

Türkçe Metinlerde Duygu Analizi: Derin Öğrenme Yaklaşımlarının ve Ön İşlem Süreçlerinin Model Performansına Etkisi

Yasin GÖRMEZ^{1*}, Halil ARSLAN², Bilal ATAK³

¹ Yönetim Bilişim Sistemleri Bölümü, İktisadi ve İdari Bilimler Fakültesi, Sivas Cumhuriyet Üniversitesi, Sivas, Türkiye

² Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Sivas Cumhuriyet Üniversitesi, Sivas, Türkiye

³ SAP CX Elektronik Ticaret Geliştirme, Detay Danışmanlık, İstanbul, Türkiye

*¹ yasingormez@cumhuriyet.edu.tr, ² harslan@cumhuriyet.edu.tr, ³ bilal.atak@detaysoft.com

(Geliş/Received: 31/01/2024;

Kabul/Accepted: 22/03/2024)

Öz: Günümüzde bilgisayar kullanımının artması ile birlikte insanlar daha fazla veri üretmeye başlamış ve verilere ulaşım kolaylaşmıştır. Bu bağlamda e-ticaret sitelerinde, sosyal medyada ya da diğer elektronik platformlarda çok fazla metin verisi üretilmiştir. Toplanan bu verilerin analiz edilerek anlamlandırılması birçok kurum, kuruluş ya da birey için faydalı bilgiler sağlamaktadır. Bu amaç doğrultusunda duygu analizi günümüzde sıklıkla uygulanmaktadır. Duygu analizi modellerinde derin öğrenme yaklaşımları oldukça yüksek performans göstermekte ve model eğitimi yapılmadan önce metinlere birkaç ön işlem uygulanmaktadır. Bu çalışmada duygu analizi için, evrimsel sinir ağı, Transfomer ve hibrit olmak üzere üç farklı derin öğrenme yaklaşımı önerilmiş ve modeller winvoker ve Beyazperde olmak üzere iki farklı veri seti kullanılarak analiz edilmiştir. Modellerin doğruluğunu artırmak için hiper-parametreleri ve model derinlikleri Bayesian optimizasyon yöntemi kullanılarak optimize edilmiştir. Ön işlem süreçlerinin model performansına etkisini ölçmek için veri setlerine çeşitli ön işlem yapılarak analizler tekrar edilmiştir. Ön işlem uygulanmamış veriler kullanıldığında, winvoker veri seti ile eğitilen modellerde %94,16, Beyazperde veri seti ile eğitilen modellerde ise %86,64 doğruluğa ulaşılmıştır. Ön işlem uygulandığında ise bu başarı oranları, winvoker veri seti ile eğitilen modellerde %94,64, Beyazperde veri seti ile eğitilen modellerde ise %89,08 değerlerine ulaşmıştır. Bu sonuçlar doğrultusunda örnek sayısı daha fazla olan winvoker veri seti için ön işlemlerin etkisinin azaldığı ve doğruluğun daha yüksek olduğu sonucu çıkarılmıştır.

Anahtar kelimeler: Duygu analizi, derin öğrenme, metin ön işlem, evrimsel sinir ağları, transfomer ağları.

Sentiment Analysis in Turkish Texts: The Effect of Deep Learning Approaches and Preprocessing Processes on Model Performance

Abstract: Nowadays, with the increased use of computers, a surge in data production has emerged, making data access more convenient. In this context, a substantial amount of textual data is generated on e-commerce sites, social media, and various electronic platforms. Analyzing and extracting meaningful insights from this amassed data proves valuable for numerous institutions, organizations, and individuals. Sentiment analysis is a commonly employed technique to derive sentiments from textual data, and contemporary sentiment analysis models often leverage the high performance offered by deep learning approaches. Prior to model training, several pre-processing steps are typically applied to the text data. In this study, three distinct deep learning approaches were proposed for sentiment analysis. These models were analyzed on two different datasets: winvoker and Beyazperde. Hyper-parameters and depth of models were optimized using the Bayesian optimization method to enhance the accuracy of model. Additionally, the impact of various pre-processing techniques on model performance were assessed. When non-preprocessed data is utilized, the models trained on the winvoker dataset achieve an accuracy of 94.16%, while those trained on the Beyazperde dataset reach 86.64%. With the application of pre-processing, these accuracies improve to 94.64% for the winvoker dataset and 89.08% for the Beyazperde dataset. Based on these findings, it was concluded that the effect of pre-processing decreased and the accuracy was higher for the winvoker data set with a higher number of samples.

Key words: Sentiment analysis, deep learning, text pre-processing, convolutional neural networks, transformer networks.

1. Giriş

Duygu Analizi (Sentiment Analysis - SA), doğal dil işleme ve metin madenciliği tekniklerini kullanarak bir metindeki öznel bilgileri tanımlamak olarak ifade edilmektedir [1]. SA yardımıyla geliştirilen uygulamalar, bireyler, firmalar ve devletler tarafından kullanılmakta ve kullanımını gün geçtikçe artmaktadır. Özellikle sosyal medya verilerini kullanarak bir konu hakkında fikir çıkarımı yapılması ya da ürün yorumlarının otomatik analiz edilmesi gibi süreçlerde sıklıkla kullanılmaktadır [2]. E-ticaret sistemlerinin kullanımının artmasıyla birlikte ürün yorumlarına SA tekniklerinin uygulanması da giderek artmakta ve gelecekte de popülerliğini koruyacağı ön görülmektedir. İnternet kullanımının yaygınlaşması ve internet sitelerinden veri toplama araçlarının çoğalmasıyla

* Sorumlu yazar: yasingormez@cumhuriyet.edu.tr, Yazarların ORCID Numarası: ¹ 0000-0001-8276-2030, ² 0000-0003-3286-5159, ³ 0009-0002-6157-2849

birlikte toplanan metin verisi sayısı da artmıştır. Bu veriler genellikle kullanıcılar tarafından üretilmekte ve çok fazla gürültü içerebilmektedir [3]. Bunun yanı sıra, kullanıcılar tarafından oluşturulan verilerde dil bilgisi kuralına çok fazla uyulmaması verilerdeki bozulmanın artmasına neden olmaktadır. Verilerde meydana gelen bu bozulmaları en aza indirmek ve SA yöntemlerinin başarı oranlarını artırmak için ise metin normalleştirme yöntemlerine başvurulmaktadır [4].

Toplanan verilerin ait olduğu dilin özellikleri, oluşturulan SA modelin performansını doğrudan etkilemektedir. Bir dilin bükümlü ya da eklemeli dil grubunda yer alması ön işlem süreçlerinin değişmesine neden olabilmektedir. Bir dil grubu için oluşturulmuş olan modelin diğer dil grubunda iyi sonuçlar vermemesi olası bir durum olarak değerlendirilmektedir ve dil gruplarına uygulanan ön işlemlerin zorlukları da değişebilmektedir. Örneğin, bükümlü dil grubunda giren İngilizce için ön işlem süreçleri, eklemeli dil grubunda yer alan Türkçeye göre daha kolay olabilmektedir [5]. Bu kapsamda, Türkçe gibi morfolojik açıdan zengin olan dillerde SA yapmak daha zordur ve bu zorluktan dolayı Türkçe dilinde yapılan SA çalışmaları henüz istenilen seviyelere ulaşamamıştır [6].

SA, doküman, cümle, deyim ve hedef tabanlı olmak üzere dört üst başlıkta incelenebilmektedir [3]. Bu başlıklar altında günümüze kadar SA için teori ve pratikte oldukça fazla çalışma yapılmıştır. Singh ve arkadaşları üç farklı veri setinde Naive Bayes, J-48, BFTree ve OneR yöntemlerini karşılaştırdıkları çalışmada, OneR yönteminin IMDB veri setinde seti üzerinde doğruluk ve F1 metriklerinde diğer yöntemlere üstünlük sağladığı kanısına varmışlardır [7]. Jagdale ve arkadaşları Amazon e-ticaret platformundan toplanan elektronik ürün yorumları veri setinde Naive Bayes ve Destek vektör makinaları yöntemlerinde sırasıyla %98,17 ve %93,54 doğruluk elde etmişlerdir [8]. Jain ve Dandannavar Apple, ICICI ve BSNL olmak üzere üç farklı veri seti üzerinde karar ağacı ve Multinomial Naive Bayes yöntemleri eğiterek %76 ile %100 arasında doğruluk elde etmişlerdir [9]. Gupta ve diğerleri film yorumları veri seti üzerine önce ön işlem uygulamış daha sonra sekiz farklı makine öğrenmesi modeli eğitmişlerdir. Yaptıkları analiz sonucunda en düşük doğruluğu %66 ile Naive Bayes algoritmasının, en yüksek doğruluğu ise %90 ile Maksimum Entropi ve topluluk yöntemlerinin elde ettiğini göstermişlerdir [10]. Shamantha ve diğerleri Twitter üzerinden topladıkları verileri kullanarak Naive Bayes, Destek Vektör Makinaları ve Rastgele Orman algoritmaları modellerini eğitmiş ve %60 ile %70 arasında değişen oranlarda doğruluk elde etmişlerdir. Bunun yanı sıra toplanan veri sayısı arttıkça doğrulukta da iyileşme olduğunu belirtmişlerdir [11]. Naresh ve Venkata Krishna önerdikleri karar ağacıyla sıralı minimum optimizasyon yöntemiyle Twitter verisi üzerinde %89,47 doğruluk elde etmişlerdir [12].

Geleneksel makine öğrenmesi yöntemlerinin yanı sıra derin öğrenme yaklaşımları kullanılarak da SA için oldukça fazla çalışma yapılmıştır. Araque ve arkadaşları el ile çıkarılan öznitelikleri, otomatik çıkarılan öznitelikler ile birleştirdikleri çalışmada önerdikleri derin öğrenme modeliyle PL04 veri setinde %94,49 F1 skoruna ulaşmışlardır [13]. Jianqiang ve arkadaşları kelime gömme, n-gram ve kelime duyarlılığı polarite puanı özniteliklerini birleştirerek eğittikleri derin evrimsel sinir ağı (Convolutional Neural Network - CNN) modelini kullanarak beş farklı Twitter veri setinde ortalama %85,63 doğruluk elde etmişlerdir [14]. Xu ve diğerleri önerdikleri uzun kısa vadeli ağlar (Long Short-Term Memories - LSTM) tabanlı derin öğrenme modeliyle otel verisi üzerinde kesinlik, duyarlılık ve F1 skor metriklerinde sırası ile %91,54, %92,82 ve %92,18 değerlerini elde etmişlerdir [15]. Zhao ve arkadaşları önerdikleri CNN ve LSTM tabanlı derin öğrenme modellerinde ürün yorum veri setinde sırası ile %87,70 ve %87,90 doğruluk elde etmişlerdir [16]. Basiri ve arkadaşları dikkat temelli çift yönlü CNN ve LSTM kullanarak geliştirdikleri model ile App, CD, Movie, Twitter, Electronics ve Sentiment140 veri setlerinde sırası ile %92,18, %88,70, %90,55, %92,75, %90,65 ve %81,82 doğruluk elde etmişlerdir [17]. Mishev ve diğerleri finansal verilerde duygu analizi için önerdikleri Transformer tabanlı modellerle SNLI veri setinde %94,79 doğruluğa ulaşmışlardır [18]. Naseem ve arkadaşları önerdikleri Transformer tabanlı model sayesinde havayolu firmalarına yapılan yorumları %96,20'ye varan başarı oranları ile sınıflandırmışlardır [19]. Zhang ve diğerleri önerdikleri geniş çok görevli Transformer ağları tabanlı derin model kullanarak Twitter veri setinde %77,80 F1 skor ve Stanford Sentiment Treebank veri setinde %94,00 doğruluk elde etmişlerdir [20]. Myagmar ve arkadaşları çift yönlü Transformer ağlarını kullanarak önerdikleri dil modeli sayesinde alanlar arası duygu analizinde başarı oranını Amazon veri seti için 120 kat daha az örnek kullanarak artırmışlardır [21]. Potamias ve diğerleri, önceden eğitilmiş derin modeli, tekrarlayan CNN ile birleştirerek SemEval-2018, Reddit Politics, Sarcastic Rillof's ve SemEval-2015 veri setlerinde sırasıyla %82, %79, %91 ve %81 doğruluk elde etmişlerdir [22]. Murfi ve arkadaşları Endonezya dilinde duygu analizi için önerdikleri BERT tabanlı derin modelde %87,68 doğruluğa ulaşmışlardır [23]. Liu ve arkadaşları, BERT yöntemini kullanarak geliştirdikleri çok modlu duygu analizi modeliyle IEMOCAP veri setinde %86,14 doğruluğa ulaşmışlardır [24].

Model eğitmek için kullanılan veri setinin oluşturulduğu dil SA için önemli bir parametredir. SA yapılmak istenen dile göre farklı süreç ya da modellerin kullanılması gerekebilmektedir. Bu kapsamda literatürde Türkçe dili özelinde yapılmış birçok çalışmada farklı süreçlerin yürütüldüğü görülmektedir. Dehkharghani ve arkadaşları, polarite sözlüğü yaklaşımı kullanarak öznitelik çıkarımı yaptıkları çalışmada sıralı minimum optimizasyon algoritması, yapay sinir ağları ve lojistik regresyon modellerinde sırasıyla %82,89, %83,32 ve %83,13 doğruluk

elde etmişlerdir [25]. Salur ve Aydın, Türkçe Twitter veri seti üzerine çeşitli ön işlem yöntemleri uygulamış ve farklı yöntemler kullanarak öznelik çıkarımı yapmışlardır. Önerdikleri farklı derin öğrenme modellerini eğiterek %75,71'e varan başarı oranları elde etmişlerdir [26]. Ayvaz ve diğerleri, yaptıkları çalışmada Twitter üzerinden bir yarışma programı hakkında yazılan yorumları toplayarak veri seti oluşturmuşlardır. Oluşturdukları veri setine ön işlem uygulayarak duygu analizi yapmışlar ve kullanıcı duygularını mevsimlerinde etkilediği kanaatine varmışlardır [27]. Çetin ve Eryiğit hedef tabanlı duygu analizi için önerdikleri koşullu rastgele alanlar modelini, kelime bölütleyici, Türkçe karakter düzeltici, morfolojik çözümleyici, morfolojik belirsizlik giderici ve bağlılık ayırıştırıcısı gibi ön işlem yöntemlerini uyguladıkları veri seti ile eğiterek %66,7 F1 skoruna ulaşmışlardır [28]. Onan, yapmış olduğu çalışmada CNN katmanını kullanarak geliştirdiği derin öğrenme modellerini farklı öznelik çıkarma yöntemleri kullanarak analiz etmiş ve en yüksek doğruluğu %92,53 ile CNN tabanlı Word2vec modeli ile elde etmiştir [29]. Tuzcu çalışmasında, çevrimiçi kitap satış sitesinde yapılan yorumlardan oluşan veri setinde eğittiği çok katmanlı algılayıcı, Naive Bayes, destek vektör makinaları ve lojistik regresyon modellerinde sırasıyla %89, %77,57, %80,93 ve %84,07 doğruluk elde etmiştir [30].

Literatür incelendiğinde yapılan SA çalışmalarında, analizlerin yapıldığı dile bağlı olarak çok fazla ön işlem aşaması olduğu görülmektedir. Türkçe gibi eklemeli grupta yer alan bir dilde ise, ön işlem için geliştirilen modellerde de hata oranları yüksek olabilmektedir. Bunun yanı sıra birçok ek analizin yapılması gereksinimi ortaya çıkmakta bu nedenle zaman ve kaynak kayıpları yaşanabilmektedir. Literatür araştırmasından çıkan diğer bir sonuç ise, derin öğrenme yaklaşımlarının geleneksel modellere göre daha başarılı sonuçlar elde ettiğidir. Bu çalışmada Türkçe SA için üç farklı özgün derin öğrenme modeli Transformer ve CNN katmanları kullanılarak geliştirilmiştir. Geliştirilen derin öğrenme modellerini eğitmek için winvoker [31] ve Beyazperde [32] olmak üzere iki farklı veri seti kullanılmıştır. Geliştirilecek bir SA modeli, özellikle e-ticaret sitelerinde ürünlerin karşılaştırılması, ürünlerin eksiklerinin anlaşılması ve ürün hakkında genel algıların tespit edilebilmesi için kullanılabilir. Bu kapsamda önerilen modelin e-ticaret siteleri ile entegre bir şekilde çalışarak ürünler hakkındaki yorumların genel anlamını çıkarması istenmektedir. Bu nedenle yorum içeren iki farklı veri seti tercih edilmiştir. İlk veri setinde sadece olumlu ve olumsuz ayrımı yapılırken, ikinci veri setinde olumlu, olumsuz ve nötr ayrımı yapılabilmektedir. Bu sayede farklı sınıf sayılarında SA modellerinin performansının da ölçülmesi amaçlanmaktadır. Bu veri setlerine çeşitli ön işlem aşamaları uygulanmış ve yeni veri setleri oluşturulmuştur. Her bir makine öğrenmesi modeli hem ön işlem uygulanmış hem de uygulanmamış veri setleri kullanılarak eğitilmiştir. Önerilen modellerin çeşitli hiper-parametreleri ve derinlikleri Bayesian optimizasyon yöntemi kullanılarak optimize edilmiştir. Bu bağlamda, çalışmanın ilk amacı Transformer ve CNN tabanlı derin öğrenme modellerinin Türkçe SA analizindeki performanslarının ölçülmesidir. Çalışmanın ikinci amacı ise, ön işlem süreçlerinin derin öğrenme modellerinin performansına etkisinin hesaplanmasıdır. Çalışmaya ait derin öğrenme modellerinin geliştirilecek olması ve ön işlem süreçlerinin Türkçe SA için karşılaştırılacak olması ise çalışmanın özgün yanları olarak değerlendirilmektedir.

2. Veri Seti ve Yöntemler

2.1. Veri Seti

Çalışmamızda winvoker [31] ve Beyazperde [32] olmak üzere iki farklı veri seti kullanılarak analizler yapılmıştır. Winvoker veri seti, humir [33], e-ticaret sitesi yorumları ve Wikipedia gibi farklı kaynaklardan derlenmiş 262.166 pozitif, 56.561 negatif ve 170.917 nötr sınıfına ait olmak üzere toplam 489.644 örnekten oluşmaktadır. Beyazperde veri seti ise, 5.331 pozitif ve 5.331 negatif sınıfa ait olmak üzere toplam 10.662 film yorumundan oluşmaktadır. Bu veri setleri sayesinde herhangi bir ürün hakkında ya da bir olaya yapılan yorumların olumluluk ve olumsuzluk durumlarını ölçebilen sistemlerin eğitilmesi mümkün olmaktadır. Çalışmamızın amaçlarından bir diğeri olan, e-ticaret sistemlerinde yapılan yorumların analizi için ilgili veri setlerinin uygun olduğu kanaatine varılmıştır. Veri setleri hakkındaki detaylı bilgilere verilen referanslardan ulaşılabilmektedir.

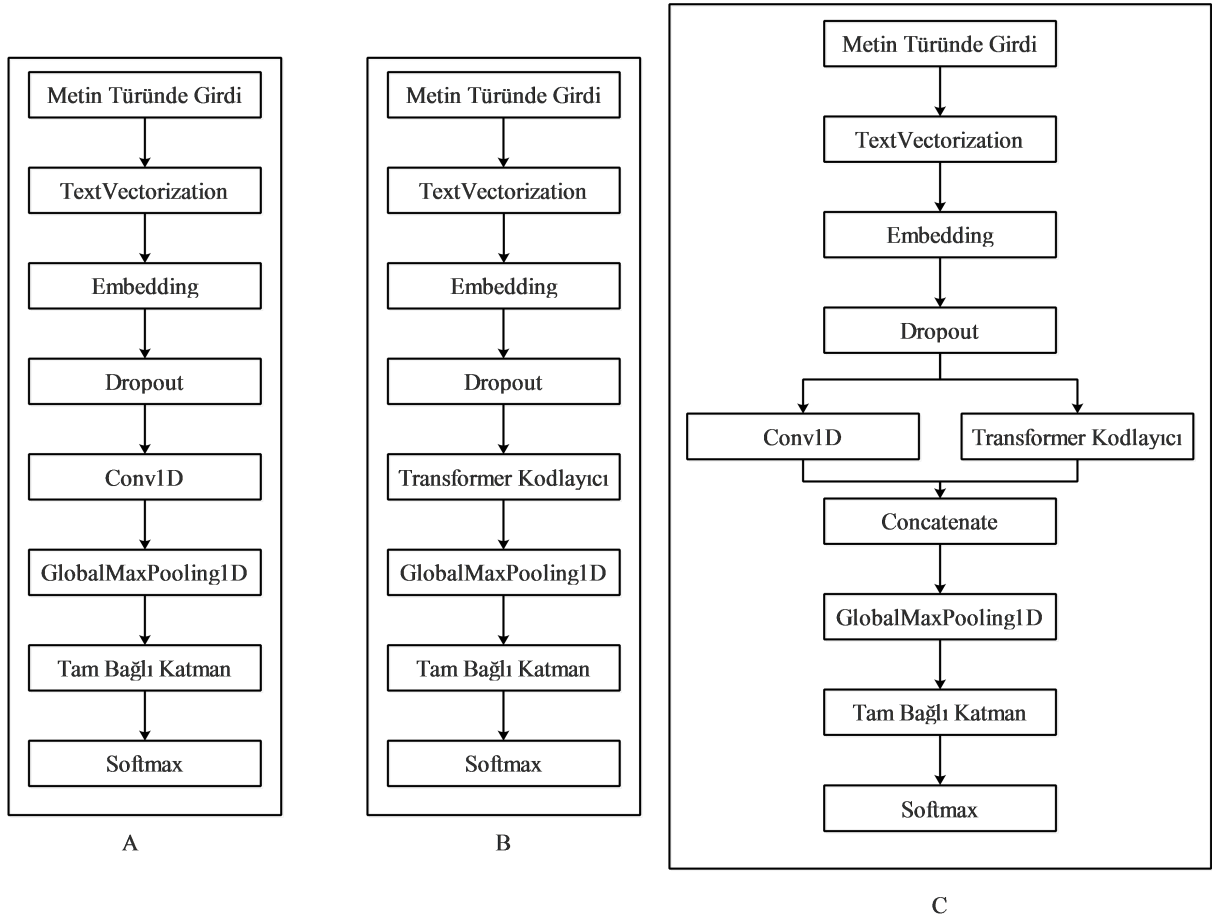
2.2. Metin Ön İşleme

Çalışmanın amaçlarından biri olan, ön işlem tekniklerinin derin öğrenme modellerinin performansına etkisini ölçmek için kullanılan veri setlerine ön işlem teknikleri uygulanarak temizleme işlemi yapılmıştır. Bu kapsamda ilk olarak, veri setindeki örneklerde bulunan html işaretleri temizlenmiştir. Daha sonra bu örnekler noktalama işaretlerinden ve etkisiz kelimelerden (stop words) arındırılmıştır. Bu aşamada noktalama işaretlerini tanımlamak için python dilinde bulunan string kütüphanesi kullanılırken [34], Türkçe etkisiz kelimeleri tanımlamak için python dilinde bulunan nltk kütüphanesi kullanılmıştır [35]. Ön işlemin üçüncü aşamasında ise temizlenmiş örneklere metin normalleştirme uygulanmıştır. Kullanılan veri setlerindeki içerikler birçok farklı kullanıcıdan elde edilmiştir ve bu kullanıcıların genellikle yazım kurallarına dikkat etmediği görülmüştür. Kelime yanlış yazımı, olmayan

kelime kullanımı ya da eksik harf kullanma gibi durumlarla sıklıkla karşılaşmaktadır. Metin normalleştirme sayesinde kullanıcılar tarafından hatalı bir şekilde oluşturulan içeriklerin düzenlenmesi amaçlanmaktadır. Bu çalışmada, metin normalleştirme için python dilinde var olan zemberek kütüphanesi kullanılmıştır [36]. Ön işlemin son aşamasında ise veri setinde üç kereden daha az geçen kelimeler elenerek metin boyutları düşürülmüştür.

2.3. Derin Öğrenme Modelleri

Çalışmamızda duygu analizi için üç farklı derin öğrenme modeli CNN ve Transformer katmanları kullanılarak geliştirilmiştir. Kullanılan bu katmanlar, TextVectorization, Embedding ve tam bağlı katmanlarla da desteklenmiştir. TextVectorization, Embedding ve tam bağlı katmanların yanı sıra, bu modellerin ilki sadece CNN kullanarak, ikincisi sadece Transformer kullanarak, üçüncüsü ise CNN ve Transformer katmanlarını hibrit bir şekilde kullanarak geliştirilmiştir. Her bir model için mimari yapı şekil 1’de gösterilmektedir.



Şekil 1. Önerilen derin öğrenme modellerinin mimari yapıları. A: CNN model, B: Transformer Model, C: Hibrit Model

Şekil 1’de gösterilen mimari yapılar, CNN ve Transformer katmanlarının bir kez kullanıldığı modeller gösterilmektedir. Çalışmada eğitilen modellerde ise CNN ve Transformer katman sayıları her bir veri setinde ayrı ayrı optimize edilmiştir. Modellerde kullanılan katman sayıları deney sonuçları kısmında sunulmaktadır. Önerilen modeller python dilinde var olan keras kütüphanesi kullanılarak geliştirilmiştir [37]. Bu modellerin TextVectorization katmanında *max_tokens* parametresi 20.000, *output_sequence_length* parametresi ise 500 olarak ayarlanmıştır. Sınıflandırma katmanı hariç, tüm katmanların aktivasyon fonksiyonu (activation function) *relu*, sınıflama katmanının aktivasyon katmanı *softmax*, kayıp fonksiyonu (loss function) *categorical_crossentropy*, optimizer ise *Adam* olarak ayarlanmıştır. Transformer kodlayıcı ise iki ana modülden oluşmaktadır. İlk modül, MultiHeadAttention, Dropout ve LayerNormalization katmanlarının seri olarak birbirine bağlanması ile oluşturulmuştur. Daha sonra bu modülün çıktısı, ikinci modülün girdisi olarak kullanılmak üzere girdi ile birleştirilmiştir. İkinci modülde ise sırası ile seri şekilde bağlı CNN, Dropout, CNN ve

LayerNormalization katmanları bulunmaktadır. Bu modülde bulunan CNN katmanlarının çekirdek genişliği (Kernel Size) 5 olarak ayarlanmıştır, birim sayıları ise optimize edilmiştir. Modellerdeki katmanlara özgü hiper-parametreler ise optimize edilmiştir. Optimizasyon süreçlerine ait detaylı bilgi deney sonuçları kısmında verilmiştir.

3. Deneysel Sonuçları

Deneylerin ilk aşamasında SA için kullanılacak veri setleri bölütlenerek eğitim, test ve validasyon veri setleri oluşturulmuştur. Veri setlerinin indirildiği kaynaklarda, her iki veri setinin de eğitim ve test olmak üzere hali hazırda bölütlendiği görülmektedir [31], [32]. Bu duruma ek olarak çalışmada, validasyon veri seti, indirilen eğitim veri seti içerisinde rastgele %20 örnek seçilerek oluşturulmuştur. Kalanlar ise çalışmada kullanılacak eğitim veri setini oluşturmak üzere kullanılmıştır. Eğitim veri seti modelleri eğitmek, validasyon veri seti hiper-parametre optimizasyonu aşamasında ve gerçeğe çağırma fonksiyonlarında performans hesaplamak, test veri seti ise eğitilmiş modellerin performanslarını hesaplamak için kullanılmıştır. Her bir veri setinin sınıflara ait örnek sayıları tablo 1’de gösterilmektedir.

Tablo 1. Bölütleme aşamasından sonra oluşturulan veri setlerinde sınıflara ait örnek sayıları.

Veri Seti Adı	Veri Seti Türü	Olumsuz	Nötr	Olumlu	Toplam
winvoker	Eğitim	40 521	123 243	188 780	352 544
	Validasyon	10 384	30 582	47 169	88 135
	Test	5 656	17 092	26 217	48 965
BeyazPerde	Eğitim	3 208	0	3 189	6 397
	Validasyon	790	0	809	1 599
	Test	1 333	0	1 333	2 666

Veri seti bölütleme aşamasından sonra ön işlem uygulanmamış veri setleri kullanılarak hiper-parametre optimizasyonu yapılmıştır. Bu aşamada derin öğrenme modellerinde, ızgara ve rastgele arama gibi diğer hiper-parametre optimizasyon yöntemlerine göre üstünlüğü kanıtlanmış olan Bayesian optimizasyon yöntemi kullanılmıştır [38], [39], [40]. Bayesian optimizasyon yöntemini geliştirmek için python dilinde var olan skopt kütüphanesi kullanılmıştır [41]. Bu kütüphanede var olan *gp_minimize* fonksiyonu, Gauss süreç regresyonuna dayalı yöntemle optimizasyon sürecini yönetmektedir. Bu yöntemin edinim fonksiyonu (Acquisition Functions – *acq_func*) parametresi *EI* (beklenen iyileşme), çağırılma sayısı (*n_calls*) parametresi ise 25 olarak ayarlanmıştır. *gp_minimize* fonksiyonu bir hiper-parametre uzayı almakta ve bu uzay içerisindeki en uygun hiper-parametre değerlerini istenen performans skoruna göre hesaplamaktadır. Bu çalışmada performans skoru $\frac{\text{sensitivity} + \text{specificity}}{2}$ olarak belirlenmiştir. Ön işlem uygulanmamış veri setleri optimize edilen hiper-parametre türleri, hiper-parametre uzayları ve en uygun hiper-parametre değerleri tablo 2’de gösterilmektedir.

Tablo 2’de yer alan öğrenme adımı, her yinelemedeki adım boyutunu (learning rate); CNN birim sayısı, CNN katmanındaki filtre sayısını (filters); CNN katman derinliği, kullanılan CNN modül sayısını; CNN katmanı çekirdek genişliği, evrişim penceresinin boyutunu (kernel_size); tam bağlı katman birim sayısı, sınıflama katmanından önceki tam bağlı katman nöron sayısını (number of units); devir sayısı, tüm veri üzerinde yapılan bir yinelemeyi (epoch); transformer katman derinliği, kullanılan transformer modül sayısını; dikkat katman sayısı, MultiHeadAttention katmanı için birim sayısını (num_heads); dikkat katman genişliği ise MultiHeadAttention katmanındaki her birimin boyutunu (key_dim) temsil etmektedir. Hibrit modelde iki adet optimum CNN birim sayısı verilmesinin nedeni, hem model içindeki CNN katmanlarının birim sayısının hem de Transformer modülü içerisinde bulunan CNN katmanlarının birim sayısının optimize edilmiş olmasıdır. Bu bağlamda ilk değer, CNN modülünde birim sayısını, ikinci değer ise Transformer modülünde bulunan CNN modüllerinin birim sayısını göstermektedir.

Hiper-parametre optimizasyonu tamamlandıktan sonra her bir model, ön işlem yapılmamış eğitim veri seti kullanılarak eğitilmiş ve model performansları test veri setinde hesaplanmıştır. Bu aşamada modele öğrenme adımı (*lr_callback*) ve erken durdurma (*early_stopping_callback*) olmak üzere iki adet geri çağırma fonksiyonu da eklenmiştir. *lr_callback*, validasyon veri setinde elde edilen kayıp değerinde iki devir boyunca iyileşme olmaz ise öğrenme adımının iki ile bölünmesini sağlarken; *early_stopping_callback*, validasyon veri setinde elde edilen kayıp değerinde altı devir boyunca iyileşme olmaz ise eğitimin durmasını sağlamaktadır. Tablo 3’te, her bir model için test veri seti üzerinde elde edilmiş doğruluk (accuracy), kesinlik (precision), f1-skor, ve duyarlılık (recall)

değerleri gösterilmektedir. Kullanılan metriklerin hesaplama formülleri için sklearn kütüphanesinde metrics kütüphanesi kullanılmıştır [42].

Tablo 2. Ön işlem uygulanmamış veri setleri için hiper-parametre optimizasyon detayları.

Hiper-Parametre Adı	Hiper-Parametre Türü	Uzay Aralığı	Model Adı	Veri Seti Adı	Optimum Değer
Öğrenme adımı	Gerçek değer	En büyük: 10^{-1} , En küçük: 10^{-4}	CNN	winvoker	0.00057122
				Beyazperde	0.00027370
			Transformer	winvoker	0.00181879
				Beyazperde	0.06358986
			Hibrit	winvoker	0.01153229
				Beyazperde	0.00154756
CNN birim sayısı	Tamsayı	En büyük: 16 En küçük: 256	CNN	winvoker	90
				Beyazperde	256
			Transformer	winvoker	256
				Beyazperde	71
			Hibrit	winvoker	16, 238
				Beyazperde	82, 111
CNN katman derinliği	Tamsayı	En büyük: 1 En küçük: 10	CNN	winvoker	2
				Beyazperde	2
			Hibrit	winvoker	9
				Beyazperde	5
CNN katmanı çekirdek genişliği	Tamsayı	En büyük: 3 En küçük: 15	CNN	winvoker	13
				Beyazperde	14
			Hibrit	winvoker	11
				Beyazperde	9
Tam bağlı katman birim sayısı	Tamsayı	En büyük: 100 En küçük: 1500	CNN	winvoker	1219
				Beyazperde	335
			Transformer	winvoker	100
				Beyazperde	1168
			Hibrit	winvoker	100
				Beyazperde	426
Devir sayısı	Tamsayı	En büyük: 4 En küçük: 100	CNN	winvoker	96
				Beyazperde	34
			Transformer	winvoker	89
				Beyazperde	4
			Hibrit	winvoker	100
				Beyazperde	25
Transformer katman derinliği	Tamsayı	En büyük: 1 En küçük: 10	Transformer	winvoker	1
				Beyazperde	1
			Hibrit	winvoker	1
				Beyazperde	2
Dikkat katmanı sayısı	Tamsayı	En büyük: 1 En küçük: 10	Transformer	winvoker	10
				Beyazperde	1
			Hibrit	winvoker	1
				Beyazperde	1
Dikkat katmanı genişliği	Tamsayı	En büyük: 16 En küçük: 256	Transformer	winvoker	256
				Beyazperde	251
			Hibrit	winvoker	192
				Beyazperde	129

Tablo 3. Ön işlem uygulanmamış veri setleri ile eğitilen modellerin performans skorları

Model Adı	Veri Seti	Doğruluk (%)	Kesinlik (%)	Duyarlılık (%)	F1-skor (%)
CNN	winvoker	94,15	92,46	88,66	90,28
	Beyazperde	86,64	87,72	85,52	86,45
Transformer	winvoker	93,98	92,77	88,25	90,11
	Beyazperde	83,98	86,24	80,87	83,46
Hibrit	winvoker	94,16	91,75	89,58	90,58
	Beyazperde	86,23	89,98	81,54	85,55

Tablo 3'te gösterilen sonuçlar incelendiğinde, winvoker veri setinde hesaplanan performans skorlarının, Beyazperde veri setinde hesaplanan performans skorlarına göre daha yüksek olduğu görülmektedir. Bu durumun winvoker veri setinde daha fazla örnek olmasıyla ilgili olduğu düşünülmektedir. Tabloda koyu olan değerler iki veri seti içinde her bir metrik türünde en yüksek skoru elde eden modeli belirtmektedir. Winvoker veri seti hibrit model ile kesinlik hariç diğer metriklerde en yüksek skorları elde ederken, Beyazperde veri seti kesinlik hariç diğer metriklerde CNN modeli ile en yüksek skorları elde etmiştir. Ön işlem uygulanmamış veri setleri ile yapılan analizler tamamlandıktan sonra, metin ön işlem bölümünde anlatılan aşamalar veri setlerine uygulanarak yeni veri setleri oluşturulmuştur. Daha sonra ön işlem uygulanmamış veri setlerindeki ayarların ve uzay aralıklarının aynı kullanılarak modellere hiper-parametre optimizasyonu yapılmıştır. Ön işlem uygulanmış veri setleri için en uygun hiper-parametre değerleri tablo 4'te gösterilmektedir.

Tablo 4'te de CNN birim sayısı için hibrit modellerde iki tane olma sebebi hem CNN modülünde hem de transformer modülünde var olan CNN katmanlarının birim sayılarının optimize edilmesidir. Bu bağlamda ilk değer CNN katmanındaki, ikinci değer ise Transformer modülünde bulunan CNN katmanının birim sayısını vermektedir. Hiper-parametre optimizasyonu tamamlandıktan sonra her bir model, ön işlem yapılmış eğitim veri seti kullanılarak eğitilmiş ve model performansları test veri setinde hesaplanmıştır. Model eğitimleri aşamasında hiper-parametreler hariç ön işlem yapılmamış veri setindeki modeller ile aynı ayarlar kullanılmıştır. Tablo 5'te, her bir model için test veri seti üzerinde elde edilmiş doğruluk, kesinlik, f1-skor, ve duyarlılık değerleri gösterilmektedir.

Tablo 5'te gösterilen sonuçlar incelendiğinde, ön işlem uygulanmamış veri seti sonuçlarına benzer şekilde winvoker veri setinde hesaplanan performans skorlarının, Beyazperde veri setinde hesaplanan performans skorlarına göre daha yüksek olduğu görülmektedir. Tabloda koyu olan değerler iki veri seti içinde her bir metrik türünde en yüksek skoru elde eden modeli belirtmektedir. Winvoker veri seti için hibrit model ile kesinlik hariç diğer metriklerde en yüksek skorlar elde edilirken, Beyazperde veri seti için CNN modeli ile doğruluk ve f1-skor metriklerinde en yüksek skorlar elde edilmiştir. Bu yapısı ile de sonuçlar ön işlem uygulanmamış veri seti sonuçlarına benzemektedir.

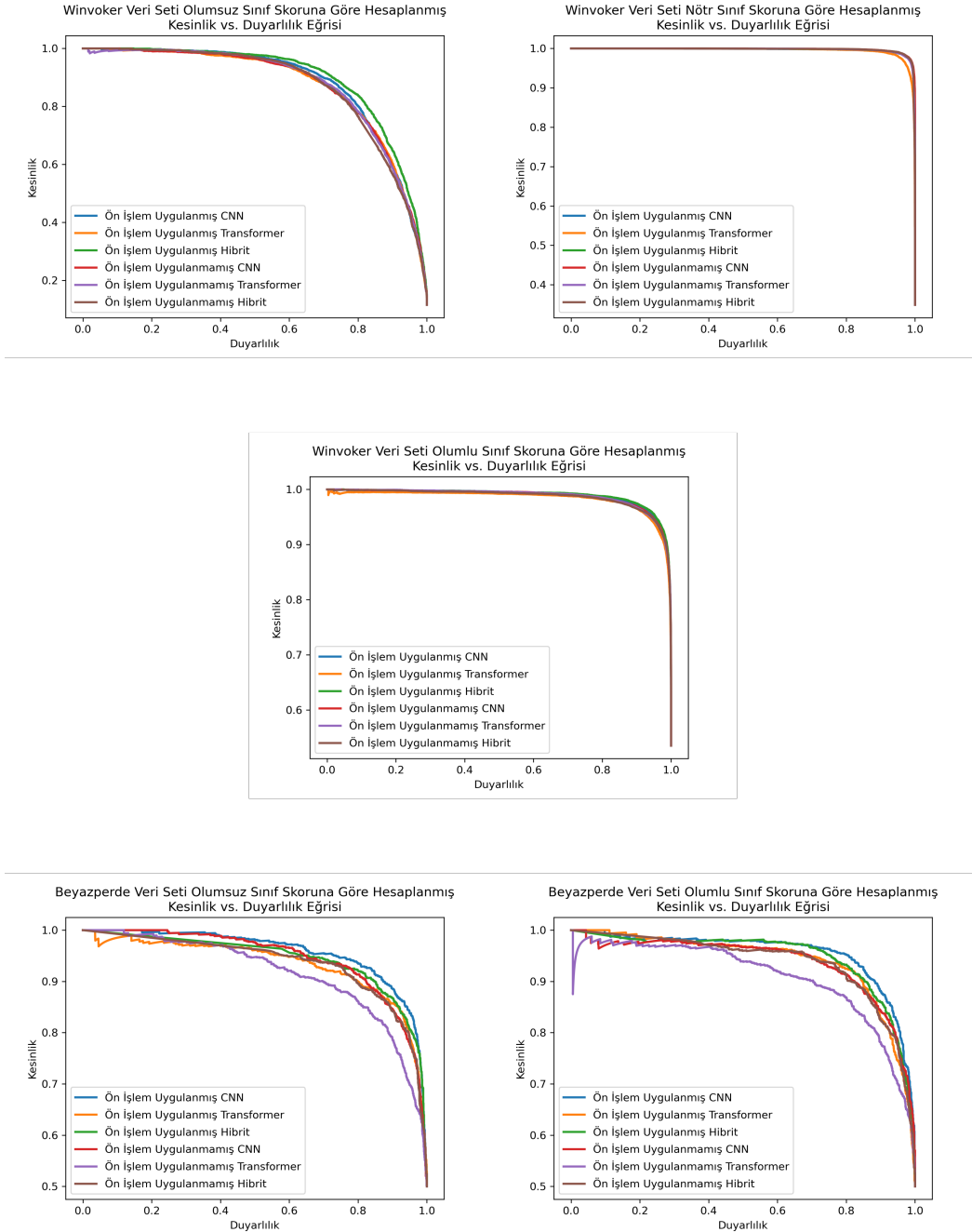
Ön işlem uygulanmamış ve ön işlem uygulanmış veri setleri ile eğitilen modeller kullanılarak hesaplanan başarı oranları karşılaştırıldığında winvoker veri seti için CNN ve hibrit modeller ön işlem uygulanmış veri seti ile eğitildiğinde daha iyi sonuç verirken, Transformer modeli ön işlem uygulanmamış veri seti ile eğitildiğinde daha iyi sonuç vermiştir. Winvoker veri seti için aradaki doğruluk farkları ise %0,27 ile %0,48 arasında değişiklik göstermektedir. Beyazperde veri setinde ise tüm modeller ön işlem uygulanmış veri seti ile eğitildiğinde daha yüksek başarı oranları elde etmiştir. Beyazperde veri setindeki doğruluk farkları ise %1,69 ile %2,65 arasında değişiklik göstermektedir. Model performanslarının daha sağlıklı değerlendirilebilmesi için farklı metrikler kullanılarak karşılaştırılma yapılması büyük önem arz etmektedir. Bu bağlamda, veri setlerindeki her bir sınıf türü için şekil 2'de gösterilen kesinlik vs. duyarlılık eğrisi çizilerek, sınıf bazında model performansları karşılaştırılmıştır.

Tablo 4. Ön işlem uygulanmış veri setleri için en uygun hiper-parametre değerleri.

Hiper-Parametre Adı	Model Adı	Veri Seti Adı	Optimum Değer
Öğrenme adımı	CNN	winvoker	0.00096460
		Beyazperde	0.0001
	Transformer	winvoker	0.0001
		Beyazperde	0.00606042
	Hibrit	winvoker	0.00087595
		Beyazperde	0.00036356
CNN birim sayısı	CNN	winvoker	232
		Beyazperde	183
	Transformer	winvoker	16
		Beyazperde	246
	Hibrit	winvoker	256, 256
		Beyazperde	255, 53
CNN katman derinliği	CNN	winvoker	1
		Beyazperde	1
	Hibrit	winvoker	10
		Beyazperde	1
CNN katmanı çekirdek genişliği	CNN	winvoker	11
		Beyazperde	15
	Hibrit	winvoker	15
		Beyazperde	10
Tam bağlı katman birim sayısı	CNN	winvoker	334
		Beyazperde	1500
	Transformer	winvoker	144
		Beyazperde	686
	Hibrit	winvoker	1277
		Beyazperde	1500
Devir sayısı	CNN	winvoker	27
		Beyazperde	100
	Transformer	winvoker	4
		Beyazperde	10
	Hibrit	winvoker	4
		Beyazperde	81
Transformer katman derinliği	Transformer	winvoker	2
		Beyazperde	1
	Hibrit	winvoker	1
		Beyazperde	9
Dikkat katmanı sayısı	Transformer	winvoker	10
		Beyazperde	4
	Hibrit	winvoker	1
		Beyazperde	9
Dikkat katmanı genişliği	Transformer	winvoker	183
		Beyazperde	29
	Hibrit	winvoker	16
		Beyazperde	202

Tablo 5. Ön işlem uygulanmış veri setleri ile eğitilen modellerin performans skorları

Model Adı	Veri Seti	Doğruluk (%)	Kesinlik (%)	Duyarlılık (%)	F1-skor (%)
CNN	winvoker	94,42	92,75	89,58	90,99
	Beyazperde	89,08	88,82	89,42	89,12
Transformer	winvoker	93,36	92,65	86,84	89,19
	Beyazperde	86,63	82,71	90,09	86,24
Hibrit	winvoker	94,64	92,63	90,43	91,44
	Beyazperde	87,92	90,73	84,47	87,49

**Şekil 2.** Winvoker ve Beyazperde veri setleri ile eğitilen modellerde her bir sınıf baz alınarak çizilmiş kesinlik vs. duyarlılık eğrisi

Şekil 2 ile verilen eğriler incelendiğinde, modeller arasındaki fark, başarı performans metriklerine benzer şekilde sınıf bazında kesinlik vs. duyarlılık eğrilerinde de Beyazperde veri seti için belirgindir. Winvoker veri seti için ise modeller arasında kesinlik vs. duyarlılık eğri farkı ayırt edilemeyecek kadar yakın olmaktadır. Özellikle nötr ve olumlu sınıf için bu farkların çok daha az olduğu görülmektedir. Bunun yanı sıra winvoker veri seti ile eğitilmiş model eğrilerinin, Beyazperde veri seti ile eğitilmiş model eğrilerine göre daha düzgün, yani ideal duruma daha yakın olduğu görülmektedir. Veri setleri arasında gözlemlenen bu farklılıkların örnek sayısı ile doğrudan ilişkili olduğu düşünülmektedir.

4. Tartışma ve Sonuçlar

Bu çalışmada duygu analizi için özgün derin öğrenme modelleri geliştirilerek modeller, winvoker ve Beyazperde olmak üzere iki farklı veri seti kullanılarak analiz edilmiştir. Modellerin hiper-parametreleri Bayesian optimizasyon yöntemi kullanılarak optimize edilmiştir. Veri setlerine etkisiz kelime temizleme, noktalama temizleme, metin normalleştirme ve yaygın olmayan kelimelerin elenmesi ön işlemleri uygulanarak temiz veri setleri elde edilmiş ve analizler tekrar edilmiştir. Bu kapsamda üç farklı derin öğrenme modeli için iki veri setinde de hem ön işlem uygulanmış hem de uygulanmış örnekler için analiz sonuçları elde edilmiştir. Sonuçlar değerlendirildiğinde winvoker veri seti ile eğitilen modellerin, Beyazperde veri seti ile eğitilen modellere göre daha iyi performans skorları elde ettiği görülmektedir. Bu durumunun en önemli sebebinin veri setindeki örnek sayısı olduğu öngörülmekte, bu bağlamda örnek sayısı arttıkça SA derin öğrenmesi modellerinin performanslarının da iyileştiği değerlendirilmektedir.

Her bir model ön işlem uygulanmış ya da uygulanmamış veri ile eğitilme duruma göre değerlendirildiğinde ise, Beyazperde veri setine ön işlem uygulanarak eğitilen modellerin performansında, ön işlem uygulamadan eğitilen modellerin performansına göre dikkate değer iyileşmeler olmaktadır. Winvoker veri setinde ise, bu farklılığın bazı durumlarda çok az olduğu, bazı durumlarda ise ön işlem uygulanmayan veri seti kullanılarak eğitilen modellerin daha yüksek performans skoru elde ettiği görülmektedir. Sınıf bazlı kesinlik vs. duyarlılık eğrisi sonuçlarına bakıldığında da iki veri seti arasındaki bu fark net bir şekilde görülmektedir. Winvoker veri seti ile eğitilen modellerin kesinlik vs. duyarlılık eğrilerinin ideal duruma daha yakın olduğu ve model eğrilerinin iç içe girdiği görülürken, Beyazperde veri seti ile eğitilen modellerin kesinlik vs. duyarlılık eğrilerinin ideal durumdan uzaklaştığı ve modeller arası farkın arttığı görülmektedir. Tüm bu sonuçlar dikkate alındığında çalışma kapsamında yapılan analizlerden elde edilen diğer önemli sonuç ise, veri setindeki örnek sayısının artmasıyla birlikte duygu analizinde ön işlemin performansına etkisinin azaldığı hatta bazı durumlarda ortadan kalktığı ya da olumsuz etki ettiği görülmektedir. Bu sonuçlar doğrultusunda, daha yüksek boyutlu veri setleri kullanılarak eğitilen SA modellerinin ön işleme gerek olmadan yüksek performans skorlarına ulaşacağı değerlendirilmektedir.

Yapmış olduğumuz araştırmalar doğrultusunda winvoker veri seti ile yapılmış bir çalışmaya literatürde rastlanmamıştır ancak Beyazperde veri seti ile yapılmış olan birkaç çalışma bulunmaktadır. Bu kapsamda önerilen model ile elde edilen sonuçlar, Yıldırım ve diğerleri tarafından önerilen LR tabanlı model (Model 1) [43], Alqaraleh tarafından önerilen Adabost tabanlı model (Model 2) [44] ve Açıklalın ve diğerleri tarafından önerilen BERT tabanlı model (Model 3) [45] ile karşılaştırılmıştır. Karşılaştırma sonuçları tablo 6'da gösterilmektedir.

Tablo 6. Beyazperde veri seti için önerilen modelin literatürdeki modellerde karşılaştırılması

Model Adı	Doğruluk (%)	Kesinlik (%)	Duyarlılık (%)	F1-skor (%)
Önerilen Model (CNN)	89,08	88,82	89,42	89,12
Model 1	83	78	90	84
Model 2	86,8	-----	-----	-----
Model 3	91.34	-----	-----	-----

Tablo 6'da yer alan sonuçlar incelendiğinde önerilen modelin Model 1 ve Model 2'den daha iyi sonuçlar elde ettiği, Model 3'ün ise önerilen modelden daha iyi sonuçlar elde ettiği görülmektedir. Model 1 ve Model 2'de kullanılan veri setleri, çalışmamızda kullanılan veri setleri ile birebir aynı iken, Model 3'te kullanılan veri seti Beyazperde veri setinin farklı bir versiyonudur. Model 3'te kullanılan versiyonda, önerilen modelde kullanılan versiyona göre daha fazla örnek bulunmaktadır. Bu bağlamda Model 3'te daha iyi sonuç elde edilmesinin bir nedeninin ise örnek sayısındaki bu artışın olduğu düşünülmektedir.

Teşekkür

Bu çalışma Detay Danışmanlık Bilgisayar Hizmetleri Sanayi ve Dış Ticaret Anonim Şirketi Ar-Ge merkezinde yapılan çalışmaların bir çıktısıdır. Destekleri için teşekkür ediyoruz. Bu makalede bildirilen deneyler Tübitak Ulakbim, Yüksek Performans ve Grid Hesaplama Merkezi'nde (TRUBA kaynakları) gerçekleştirilmiştir. Y.G., fikir sahibidir, makale yazımında görev almıştır ve derin öğrenme analizleri gerçekleştirmiştir, B. A. Veri seti toplama ve ön işlem süreçlerini gerçekleştirmiştir, H. A. Makale yazımında görev almıştır, sonuçları yorumlamıştır ve derin öğrenme modeli geliştirme süreçlerinde yer almıştır.

Kaynaklar

- [1] Hovy EH. “What are Sentiment, Affect, and Emotion? Applying the Methodology of Michael Zock to Sentiment Analysis”, *Language Production, Cognition, and the Lexicon*, 2015, 13-24.
- [2] Sánchez-Rada JF, Iglesias CA. “Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison”, *Information Fusion*, 2019, 52, 344-356.
- [3] Wankhade M, Rao ACS, Kulkarni C. “A survey on sentiment analysis methods, applications, and challenges”, *Artif Intell Review*, 2022, 55(7), 5731-5780.
- [4] Singh T, Kumari M. “Role of Text Pre-processing in Twitter Sentiment Analysis”, *Procedia Computer Science*, 2016, 89, 549-554.
- [5] Adali E, Adamov AZ. “Sentiment analysis for agglutinative languages”, *IEEE 10th International Conference on Application of Information and Communication Technologies*, 2016, Bakü, Azerbaycan, 1-3.
- [6] Yıldırım E, Çetin FS, Eryiğit G, Temel T. “The Impact of NLP on Turkish Sentiment Analysis”, *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, 2014, 7(1).
- [7] Singh J, Singh G, Singh R. “Optimization of sentiment analysis using machine learning classifiers”, *Human-centric Computing and information Sciences*, 2017, 7(1), 1-12.
- [8] Jagdale RS, Shirsat VS, Deshmukh SN. “Sentiment Analysis on Product Reviews Using Machine Learning Techniques”, *Cognitive Informatics and Soft Computing*, 2019, 639-647.
- [9] Jain AP, Dandannavar P. “Application of machine learning techniques to sentiment analysis”, *2nd International Conference on Applied and Theoretical Computing and Communication Technology*, 2016, Karnataka, Hindistan, 628-632.
- [10] Gupta B, Negi M, Vishwakarma K, Rawat G, Badhani P. “Study of Twitter Sentiment Analysis using Machine Learning Algorithms on Python”, *International Journal of Computer Applications*, 2017, 165, 29-34.
- [11] Shamantha RB, Shetty SM, Rai P. “Sentiment Analysis Using Machine Learning Classifiers: Evaluation of Performance”, *IEEE 4th International Conference on Computer and Communication Systems*, 2019, Singapore, 21-25.
- [12] Naresh A, Venkata Krishna P. “An efficient approach for sentiment analysis using machine learning algorithm”, *Evolutionary intelligence*, 2021, 14(2), 725-731.
- [13] Araque O, Corcuera-Platas I, Sánchez-Rada JF, Iglesias CA. “Enhancing deep learning sentiment analysis with ensemble techniques in social applications”, *Expert Systems with Applications*, 2017, 77, 236-246.
- [14] Jianqiang Z, Xiaolin G, Xuejun Z. “Deep Convolution Neural Networks for Twitter Sentiment Analysis”, *IEEE Access*, 2018, 6, 23253-23260.
- [15] Xu G, Meng Y, Qiu X, Yu Z, Wu X. “Sentiment Analysis of Comment Texts Based on BiLSTM”, *IEEE Access*, 2019, 7, 51522-51532.
- [16] Zhao W vd. “Weakly-Supervised Deep Embedding for Product Review Sentiment Analysis”, *IEEE Transactions on Knowledge and Data Engineering*, 2018, 30(1), 185-197.
- [17] Basiri ME, Nemati S, Abdar M, Cambria E, Acharya UR. “ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for sentiment analysis”, *Future Generation Computer Systems*, 2021, 115, 279-294.
- [18] Mishev K, Gjorgjevikj A, Vodenska I, Chitkushev LT, Trajanov D. “Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers”, *IEEE Access*, 2020, 8, 31662-31682.
- [19] Naseem U, Razzak I, Musial K, Imran M. “Transformer based Deep Intelligent Contextual Embedding for Twitter sentiment analysis”, *Future Generation Computer Systems*, 2020, 113, 58-69.
- [20] Zhang T, Gong X, Chen CLP. “BMT-Net: Broad Multitask Transformer Network for Sentiment Analysis”, *IEEE Transactions on Cybernetics*, 2022, 52(7), 6232-6243.
- [21] Myagmar B, Li J, Kimura S. “Cross-Domain Sentiment Classification With Bidirectional Contextualized Transformer Language Models”, *IEEE Access*, 2019, 7, 163219-163230.
- [22] Potamias RA, Siolas G, Stafylopatis AG. “A transformer-based approach to irony and sarcasm detection”, *Neural Computing and Applications*, 2020, 32(23), 17309-17320.
- [23] Murfi H, Syamsyuriani, Gowandi T, Ardaneswari G, Nurrohmah S. “BERT-based combination of convolutional and recurrent neural network for indonesian sentiment analysis”, *Applied Soft Computing*, 2024, 151, 111112.
- [24] Liu Z, Zhou B, Chu D, Sun Y, Meng L. “Modality translation-based multimodal sentiment analysis under uncertain missing modalities”, *Information Fusion*, 2024, 101, 101973.
- [25] Dehkharghani R, Saygin Y, Yanikoglu B, Oflazer K. “SentiTurkNet: a Turkish polarity lexicon for sentiment analysis”, *Lang Resources & Evaluation*, 2016, 50(3), 667-685.

- [26] Salur MU, Aydın İ. “Türkçe Tweetler için Derin Özellik Çıkarımı Tabanlı Yeni Bir Duygu Sınıflandırma Modeli”, Fırat Üniversitesi Mühendislik Bilimleri Dergisi, 2022, 34(1).
- [27] Ayvaz S, Yıldırım S, Salman YB. “Türkçe Duygu Kütüphanesi Geliştirme: Sosyal Medya Verileriyle Duygu Analizi Çalışması”, European Journal of Science and Technology, 2019, 16, 51-60.
- [28] Çetin FS, Eryiğit G. “Türkçe Hedef Tabanlı Duygu Analizi İçin Alt Görevlerin İncelenmesi – Hedef Terim, Hedef Kategori ve Duygu Sınıfı Belirleme”, Bilişim Teknolojileri Dergisi, 2018, 11(1), 43-56.
- [29] Onan A. “Evrimsel Sinir Ağı Mimarilerine Dayalı Türkçe Duygu Analizi”, European Journal of Science and Technology, 2020, 374-380.
- [30] Tuzcu S. “Çevrimiçi Kullanıcı Yorumlarının Duygu Analizi ile Sınıflandırılması”, Eskişehir Türk Dünyası Uygulama ve Araştırma Merkezi Bilişim Dergisi, 2020, 1(2).
- [31] “winvoker/turkish-sentiment-analysis-dataset · Datasets at Hugging Face”, 23 Ocak 2024, <https://huggingface.co/datasets/winvoker/turkish-sentiment-analysis-dataset>.
- [32] “Turkish Sentiment Analysis Data - beyazperde.com”, 23 Ocak 2024, <https://www.kaggle.com/datasets/ozcan15/turkish-sentiment-analysis-data-beyazperdecom>.
- [33] “Multimedia Information Retrieval Laboratory - HUBBM”, 21 Mart 2024, <http://humirapps.cs.hacettepe.edu.tr/tsad.aspx>.
- [34] “string — Common string operations”, Python documentation, 24 Ocak 2024, <https://docs.python.org/3/library/string.html>.
- [35] “NLTK :: Natural Language Toolkit”, 24 Ocak 2024, <https://www.nltk.org/>.
- [36] “zemberek-python: Python port of open source text processing library for Turkish, zemberek-nlp”, 24 Ocak 2024, <https://www.lodoss.com.tr/>.
- [37] “Keras: Deep Learning for humans”, 24 Ocak 2024, <https://keras.io/>.
- [38] Jones DR. “A Taxonomy of Global Optimization Methods Based on Response Surfaces”, Journal of Global Optimization, 2011, 21(4), 345-383.
- [39] Wu J, Chen XY, Zhang H, Xiong LD, Lei H, Deng SH. “Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization”, Journal of Electronic Science and Technology, 2019, 17(1), 26-40.
- [40] Görmez Y, Aydın Z. “IGPRED-MultiTask: A Deep Learning Model to Predict Protein Secondary Structure, Torsion Angles and Solvent Accessibility”, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2023, 20(2), 1104-1113.
- [41] “scikit-optimize: sequential model-based optimization in Python — scikit-optimize 0.8.1 documentation”, 25 Ocak 2024, <https://scikit-optimize.github.io/stable/>.
- [42] “Metrics and scoring: quantifying the quality of predictions”, scikit-learn, 21 Mart 2024, https://scikit-learn/stable/modules/model_evaluation.html.
- [43] Yıldırım M, Okay FY, Özdemir S. “Sentiment Analysis for Turkish Unstructured Data by Machine Translation”, IEEE International Conference on Big Data (Big Data), 2020, Atlanta, GA, USA, 4811-4817.
- [44] Alqaraleh S. “Turkish Sentiment Analysis System via Ensemble Learning”, European Journal of Science and Technology, 2020, 4(3), 122-129.
- [45] Acikalin UU, Bardak B, Kutlu M. “Turkish Sentiment Analysis Using BERT”, 28th Signal Processing and Communications Applications Conference (SIU), 2020, Gaziantep, Türkiye, 1-4.