C¸ ok Sınıflı Solunum Sesi Sınıflandırmasında

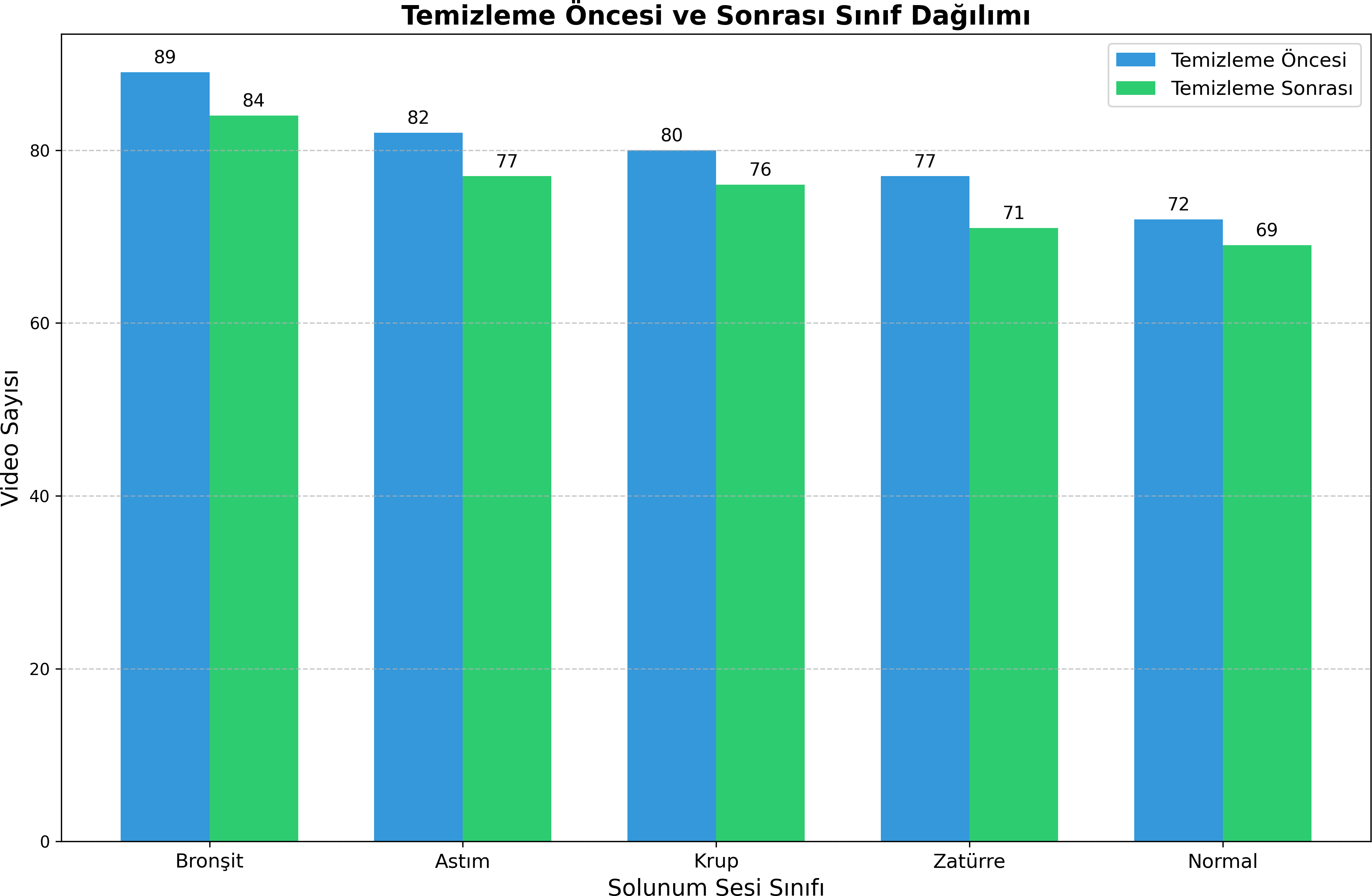
Modern Derin O¨ g˘renme Yo¨ntemlerinin

Kars¸ılas¸tırılması

˙Ibrahim Bug˘ra San

Kocaeli U¨ niversitesi / Bilis¸im Sistemleri Mu¨hendislig˘i 221307059

[ibugrasan@gmail.com](mailto:ibugrasan@gmail.com)

***Abstract*—Bu c¸alıs¸mada, astım, brons¸it, krup, normal ve zatu¨ re olmak u¨ zere bes¸ farklı solunum hastalıg˘ına ait ses verileriyle, modern derin o¨g˘renme tabanlı ses sınıflandırma modellerinin (AST, Wav2Vec2, SEW, Hubert, Data2Vec) kars¸ılas¸tırılması amac¸lanmıs¸tır. Projede veri toplama, temizleme, o¨n is¸leme, model eg˘itimi ve deg˘erlendirme su¨ rec¸leri detaylı olarak ele alınmıs¸tır.**

1. GIRIS¸

Bu c¸alıs¸mada, bes¸ farklı solunum hastalıg˘ına ait ses veri- leriyle, modern derin o¨g˘renme tabanlı ses sınıflandırma mod- ellerinin kars¸ılas¸tırılması amac¸lanmıs¸tır. Projede veri toplama,

temizleme, o¨n is¸leme, model eg˘itimi ve deg˘erlendirme

su¨rec¸leri gerc¸ekles¸tirilmis¸tir.

1. Veri Toplama ve Temizleme
2. *Web Scraping ile Veri Toplama*

Veriler, YouTube gibi c¸evrimic¸i platformlardan Python ile otomatik olarak c¸ekilmis¸tir. Her sınıf ic¸in 50 adet, toplamda

250 adet 5 saniyelik .wav formatında ses dosyası elde edilmis¸tir.

# Kullanılan Koddan Parc¸a:

with YoutubeDL(ydl\_opts) as ydl:

info = ydl.extract\_info(link, download= True)

downloaded\_file = ydl.prepare\_filename( info)

**print**(f" Haz rlanan dosya a d : { downloaded\_file}")

actual\_file = find\_downloaded\_file( downloaded\_file)

1. *Veri Temizleme ve O¨ n I˙s¸leme*
   * ˙Indirilen ses dosyaları, sınıf isimlerine go¨re klaso¨rlenmis¸tir.
   * Her dosya 5 saniyeye kırpılmıs¸tır.

Fig. 1. Veri temizleme o¨ncesi/sonrası o¨rnek go¨rselles¸tirme.

1. Kullanilan Modeller
2. Deneysel Kurulum
   * Her model ic¸in aynı eg˘itim, dog˘rulama ve test bo¨lmeleri kullanıldı.
   * Eg˘itim parametreleri: batch size, epoch, learning rate, optimizer vs.
   * Tu¨m modellerde eg˘itim ve c¸ıkarım zamanı o¨lc¸u¨ldu¨.
   * Deg˘erlendirme metrikleri: Accuracy, Recall, Precision, Sensitivity, Specificity, F-Score, AUC.
   * Sonuc¸lar tablo ve grafiklerle sunuldu.
3. Sonuc¸ lar
4. *Metrik Sonuc¸ları*

TABLE I

Modellere Go¨ re Bas¸ ari Metrikleri

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Wav2Vec2 | 0.9211 | 0.9333 | 0.9214 | 0.9214 | 0.9333 0.9193 1.0 | |
| SEW | 0.7632 | 0.8267 | 0.7679 | 0.7679 | 0.8267 | 0.7684 0.9570 |
| Hubert | 0.8158 | 0.8791 | 0.8143 | 0.8143 | 0.8791 | 0.8188 0.9416 |
| Data2Vec | 0.8158 | 0.6971 | 0.6786 | 0.6786 | 0.6971 | 0.6738 0.0 |

Model Acc. Prec. Recall Sens. Spec. F1 AUC AST 0.9211 0.9333 0.9250 0.9250 0.9333 0.9213 0.9983

* + Tu¨m dosyalar mono ve 16kHz do¨nu¨s¸tu¨ru¨lmu¨s¸tu¨r.

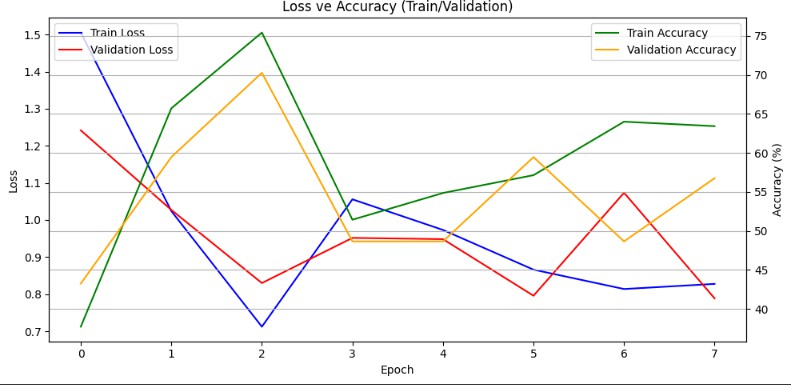
o¨rnekleme oranına

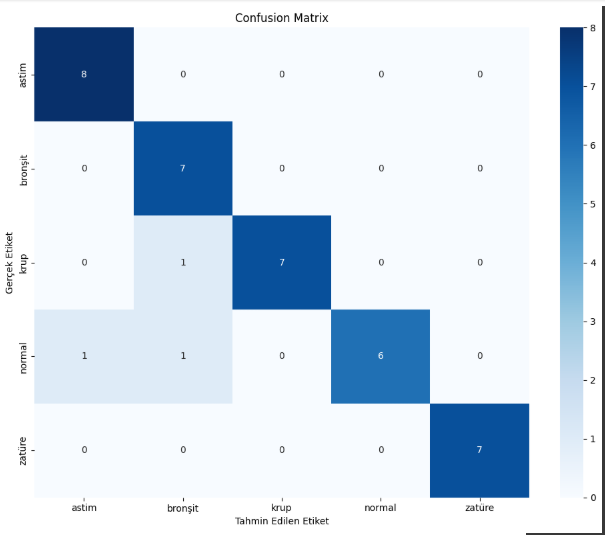
* + Ayrıca, MFCC o¨znitelikleri de c¸ıkarılmıs¸ ve .npy olarak kaydedilmis¸tir.

# Veri Temizleme Sonrası/O¨ ncesi Go¨rselles¸tirme:

1. *Karmas¸ıklık Matrisi (Confusion Matrix)*

Her model ic¸in karmas¸ıklık matrisi as¸ag˘ıda verilmis¸tir:



Fig. 2. AST O¨ rnek karmas¸ıklık matrisi.

1. *ROC Eg˘rileri ve AUC*

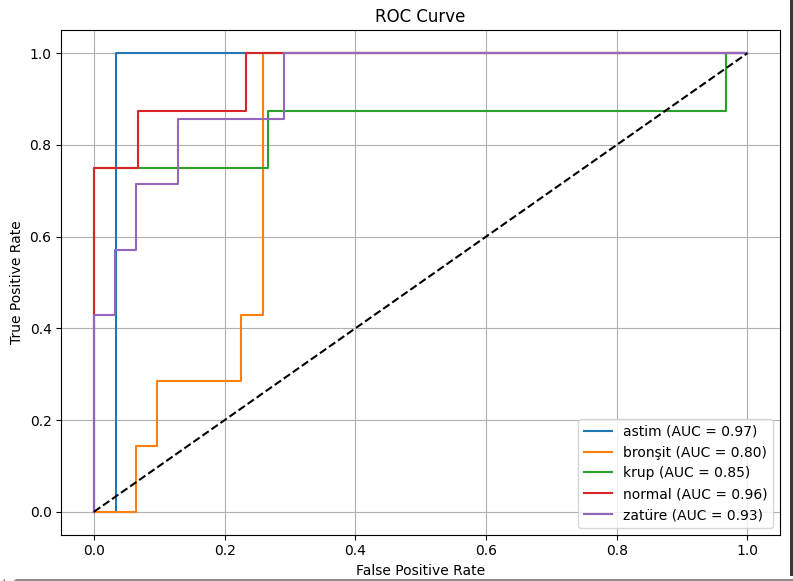
Her model ve sınıf ic¸in ROC eg˘rileri ve AUC deg˘erleri as¸ag˘ıda sunulmus¸tur:

Fig. 4. SEW O¨ rnek Eg˘itim ve dog˘rulama loss/accuracy grafig˘i.

sag˘ladıg˘ını go¨stermektedir. Gelecekte, veri setinin bu¨yu¨tu¨lmesi ve farklı o¨zniteliklerin denenmesi planlanmaktadır.

Kaynaklar References

1. Y. Gong, Y. A. Chung, and J. Glass, “AST: Audio Spectrogram Trans- former,” arXiv preprint arXiv:2104.01778, 2021.
2. A. Baevski, H. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A frame- work for self-supervised learning of speech representations,” NeurIPS, 2020.
3. H. Bao et al., “SEW: Self-Supervised Learning for Speech with Efficient Conformer,” arXiv preprint arXiv:2110.06811, 2021.
4. S. Chen et al., “BEATs: Audio Pre-Training with Acoustic Tokenizers,” arXiv preprint arXiv:2212.09058, 2022.
5. A. Baevski et al., “data2vec: A General Framework for Self- supervised Learning in Speech, Vision and Language,” arXiv preprint arXiv:2202.03555, 2022.

Ekler

*Ek-1: Web Scraping O¨ rnek Kodları*

**def** download\_audio\_segment(link, start\_time, end\_time, output\_path):

temp\_dir = ’temp’

**if not** os.path.exists(temp\_dir): os.makedirs(temp\_dir)

**for file in** os.listdir(temp\_dir): file\_path = os.path.join(temp\_dir,

**file**) **try**:

Fig. 3. DATA2VEC O¨ rnek ROC eg˘risi.

1. *Eg˘itim ve Test Loss/Accuracy Grafikleri*

Her model ic¸in epoch bazında loss ve accuracy grafikleri:

1. Tartis¸ ma ve Yorumlar

* Hangi model hangi sınıfta daha bas¸arılı oldu?
* Dengesiz veri, overfitting, modelin avantaj/dezavantajları.
* Eg˘itim ve c¸ıkarım su¨relerinin kars¸ılas¸tırılması.
* Gerc¸ek hayatta uygulanabilirlik ve o¨neriler.

1. SONUC¸

Bu c¸alıs¸mada, bes¸ farklı solunum sesi sınıfı u¨zerinde mod- ern derin o¨g˘renme modelleri kars¸ılas¸tırılmıs¸tır. Sonuc¸lar, [en iyi modelin adı] modelinin genel olarak en yu¨ksek bas¸arıyı

**if** os.path.isfile(file\_path): os.remove(file\_path)

**except** Exception as e:

**print**(f"Temp dosyas silinirken hata: {e}")

video\_id = None

**if** ’youtube.com’ **in** link:

match = re.search(r’v=([ˆ&]+)’, link)

**if** match:

video\_id = match.group(1)

**elif** ’youtu.be’ **in** link:

video\_id = link.split(’/’)[-1]

**if not** video\_id:

**import** hashlib

video\_id = hashlib.md5(link.encode()). hexdigest()[:10]

ydl\_opts = {

’format’: ’bestaudio/best’, ’outtmpl’: os.path.join(temp\_dir, f’{

video\_id}.%(ext)s’),

’quiet’: False, # Set to False to see download progress

’no\_warnings’: False, ’postprocessors’: [{

’key’: ’FFmpegExtractAudio’, ’preferredcodec’: ’wav’, ’preferredquality’: ’192’,

}],

}

**print**(f"Video indiriliyor: {link}")

*Ek-2: Google Drive Linki*

Veri setine eris¸mek ic¸in: [Veri seti Google Drive linki](https://drive.google.com/drive/u/1/folders/1XADkuWGJVvxvtqV9zuuSkaGYZg5Gixzo)