Derin Öğrenme ile Sesli Komut Tanıma

Emre Ateş

YÜKSEK LİSANS TEZİ

Elektrik Elektronik Mühendisliği Anabilim Dalı

Ağustos 2019

Voice Command Recognition with Deep Learning

Emre Ates

MASTER OF SCIENCE THESIS

Department of Electrical and Electronics Engineering

August 2019

Derin Öğrenme ile Sesli Komut Tanıma

Emre Ateş

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Elektrik Elektronik Mühendisliği Anabilim Dalı
Telekomünikasyon-Sinyal İşleme Bilim Dalında
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır

Danışman: Prof. Dr. Rifat Edizkan

Ağustos 2019

ONAY

Elektrik Elektronik Mühendisliği Anabilim Dalı Yüksek Lisans öğrencisi Emre Ateş'in YÜKSEK LİSANS tezi olarak hazırladığı "Derin Öğrenme ile Sesli Komut Tanıma" başlıklı bu çalışma, jürimizce lisansüstü yönetmeliğin ilgili maddeleri uyarınca değerlendirilerek oybirliği ile kabul edilmiştir.

Danışman **:** Prof. Dr. Rifat Edizkan

İkinci Danışman **:** -

Yüksek Lisans Tez Savunma Jürisi:

Üye : Prof. Dr. Rifat Edizkan

Üye : Doç. Dr. Emin Germen

Üye : Dr. Öğr. Üyesi Hasan Serhan Yavuz

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..... tarih ve ..... sayılı kararıyla onaylanmıştır.

Prof. Dr. Hürriyet ERŞAHAN
Enstitü Müdürü

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Prof. Dr. Rifat Edizkan danışmanlığında hazırlamış olduğum "Derin Öğrenme ile Sesli Komut Tanıma" başlıklı YÜKSEK LİSANS tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 19/08/2019

Emre Ateş

İmza

ÖZET

Sesli komut tanıma sistemleri, insan-makine etkileşimde yaygın olarak kullanılmaktadır. Ses tanımada akustik ve dil modelleri kullanılmaktadır. Ses tanımadaki başarım bu iki modele bağlıdır. Veritabanı dile ait özellikleri ne kadar iyi kapsarsa modelleme de o kadar iyi olmaktadır. Akustik ve dil modellemesi için çok disiplinli bir çalışma yapılması gereklidir. Bu tez çalışmasında, ses tanıma sistemlerinin daha kolay geliştirilmesi için derin öğrenme tabanlı yöntemin uçtan uca ses tanımada kullanılması ve sesli komut tanıma başarımlarının elde edilmesi üzerinde çalışılmıştır. Bu çalışmada, RNN yapısı ile Speech Commands Dataset içerisindeki temel komut kelimelerinin sınıflandırma başarımları elde edilmiştir. Deneysel çalışma sonunda %70,63 dogru sınıflandırma başarımı elde edilmiştir.

Anahtar kelimeler: Uçtan Uca, Ses Tanıma, Komut Tanıma, Derin Öğrenme, RNN.

SUMMARY

Voice command recognition systems are widely used in human-machine interaction. Acoustic and language models are used in voice recognition. The performance of voice recognition depends on these two models. The better the database covers the language features, the better the modeling. A multidisciplinary study is required for acoustics and language modeling. In this thesis, it has been studied to use end-to-end voice recognition method and to achieve voice command recognition achievements in order to develop voice recognition systems more easily. In this study, classification performance of basic command words in Speech Commands Dataset is obtained with RNN structure. At the end of the experimental study, 70.63% correct classification performance was obtained.

Keywords: End-to-End, Voice Recognition, Command Recognition, Deep Learning, RNN.

TEŞEKKÜR

Tez çalışmasındaki desteklerinden dolayı tez danışmanım Sayın Prof. Dr. Rifat Edizkan'a, eğitim hayatım boyunca başarılarımı borçlu olduğum, maddi ve manevi desteklerini ve yardımlarını benden hiç esirgemeyen sevgili aileme ve her zaman olduğu gibi bu süreçte de her türlü desteğiyle yanımda olan canım eşime teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET vi
SUMMARY vii
TEŞEKKÜR viii
İÇİNDEKİLER ix
ŞEKİLLER DİZİNİ xi
ÇİZELGELER DİZİNİ xiii
SİMGELER VE KISALTMALAR DİZİNİ xiv
1. GİRİŞ VE AMAÇ 1
2. LİTERATÜR ARAŞTIRMASI 3
3. MATERYAL VE YÖNTEMLER 7
3.1. Ses Tanıma Yöntemleri 7
3.1.1. Öznitelik çıkarma 8
3.1.1.1. Mel-frekans kepstral katsayıları yöntemi 8
3.1.1.2. Algısal doğrusal tahmin yöntemi 12
3.1.2. Sınıflandırma yöntemi 14
3.1.3. Hidden Markov ve derin öğrenme hibrit model 17
3.1.4. Uçtan uca ses tanıma modeli 18
3.1.4. Tekrarlayan yapay sinir ağları (RNN) 20
3.2. Dil modeli oluşturma yöntemleri 23
3.2.1. N-gram dil modeli 24
3.2.3. Fonem tabanlı yaklaşım 24
3.3. Gürültüyü en aza indirgeme yöntemleri 25
3.4. Derin Öğrenme Araçları 25
3.4.1. Tensorflow 26
3.4.2. CUDA 27
3.4.3. Colaboratory 28
4. DENEYSEL ÇALIŞMALAR 30
4.1. Test 1 33
4.2. Test 2 36
4.3. Test 3 38

İÇİNDEKİLER (devam)

Sayfa

5. BULGULAR VE TARTIŞMA..... 40
6. SONUÇ VE ÖNERİLER..... 41
KAYNAKLAR DİZİNİ 43

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

1.1. Geleneksel ses tanıma adımları ..2
3.1. Otomatik ses tanıma temel yapısı ..7
3.2. Mel Frekans Değişim Grafiği ..9
3.3. MFCC öznitelik çıkarma aşamaları ..9
3.4. Hamming ve Hanning Pencereleme Fonksiyonu uygulamaları ..10
3.5. Hamming ve Hanning Pencereleme Fonksiyonu sonuçları ..11
3.6. PLP öznitelik çıkarma aşamaları ..12
3.7. Eşit ses yüksekliği grafiği ..13
3.8. Markov Zinciri gösterimi (Kang, 2017) ..14
3.9. Ergodik HMM Gösterimi ..16
3.10. Soldan Sağa (Bakis) HMM Gösterimi ..16
3.11. Hibrit model gösterimi (a) Elman, (b) Jordan, (c) Robinson and Fallside, (d) Williams and Zipser RNN model (Rallabandi vd., 2015) ..18
3.12. Uçtan uca ses tanıma model adımları (Miao, 2017) ..19
3.13. DeepSpeech uçtan uca ses tanıma modeli kelime hata oranı sonuçları (Amodei, 2016) ..20
3.14. Tekil tekrarlayan sinir ağ gösterimi (Olah, 2015) ..21
3.15. Tekil tekrarlayan sinir ağ geçmiş veri kullanımı (Olah, 2015) ..21
3.16. RNN'nin geçmiş verileri kullanarak hatırlama işlemi gösterimi (Olah, 2015) ..22
3.17. RNN fazla geçmiş veri kullanımının olumsuz etkileri (Olah, 2015) ..23
3.18. Dil modeli karar basamakları ..24
3.19. Tensör Geometrik Gösterimi (Anonim, 2014) ..26
3.20. CUDA Performans Grafiği (Prasanna, 2018) ..28
3.21. CUDA Performans Grafiği (Harris, 2007) ..28
4.1. Fisher veri seti katılımcı dağılımı ..34
4.2. Uçtan uca metot oransal sonuç grafiği ..35
4.3. Anahtar kelime arama metodu karıştırma matrisi ..37
4.4. Anahtar kelime arama metodu ile elde edilen başarımlar ..37

ŞEKİLLER DİZİNİ (devam)

Şekil **Sayfa**

4.5. Uçtan uca metodu ve KWS metodu ile elde edilen başarımlar 38

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

3.1. Tensör Derece Gösterimi 28
4.1. Uçtan uca metot kullanılarak elde edilen deneysel sonuçlar 37
4.2. Uçtan uca metot kullanılarak elde edilen deneysel sonuçlar 40

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
HMM	Hidden Markov Model (Gizli Markov Modeli)
ARGE	Araştırma Geliştirme
RNN	Recurrent Neural Networks (Tekrarlayan Sinir Ağları)
HPC	High Performance Computing (Yüksek Performanslı İşlem)
IBM	International Business Machines (Uluslararası İş Makineleri)
CTC	Connectionist Temporal Classification
CNN	Convolutional Neural Network (Evrişimsel Sinir Ağları)
DBN	Dinamik Bayesian Network
CPU	Central Processing Unit (Merkezi İşlem Birimi)
GPU	Graphical Processing Unit (Grafiksel İşlem Birimi)
Colab	Colaboratory
KWS	Keyword Spotting
MFCC	Mel-Frequency Cepstral Coefficient
FFT	Fast Fourier Transform
PLP	Perceptual Linear Prediction

1. GİRİŞ VE AMAÇ

Günümüzde otomatik sistemler her alanda karşımıza çıkmaktadır. Bu sistemler kullanıcı için özel imkânlar sunarak kullanımı kolaylaştırmanın yanında bu sistemleri çalıştırmak için harcanan iş gücünü de önemli ölçüde azaltmaktadır. Özellikle iş güvenliği risklerinin yüksek olduğu işlemlerde otomatik sistemler kullanılarak çalışanların can güvenlikleri artırılabilmektedir.

Çalışma şartları göz önüne alındığında birçok yazılımda olduğu gibi otomatik sistemlerde kullanılan yazılımın da oluşabilecek değişikliklere ve yeni ortamlara uyarlanabilir olması sağlanmalıdır. Bu sebeple günümüzde otomatik sistemlerde yapay zekâ ve derin öğrenme teknikleri yaygın olarak kullanılmaktadır. Yapay zekâ ve derin öğrenme metotları, çalışan sistem üzerinden toplanan verileri sürekli olarak değerlendirerek isterleri optimum seviyede tutatacak şekilde uygulanacak gerekli adımları güncelleyebilmektedirler.

Belirtilen otomatik sistemlerin kontrolleri çoğunlukla sisteme özel hazırlanan bir arayüz aracılığı ile yapılmaktadır. Bu sebeple neredeyse sistemlerin kurulduğu her projede farklı bir çalışma yapılarak bu arayüzler oluşturulmaktadır. Bu arayüzler kişiye özel olarak hazırlanmadığı için farklı kullanıcılardan, çok çeşitli kullanım zorluğu ile ilgili geri dönüşler alınmaktadır. Bu noktada, insan-makine etkileşimini daha kolay hale getirmek için ses tanıma teknolojilerinden faydalanılmaktadır. Ses tanıma teknolojileri bankacılık, eğitim, endüstri ve savunma sanayi alanlarındaki uygulamalarda karşımıza çıkmaktadır.

Ses tanıma uygulamaları genel olarak HMM yapısı kullanılarak geliştirilmektedir. HMM ile hazırlanan otomatik ses tanıma sistemlerinde kelimelerin ard arda gelme olasılıklarını temel aldığı için tasarım aşamasında kullanılan dil modeli kütüphanesi önemli bir yer tutmaktadır. Bu nedenle, farklı alanlar için dil modeli kütüphaneleri kullanılması gerekmektedir.

Her farklı alan için dil modeli elde etmek zorlu bir süreçtir. Uygulamada elde edilen başarım dil modeline de bağlıdır. Dil modeli kullanmadan da uçtan uca ses tanıma yöntemleri ile ses tanıma yapılabilir.

Uçtan uca ses tanıma sistemlerinde de ses tanıma için kullanılan işlemler yapılmaktadır. Bu sistemleri geleneksel sistemlerden ayıran fark, ses tanımadaki işlemlerde derin öğrenme metotları kullanılmasıdır. Derin öğrenme metotları kullanılararak sistemin farklı konularda karşılaştığı yeni olasılıklar yeniden işlenebilmektedir. Bu sayede içerik bağımsız sistemler oluşturulabilmektedir. Şekil 1.1'de gösterilen ses tanıma adımlarının derin öğrenme metotları ile desteklendiği sistemler birçok alanda çok sık kullanılmaktadır (Hain, 2001). Ses tanıma sistemlerinin gelişimine bakıldığında belirtilen adımların tamamının derin öğrenme metotları kullanılarak konu tabanlı değişkenlerden bağımsız olarak çalışması hedeflenmektedir. Ancak bu doğrultuda hazırlanacak bir sistemin yazılımsal ve donanımsal işlem kabiliyetlerinin yüksek olması gerekmektedir. Yüksek isterleri karşılamak için kullanılan donanımsal parçaların maliyetlerinin fazla olması sebebiyle Ar-Ge çalışmaları genel olarak güçlü bir sunucu sisteminin kurulması ve sunucu üzerinden istemcilerin isteklerinin karşılanması üzerine yoğunlaşmış durumdadır.

Bu tez çalışmasında Şekil 1.1'de belirtilen adımların derin öğrenme metotlarından RNN kullanılarak uçtan uça ses tanıma yöntemi kullanılarak komut tanıma üzerinde çalışılmıştır. Konuşma içinde geçmişe yönelik elde edilen verilerden kelime tahminindeki başarımı arttırması sebebiyle derin öğrenme metotlarından RNN seçilmiştir.

Şekil 1.1. Geleneksel ses tanıma adımları

2. LİTERATÜR ARAŞTIRMASI

Otomatik sistemlerde kullanılan ses tanıma sistemleri çevre şartlarına, farklı kullanıcılara hitap etmesi sebebiyle aksan ve diksiyon farklılıklarına ve aynı anda tıp, hukuk ve mühendislik gibi farklı disiplinlere hizmet etmeye uygun olarak geliştirilmelidir. Değişkenlere dayanıklı olarak ses tanıma sistemleri üzerine çalışmalar yapan Baidu Reserach ekibinin yapmış olduğu çalışmalarda İngilizce ve Çince gibi çok farklı dil bilgisi ve kelime türetim dinamikleri olan dillerde dahi başarılı bir şekilde çalışan uçtan uca ses tanıma sistemi oluşturulmuştur. Belirtilen zorlukları aşabilmek için yapay sinir ağları ile kurulan bir sistem kullanılmıştır. Bu mertebedeki yapay sinir ağlarını yönetmek için yüksek performansta çalışabilen ekipmanların kullanılması gerekmektedir. Yapılan araştırmalarda istenilen performans değerlerine ulaşabilmek için eğitim aşamasında Batch Normalizasyon yöntemi kullanılarak her bir veri için 8 kat daha kısa süre harcanmıştır. Bu sebeple haftalar alabilen çalışmalar günler mertebesine düşürülebilmiştir (Amodei vd., 2016).

Endüstriyel sistemlerde hazırlanacak otomatik sistemin ses tanıma işlevini gerçekleştirmesinin yanında bazı işlemleri tetiklemesi de istenmektedir. Sistemin çalışmasını yönlendirmesi beklenen bu tetiklemeler sesli komut tanıma sistemleri ile yönetilebilir. IBM'in yapay zekâ çözümü olan Watson'da bu konu üzerine çalışmalar yapılmıştır. Watson bünyesinde çok güçlü bir yapay sinir ağı modeli barındırmaktadır. Bu sayede sesli verilen komutları yüksek oranlarla doğru algılayarak doğru cevaplar ile kullanıcıları yönlendirebilmektedir. Aynı zamanda sesli komutları algılayarak önceden hazırlanmış protokoller dâhilinde, çalışan sistemler üzerinde eylem gerçekleştirebilmektedir. Sesli komut işleme üzerine Watson Araştırma Merkezi'nde yapılan çalışmalar sonucunda HMM ve derin öğrenme metotlarının kombine kullanıldığı metotlara kıyasla yakın sonuçlar veren uçtan uca sesli komut algılama sistemi oluşturulmuştur. Sistem, akustik modelin oluşturulması, karakter seviyesinde dil modelinin oluşturulması ve ses tanıma işleminin yapılması olmak üzere üç alt sistemden meydana gelmektedir. Bu alt sistemlerde derin öğrenme metotlarından RNN metodu kullanılmıştır. Yapılan testlerde oluşturulan bu sistemin hibrit sisteme göre daha hızlı çalışarak daha verimli sonuçlar elde ettiği görülmüştür (Audkhasi vd., 2017).

Konuşma ile iletişim kurulurken yanlış anlaşılmaların önüne geçmek için anlatımı kuvvetlendirmek gereklidir. Anlatımı kuvvetlendirirken uzun cümleler ve örneklemeler kullanılabilir. Ses tanıma sistemlerinde bu şekilde anlatımlar karmaşık problemlere sebebiyet vermektedir. Anlatılmak istenen konuyu takip edebilmek için konuşmanın başından sonuna her bir adımı kontrol edilebilmelidir. Derin öğrenme metotlarından RNN kullanılarak geçmişte kullanılan değişkenler son değişkene veri olarak taşınabilmektedir. Ancak RNN metodu yüksek başarı oranları yakalamak için önemli miktarda işlenmiş eğitim verisine ihtiyaç duymaktadır. Yeterli miktarda ve hızlı şekilde veri işleyebilmek için CTC metodu önerilmektedir. Temel olarak CTC metodu geçmiş verilerden elde ettiği oransal değerleri kullanarak gelecek değerlerin işlenmesinde kullanmaktadır. Bu sayede işlenmemiş verileri işleyerek daha fazla eğitim verisi elde edilebilir ve RNN başarı oranı yükseltilebilir. CTC metodu ile veri setlerinin işlenmesi adımları geleneksel yöntem olan dikte yöntemini otomatikleştirmiş ve bu sayede hızlandırmıştır. Araştırmada önerilen otomatik sistemin Toy veri seti üzerinde yapılan testlerde hata oranının 0'a yakın, TIMIT veri seti üzerinde yapılan testlerde hata oranının %50 civarında olduğu görülmüştür. Araştırma, kullanılan yöntemlerde tamamen yazılımsal işlemler yapıldığı göz önüne alındığında, ses tanıma sistemlerinin eğitim setlerini hazırlamak için tek adımda hızlı bir çözüm sunmaktadır (Auvolat ve Mesnard, 2016).

Carnegie Mellon Üniversitesi'nde yapılan uçtan uca ses tanıma çalışmalarında HMM hibrit kullanım metoduna alternatif olarak belirli bir konu sınırlaması olmadan konuşmaları algılayabilen derin öğrenme metotlarından RNN kullanılarak oluşturulmuş bir sistem önerilmiştir. Sistemin çalışması için gerekli ekipman isterlerini minimuma indirmek için performans çalışmaları yapılmıştır. Yapılan çalışmalar EESEN (End-to-End Speech Recognition) framework adıyla tanıtılmıştır. Dil modelinin oluşturulması ve ses tanıma işleminin tamamlanması aşamalarında RNN kullanılmıştır. RNN eğitim setinin hazırlanması aşamasında hızlı sonuçlar alabilmek için CTC metodu kullanılmıştır. CTC metodunun başarı seviyesini arttırmak için çözümleme aşaması Weighted Finite State Transducers (WFSTs) yaklaşımından faydalanılmıştır. Yapılan çalışmalar sonucunda daha düşük seviyede ekipmanlar ile daha hızlı sonuçlara ulaşılmıştır. Derin öğrenme metoduyla HMM hibrit metodun hata oranlarının birbirne yakın olduğu görülmüştür. (Miao vd., 2015).

Ses tanıma sistemlerin temelinde dil modeli bulunmaktadır. Sonuçlar kullanılan dil modelinden elde edilen olasıklara göre oluşturulmaktadır. Johns Hopkins Üniversitesi'nde yapılan çalışmalarda RNN temelli dil modeli çözümü sunulmuştur. Hazırlanan dil modeli sık kullanılan Backoff Dil Modeline alternatif olarak gösterilmiştir. Backoff Dil Modeli kullanışlı bir N-gram dil modeli çözümüdür. N değeri değişken olacak şekilde kelimenin olasılığını belirlemek için kendinen önceki değişken değeri kadar kelimeyi de kontrol eder. Bu model tahmin için ne kadar çok kelime kullanırsa o kadar yüksek doğruluk oranında sonuçlar vermektedir. Ancak kullanılan girdilerde geçen kelime sayısının değişken olması ve adım sayısının yükselmesi ile yapılan işlem miktarı da artacağı için yüksek işlem gücü kullanmaktadır. RNN temelli önerilen dil modeli ile eğitim aşaması daha hızlı geçilebildiği için aynı sürede sistem daha fazla veri ile eğitilebilir ve başarı oranı yükseltilebilir. Bunun yanında RNN ile beraber geçmiş verilere ait değerler her adımda yeni değerlere aktarıldığı için olasılık hesaplamalarında tüm veriler kullanılabilmektedir bu sebeple daha doğru sonuçlar elde edilebilmektedir. Wall Street Journal verileri kullanılarak yapılan testte sistemler aynı miktarda veri ile eğitildiklerinde önerilen modelin %18 daha az hata yaptığı, önceki testten %5 oranında daha zorlayıcı olan NIST RT05 verileri kullanılarak yapılan testlerde Backoff Model daha fazla veri ile eğitilmesine rağmen daha düşük sonuçlar elde etmiştir (Mikolov vd., 2010).

Seslerden kelime analizi yaparken kelimeleri dil modeline göre dil bilgisi kurallarına göre ayırt etmek zorlayıcı bir durum ortaya koymaktadır. İngilizce alfabesi temel alınarak hazırlanan bir çalışmada analizler alfabetik karakter tespiti üzerine yoğunlaştırılmıştır. Bu sayede kelimenin dil bilgisi değişikliklerinden dolayı yanlış tespit edilmesinin önüne geçilmesi hedeflenmiştir. Google voice araştırma setlerinde yapılan testlerde kelimeler bir sözlük veya dil modeli kullanılmadan %14.1 hata oranı ile tespit edilebilmiştir. Aynı verilerle hazırlanan HMM hibrit model ile testler tekrarlandığında %8 hata oranı gözlemlenmiştir. Karakter tabanlı tespit edilen hata oranları hibrit modele oranla yüksek olmasına rağmen bir dil modeli veya sözlük kullanılmaması göz önüne alındığında rekabetçi sonuçlar alınmıştır (Chan vd., 2015) .

Otomatik ses tanıma sistemlerinin her türlü ortamda yüksek başarım vermesi istenir. Fakat çevresel etkenler ses tanıma işlemleri sırasında girdi olarak kullanılan sesleri bastırabilir veya kolayca anlaşılamayacak derecede bozulmasına sebebiyet verebilir. Bu

sebeple hazırlanacak sistemin çevresel etkenlere dayanıklı olmalıdır. Maas vd.nin (2012) önermiş olduğu modelde girdi olarak kullanıcak ses verileri öncelikle bir RNN temelli bir dönüştürücüden geçirilerek temiz ses verileri elde edilmek istenmiştir. Aurora2 veri seti üzerinde yapılan testlerde 5 dB, 10 dB, 15 dB ve 20 dB olacak şekilde gürültüler temiz verilere uygulanmıştır. İşlem sonucunda araştırmada karşılaştırılan benzer yaklaşımlara oranla daha başırılı sonuçlar elde edildiği görülmüştür.

Sesi çevresel etkenlerden arındırabilmek için, girdi olarak alınan ses verisinin içerisinde anahtar kelime arama yöntemi izleyerek ses tanıma sürecinde başarı oranları yükseltilebilir. Ses verisi içerisinde anahtar kelime aramak için derin öğrenme metotları ile oluşturulan bir KWS yapısı kullanılabilir. Bu şekilde oluşturulacak yapı her bir giriş verisinde istenilen verilerin varlığını araştıracağı için sistem performansını olumsuz olarak etkileyebilmektedir. Bu sebeple KWS yapısı kurulurken CNN metodu kullanılmıştır ve kullanılacak katman sayısı beş olarak seçilmiştir. Google Home ile canlı ortamda yapılan testlerde 0 dB ile 10 dB arası müzik ve televizyon sesi konuşma seslerine eklenmiştir. Mevcut çalışan yapıyla karşılaştırıldığında komut tanıma işlemlerinde %32 oranında gelişme gözlemlenmiştir (Huang vd., 2018).

Büyük (2018) tarafından yapılan çalışmada Türkçe komut tanıma için konuşma tanıma sistemi önerilmiştir. Bu model, temel televizyon komutları, sesli mesaj komutları ve metin yazdırma uygulaması ile test edilmiştir. Yapılan testler sonucunda temel televizyon komutlarında %95'in üzerinde bir başarı oranı, sesli mesaj ve genel metin yazdırma uygulamalarında %40 ve %60 başarı oranları elde edilmiştir. Hazırlanan modelde öz nitelik çıkarma işlemleri için MFCC metodu ve dil modeli hazırlıkları için N-gram dil modeli metodu kullanılmıştır. Ek olarak akustik modelleme aşamalarında HMM kullanılmıştır (Büyük, 2018).

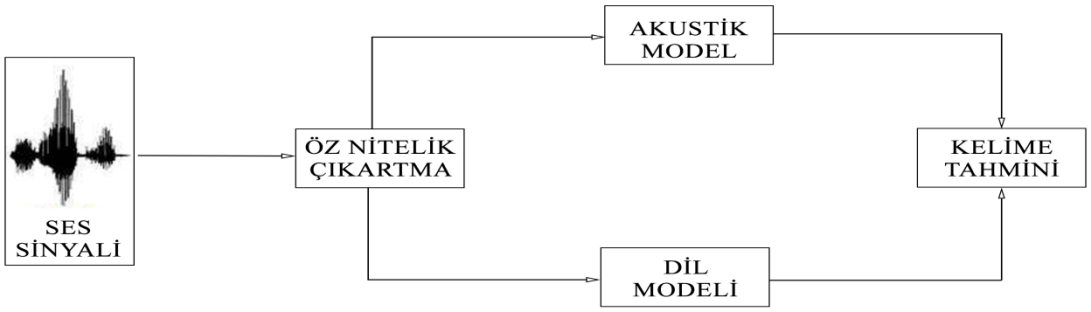
Edizkan ve Barkana'nın (2000) çalışmasında içerik bağımlı uygulama için HMM ile ses tanıma yapılmıştır. Bu çalışmadaki veritabanı 0-9 rakamları ile "Evet" ve "Hayır" kelimlerine ait ses verilerinden oluşmaktadır. Yapılan çalışmada başarım oranları, ortak değişinti matrisinin tipi, gözlem vektörünün boyutu ve durum sayısı değiştirilerek test edilmiştir. Yapılan çalışmada ortalama %90 başarı oranı elde edilmiştir.

3. MATERYAL VE YÖNTEMLER

3.1. Ses Tanıma Yöntemleri

Ses tanıma sistemlerine ait ihtiyaçlar geçmişten günümüze hızlıca artış göstermektedir. 1950'li yıllarda Bell Laboratuarlar'ında yapılan çalışmalar bu araştırmalara öncü olmuştur (Moskvitch, 2017) . 1970'ler de DARPA SUR programı ile Carneige Mellon'un çalışmaları etkin olarak kullanılabilecek ilk sistem örneklerinden Harpy'i sunmuştur. 1970'ler de çalışmalar yapılmaya başlanan ancak 1980'ler de aktif olarak kullanılmaya başlayan Hidden Markov Model'in sonuçları ile araştırmalar hız kazanmıştır (Kincaid, 2018). Günümüzde yapay zekâ ve derin öğrenme metotları kullanılarak yeni model yaklaşımları oluşturulmaya ve kurulan sistemlerin başarı oranları yükseltilmeye çalışılmaktadır.

Otomatik ses tanımadaki temel işlemler Şekil 3.1'de gösterilmektedir. Tanınmak istenen ses sinyali bir ekipman aracılığı ile alınır ve işlenmek üzere dijital formata dönüştürülür. Bu işlemler sırasında kullanılan ekipmanların özelliklerine bağlı olarak ses verilerinde kayıplar veya gürültü tabanlı bozulmalar meydana gelebilmektedir.



Şekil 3.1. Otomatik ses tanıma temel yapısı

Sinyal dönüştürme işlemleri sonrası ses sinyaline ait öznitelikler başarı oranını arttırmak için en doğru şekilde seçilerek kelime tahmini aşamasına hazırlanılır. Otomatik ses tanıma sistemlerinde pratik ve başarılı öznitelikler çıkarabilmesi sebebiyle bu aşamada genellikle mel-frekans kepstral katsayıları yöntemi kullanılmaktadır.

Örüntü tanıma adımlarında kullanılmak üzere eğitim aşamasında en iyi sınıflandırma sağlayan parametreler elde edilir. Test aşamasında ise bilinmeyen ses, seçilen benzerlik ölçüsüne göre sınıflandırılır. Ses tanımadaki başarımı arttırmak için kelimelere ait verileri tutan dil modelleri kullanılmaktadır. Ses tanıma işlemleri sırasında çevresel etkenlerden kaynaklı gürültünün tipi ve seviyesi doğru tahmin oranlarını düşürebilmektedir. Bu nedenle, ses tanımadaki başarımlar ses sinyalinde yapılan gürültüyü azaltmaya yönelik ön işlemler ile arttırılmaya çalışılmaktadır.

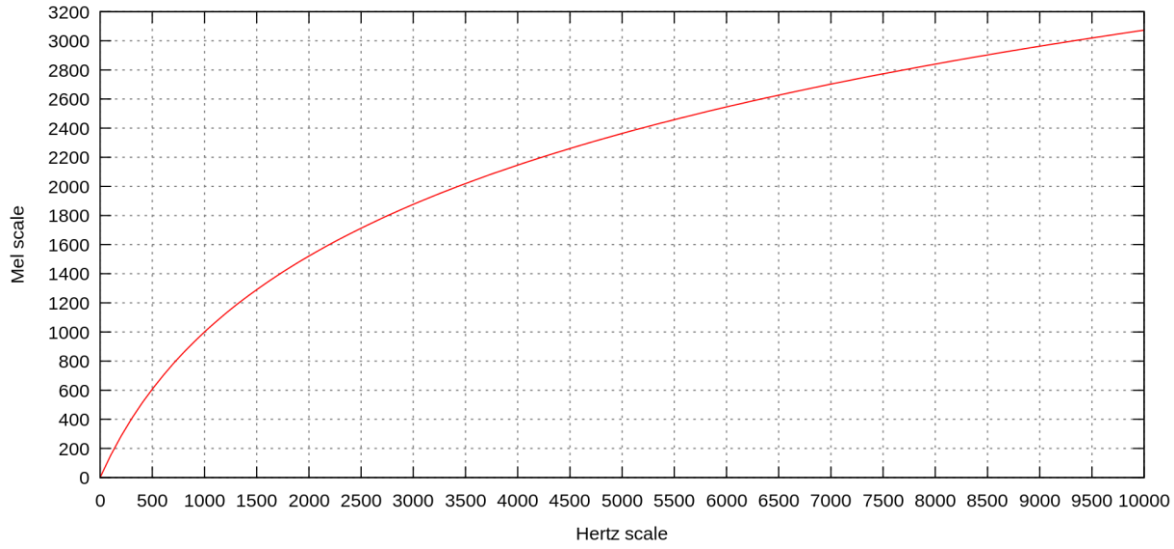
3.1.1. Öznitelik çıkarma

Öznitelik çıkartmadaki amaç sese ait tanımlayıcı ve ayırtedici özelliklerin elde edilmesidir. Genel olarak otomatik ses tanıma sistemlerinde aşağıda verilen öznitelik çıkarma metotları kullanılır.

- Mel-frekans kepstral katsayılar yöntemi
- Algısal doğrusal tahmin yöntemi (PLP)

3.1.1.1. Mel-frekans kepstral katsayıları yöntemi

MFCC, mel frekans ölçekleri kullanılarak sesin spektral özelliklerinin elde edilmesine dayalı bir özntelik çıkartma yöntemidir. Mel ölçeği deneysel olarak hesaplanırken, 40 dB'in üzerinde 1000 Hz'lik bir ton referans olarak 1000 mels olarak kabul edilmiştir. Sonrasında katılımcılardan dinlenen frekans ölçeklerini 2, 10, 100 vb. katlarında değiştirmeleri istenmiştir. Bu katsayılara karşılık gelen mel değerleri not edilmiştir. Bu ölçümler sonucunda 1 kHz üstünde değişimlerin logaritmik ve 1 kHz altnda değişimlerin doğrusala yakın oldukları gözlemlenmiştir (Öcal, 2005). Ölçümlere ait değerlerin değişimleri Şekil 3.2'de gösterilmektedir.

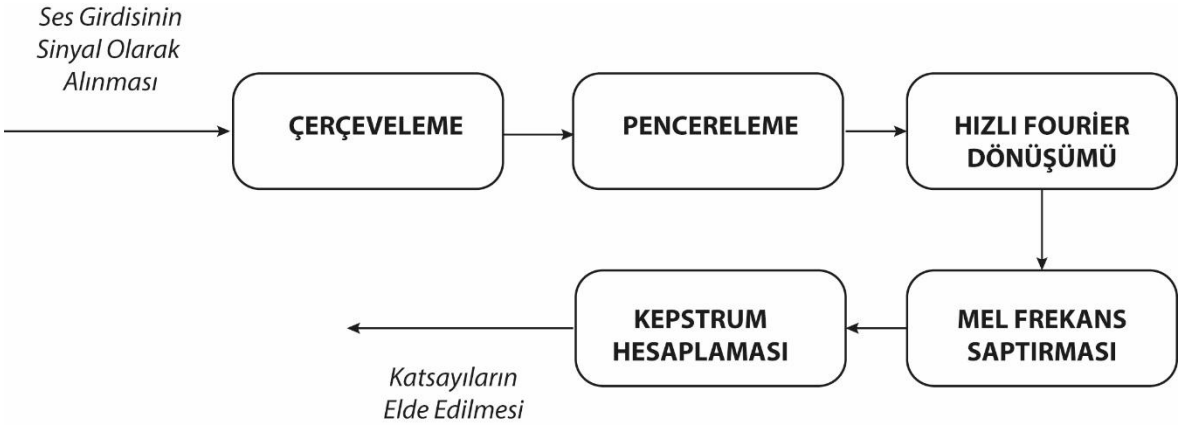


Şekil 3.2. Mel Frekans Değişim Grafiği

Mel katsayılarının dönüşümleri üzerine yapılan deneyler sonucunda elde edilen bilgilerin matematiksel ifade edilmesi için popüler olarak kullanılan formül Denklem (3.1)'de gösterilmiştir.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{3.1}$$

MFCC öznitelik çıkarma yöntemi çerçeve bloklama, pencereleme, hızlı fourier dönüşümü, mel frekansı saptırması ve kepstrum hesaplaması aşamalarından oluşmaktadır. Bu aşamalara ait sıralandırma Şekil 3.3'te gösterilmiştir.

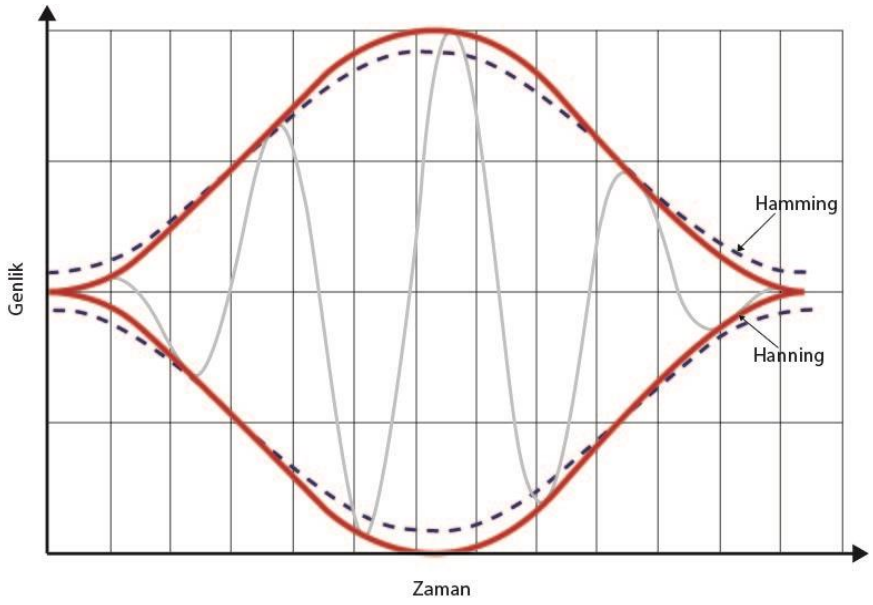


Şekil 3.3. MFCC öznitelik çıkarma aşamaları

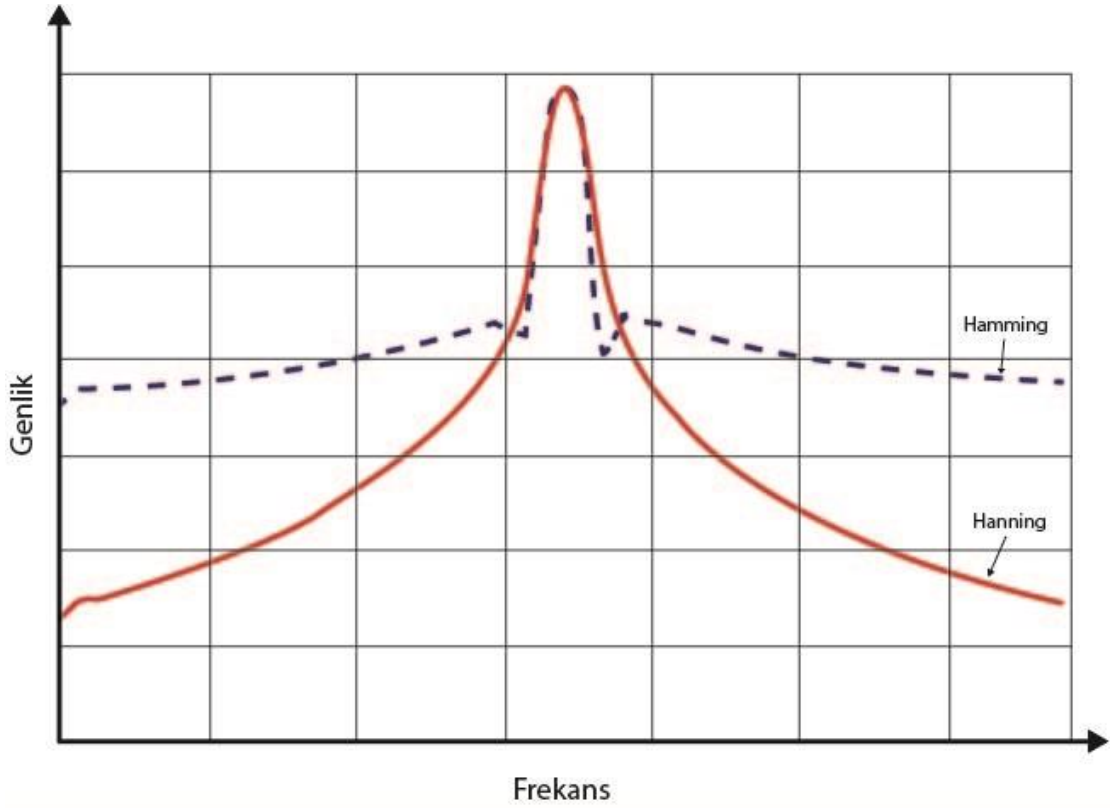
Çerçeveleme aşamasında girdi olarak alınan ses sinyali eşit ve küçük parçalanarak çerçevelenir. Bu işlem küçük parçalar halinde incelenecek sinyalde işlenen seslerin tekil olarak karakteristiklerini ortaya çıkarmak için yapılmaktadır.

Çerçeveleme işlemi sonrası gelen verilerin devam eden adımlarda işlenebilmesi için dağılımları en doğru şekilde yapılmalıdır. Bu sebeple çerçevelenen veriler bir pencereleme fonksiyonundan geçirilerek, bu verilerden elde edilen spektrumdaki frekans çözünürlüğü iyileştirilir.

Ses tanıma sistemlerinde pencereleme fonksiyonları arasında Hamming ve Hanning Pencereleme fonksiyonları en sık kullanılan fonksiyonlardır. Hamming ve Hanning fonksiyonları benzer hesaplamalar yapmakla beraber en temel farklılıkları Şekil 3.4 üzerinde görüleceği üzere Hanning fonksiyonu sinyal başı ve sinyal sonu verilerini sıfıra bastırmayı hedeflerken, Hamming fonksiyonu sinyal başı ve sinyal sonu verilerini sıfara yaklaştırmayı hedefler. Hamming ve Hanning fonksiyonlarına ait spektrum frekans/büyüklük değerleri Şekil 3.5 üzerinde incelendiğinde iki fonksiyonun özellikle tepe noktası hesaplarının neredeyse aynı olduğu görülebilmektedir. Bu noktada Hamming fonksiyonu yan lob gürültü bastırma sonuçlarının daha iyi olması sebebiyle otomatik ses tanıma sistemlerinde daha fazla kullanılmaktadır.



Şekil 3.4. Hamming ve Hanning Pencereleme Fonksiyonu uygulamaları



Şekil 3.5. Hamming ve Hanning Pencereleme Fonksiyonu sonuçları

Bu yaklaşıma en uygun olduğu düşünülerek işlemler sırasında Hamming Pencereleme Fonksiyonu kullanılmıştır. Hamming fonksiyonu Denklem (3.2)'de gösterilmiştir. Fonksiyon üzerinde gösterilen *t* değeri ayrık zamanı ve *N* değeri pencereleme için kullanılan kesitin uzunluğunu temsil etmektedir.

$$P(t) = 0.54 + 0.46\cos\frac{2\pi t}{N} \qquad -N/2 \le t \le N/2 \tag{3.2}$$

Pencereleme işlemi sonrası yapılacak hesaplamalar bu aşamaya kadar zaman uzayında kullanılan verilerin frekans uzayına çevrilmesi gereklidir. Bu sebeple verilerin frekans uzayında işlenebilmesi için elde edilen veriler hızlı fourier dönüşümünden geçirilir. Hızlı fourier dönüşümü için Denklem (3.3)'te gösterile ayrık fourier dönüşümü denklemi kullanılır.

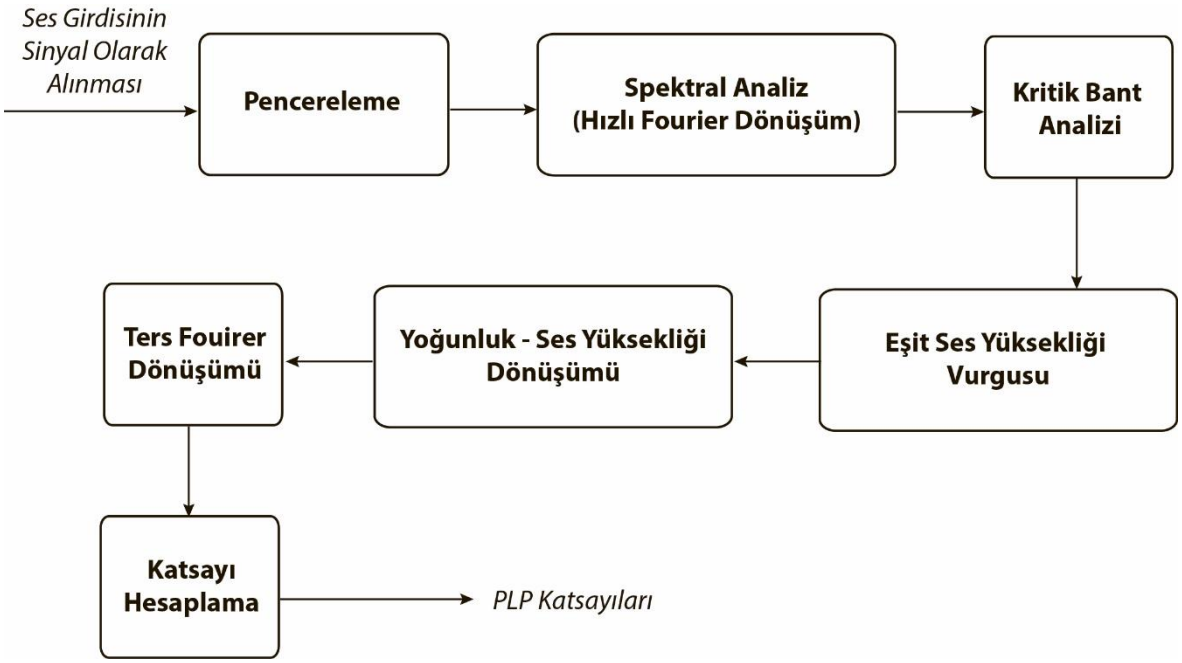
$$x_k = \sum_{n=0}^{N-1} e^{-2\pi i k(n/N)}\, x_n \tag{3.3}$$

MFCC yönteminin temelinde mel frekans ölçekleri bulunmaktadır. Bu sebeple yöntem sonunda elde edilecek katsayılar hesaplanmadan önce şimdiye kadar elde edilen verilen mel frekansı cinsinden temsil edilmesi gereklidir.

MFCC yöntemi kepstrum hesaplamaları yapılarak katsayıların elde edilmesi tamamlanır. Spektral alanda yapılan işlemler sonucunda elde edilen verilerin zaman alanında kullanılabilmesi için tersine fourier dönüşümü yapılır ve bu işleme kepstrum adı verilir.

3.1.1.2. Algısal doğrusal tahmin yöntemi

Algısal doğrusal tahmin yöntemi ses tanıma sistemlerinde kullanılan bir diğer öznitelik çıkarma yöntemidir. Yöntemin temeli insan işitme sürecinin psikofiziksel olarak modellenmesine dayanır (Hermansky, 1990). Algısal doğrusal tahmin yöntemi spektral ve doğrusal tahmin analiz yöntemlerinin birleşik kullanımı ile oluşturulmaktadır. Bu yönteme ait adımlar Şekil 3.6'da gösterilmiştir.



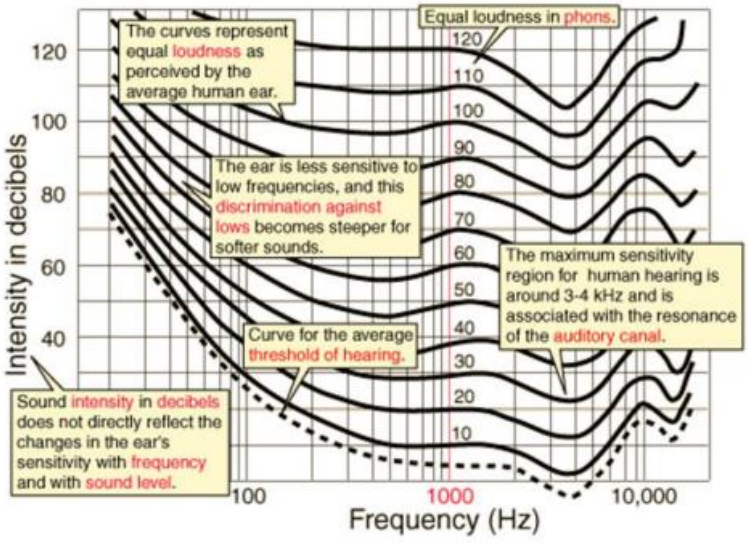
Şekil 3.6. PLP öznitelik çıkarma aşamaları

PLP yönteminde, ses sinyal verilerinin pencereleme işlemi genellikle MFCC adımlarında tanıtılan pencereleme işlemi ile aynı şekilde Hamming pencereleme fonksiyonu kullanılarak gerçekleştirilmektedir.

Spektral analiz aşamasında FFT denklemleri kullanılarak frekans uzayına taşınan veriler güç spektrumlarına çevrilir. Denklem (3.7)'de gösterilen Bark ölçeği yaklaşımı kullanılarak değerler hesaplanır. Denklemde W rad/s cinsinden açısal frekansı temsil etmektedir.

$$\Omega\,(w) = 6\,\ln\left(\frac{w}{1200\pi} + \sqrt{\left(\frac{w}{1200\pi}\right)^2 + 1}\right) \tag{3.4}$$

Bu yaklaşım temelinde insan işitme yapısının modellenmesi yatmaktadır. Bu sebeple hazırlanacak modelde verilerin genel insan işitme aralığı olan 20 Hz ve 20 KHz bandında olması beklenmektedir. Şekil 3.7 üzerinde bir insanın 20 Hz ve 20 KHz bandında belirtilen frekanslarda bir sesi duyabilmesi için gerekli ses yükseliği dB cinsinden ifade edilmiştir. Bark ölçeğine uygulanan değerler sırasıyla kritik bant filtreleme ve eşit ses yüksekliği vurgusu işlemlerine sokulur ve bu işlemler sonucunda veriler insan işitmesine en uygun olan değer aralıklarına indirgenmiş olur.



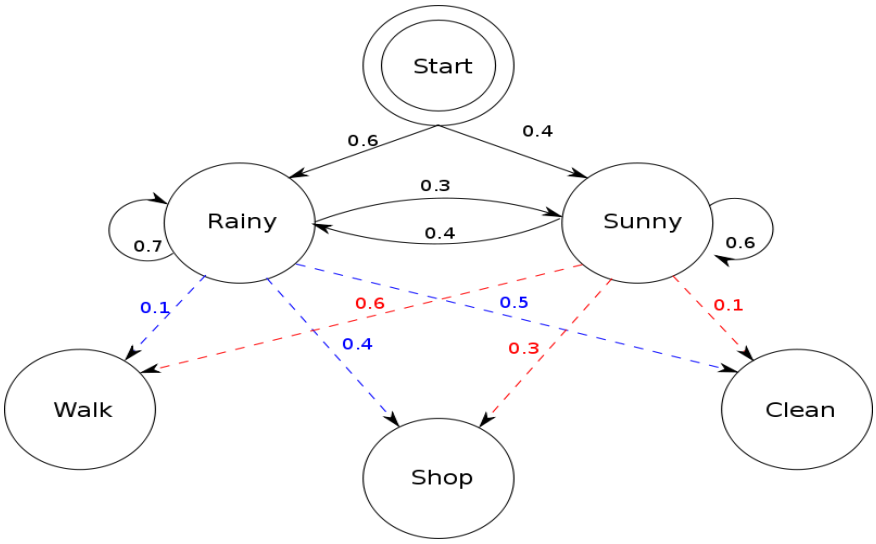
Şekil 3.7. Eşit ses yüksekliği grafiği (Feng, 2012)

Belirtilen adımlar sonucunda elde edilen verileri zaman alanında kullanabilmek için ters fouirer dönüşümü uygulanır ve algısal doğrusal tahmini katsayıları hesaplanarak ses tanımada öznitelik olarak kullanılırlar.

3.1.2. Sınıflandırma yöntemi

Otomatik ses tanıma sistemlerinde sınıflandırma işlemleri ses sinyallarinin analizi sonrası kelime tahmini ve cümle oluşturma aşamalarında kullanılmaktadır. Sınıflandırma yöntemleri hazırlanacak sisteme göre hız ve performans isterleri değerlendirilerek seçilmektedir. Otomatik ses tanıma sistemlerinde genelde olasılık temelli sınıflandırma kullanılır. Sistemlerde genel olarak Bayesian temelli sınıflandırma metodu olan Hidden Markov Model kullanılmaktadır. Sınıflandırmadaki doğruluk, sınıflandırma aşamasında dil modelleri kullanılarak arttırılabilmektedir.

HMM istatistiksel bir modeldir ve basit bir DBN (Dinamik Bayesian Network) olarak tanımlanabilir. HMM ses tanıma sistemlerinin yaygınlaştığı ilk dönemlerden itibaren sıklıkla kullanılan bir ses tanıma yöntemidir. Bu model ve matematiksel mimarisi Rus matematikçi Markov'un yirminci yüz yılın başında sunmuş olduğu Markov Zincirleri çalışmalarından temel alınarak, Leonard Esau Baum ve arkadaşları tarafından hazırlanmıştır (Matfeld, 2014).



Şekil 3.8. Markov Zinciri gösterimi (Kang, 2017)

Şekil 3.8.'de görüleceği üzere Markov Zinciri Modeli matematiksel olarak olayların gerçekleşme olasılıklarını kullanarak bir sonraki eylemin ne olması gerektiğini tahmin etmek üzere kurgulanmıştır. Oluşturulan model geçmiş verileri olasılık işlemlerinde kullanmaz; bu sebeple modelin bir hafıza değişkeni bulunmamaktadır. Ancak olaylar arası geçiş olasılıkları zaman ve duruma göre değişiklik gösterebilirler bu değişiklikler matematiksel olarak sonuçta tespit edilebilseler de adım olarak gözlemlenemezler; bu sebeple bu adımlara Hidden (Gizli) adımlar denilmektedir. Gizli adımların bu Markov Model ile birlikte kullanımı ile hazırlanan yeni model, Hidden Markov Model olarak isimlendirilmiştir.

HMM durumları için Denklem (3.5)'de geçiş olasılıklarının, Denklem (3.6)'de gözlem olasılıklarının dağılımları verilmiştir. Denklem (3.5) ve (3.6)'de kullanılan değerler şu şekilde açıklanmaktadır;

- N modeldeki durum sayısını temsil etmektedir. Modeldeki durumlar $S = \{ S_1, S_2, S_3, \ldots, S_N\}$ ve t anında q_t olarak gösterilmiştir.
- M modeldeki her bir durum için gözlemlenen farklı gözlem sayısını ifade etmektedir. Modelde muhtemel gözlem setleri $V = \{0, 1, 2, \ldots, M-1\}$ olacak şekilde kullanılmıştır.
- Modele ait başlangıç durum dağılımı Denklem (3.7)'de gösterilmiştir.

$$a_{ij} = P\left(q_{t+1} = S_j \mid q_t = S_i\right) \;\; 1 \leq i, j \leq N \tag{3.5}$$

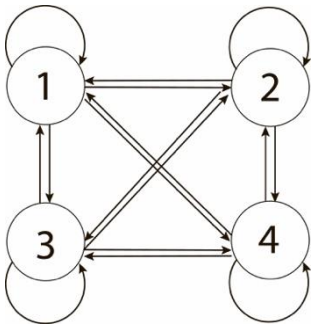
$$b_j(k) = P(t\ anında\ V_k \mid q_t = S_j) \;\; 1 \leq j \leq N, 1 \leq k \;\leq M \tag{3.6}$$

$$\pi = \pi_i$$
$$\pi_i = P(q_1 = S_i), \quad 1 \leq i \;\leq N \tag{3.7}$$

Açıklanan HMM olasılıksal hesaplama denklemleri ile elde edilen veriler toplanarak oluşturulan a, b ve π setleri kullanılarak bir HMM modeli temsil edilebilir.

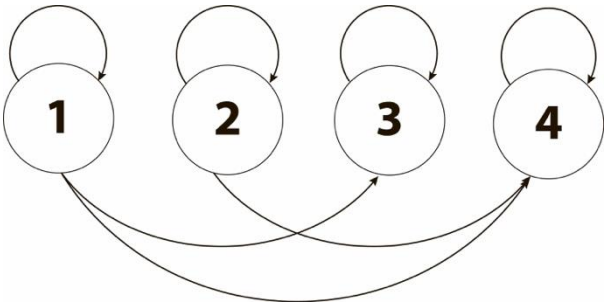
HMM modellerinde kullanılan markov zincirleri kurulmak istenen sisteme göre farklılık gösterebilmektedir. Ses tanıma sistemlerinde yaygın olarak kullanılan ergodik ve soldan sağa olmak üzere iki farklı HMM çeşidi vardır. Ergodik Model Markov Zinciri'nde

Şekil 3.9.'de gösterildiği gibi temsil edilen tüm olasıkların bir biri ile ilişkili olması durumudur. Bu sebeple ergodik model aynı zamanda tam bağlı model olarak da isimlendirilir.



Şekil 3.9. Ergodik HMM Gösterimi

Soldan Sağa Model Markov Zinciri'nde temsil edilen olasılıkların Şekil 3.10.'de gösterildiği gibi soldan sağa olacak şekilde bir biri ile ilişkili olması durumudur. Soldan Sağa Markov Model geçmiş verilere istenilen derecede bağlı olması sebebiyle fazla bağlantılı olasılık değerlendirme işlemi yapılmayacağı için sık tercih edilen bir modeldir.



Şekil 3.10. Soldan Sağa (Bakis) HMM Gösterimi

HMM modellerinde çalışma yaparken birçok veri işleme uygulamalarında olduğu gibi öznitelik çıkarma işlemleri tüm süreç içerisinde önemli bir yere sahiptir. Ses tanıma sistemlerinde MFCC ve PLP genel olarak başarılı şekilde kullanılan öznitelik çıkarma yöntemlerindendir.

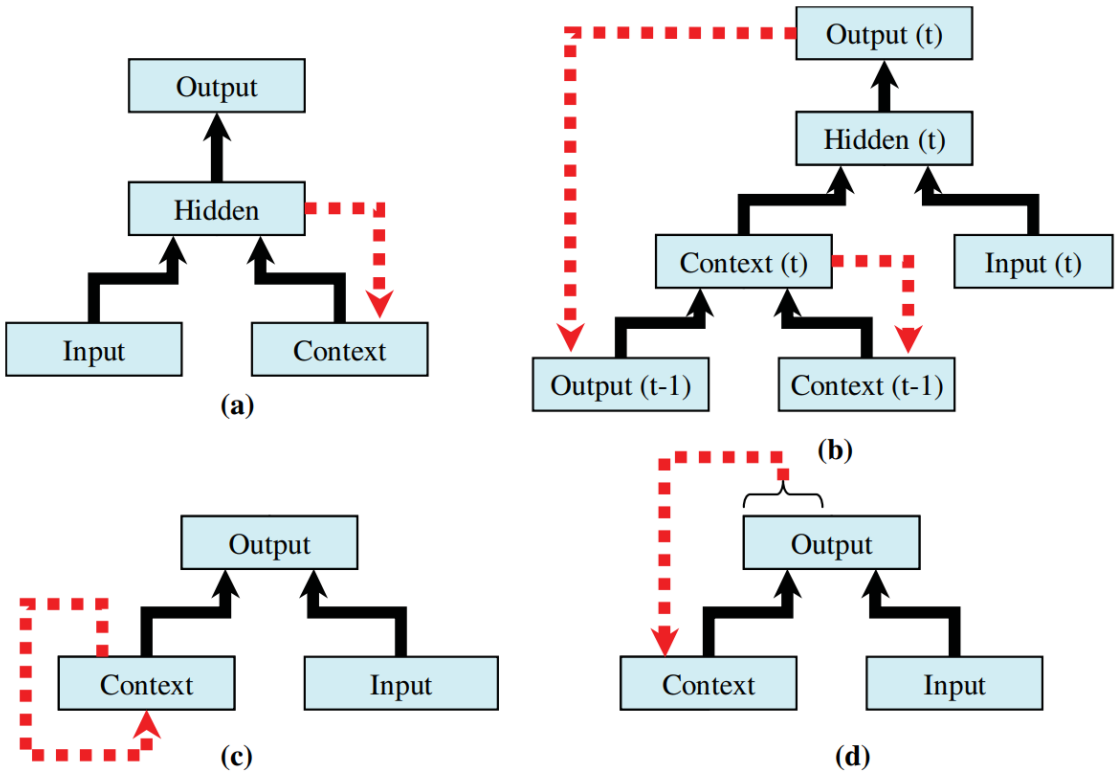
3.1.3. Hidden markov ve derin öğrenme hibrit model

Teknolojik gelişmeler ile beraber yapay zekâ ve derin öğrenme uygulamarının kullanımınında da artış görülmüştür. İhtiyaçlar doğrultusunda ses tanıma sistemlerinin uygulanabilirlik ve kullanılma oranlarını yükseltmek için yapay zekâ ve derin öğrenme metotlarından destek alınmıştır. HMM ile beraber derin öğrenme metotlarının kullanımı ile oluşan yeni modellere hibrit model adı verilmiştir.

Hibrit modellerde derin öğrenme metotları ses tanıma işlem adımlarında değiştirilerek kullanılabilmektedir. Kurulmak istenen sistemin isterlerine göre alternatif modeller oluşturulabilmektedir.

Derin öğrenme metotları, eğitim aşamalarında klasik HMM'e oranla performans olarak geride kalmaktadır. Ancak derin öğrenme metotları geçmiş verilerden yararlanarak tahminlerini güncellemesi sebebiyle klasik HMM'e oranla daha doğru sonuçlar verebilmektedir. Bu sebeple derin öğrenme metotları ve HMM bir birlerinin eksik kaldıkları noktalarda daha başarılı sonuçlar elde etmek için beraber kullanılabilir. Oluşturulan bu yeni sisteme hibrit sistem adı verilir. Hibrit sistemin klasik metoda oranla daha iyi performans ile daha doğru sonuçlar verdiği görülmüştür (Rallabandi vd., 2015).

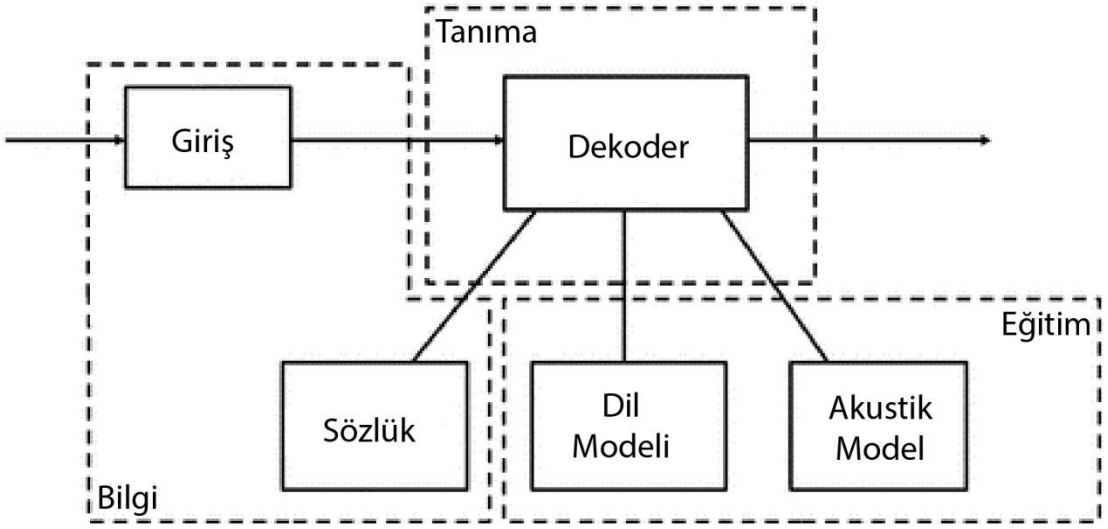
Şekil 3.11'de görülebileceği gibi ses tanıma sistemlerinde çok farklı çeşitlerde RNN modelleri kullanılabilmektedir. Elman RNN, gizli katman verileri ile içerik bilgisini harmanlayarak kullanmayı temel almaktadır. Jordan RNN, içerik bilgisine ait verileri geçmiş içerik bilgileri ve geçmiş sonuç verileri kullanarak eğitmeyi temel almaktadır. Robinson and Fallside RNN ile iki katmanlı bir model ortaya konulmuştur. Bu model içerik bilgisine ait verileri geri bildirim olarak kullanmayı temel almaktadır. Williams ve Zipser RNN modelde yine iki katmanlı bir model ortaya konulmuştur. Bu modelde sonuç verileri içerik bilgilerinin değerlendirildiği adımda geri besleme olarak kullanılmaktadır. Ses tanıma sistemlerinde isterlere göre çeşitli avantajlar sağlayan farklı metotlar kullanılabilmektedir.



Şekil 3.11. Hibrit model gösterimi (a) Elman, (b) Jordan, (c) Robinson and Fallside, (d) Williams and Zipser RNN model (Rallabandi vd., 2015)

3.1.4. Uçtan uca ses tanıma modeli

Ses tanıma sistemlerine ait litaratürlerde son dönemde sıkça görülmeye başlayan uçtan uca ses tanıma modeli; dilden, konudan ve konuşmacıdan bağımsız işlemlerin başarıyla tamamlanması için geliştirilmeye çalışılan bir modeldir. Uçtan uca ses tanıma modelinin temel özelliği Şekil 3.12'de gösterilen tüm ses tanıma adımlarında yapay zekâ veya derin öğrenme metotlarını kullanmasıdır. Derin öğrenme metotları bu adımlarda klasik modellerde kullanılan metotları desteklemek için kullanılabileceği gibi tek başına da kullanılabilmektedir.



Şekil 3.12. Uçtan uca ses tanıma model adımları (Miao, 2017)

Derin öğrenme metotları, kullanıldığı sistemlerde kullanıcı bağımlılığını en aza indirgemesi ile ön plana çıkmaktadır. Sistem kurulurken karşılaşılabilecek hataları en aza indirgeyerek daha istikrarlı ve hatalara dayanıklı sistemler kurulabilmesine olanak sağlamaktadır.

Oluşturulan modelde derin öğrenme metotlarının kullanımından dolayı sistemin yeterli miktarda veri ile eğitilebilmesi büyük önem arz etmektedir. Bu noktada yeteri miktarda veriyi eğitmek performans açısından sistemi zorlayarak süreçlerin uzamasına sebebiyet verecektir. Ancak yapay zekâ işlemlerinde GPU kullanımının artması ile beraber bu işlem sürelerinde önemli miktarda azalma gözlemlenmiştir.

Baidu araştırma ekibinin sunmuş olduğu uçtan uca ses tanıma modeli, derin öğrenme metotlarından RNN kullanılarak geliştirilmiştir. Yapılan çalışmalardan elde edilen sonuçlar incelendiğinde tanıma sistemlerinin uçtan uca modellenerek aktif olarak kullanılabilecek seviyelere geldiği görülebilmektedir (Amodei, 2016).

DeepSpeech modelinin Şekil 3.13'da belirtilen veri setlerini kullanarak elde ettiği kelime hata oranı (WER) sonuçlarının insanların aynı verileri dinleyerek elde ettiği sonuçlara yakın olması bu ve benzeri sistemler üzerine yapılan çalışmalarının sonuçlarına olan güvenilirliği arttıracaktır.

	Test Seti	DeepSpeech2 (WER %)	Mechanical Turk (WER %)
Okuma	WSJ 92	3.10	5.03
	WSJ 93	4.42	8.08
	LibriSpeech test-temiz	5.15	5.83
	LibriSpeech test-diğer	12.73	12.69
Aksanlı	VoxForge Amerikan-Kanadalı	7.94	4.85
	VoxForge Genel	14.85	8.15
	VoxForge Avrupalı	18.44	12.76
	VoxForge Hintli	22.89	22.15
Gürültülü	CHiME Gerçek	21.59	11.84
	CHiME Sim	42.55	31.33

Şekil 3.13. DeepSpeech uçtan uca ses tanıma modeli kelime hata oranı sonuçları (Amodei, 2016)

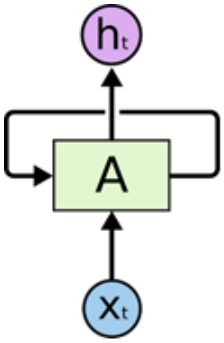
3.1.5. Tekrarlayan Yapay Sinir Ağları (RNN)

Tekrarlayan yapay sinir ağları bir yapay sinir ağı alt sınıfıdır. Yapay sinir ağlarında kullanılan düğümler arasında döngüsel bağların kurulduğu bir modeldir. İleri beslemeli sinir ağlarından farklı olarak, RNN'ler kendi giriş belleklerini, girdileri işlemek için kullanabilirler.

RNN model olarak doğal öğrenme yöntemlerinden tecrübe ile öğrenmeyi temel almaktadır. İnsanlar her adımda öğrenmeye yeniden başlamazlar, her adımda eski tecrübelerinden yararlanarak öğrenmeye devam ederler. Ancak geleneksel yapay sinir ağlarında insanlarda bulunan bu tecrübe ile anlamlandırma özniteliği bulunmaz ve bu onların en büyük eksikliğidir. Örneğin, videodaki tüm karelere bakarak aktiviteler sınıflandırmak istendiğinde, geleneksel sinir ağları kareler arasında insanlar gibi anlamlandırma kuramadığından, sınıflandırma yapamayacaktır.

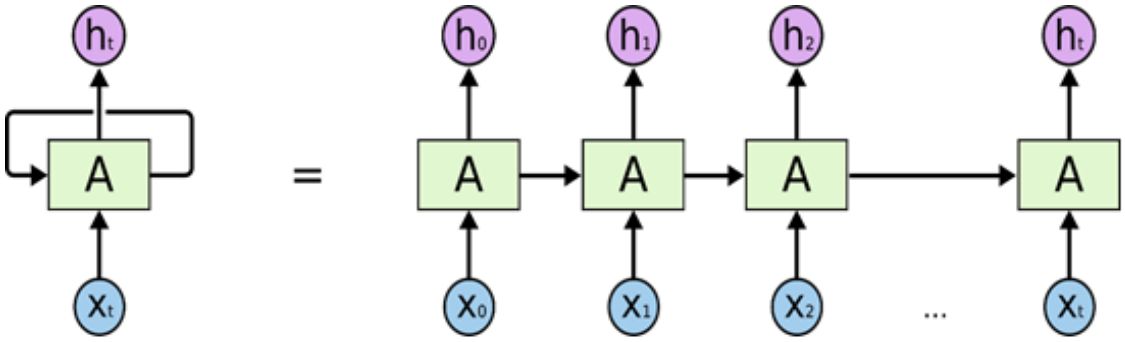
Tekrarlayan Yapay Sinir Ağları ise bir döngü oluşturarak, geçmiş bilgilerin kullanılmasını sağlayacak ve böylelikle kareler arasında anlamlandırma yaparak sınıflandırma yapabilecektir.

Şekil 3.14'de basit tekil tekrarlayan sinir ağ görüntülenmektedir. *A* ismi verilen dikdörtgen bir yapay sinir ağındaki düğümü temsil etmektedir. Ağın girdi değeri *X*'dir. Yapay sinir ağının çıktı değeri *h*'dir. Düğümden değerlendirme sonucu çıkan değer yine kendisine dönerek, döngüsel bir eğitim modeli oluşturmaktadır. Bu döngü ile geçmiş verilerde kullanılabildiğinden yeni bilgi, eski bilgi harmanlanarak bir sınıflandırma yapılabilmektedir.



Şekil 3.14. Tekil tekrarlayan sinir ağ gösterimi (Olah, 2015)

Zaman diliminde, aynı hücre kendini birden fazla tekrar edebilmektedir. RNN uzun süreçte tekrarlayan adımlarda kullanılarak daha detaylı öğrenme imkânı sunabilmektedir. Şekil 3.15'de tekil tekrarlayan yapay sinir ağlarının geçmiş verileri geri besleme olarak her adımda kullanması görselleştirilmiştir.

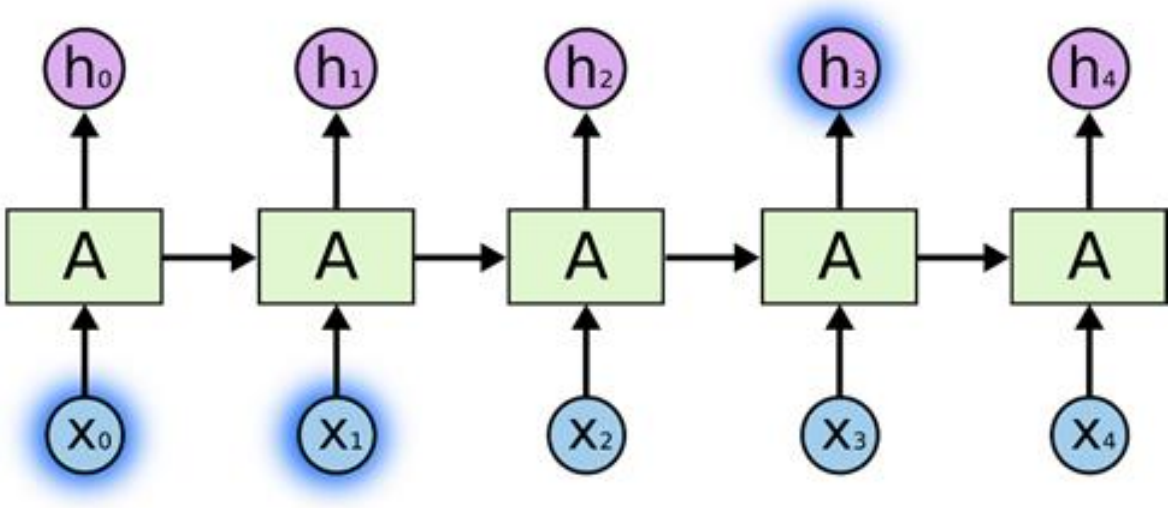


Şekil 3.15. Tekil tekrarlayan sinir ağ geçmiş veri kullanımı (Olah, 2015)

RNN'ler döngüsel olarak çalıştıklarından, sıralı gelişen aktiviteleri birbirleriyle anlamlandırabilmektedir. Akış içerisindeki aktivitelerin anlamlandırılarak yüksek doğrulukla sınıflandırabilmesinden dolayı son yıllarda kullanıcı etkileşimleri, dil modelleme ve ses tanıma sistemlerinde yaygın olarak kullanılmaktadır.

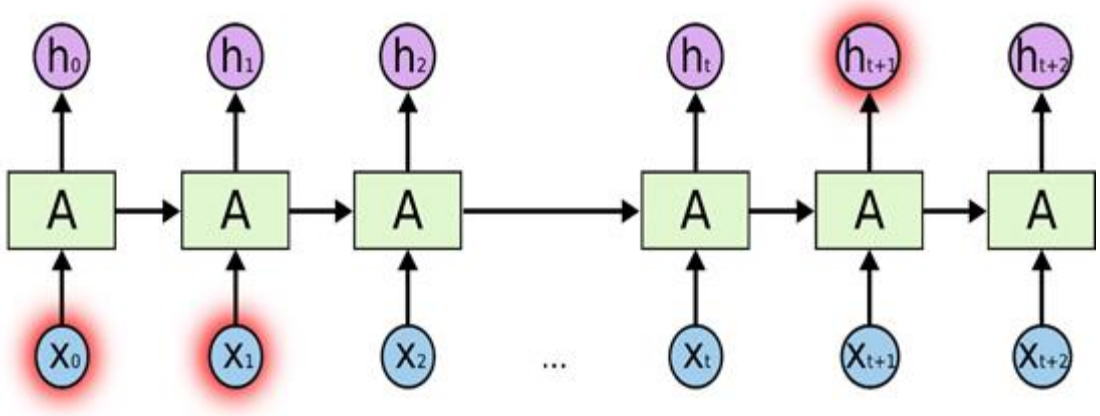
RNN ile oluşturulan sistemler geçmiş verileri kullanarak zaman temelli problemlerde başarılı sonuçlar vermektedir. Ancak bu sistemlerde hangi aktiviteler ne kadar süre ile hatırlanacak bilinmemektedir. Bütün bilgiler, modelin içerisinde tutulmaktadır. Aktiviteler için bazı bilgiler önemliyken, bazı bilgiler gereksizdir. Bu yüzden bazı aktivitelerin sınıflandırılmasında, tüm geçmişin saklanmasına gerek yoktur.

Aktivite sınıflandırılmasında, gerekli bilgi çok önceden oluşmuş ise, bu bilgiye ulaşılamayabilir. Bazı durumlarda ise, bir önceki karedeki olay ile şimdiki karedeki olay birbiri ile bağlantılıdır. Örneğin, aktivite sınıflandırılması yapılacak olan videoda, sırasıyla bazı kişiler yemek masasına oturuyor olsun; böyle bir videoda, önceden masaya oturmuş kişilerden, bir sonraki gelen kişinin de masaya oturacağını tahmin etmek RNN için zor değildir. Çünkü bir sonraki karenin tahmin edilebilmesi için gerekli olan masaya oturma aktivitesi, çok yakın zamanda gerçekleşmiştir. Şekil 3.16'de hafızadan hatırlanabilir bir RNN gösterilmektedir.



Şekil 3.16. RNN'nin geçmiş verileri kullanarak hatırlama işlemi gösterimi (Olah, 2015)

Bazı aktiviteler, yemek masasına oturma aktivitesinden çok daha karmaşıktır. Örneğin, sınıflandırma yapılacak video, yemek masasında oturulup yemeğe başlandıktan sonra gelen bir kişinin aktivitesi olsun. Yemeğe geç gelen kişinin, yemeğe oturacağının tahminin yapılması insanlar için zor değildir. Fakat RNN için yemeğe oturma aktivitesi bitip, aralara başka aktiviteler girdiği için, yeni kişinin yemeğe oturacağını tahmin etmek zor olmaktadır. Şekil 3.17'de hafızadan hatırlama için bir engel olan ara olayların artması durumu görselleştirilmiştir.



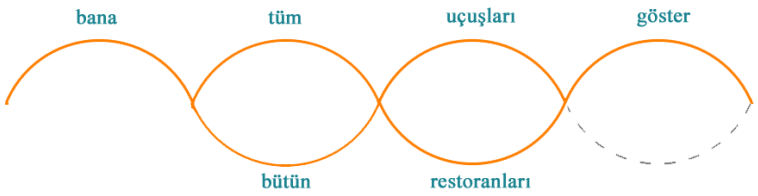
Şekil 3.17. RNN fazla geçmiş veri kullanımının olumsuz etkileri (Olah, 2015)

Teoride, RNN'lerin uzun geçmişteki aktiviteleri, iç mimarisinde kendini tekrarlamakta olduğu için, hatırlayabilme kapasitesine sahiptir. Ancak hatırlanabilmesi için parametrelerin titizlikle seçilmesi gerekmektedir.

3.2. Dil Modeli Oluşturma Yöntemleri

Dil modelinin oluşturulması ve kullanılması ses tanıma sistemlerinin ana basamaklarından biridir. Ses tanıma sistemleri, ses girdisi işlenerek ardışık kelime gruplarının olasılığı en yüksek olanını seçmek üzerine kurulmaktadır. Bu tahmin aşamasında dil modelleri kullanılmaktadır. Dil modelleri içerdikleri verileri kullanarak ardışık kelimelerin veya kelime gruplarının olasılıksal tahminlerini oluşturmaktadır. Ses tanıma sistemleri ile beraber kullanılan birçok dil modeli bulunmaktadır. Bu dil modellerinden N-gram ve türevleri, kullandığı veriler ile kelime olasılıklarını hazırlaması sebebiyle daha çok veride daha doğru sonuçlar vermesi, kalıp sözcükleri daha kolay öğrenebilmesi ve

matematiksel olarak kolay ifade edilerek sistemlere kolay adapte olabilmesi ile ön plana çıkmaktadır. Şekil 3.18.'de dil modelinin karar adımları örnek üzerinde gösterilmiştir.



Şekil 3.18. Dil modeli karar basamakları

3.2.1. N-gram dil modeli

N-gram dil modeli, ses verisinden elde edilen kelimenin veya kelime gruplarının istatistiksel olarak değerlendirilerek sonrasında gelecek kelimenin tahmin edilmesi yöntemidir. Modelin matematiksel gösterimi Denklem (3.8) üzerinden incelenebilir.

$$P(W_1, W_2, \dots, W_m) \approx \prod_{i=1}^{m} P(W_i \,|W_{i-n+1}, \dots, W_{i-1}) \tag{3.8}$$

Takip eden kelimelerin tahmini için bir önceki kelimeyi kullanmak yanlış sonuçlar doğurabilmektedir. Bu hatalı sonuçların en aza indirilebilmesi için tahmin edilecek kelimeden önce birden daha fazla kelime grubunu incelemek gerekmektedir. N-gram modelde N değeri bu tahmin için kullanılması gereken geçmiş adım sayısını ifade etmektedir.

3.2.2. Fonem tabanlı yaklaşım

Ses tanıma sistemlerinde karakter tabanlı dil modeli harflerin okunuşlarını gelen veriler ile eşleştirerek karakter tahmini yapan bir yöntemdir. Bu yöntem ile tespit edilen karakterler yan yana dizilerek kelimeleri ve kelimeler de cümleleri oluşturacak şekilde düzenlenmektedir. Konuşmalardan karakterleri ayıklamak zorlayıcı bir görev olsa da bu yöntemle hafıza değişkenlerinden bağımsız bir şekilde elde edilen veriler ile dil bilgisine ait temel kurallar takip edilerek doğru sonuçlar elde edilebilmektedir (Amodei vd., 2016)

3.3. Gürültüyü En Aza İndirgeme Yöntemleri

Çevresel etkenler, ses tanıma adımlarında girdi verilerini bozarak yanlış sonuçlar elde edilmesine sebebiyet verebilmektedirler. Çevresel etkenlerin en aza indirgenmesi için ses verilerini toplamak için kullanılan cihazlar gürültü engelleme özelliklerinin olmasına göre seçilebilir. Ekipman katmanında bu etkenler azaltılabileceği gibi farklı yöntemler de kullanılabilir. Gürültü etkisini azaltmak için gürültüye dayanıklı özniteliklerde kullanılabilir (Pardede, 2015). Ayrıca gelen ses verisinin ses tanıma adımlarına gönderilmeden önce derin öğrenme metotları ile desteklenen bir çözücü yardımıyla yeniden işlenmesi sağlanabilir. Bu yöntem kullanılarak yapılan araştırmalarda 5 dB, 10 dB, 15 dB ve 20 dB gürültü değerleri kullanılarak hazırlanan verilerle yapılan testlerde başarılı sonuçlar elde edilebilmiştir (Maas vd., 2012). Farklı bir yaklaşım olarak, elde edilen verilerde daha önceden belirlenmiş anahtar kelimeler aranarak incelenmek istenen sekans ayrıştırılabilmektedir ve bu şekilde istenen veriye ait değerler gürültüden ayrıştırılarak kullanılabilir olmaktadır.

3.4. Derin Öğrenme Araçları

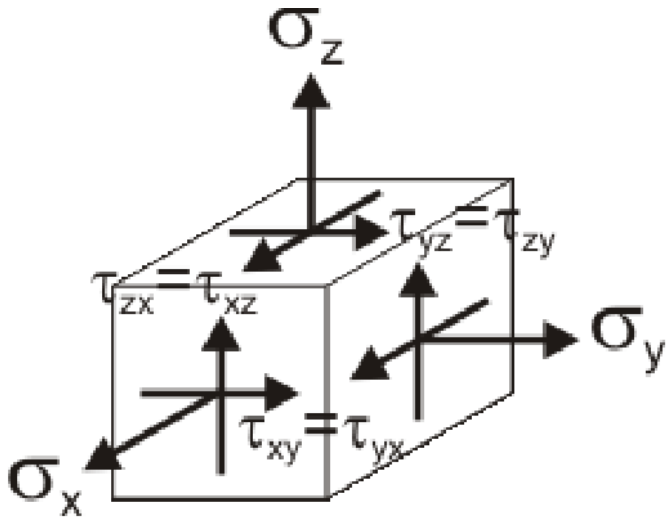
Derin öğrenme, hazırlanacak ses tanıma sisteminin her aşamasında kullanılabilir bir metot olarak karşımıza çıkmaktadır. Ses tanıma sistemlerinde geçmişe dönük bilgilere ihtiyaç duyulması sebebiyle derin öğrenme metotlarından RNN kullanılmaktadır. Sistemin çalışma performansı, sistemin minimum isterleri ve perfromans sonuçlarını optimum seviyede tutmak için uygulamada genel olarak 5 katmanlı RNN modeli kullanılmaktadır.

Otomatik tanıma sistemi için kullanılacak dil modeli eğitiminde Tensorflow kullanılabilir. Hazırlanan dil modelini kullanarak yapılacak ses tanıma işlemlerinde yapılan işlemleri hızlandırmak için CUDA gibi yazılımlardan yararlanılmaktadır.. Bunların yanında ses tanıma sistemi Colaboratory üzerinde oluşturulan bilgisayar ortamı üzerine çalıştırılmıştır.

3.4.1. Tensorflow

Tensorflow 2015 yılında Google Brain ekibi tarafından geliştirilmiştir (Abadi vd., 2015). Genel kullanımda olan makine öğrenmesi ve derin öğrenme metotlarını bünyesinde bulundurmaktadır. Bu metotların mimarisinde C++ dili kullanılmış olsa da, kullanımını kolaylaştırmak için hazırlanan bir uç birim yardımıyla Python kodları çalıştırabilmektedir. Python etkili bir script dili olmasının yanında platformdan bağımsız birçok geliştiricisi de bulunmaktadır. Tensorflow bu sayede yapay zekâ ve derin öğrenme geliştiricilerinin birçoğunu kendi bünyesindeki çalışmalara çekerek büyük bir geliştirici kitlesi oluşturmuştur.

Tensorflow, adından anlaşılacağı üzere tensörler üzerine kurulmuş bir yapıdır. Tensörler skaler büyüklükler, vektörel büyüklükler ve diğer tensörler arasındaki ilişkileri tanımlayan nesnelerdir (Anonim, 2008). Tensörler geometrik olarak bir sayısal değerin çok boyutlu bir dizisi olacak şekilde Şekil 3.19'daki gibi gösterilebilir. Bir dizin tensör ile temsil edildiğinde, dizinin boyutu tensörün derecesini göstermektedir. Tensörlerin temsil ettiği değerlerin boyutlarına göre Çizelge 3.1 üzerinden incelenebileceği üzere özel isimlendirmeleri bulunmaktadır.



Şekil 3.19. Tensör Geometrik Gösterimi (Durán vd., 2012)

Çizelge 3.1. Tensör Derece Gösterimi

Skaler Sayılar	**0. Derece Tensör**
Vektörler	**1. Derece Tensör**
Matrisler	**2. Derece Tensör**
***N* Boyutlu Değer**	***N*. Derece Tensör**

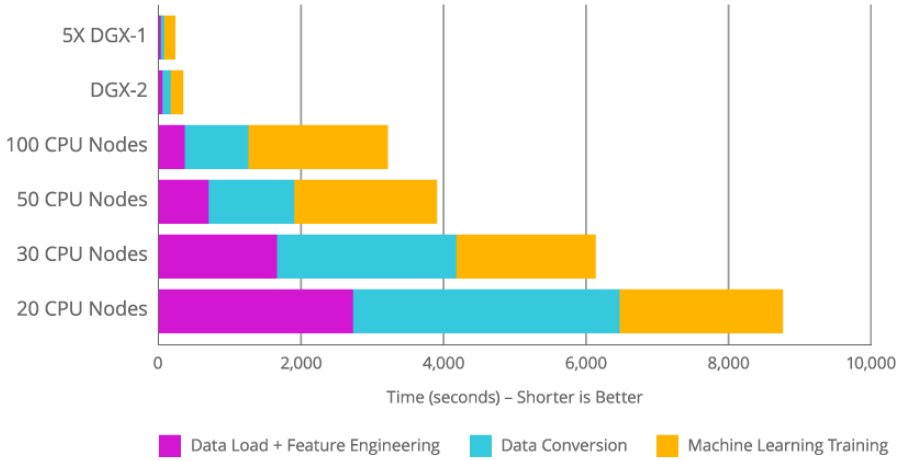
3.4.2. CUDA

Yapay zekâ işlemlerinde kullanılan bilgisayarların işlem kabiliyetlerine göre süreçlerin performans değerleri değişkenlik gösterebilmektedir. Yapay zekâ sistemleri tüm aşamalarında yüksek mertebede matematiksel işlemler kullanmaktadır. Bu durumda genel olarak kullanılan CPU performasları yeterli olmamaktadır.

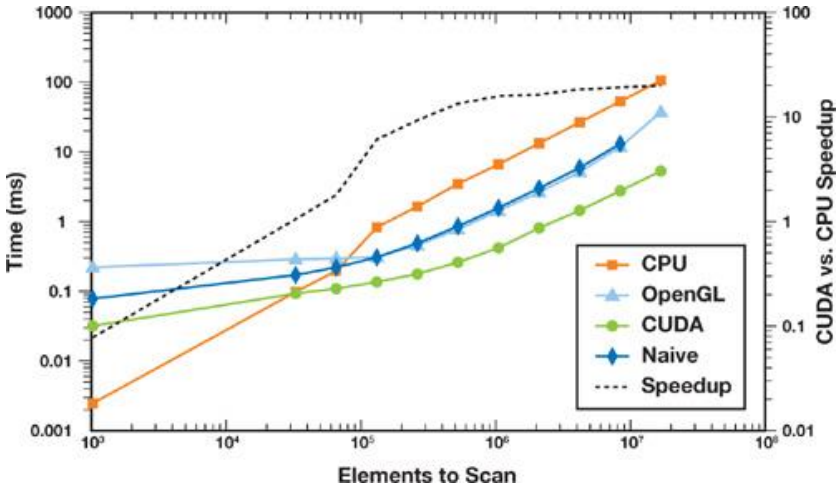
Genel olarak grafik işlem birimleri ile piyasada önemli bir yere sahip olan NVIDIA bu konu üzerinde alternatifler oluşturmak için çalışmalar yapmaktadır. CUDA, NVIDIA'nın bu çalışmalar sonucunda tanıttığı GPU gücünü kullanarak hesaplama perforamsını yükselten paralel hesaplama mimarisidir.

CUDA hesaplama performanslarını önemli ölçüde arttırması sebebiyle yapay zekâ uygulamalarının neredeyse hepsinde aranan bir özellik olarak karşımıza çıkmaktadır. Bu sebeple, ses tanıma sistemlerinde yapılan matematiksel işlemlerin getirdiği iş yükünü azaltması amacıyla tez çalışması kapsamında CUDA mimarisi kullanılmıştır.

Şekil 3.20 ve Şekil 3.21 incelendiğinde işlem sırasında kullanılan veri miktarı arttığında CUDA'nın açık kaynaklı CPU uygulamarından daha kısa sürede işlemleri tamamladığı görülmektedir.



Şekil 3.20. CUDA Performans Grafiği (Prasanna, 2018)



Şekil 3.21. CUDA Performans Grafiği (Harris, 2007)

3.4.3. Colaboratory

Colab (Colaboratory) kâr gözetmeyen bir grup olan Jupyter'in geliştirmeye başladığı çalışmaları temel alan Google'ın bir uygulama aracıdır (Nóbrega, 2018). Colaboratory temel olarak Jupyter notebook ortamını kullanmaktadır. Colab, browser üzerinde Google bulut servislerine bağlanarak işlem yapabilmeye ve hazırlanmış olan kodları kolayca sanal bir bilgisayar üzerinde çalıştırabilmeye olanak sağlamaktadır.

Colab, araştırma yapılırken kullanılmak istenen yüksek performas sağlayan bilgisayarlarda bir program kurulmasına gerek kalmadan, mevcut bilgisayarın özelliklerini kullanarak bünyesinde bulunan programların kullanılmasını sağlayabilmektedir. Bu sayede bilgisayarınızın performansını kullanırken, bilgisayara program kurmak için gerekli olabilecek prosedürlerin ortaya çıkaracağı problemlerin önüne geçilmiş olur.

Aynı zamanda mevcut kullanılan bilgisayarların özelliklerinin yetersiz olduğu durumlarda Colab kullanıcılara kullanmak üzere anlık olarak sanal bir makine hazırlayabilmektedir. Gelişen teknoloji ile beraber sanal cihazlarda kullanılan donanımsal ve yazılımsal özellikler değişkenlik gösterebilmektedir. Sanal makinenin özellikleri şu şekildedir;

- GPU: 1 adet Tesla K80 , 2496 adet CUDA çekirdeği , 12GB(11.439GB Kullanılabilir) GDDR5 VRAM.

- CPU: 1x tek çekirdek hyper threaded (1 çekirdek, 2 tred) Xeon İşlemci @2.3Ghz (Turbo Boost Yok) , 45MB Cache.

- RAM: 12.6 GB yaklaşık olarak kullanılabilir kullanılabilir.

- Disk: 320 GB yaklaşık olarak kullanılabilir.

4. DENEYSEL ÇALIŞMALAR

Geleneksel yöntemlerden farklı olarak yapay zekâ teknolojilerinin gelişmesi ile beraber uçtan uca ses tanıma sistemlerinin yaygınlaşması beklenmektedir. Araştırma sonucunda tıptan hukuka çok farklı konu başlıklarında çalışmaya en uygun, uçtan uca ses tanıma işlemlerini gerçekleştirebilen bir modelin hazırlanması ve test edilmesi hedeflenmiştir.

Uçtan uca ses tanımını gerçekleştiren komple bir çözüm olarak DeepSpeech modeli karşımıza çıkmaktadır. Baidu araştırmacılarının başlatmış olduğu çalışmanın ilk sonuçları benzer sistemlere olan güveni arttırmıştır.

Yapay zekâ yatırımları birçok alanda hızlıca artmaya devam etmektedir. Bu alanda genel olarak browser ürünü ile tanınan Mozilla markasınında çalışmaları bulunmaktadır. Baidu araştırmalarını temel alarak geliştirilen Mozilla's DeepSpeech, geliştirme çalışmalarına güçlü bir ekiple devam ettikleri ve açık kaynak olarak destek vermek isteyen kullanıcılara vermiş oldukları dokümantasyon ve altyapı destekleri ile araştırmalarda ön plana çıkmaktadır. Yapay zekâ ve makine öğrenmesi uygulamarında sıkça kullanılan Tensorflow ile bütünleşmiş ve uyumlu olarak çalışan versiyonları da bulunmaktadır (Hannun vd., 2014)

Çalışmalarda oluşturulan ses tanıma sistemi için Mozilla's DeepSpeech temel alınmıştır ve Tensorflow, CUDA ve CuDNN araçları aktif olarak kullanılmıştır. Test yapılma tarihi itibari ile belirtilen araçlara ait sırasıyla 1.13, 10 ve 7.5 versiyonları kullanılmıştır. Uçtan uca olarak çalışması istenen sistem kurulurken ses verilerinin 16 kHz ölçeğinde olması gerekmektedir bu sebeple değerlendirilen ses verileri gerekli olduğu durumlarda up sample ve down sample işlemleri uygulanarak uygun formata çekilmiştir. Bu format kontrol işlemleri için libsox2 kitaplığına ait fonksiyonlar kullanılmıştır. DeepSpeech modeli gizli katmanlarda yapılan işlemlerde RNN metodunu kullanmaktadır. Bu gizli katman sayısı isterlere göre arttırılabilir olmakla birlikte yapılan testlerde için beş katmanlı olarak kullanılmıştır.

Bu tez çalışmasında Google Brain ekibinin hazırlamış olduğu ve KWS çalışmalarında kullanmak üzere topladıkları komut veri seti üzerinde çalışmalar yapılmıştır (Sainath vd., 2015).

Çalışmalar sırasında bilgisayar ortamında kurulan yapay sinir ağlarının kontrolünün sağlanması için gerekli bazı parametreler kullanılmaktadır. Bu parametreler hiper parametreler olarak da isimlendirilmektedir. Yapılan çalışmalarda train, dev ve test batch size, n hidden layer, learning rate, dropout rate ve epoch hiper parametreleri kullanılmıştır.

Derin öğrenme uygulamalarında tüm verilerin işlenmesi aşamalarında birçok matematiksel işlem yapılmaktadır. Bu hesaplamaların tamamlanması için geçen süre hesaplamalarda kullanılan veri boyutu ile doğru orantılı olarak değişmektedir. Bu sebeple derin öğrenme uygulamalarında aynı anda ne kadar verinin işleneceğini belirten batch size parametresi kullanılmaktadır. Batch size parametresi ne kadar küçük seçilirse işlem süresi o kadar kısalacaktır ancak tersi şekilde aynı anda daha az veri ile işlem yapılacağı için sonuçta daha fazla hatalı veri gözlemlenecektir. Bu hesaplamalar da fiziksel işlemcilerden en üst düzeyde performans alabilmek için batch size değeri 2'nin katları şeklinde seçilmelidir. Derin öğrenme uygulamalarında veriler belirli sayıda parçalar halinde eğitimde yer alırlar. İlk parça eğitilir, modelin başarımı test edilir, başarıma göre backpropagation ile ağırlıklar güncellenir. Daha sonra yeni eğitim kümesi ile model tekrar eğitilip değerler tekrar hesaplanarak güncellenir. Bu işlem her bir eğitim adımında tekrarlanır ve model için en uygun değerler hesaplanmaya çalışılır. Bu hesaplamalar da gerçekleştirilen her bir adıma epoch adı verilmektedir. Verilerin doğru tahmin edilmesi için yapay sinir ağlarında oluşturulan düğümlerin veri ve sonuç arasında kurmuş olduğu ilişkisel bağlantılar kullanılmaktadır. Bu düğümlerin veri ile olan ilişkisel bağ da kontrol edilerek belirli bir ilişkisel bağı olmadığı düşünülen düğümler daha doğru sonuçlar elde etmek için derin öğrenme hesaplamalarında kullanılmazlar. Bu ilişkisel bağ için kullanılan sınır değeri parametresi dropout rate olarak isimlendirilmektedir. Kurulan yapay sinir ağında kaç gizli katmanın olacağı n hidden layer parametresi ile belirlenmektedir.(Wilson vd., 2018)

Speech commands dataset TensorFlow ve AIY ekipleri tarafından Google bünyesinde geliştirmiş oldukları ses tanıma sistemlerini test etmek ve geliştirecekleri sistemlerde kullanılabilecek bir kaynak olarak hazırlanmıştır. Kaynak toplama işlemlerinin son halini almasıyla beraber 2017 yılında veri seti herkesin kullanımına açılmıştır. Veri setleri belirtilen katılımcı oluşumların üyeleri tarafından oluşturulmuştur. Bu oluşumların kullanıcılarının çok çeşitli olması sebebiyle veri setinde cinsiyet, yaş ve uyruk fark etmeksizin otuz ayrı kelime için altmış beş bin ses dosyası bulunmaktadır. Veri seti, farklı insanlardan toplanan verileri ortak bir paydada toplayabilmek için İngilizce olarak hazırlanmıştır.

Veri setinde sıfırdan dokuza kadar sayılar ve evet, hayır, git, çalıştır, sağ, sol gibi kelimeler İngilizce olarak birçok farklı kişi tarafından seslendirilmiştir. Bu veriler telefon ve bilgisayar üzerinden kayıt altına alınmıştır. Bu sebeple profesyonel ekipmanlar kullanılmadan alınan bu ses verilerinde ekipmanlardan ve çevreden kaynaklı gürültüler bulunmaktadır.

Ses tanıma işlemleri gerçekleştirilmeden önce Colab üzerinde bir ortam hazırlanmıştır. DeepSpeech ile ilgili gerekli kurulumlar tamamlanmıştır. Test işlemleri Colab Hosted Runtime üzerinde çalıştırılmıştır. Hosted Runtime işlenmesi istenen verilere internet üzerinden erişmek istediği için öncelikle veriler Google Drive aracılığıyla oluşturulan bir bulut hesabına yüklenmiştir ve sonrasında bu veriler Colab ortamı ile paylaşılmıştır.

Sayılara ve kelimelere ait ses verileri örneklemler ile incelendiğinde bazı verilerin gürültü haricinde bir ses verisi içermediği görülmüştür. Bu şekilde karşımıza çıkan veriler sistem tarafından boş olarak kabul edilmiş ve tezde hatalı olarak belirtilmiştir.

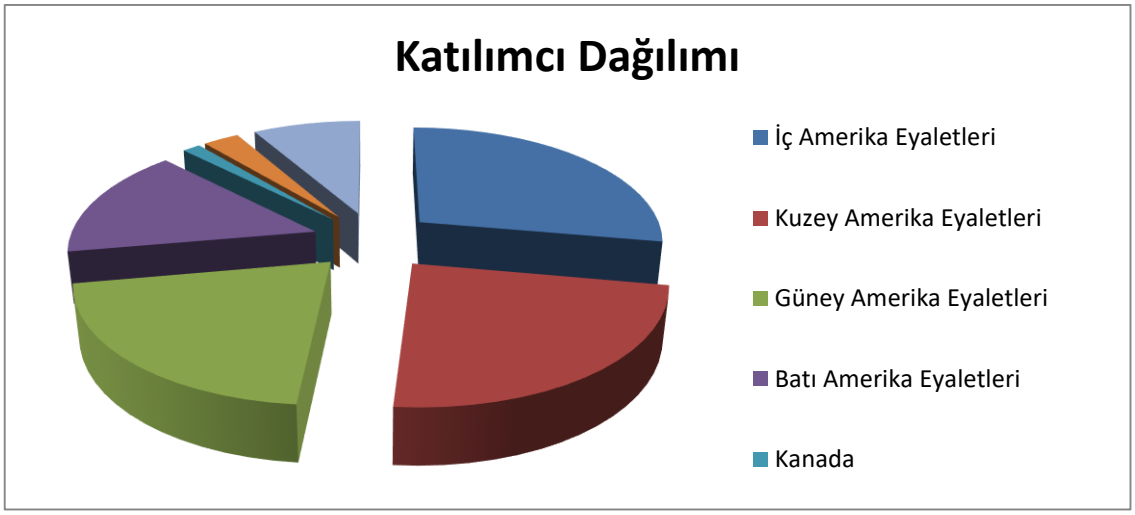
4.1. Test 1

Test 1 ile kurulan uçtan uca modelin performansı sıfırdan dokuza sayılar ve evet, hayır, git, sol, sağ, dur, kapat ve aç temel komutların tanınması ile test edilmiştir. Bu testler Google Brain ekibinin toplayıp kullanıma açmış olduğu Speech Command Dataset kullanılarak yapılmıştır. Bu veri seti çeşitli yaşlardan ve ülkelerden kadın ve erkeklerin katılımıyla oluşturulmuştur. Farklı yaşlardan ve ülkelerden insanların katılımıyla hazırlanan veriler kullanılarak sistemin kullanıcıdan bağımsız çalışma performansı ve kullanılan kapsamlı dil modeli ile içerik bağımsız çalışma performansları test edilmiştir.

Testler için Mozilla Deepspeech üzerinde hazırlanan İngilizce dil modeli kullanılmıştır. Hazırlanan İngilizce dil modeli Fisher, LibriSpeech ve Switchboard eğitim korpusları kullanılarak hazırlanmıştır.

Fisher veri seti, DARPA EARS programı kapsamında kullanılan verilerden derlenmiştir ve ortalama 10'ar dakikalık 16454 telefon görüşmesinden yaklaşık 2742 saatlik veriden oluşmaktadır. Veri seti oluşturulmasında görev alan kişilerin yaş ve cinsiyet dağılımları aşağıdaki gibidir.

- %38'i 16-29 yaşında
- %45'i 30-49 yaşında
- %17'si 50 yaş üstündedir
- %53'ü kadın
- %47'si erkektir



Şekil 4.1. Fisher veri seti katılımcı dağılımı

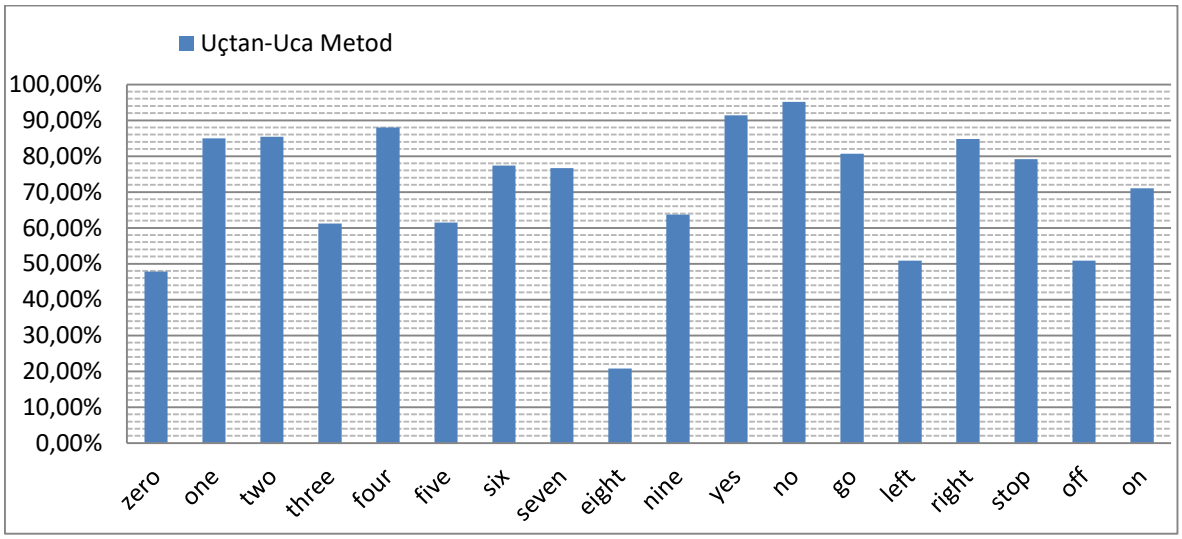
Librispeech eğitim veri seti LibriVox üzerinde toplanan 2956 farklı kitabın 1000 saatlik seslendirme verilerinden oluşturulmuştur. Bu kitapları seslendirenlerin %48'i kadın %52'si erkektir. Korpus açıklamarında seslendirmeciler hakkında daha detaylı bilgiler paylaşılmamıştır.

Switchboard eğitim veri seti Texas Instruments tarafından DARPA destekli bir proje kapsamında hazırlanmıştır. Korpus içerisinde yaklaşık olarak 3 milyon kelime ile oluşturulan 250 saatlik veri bulunmaktadır. Veri toplama işlemleri için ABD'nin çeşitli bölgelerinden Amerikan İngilizcesi konuşan kadın ve erkeklerden yaklaşık 500 kişilik bir katılımcı grubu oluşturulmuştur.

Dil modeli oluşturulurken yapay zeka fonksiyonlarından yararlanılmıştır. Veri setlerindeki verilerin %70'i eğitim, %30'u test için kullanılmıştır. Bu fonksiyonlara ait değişkenler; batch size 24, n hidden 2048, learning rate 0.0001, dropout rate 0.15, epoch 75 olacak şekilde kullanılmıştır.

Testlerde kullanılan dil modeli genel konuşma metinleri kullanılarak oluşturulmuştur. Bu sebeple testlerde kullanılan komutların genel konuşma metinlerinde kullanım sıklıklarıda performansı etkilemektedir.

Çizelge 4.1 ve Şekil 4.2 üzerinde gösterilen veriler incelendiğinde 41677 farklı komut ses verisinin uçtan uca model ile doğru tanınma oranının ortalama %70,63 olduğu görülmektedir. Bu verilerden 2037'si çevre gürültüleri ve yanlış kayıtlar sebebiyle kaliteli bilgi içermediği tespit edilerek hatalı veri olarak işlenmiştir. Hatalı veriler başarı oranlarında yanlış tanınmış veri olarak kaydedilmiştir. Bu sebeple hatasız kayıtlar ile daha başarılı sonuçlar elde edilmesi beklenmektedir. Yapılan testler süresince her bir ses verisinin işlenip tahminde bulunulması için veri başına ortalama 1,17 sn süre harcanmıştır. Yapılan işlemler sanal makinalar üzerinde gerçekleştirildiği için hali hazırda sunulan GPU ve CPU özelliklerinin verimli kullanımı için çalışmalar yapılamamıştır. Bu noktada kurulabilecek sistemde donanımsal özellikler optimum seviyede çalıştırılarak daha hızlı sonuçlar elde edilebilir. Çizelge 4.1 incelendiğinde en düşük tanınma oranının %20,80 ile eight sayısına ait olduğu ve bu noktada eight sayısı ile en çok at kelimesinin karıştırıldığı görülmektedir. Eight sayısı ve at kelimesi ses yapısı itibari ile benzer olmaları ve kullanılan genel dil modelinde at kelimesinin çok sık kullanılması sebebiyle bu sapma oranının yüksek olduğu düşünülmektedir.



Şekil 4.2. Uçtan uca metot oransal sonuç grafiği

Çizelge 4.1.Uçtan uca metot kullanılarak elde edilen deneysel sonuçlar

Kelime	Başarı Oranı	Toplam Veri	Hatalı Veri	İşlem Süresi (sn)	Sapma Verisi
Zero	%47,79	2375	65	1,2457	on
One	%84,98	2370	67	0,8661	there
Two	%85,42	2356	106	1,0129	to
Three	%61,16	2356	111	1,1174	there
Four	%88,03	2372	79	1,1057	for
Five	%61,48	2357	95	1,5517	i
Six	%77,41	2372	147	1,2318	the
Seven	%76,69	2377	64	1,5306	so
Eight	%20,80	2298	302	1,5399	at
Nine	%63,71	2364	65	1,1302	i
Yes	%91,38	2371	76	1,1499	yet
No	%95,06	2168	61	0,9783	now
Go	%80,71	2017	134	1,0897	oh
Left	%50,86	2052	85	1,0393	let
Right	%84,83	2367	98	1,1813	it
Stop	%79,24	2380	84	1,1592	so
Off	%50,86	2357	262	1,0393	oh
On	%70,99	2368	136	1,1193	in
	Ort = %70,63	Top = 41677	Top = 2037	Ort = 1,17	

4.2. Test 2

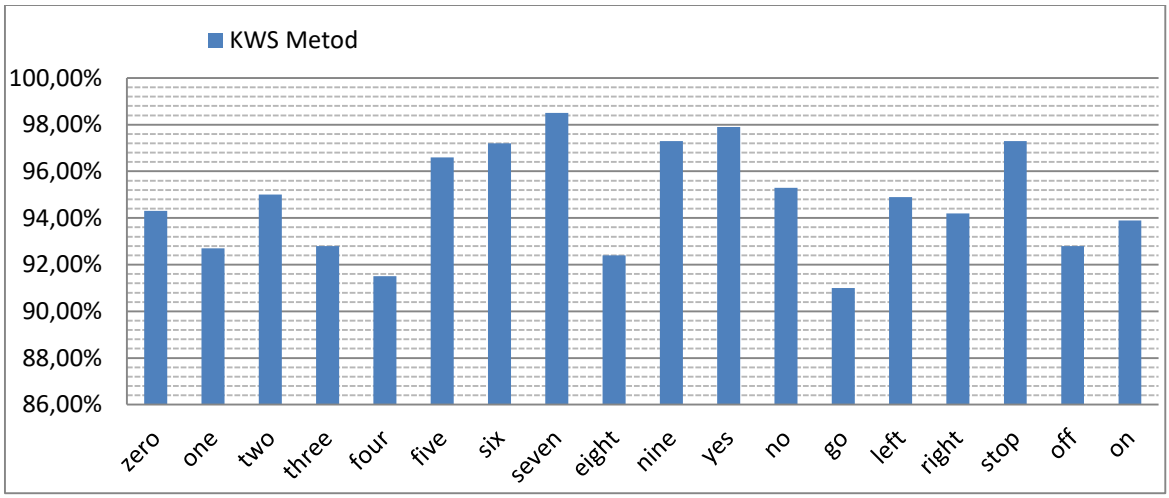
Bu test aşamasında Test 1 sürecinde kullanılan Speech Command Dataset kullanılmıştır. Önceki testten farklı olarak bu test aşamasında büyük bir dil modeli kullanmak yerine, komut veri setinde kullanılan veriler ile yeni bir dil modeli oluşturulmuştur ve bu yönteme KWS (keyword spotting) adı verilmiştir. KWS metodunda hali hazırda kullanılabilir durumda olan on altı ses verisinin tanınması için yine sadece komutlara özel ses verilerini içeren kısıtlı bir dil modeli ile çalışma yapılmıştır. Bu durumda çok çeşitli ses verileri kullanılmadığı için önceki teste oranla çok daha başarılı sonuçlar elde edilebilmiştir (Andrade vd., 2018).

Veri setinde bulunan on sekiz komut bilgisi arasından sırasıyla sıfır, bir, iki, üç, dört, beş, altı, yedi, sekiz, dokuz, evet, hayır, git, sol, sağ, kapat, aç ve dur komutları test edilmiştir. Testler sonucunda tanınmak istenen ses veriler ortalama %94,76 oranında doğru tanınmıştır. Şekil 4.3.'de gösterilen karıştırma matris incelendiğinde ses verilerinin tanınma işlemlerinin başarı oranları ve sapma değerleri detaylı olarak gözlemlenebilmektedir.

	zero	one	two	three	four	five	six	seven	eight	nine	yes	no	go	left	right	stop	off	on
zero	0,943	0	0,012	0,002	0,002	0	0,002	0,014	0	0	0	0	0	0	0	0	0	0
one	0	0,927	0	0	0	0	0	0	0	0,013	0	0	0	0	0	0	0	0,013
two	0,005	0	0,95	0,002	0	0	0	0	0,002	0	0	0,005	0,017	0	0	0	0	0
three	0	0	0,05	0,928	0	0	0	0	0,012	0	0	0	0	0	0	0	0	0
four	0	0,003	0	0	0,915	0,003	0	0	0	0	0	0	0,007	0	0	0	0	0,003
five	0	0,002	0	0,002	0,002	0,966	0	0	0	0,007	0	0,002	0	0	0	0	0,002	0,004
six	0,003	0	0	0,003	0	0	0,972	0,01	0	0	0	0	0	0	0,003	0,003	0	0
seven	0,002	0	0	0	0	0	0,007	0,985	0	0	0	0	0	0	0	0	0	0
eight	0	0	0,012	0,002	0,002	0	0	0	0,924	0,005	0,002	0	0	0	0,002	0	0	0,007
nine	0	0	0	0	0	0,007	0	0	0	0,973	0	0,005	0	0	0	0	0	0
yes	0	0	0	0	0	0	0,002	0	0,002	0	0,979	0	0	0,002	0	0	0	0
no	0	0	0	0	0	0	0	0	0	0,01	0	0,953	0	0,002	0	0	0	0
go	0	0,002	0,005	0	0	0	0	0	0	0,002	0	0,032	0,91	0	0	0	0	0
left	0	0	0	0,002	0	0	0	0	0	0	0,005	0,002	0	0,949	0	0	0	0
right	0	0	0	0,005	0,003	0,018	0	0	0	0,015	0	0	0	0,003	0,942	0	0	0
stop	0	0	0	0	0,02	0	0,02	0,007	0	0	0	0	0	0	0	0,973	0	0
off	0	0	0	0	0	0	0	0	0,002	0	0	0,007	0,005	0,002	0	0	0,928	0,015
on	0	0,0013	0	0	0,003	0,008	0	0	0	0	0	0	0	0	0	0	0,01	0,939

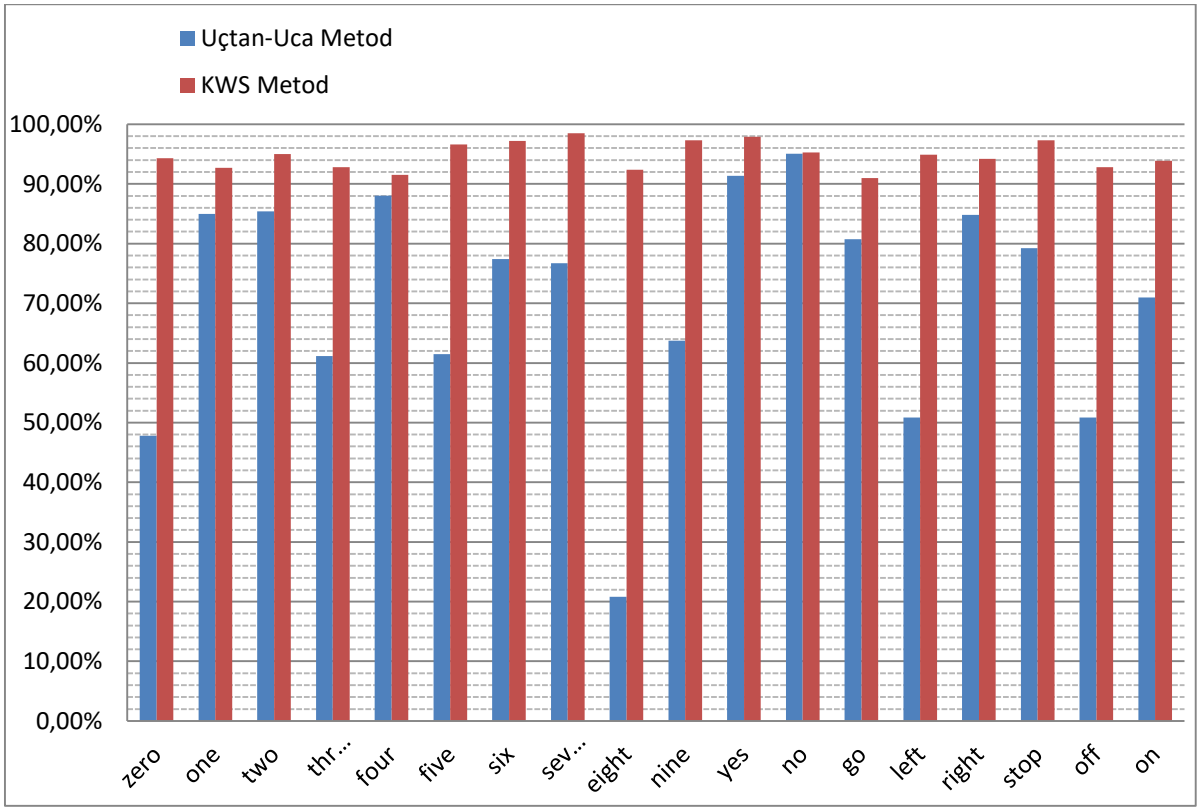
Şekil 4.3. Anahtar kelime arama metodu karıştırma matrisi

Şekil 4.4 üzerinde gösterilen veriler incelendiğinde ses verilerinin tanınma sonuçlarının %90 oranının altına inmediği görülmektedir. Bu sebeple sınırlı sayıdaki veri ile hazırlanan dil modelinin Test 1 sonuçlarına kıyasla başarı oranlarını yükselttiği görülebilmektedir. Şekil 4.4'den %98,5 oranı ile yedi sayısının en başarılı sonucu ve %91 oranı ile git kelimesinin en başarısız sonucu verdiği görülebilmektedir.



Şekil 4.4. Anahtar kelime arama metodu ile elde edilen başarımlar

Hazırlanan test modelinde kullanılan dil modelinin kısıtlı olması sebebiyle uçtan uca ses tanıma modelinde amaçlanan içerik bağımsız ses tanıma işlemlerini gerçekleştiremeyeceği göz önüne alınmalıdır. Bu noktada kısıtlı veri ile hazırlanan modelin başarılı sonuç vermesi değerlendirildiğinde Test 1'de özellikle sıfırdan dokuza sayıların tanınması oranının arttırılması için ses tanıma adımlarına özel seçenekler eklenebilir. Örneğin, Test 1 modelinde çalışırken tuşla gibi özel bir komut algılandıktan sonra tanınmak istenen ses verilerinin sayılardan oluşacağı ön görülerek kısıtlı bir dil modeline geçiş yapılarak başarı oranları arttırılabilir. Bu sayede içerik bağımsız olarak çalışabilecek uçtan uca bir model ile daha başarılı sonuçlar elde edilebilir ve Şekil 4.5 üzerinde karşılaştırılan sonuçların arasındaki farklar azaltılabilir.



Şekil 4.5. Uçtan uca metodu ve KWS metodu ile elde edilen başarımlar

4.3. Test 3

Bu test aşamasında Türkçe bir veri seti olan TURTEL kullanılmıştır. TURTEL veri seti TÜBİTAK-UEKAE tarafından hazırlanmış ve lisanslanmıştır. Bu veri seti 373 kelime ve 15 cümleden meydana gelmektedir.

Veri setine ait eğitim verileri için 25'i kadın 40'ı erkek 65 konuşmacı, test verileri için 11'i kadın 17'si erkek 28 konuşmacıdan destek alınmıştır. Konuşmacıların yetiştikleri iller en çok yüzdeye sahip 9 il oranlarına göre sırasıyla %22.5 İstanbul, %21 Kocaeli, %7.5 Ankara, %5 İzmir, %2.5 Kayseri, %2.5 Samsun, %2.5 Konya, %2.5 Adana ve %2.5 Eskişehir olarak belirtilmiştir. Eğitim verilerine katılan konuşmacaların yaş ortalama 30.2 ve test verilerine katılan konuşmacaların yaş ortalaması 29.1 olarak açıklanmıştır. Tez çalışmaları için TURTEL veri setinden 38 kelime seçilmiştir. Seçilen kelimeler kullanılarak oluşturulan sistemin eğitimi için her kelime için ortalama 36 ses verisinden toplamda 1839 ses verisi kullanılmıştır. Eğitim setinin testleri için 520 ses verisi ve sistemin testleri içinse 240 ses verisi kullanılmıştır. Toplamda 2599 ses verisi kullanılmıştır. Rakamlar oransal olarak incelendiğinde toplam ses verisinin %70'i eğitim, %20'si eğitim testi ve %10'u sistem testi için kullanılmıştır.

Çizelge 4.2.Türkçe veri seti elemanları

Açık	Altı	Altmış	Beş	Bin	Bir
Doğru	Doksan	Dokuz	Dört	Elli	Evet
Hata	Hayır	İki	İyi	Katrilyar	Katrilyon
Kırk	Milyar	Milyon	Oku	On	Otuz
Oynatma	Sekiz	Seksen	Ses	Sıfır	Tamam
Trilyar	Trilyon	Üç	Yapma	Yedi	Yetmiş
Yirmi	Yüz				

Yukarıdaki çizelgedeki veriler kullanılarak hazırlanan uçtan uca sisteme ait sonuçlar incelendiğinde her bir kelime veya sayı için 6 ses verisi teste alınmıştır ve maksimum 1 ses verisi doğru olarak tanınabilmiştir. Oransal olarak incelendiğinde kelimelerin birçoğu hatalı olarak tespit edilmişken altmış, doğru, iki, katrilyon, otuz ve sekiz kelime veya sayıları %16,67 oranda doğru tahmin edilmiştir. Elde edilen sonuçlar yetersiz veri nedeniyle sistemin iyi eğitilemediğini göstermektedir. Eğitim için kullanılan veri sayısının arttırılmasıyla sistemin doğru sınıflama başarımı yükseltilebilir.

5. BULGULAR VE TARTIŞMA

Hazırlanan model içerik ve kullanıcıdan bağımsız olarak mevcut sistemlere oranla daha başarılı sonuçlar vermektedir. Test sonuçları incelendiğinde mevcut model, bazı kelime ve sayılarda düşük sonuçlar elde etmiş olsa da ortalama %70,78 oranda girdi verilerini doğru tahmin etmeyi başarmıştır. Ek olarak literatür kaynakları incelendiğinde incelenen modelin gürültülü ortamlarda gözlemlenen hata oranları (%30), gürültüsüz ortamlarda gözlemlenen hata oranları (%8) ile karşılaştırıldığında yaklaşık 4 kat daha başarısız sonuçlar elde ettiği görülmüştür. Bu noktada gürültü değerlerinin etkisini düşürmek için ses girdisinin temin edildiği cihazın gürültü bastırma özelliğinin olması sisteme ekstra yük getirmeden hata oranlarını düşürecektir. Gürültünün ses tanıma adımlarına olan olumsuz etkisi; donanımsal olarak, gürültüye dayanıklı özniteklik kullanılarak ve anahtar kelime arama yöntemi kullanılarak en aza indirgenebilir.

Hazırlanan sistemin ortalama performans değerleri, içerik bağımsız olarak hazırlanan dil modelleri ile çalışması göz önüne alındığında mevcut kullanılan modellere oranla düşük başarı oranları elde etmesine rağmen rekabet edebilir seviyede olduğu kabul edilebilir. Mevcut sistemler kullanım alanlarına göre hazırlanan dil modelleri ve kullanıcıya özgün veriler ile çok yüksek başarı sonuçları elde etmektedir. Uçtan uca ses tanıma sisteminin hazırlanmasında amacın içerik ve kullanıcıdan bağımsız çalışması olduğu düşünüldüğünde, bu alanda yapılacak ek çalışmalar endüstride kullanıma daha uygun sonuçlar ortaya çıkarabilecektir.

Teknolojik gelişmeler ışığında önerilen yeni sistemlerin performans açısından mevcut sistemler ile hızlıca rekabet edebilir noktaya geldiği görülebilir. Ancak teknolojik gelişmelerin kullanılması için uygun araştırma ve geliştirme ortamlarının hazırlanması büyük maliyetlere neden olmaktadır.

Endüstriyel ortamlarda kullanılması hedeflenen bir sistemin maliyet performansınında da yeterli seviyede olması gerekmektedir. Bu gelişmeler ışığında çözüm olarak bulut sistemleri kullanılabir bir seçenek olarak ön plana çıkmaktadır. Sektörün öncü firmaları bu noktada kullanıcılara uygun web servisleri sunmaktadırlar.

6. SONUÇLAR VE ÖNERİLER

Endüstriyel ortamda yapılacak çalışmalarda sistemi yöneten operatörün cinsiyeti ve aksanı başta olmak üzere ses verisini değiştirebilecek parametrelerden bağımsız olarak çalışabilecek bir sistem tasarımı geliştirilmeye çalışılmıştır. Hazırlanan sistemin uçtan uca olarak çalışacak olması bütün işlemlerin otomatikleştirilmesi için önem arz etmektedir. Bu süreçlerin otomatikleştirilmesi ile beraber sistem tasarım sırasında harcanan zaman azaltılmış olacaktır.

Ses tanıma sistemlerinin içerik bağımsız çalışması önemlidir. Bu sebeple hazırlanacak dil modelinin o dile özgü her türlü bilgiyi içerecek şekilde oluşturumalı ve modelleme için yeterli sayıda ses verisi içermelidir. Ses tanımada tek bir dil modeli ile içerik bağımsız ses tanıma yapılırken yüksek başarımlar elde edilemektedir. Bu nedenle, genel olarak uygulamaya yönelik hazırlanmış veritabanları kullanılmaktadır. İçerik bağımsızı veriseti üzerinde yapılan ses tanıma çalışmalarında, kelimeler arasında karıştırma oranı yüksek çıkmaktadır. Test 1 sonuçları kelimeler ve sayılar arasında karıştırma oranlarını göstermiştir. Seçilen kelime ve sayı için hazırlanan özel dil modelleri ile doğru tanıma başarımlarının arttığı Test 2'de yapılan çalışma ile gösterilmiştir. Tez çalışmasında önerilen sistem, istenen konuya özel dil modelini kullanarak anahtar kelimeleri tanıyacak hale getirilebilir.

Yapılan çalışmalar sonucunda Test 3 verileri incelendiğinde, beklenildiği gibi yapay zekâ uygulamalarının küçük veriler ile yapılan çalışmalarda çok başarısız sonuçlar verdiği görülmüştür. Diğer testlerde kullanılan veriden çok daha küçük veri seti ile yapılan çalışmada sistemin otomatik öğrenme başarımının çok düşük olduğu tespit edilmiştir. Veri setinin yetersiz olduğu durumlarda geleneksel makine öğrenme yöntemleri daha iyi başarımlar vermektedir. Sesli tanıma işlemleri için kaynaklar incelendiğinde Türkçe için maalesef büyük ve kullanışlı bir veri setine erişim sağlanamamıştır. Düzenli bir şekilde hazırlanmış daha fazla veri ile yapılacak çalışmalarda daha doğru sonuçlar elde edilebileceği düşünülmektedir.

İşlenmek istenen verilerin gürültüden arınmış olması her modelde başarı oranını yükseltecek olsa da, hazırlanan modelde geçmişe dönük bilgilerin aktif olarak kullanılması sebebiyle gürültü değerlerinin de tekrar tekrar işlenmesi tahmin süreçlerini daha da olumsuz etkilemektedir.

Sonuç olarak, uçta-uca ses tanıma yöntemleri içerik bağımlı ses tanımada sistemlerinin geliştirilmesinde kolaylıklar sağlayacaktır. Uçtan uca ses tanıma sistemlerinin başarı oranlarının hibrit modeller ile rekabet edebilir noktaya getirebilmek için çalışmalara devam edilmektedir.

KAYNAKLAR DİZİNİ

Andrade, D.C., Leo, S., Viana, M.L.D.S., Bernkopf, C., 2018, https://arxiv.org/pdf/1808.08929.pdf, erişim tarihi: 08.02.2019.

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S, Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P, Wattenberg, M, Wicke, M., Yu, Y., Zheng, X., 2015, Tensorflow: large-scale machine learning on heterogeneous distributed systems, https://arxiv.org/pdf/1603.04467.pdf, erişim tarihi : 10.08.2019.

Anonim, 2008, What is a Tensor?, https://www.doitpoms.ac.uk/tlplib/tensors/what_is_tensor.php erişim tarihi: 08.02.2019.

Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., 2016, Deep speech 2 : end-to-end speech recognition in English and Mandarin, Proceedings of Machine Learning Research, 48, p.173-182.

Audhkasi, K., Rosenberg, A., Sethy, A., Ramabhadran, B., Kingsbury, B., 2017, End-to-end asr-free keyword search from speech, International Conference on Acoustics, Speech and Signal Processing, p.4840-4844.

Büyük, O., 2018, Mobil araçlarda Türkçe konuşma tanıma için yeni bir veri tabanı ve bu veri tabanı ile elde edilen ilk konuşma tanıma sonuçları, Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 24(2), p.180-184.

Auvolat, A., Mesnard T., Research Project Report, École Normale Supérieure de Cachan, p.6. (yayımlanmamış).

Carneiro, T., Nóbrega, R.V.M.D, Nepomuceno, T., Bian, G.B., Albuquerque, V.H.C.D., Filho, P.P.R., 2018, Performance analysis of Google Colaboratory as a tool for accelerating deep learning applications, IEEE Acces, 6, p.61677-61685.

Chan, W., Jaitly, N., Le, Q.V., Vinyals, O., 2016, Listen, attend and spell: a neural network for large vocabulary conversational speech recognition, International Conference on Acoustics, Speech and Signal Processing, p.4960-4964.

Durán, J.A.R., Hernández, C.T., Castro, J.A., 2012, Addressıng some stress tensor transformatıons ın the maple software envıronment, https://www.arpapress.com/Volumes/Vol13Issue1/IJRRAS_13_1_01.pdf, erişim tarihi : 10.08.2019.

KAYNAKLAR DİZİNİ (devam)

Edizkan, R., Barkana A., 2000, Comparison of subspace methods and hmm from different view of points, 8th Signal Processing and Communication Applications Conference.

Feng, J.Q., 2012, Music in terms of science, https://arxiv.org/pdf/1209.3767.pdf, erişim tarihi : 04.02.2019.

Hain, T., 2001, Hidden model sequence models for automatic speech recognition, Doktora Tezi, University of Cambridge, 136 s. (yayımlanmamış).

Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., Ng, A.Y., 2014, Deep speech: scaling up end-to-end speech recognition, https://arxiv.org/pdf/1412.5567.pdf, erişim tarihi : 10.08.2019.

Harris, M., Sengupta, S., Owens, J.D., 2007, Parallel prefix sum (scan) with CUDA, https://developer.nvidia.com/gpugems/GPUGems3/gpugems3_ch39.html, erişim tarihi: 10.02.2019.

Hermansky, H., 1990, Perceptual Linear Predictive (PLP) analysis of speech, in J. Acoust. Soc. Am., p. 1738-1752.

Huang, Y., Hughes, T., Shabestary, T.Z., Applebaum, T., 2018, Supervised noise reduction for multichannel keyword spotting, International Conference on Acoustics, Speech, and Signal Processing 2018, p.55474-5478.

Kang E., 2017, Hidden Markov Model, https://medium.com/@kangeugine/hidden-markov-model-7681c22f5b9, erişim tarihi : 04.02.2019.

Kincaid, J., A brief history of ASR: automatic speech recognition, https://medium.com/descript/a-brief-history-of-asr-automatic-speech-recognition-b8f338d4c0e5, erişim tarihi: 08.02.2019

Maas, A.L., Le, Q.V., O'Neill, T.M., Vinyals, O., Nguyen, P., Ng, A.Y., 2012, Recurrent neural networks for noise reduction in robust asr, Interspeech 2012 13th Annual Conference of the International Speech Communication Association, p. 22-25.

Mattfeld, 2014, Implementing spectral methods for hidden Markov models with real-valued emissions, https://arxiv.org/pdf/1404.7472.pdf, erişim tarihi : 04.02.2019.

Miao, Y., Metze, F., 2017, End-to-end architectures for speech recognition, https://link.springer.com/chapter/10.1007/978-3-319-64680-0_13, erişim tarihi: 01.03.2019.

KAYNAKLAR DİZİNİ (devam)

Miao, Y., Gowayyed, M., Metze, F., 2015, Eesen: end-to-end speech recognıtıon usıng deep rnn models andwfst-based decodıng, IEEE Workshop on Automatic Speech Recognition and Understanding, p.167-174.

Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., Khudanpur, S., 2010, Interspeech 2010 11th Annual Conference of the International Speech Communication Association, p. 1045-1048.

Moskvitch, K., The machines that learned to listen, http://www.bbc.com/future/story/20170214-the-machines-that-learned-to-listen, erişim tarihi: 08.02.2019

Olah, C., 2015, Understanding LSTM Networks, http://colah.github.io/posts/2015-08-Understanding-LSTMs/, erişim tarihi: 10.08.2019

Öcal, K., 2005, Otomatik konuşma tanıma algoritmalarının uygulamaları, Yüksek Lisans Tezi, Ankara Üniversitesi, 81 s. (yayımlanmamış).

Pardade, H.F., 2015, On noise robust feature for speech recognition based on power function family, 2015 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), p. 386-390.

Prasanna, S., Harris, M., 2018, RAPIDS accelerates data science end-to-end, https://devblogs.nvidia.com/gpu-accelerated-analytics-rapids/, erişim tarihi: 08.02.2019.

Rallabandi, P.K., Patidar, K.C., 2015, A Hybrid System of Hidden Markov Models and Recurrent Neural Networks for Learning Deterministic Finite State Automata, World Academy of Science, Engineering and Technolog International Journal of Computer and Information Engineering, 9, 11.

Sainath, T.N., Parada, C., 2015, Convolutional neural networks for small-footprint keyword spotting, 16th Annual Conference of the International Speech Communication Association, p. 1478-1482.

Wilson, A.C., Roelofs, R., Stern, M., Srebro, N., Recht, B., 2018, The marginal value of adaptive gradient methods in machine learning, ttps://arxiv.org/pdf/1705.08292.pdf, erişim tarihi: 10.08.2019