Düzce Üniversitesi Bilim ve Teknoloji Dergisi

Araştırma Makalesi

Farklı Sınıflandırma Algoritmaları ve Metin Temsil Yöntemlerinin Duygu Analizinde Performans Karşılaştırılması[1]

Batuhan Cem ÖĞE [a,*], Fatih KAYAALP [b]

[a] *Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Zonguldak Bülent Ecevit Üniversitesi, Zonguldak, TÜRKİYE*

[b] *Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Düzce Üniversitesi, Düzce, TÜRKİYE*

** Sorumlu yazarın e-posta adresi: batuhan.oge@beun.edu.tr*

DOI: 10.29130/dubited.1015320

ÖZ

Son yıllarda internete erişim imkanlarının artması ve kullanıcılardaki akıllı telefon kullanımının yaygınlaşması sebebiyle sosyal medya olarak adlandırılan ve insanların çeşitli konulardaki fikirlerini paylaştığı servisler çok yaygın olarak kullanılmaktadır. Sosyal medya verilerinin analiz edilmesiyle insanların farklı konulardaki duygularına dair anlamlı çıkarımlarda bulunulması anlamına gelen ve temelde bir sınıflandırma işlemi olan Duygu Analizi çalışmaları son yıllarda öne çıkan çalışma alanlarından biridir. Bu çalışmada, Python programlama dili içindeki kütüphaneler kullanılarak Naive Bayes (NB), Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF) ve Artificial Neural Network (ANN) gibi 6 adet sınıflandırma algoritmasının Duygu Analizi kapsamında, performans karşılaştırması yapılmıştır. Veri seti olarak, açık kaynaklı, IMDB sitesinde yer alan etiketli kullanıcı yorumları kullanılmıştır. Doğal Dil İşleme yöntemleri kullanılarak temizlenen veri setinin sayısal olarak temsil edilebilmesi için Bag of Words (BoW), TF-IDF, FastText ve Word2Vec metin temsil yöntemleri kullanılmıştır. Veri setinin eğitimi ve test edilmesi aşamasında k=5 olacak şekilde k-fold cross validation yöntemi kullanılmıştır. 6 farklı sınıflandırma yöntemi için elde edilen sonuçlar accuracy, precision, recall ve f1 score hesaplanarak ayrıntılı bir karşılaştırma yapılmış ve sonuçlar kaydedilmiştir. En yüksek accuracy değerleri olarak LR ve SVM sırasıyla BOW'da %86, TF-IDF'te %87, word2Vec'de %87 ve FastText'te %83 seviyelerinde benzer sonuçlar vermiştir.

Anahtar Kelimeler: *Doğal Dil İşleme, Duygu Analizi, Makine Öğrenmesi, Metin Temsil, Sınıflandırma, Veri Madenciliği.*

Performance Comparison of Different Classification Algorithms and Text Representation Methods in Sentiment Analysis

ABSTRACT

Due to the increase in internet access opportunities and the widespread use of smartphones in recent years, services called social media where people share their opinions on various issues are widely used. Sentiment Analysis studies, which means making meaningful inferences about people's emotions on different subjects by analyzing social media data, and which is basically a classification process, is one of the prominent fields of study in recent years. In this study, 6 classification methods such as Naive Bayes (NB), Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF) and Artificial Neural Network (ANN) were used by using libraries in Python programming language. Within the scope of Sentiment Analysis of the algorithm, performance comparison was made. As the dataset, open source, labeled user comments on the IMDB site were used. Bag of Words (BoW), TF-IDF, FastText and Word2Vec text representation methods were used to represent the data set that was cleaned using Natural Language Processing methods. During the training and testing of the data set, the k-fold cross validation method was used, with k=5. The results obtained for 6 different classification methods were calculated by calculating accuracy, precision, recall and f1 score, and a detailed comparison was made and the results were recorded. As the highest accuracy values, LR and SVM gave similar results at 86% in BOW, 87% in TF-IDF, 87% in word2Vec and 83% in FastText, respectively.

Keywords: *Natural Language Processing, Sentiment Analysis, Machine Learning, Text Representation, Classification, Data Mining..*

[1] ICAIAME 2021 konferansında sunulmuş olup, tam metin olarak basılmıştır.

Geliş: 27/10/2021, Düzeltme: 17/12/2021, Kabul: 21/12/2021

I. GİRİŞ

Son yıllarda internete erişim imkanlarının artması ve kullanımı gittikçe artan, insanların çeşitli konulardaki fikirlerini paylaştığı servisler olarak adlandırılan sosyal medya platformlarının çoğalması ile çok büyük miktarda işlenmemiş veri ortaya çıkmıştır. Kullanıcılar tarafından bu servislere girilen ve birçok farklı platformda depolanmakta olan veriler çeşitli veri madenciliği yöntemleriyle analiz edilerek anlamlı bilgi çıkarımları yapılmaya çalışılmaktadır. Sosyal medya verilerinin analiz edilmesiyle insanların farklı konulardaki duygularına dair anlamlı çıkarımlarda bulunulması anlamına gelen Duygu Analizi çalışmaları da bu konuda öne çıkan çalışma alanlarından biridir. İnsanların sanal ortamlarda gerçek düşüncelerini daha rahat ve özgürce paylaştığı düşünüldüğünde, paylaşılan mesajlardan duygu analizi yapabilmek, yüz yüze sohbet edilen bir insanın duygusunu anlamaktan çok daha kolay olabilmektedir. Duygu analizi, bir metinden fikir çıkarmak, dönüştürmek ve yorumlamak ve bunları olumlu, olumsuz veya doğal duygular olarak sınıflandırmak için Doğal Dil İşleme'yi (DDİ) kullanan bir yaklaşımdır [1]. Duygu analizi, cümle düzeyi, belge düzeyi ve özellik düzeyi olmak üzere üç farklı düzeye ayrılmıştır. Amaç, düşünceyi cümle, belge veya özelliklerden olumlu ve olumsuz duygu olarak sınıflandırmaktır [2].

A. LİTERATÜR TARAMASI

400 binden fazla tüketici yorumu ile yapılan bir duygu analizi çalışmasında, kullanıcı yorumları word2vec yöntemiyle vektör gösterim şekline dönüştürülerek analiz edilmiş, 10 katlı çapraz doğrulama yöntemi kullanılarak Support Vector Machine, Naive Bayes, Logistic Regression ve Random Forest algoritmaları yardımıyla sonuçlar elde edilmiştir. Elde edilen sonuçlar incelendiğinde en yüksek doğruluk değerine Random Forest algoritması ile ulaşılmıştır [3].
Veri ön işleme adımlarının değerlendirmesinin yapıldığı bir çalışmada, Twitter mesajlarından oluşan veri seti için duygu analizi çalışması yapılmış, Support Vector Machine, Naive Bayes, Logistic Regression ve Evrişimli Sinir Ağır (Convolutional Neural Network) gibi 4 popüler makine öğrenmesi algoritması yardımı ile elde edilen sonuçlar karşılaştırılmıştır. Yapılan gözlemler sonucunda kelimeleri kök hallerine dönüştürme, rakamları silme, kesme işaretini kaldırma gibi yöntemlerin doğruluk oranını olumlu yönde etkilediği, noktalama işaretleri kaldırmanın ise doğruluk değerine çok fazla katkı sağlamadığı görülmüştür [4].
Facebook mesajlarının veri seti olarak kullanıldığı bir diğer çalışmada, Endonezya başkanlık seçimleri öncesinde sosyal medyada paylaşılan mesajlar Naive Bayes sınıflandırma algoritması kullanılarak analiz edilmiş ve elde edilen sonuçlar ile gerçek sonuçlar karşılaştırılmıştır [5].
Aşı ve aşılanma ile ilgili tweet'lerden oluşan bir veri setinin kullanıldığı çalışmada, insanların aşı ile ilgili görüşleri bag of words ve Support Vector Machine yöntemleri kullanılarak analiz edilmiş ve sonuçlar sunulmuştur [6].

II. VERİ MADENCİLİĞİ

Veri madenciliği, sonuçları tahmin etmek için büyük veri kümeleri içindeki anormallikleri, kalıpları ve bağlantıları bulma sürecidir. Veri Madenciliği çeşitli alanlarda kullanılmaktadır. Telekomünikasyon, medya ve teknoloji şirketleri, müşteri davranışlarını tahmin etmek, hedefe yönelik ve müşteri odaklı kampanyalar sunmak adına veri madenciliği yöntemlerini kullanırlar. Veri madenciliği, eğitimcilerin öğrenci verilerine erişmesine, başarı düzeylerini tahmin etmesine ve ekstra dikkat gerektiren öğrencileri veya öğrenci gruplarını belirlemesine yardımcı olur.

A. SINIFLANDIRMA YÖNTEMLERİ

A. 1. Naive Bayes

Son yıllarda duygu analizi çalışmalarında sıkça kullanılmaya başlayan Naive Bayes yöntemi, Bayes Teoremine dayanan bir sınıflandırma yöntemidir. Naive Bayes sınıflandırıcıları, belirli bir sınıftaki

elemanların (özelliklerin) birbirleriyle olan yakınlık bağlantılarının kesilmesi bekler. Naive Bayes yöntemi, metni birden çok sınıfa ayırma işinde geniş çapta kullanılmaktadır [7].

A. 2. Support Vector Machine

Support Vector Machine (SVM) yönteminin duygu analizi çalışmalarında yüksek performans gösterdiği bilinmektedir. SVM metni araştırır, yapılacak seçimlerin sınırlarını karakterize eder ve girdiler içerisinde gerçekleştirilen hesaplama için bileşenleri kullanır. Gerekli bilgi, her biri m boyutundaki iki vektör şeklinde sunulur. Bu noktada, vektör olarak ifade edilen her bir veri bir sınıfa atanır. Daha sonra makine, eğitim aşamasında örnekler aracılığı ile herhangi bir yerden uzak olan iki sınıf arasındaki sınır noktasını belirler. Yapılan ayrım sınıflandırma sınırının netleşmesine yardımcı olur. Bu aşamada sınıf kenarlarının genişletilmesi kararsız (sınıflandırılamayan) seçenekleri azaltır [8].

A. 3. Logistic Regression

Logistic Regression, adına rağmen Genelleştirilmiş Doğrusal Modeller yöntemlerine ait popüler bir algoritmadır ve Maksimum Entropi olarak da bilinir. Bu modelde, tek bir denemenin olası sonuçlarını tanımlayan olasılıklar, bir lojistik fonksiyon kullanılarak modellenir. Logistic Regression yöntemi, bir veya birden fazla bağımsız değişken olduğunda çıktıyı veya sonucu belirlemek için kullanılır. Çıkış değeri 0 veya 1, yani ikili biçimde olur [9].

A. 4. Decision Tree

Decision Tree sınıflandırmasında, verileri bölmek için bir koşul kullanılır. Koşulu sağlayan veriler bir sınıfa, kalan veriler diğer sınıfa yerleştirilir. Bu yinelemeli bir süreçtir. Birden fazla ayırma yöntemi vardır. Bunlar, sınıflandırma yapmak için belirli kelime varlığını veya yokluğunu arayan tek öznitelik bölme ve belgedeki kelimeleri önceden tanımlanmış kelimelerle eşleştiren benzerlik tabanlı çoklu öznitelik bölmedir [10].

A. 5. Random Forest

Random Forest sınıflandırma algoritması, sınıflandırma ve regresyon için bir öğrenme yöntemidir. Eğitim aşaması sırasında bir dizi karar ağacı oluşturulur. Yeni gelen durumu sınıflandırmak için yeni durum ağaçların her birine gönderilir. Her ağaç sınıflandırma yapar ve sonuç olarak bir sınıf çıkarır. Çıktı sınıfı, çoğunluk oylamasına dayalı olarak çeşitli ağaçlar tarafından oluşturulan maksimum benzer sınıf sayısı dikkate alınarak seçilir. Random Forest yöntemi, hem profesyoneller hem de sıradan insanlar için çok az araştırma ve programlama gerektirir, öğrenmesi ve kullanması kolaydır. Güçlü bir istatistiksel altyapıya sahip olmayan kişiler tarafından bile rahatlıkla kullanılabilir [11].

A. 6. Artificial Neural Network

Yapay Sinir Ağları algoritmasının ana fikri, verilerin doğrusal birleşiminden özelliklerini çıkarmak ve sonrasında elde edilen bu bilgiyi özelliklerin doğrusal olmayan bir fonksiyonu olarak modellemektir. Sinir ağları, birbirleriyle belirli şekillerde bağlanan düğümlerin yer aldığıu bir ağ şeması olarak karşımıza çıkar. Düğümler bir katmanda düzenlenir. Mimari olarak sinir ağları üç katmandan oluşur: giriş katmanı, çıkış katmanı ve gizli katman. İleri beslemeli ve geri beslemeli olmak üzere iki tür sinir ağı mevcuttur. İleri beslemeli sinir ağlarında düğümler sadece tek bir yönde bağlı olduklarından bu sinir ağı türü duygu analizi çalışmaları için daha uygundur. Düğümler arasındaki her bağlantının, gradyan iniş algoritması yardımıyla hata fonksiyonu en aza indirilerek elde edilen bir ağırlık değeri vardır. Bir nöron, iki aşamada bir değer veren bir matematiksel modelden oluşur. İlk aşamada, nöron girdisinin ağırlıklı toplamı hesaplanır ve bu toplama bir aktivasyon fonksiyonu uygulanarak çıktı alınır. Aktivasyon işlevi doğası gereği doğrusal olmayan bir işlevdir ve tüm ağdaki giriş verileri yardımıyla önceden öğrenilen doğrusal olmayan bir işlevi tahmin edebilmesini sağlar [12].

III. METİN MADENCİLİĞİ

Metin madenciliği, belgelerdeki ve veri tabanlarındaki yapılandırılmamış metni, analize uygun normalleştirilmiş, yapılandırılmış verilere dönüştürmek için doğal dil işlemeyi kullanan bir yapay zeka teknolojisidir.

A. METİN ÖN İŞLEME İŞLEMLERİ

Kullanılan metnin özelliğine ve alındığı kaynağa göre farklı metin ön işleme (temizleme) işlemleri mevcuttur. Bu çalışmada, harf dışındaki tüm sembollerin metinden çıkarılması, tüm harflerin küçük harfe çevrilmesi, etkisiz kelimelerin (stopwords) çıkarılması ve kelimelerin kök hallerine getirilmesi (stemming) yöntemleri kullanılmıştır.

B. METİN TEMSİL YÖNTEMLERİ

B. 1. Bag of Words

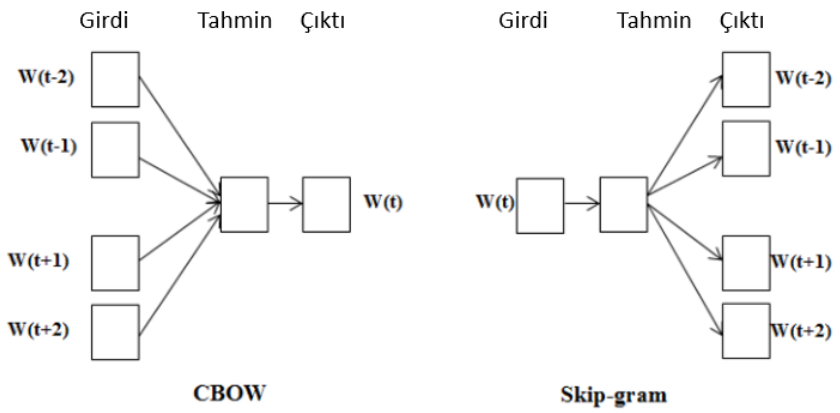
Bag of Words (kelime torbası) yöntemi, doğal dil işleme ve bilgi çıkarımında kullanılan, bir belge içerisinde yer alan verinin sayısal olarak temsil edilerek özelliklerinin çıkarılması işlemidir. Bu yöntemde bir belge, kendi içerisinde yer alan tüm kelimelerin bir çantası olarak temsil edilir. Duygu analizi çalışmalarında kelime torbası yöntemini kullanmak, belge içerisinde yer alan duyguyu ifade etmesi açısından faydalı kelimelerin bir listesini oluşturmak anlamına gelir.

	1 This	2 movie	3 is	4 very	5 scary	6 and	7 long	8 not	9 slow	10 spooky	11 good	Yorum Uzunluğu
Yorum 1	1	1	1	1	1	1	1	0	0	0	0	7
Yorum 2	1	1	2	0	0	1	1	0	1	0	0	7
Yorum 3	1	1	1	0	0	0	1	0	0	1	1	6

Şekil 1. *Bag of words yöntemi ile yorumların temsil edilmesi.*

B. 2. Word2Vec

Word2vec, büyük bir veri kümesinden sözcük yerleştirmeyi inceleyen sinir ağı tabanlı bir modeldir. Yüksek boyutlu bir uzayda her kelime için bir vektör üretir. word2vec, vektör temsili oluşturmak için iki mimari içerir: sürekli kelime torbası ve skip-gram. CBoW (continuous bag of words) modeli, kelimeleri çevreleyen bağlamlarını kullanarak tahmin eder. Skip-gram modeli, mevcut kelimenin ifade ettiği bağlamı tahmin etmek için kullanır.



Şekil 2. *Word2Vec Mimarisi.*

B. 3. Tf-Idf

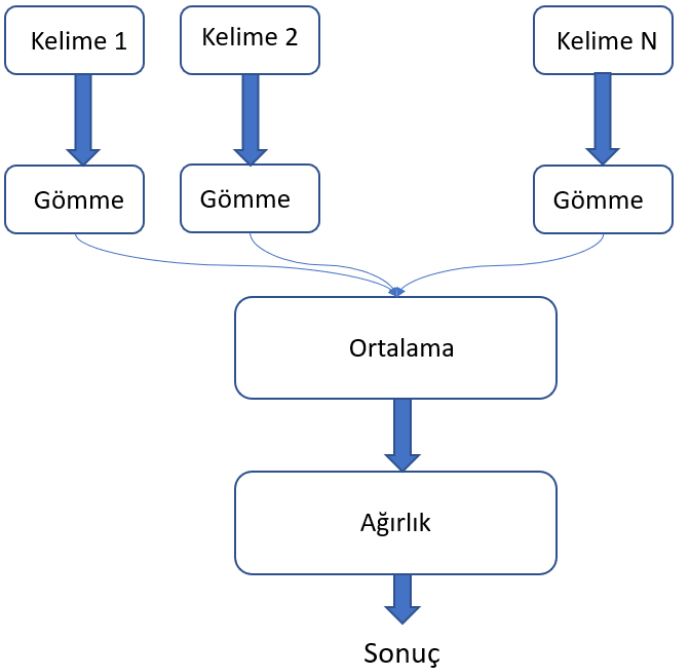
TF-IDF, bir terimin bir metin içerisinde ne sıklıkla geçtiği (TF) ve o terimin metin içerisinde ne kadar önem arz ettiği (IDF) bilgisini kullanan bir çıkarım tekniğidir. Her kelime veya terimin kendi TF ve IDF puanları vardır. Bir terimin TF ve IDF çarpım puanları, o terimin TF*IDF ağırlığını belirler. Basitçe anlatmak gerekirse, bir terimin TF*IDF puanı (ağırlık) ne kadar yüksekse terim o kadar nadirdir, bu değer ne kadar düşük ise bu terim o kadar yaygındır.

Kelime	Yorum 1	Yorum 2	Yorum 3	TF(Yorum 1)	TF(Yorum 2)	TF(Yorum 3)	IDF	TF-IDF(Yorum 1)	TF-IDF(Yorum 2)	TF-IDF(Yorum 3)
This	1	1	1	1/7	1/8	1/6	0.00	0.00	0.00	0.00
movie	1	1	1	1/7	1/8	1/6	0.00	0.00	0.00	0.00
is	1	2	1	1/7	1/4	1/6	0.00	0.00	0.00	0.00
very	1	0	0	1/7	0	0	0.48	0.068	0.00	0.00
scary	1	1	0	1/7	1/8	0	0.18	0.025	0.022	0.00
and	1	1	1	1/7	1/8	1/6	0.00	0.00	0.00	0.00
long	1	0	0	1/7	0	0	0.48	0.068	0.00	0.00
not	0	1	0	0	1/8	0	0.48	0.00	0.060	0.00
slow	0	1	0	0	1/8	0	0.48	0.00	0.060	0.00
spooky	0	0	1	0	0	1/6	0.48	0.00	0.00	0.080
good	0	0	1	0	0	1/6	0.48	0.00	0.00	0.080

Şekil 3. *Örnek Yorumlar için TF-IDF puan hesaplanması.*

B. 4. FastText

FastText, bir metin içerisinde tüm bağlamdan bir kelimeyi veya merkezdeki tüm kelimelerden bağlamı öğrenerek çalışır. Öğrenme, iki katman ağırlık ve üç katman nöron içeren bir sinir ağında bir dizi güncelleme olarak görülebilir; burada iki dış katmanın her birinde kelime dağarcığındaki her bir kelime için bir nöron bulunur ve orta katmanda gömme alanının boyutları kadar nöron bulunur.



Şekil 4. *Örnek bir FastText modeli.*

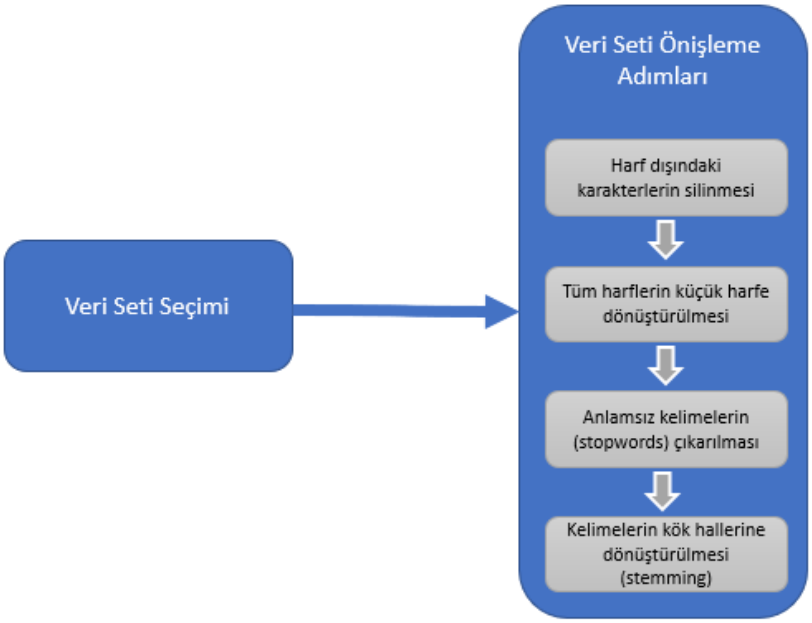
IV. METOT

A. VERİ SETİNİN OLUŞTURULMASI VE TEMİZLENMESİ

Bu çalışmada kullanılan veri seti, IMDB internet sayfasında yer alan film yorumlarından oluşmaktadır. Toplam 50.000 adet yorumdan oluşan veri setinde 25.000 adet pozitif, 25.000 adet negatif olarak etiketlenmiş yorum bulunmaktadır [13].

Veri setinin analize hazır hale getirilebilmesi için öncelikli olarak 50.000 adet yorumun tek bir dosyada toplanması gerekmektedir. Orijinal veri seti 25.000 adet pozitif ve 25.000 adet negatif olmak üzere toplam 50.000 adet bireysel .txt dosyasından oluşmaktadır. Veriler tek bir dosya altında toplanarak analizler tek bir dosya üzerinden yapılmıştır. MATLAB programı kullanılarak yazılan kod sayesinde tüm yorumlar tek bir dosya haline getirilmiştir. Veri setinin son hali 50.000 satır ve 2 sütundan oluşmakta olup ilk sütun film yorumunu ikinci sütun ise yorumun etiketini (pozitiflik veya negatiflik) belirtmektedir.
Veri seti tek bir dosya haline getirildikten sonra Jupyter Notebook programına verinin aktarılması yani analizlerin yapılabilmesi için verinin bu programlara tanıtılması gerekmektedir. Bu işlem için Jupyter Notebook'ta (python dilinde) "pandas" kütüphanesi kullanılarak "pd.read_csv" fonksiyonu ile veri seti içeri aktarılmıştır.

Veri seti programa tanıtılmış olsa bile şu anki haliyle analize tam olarak hazır değildir. Duygu analizinin verimli bir şekilde yapılabilmesi için veri setinin bir takım temizleme aşamalarından geçmesi gerekmektedir. Metin ön işleme işlemleri olarak adlandırılan bu aşamada veri setinden harf dışındaki tüm semboller çıkarılmıştır. Bunun için "re.sub" fonksiyonu kullanılmıştır. Ardından tüm harfler küçük harfe çevrilmiştir. Bunun için "lower" fonksiyonu kullanılmıştır. Sonrasında etkisiz kelimeler (stopwords) veri setinden çıkarılmıştır. Bunun için "remove_stopwords" fonksiyonu kullanılmıştır. Son olarak kelimeler kök hallerine (stemming) getirilmiştir. Bunun için "stem" fonksiyonu kullanılmıştır. Bu metin ön işleme işlemlerinin ardından veri seti duygu analizi için hazır hale getirilmiştir.



Şekil 5. *Veri Setinin Temizlenmesi için Uygulanan Adımlar.*

B. VERİ SETİNİN EĞİTİM VE TEST SETLERİ OLARAK AYRILMASI

Veri seti temizlendikten ve analize hazır hale getirildikten sonra yapılması gereken bir sonraki adım veri setinin eğitim ve test setleri olarak ayrılmasıdır. Bu aşamada farklı yaklaşımlar izlenebilir. Bu çalışmada k-fold cross validation (k-katlı çapraz doğrulama) yöntemi kullanılmıştır. Veri seti 25.000 pozitif ve 25.000 negatif yorumdan oluştuğu için dengeli dağılan bir veri setidir. Veri setinin orijinal halinde ilk 25.000 satır pozitif, kalan 25.000 satır negatif yorumlardan oluşmaktadır. Veri setinden parçalar alarak eğitim ve test setleri şeklinde ayırmadan önce veri seti rastgele karıştırılmış ve ardından bu karıştırılan veri seti k=5 olacak şekilde 5 parçaya ayrılarak %80 eğitim, %20 test olacak şekilde işlenmiştir. Bu yöntemin avantajı tüm veri setinin hem eğitim hem de test için kullanılması ve analiz sonuçlarının her bir kat için ortalamasının alınarak optimum şekilde elde edilebilmesidir. K-fold cross validation yapılabilmesi için gerekli kodlar yazılmış, gerekli döngü yapısı kurularak her bir k değeri için eğitim ve test setleri oluşturulmuştur.

C. METİN TEMSİL YÖNTEMLERİNİN UYGULANMASI

C. 1. Bag of Words

Bag of Words modelinin oluşturulabilmesi için python dilinde bulunan “bag of words” fonksiyonu kullanılmıştır. Burada dikkat edilmesi gereken nokta bag of words modeli oluşturulduktan sonra seçilecek kelime sayısıdır. Bag of words modeli oluştuğunda elimizdeki veri setinde yer alan tüm kelimeleri ve bu kelimelerin veri setinde kaç kez geçtiği bilgisi elde edilmiş olunur. Bu çalışmada bag of words değeri olarak 1500 değeri seçilmiş ve sınıflandırma algoritmaları gerçeklenmiştir.

C. 2. Word2vec

Word2vec modelinin oluşturulabilmesi için python dilinde bulunan “word2vec” fonksiyonu kullanılmıştır. Burada dikkat edilmesi gereken nokta model oluşturulurken word2vec fonksiyonunun içine girilecek dimension (boyut) parametresinin değeridir. Bu çalışmada bu değer 300 olarak belirlenmiş ve tüm programlarda model bu şekilde oluşturulmuştur. Bu yöntem ile veri setindeki her bir kelime 300’lük bir vektör ile ifade edilir ve duygu analizi her bir yorumdaki tüm kelimelerin vektör değerleri karşılaştırılarak yapılır.

C. 3. Tf-Idf

TF-IDF modelinin oluşturulabilmesi için python dilinde bulunan “tf-idf” fonksiyonu kullanılmıştır. Bag of words modeliyle benzerlik göstermesi ve karşılaştırmanın daha verimli yapılabilmesi için bu yöntemde de “max_count” değeri 1500 olarak alınmış ve sınıflandırma algoritmaları gerçeklenmiştir.

C. 4. FastText

FastText modelinin oluşturulabilmesi için python dilinde bulunan “fasttext” kütüphanesi kullanılmış ve hazır olarak kullanılabilen fasttext modeli içeri aktarılarak edilerek veri setinin fastText modeli oluşturulmuştur. Word2vec modeli ile benzerlik göstermesi ve daha sağlıklı bir karşılaştırma yapılabilmesi için bu yöntemde de vektör sayısı 300 olarak belirlenmiştir.

D. SINIFLANDIRMA YÖNTEMLERİNİN UYGULANMASI

D. 1. Naive Bayes

Naive Bayes sınıflandırma algoritması, python dilinde “GaussianNB” fonksiyonu kullanılarak gerçeklenmiştir. 5-fold cross validation yöntemi kullanıldığı için her bir kat ile algoritma çalıştırılmış,

her bir kat ayrı ayrı eğitilmiş ve test edilmiştir. 5 kat için elde edilen hata matrislerinin ortalaması alınarak Naive Bayes yöntemi için ortalama hata matrisi kaydedilmiştir.

D. 2. Support Vector Machine

Support Vector Machine sınıflandırma algoritması, python dilinde "SVC" fonksiyonu kullanılarak gerçeklenmiştir. 5-fold cross validation yöntemi kullanıldığı için her bir kat ile algoritma çalıştırılmış, her bir kat ayrı ayrı eğitilmiş ve test edilmiştir. 5 kat için elde edilen hata matrislerinin ortalaması alınarak Support Vector Machine yöntemi için ortalama hata matrisi kaydedilmiştir.

D. 3. Logistic Regression

Logistic Regression sınıflandırma algoritması, python dilinde "LogisticRegression" fonksiyonu kullanılarak gerçeklenmiştir. 5-fold cross validation yöntemi kullanıldığı için her bir kat ile algoritma çalıştırılmış, her bir kat ayrı ayrı eğitilmiş ve test edilmiştir. 5 kat için elde edilen hata matrislerinin ortalaması alınarak Logistic Regression yöntemi için ortalama hata matrisi kaydedilmiştir.

D. 4. Decision Tree

Decision Tree sınıflandırma algoritması, python dilinde "DecisionTreeClassifier" fonksiyonu kullanılarak gerçeklenmiştir. 5-fold cross validation yöntemi kullanıldığı için her bir kat ile algoritma çalıştırılmış, her bir kat ayrı ayrı eğitilmiş ve test edilmiştir. 5 kat için elde edilen hata matrislerinin ortalaması alınarak Decision Tree yöntemi için ortalama hata matrisi kaydedilmiştir.

D. 5. Random Forest

Random Forest sınıflandırma algoritması, python dilinde "RandomForestClassifier" fonksiyonu kullanılarak gerçeklenmiştir. 5-fold cross validation yöntemi kullanıldığı için her bir kat ile algoritma çalıştırılmış, her bir kat ayrı ayrı eğitilmiş ve test edilmiştir. 5 kat için elde edilen hata matrislerinin ortalaması alınarak Random Forest yöntemi için ortalama hata matrisi kaydedilmiştir.

D. 6. Artificial Neural Network

Bu çalışmada kullanılan diğer sınıflandırma yöntemlerinden farklı olarak Artificial Neural Network sınıflandırma algoritmasının çalıştırılabilmesi için öncelikli olarak bir yapay sinir ağı yapısının oluşturulması gerekmektedir. Yapılan araştırmalar sonucunda sinir ağı oluşturulurken aktivasyon fonksiyonu olarak "ReLu" fonksiyonu kullanılmış, çift gizli katman kullanılmış ve sayıları 10 olarak belirlenmiş, "batch size" değeri 32 olarak alınmıştır. 5-fold cross validation yöntemi kullanıldığı için her bir kat ile algoritma çalıştırılmış, her bir kat ayrı ayrı eğitilmiş ve test edilmiştir. Her bir k değeri için döngü başa döndüğünde sinir ağır tekrar oluşturularak herhangi bir tutarsızlık oluşmaması sağlanmıştır. 5 kat için elde edilen hata matrislerinin ortalaması alınarak Artificial Neural Network yöntemi için ortalama hata matrisi kaydedilmiştir.

E. KARŞILAŞTIRMA PARAMETRELERİ VE DONANIM BİLGİSİ

Bu çalışmada 4 farklı metin temsil yöntemi, 6 farklı sınıflandırma yöntemi kullanılarak elde edilen sonuçların sağlıklı şekilde karşılaştırılabilmesi ve hangi yöntemin nasıl sonuç verdiğinin verimli bir şekilde anlaşılabilmesi için öncelikli olarak her bir sınıflandırma algoritmasından hata matrisleri elde edilmiştir. Hata matrisinde yer alan değerler kullanılarak accuracy (doğruluk), precision (kesinlik derecesi), recall (anımsama) ve f1-score (f1 puanı) değerleri hesaplanmış, her bir yöntemin ne kadar sürdüğü sonuç tablosuna eklenmiştir. Ayrıca, HWINFO isimli program yardımıyla ortalama bellek ve işlemci kullanımları hesaplanmıştır. Hesaplamanın verimli bir şekilde yapılabilmesi için simülasyonlar

sırasında bilgisayarda herhangi başka bir program çalıştırılmamış ve başka bir işlem yapılmamıştır. Bu sayede verimli bir şekilde karşılaştırma sonuçları elde edilmiştir.

Bu çalışmada tüm simülasyon sonuçları Intel(R) Core(TM) i7-4700HQ CPU @ 2.40GHz, 12 GB RAM'e sahip bir laptop kullanılarak elde edilmiştir. Windows işletim sistemi olarak kurulu olan bilgisayarda 480GB BX500 CT480BX500SSD1 2.5" SATA 3.0 SSD yer almaktadır.

V. DENEY SONUÇLARI

A. SINIFLANDIRMA YÖNTEMLERİNİN PERFORMANSLARININ KARŞILAŞTIRILMASI

Bu çalışmada 4 farklı metin temsil yöntemi ile 6 farklı sınıflandırma yöntemi kullanılarak gerçeklenen duygu analizleri sonucunda 5-fold cross validation yöntemi kullanılarak hata matrisleri elde edilmiş, bu hata matrisleri yardımıyla karşılaştırmaların yapılacağı değerler hesaplanmıştır. Hata matrisi, 2x2 boyutunda bir matris olup içerisinde yer alan değerler aşağıdaki gibidir:

- True Positive (gerçek pozitif): Doğru şekilde pozitif olarak sınıflandırılan yorumlar.
- True Negative (gerçek negatif): Doğru şekilde negatif olarak sınıflandırılan yorumlar.
- False Positive (yanlış pozitif): Yanlış şekilde pozitif olarak sınıflandırılan yorumlar.
- False Negative (yanlış negatif): Yanlış şekilde negatif olarak sınıflandırılan yorumlar.

Accuracy (doğruluk) değerinin hesaplanabilmesi için aşağıdaki formül kullanılmıştır.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

Precision (kesinlik derecesi) değerinin hesaplanabilmesi için aşağıdaki formül kullanılmıştır.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall (anımsama) değerinin hesaplanabilmesi için aşağıdaki formül kullanılmıştır.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1-score (F1 değeri) değerinin hesaplanabilmesi için aşağıdaki formül kullanılmıştır.

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

B. ELDE EDİLEN SONUÇLAR

Accuracy, precision, recall ve f1-score değerleri için en iyi sonucu veren iki yöntem Logistic Regression ve Support Vector Machine olmuştur. Çalışma süresi açısından en hızlı yöntem word2vec ve fastText ile Naive Bayes, en yavaş yöntem ise bag of words ile Support Vector Machine olmuştur. Karşılaştırma için seçilen sınıflandırma yöntemleri arasında çalışma prensibi gereği en yavaş yöntemin Support Vector Machine olduğu görülmüştür. Bunun nedeni olarak Support Vector Machine algoritmasının iki ana ayar parametresi kullanması ve bu durumun karmaşıklığı, dolayısıyla çalışma süresini etkilediği tespit

edilmiştir. Ortalama bellek kullanımı açısından en fazla bellek kullanan yöntem fastText ile Logistic Regression, en az bellek kullanan yöntem ise word2vec ile Support Vector Machine olmuştur. Ortalama işlemci kullanımı açısından en fazla işlemci kullanan yöntem TF-IDF ile Logistic Regression, en az işlemci kullanan yöntem word2vec ile Random Forest olmuştur.

Tablo 1. *Python ile Elde Edilen Tüm Sonuçlar.*

Sınıflandırma Algoritması	**Metin Temsil Yöntemi**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**Süre** *(Sn)*	**Bellek Kullanımı** *(GB)*	**İşlemci Kullanımı** *(GHz)*
Naive Bayes	Bag of Words	0,76	0,82	0,67	0,74	11,34	7,93	0,37
	TF-IDF	0,81	0,81	0,81	0,81	11,39	8,33	0,44
	Word2vec	0,77	0,77	0,78	0,77	2,48	6,35	0,35
	FastText	0,70	0,70	0,71	0,70	2,52	9,82	0,40
Support Vector Machine	Bag of Words	0,87	0,86	0,88	0,87	87968,68	7,76	0,34
	TF-IDF	0,87	0,86	0,88	0,87	4342,28	7,68	0,34
	Word2vec	0,87	0,87	0,88	0,87	1400,32	6,07	0,34
	FastText	0,84	0,83	0,85	0,84	1707,09	8,08	0,35
Logistic Regression	Bag of Words	0,87	0,86	0,88	0,87	38,64	8,11	1,27
	TF-IDF	0,87	0,87	0,88	0,88	33,08	8,65	1,35
	Word2vec	0,87	0,87	0,88	0,87	9,7	6,55	1,27
	FastText	0,83	0,82	0,84	0,83	8,74	10,08	1,26
Random Forest	Bag of Words	0,79	0,82	0,73	0,78	35,52	7,78	0,33
	TF-IDF	0,79	0,82	0,74	0,78	47,54	8,22	0,35
	Word2vec	0,81	0,83	0,77	0,80	46,7	6,33	0,33
	FastText	0,73	0,77	0,68	0,72	49,38	9,87	0,34
Decision Tree	Bag of Words	0,72	0,72	0,71	0,72	158,32	7,68	0,33
	TF-IDF	0,72	0,72	0,72	0,72	217,13	8,10	0,35
	Word2vec	0,74	0,74	0,74	0,74	126,98	6,62	0,34
	FastText	0,67	0,67	0,67	0,67	128,31	9,90	0,33
Artificial Neural Network	Bag of Words	0,82	0,83	0,81	0,82	838,96	7,71	0,77
	TF-IDF	0,82	0,83	0,80	0,81	911,13	7,91	0,79
	Word2vec	0,87	0,87	0,88	0,87	578,78	7,53	0,75
	FastText	0,84	0,84	0,85	0,84	575,75	9,88	0,76

VI. SONUÇ

Bu çalışmada, Python programlama dili içindeki kütüphaneler kullanılarak Naive Bayes, Logistic Regression, Support Vector Machine, Decision Tree, Random Forest ve Artificial Neural Network gibi 6 adet sınıflandırma algoritmasının Duygu Analizi kapsamında, performans karşılaştırması yapılmıştır. Veri seti olarak, açık kaynaklı, IMDB internet sitesinde yer alan etiketli kullanıcı yorumları kullanılmıştır. Doğal Dil İşleme yöntemleri kullanılarak temizlenen veri setinin sayısal olarak temsil edilebilmesi için Bag of Words, TF-IDF, Word2Vec ve FastText metin temsil yöntemleri kullanılmıştır. Veri setinin eğitimi ve test edilmesi aşamasında k=5 olacak şekilde k-fold cross validation yöntemi kullanılmıştır. 6 farklı sınıflandırma yöntemi için elde edilen sonuçlar accuracy, precision, recall ve f1

score hesaplanarak ayrıntılı bir karşılaştırma yapılmış ve sonuçlar kaydedilmiştir. Elde edilen sonuçlar incelendiğinde en iyi performans word2vec ile LR yöntemi kullanılarak elde edilmiştir.

V. KAYNAKLAR

[1] B. Agarwal, N. Mittal, P. Bansal, and S. Garg, "Sentiment analysis using common-sense and context information," *Computational Intelligence and Neuroscience*, vol. 2015, pp. 1–9, 2015.

[2] N. Mishra and C. K. Jha, "Classification of opinion mining techniques," *International Journal of Computer Applications*, vol. 56, no. 13, pp. 1–6, 2012.

[3] B. Bansal ve S. Srivastava, "Sentiment classification of online consumer reviews using word vector representations", *Procedia Computer Science*, vol. 132, pp. 1147–1153, 2018.

[4] S. Symeonidis, D. Effrosynidis, ve A. Arampatzis, "A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis", *Expert Systems with Applications*, vol. 110, pp. 298–310, 2018.

[5] B. Haryanto, Y. Ruldeviyani, F. Rohman, T. N. Julius Dimas, R. Magdalena, ve F. Muhamad Yasil, "Facebook analysis of community sentiment on 2019 Indonesian presidential candidates from Facebook opinion data", *Procedia Computer Science*, vol. 161, pp. 715–722, 2019.

[6] E. D'Andrea, P. Ducange, A. Bechini, A. Renda, ve F. Marcelloni, "Monitoring the public opinion about the vaccination topic from tweets analysis", *Expert Systems with Applications*, vol. 116, pp. 209–226, 2019.

[7] A. Alsaeedi and M. Z. Khan, "A study on sentiment analysis techniques of Twitter data," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 2, pp. 361–374, 2019.

[8] J. Khairnar and M. Kinikar, "Machine learning algorithms for opinion mining and sentiment classification," *International Journal of Scientific and Research Publications*, vol. 3, no. 6, pp. 1–6, 2013.

[9] A. Tyagi and N. Sharma, "Sentiment Analysis using logistic regression and effective word score heuristic," *International Journal of Engineering and Technology (UAE)*, vol. 7, no. 2, pp. 20–23, 2018.

[10] H. Kaur, V. Mangat, and Nidhi, "A survey of sentiment analysis techniques," *Proceedings of the International Conference on IoT in Social, Mobile, Analytics and Cloud, I-SMAC 2017*, 2017, pp. 921–925.

[11] M. M and S. Mehla, "Sentiment analysis of movie reviews using machine learning classifiers," *International Journal of Computer Applications*, vol. 182, no. 50, pp. 25–28, 2019.

[12] F. Hemmatian and M. K. Sohrabi, "A survey on classification techniques for opinion mining and sentiment analysis," *Artificial Intelligence Review*, vol. 52, no. 3, pp. 1495–1545, 2019.

[13] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," *ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 142–150, 2011.