VERİ BİLİMİ DERGİSİ
www.dergipark.gov.tr/veri

Ses Özniteliklerini Kullanan Ses Duygu Durum Sınıflandırma İçin Derin Öğrenme Tabanlı Bir Yazılımsal Araç

Emir Ali KIVRAK[1]*, Bahadır KARASULU[1], Can SÖZBİR[1], Atakan TÜRKAY[1]

[1]*Çanakkale Onsekiz Mart Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Çanakkale, TÜRKİYE*

Makale Bilgisi

Başvuru:
06/07/2021
Kabul:
17/10/2021

Özet

Ses duygu durum analizi için kullanıcı grafik arayüzü yardımıyla ses verilerini kullanarak ses duygu durumları herhangi bir kaynak kodu satırı yazmadan sınıflandıran derin öğrenme mimari modellerini oluşturan bir yazılımsal araç çalışmamızda tasarlanmıştır. Veri kümelerinin elde edilmesi, ses verilerine yönelik ses özniteliklerinin elde edilmesi, mimarinin oluşturulması ve derin öğrenme modelinin istenilen sinir ağı katmanları ve üstün parametreler ile modelin eğitilmesi sağlanmıştır. Model eğitilirken, eğitim değerlerinin gerçek zamanlı izlenmesi yazılımsal araç ile yapılabilmektedir. Çalışma boyunca, ilgili adımlar hem salt kaynak kodu düzenleme hem de yazılımsal araç kullanılarak gerçekleştirilmiştir. Kod düzenleme tabanlı melez model, mimarisinde uzun kısa süreli bellek ve evrişimli sinir ağları kullanılarak oluşturulmuş, %81,49 doğruluk oranına ulaşmıştır. Ayrıca, herhangi bir kodlama müdahalesi olmaksızın grafik yazılımsal araç tabanlı tekil model, mimarisinde evrişimli sinir ağı ile oluşturulmuştur. Böylece %75,76 doğruluk oranına ulaşmıştır. Yazılımsal aracın geliştirilmesindeki ana motivasyon, farklı ses duygu durumları sınıflandırmak için kullanılabilecek potansiyel bir derin öğrenme mimari modeli oluşturmaktır. Deneysel sonuçlar, yazılımsal aracın yüksek doğrulukla sınıflandırmayı oldukça başarılı bir şekilde gerçekleştirdiğini kanıtlamaktadır. Elde edilen sonuçlara dair tartışmaya da çalışmamızda yer verilmiştir.

Anahtar Kelimeler: *Ses duygu analizi, duygu durum, derin öğrenme, yazılımsal araç*

A Deep Learning based Software Tool for Audio Emotional State Classification using Audio Features

Abstract

For audio emotional state analysis, a software tool was designed in our study that build deep learning architectural models that classify audio emotional states using audio data with the help of the user graphical interface without writing any line of source codes. Obtaining the desired data sets and audio features for audio data, creating the architecture and training the model with the desired neural network layers and hyperparameters of deep learning model were provided. While the model is being trained, real-time monitoring of training values can be performed over the software tool. Throughout the study, the relevant steps were carried out using both pure source code editing and software tool. The code editing based hybrid

1* İletişim e-posta: emirccus@gmail.com
** Bu çalışmanın bir kısmı IV. International Conference on Data Science and Applications 2021'de sözlü olarak sunulmuştur.

model built with long short-term memory and convolutional neural networks in its architecture that achieved an accuracy rate of 81.49%. In addition, the graphical software tool based standalone model without any coding intervention was built with convolutional neural network in its architecture. Thence, it achieved 75.76% accuracy rate. The main motivation in the development of software tool is to build a potential deep learning architectural model that can be used to classify different audio emotional states. Experimental results prove that the software tool performs classification with high accuracy quite successfully. The discussion on the results obtained is included in our study.

Keywords: *Audio emotion analysis, emotional state, deep learning, software tool*

1 Giriş

Duygu durum analizi (emotional state analysis), insanların belirli bir süre zarfında, belirli bir varlığa karşı içinde bulunduğu duyguyu ya da fikri tahmin etme veya sınıflandırma üzerine bir hesaplamalı çalışma alanıdır. Internet 2.0 ve sosyal medyanın yükselişi ile kullanıcılar tarafından üretilen veri boyutu muazzam büyüklüklere ulaşmış ve bu ham veriden insanların duygularına yönelik anlamlı bilgi elde etme isteği, toplum ve işletmeler için önemli bir konu haline gelmiştir. Duygu durum analizi, sosyal medya uygulamaları, müşteri hizmetleri servisleri, e-ticaret platformları ve birçok farklı alanda kullanılmakta ve hızla gelişmekte olan bir alandır [1, 2]. İnsan duygu durum analizi, hesaplamalı dilbilimleri, anlambilim, yapay zeka ve makine öğrenimi gibi bir çok alan ile birlikte çalıştığından çok disiplinli bir çalışma alanıdır. Duygu analizi sözlük bazlı ve makine öğrenimi bazlı olmak üzere iki farklı yöntem ile gerçekleştirilebilir. Makine öğrenimi bazlı duygu durum analizi kullanılacak olan etiketli veri kümelerinden farklı yöntemler ile özniteliklerin elde edilmesi ve farklı makine öğrenimi yöntemleri kullanılarak bir model oluşturmaktır. Makine öğrenimi yaklaşımının denetimli, yarı denetimli ve denetimsiz olmak üzere üç farklı kategorisi bulunmaktadır. Makine öğrenimi yöntemi otomasyon yeteneğine sahip olması ve büyük verileri işleyebildiğinden dolayı duygu analizi için iyi bir tercihtir. Destek vektör makineleri (Support vector machine - SVM), saf Bayesçi yaklaşımlar, gizli Markov modeli gibi hem denetimli hem de denetimsiz makine öğrenimi gibi derin öğrenme de makine öğrenimi sürecinin yapay sinir ağları kullanılarak oluşturulmasını ifade eder. Yapay sinir ağı, insan beyninin yapısından esinlenilerek geliştirilmiştir. Denetimli ve denetimsiz makine öğrenimi, sinir ağları kullanılarak gerçekleştirilebilmektedir [1, 3]. Derin öğrenme günümüzde güçlü grafik kartlarının (Graphic Processing Unit - GPU) kolay ulaşılabilir olması ve makine öğrenimi algoritmaların gelişmesi nedeniyle önem kazanmıştır. Bundan dolayı çalışmalarda sıklıkla tercih edilmektedir. Derin öğrenmenin kendi kendine ve daha az veri kümesi ile öğrenme yeteneği derin öğrenmeyi geleneksel makine öğrenimi algoritmalarından daha başarılı kılmaktadır. Bundan dolayı bilgisayarlı görü, konuşma tanıma, doğal dil işleme ve duygu analizi gibi birçok konuda kullanılmaktadır [3, 4].

Bu makale şu şekilde organize edilmiştir. Makalenin ikinci bölümünde kullanılan veri kümeleri ve benzer çalışmalardan bahsedilmektedir. Üçüncü bölümde çalışmanın gerçekleştirilmesinde kullanılan metot ve materyalden, dördüncü bölümünde ise programatik olarak kodlanarak ve geliştirilen yazılımsal aracın arayüzü ile oluşturulmuş iki farklı derin öğrenme modelinin oluşturulma aşamalarından bahsedilmiş ve karşılaştırılması yapılmıştır. Sonuçlar bölümünde ise yapılan çalışmaya dair bilimsel bulgular değerlendirilmektedir.

2 Literatür İncelemesi

Çalışmalar incelendiğinde, Gizli Markov modeli, SVM, karar ağaçları gibi klasik makine öğrenimi yöntemleri duygu durum tanıma problemlerinin çözümü için uygulanmış olduğu görülmektedir [5, 6, 7]. Ses işleme ile elde edilen öznitelikler Evrişimli Sinir Ağı (Convolutional Neural Network-CNN) ve Uzun Kısa Süreli Bellek Ağları (Long Short Term Memory-LSTM) tabanlı sinir ağı modellerinde kullanılmıştır [8, 9, 10]. Sinir ağları kullanılarak oluşturulan bu modellerin klasik makine öğrenimi modelleri ile kıyaslandığında daha büyük bir başarım elde ettiği görülmüştür. Ses duygu durum tanıma araştırmaları için hazırlanmış, araştırma ve geliştirme topluluklarının faydalanması amacıyla kamuya açılmış farklı veri kümeleri bulunmaktadır. Sinir ağını beslemek için bu veri kümeleri oldukça

önemlidir. Bazı veri kümeleri profesyonel aktörler ile stüdyo şartlarında hazırlanmışken, bazıları ise videolardan ve ses kayıtlarından kesitler alınarak hazırlanmıştır. Örnek olarak RAVDESS veri kümesi ses aktörleri ile stüdyo şartlarında hazırlanması ve her bir ses kaydının 247 kişiden oluşan bağımsız bir jüri tarafından içerdikleri duygu durum doğruluğunu değerlendirmek amacı ile 10 kere test edilmesinden dolayı araştırmacılar için güvenilir ve profesyonel bir veri kümesidir [11]. CREMA-D veri kümesi 91 İngilizce konuşan aktör ile hazırlanmıştır. Aktör sayısının diğer veri kümelerinden fazla oluşu onu bilgisayar ile yapılan öğrenme uygulamaları için cazip hale getirmektedir [12]. Berlin EMO-DB veri kümesi 5 kadın ve 5 erkek aktör ile kaydedilmiş olup toplamda 535 kayıt alınmıştır. Kullanılan diğer veri kümelerinden farkı Almanca olmasıdır [13]. Surrey Audio-Visual Expressed Emotion (SAVEE) veri kümesi 4 erkek aktör ile 7 farklı duygu içerir. Toplamda 480 İngilizce kayıttan oluşmaktadır[14]. Çalışmamızda önerilen yazılımsal araç ile benzer bir mantıkta çalışan Google Teachable Machine [15] kullanıcılara kendi veri kümeleri ile bir makine öğrenimi modelini eğitebilecekleri temel bir platform sunmaktadır. Kullanıcı sadece, eğitim yenilenme sayısı (epoch), yığın büyüklüğü (batch size), öğrenme oranı (learning rate) üstün parametrelerini (hyperparameter) kendi belirledikleri değerlere göre değiştirebilmektedir. Önerilen yazılımsal araç, daha düşük seviyeli bir süreç sunmakta olup, modelin mimarisinin oluşturulması ve özniteliklerin seçimi kullanıcıdan beklenmektedir.

3 Metot ve Materyal

Öznitelik elde etme işlemi, ham veriye çeşitli işlemler uygulanması ile yapılır [16]. Ham ses verisi üzerinden mel frekans kepstrum katsayıları (Mel Frequency Cepstral Coefficients - MFCC), mel spektrogramı, harmonik adım sınıf profilleri (chromogram), zıtlık (contrast), tonnetz ve MFCC_delta gibi öznitelikler elde edilebilmektedir. Bu özniteliklerin verinin hangi özelliklerini temsil ettiği aşağıda verilmiştir:

- ✓ *MFCC:* Ses sinyalinin kısa zamanlı güç spektrumunun insan kulağının ses frekanslarındaki değişimi algılayışını gösteren mel ölçeği üzerindeki ifadesidir.
- ✓ *Mel spektrogramı:* Ses frekansların mel ölçeğine çevrildiği bir spektrogramdır.
- ✓ Harmonik adım sınıf profilleri*:* Spektrum müzikal oktavının 12 farklı yarı tonunu (chroma) temsil eden 12 parçanın belirtildiği ses için güçlü bir sunumudur.
- ✓ *Zıtlık:* Ses sinyalindeki tepeler ve çukurların arasındaki farkların seviyeleridir.
- ✓ *Tonnetz:* Belirli bir ses sinyalinin harmonik içeriğini içerir.
- ✓ *MFCC_delta:* Seçilen eksen boyunca girdi MFCC verilerinin türevinin yerel tahmini olarak belirlenir.

3.1 Kullanılan Derin Öğrenme Metotları

Geliştirilen yazılımsal araçta desteklenen ve ses verisi üzerinde sıklıkla kullanılan katman yapıları CNN, azami birikme (max-pooling), iletim sönümü (dropout), yığın normalizasyonu (batch normalization) ve yoğun (dense) isimli katmanlar olarak sıralanabilir. Bu katmanların yanı sıra kıyaslama amacıyla oluşturulmuş modelde LSTM katmanı da kullanılmıştır, kullanılan katmanların açıklamaları aşağıda verilmektedir:

- ✓ *1 Boyutlu Evrişim (Conv1D):* Tek boyutlu bir evrişim çekirdeği oluşturur. Bu çekirdeğin sahip olduğu ağırlık değerleri girdi matrisi ile çarpılarak filtre olarak uygulanmış ve girdiler özetlenmiş olur [17].
- ✓ *LSTM:* Uzun kısa süreli bellek katmanı, uzun vadeli bağımlılıkları öğrenebilen bir çeşit özyinelemeli katman çeşitidir. Klasik tekrarlayan (recurrent) sinir ağlarından farklı olarak kaybolan eğim (vanishing gradients) problemini gideren bir çeşit hafızaya sahiptirler [17, 18].
- ✓ *Azami biriktirme (Max-pooling):* Bu katmanda öğrenilen bir parametre yoktur. Girdi olarak alınan matrisin belirli aralıklarla oluşturulan alt matrislerinden en büyük değere sahip olan değer bu alt matrisin yerine konulur [17].
- ✓ *İletim sönümü (Dropout):* Önceden belirlenmiş olasılık ile ağdaki bazı düğümlerin rastgele şekilde kaldırılması tekniğidir [19].
- ✓ *Yığın normalizayonu (Batch normalization):* Yığın normalleştirme, katman girdilerini ortalamaları sıfır değerinde olacak şekilde *[0,1]* aralığına ölçeklemektedir. Bu işlem eğitimin daha etkin gerçekleştirmesine olanak sunar [20, 21].
- ✓ *Yoğun (Dense):* En temel katmanlardan biridir. Birbirleri ile sık bağlantılı birimlerden oluşur. Genellikle sinir ağlarının çıktı katmanları ve

- ✓ öncesindeki birkaç katman bu katman tipinden seçilir [17].

Çalışmamız boyunca ses verisi üzerinden bazı öznitelikler elde edilerek öznitelik vektörleri oluşturulup bu vektörler üzerinden model eğitimi ve sınıflandırma (tahminleme) yapılmıştır. Bu vektörler belirli formüller sonucu hesaplanmaktadır. Bu nedenle, el yapımı (handcrafted) öznitelikler olarak adlandırılırlar. Derin öğrenme sinir ağı katmanları da bu düşük seviye bilgiler içeren vektörler üzerinden çok daha soyut bilgiler elde etmekte ve ileriki katmanlara aktarmaktadır. Bilgilerin bu şekilde aktarılması ise temsili öğrenme (representative learning) olarak adlandırılmaktadır. Dolayısıyla bu çalışmada hem el yapımı (handcrafted) öznitelikler hem de temsili öğrenme (representative learning) ile elde edilen soyut öznitelikler (abstract features) kullanılmıştır [22]. Eğitim için kullanılacak verilerin hazırlanmasında veri kümesinin genişletilip eğitim başarımının iyileştirilmesi için veri artırma (data augmentation) teknikleri kullanılmıştır. Bu veri artırma teknikleri;

- ✓ *Beyaz gürültü (White noise):* Ses verisine farklı frekanslarda eşit yoğunluklu gürültü eklenmesidir. Çalışmamızda, ses frekans değerini sesin azami frekans aralığından seçilen rasgele bir sayının 0,05 katı ile toplayarak gürültülü ses verisi oluşturulmaktadır.
- ✓ *Kaydırma (Shift):* Ses verisinin belirli bir yönde kaydırılma işlemidir. Çalışmamızda, ses verisi ile oluşturulan vektör zaman ekseninde ileri yönde 1 saniye kadar kaydırılmıştır, Bu işlem sonucunda oluşturulan vektörün sonundan kaydırma nedeniyle taşan ses verileri bu vektörün başına eklenmektedir.
- ✓ *Perde değiştirme (Pitch changing):* Ses sinyalinin frekans ölçeğine bağlı algısal derecesini belirleyen perde (pitch) değerinin bir miktar değiştirilmesidir. Çalışmamızda, sesin dalga formunun perdesi bant sınırlamalı enterpolasyon yoluyla yüksek kaliteli bir yöntemle 12 adımda yarım ton kadar kaydırılmıştır.
- ✓ *Hız değiştirme veya Esnetme (Speed changing-Stretching):* Ses verisinin zaman ekseninde sabit bir oranda gerdirilmesi veya sıkıştırılmasıdır. Çalışmamızda kullanın yazılımsal araç, eğer hız değiştirme parametresi için 1'den büyük değerler verilirse ses hızını artırmakta, küçük değerler verilirse ses hızını yavaşlatmaktadır. Deneylerde, ses verisine hız değiştirme işlemi 1,1 değeri verilmesiyle %10 oranında hızlandırılarak uygulanmıştır.

3.2 Yazılımsal Araç

Yazılımsal araç kendi başına, kod yazma işlemi gerçekleştirilmeden, ses ile duygu durum analizi yapabilen bir derin öğrenme modeli oluşturmayı ve aracın geri bildirimleri sayesinde oluşturulan modelin üstün parametrelerinin eniyilenmesini hedeflemektedir. Bunun için grafik kullanıcı arayüzü (Graphical User Interface - GUI) oluşturulmuştur. Buna dair detaylar bu bölümde verilmektedir.

3.2.1 Yazılımsal Araç İskeleti ve Kütüphaneler

Yazılımsal araç için arayüzün gerçekleştirildiği platform olarak Web tercih edilmiştir. Bölümün kalan kısmında yazılımsal araç için kullanılan yazılım iskeleti ve kütüphaneler açıklanacaktır. Sistemin geliştirilmesi boyunca yazılım dili olarak *Python* kullanılmıştır. *Python* dilinin *versiyon 3* altyapısı ile sistemin tasarlanması için seçilmesinin en büyük sebebi, makine öğrenmesi ve veri işleme alanında birazdan bahsedilecek olan birçok güçlü iskelet (*framework*) yapısı sunması ve buna ek olarak yüksek seviyeli bir programlama dili olmasının getirdiği okunabilirlik ve daha az karmaşıklıktır [23]. Yapay sinir ağı Uygulama Programlama Arayüzü (Application Programming Interface - API) olarak *Keras* kullanılmıştır [17]. *Keras* versiyon 2.4.3 açık kaynak kodlu Python ile yazılmış, geliştirici dostu bir API'dir. TensorFlow, Theno ve CNTK gibi arka uçların (backend) üstünde çalışmaktadır. Bu sebeple kendi başına bir derin öğrenme platformu değildir [17]. Ses sinyallerinin elde edilmesi, ses verisinin ön işlemesi, ses verisinden özniteliklerin elde edilmesi gibi birçok ses verisine dayalı işlemi gerçekleştirebilmek için *Librosa* kullanılmıştır. Librosa versiyon 0.8.0, müzik ve ses analizi için geliştirilmiş bir Python paketidir [24]. Web ortamının geliştirilmesinde, yazılan kütüphaneler ile etkileşimin daha hızlı ve kolay kodlanabilir olması amacıyla Flask kullanılmıştır. Flask, Python için Web iskeletidir. Boyutu ve getirdiği minimum bağımlılık sayesinde hızlı bir şekilde Web uygulaması oluşturmayı sağlamaktadır. En önemli bağımlılıkları Werkzeug, Jinja2 ve Click şablonları ve eklentilerine olmaktadır [25, 26, 27, 28]. Veri tabanı olarak özlü biçimde ve tek dosya üzerinden çalışabilmesi, kurulum ve konfigürasyon işlemlerinin vakit almaması gibi avantajlarından dolayı *SQLite* kullanılmıştır. SQlite, sunucusuz, konfigürasyon

gerektirmeyen, işlemsel bir SQL veri tabanı motoru uygulayan bir süreç içi kitaplıktır. SQlite dünyada en yaygın kullanılan veri tabanlarından biridir [29]. Veri tabanı üzerinde yapılacak işlemlerde Nesne İlişkisel Eşleştirici (Object-Relational Mapper - ORM) yapısı kullanılması için *SQLalchemy* kütüphanesi kullanılmıştır [30]. Derin öğrenme aşamasındaki sürecin görüntülenebilmesi için *Tensorboard* kullanılmıştır. *Tensorboard*, makine öğrenimi deneyleri için gereken görselleştirme araçlarını sağlar. Bu araçlar ile kayıp ve doğruluk değerlerine dair plot çizimleri, kullanılan derin sinir ağı modeli ile ilgili grafiklerin görselleştirilmesi, zamanla değişen ağırlıklar ve tahminlemenin görselleştirilmesi gibi olgular Web arayüzü üzerinden sunulmaktadır [31]. Ön yüz (frontend) için *HTML*, *CSS*, *JavaScrip*t dilleri kullanılırken sade ve kullanışlılık açısından "*sbadmin-2"* adlı açık kaynak lisanslı bir kontrol paneli kullanılmıştır [32]. Kullanıldığı belirtilen araçlar ve kütüphaneler ile oluşturulan yazılımın yetenekleri bir sonraki bölümde açıklanmaktadır.

3.2.2 Önerilen Yazılımsal Aracın Yetenekleri

Yazılımsal araç, bir duygu durum sınıflandırıcı derin öğrenme modeli oluşturmak için gereken yeteneklere sahiptir. Yazılımsal araç tamamen arayüz üzerinden kontrol edilmesine rağmen standart çıktıya da anlık olarak geri bildirim vermektedir. Aşağıda yazılımsal aracın yeteneklerine maddeler halinde değinilmektedir.

Modelin eğitimi ve test edilmesi için gereken veri kümelerinin elde edilmesi: Arayüz aracılığı ile çevrimiçi olarak sistem tarafından tanınan veri kümeleri uzak bir sunucudan yerel depolama alanına indirilebilir. İndirilen veri kümeleri sistem tarafından arşivden çıkartılıp, kullanım için gereken yerlere kopyalanır. Her bir veri kümesi için dosya isimlerine bakılarak, her bir ses dosyasının hangi veri kümesine ait olduğu, hangi duyguyu barındırdığı ve hangi cinsiyete ait olduğu bilgilerini içeren bir üst veri (meta data) oluşturulur ve veri tabanına kaydedilir. Böylelikle bu işlemlerin tekrarlanması gerekmez.

İstenilen verilerden istenilen özniteliklerin elde edilmesi: Yazılımsal araç kullanılarak, her bir ses verisi üzerinden MFCC, diferansiyel MFCC, Mel ölçekli spektogram, harmonik adım sınıf profilleri, tonnetz, spektral zıtlık değerleri elde edilebilir. İstenilen öznitelikler elde edildikten sonra, her bir ses dosyası için üretilen veri, uç uca eklenir ve sinir ağına yollamaya uygun bir dizi oluşturur. Özniteliklerin elde edilmesinde gereken parametrelerin bir kısmı sistem tarafından varsayılan olarak verilmekte bazılarının ise kullanıcı tarafından girilmesi beklenmektedir. Çalışmamızdaki deneylerde yazılımsal araç sayesinde MFCC, Mel ölçekli spektogram, harmonik adım sınıf profilleri, zıtlık, tonnetz, MFCC_delta özniteliklerini içeren öznitelik vektörleri kullanılmıştır. Deneylerde model başarım kıyaslamaları bu öznitelikler üzerinden yapılabilmektedir.

Veri artırımı ile veri kümesinin genişletilmesi: Veri kümelerinin yetersiz kaldığı düşünüldüğü durumlar için yazılımsal araç kullanılarak, sesin hızı artırılıp azaltılabilir, ses kaydırılabilir, hızı değiştirilebilir veya sese gürültü eklenebilir.

Derin öğrenme modeli mimarisi oluşturulabilir: Model mimarisi oluşturulurken, *keras.layers* altında bulunan katmanlar kullanılabilmektedir. Kullanılabilecek katmanlar; 1 Boyutlu evrişim (conv_1d), 1 Boyutlu azami biriktirme (max_pooling_1d), yoğun (dense), iletim sönümü (dropout), yığın normalizayonu (batch normalization) ve düzleştirme (flatten) katmanlarıdır. Bütün bu katmanlar sonucunda bir adet *keras.sequential* modeli oluşturulmaktadır.

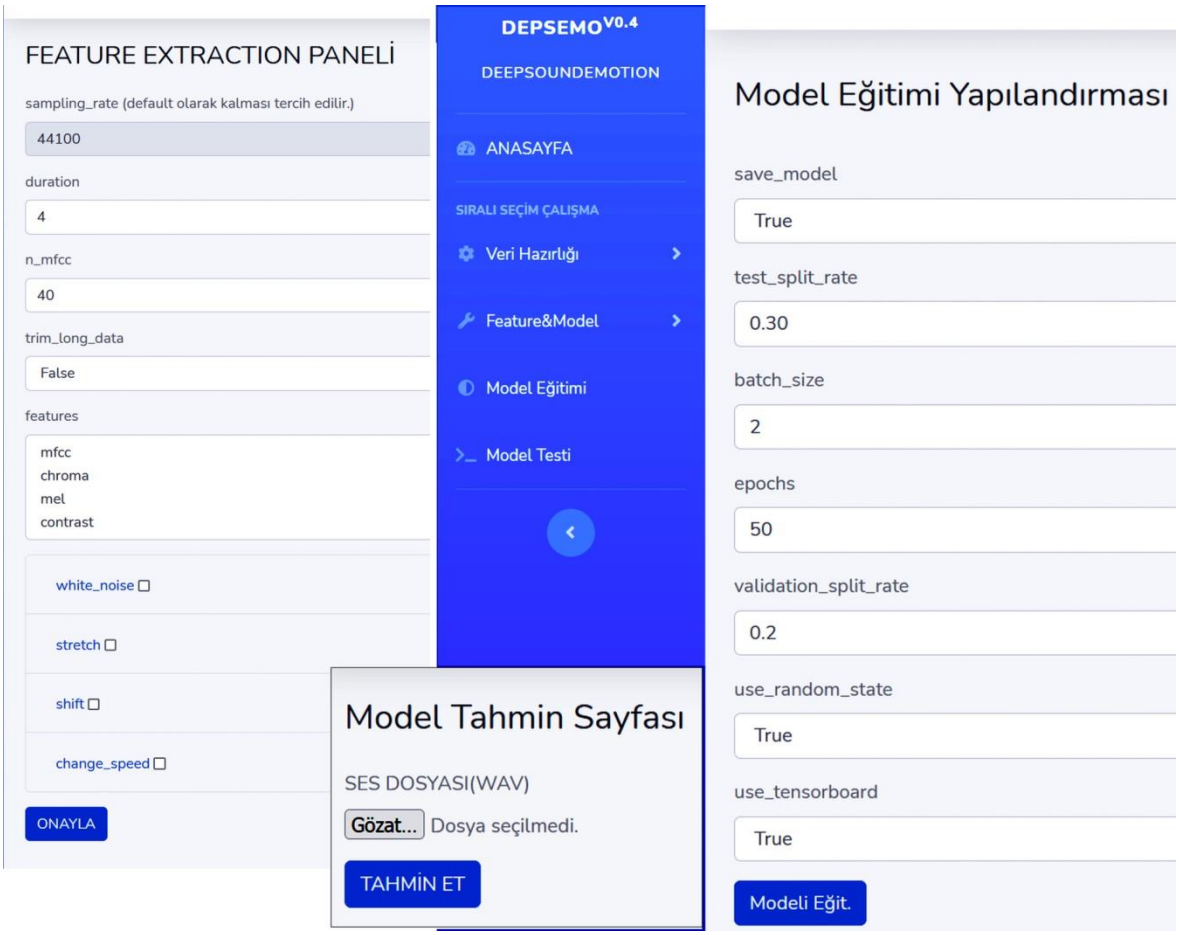
Derin öğrenme modeli derlenip eğitilebilir: Yazılımsal araç ile oluşturulan model derleme ve eğitim ile ilgili parametreleri kullanıcıdan aldıktan sonra eğitim aşamasına geçilir. Eğitim süreci sinir ağının yapısı, parametreler ve eğitim grafikleri sunan *Tensorboard* aracı ile izlenebilmektedir [31].

Derin öğrenme modeli test edilebilir: Oluşturulan modeller araç kullanılarak hem test için ayrılan veri kümesi ile hem de sisteme farklı ses dosyaları yüklenerek test edilebilir. Bunun yanında yazılımsal araç kullanılarak, test veri kümesi için karmaşıklık matrisi (confusion matrix), Alıcı İşletim Karakteristik (Receiver Operating Characteristic - ROC) eğrisi [33] gibi başarım sonuçlarını gösteren plot çizimlerine ve grafiklere de erişebilir [34].

3.2.3 Yazılımsal Araç İçin Geliştirilen Modüller

Yazılım aracına modülerlik kazandırmak ve daha anlaşılabilir bir kod yazmak amacı ile yazılım modüller halinde geliştirilmiştir. Geliştirilen modüllerin amacı, arayüz kısmından alınan bilgilerin ve isteklerin işlenerek kullanıcıya vaat

edilen işlemleri tamamlamaktır. *DatasetExplorer* modülü, veri kümelerinin indirilmesi ve organizasyonunu sağlar. *MetaDataCreator* modülü, veri kümelerindeki her bir ses dosyası için ses dosyası etiketlerini çözer ve *Metaveri* oluşturur. *FeatureExtractor* modülü, özniteliklerin elde edilmesini ve kaydedilmesini sağlar. *DataAugmentator* modülü, Veri artırımı metotlarını barındırır. *ModelBuilder* modülü, derin öğrenme modelinin oluşturulması ve derlenmesinden sorumludur. *ModelTrainer* modülü, derin öğrenme modelinin eğitilmesi ve test edilmesinden sorumludur. Aşağıdaki Şekil 1'de yazılımsal aracın kullanıcı arayüzüne dair modüllerinin kullanımını sağlayan panellerinin ekran görüntüleri kolaj olarak sunulmaktadır. Buna göre, öznitelik elde etme (feature extraction) paneli, model eğitimi yapılandırması paneli ve model ile tahminleme paneli Şekil 1'de görülmektedir. Yazılımsal aracın kendisine (program kaynak kodu) ve kullanıcı arayüzüne ilgili Web sitesinin [35] üzerinden ulaşılabilmektedir.



Şekil 1. Yazılımsal aracın kullanıcı arayüzü ile ulaşılan panellerinin ekran görüntüsü

4 Deneysel Sonuçlar

Çalışmada hem programatik olarak kodlama ile hem de yazılım aracı (grafik arayüz) kullanılarak modeller oluşturulmuş ve bu modellerin başarımları incelenmiştir. Oluşturulan modeller girdi olarak aldığı ses verilerine karşılık sınıflandırma sonucunda duygu durum olarak "*doğal*" (neutral), "*mutlu*" (happy), "*üzgün*" (sad), "*kızgın*" (angry), "*korkmuş*" (fear), "*iğrenmiş*" (disgust), "*şaşırmış*" (suprise) ve "*sıkılmış*" (bored) olmak üzere 8 farklı sınıftan birisini çıktı olarak verecek şekilde eğitilmiştir.

4.1 Yazılımsal araç ile veri kümesinin oluşturulması

Çalışmada oluşturulan derin sinir ağı modelleri RAVDESS [11], CREMA-D [12], Berlin (EMO-DB) [13] ve SAVEE [14] veri kümelerinin birleştirilmesiyle oluşturulmuş bir büyük veri

kümesi üzerinde eğitilmiştir. Tablo 1'de yukarıda bahsi geçen veri kümelerinin birleştirilmesi ve veri artırma yöntemi sayesinde oluşturularak, deneylerde üzerinde sınıflandırma işlemi yapılan büyük veri kümesindeki 8 farklı sınıf etiketi için hangi veri kümesinin kaç adet ses verisinin (örnek sayısı) veri artırma sonucu bu büyük kümeye dahil edildiği gösterilmektedir. Her veri kümesinde eşit sayıda sınıf etiketi mevcut olmasa da toplamda 8 sınıf etiketi bulunacak şekilde bir büyük veri kümesi oluşturarak bir birleştirmeye gidilmiştir.

Tablo 1. Deneylerdeki sınıf etiketleri için ses verisi (örnek) sayıları

	RAVDESS	CREMA-D	EMO-DB	SAVEE
Doğal	576	2096	78	240
Mutlu	384	2542	142	120
Üzgün	384	2542	124	120
Kızgın	384	2542	254	120
Korkmuş	384	2542	138	120
İğrenmiş	384	2542	92	120
Şaşırmış	384	0	0	120
Sıkılmış	0	0	162	0

Eğitim öncesinde veri kümesi *beyaz gürültü* (white noise), *kaydırma* (shift), *esnetme* (stretch) ve *perde değiştirme* (pitch changing) yöntemleri sayesinde veri artırma işleminden geçirilmiştir. Kullanılan bu veri artırma işlemlerinin ardından 19636 adet veri (örnek) elde edilmiştir, bu veri sayısının %20 kadarı test kümesinde, %16'sı ise doğrulama (*validation*) kümesinde kullanılmıştır. Geriye kalan %64'lük veri ise modeli eğitmek (*training*) amacıyla kullanılmıştır.

4.2 Başarım değerlendirme

Bu çalışmada eğitilen modellerde modelin başarımını ölçmek amacıyla doğruluk (accuracy) değeri dikkate alınmıştır. Bu değerin hesaplanması için gerekli eşitlik *Denklem (1)* ile verilmektedir. Ayrıca, sınıflandırmada oluşan hata oranı (missclassification rate) ise *Denklem (2)* ile ifade edilmektedir. Bu eşitliğe göre, doğru pozitif (True Positive - *TP*) ölçümü doğru şekilde sınıflandırılmış pozitif etiketli değerleri göstermektedir. Doğru Negatif (True Negative – *TN*) ölçümü doğru şekilde sınıflandırılmış negatif etiketli değerleri göstermektedir. Yanlış Pozitif (False Positive – *FP*) ölçümü yanlış şekilde sınıflandırılmış pozitif etiketli değerleri göstermektedir. Yanlış Negatif (False Negative – *FN*) ölçümü ise yanlış şekilde sınıflandırılmış negatif etiketli değerleri göstermektedir.

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

$$Hata\ oranı = \frac{FP+FN}{TP+FN+FP+TN} = 1 - Doğruluk \quad (2)$$

Bu ölçümler kullanılarak literatürdeki çeşitli başarım ölçütleri olarak duyarlık (precision), anma (recall) ve bu iki değerin harmonik ortalaması olan *F1* ölçütü gibi değerler hesaplanabilmektedir. Bu ölçütler sırasıyla aşağıdaki *Denklem (3) ilâ Denklem (5)* arasındaki denklemlerle verilmektedir.

$$Duyarlık = \frac{TP}{TP+FP} \quad (3)$$

$$Anma = \frac{TP}{TP+FN} \quad (4)$$

$$F1\ ölçütü = 2 * \frac{Duyarlık * Anma}{Duyarlık + Anma} \quad (5)$$

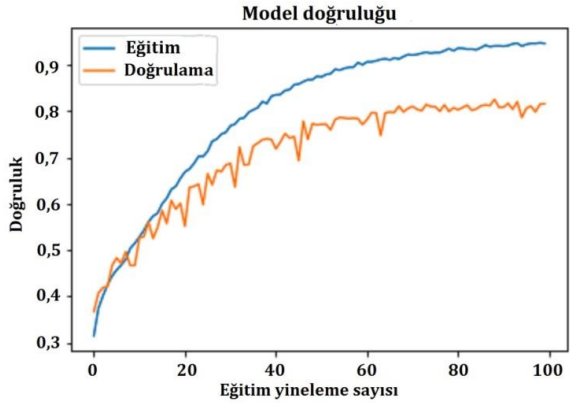
Çalışmamızda bu ölçüm ve ölçütlerin değerleri yüzdelik oran (%) olarak kullanılmıştır. Bunun yanı sıra Alıcı İşletim Karakteristik (Receiver Operating Characteristics - ROC) eğrisine dair plot çizimleri de oluşturulabilmektedir [33, 34].

4.3 Programatik olarak kodlama ile model oluşturulması ve eğitilmesi

Bu çalışmada oluşturulan model, 6 adet evrişim katmanı bloğu ve 3 adet LSTM katmanını içeren bloka sahiptir. Evrişim bloklarından bazılarında yığın normalizasyonu (batch normalization) katmanı kullanılarak eğitimin doğruluk oranının iyileştirilmesi amaçlanmıştır [20]. Bu katman ve bloklarda aktivasyon fonksiyonu olarak Doğrultulmuş Doğrusal Birim (Rectified Linear Unit - ReLU) aktivasyon fonksiyonu kullanılmıştır [31]. Çıktılar, birleştirme (Concatenate) katmanı ile birleştirilmiştir.

Derin ağ mimari modelinin son katmanlarında yoğun (dense) katmanlarının aralarında oluşabilecek ezberlemeyi (overfitting) engellemek amacıyla *iletim sönümü* (dropout) katmanları da kullanılmıştır [18]. Çıktı katmanında ise *softmax* aktivasyon fonksiyonu kullanılarak sınıflandırma yoluyla tahminleme (prediction) yapılmaktadır. Oluşturulan bu derin sinir ağı mimarisi modeline

ait blok diyagramı aşağıdaki Şekil 2'de verilmektedir.



Şekil 3. Programatik olarak kodlama ile oluşturulmuş modelin eğitim sırasındaki doğruluk değeri grafiği

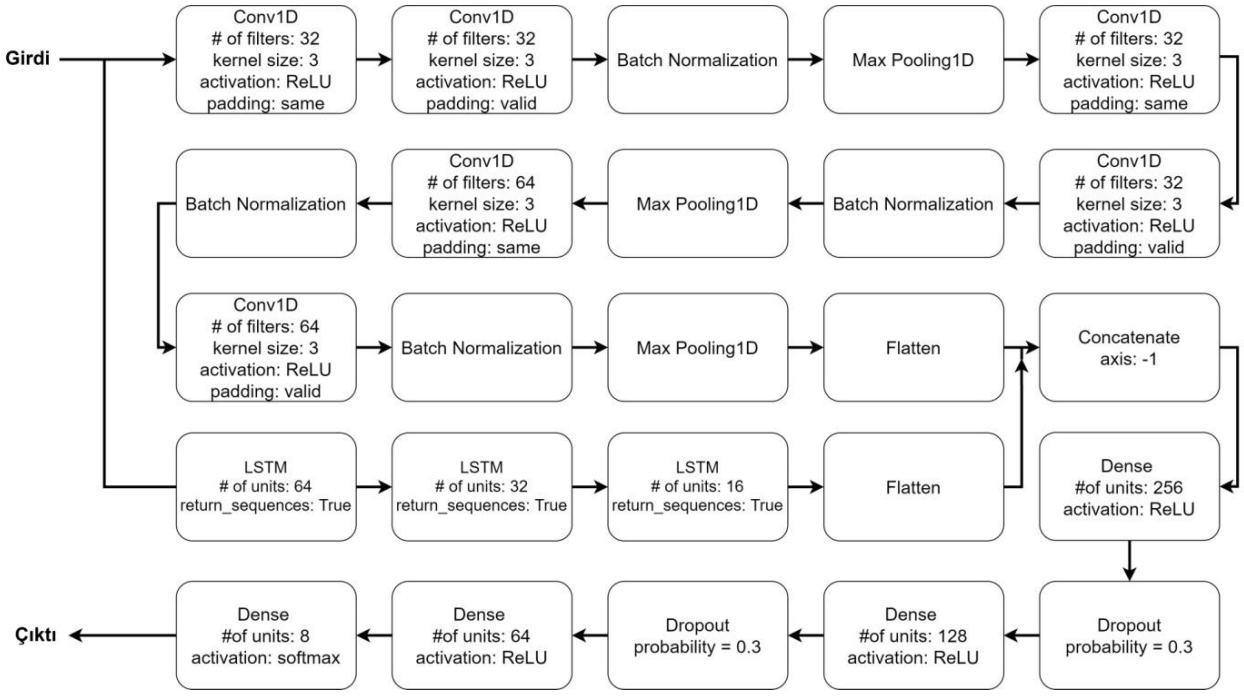
Şekil2'deki model eğitilirken farklı üstün parametre değerleri denenerek en yüksek doğruluk oranına sahip ağ mimari modeli elde edilmeye çalışılmıştır. Model eğitilirken kullanılan eğitim veri kümesi ve doğrulama veri kümesi üzerindeki doğruluk değerlerine ait grafik Şekil 3'te görülmektedir.

Şekil 3'ten anlaşılacağı üzere yirminci eğitim yineleme adımı (epoch) değerinden itibaren modelin doğruluğu, doğrulama veri kümesine bakıldığında eğitim veri kümesi değerinin gerisinde kalmaya başlamıştır.

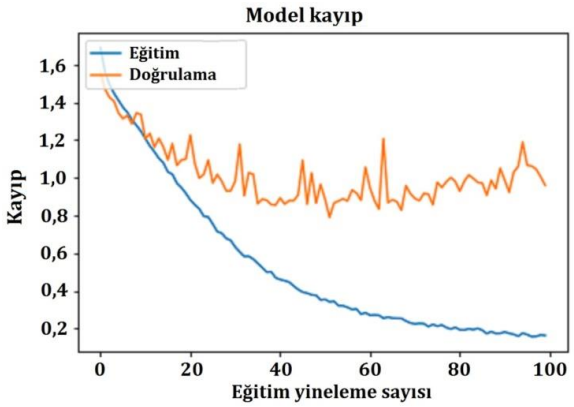
Deneylerdeki başarım ölçüt sonuçlarının hesaplanmasında Keras altyapısının kullanımı nedeniyle ilgili değerler ağırlıklandırılmış ortalama değerleri üzerinden elde edilmektedir [17].

En son adımda ise modelin doğrulama veri kümesi üzerinde %81,57 doğruluk değerine ulaşabildiği gözlenmiştir. Eğitim doğruluk değeri ise %96 olarak elde edilmiştir. Şekil 4'te de benzeri bir durum mevcuttur.

Şekil 4'teki kayıp (loss) değerine de baktığımızda eğitim sırasında yaklaşık yirminci eğitim yineleme adımından (epoch) itibaren doğrulama veri kümesi üzerindeki kayıp değerinin eğitim veri kümesi üzerindeki kayıp değerinin gerisinde kaldığı görülmektedir.

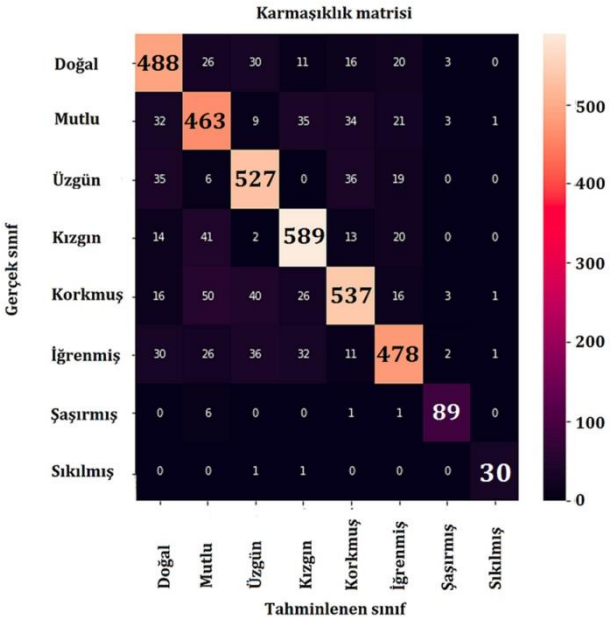


Şekil 2. Programatik olarak kodlanan derin sinir ağı mimari modeli blok diyagramı



Şekil 4. Programatik olarak kodlama ile oluşturulan modelin eğitim esnasındaki kayıp değeri grafiği

Modelin son adımda doğrulama veri kümesi üzerindeki kayıp değeri 0,9621 olmuştur. Eğitimi tamamlanan model test veri kümesiyle test edilmiş, böylece doğruluk değeri %81,49 olarak ölçülmüştür. Elde edilen tahminlere dair gerçek sınıf ve tahminlenen sınıfları gösteren karmaşıklık matrisi Şekil 5'te görülmektedir.



Şekil 5. Programatik olarak kodlama ile oluşturulan modelin test veri kümesi üzerinden yapılan tahminleme ile elde edilen karmaşıklık matrisi

Karmaşıklık matrisinden de görülebileceği gibi, matrisin diyagonal ekseni üzerindeki en yüksek doğru pozitif (*TP*) tabanlı başarım en açık renkle gösterilen "*kızgın*" (angry) etiketli duygu durum sınıfı için 589 adet doğru tespitle olduğu görülmektedir.

Ses duygu durumlarındaki insan kulağı algısına dayanan sesin ne kadar güçlü ve gür, sinyalin ne kadar yoğun olduğu ile ilgili bir ölçüm olarak genlik değeri desibel (dB) cinsinden dikkate alınmaktadır. Bu bakış açısıyla, çalışmamızda kullanılan RAVDESS, CREMA-D, EMO-DB ve SAVEE veri kümelerinin birleştirilmesi ve veri artırma yöntemi yoluyla oluşturulan büyük veri kümesinin duygu durum bazında dinamik aralıklar da göz önüne alınarak, asgari ve azami genlik değerleri ölçülerek genel ortalama genlik değerleri hesaplanmıştır. Buna göre; oluşturulan büyük veri kümesindeki sınıfların sırasıyla; "*doğal*" (neutral) için -92,41 dB ilâ -31,58 dB aralığında, "*mutlu*" (happy) için -141,77 dB ilâ -54,46 dB aralığında, "*üzgün*" (sad) için -105,95 dB ilâ -36,87 dB aralığında, "*kızgın*" (angry) için -107,36 dB ilâ -41,38 dB aralığında, "*korkmuş*" (fear) için -109,83 dB ilâ -40,20 dB aralığında, "*iğrenmiş*" (disgust) için -105,72 dB ilâ -39,97 dB aralığında, "*şaşırmış*" (suprise) için -120,28 dB ilâ -50,54 dB aralığında ve "*sıkılmış*" (bored) için -64,28 dB ilâ -11,81 dB aralığındaki genlik değerlerine sahip oldukları görülmektedir. Tüm duygu durum etiketlerine dair sınıflara düşen ses örneklerinin yerel yoğunluk düzeylerinin (genlik) üzerinden alınan veri kümesinin genel ortalamasını göz önüne aldığımızda ilgili ses sinyallerinin genlik aralığının -141,77 dB ilâ -11,81 dB aralığında olduğu görülmektedir.

Frekans, sesin enerjisindeki değişimin birim zamandaki değerinin sıklığına dayanan ve örnekleme oranı sayesinde sayısallaştırma yoluyla verinin işlenebilmesinde kullanılan bir değer olarak Hertz (Hz) birimi ile verilmektedir. Çalışmamızda, örnekleme oranı belirlenirken Librosa [24] altyapısı kullanılmıştır. Bu nedenle ilgili deneylerde kullanılan büyük veri kümemizdeki girdi ses sinyallerinin frekans aralığı 0 Hz ilâ 22 kHz arasında alınmıştır. Konuşma tonu formantlarıyla temel frekans değer aralıkları göz önüne alındığında, "*üzgün*" (sad), "*kızgın*" (angry), "*korkmuş*" (fear) ve "*iğrenmiş*" (disgust) gibi yerel yoğunluk (genlik) değer aralıkları benzerlik gösteren duygu durumların ayrıştırılmasında frekans aralıklarının kullanılabileceği anlaşılmıştır.

Çalışmamızda kullanılan büyük veri kümesinin mevcut duygu durumlarındaki tüm girdi ses örneklerinden ölçülen ortalama asgari ve ortalama azami frekans değerlerinin tüm duygu durumlar

için normalize değerleri dikkate alınarak sırasıyla; asgari frekansın 45 Hz ve azami frekansın 19 kHz değerlerinde olduğu görülmektedir.

Duygu durum frekans aralıkları hesaplanırken her bir duygu durumun her bir ses örneğindeki her bir ses çerçevesi başına ilgili değerlerin ölçülmesiyle, çalışmamızda spektral ağırlık merkezi (*spectral centroid*) hesabı yapılarak her bir çerçeveden elde edilen değerlerin normalize edilmesi sayesinde bu aralıklar frekans dağılımı şeklinde ele alınmıştır. Hesaplamada Librosa [24] altyapısı kullanılmıştır.

Böylece herhangi bir duygu durumun normalize edilmiş frekans aralığı belirlenmekte, o duygu durumu ifade eden düşük frekans veya yüksek frekans değerlerinin hangi belirgin öznitelikleri etkilediği, hangi aralıklara dair bilgi içerebildiği de anlaşılmaktadır. Çalışmadaki tüm duygu durumlar üzerinden ölçülen spektral ağırlık merkezi frekans değerleri 381 Hz ilâ 14,15 kHz aralığındadır. Oluşturulan büyük veri kümesindeki sınıfların içerdiği ses örnekleri için sınıf bazında spektral ağırlık merkezi ortalama değerlerinin sırasıyla; "*doğal*" (neutral) için 3,68 kHz, "*mutlu*" (happy) için 5,34 kHz, "*üzgün*" (sad) için 4 kHz, "*kızgın*" (angry) için 4,02 kHz, "*korkmuş*" (fear) için 4,24 kHz, "*iğrenmiş*" (disgust) için 4,22 kHz, "*şaşırmış*" (suprise) için 4,88 kHz ve "*sıkılmış*" (bored) için 1,37 kHz olduğu görülmektedir.

Tablo 2'de programatik olarak kodlama ile oluşturulan modele dair sınıf bazında deneysel ölçüm sonuçları (*TP*, *TN*, *FP* ve *FN* değerleri) verilmektedir.

Tablo 2. Programatik olarak kodlama ile oluşturulan modele dair sınıf bazında deneysel ölçüm sonuçları

	TP	TN	FP	FN
Doğal	488	3207	127	106
Mutlu	463	3175	155	135
Üzgün	527	3187	118	96
Kızgın	589	3144	105	90
Korkmuş	537	3128	111	152
İğrenmiş	478	3215	97	138
Şaşırmış	89	3820	11	8
Sıkılmış	30	3893	3	2

Yüksek TP değerinin elde edilmesinin başlıca nedenleri arasında, bu sınıfa ait ses verisinden elde edilen özniteliklerin diğer duygu durumlarda elde edilen özniteliklere göre ayrıştırıcılık sağlayacak genlik ve frekans aralığı olarak daha düşük veya yüksek değerlere sahip olabilmesi, yanı sıra belirgin öznitelikleri içerecek bu şekildeki girdi verisiyle eğitilen yapay sinir ağının verideki diğer özniteliklere göre oransal olarak daha fazla soyut özniteliği barındıran veri ile çalışmasıdır.

Tablo 3'te ise programatik olarak kodlama ile oluşturulan modele dair sınıf bazında başarım ölçüt değerleri (doğruluk, sınıflandırmadaki hata oranı, duyarlık, anma ve *F1* ölçütü) yüzdelik oran olarak verilmektedir. Tablo 3'ten görülebileceği gibi *F1* ölçütü (%92,30) ve doğruluk oranı (%99,87) göz önüne alındığında en yüksek başarım değerine "*sıkılmış*" (bored) etiketli sınıf için ulaşıldığı anlaşılmaktadır.

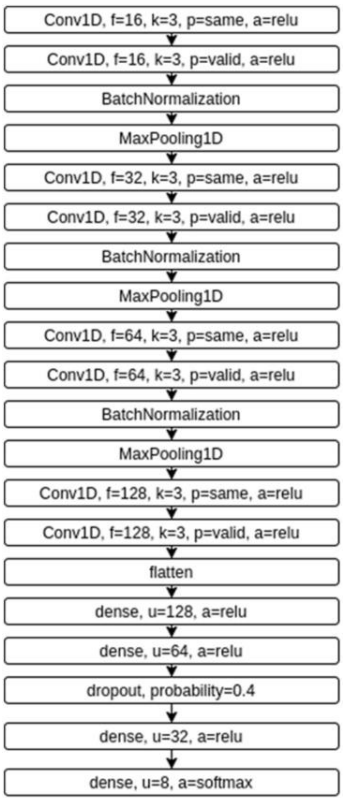
Tablo 3. Programatik olarak kodlama ile oluşturulan modele dair sınıf bazında başarım ölçüt değerleri

	Doğruluk (%)	Hata oranı (%)	Duyarlık (%)	Anma (%)	*F1* ölçütü (%)
Doğal	94,06	5,94	79,34	82,15	80,72
Mutlu	92,61	7,39	74,91	77,42	76,15
Üzgün	94,55	5,45	81,70	84,59	83,12
Kızgın	95,03	4,97	84,87	86,74	85,79
Korkmuş	93,30	6,70	82,87	77,93	80,32
İğrenmiş	94,01	5,98	83,13	77,59	80,26
Şaşırmış	99,51	0,49	89,00	91,75	90,35
Sıkılmış	99,87	0,13	90,90	93,75	92,30

4.4 Yazılımsal araç ile model oluşturulması ve eğitilmesi

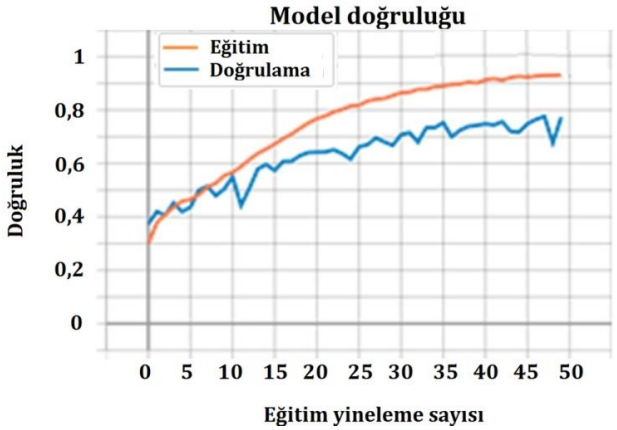
Yazılımsal aracın sunduğu kullanıcı grafik arayüzü kullanılarak oluşturulan modelde 6 adet 1 Boyutlu evrişim (Conv1D) katmanı ve eğitimin doğruluğunu iyileştirmek amacıyla bu katmanların arasına yığın normalizasyonu (batch normalization) katmanları eklenmiştir. Azami biriktirme katmanlarının da yardımıyla girdi verisi istenilen uygun boyutlarda bir öznitelik haritası haline getirilmektedir. Oluşturulan modelin mimarisi Şekil 6'da görülebilir. Çalışmada oluşturulan yazılımsal araca dair grafik kullanıcı arayüzü ile eğitim başlatıldığında program otomatik olarak tarayıcı üzerinde Tensorboard aracını başlatmaktadır. Buna göre, deneylerdeki başarım ölçüt sonuçlarının hesaplanmasında Keras

altyapısının kullanımı nedeniyle ilgili değerler ağırlıklandırılmış ortalama değerleri üzerinden elde edilmektedir [17].



Şekil 6. Yazılımsal araç ile oluşturulan modelin mimarisi

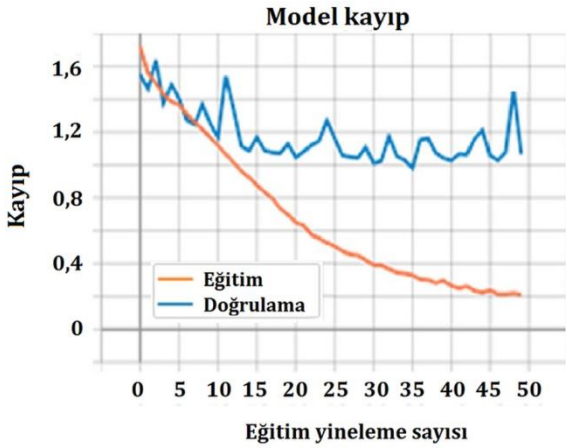
Şekil 7'de bu araç üzerinden alınmış, oluşturulan modelin eğitimi esnasında eğitim ve doğrulama veri kümelerinin üzerindeki her bir eğitim yineleme sayısı (epoch) için doğruluk değerlerinin gösterildiği plot çizimi bulunmaktadır.



Şekil 7. Yazılımsal araç kullanıcı arayüzü ile oluşturularak eğitilen modele ait doğruluk değerleri

Eğitim sonucunda model, eğitim veri kümesi üzerinde %93 doğruluk değerine, doğrulama veri kümesi üzerinde ise %77,34 doğruluk değerine ulaşmıştır. Yazılımsal araç ile oluşturulan modelin eğitim sonucuna bakıldığında, programatik kodlama ile oluşturulan modelin sonuçlarına oldukça yakın sonuçlar olduğu anlaşılmaktadır.

Şekil 8'de *Tensorboard* üzerinden alınmış, modelin eğitimi esnasında eğitim ve doğrulama veri kümelerinin üzerindeki her bir eğitim yineleme sayısı (epoch) için kayıp değerlerini gösterdiği plot çizimi bulunmaktadır.

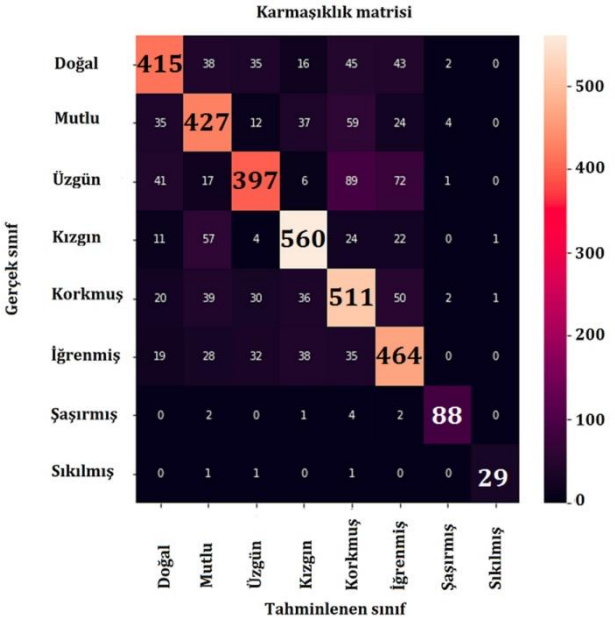


Şekil 8. Yazılımsal araç kullanıcı arayüzü ile oluşturularak eğitilen modele ait kayıp değeri grafiği

Eğitim sonucunda model eğitim veri kümesi üzerinde 0,2113, doğrulama veri kümesi üzerinde ise 1,066 kayıp değerine sahiptir. Eğitimi tamamlanmış model için, daha önce görmediği test için ayrılmış olan, toplam veri kümesinin %20 kadarından oluşan test veri kümesi ile işleme tabi tutulmuştur. Test veri kümesi üzerinde modelin yaptığı sınıflandırma sonucu doğruluk oranı %75,76 olup aynı modele ait tahminlenen sınıflar ve gerçek sınıfların eşleşme oranlarını gösterir karmaşıklık matrisi Şekil 9'da verilmektedir.

Karmaşıklık matrisinden de görülebileceği gibi, matrisin diyagonal ekseni üzerindeki en yüksek doğru pozitif (*TP*) tabanlı başarım en açık renkle gösterilen "*kızgın*" (angry) etiketli duygu durum sınıfı için 560 adet doğru tespitle olduğu görülmektedir. Bu değer, çalışmamızda oluşturulan hem programatik olarak kodlanan model hem de yazılımsal araç kullanıcı arayüzü ile oluşturulan modelin yaklaşık olarak aynı başarımı gösterdiğini ve bu yolla yazılımsal aracın oldukça etkin ve

doğru çalıştığının kanıtlandığını da göstermektedir.



Şekil 9. Yazılımsal araç kullanıcı arayüzü ile oluşturularak eğitilen modele ait karmaşıklık matrisi

Tablo 4'te yazılımsal aracın kullanıcı arayüzü ile oluşturulan modele dair sınıf bazında deneysel ölçüm sonuçları (*TP*, *TN*, *FP* ve *FN* değerleri) verilmektedir. Tablo 5'ten görülebileceği gibi *F1* ölçütü (%92,06) ve doğruluk oranı (%99,87) göz önüne alındığında en yüksek başarım değerine "*sıkılmış*" (bored) etiketli sınıf için ulaşıldığı anlaşılmaktadır.

Tablo 5'te ise yazılımsal araç kullanıcı arayüzü ile oluşturulan modele dair sınıf bazında başarım ölçüt değerleri (doğruluk, sınıflandırmadaki hata oranı, duyarlık, anma ve *F1* ölçütü) yüzdelik oran olarak verilmektedir.

Tablo 4. Yazılımsal aracın kullanıcı arayüzü ile oluşturulan modele dair sınıf bazında deneysel ölçüm sonuçları

	TP	TN	FP	FN
Doğal	415	3208	126	179
Mutlu	427	3148	182	171
Üzgün	397	3191	114	226
Kızgın	560	3115	134	119
Korkmuş	511	2982	257	178
İğrenmiş	464	3099	213	152
Şaşırmış	88	3822	9	9
Sıkılmış	29	3894	2	3

Çalışmamızda oluşturulan tüm modellerin (programatik olarak kodlanan veya yazılımsal araç kullanıcı arayüzü ile oluşturulan model) eğitilmesi esnasında kullanılan üstün parametrelerin deneylerde kullanılan değerleri olarak; öğrenme katsayısı (learning rate) değeri 0,001 alınmış, ağ eğitimi için kayıp fonksiyonu (loss function) olarak "*ayrık kategorik çapraz entropi*" (sparse categorical crossentropy) kullanılması tercih edilmiştir [16, 17, 18].

Ayrıca, yığın büyüklüğü (batch size) sırasıyla programatik olarak kodlanan model için 10, diğer model için 32 alınmıştır. Eğitim yineleme sayısı (epoch) değeri sırasıyla programatik olarak kodlanan model için 100, diğer model için ise 50 alınmıştır.

Tablo 5. Yazılımsal aracın kullanıcı arayüzü ile oluşturulan modele dair sınıf bazında başarım ölçüt değerleri

	Doğruluk (%)	Hata oranı (%)	Duyarlık (%)	Anma (%)	*F1* ölçütü (%)
Doğal	92,23	7,77	76,70	69,86	73,12
Mutlu	91,01	8,99	70,11	71,40	70,75
Üzgün	91,34	8,66	77,69	63,72	70,01
Kızgın	93,55	6,45	80,69	82,47	81,57
Korkmuş	88,92	11,08	66,53	74,16	70,14
İğrenmiş	90,70	9,30	68,53	75,32	71,77
Şaşırmış	99,54	0,46	90,72	90,72	90,72
Sıkılmış	99,87	0,13	93,54	90,62	92,06

Her iki model için de, eğitim ve test esnasında veri karıştırma (shuffle) yapılmış, toplam veri kümesi belirli oranlarda bölütlere rasgele bir yoldan parçalanarak kullanılmıştır. Buna göre; doğrulama (validation) için %16, eğitim (training) için %64 ve test kümesi için %20 oranları olarak belirlenmiştir.

Eniyileme algoritması (optimization algorithm) olarak uyarlanır momentler (Adaptive Moments Estimation - ADAM) eniyileme yöntemi kullanılmıştır. Literatürde oldukça sık tercih edilen stokastik eğimli iniş tabanlı bir yöntemdir [17, 21].

5 Sonuçlar

Ses verisi üzerinden duygu analizi konusunda, derin öğrenme gittikçe daha işlevsel hale gelmektedir. Bu çalışmada literatürde mevcut olmayan bir biçimde ses verisi üzerinden insan duygu analizi yapan son kullanıcının etkileşimli kullanabildiği derin öğrenme altyapısına dayanan kompakt bir sistemin yazılımsal araç olarak geliştirilmiştir. Buradaki motivasyon, araştırmacı ve geliştiricilerin daha az zaman ve maliyet harcayarak süreci tamamlamalarını sağlamaktır. Bu amaçla oluşturulan yazılım aracını kıyaslamak için doğrulama kümesi üzerinde %81,57, test veri kümesi üzerinde ise %81,49 doğruluk değerlerine sahip bir model oluşturulmuştur. Yazılım aracı kullanarak çok daha az zaman maliyeti ile ve hiç kod yazmadan benzer bir model oluşturulmuştur. Oluşturulan bu modelin doğrulama kümesi üzerindeki doğruluk değeri %77,34, test veri kümesi üzerindeki doğruluk değeri ise %75,76'dır. Bu sonuçlar göz önüne alındığında kullanıcıların bir programlama dili bilgisine ihtiyaç duymadan çok daha kısa sürede klasik yöntemler ile oluşturulan modellere kıyasla başarımları bu modellere yakın modeller üretebilmektedir. Gelecekteki çalışmalarımızda, üstün parametre ve katman çeşitliliği yoluyla sistemin farklı derin ağ mimari modelleri desteklemesi planlanmaktadır.

Kaynaklar

[1] Liu B. *Sentiment Analysis and Opinion Mining*. California, USA, Morgan Claypool Poblishers, 2012.

[2] Neri F, Aliprandi C. Capeci F, Cuadros M, By T. "Sentiment Analysis on Social Media". *IEEE/ACM 2012 Internation Conferance on Advances in Social Networks Analysis and Mining, 919-926, 2012.*

[3] Agarwal B, Mittal N. *"Machine Learning Approach for Sentiment Analysis". Prominent feature extraction for sentiment analysis.* Springer, Cham, 21-45 2016.

[4] Aldeneh Z, Provost EM. "Using Regional Saliency for Speech Emotion Recognition". *IEEE Int'l Conferance Acoustics Speech and Signal Processing* (ICASSP), 2741-2745, 2017.

[5] Seehapoch T, Wongthanavasu S. "Speech Emotion Recognition Using Support Vector Machines". *International 5th conferance on Knowledge and Smart Technology (KST), 86-91, 2013.*

[6] Schuller B., Rigoll G, Lang M. "Hidden Markov Model-Based Speech Emotion Recognation*". IEEE 2th International Conferance on Acoustics, Speech, and Signal Processing, II-1, 2013.*

[7] Lee CC, Mower E, Busso C, Lee S, Narayanan S. " Emotion Recognation Using a Hierarchial Binary Decision Tree Approach". *Speech Communication 55(9-10), 1162-1171, 2011.*

[8] Bertero D, Fung P. "First Look Into a Convolutional Neural Network For Speech Emotion Detection*" Acoustics Speech and Signal Processing (ICASSP) 2017 IEEE Intl. Conference*, 5115-5119, 2017.

[9] Badshah AM, Jamil A, Rahim N, Baik SW. "Speech Emotion Recognition From Spectrograms With Deep Convolutional Neural Network". *IEEE Int'l Conference On Platform Technology And Service (Platcon)*, 1-5, 2017.

[10] Yoon S, Byun S, Jung K. *"Multimodal Speech Emotion Recognition Using Audio and Text". IEEE Spoken Language Technology Workshop (SLT)*, 112-118, 2018.

[11] Livingstone SR, Russo FA. "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) A dynamic, multimodal set of facial and vocal expressions in North American English". *PIoS one, 13(5), e0196391, 2018.*

[12] Cao H, Copper DG, Keutmann MK, Gur RC, Nenkova A, Verma R. "CREMA-D: Crowd-sourced Emotional Multimodal Actors Dataset". *IEEE Transactions on Affective Computing, 5(4),* 377-390, 2014.

[13] Burkhardt F, Paescheke A, Rolfes M, Sendlmeier F, Weiss B. *"*A database of German emotional speech*". 9th European Conference on Speech Communication and Technology*, 2005.

[14] Haq S, Jackson PJB. "Speaker-Dependent Audio-Visual Emotion Recognition (SAVEE)". *AVSP, 53-58, 2009.*

[15] Google LLC. "Google Teachable Machine" https://teachablemachine.withgoogle.com/ (18.04.2021).

[16] Dey N, Borra S, Ashour AS, Shi F. "Medical Images Analysis Based on Multilabel Classification". *Machine Learning in Bio-Signal Analysis and Diagnostic Imaging. Academic Press*, Chap. 9, 2018.

[17] Github. "Github Keras Repository". https://github.com/fchollet/keras

[18] Sundermeyer M, Schlüter R, Ney H. *"LSTM neural networks for language modeling".* 13th Annual Conference Of The İnternational Speech Communication Association, 2012.

[19] Srivastava N, Hingon G, Krizhevsky A, Sutskever I, Salakhutdinov R. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". *The Journal of Machine Learning Research*, 15(1), 1929-1958, 2014.

[20] Ioffe S, Szegedy C. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". *International conference on machine learning. PMLR,* 448-456, 2015.

[21] Salimans T. Kingma DP. "Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks". arXiv preprint arXiv1602.07868, 2016.

[22] Le X, Wang Y, Jo J. *"*Combining Deep and Handcrafted Image Features for Vehicle Classification in Drone Imagery*". Digital Image Computing: Techniques and Applications (DICTA),* IEEE, 1-6, 2018.
[23] Rossum GV. *Python Reference Manual.* Amsterdam, Netherlands, Centrum voor Wiskunde en Informatica, 1995.
[24] McFee B, Raffel C, Liang D, Ellis DPW, McVicar M, Battenbergk E, Nieto O. *"*Librosa: Audio and Music Signal Analysis İn Python*". Proceedings Of The 14th Python İn Science Conference, 18-25, 2015.*
[25] Grinberg M. *Flask Web Development: Developing Web Applications with Python.* O'Reilly Media INC., California, USA, 2018.
[26] The Pallets Projects. "Werkzeug The Python WSGI Utility Library". https://werkzeug.palletsprojects.com (18.04.2021).
[27] The Pallets Projects. "Click". www.palletprojects.com/p/click (18.04.2021).
[28] The Pallets Projects. "Jinja". www.palletprojects.com/p/jinja (18.04.2021).
[29] Allen G, Owens M. *The Definitive Guide to SQLite.* Apress LP, New York, USA, 2010.
[30] Copeland R. *Essential SQLAlchemy.* O'Reilly Media INC., California, USA, 2008.
[31] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M, Levenberg J, Monga R, Moore S, Murray DG, Steiner B, Tucker P, Vasudevan V, Warden P, Wicke M, Zheng X, Google Brain. "Combining Deep and Handcrafted Image Features for Vehicle Classification in Drone Imagery". *12th Symp. On Operating Systems Design And Implementation, 2016*
[32] Start Bootstrap, "SB Admin No 2". https://startbootstrap.com/theme/sb-admin-2 (18.04.2021).
[33] Bewick V, Liz C, Ball J. "Statistics review 13: Receiver operating characteristic curves". *Critical Care*, 8(6) , 1-5, 2004.
[34] TowardsDataScience, "Confusion Matrix for Your Multi-Class Machine Learning Model". https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826 (29.08.2021).
[35] Ses duygu durum analizi yazılımsal araç (DepSemo).https://github.com/CanakkaleDevelopers/audio-sentiment-analysis-deep-learning-tool (29.08.2021).