Fırat Üniversitesi Müh. Bil. Dergisi
32(2), 561-570, 2020

Derin Öğrenme ve Aşağı Örnekleme Yaklaşımları Kullanılarak Duygu Sınıflandırma Performansının İyileştirilmesi

Yunus SANTUR
Enformatik Bölümü, Fırat Üniversitesi, Elazığ, Türkiye
ysantur@firat.edu.tr

(Geliş/Received: 28/06/2020; Kabul/Accepted: 14/08/2020)

Öz: Bir metnin hangi duygu sınıfına ait olduğunu bulma problemi duygu sınıflandırma olarak bilinmektedir. Bu işlemin otomatize bir şekilde yapılması çevrimiçi ortamda büyük miktarda verinin çok kısa sürelerde analiz edilebilmesine olanak sağlamaktadır. Böylece müşteri memnuniyetini ölçme, reklam ve içerik önerme gibi birçok farklı amaçla kullanılabilmektedir. E-ticaret uygulamalarında duygu sınıflandırma için kullanıcı yorumlarının yanı sıra, memnuniyet derecesini ölçen sayısal bir puanlama ya da duygu durumunu kategorik bir değişken olarak ifade edecek bir değişkene daha ihtiyaç duyulmaktadır. Bu sayede etiketli verilerden oluşan veri seti üzerinde denetimli öğrenme ile model oluşturulmaktadır. Burada yaşanan bir dezavantaj kullanıcıların bir üründen çoğunlukla memnun olmaları ya da tam tersi şikâyetçi olmalarıdır. Bu durumda oluşan veri seti dengesiz olmaktadır. Bu çalışmada Türk e-ticaret platformu Hepsiburada firmasına ait 243 bin kullanıcı yorumundan oluşan veri seti kullanılmıştır. Dengesiz olan bu veri setinde, sınıflandırma performansının iyileştirilmesi için derin öğrenme algoritmaları kullanılmış ve dengesiz veri seti yaklaşımı sunulmuştur. Sunulan yaklaşım ile yanlış pozitif oranı % 69'dan % 90'a, doğruluk değeri ise % 95.5'ten % 99'a iyileştirilmiştir.

Anahtar kelimeler: Derin Öğrenme, Metin Sınıflandırma, Yinelenen Sinir Ağları

Improving Sentiment Classification Performance Using Deep Learning and Undersampling Approaches

Abstract: The problem of finding out which emotion class a text belongs to is known as sentiment classification. Performing this process in an automated manner enables large amounts of data to be analyzed in a very short time, online. Thus, it can be used for many different purposes such as measuring customer satisfaction, recommending advertisements and content. In e-commerce applications, besides user comments for emotion classification, a numerical scoring that measures the degree of satisfaction or another variable that will express the emotional state as a categorical variable is needed. In this way, a model is created with supervised learning on the data set consisting of labelled data. A disadvantage here is that users are mostly satisfied with a product or vice versa complained. In this case, the data set is unbalanced. In this study, a data set consisting of 243 thousand user comments of the Turkish e-commerce platform Hepsiburada was used. In this unbalanced dataset, deep learning algorithms were used to improve classification performance and an unbalanced dataset approach was presented. With the approach presented, the false positive rate was improved from % 69 to % 90 and the accuracy was improved from % 95.5 to % 99.

Keywords: Deep Learning, Recurrent Neural Network, Sentiment Classification

1. Giriş

Yapay zekânın alt dallarından biri olarak kabul edilen Doğal Dil İşleme (DDİ) metin verileri üzerinde duygu sınıflandırma, özet bilgi çıkarma, yazar tanıma, benzerlik ölçme, varlık tanıma, soru-cevap robotları geliştirilmesi, makine çevirisi gibi görevlerle ilgilenen bir disiplindir [1]. DDİ, dilin önden/sondan eklemeli olması, kullanılan kısaltma, bağlaç, noktalama işaretleri, cümle öğelerinin sırası, özellikle çevrimiçi kaynaklarda kullanılan ve duygu durumları ifade eden emojiler nedeni ile karmaşık bir süreç olarak ele alınmaktadır. DDİ süreci verinin hazırlanması ve makine öğrenmesi kullanılarak model geliştirilmesi şeklinde ana adımda ele alınmaktadır.

DDİ'de kullanılan metin veri seti "Külliyat" olarak adlandırılmaktadır. DDİ süreci, bu külliyattan gelen veriler üzerinde faydasız kelime tespiti, kök bulma, vektörleştirme gibi ön işleme ve veri temizleme yapılması, ardından denetimli/denetimsiz/pekiştirmeli veya hibrit makine öğrenmesi algoritmaları kullanılarak bir model geliştirilmesi ve çıktı olarak amaçlanan bilginin elde edilmesi şeklinde işlemektedir [2].

* Sorumlu yazar: ysantur@firat.edu.tr. Yazarın ORCID Numarası: 0000-0002-8942-4605

DDİ'de kullanılan yaklaşımlardan birisi olan duygu analizi bir metnin daha önce belirlenen sınıflardan hangisine ait olduğunun tespit edilmesidir. Duygu analizi günümüzde sosyal medya başta olmak üzere sağlık, bilişim, e-ticaret gibi birçok farklı alanda kullanıcıların haber, paylaşım ve e-ticaret platformlarında ürünlere yaptıkları yorumların adaptif olarak sınıflandırılması ve bu bilgilerin analiz edilmesi amacıyla yaygın olarak kullanılmaktadır. Bu sayede insan eliyle işlenemeyecek ölçüde büyük veri makine öğrenmesi ile işlenebilmektedir [3]. E-ticaret alanında duygu analizi, kullanıcı-ürün ve kullanıcı-platform etkileşimi kullanılarak fiyat ve tanıtımların belirlenmesi, müşteri memnuniyetinin ölçülmesi, müşteri kayıp analizi ve kişiye özgü kampanyaların yapılması gibi farklı amaçlarla kullanılabilmektedir. Bu amaçla kullanıcıdan ürün ve/veya mağaza/süreçle ilgili metin ve duygu durumun sınıfını ifade edecek kategorik bir giriş istenmektedir. Bu kategorik giriş puanlama ya da doğrudan sınıfa ait bir emoji olabilmektedir. Puanlama olması durumunda "Denklem 1" de verildiği gibi eşik değer karşılaştırması yapılarak kullanıcıdan alınan yorum pozitif/negatif gibi sınıflandırılabilir. Çoklu giriş aracılığı ile verilecek kategorik girişlerde ise daha çok sınıf ifade edilebilmektedir. "Denklem 1" de verilen t giriş cümlesi olmak üzere e seçilen eşik değer ve p ise yoruma verilen puandır. Bu durumda pozitif ve negatif olmak üzere iki sınıflı bir etiketleme yapılmış olunmaktadır. Bazı durumlarda skor skalasındaki ortanca değer "nötr" olarak kabul edilerek üç sınıflı bir etiketleme yapılabilmektedir. Sonuç olarak giriş metni ve sınıf etiketinden oluşan iki sütunlu etiketli bir seti ile denetimli makine öğrenme süreci kullanan makine öğrenmesi algoritmaları ile eğitim işlemi gerçekleştirilmektedir [4].

Duygu analizinde, ön işleme aşamasında gerçekleştirilen bir diğer işlemde vektörleştirme yani metin verilerinin sayısallaştırılmasıdır. Sayısallaştırma için kullanılacak en basit algoritma her bir kelimenin benzersiz lojik bir değer olarak ifade edilmesini sağlayan "one-hot-encoding" yaklaşımıdır. Ancak bu yaklaşım külliyat içinde geçen her kelime için ayrı bir kodlama gerektirdiğinden giriş verilerinin boyutunu önemli ölçüde arttırmaktadır. Metin işleme algoritmalarının gerektirdiği bellek ve işlem gücü yüksektir, bu nedenle geleneksel makine öğrenmesi algoritmaları yerine derin öğrenme algoritmalarına ihtiyaç duyarlar. Derin öğrenme kullanmanın bir diğer avantajı, makine öğrenmesinde kullanılan boyut indirgeme ve özellik seçim işlemlerinin bu yaklaşımda model tarafından keşfedilebilir ve öğrenilebilir olmasıdır [5].

$$s(t) = \begin{cases} 1, & t(p) \geq e \\ -1, & t(p) < e \end{cases} \quad (1)$$

2. Duygu Analizi

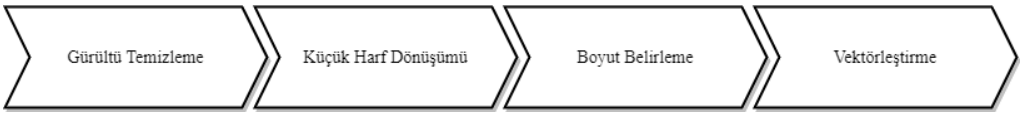
Duygu analizi, etiketli veriler üzerinde denetimli makine öğrenmesi ile eğitilen bir model kullanılarak giriş metninin önceden belirlenen hangi duygu sınıfına ait olduğunun sınıflandırılması şeklinde işleyen bir süreçtir. Literatürde birçok amaçla faydalı olarak kullanılabilmektedir. En bilinen kullanım alanı, sosyal medya kullanıcılarının gündem konuları ile ilgili yazdıkları yorumların otomatize sınıflandırılmasıdır [6, 7]. Bunun yanı sıra duygu analizinin turizm alanında geliştirilmesi için büyük veri ile entegre edilen yaklaşımlar geliştirilmiştir [8].

Abualigah ve diğerleri (2020) çevrimiçi dökümanların kullanılarak sağlık bakım kalitesinin arttırılması amacıyla bu alana özgü yapılan çalışmaları inceleyen bir derleme çalışma gerçekleştirmiştir [9]. Doğal dil işlemenin bir alt dalı olan duygu analizi, dilin gramatik yapısı ve öğe dizilimi gibi birçok alt problemle de uğraşmaktadır, bu açıdan Çince ve Arapça gibi dillere özgü farklı yaklaşımlar gerçekleştirilmesi gerekmektedir [10]. Yousif ve diğerleri (2017) bilimsel çalışmalarda benzerlik analiz yapılması, bilimsel atıflardaki duygu analizi yapılması için hibrit yaklaşımlar geliştirmişlerdir [11]. Duygu analizinin son yıllarda artan bir ilgiyle kullanıldığı alanlardan birisi de finans sektörüdür. Hisse senetlerinin gelecekteki fiyat hareketleri için finans ile ilgili çevrimiçi veri izleme platformlarında kullanıcı ve analistlere ait yorumların adaptif olarak sınıflandırılması ve zaman serileri üzerinde hibrit yaklaşımların kullanılması ile hisse senedi trend tahmini metrik değerlerinin iyileştiği görülmüştür [12]. Bu çalışmada ise bir e-ticaret sistesine ait kullanıcı yorumlarını içeren ve oldukça dengesiz olan bir veri setinde sınıflandırma performansının iyileştirilmesi amaçlanmıştır, genel olarak duygu sınıflandırma çalışmalarında kullanılan ön işlemler sıralı olarak aşağıdaki gibi verilmiştir [13].

Gürültü temizleme: Metin içinde yer alan noktalama işaretleri metinden atılır. Metinden atılması gereken diğer grup ise İngilizcede "Stop Words" olarak bilinen eklerdir. Türkçede faydasız kelimeler olarak adlandırılan bağlaç gurubu bu ekler öğrenme modelinin eğitiminde etkisi olmadığı için boyut küçültme ve eğitim/test adımında gereksinim duyulan işlem gücünün gereksiz kullanılmaması amacıyla metinden atılır.

Dönüşüm: Metin içinde büyük/küçük harfler karışık olabilir, metnin eğitimden önce sayısallaştırılmasında farklı temsillere neden olacağı ve bu durumunda modelin başarımını kötü yönde etkileyeceği için tüm metin genel olarak küçük harflere dönüştürülür.
Boyut seçimi: Görüntü işleme uygulamalarında olduğu gibi veri setinin eş bir boyuta getirilmesi işlemidir. Eğitim ve test işleminde giriş verilerinin aynı boyutta olması gerekmektedir. Metin verileri birbirinden çok farklı boyutta olabilmektedir. Aynı boyuta getirme için "Denklem 2" de verildiği gibi külliyat içinde yer alan kelimelerin frekans değerleri kullanılır. Eşitlikte metinde yer alan her kelimenin frekansını temsil eden C_i değeri külliyat için seçilen eşik değer olan C_e değerinden daha büyük ise metin vektöründe tutulur. Her bir giriş metninde frekansı düşük kelimeler metinden atılır. Bu noktada yaşanabilecek bir problem çok kısa yorumların belirlenen boyuttan daha düşük kelime içermesidir. Bu durumda metin vektörleştirme aşamasında 0 gibi seçilen bir değerle boyuta eşit olana kadar doldurulur.
Vektörleştirme: Eğitim işleminden önceki en önemli adımdır. Makine öğrenmesi algoritmaları sayısal değerlere ihtiyaç duyarlar. Metin verilerinin sayısallaştırılması için birçok yaklaşım kullanılabilmektedir. Yaklaşımların ortak özelliği metin verilerinin sayısallaştırılarak temsil edilmesi ve öğrenme modeline giriş oluşturacak özelliklerin seçiminin sağlanmasıdır.

$$b = C_i \sum Eğer\ C_i(f) > C_e(f) \tag{2}$$



Şekil 1. Metin sınıflandırma için kullanılan ön işlem adımları

2.1. Vektörleştirme Yöntemleri

One-hot Encoding: Şekil 1'de verilen ilk vektörleştirme yönteminde metin içindeki her kelime tek bir bit lojik "1" diğerlerinin ise lojik "0" olarak kodlanması ise elde edilir. Vektörleştirme için kullanılabilecek en basit yöntemlerden bir tanesidir. Ancak külliyatın boyu ile orantılı olarak oldukça fazla sayıda vektör elde edilmiş olunur, bu durum öğrenme algoritmasının eğitim işleminin uzamasına neden olmaktadır. Bu tekniğin, en önemli dezavantajı ise metinde yer alan öğelerin sırası ve birbirleri ile olan ilişkilerini temsil edemeyecek oluşudur [14].
Bag of Words: Kelime torbası anlamına gelmektedir. Bu yaklaşımda külliyat içinde geçen kelimelerin sıklığı temsil olarak kullanılır, "Denklem 3" te $x_{i,j}$ metin içinde kelimenin geçip geçmediğini gösteren mantıksal değişken olmak üzere, bow külliyat içindeki kelimelerin kelime çantası haline getirilmesini temsil eder, sayısallaştırma için her kelime için 1 veya 0 değerleri külliyat içindeki sıklıklarına göre belirler. Bu yöntemde "one-hot encoding" yönteminde olduğu gibi kelime ve öğe sırası bilgisi tutulmaz. Ancak temsil edilen boyut daha küçüktür. Bu yaklaşım metin sınıflandırmadan ziyade, öğe diziliminin daha az önemli olduğu spam belirlemede yaygın olarak kullanılmaktadır [14].

$$bow = \begin{pmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{pmatrix}$$
$$x_{i,j} = \begin{cases} 1, & Eğer\ x_{i,j}\ içindeyse\ C \\ 0, & Değilse \end{cases} \tag{3}$$

TF-IDF: Bir kelimenin külliyat içindeki önemini gösteren istatistik temelli bir ağırlıklandırma yöntemidir. "Denklem 4" te $tf_{i,j}$ i'nin j içindeki frekansını, df_i i terimini içeren toplam doküman sayısını ve N ise toplam doküman sayısını vermektedir. Metin işlemede faydasız kelime tespiti genelde kural tabanlı olarak çalışmaktadır. Ancak bu yöntem en sık geçen terimlerin ağırlığını düşürdüğünden noktalama işareti ve bağla gibi faydasız kelimelerin elenmesi amacı ile de kullanılabilmektedir. BoW yönteminde olduğu gibi öğe dizilimi ile ilgili bir bilgi tutmazlar. Ancak web sitelerinin pagerank gibi puanlaması gibi algoritmalarda kullanılmaktadır [15].

$$w_{i,C} = tf_{i,j} x log(\frac{N}{df_i}) \tag{4}$$

N-gram: Veri üzerinde arama ve karşılaştırma yaparken tekrar sayısını belirlemek için kullanılan bir algoritmadır. N tekrar derecesini ifade etmektedir. N sayısına göre 1:unigram, 2:bigram, 3:trigram olarak isimlendirilmektedir. Arama motorlarında olduğu gibi metin arama ve birbirleri ile en çok ilişkili kelimelerin bulunmasında kullanılmaktadır. Döküman aramada en sık kullanılan algoritmalardan bir tanesidir, bu özelliklerinin yanı sıra diğer bazı vektörleştirme algoritmalarında da kullanılırlar [16].

Word2vec: Detimsiz öğrenme ve tahmin tabanlı olarak çalışan, metin içindeki öğelerin dizilimini temsil edebilen vektörleştirme algoritmalarından bir tanesidir. Google araştırmacısı Mikolov (2013) tarafından geliştirilmiştir [17]. Word2vec CBOW ve Skip-gram olmak üzere iki algoritma kullanmaktadır. Her ikisi de gizli katmandan oluşan bir yapay sinir ağı (YSA) kullanmaktadır. Bu YSA'da giriş ve gizli katmanda aktivasyon fonksiyonu kullanılmaz, çıkış katmanında ise softmax bulunmaktadır. Word2vec algoritmasının en önemli hiper parametrelerinden bir tanesi w_t giriş penceresi sayısının seçimidir. Bu giriş penceresi sayısına göre öncesinde/sonrasında geçen kelimeler tahmin edilmeye çalışıldığı için öğe dizilimi temsil edebilmektedir. YSA girişinde külliyat içinde yer alan benzersiz kelime sayısı kadar vektör oluşturularak one-hot encoding yapılmaktadır. Skip-gram algoritmasında w_t giriş olarak alınarak, diğer kelimeler tahmin edilmeye çalışılmaktadır, CBOW algoritması ise diğer kelimeleri giriş olarak alıp merkezdeki tek kelimeyi tahmin etme prensibine göre çalışmaktadır. CBOW algoritması küçük veri setlerinde daha etkindir, büyük veri setleri kullanan uygulamalarda ise Skip-gram daha performanslı çalışmaktadır. Benzer şekilde pencere sayısı küçük ise CBOW, büyük ise Skip-gram algoritması tercih edilmektedir [17].

Kullanılan YSA'nın çıkış katmanı olasılıksal değerleri içermektedir. Verilen girişe göre çıkış değeri olacak e tüm e^x lere bölündüğünde, her bir çıkışın kendi olasılıksal değeri elde edilmiş olunur. Bu yaklaşım "Denklem 5" te olduğu gibi eşitlik olarak ifade edilebilmektedir. Verilen eşitlikte $P(w_{t+j}|w_t)$ w_t kelimesinden pencere sayısı (j) sonra w_{t+j} kelimesinin gelme olasılığıdır. Word2vec algoritmasında varsayılan eğitim tur sayısı 5, pencere sayısı 10, vektör boyutu ise 300'dür. Metinde ki öğe sırasını temsil edebilmesinin yanı sıra, eğitim işleminin uzun olması en büyük dezavantajlarıdır. Google geliştiricileri bu dezavantajı giderebilmek için negatif örneklemeye dayalı bir başka yaklaşım geliştirmişlerdir. İngilizce için eğitilmiş modeller kullanmak avantaj yaratmaktadır, ancak Türkçe için word2vec kullanılarak eğitilmiş veri seti bulmak zordur [18].

$$P(w_{t+j}|w_t) = e^{U_{t+j}^T V_{wt}} \Big/ \sum_{w=1}^{v} e^{U_{wt}^t} \tag{5}$$

Fasttext: Word2vec kullanılarak Facebook (2016) araştırmacıları tarafından geliştirilmiştir. Word2vec algoritmasında kullanılan YSA yapısında giriş olarak one-hot encoding ile kodlanmış vektörleri kullanmadan önce metin n-gram algoritmasına göre prçalanır. Word2vec ile kıyaslandığında birlikte geçen kelimeler daha iyi düzeyde temsil edilebilmektedir. Bu bağlamda word2vec ile kıyaslandığında pencere sayısı ile birlikte n-gram algoritmasındaki n sayısınıda hiper parametre olarak verilmektedir [19].

GloVe: Word2vec algoritmasında kullanılan CBOW ve Skip-gram istatistik temelli olarak kelimelerin anlamsal birlikteliklerini yakalayabilir ancak birlikte kullanılma istatistiklerini kullanmazlar. Pennington (2014) tarafından önerilen bu modelde "Denklem 6" da verilen yeni yaklaşımla istatistiklerin daha etkin kullanımını amaçlamışlardır. Eşitlikte X_{ij} külliyat içinde kelime çiftlerinin birlikte geçme sayısı, w_i, w_j ise külliysat içindeki geçme sayısı, V külliyattaki kelime sayısıdır [20].

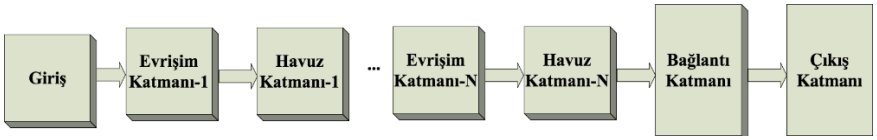
$$J = \sum_{i,j=1}^{V} f(X_{ij})\left(w_i^T w_j + b_i + b_j - logX_{ij}\right)^2 \tag{6}$$

3. Derin Öğrenme

Makine öğrenmesi ile kıyaslandığında derin öğrenme büyük veri setleri ve karmaşık görevler için daha uygundur. Makine öğrenmesinde, daha küçük veri setlerinde öğrenme başarımını arttırmak için özellik seçimi gibi işlemler elle yapılabilir; derin öğrenme algoritmalarında ise elde bulunan nitelikli verilerden özellik seçimi gibi işlevlerinde adaptif yapılması sağlanmaktadır. Bu çalışmada kullanılan derin öğrenme algoritmaları aşağıda incelenmiştir [21].

3.1. Evrişimli Sinir Ağları

Evrişimli Sinir Ağları (ESA) özellikle, Dünya genelinde düzenlenen Imagenet yarışmalarında en iyi performans gösteren algoritmalardan birisi olduğu için oldukça popüler hale gelmiştir. Çekişmeli üretici ağlar ve oto kodlayıcılar gibi birçok farklı sinir ağında kullanılan ESA'lar tek ve çok boyutlu verilerle çalışabilmektedir [22].



Şekil 2. ESA yapısı

Şekil 2'de ESA modelinin genel blok diyagramı ve çalışma adımları verilmiştir. Bu adımlar sırası ile giriş verisini alma, giriş verisi üzerinde alt-örnek oluşturma (evrişim) ve oluşturulan alt-örnekler üzerinde havuz (özellik seçimi) yapma işlemlerini kapsamaktır. Öğrenme ağında istenildiği kadar özyinelemeli olarak evrişim ve havuz ara katmanları çoğaltılabilir. Son adımda tüm ağdaki tüm sinir hücrelerinin bağlandığı bağlantı katmanı ve sınıflandırıcı sayısı kadar çıkış hücresi bulunmaktadır [22].

Evrişim Katmanı: Bu aşamada giriş verisi üzerinde daha küçük bir çekirdek matris seçilerek konvolüsyon işlemi uygulanmaktadır. Giriş matrisi üzerinde satır ve sütunlar boyunca gezdirilen bu çekirdek matris sonrası giriş verisinden n adet alt veri matrisi oluşturulmuş olur. Bu işlem ESA sınıflandırma başarısı açısından büyük önem arz etmektedir. Saha uygulamalarında sınıflandırıcının başarısını arttırmak için giriş verisine gürültü ekleme, simetri alma, döndürme gibi ön-işlemler de uygulanarak veri kümesi genişletilir. Eğitim işlemine bu ön adımları eklemek sözü edilen gürültülü, simetrik, döndürülmüş verinin de tanınmasını sağlamaktadır. "Denklem 7" de f giriş verisi k seçilen mxn boyutunda kernel matrisi olmak üzere bu konvolüsyon sonrası i adet E ile ifade edilen alt örnek elde edilmektedir.

$$E_{ij}^{l} = \sum_{a=0}^{m-1} \sum_{b=0}^{m-1} w_{ab} f_{(i+a)(j+b)}^{l-1} \tag{7}$$

Havuz Katmanı: Elde edilen alt-örnekler üzerinde mxn boyutunda ikinci bir kernel matris ile tekrar konvolüsyon işlemi uygulanır ancak bu defa alt-örnek elde etmek yerine çekirdek matris içindeki verilerden ağdaki hücrelerin eğitiminde kullanılacak ve alt-örneği ifade eden özellikler seçilir. Bu aşamada çekirdek boyutu minimum $2x2$ boyutunda olmaktadır. Özellik olarak matris içindeki minimum, maksimum ya da ortalama değer seçilebilir. "Denklem 8" de verilen E_i bir önceki adımda elde edilen alt-örnek olmak üzere bu alt-örnekler üzerinde seçilen mxn boyutundaki p kernel matrisi gezdirilerek her bir pencereden "Denklem 9" da olduğu gibi özellik değerler elde edilir. Geleneksel yapay sinir ağları kullanılarak gerçekleştirilen çalışmalarda veri girişi olarak kullanılacak görüntü üzerinde boyut küçültme ve öz-değer elde etme işlemi kullanılmaktadır. Ancak ESA modelleri üzerinde giriş verisi üzerinde verinin anlamını düşürmemek için boyut küçültme/öz-değer elde etme kullanılmaz. Hatta tam aksine modelin iyi eğitilebilmesi için gürültü ekleme gibi ön-işlemler ile veri genişletilebilir.

$$P_{vec} = \sum_i E_i \text{*p} \tag{8}$$

$$\sum_{1}^{n} P_{vec} = \begin{Bmatrix} min \\ max \\ mean \end{Bmatrix} \sum_i E_i \tag{9}$$

Bağlantı Katmanı: Bu katman çıkış katmanından bir önceki katmandır. ESA modelinde yukarıda verilen evirişim ve havuz katmanları öz-yinelemeli olarak tekrar edilebilir. Seçilecek katman sayısı ağın eğitiminde aşırı öğrenme ve eksik öğrenme oluşturmayacak şekilde belirlenmelidir.

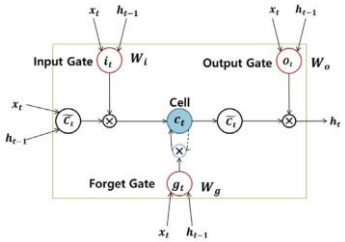
Çıkış Katmanı: ESA modelinde elde edilmek istenen sınıf sayısı kadar hücre bulunan ağdaki son katmandır. Genellikle çıkış değerleri yüzdelik tahmini ifade eden {0-1} aralığına normalize edilmiş değerler olmaktadır.

3.2. Tekrarlayan Sinir Ağları

Tekrarlayan Sinir Ağları (TSA) hafızalı ağlar olarak bilinirler. İleri beslemeli ve/veya çok katmanlı sinir ağları anlık girişlere göre çıkış üretirler [23]. TSA'lar ise t-1 ve daha önceki çıkışları da ağ eğitiminde kullanırlar. Bu özellikleri nedeni ile birbirlerine bağlı ardışıl verilerden oluşan zaman serileri üzerinde daha etkindirler. Bu bağlamda öğeleri birbirine bağlı olan metin verileri de zaman serilerine benzedikleri için TSA ağları için daha uygundurlar. Bu nedenle sohbet robotları, makine çevirisi gibi birçok metin işleme uygulamasında TSA'lar yaygın olarak kullanılırlar. TSA'lar bu avantajlarının yanı sıra gradyan problemleri ortaya çıkarabilmektedir. Bu dezavantaj gradyan değerinin aşırı büyüyerek optimum değerden uzaklaşma ya da sıfıra çok yaklaşarak yok olması şeklinde gerçekleşmektedir. Bu dezavantajların giderilmesi için eşik değer ya da ağ çıkışında Relu gibi doğrusal olmayan aktivasyon fonksiyonları kullanılabilir [24]. Hochreiter ve Schmidhuber (1997) TSA'lar da yaşanan gradyan problemlerini çözen UKSB algoritmasını geliştirmiştir [25]. UKSB genelleştirilmiş bir çeşidi olan GTB'ler ise daha küçük boyutlu veriler üzerinde kullanılan ve daha kolay eğitilebilen yapı olarak daha basit hafızalı ağlardır [26].

3.2.1. Uzun Kısa Süreli Bellek (UKSB)

Şekil 3'te verilen UKSB'ler YSA'larda kullanılan nöronlar yerine hafıza blokları içermektedir. Şekilde görüldüğü gibi bir UKSB yapısı bir bellek hücresi (c_t) ile giriş kapısı (i_t), çıkış kapısı (o_t) ve unutma kapısı (g_t) olmak üzere 3 kapıdan oluşmaktadır. $X_t$ve h_tsırası ile t anındaki giriş ve gizli durumdur. "Denklem 10-15" te verilen *U, W* ağırlık *b* ise bias değerleridir (Kim et al., 1997).



Şekil 3. UKSB yapısı [27]

$$g_t = \sigma(U_g x + W_g h_{t-1} + b_f) \quad (10)$$
$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \quad (11)$$
$$\widetilde{c_t} = tanh(U_c X_t + W_c h_{t-1} + b_c) \quad (12)$$
$$c_t = g_t * c_{t-1} + i_t * \widetilde{c_t} \quad (13)$$
$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \quad (14)$$
$$h_t = o_t * \tanh(c_t) \quad (15)$$

3.2.2. Geçitli Tekrarlayan Birim (GTB)

UKSB ile kıyaslandığında GTB eğitimleri daha kolay, daha genelleştirilmiş derin öğrenme algoritmalarından birisidir. GTB'ler de hafıza birimi seçimliktir, en önemli fark ise unutma kapısının olmamasıdır. Daha basit olmalarına rağmen küçük veri setleri ve birçok problemde UKSB'ye çok yakın performans elde edilebilmekte bunun yanısıra eğitimleri daha kolay ve daha az süre almaktadır. Daha kompleks uygulamalarda ve hatırlanması gereken uzun süreli çıkışlar var ise UKSB'ler daha avantajlı olmaktadır.

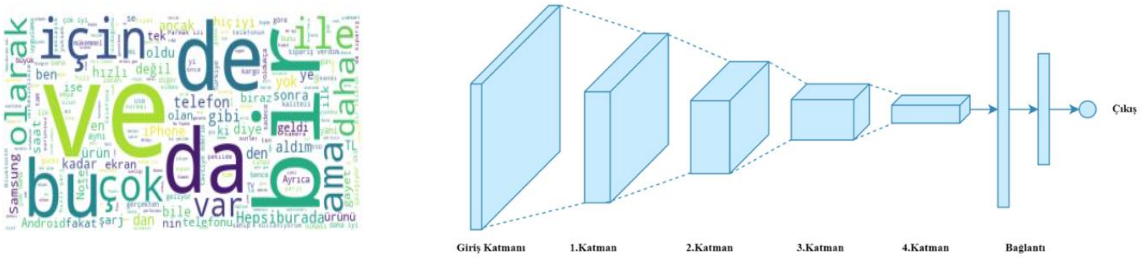
4. Veri Seti ve Metot

Çalışmada derin öğrenme algoritmalarının metin sınıflandırma performanslarının kıyaslanması için Türk e-ticaret platformu Hepsiburada firmasının Kaggle platformunda yayınlanan kullanıcı yorumlarını içeren veri seti kullanılmıştır [28]. Çalışmada kullanılan veri seti 243.000 kullanıcı yorumunu ve yorumun duygu sınıfını (pozitif, negatif) içermektedir. Veri setinin yaklaşık % 98'i pozitif, % 2'si ise negatif yorumları içermektedir. Bu yönü ile oldukça dengesiz bir veri setidir.

Tablo 1. Hepsiburada veri setine ait örnekler [28]

Yorum	Sınıf etiketi
Logitech M175 Kablosuz Nano Mouse gerçek güzel bir ürün ben memnun kaldım tavsiye ederim sonuçta bilindik bir marka ayraca fiyat bakımından da uygun	1
Daha önce elimde olan m195 modeline göre çok daha kötü çıktı.Kaydırma tekerleği bi garip bazen işlemiyor gibi bir de çok sesli.Bunun yanında tıklarken yada gezinirken mi tam anlamadım ama bi gıcırtı g...	0
Mausum tıklama esnasında tutukluk yapıyor aynı noktaya birkaç defa tıklamam gerekiyor anlatacağım bunlar ama 20 kelime söyleyecek bişey yok maalesef	0
ürünü sipariş ettim.hemen elime geçti.gayet iyi çalışıyor.kablolu mause bana göre değil sürekli çocuklar kablosunu koparıyorlar.bu derten de bu sayede kurtulmuş oldum.herkese almasını tavsiye ederim.	1

Tablo 1'de verilen veri seti 13.676 negatif yorum, 229.821 pozitif yorum içermektedir. En kısa yorumlar "harika", "başarılı" gibi tek kelimeden oluşmaktadır. En uzun yorum ise 329 kelimeden oluşmaktadır. Veri setindeki yorumların ortalama kelime sayısı ise 23'tür. Bu değerler bağlaç ve bazı noktalamalar atıldıktan sonra elde edilmiştir. Veri setinde faydasız kelime tespiti için en sık geçen kelimelerin frekansı ölçülmüştür. Aşağıda veri setinde pozitif ve negatif yorumlarda en sık geçen 20 kelime frekans büyüklüğü sırasına göre verilmiştir. Şekil 4'te ise veri setinin kelime bulutu gösterimi verilmiştir.



Şekil 4. Veri seti kelime bulutu gösterimi ve önerilen yaklaşım

4.1. Önerilen Yaklaşım

Çalışmada aynı veri seti üzerinde, aynı koşullar altında çalışma süresi, bellek ve doğruluk performanslarının ölçümü içine ESA, UKSB ve GTB kullanılmıştır. Önerilen genel yaklaşım Şekil 4'te verilmiştir.

Tablo 2. Öğrenme modelinde kullanılan hiper parametreler

Nöron/hücre sayısı (ESA, UKSB ve GTB için aynı seçilmiştir)				Giriş uzunluğu	Optimizasyon	Budama	Aktivasyon fonksiyonu
1.katman	2.katman	3.katman	4.katman				
32	16	8	4	50	Adam	.1	Sigmoid, .5

4.2. Değerlendirme

Önerilen yaklaşımın değerlendirilmesi için çalışma süresi ve Tablo 3'te verilen karşıtlık matrisi üzerinde elde edilen performans metrikleri kullanılmıştır [30]. Test verileri üzerinde, tahmin ve gerçek sınıf "1" ise DP, tahmin ve gerçek sınıf "0" ise DN değeri "1" arttırılır. Eğer gerçekte negatif olan yorum pozitif olarak tahmin edilmiş ise YP, gerçekte pozitif olan yorum negatif olarak tahmin edilmiş ise YN değeri 1 arttırılarak bu matris oluşturulmakta ve "Denklem 16-20" da verilen metrik ölçüler hesaplanmaktadır. Genel doğruluğun (D) yanı sıra sırası ile pozitiflerin doğru tahmin oranını (DPO) ve negatiflerin doğru tahmin (YPO) oranı yine karşıtlık matrisi kullanılarak hesaplanabilmektedir.

Tablo 3. Karşıtlık matrisi

	Gerçek Sınıf	
Model Tahmini	DP	YP
	YN	DN

$$D = (DP + DN)/N \tag{16}$$

$$DPO = DP/(DP + YP) \quad (17)$$
$$YPO = DN/(DN + YN) \quad (19)$$
$$DD = (DPO + YPO)/2 \quad (20)$$

4.2. Deneysel Sonuçlar ve Dengesiz Veri Seti Yaklaşımı

Bu çalışma IDAP konferansında sunulan çalışmanın genişletilmiş halidir [13]. Bildiri çalışmasında aynı veri seti kullanılarak GTB algoritması ile % 95 doğruluk elde edilmiştir. Bu çalışmada ise aynı veri seti için GTB, UKSB ve ESA algoritmaları aynı koşullar altında karşılaştırılmış, hiper parametre iyileştirilmesi ile doğruluk değeri iyileştirilmiş ve dengesiz veri setine özgü yaklaşım geliştirilerek tüm performans metrikleri iyileştirilmiştir.

Kaggle platformunda yayınlanan e-ticaret platformu Hepsiburada kullanıcı yorumları veri seti ile elde edilen deneysel sonuçlara ait karşılaştırma matrisleri ve metrik performans değerleri Tablo 4'te toplu olarak gösterilmiştir. En yüksek doğruluk GTB yöntemi kullanılarak % 95.9 olarak elde edilmiştir. Ancak veri seti % 98 oranında pozitif yorum içeren oldukça dengesiz bir setidir bu nedenle en iyi YPO değeri GTB ile % 69 olarak elde edilebilmiştir. En yüksek DPO ise UKSB kullanılarak % 97.8 olarak elde edilmiştir. Görüntü işleme uygulamalarında dengesiz veri setleri üzerinde baskın sınıfın aşağı örneklenmesi ya da azınlık sınıfın yukarı örneklenmesi ile veri seti dengeli hale getirildiğinde daha başarılı sonuçlar elde edilebilmektedir [29].

Azınlık sınıfın yukarı örneklenmesi bu veri setinde mümkün değildir. Baskın sınıfın aşağı örneklenmesi durumunda ise veri seti boyutu oldukça küçülecek ve model yeterince öğrenemeyecektir.

Bu uygulama için pozitif yorumlarda geçen kelimelerin sıklık frekans ölçümleri yapılarak en sık geçen 20 kelime pozitif yorumlardan çıkarılmıştır. Vektörleştirme işlemi her ne kadar sık geçen kelimeleri düşük ağırlıklarla ağırlıklandırmış olsa da öğrenme modelinin negatif yorumları yeterince öğrenemediği çıkarımı deneysel sonuçlardan elde edilmiştir. Bu bağlamda geliştirilen basit yaklaşım "Denklem 21" de olduğu gibi pozitif yorumları içeren C_p külliyatından en sık geçen 20 kelime olan C_{pn} çıkarılmıştır. Bu yaklaşımla yeniden eğitilen modelde YPO oranı ESA ile % 92 olarak gerçekleşmiştir. Modelin eğitim tur sayısı, kayıp fonksiyonu ve giriş kelime uzunluğu gibi hiper parametreler deneysel gözlemlere dayalı olarak iyileştirilmiştir. Eğitim süreleri göz önüne alındığında en hızlı algoritma ESA, en yavaş algoritma ise UKSB'dir. Tablo 4-6'da elde edilen en iyi sonuçlar okunabilirlik için koyu gösterilmiştir. Tabloda Doğruluk, DPO ve YPO yanı sıra "Denklem 20"'de verilen ve DPO ve YPO değerlerinin toplamının ikiye bölümünden elde edilen dengelenmiş doğruluk değeri (DD) de verilmiştir. Bu bağlamda en iyi DD GTB ile % 83 olarak elde edilmiştir. Dengesiz veri seti için ile önerilen basit yaklaşım ile bu değer ESA ile % 95.5 değerine, DPO % 99 değerine iyileştirilmiştir.

C_{pn}={bir, ve, çok, bu, için, ürün, iyi, tavsiye, güzel, daha, da, ama, gayet, hızlı, ederim, olarak, gibi, ürün, elime}

$$C_p = C_p - C_{pn} \quad (21)$$

Tablo 4. GTB algoritması için deneysel sonuçlar ve karşıtlık matrisleri

Tahmin Edilen Sınıflar		**Gerçek Sınıflar**					
		1-tur		**10-tur**		**16-tur**	
Yöntem		68677	4373	67544	1829	67481	1784
		0	0	1133	2544	1196	2589
GTB	D	.94		.959		.959	
	DPO	.94		.973		.974	
	YPO	-		.691		.684	
	DD	-		.832		.829	
GTB + Dengesiz veri seti yaklaşımı		68308	953	68295	693	68300	762
		369	3420	382	3680	377	3611
	D	.981		**.985**		.984	
	DPO	.986		.989		.988	
	YPO	.90		.905		.905	
	DD	.943		.947		.944	

Tablo 4'te en sık geçen kelimelerin pozitif yorumları içeren veri kümesinden çıkarılmadan ve çıkarıldıktan sonra eğitilen modele ait sonuçları GTB algoritması için verilmiştir. Modelin doğruluk değeri % 95.9'tan % 98.5'e, YPO oranı % 69.1'den % 90'a, DD ise % 83'ten % 94'e iyileştirilmiştir.

Tablo 5. UKSB algoritması için deneysel sonuçlar ve karşıtlık matrisleri

Tahmin Edilen Sınıflar		**Gerçek Sınıflar**					
		1-tur		**10-tur**		**16-tur**	
UKSB		68677	4373	66926	1490	67360	1874
		0	0	1751	2883	1317	2499
	D	.94		.955		.956	
	DPO	.94		.978		.972	
	YPO	-		.622		.654	
	DD	-		.8		.81	
UKSB + Dengesiz veri seti yaklaşımı		68431	1660	68173	677	68133	644
		246	2713	504	3696	544	3729
	D	.973		.983		.983	
	DPO	.976		**.99**		**.99**	
	YPO	.916		.88		.87	
	DD	.946		.935		.93	

Tablo 5'te en sık geçen kelimelerin pozitif yorumları içeren veri kümesinden çıkarılmadan ve çıkarıldıktan sonra eğitilen modele ait sonuçları UKSB algoritması için verilmiştir. UKSB algoritması ile eğitilen modelin doğruluk değeri % 95.5'ten % 98.3'e, YPO oranı % 62.1'den % 88'e, DD ise % 80'den % 93.5'e iyileştirilmiştir.

Tablo 6. ESA algoritması için deneysel sonuçlar ve karşıtlık matrisleri

Tahmin Edilen Sınıflar		**Gerçek Sınıflar**					
		1-tur		**10-tur**		**16-tur**	
ESA		68677	4373	67302	2017	67543	2355
		0	0	1375	2356	1134	2018
	D	.94		.953		.952	
	DPO	.94		.97		.966	
	YPO	-		.631		.64	
	DD	-		.8		.8	
ESA + Dengesiz veri seti yaklaşımı		68583	2411	67996	646	68386	874
		94	1962	681	3727	291	3499
	D	.965		.981		.984	
	DPO	.966		**.99**		.987	
	YPO	**.954**		.845		.923	
	DD	.96		.917		**.955**	

Tablo 6'da en sık geçen kelimelerin pozitif yorumları içeren veri kümesinden çıkarılmadan ve çıkarıldıktan sonra eğitilen modele ait sonuçları ESA algoritması için verilmiştir. ESA algoritması ile eğitilen modelin doğruluk değeri % 95.3'ten % 98.1'e, YPO oranı % 63.1'den % 84.5'e, DD ise % 80'den % 91.7'ye iyileştirilmiştir

5. Sonuçlar ve Gelecek Çalışmalar

Bu çalışmada metin sınıflandırma problemi için aynı koşullar altında ESA, GTB ve UKSB derin öğrenme algoritmalarının performansları karşılaştırılmıştır. Çalışmada kullanılan veri seti % 98 olumlu, % 2 olumsuz yorum içeren oldukça dengesiz bir setidir, bu yönü ile model makine öğrenmesi ile eğitilmeden tüm yorumların olumlu olarak etiketlenmesi durumunda bile % 98 doğrulukla çalışacak ancak bu durumda ürünlerden memnun olmayan müşteri kitlesi ve şikâyet edilen ürünler tespit edilememiş olunacaktır. Görüntü işleme uygulamalarında dengesiz veri seti içeren çalışmalarda baskın sınıfın aşağı örneklenmesi prensibine dayanan basit bir yaklaşım bu çalışmada kullanılmış ve önerilen yaklaşım günümüzde yaygın olarak kullanılan üç farklı derin öğrenme algoritması ile test edilmiştir. Önerilen yaklaşımla baskın sınıfta en sık geçen kelimeler eğitim veri setinden çıkarılarak, azınlık sınıfa ait örneklerin daha iyi tanınması amaçlanmıştır. Yaklaşımın test edildiği üç algoritmada da sınıflandırma değerlerinin iyileştiği görülmüştür. Özellikle olumsuz yorumların doğru sınıflandırılmasını gösteren YPO metrik değeri oldukça tatmin edici ölçüde iyileştirilmiştir. Sonuç olarak dengesiz veri setine sahip duygu sınıflandırma problemleri için baskın sınıfın aşağı örneklenmesinin sınıflandırma metrik değerlerini iyileştirdiği deneysel çalışmalar yapılarak görülmüştür. Gelecekte dengesiz metin veri seti içindeki azınlık sınıfları etkin öğrenecek çekişmeli üretici derin öğrenme algoritmaları kullanan yaklaşımların geliştirilmesi planlanmaktadır.

Teşekkür

Bu çalışmada Kaggle platformunda yayınlanan e-ticaret platformu Hepsiburada veri seti kullanılmıştır.

Kaynaklar

[1] Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P. Natural language processing (almost) from scratch. Journal of machine learning research 2011; 12, 2493-2537.
[2] Young T, Hazarika D, Poria S, Cambria, E. Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine 2018; 13(3), 55-75.
[3] Goldberg Y. Neural network methods for natural language processing. Synthesis Lectures on Human Language Technologies 2017; 10(1), 1-309.
[4] Zhuang H, Wang C, Li C, Wang Q, Zhou, X. Natural language processing service based on stroke-level convolutional networks for Chinese text classification. In 2017 IEEE Int. Conf. on Web Services, 25-30 June, HI, USA, pp. 404-411.
[5] Young T, Hazarika D, Poria S, Cambria E. Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine 2018; 13(3), 55-75.
[6] Feldman, R. Techniques and applications for sentiment analysis. Communications of the ACM, 56(4), 82-89.
[7] Neri, F, Aliprandi, C, Capeci, F, Cuadros, M, By, T. Sentiment Analysis on Social Media. ASONAM, 12, 919-926.
[8] Alaei, A. R, Becken, S, Stantic, B. Sentiment analysis in tourism: capitalizing on big data. Journal of Travel Research, 58(2), 175-191.
[9] Abualigah, L, Alfar, H. E, Shehab, M, Hussein, A. M. A. Sentiment Analysis in Healthcare: A Brief Review. In Recent Advances in NLP: The Case of Arabic Language, (pp. 129-141). Springer, Cham.
[10] Tubishat, M, Abushariah, M. A, Idris, N, Aljarah, I. Improved whale optimization algorithm for feature selection in Arabic sentiment analysis. Applied Intelligence, 49(5), 1688-1707.
[11] Yousif, A, Niu, Z, Tarus, J. K, Ahmad, A. A survey on sentiment analysis of scientific citations. Artificial Intelligence Review, 52(3), 1805-1838.
[12] Derakhshan, A, Beigy, H. Sentiment analysis on stock social media for stock price movement prediction. Engineering Applications of Artificial Intelligence, 85, 569-578.
[13] Santur Y. Sentiment Analysis Based on Gated Recurrent Unit. In 2019 International Artificial Intelligence and Data Processing Symposium (IDAP),21-22 Sep. Malatya, Turkey, pp. 1-5.
[14] Kulkarni A, Shivananda A. Converting text to features. In Natural Language Processing Recipes, Apress, Berkeley, CA. 2019; 67-96.
[15] Arroyo-Fernández I, Méndez-Cruz C. F, Sierra G, Torres-Moreno J. M, Sidorov, G. Unsupervised sentence representations as word information series: Revisiting TF–IDF. Computer Speech & Language 2019; 56, 107-129.
[16] Song Y. MIHNet: Combining N-gram, Sequential and Global Information for Text Classification. In Journal of Physics: Conference Series 2020; 1453, 012156.
[17] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv preprint arXiv:2013; 1301.3781.
[18] Guo D, Wang Q, Liang M, Liu W, Nie, J. Molecular Cavity Topological Representation for Pattern Analysis: A NLP Analogy-Based Word2Vec Method. International Journal of Molecular Sciences 2019; 20(23), 6019.
[19] Joulin A, Grave E, Bojanowski P, Douze M, Jégou H, Mikolov T.Fasttext. zip: Compressing text classification models. arXiv preprint arXiv:2016; 1612.03651.
[20] Pennington J, Socher R, Manning CD. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 25-29 Oct., Doha, Qatar, pp.1532-1543.
[21] LeCun Y, Bengio Y, Hinton G. Deep learning. Natüre 2015; 521(7553), 436-444.
[22] He T, Huang W, Qiao Y, Yao J. Text-attentional convolutional neural network for scene text detection. IEEE transactions on image processing 2016; 25(6), 2529-2541.
[23] Lukoševičius M, Jaeger H. Reservoir computing approaches to recurrent neural network training. Computer Science Review 20019; 3(3), 127-149.
[24] Song S, Huang H, Ruan T. Abstractive text summarization using LSTM-CNN based deep learning. Multimedia Tools and Applications 2019; 78(1), 857-875.
[25] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation 1997; 9(8), 1735-1780.
[26] Luo LX. Network text sentiment analysis method combining LDA text representation and GRU-CNN. Personal and Ubiquitous Computing 2019; 23(3-4), 405-412.
[27] Kim HY, Won CH. Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. Expert Systems with Applications 2018; 103, 25-37.
[28] Hepsiburada kullanıcı yorumları [2018], Available: https://www.kaggle.com/bulentsiyah/hepsi-burada-yorum.
[29] Segura-Bedmar I, Colón-Ruíz C, Tejedor-Alonso MÁ, Moro-Moro M. Predicting of anaphylaxis in big data EMR by exploring machine learning approaches. Journal of biomedical informatics 2018; 87, 50-59.