related to Infection with Influenza", *Proceedings of the IEEE SoutheastCon 2015*, Fort Lauderdale, Florida, 1-5.

Dai, X., Bikdash, M., 2016, "Distance-based Outliers Method for Detecting Disease Outbreaks using Social Media", *Proceedings of the IEEE SoutheastCon 2015*, Norfolk, VA, USA, 1-8.

Edo-Osagie, O., Iglesia, B.D.L., Lake, I., Edeghere, O., 2020, "A scoping review of the use of Twitter for public health research", Computers in Biology and Medicine, Available Online, 103770, doi: 10.1016/j.compbiomed.2020.103770.

Hartigan, J.A., Wong, M. A., 1979, "Algorithm AS 136: A k-means clustering algorithm", *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, Cilt 28, Sayı 1, ss. 100-108.

Kohavi, R., 1995, "A study of cross-validation and bootstrap for accuracy estimation and model selection", *IJCAI'95 Proceedings of The 14th International Joint Conference on Artificial Intelligence*, Montreal, Quebec, Canada, 2: 1137-1143.

Lerman, P.M., 1980, "Fitting segmented regression models by grid search", *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, Cilt 29, Sayı 1, ss. 77-84.

Manning, C., Schütze, H., 1999, "Foundations of Statistical Natural Language Processing", MIT press, Cambridge, MA, A.B.D.

Morita, M., Maskawa, Aramaki, S., E., 2013, "Comparing Social Media and Search Activity as Social Sensors for the Detection of Influenza", *5th International Symposium of Languages in Biology and Medicine*, Tokyo, Japan, 75-79.

Salton, G., Buckley, C., 1988, "Term-weighting approaches in automatic text retrieval", *Information Processing & Management*, Cilt 24, Sayı 5, ss. 513-523.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014, "Dropout: a simple way to prevent neural networks from overfitting", *The Journal of Machine Learning Research*, Cilt 15, Sayı 1, ss. 1929-1958.

Robertson, S., 2004, "Understanding inverse document frequency: on theoretical arguments for IDF", *Journal of Documentation,* Cilt 60, Sayı 5, ss. 503-520.

Rudra, K., Sharma, A., Gaungly, N., Imran, M., 2017, "Classifying Information from microblogs during epidemics", *Proceedings of the 2017 International Conference on Digital Health*, London, United Kingdom, 104-108.

Rudra, K., Sharma, A., Gaungly, N., Imran, M., 2018, "Classifying and Summarizing Information from Microblogs During Epidemics", Information Systems Frontiers, Cilt 20, Sayı 1, ss. 933-948.

Tavoschi L., Quattrone F., D'Andrea E., Ducange P., Vabanesi M., Marcelloni F., Lopalco P.L., 2020, "Twitter as a sentinel tool to monitor public opinion on vaccination: an opinion mining analysis from September 2016 to August 2017 in Italy", Human Vaccines & Immunotherapeutics, Available Online, doi: 10.1080/21645515.2020.1714311.

Zhang, Y., Jin, R., Zhou, Z., 2010, "Understanding bag-of-words model: a statistical framework", *International Journal of Machine Learning and Cybernetics*, Cilt 1, Sayı 4, ss. 43-52.