BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ
DOKTORA PROGRAMI

TÜRKÇE OTOMATİK KONUŞMA TANIMA VE İŞARET DİLİNE ÇEVİRME

HAZIRLAYAN

BURAK TOMBALOĞLU

DOKTORA TEZİ

ANKARA – 2021

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI
EKETRİK ELEKTRONİK MÜHENDİSLİĞİ
DOKTORA PROGRAMI

TÜRKÇE OTOMATİK KONUŞMA TANIMA VE İŞARET DİLİNE ÇEVİRME

HAZIRLAYAN

BURAK TOMBALOĞLU

DOKTORA TEZİ

TEZ DANIŞMANI

PROF.DR. HAMİT ERDEM

ANKARA – 2021

BAŞKENT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Elektrik Elektronik Mühendisliği Anabilim Dalı Elektrik Elektronik Mühendisliği Doktora Programı çerçevesinde Burak TOMBALOĞLU tarafından hazırlanan bu çalışma, aşağıdaki jüri tarafından Doktora Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi: 11/01/ 2021

Tez Adı: Türkçe Otomatik Konuşma Tanıma ve İşaret Diline Çevirme

Tez Jüri Üyeleri (Unvanı, Adı - Soyadı, Kurumu)	**İmza**
Prof. Dr. Emin AKATA (Başkan), Başkent Üniversitesi	……..……..
Prof. Dr. Hamit ERDEM (Danışman), Başkent Üniversitesi	……..
Prof. Dr. Hasan OĞUL, Çankaya Üniversitesi	……..
Prof. Dr. Hasan Şakir BİLGE , Gazi Üniversitesi	……..
Dr. Öğr. Üyesi Emre SÜMER, Başkent Üniversitesi	……..

ONAY

Prof. Dr. Ömer Faruk ELALDI

Fen Bilimleri Enstitüsü Müdürü

Tarih : … / … / ….…….

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
DOKTORA TEZ ÇALIŞMASI ORİJİNALLİK RAPORU

Tarih: 29 / 01 / 2021

Öğrencinin Adı, Soyadı : Burak Tombaloğlu
Öğrencinin Numarası : 20620010
Anabilim Dalı : Elektrik Elektronik Mühendisliği
Programı : Elektrik Elektronik Mühendisliği Doktora
Danışmanın Unvanı/Adı, Soyadı : Hamit Erdem
Tez Başlığı : Türkçe Otomatik Konuşma Tanıma ve İşaret Diline Çevirme

Uygulanan filtrelemeler:

1. Kaynakça hariç

2. Alıntılar hariç

3. Beş (5) kelimeden daha az örtüşme içeren metin kısımları hariç

"Başkent Üniversitesi Enstitüleri Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Usul ve Esaslarını" inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Öğrenci İmzası:…….

ONAY

Tarih: -- / -- / 2021

Öğrenci Danışmanı Unvan, Adı, Soyadı

TEŞEKKÜR

Tez çalışmalarım boyunca beni yönlendirdiği, bilgi ve birikimlerini bana aktardığı için danışmanım Sayın Prof. Dr. Hamit ERDEM'e teşekkürlerimi sunarım.

Bana her konuda destek olan ve her türlü fedakârlıkta bulunan annem, babam ve eşime teşekkürlerimi borç bilirim.

ÖZET

Burak TOMBALOĞLU

TÜRKÇE OTOMATİK KONUŞMA TANIMA VE İŞARET DİLİNE ÇEVİRME

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik Elektronik Mühendisliği Anabilim Dalı

2021

Bu tezde, işitme engelli insanlar ile işitme engelli olmayan insanların aktif iletişimine yardımcı olabilecek bir sistem üzerinde çalışılmaktadır. Sistem genel olarak, iki adımda çalışmaktadır. İlk olarak konuşma metne çevrilir. Daha sonra metnin karşılık geldiği işaret dili videosu gösterilir. Metne çevrim aşamasında, Türk Dili incelenmiş olup fonem tabanlı bir dildir. Endüstri, güvenlik, iletişim ve robotik sistemlerin gelişmesiyle Otomatik Konuşma tanıma (ASR) tabanlı uygulamaların kullanımı gün geçtikçe artmaktadır. Teknolojik gelişmelerle beraber, birçok dilde ASR uygulamaları sıkça yaygınlaşırken, Türkçe, Fince ve Macarca gibi sondan eklemeli dil gruplarında bu uygulamalar çok fazla değildir. Türkçe hece yapısı olarak sondan eklemeli bir morfolojiye sahiptir. Bu yapısı sözcük dağarcığında büyük bir artışa neden olmaktadır. Sistem ileriye dönük olarak, veri tabanı haricindeki kelimeleri de yaklaşık olarak tespit edebilsin diye fonem ve alt kelime tabanlı bir tanıma sistemi tasarlanmıştır. Klasik yöntemlerin yanı sıra, akıllı ve öğrenebilen yöntemler de bu alanda sıkça kullanılmaktadır Bu çalışmada, Türk dilinde ASR problemi çözümüne yönelik güncel Derin Öğrenme kapsamlı uygulamalar geliştirilmiştir. Çalışmamızın, Konuşma Tanıma adımında klasik yöntemlerle beraber, Derin İnanç Ağları (DBN), Uzun-Kısa Vadeli Hafıza Ağları (LSTM) ve Geçitli Tekrarlayan Birimler (GRU) Derin Öğrenme teknikleri uygulanıp performansları karşılaştırılmıştır. En başarılı yöntemin dil modellemenin de Derin öğrenme ile yapıldığı GRU metodu ile olduğu görülmüştür. Yöntemlerin performansı, standart ölçütlere göre karşılaştırılmıştır. Yapılan çalışma, ASR uygulamaları ile ilgili bir taraftan konuyu detaylı araştırırken, yöntemin uygulama biçimi hakkında da detaylı bilgi vermiştir. Konuşma tanımda adımında Türkçe için yapılan iyileştirmeden sonra, konuşmanın yazıya dönüştürülmesi ile elde ettiğimiz kelimenin İşaret Dilinde hangi işarete karşılık geldiği bulunarak ve bu işaret videolarıyla Türk İşaret Diline çevrimi de gerçekleştirilmiştir. Türkçe için literatürde görünmeyen, Türkçe Konuşmayı Türk İşaret Diline çeviren bu çalışma, Türkçe konuşma tanıma sistemlerinde performans artırma ve işitme engelli insanların hayatını kolaylaştırmak için öncü bir çalışma olduğu düşünülmüştür.

ANAHTAR KELİMELER: Türkçe, Konuşma Tanıma, Derin Öğrenme, Türk İşaret Dili.

ABSTRACT

Burak TOMBALOĞLU

AUTOMATIC SPEECH RECOGNITION AND SIGN LANGUAGE TRANSLATION FOR TURKISH

Başkent University Institute of Science and Engineering

Department of Electric Electronic Engineering

2021

In this thesis, we are working on a system that can help people with hearing impairments to communicate actively with people who are not hearing impaired. The system generally works in two steps. First, the speech is translated into text. The sign language animation or video to which the text corresponds is then shown. During the translation into text phase, Turkish Language has been analyzed and it is a phoneme-based language. With the development of industry, security, communication and robotic systems, the use of Automatic Speech recognition (ASR) based applications is increasing day by day. Along with technological developments, while ASR applications are becoming common in many languages, these applications are not much in agglutinative language groups such as Turkish, Finnish and Hungarian. Turkish has an additive morphology as a syllable structure. This structure causes a great increase in vocabulary. A phoneme and subword based recognition system has been designed for the future so that the system can detect the words other than the database approximately. In addition to classical methods, intelligent and learning methods are frequently used in this field. In this study, current Deep Learning comprehensive applications have been developed for solving the ASR problem in Turkish language. In the Speech Recognition step of our study, Deep Belief Networks (DBN), Long-Short-Term Memory Networks (LSTM) and Gated Repetitive Units (GRU) Deep Learning techniques were applied and their performances were compared. It has been seen that the most successful method is with the GRU method, where language modeling is also done with deep learning. The performance of the methods was compared against standard criteria. The study, while investigating the subject in detail about the ASR applications, also gave detailed information about the application method of the method. After the improvement made for Turkish in the step of speech definition, the word we obtained by converting the speech into writing was found to correspond to the sign in Sign Language and translated into Turkish Sign Language with these sign videos. This study, which does not appear in the literature for Turkish, converts Turkish Speech to Turkish Sign Language, is thought to be a pioneering work to increase performance in Turkish speech recognition systems and facilitate the lives of hearing impaired people.

KEYWORDS: Turkish Language, Automatic speech recognition, Deep Learning, Turkish Sign Language.

İÇİNDEKİLER

ŞEKİLLER LİSTESİ

Sayfa

TABLOLAR LİSTESİ

Sayfa

SİMGELER VE KISALTMALAR LİSTESİ

ASR	Otomatik Konuşma Tanıma
DBN	Derin İnanç ağları
DNN	Derin Sinir Ağları
SVM	Destek Vektör Makinesi
GMM	Gauss Karışım Modelleri
GRU	Geçitli Tekrarlayan Birimler
HMM	Saklı Markov Modelleri
LDC	Linguistic Data Consortium
LM	Dil Modelleme
LPCC	Doğrusal Öngörü Kepstral Katsayıları
LSTM	Uzun-Kısa Süreli Bellek
MFCC	Mel Frekans Kepstral Katsayısı
PDF	Olasılık Yoğunluk Fonksiyonu
PER	Fonem Hata Oranı
RBM	Kısıtlı Boltzman Makinesi
RNN	Tekrarlayan Sinir Ağları
SVM	Destek Vektör Makinaları
TDK	Türk Dil Kurumu
TDNN	Zaman Gecikmeli Sinir Ağı
TİD	Türk İşaret Dili
WAR	Kelime Doğruluk Oranı
WER	Kelime Hata Oranı
YSA	Yapay Sinir Ağı

1. GİRİŞ

1.1. Konunun Tanımı

Bu tezde, işitme engelli insanlar ile işitme engelli olmayan insanların aktif iletişimine yardımcı olabilecek bir sistem üzerinde çalışılmaktadır. Sistem genel olarak, iki adımda çalışmaktadır. İlk olarak konuşma metne çevrilir. Daha sonra metnin karşılık geldiği işaret dili videosu gösterilir.

Konuşma, en yaygın iletişim yöntemidir. Teknolojinin ilerlemesi, makinelerin insan konuşmasını işlemesine olanak vermiş olup, insan ile makine arasındaki iletişimin kullanımını artırmıştır. Sonuç olarak, teknolojinin herkes tarafından kullanımı yaygınlaşacak ve engellilerin ihtiyaçlarını karşılamaları kolaylaşacaktır.

Çalışmamızın ilk adımında konuşmanın metne çevrilmesi için Otomatik Konuşma Tanıma (ASR) gerçekleştirilmektedir. Otomatik konuşma tanıma (ASR) temel olarak, insan sesinin ve konuşmalarının bilgisayar programları veya elektronik aygıtlarla algılanıp, tanınmasıdır. Bu sistem çeşitli ses algılama ve tanıma metotlarını kullanarak, konuşmayı yazıya çevirir. Sistem, konuşmanın özniteliklerini çıkarır ve ses birimleri(fonem) ve kelime bileşenlerini sınıflandırır. Bilgisayar ve işlemcilerin gelişimi ile birlikte ASR uygulamalarının doğruluk oranı ve kullanım alanları artmıştır. Bu uygulamalar, Hava trafik kontrolü, bilet rezervasyonları, güvenlik, biyometrik tanımlama, oyunlar, otomobillerdeki cihazların kontrolü, ev otomasyonu ve robotik uygulamalar yaygın kullanım alanları arasındadır. Ayrıca bu sistemler engelli insanların yaşam kalitelerini yükseltmek için de kullanılmaktadır.

En gelişmiş ASR sistemleri, popüler ve yaygın olarak konuşulan diller için geliştirilmiştir, ör. Çince, İspanyolca ve İngilizce. Bu diller, akustik ve dil modelleri oluşturmak için kapsamlı transkripsiyonel konuşma ve metin verilerine sahiptir. Telaffuz sözlükleri, dilbilimciler tarafından incelenen telaffuz kuralları ve akustik birimler yardımıyla oluşturulur.

Türkçe, fonem tabanlı bir dildir ve Fince veya Macarca gibi sondan eklemeli bir dil grubuna dahildir. Kelime köküne eklenen son ekler vardır. Bu nedenle farklı kelime dağarcığının sayısı arttıkça konuşma tanıyıcıların performansı düşer. Ayrıca kelime dizilimlerindeki serbestlik dilin modellenmesindeki karmaşıklığı artırmaktadır. Kelimeler

yerine alt kelimelerden (morfemler) oluşan bir kelime dağarcığı kullanmak da yaygın bir çözümdür.

Son yıllarda bilgisayar teknolojisinin gelişmesi ve hesaplama için GPU (Grafik İşleme Birimi) kullanılması ile Derin Öğrenme metodları, ASR uygulamalarında istatistiksel yöntemlerin yerini almış ve önemli performans artışları sağlamıştır. Çalışmamızda da, Derin Öğrenme Metodlarının güncel ve en başarılı olanları, Derin İnanç Ağları (DBN), Uzun-Kısa Süreli Bellek (LSTM) ve Geçitli Tekrarlayan Birimler (GRU) Türkçe Konuşma Tanıma problemine uygulanmış ve sonuçları incelenmiştir.

Çalışmamızda, konuşmanın yazılı metne çevrim adımından sonra işaret diline çevirme adımı gerçekleştirilmiştir. Konuşmayı işaret diline çeviren çalışmalar, İngiliz İşaret Dili, Amerikan İşaret Dili ve İspanyol İşaret Dili için yapılmıştır. Türk İşaret Dili (TİD) için yapılan çalışmalar oldukça sınırlıdır. Türkçe Konuşmanın (Konuşma İşaretinin) İşaret Diline çevrildiği çalışmaya literatür taramasında rastlanmamıştır.

1.2. Konuya İlişkin Yapılan Önceki Çalışmalar

Türkçe, Estonca, Fince, Tayca, Macarca, Slovence ve Çekçe gibi sondan eklemeli diller grubundadır. Alt kelime Tabanlı ASR, sondan eklemeli diller için yaygın olarak uygulanmaktadır [1]. Bir alt kelime n-gram modeli, görünmeyen kelime formlarına olasılıklar atayabilir [2]. Aşırı büyük kelime haznesi, girdi ve çıktı katmanlarının boyutunun artmasına neden olur. Bu nedenle pratik değildir. Sınıf temelli modeller, hiyerarşik softmax gibi katmanları, yöntemleri veya kısa listeleri azaltabilse de, alt kelime modelleri, boyutluluğu düşürmek için etkili ve doğal bir yol sağlar [3]. Macarca Fince ve Türkçe gibi sondan eklemeli dillerin ASR uygulamalarında, alt kelime tabanlı dil modelleri kelime tabanlı dil modellerinden daha iyi performans göstermektedir. Bu nedenle, önerilen ASR sisteminin eğitilmesi için alt kelime tabanlı dil modeli tercih edilmektedir[1-6].

Mevcut konuşma tanıma sistemleri, öznitelik çıkarma, akustik model, Dil Modelleme (LM), kelime sözlüğü ve sınıflandırma bölümlerini içerir. Cümlelerin kelimelerini tanımak için kelimeleri oluşturan ses bileşenlerinin akustik olarak modellenmesi gerekir. Akustik analiz Gaussian Karışım Modelleri (GMM) ile yapılır ve son olasılıklar oluşturulur. Akustik Modeller, Saklı Markov Modelleri (HMM) kullanılarak oluşturulmakta ve son

yıllarda bilgisayarların ve gelişmiş mikroişlemcilerin gelişmesiyle Derin Öğrenme yöntemleriyle işlenmektedir. Bu şekilde kelimeler veya cümleler tahmin edilebilir.

Son yıllarda bilgisayar teknolojisinin gelişmesi ve hesaplama için GPU (Grafik İşleme Birimi) kullanılması ile Derin Öğrenme, ASR uygulamalarında GMM'nin yerini almış ve önemli performans artışları sağlamıştır. Bu kapsamdaki sınıflandırıcılar GMM-HMM, Deep Neural Networks (DNN) -HMM olarak gruplanabilir. DNN ve GMM, her bir fonemik izi temsil eden HMM için durum bilgisi sağlar. DNN'ler, HMM'ye ses birimleri arasındaki farkları daha iyi temsil eden daha fazla durum bilgisi sağlar. GMM'nin DNN ile değiştirilmesi, birçok araştırmacı tarafından HMM durumlarının olasılıklarını tahmin etmek için önerilmiştir [7-12].

Akıllı haberleşme cihazlarında "Apple-Siri" ve "Google Voice Transcription" gibi çeşitli sesli yardımcılar kullanılıyor. Bu uygulamalar, her an sesinizin akustik modelini konuşma sesiniz üzerinden olasılık dağılımına dönüştürmek için bir Derin Sinir Ağlarını (DNN) kullanır. Bu uygulamaların ASR uygulamaları ilgili şirketlerin ağlarına ait bulutta bulunmakdır. Bulut sunucuları, ASR tarafından kullanılan akustik modellere büyük depolama tesisleri ve güncellemeler sağlarlar.[13,14]

GMM'in yerine Destek Vektör Makinalarının da kullanımı tez çalışmaları sırasında araştırılmış ve denenmiş olup sonuçları tartışılmıştır. Geliştirilen sistemlere örnek olarak, [15]'te ses birim (fonem) tabanlı bir sistem geliştirilmiş olup, alt uzay sınıflandırma yöntemi kullanılmıştır. Fonem tabanlı sınıflandırma yapan [16]'daki çalışmada Destek Vektör Makinaları (SVM) sınıflandırıcıların kullanıldığı bir kelime tanıma uygulaması geliştirilmiştir. MFKK öznitelikleri ve çoklu sınıf (multiclass) SVM sınıflandırıcıları kullanılarak, yazarların önceki çalışmasında, Türkçeye yönelik bir çalışma yapılmıştı [17]. Bu çalışmada sistemin, farklı öznitelikler, Saklı Markov Modeli (HMM) Sınıflayıcı ve SVM Sınıflayıcı kullanıldığında gösterdiği performansı incelenmiştir. Aynı sistemin performansını artırmak için, veri tabanına 7 kişiye ait fonem bilgileri ilave edilerek kişi çeşitliliği artırılmıştır. Çoklu sınıflandırma yapan katmanların sayısı artırılarak, katman başına düşen sınıf sayısı azaltılmış olup, fonem ayırt etme gücü artırılmıştır[18].

GMM genellikle, HMM'deki HMM fonetik / durum etiketleri (Q) tarafından koşullandırılan akustik özellik vektörlerinin (A) olasılığını (p (A | Q)) hesaplarken kullanılır[19]. HMM, genellikle ASR sistemleri tarafından akustik model olarak kullanılır. İfadedeki ses birimleri GMM-HMM modeli kullanılarak tahmin edilir. Daha sonra

konuşulan kelime veya sürekli kelime kümesi belirlenir [20]. HMM'deki her durumun posterior olasılığı, öznitelik olarak Mel Frekans Kepstral Katsayısını (MFCC'ler) kullanan bu mimariler tarafından tahmin edilir. Türk dili için ASR sorunu geleneksel yöntemlerle çözülmeye çalışılmıştır. Bunlardan biri, daha çok uygulamalarda kullanılan GMM-HMM'dir [21-23]. [21] 'de, Fonem Hata Oranı (PER)% 29,3, [22], [23]' te gözlenen Kelime Hata Oranı (WER) sırasıyla %21,46 ve %32,88'dir. Türkçe ASR'nin performansı, Derin öğrenmeye benzer gelişmiş yöntemler kullanılarak geliştirilebilir.

ASR uygulamalarında, konuşma tanıma için Gauss karışımları, Derin Öğrenme yaklaşımıyla başarılı bir şekilde değiştirildi. Önceki çalışmalar, ASR uygulamalarında DNN'lerin GMM-HMM sistemlerinden daha iyi çalıştığını göstermektedir [20]. DNN eğitimli gizli katmanlar içerir. Gizli katmanlar, çok sayıda HMM durumu için gerekli olan büyük çıktı sağlar. Her fonem, çok sayıda duruma neden olan triphone HMM'leri tarafından modellenmiştir. Birçok çalışma, büyük veri kümelerine ve kelime dağarcığına sahip ASR'ler için kullanılan akustik modellemede DNN'nin GMM'den daha iyi performans gösterdiğini göstermektedir. GMM'leri kullanmak yerine, DNN'ler çıktı olarak HMM durumları üzerinden posterior olasılıklar üretmek için kullanılır. Çalışmada, akustik özellik vektörlerinin olasılığı hesaplanırken önceden eğitilmiş DBN-DNN kullanılmıştır. HMM veya HMM durumları, konuşma birimlerini (telefonlar, alt telefonlar, telefon durumları, heceler, kelimeler vb.) Modellemek için kullanılır. Derin Öğrenme, Türkçe konuşmanın tanınmasında hala etkili bir şekilde uygulanmamaktadır. Türk Dilinin DNN tabanlı ASR Uygulamaları ile ilgili az sayıda çalışma bulunmaktadır. [19] 'da, Türkçe sözlü dersleri işlemek için otomatik bir dikte ve anahtar kelime arama sistemi tasarlanmıştır. DNN tabanlı bir LVCSR sistemi geliştirildi. Amaçlanan sistem, Türkçe Haber TV program kayıtları ve Hukuk dersi video kayıtları ile eğitilmektedir. [24] 'de DNN, akustik model oluşturmak için kullanılmaktadır. [25] 'te GMM ve DNN tabanlı modeller, [26]' da geliştirilen derlem kullanılarak eğitilir ve test edilir. Gözlemlenen WER'ler sırasıyla% 14.18,% 12.1 ve% 14.65 olarak rapor edilmektedir. Söz konusu çalışmalar dikkate alındığında, Türk ASR sistemlerinin performansı, alt kelime (morfem) tabanlı dil modelleri uygulanarak ve GMM akustik modelleme DNN tabanlı modellerle değiştirilerek iyileştirilebilir. Söz konusu çalışmalarda alt kelime ve kelime tanıma performansları dikkate alınarak tüm performans ölçütleri tanımlanmıştır.

DNN'ler, akustik çerçevelerin yalnızca sabit boyutlu kayan pencerelerini modelleyebilir, ancak farklı konuşma hızlarını modelleyemez. Tekrarlayan Sinir Ağları (RNN'ler), gizli katmanlardaki döngüler içeren başka bir ağ sınıfıdır. Önceki zaman adımındaki bilgi tutulur ve mevcut adımdaki değer, bu döngüler yardımıyla tahmin edilir. Bu şekilde, RNN'ler farklı konuşma hızlarını idare edebilir [7]. Geçici bağımlılıklar, konuşma tanıma sorununun çözümü sırasında sorun teşkil etmektedir. ASR'ye bağlı olarak uzun veya kısa vadede zamansal bağımlılıklar ortaya çıkabilir. RNN'ler, kaybolan / patlayan gradyan sorununa bağlı olarak yalnızca kısa vadeli bağımlılıkları hesaba katar. RNN'ler son yıllarda konuşma tanıma problemlerine uygulanmıştır. RNN'ler konuşmadaki dinamik süreci daha iyi idare edebildikleri için, geleneksel ileri beslemeli ağa kıyasla iyi bir seçimdir [27]. DNN'lere kıyasla, RNN'lerin verileri geri çağırmalarına izin veren ek tekrarlayan bağlantıları ve hafıza hücreleri vardır. Önceden belirlenen kelimelere ve sıralamalara göre bir sonraki kelime tahmin edilir.

LSTM ve GRU olarak adlandırılan iki tür RNN ağı kullanılır. RNN ağları, konuşma tanıma, doğal dil işleme, görüntü tanıma dahil olmak üzere birçok alanda yaygın olarak kullanılmaktadır. RNN'ler, son teknoloji karakter tanıma yöntemleri haline geldi. Unutma geçidi (LSTM için) veya güncelleme geçidi (GRU için) gibi RNN'lerdeki kapılar daha uzun bağlamsal bağımlılıkları koruyabilir. Hata bilgilerinin geriye doğru yayılması ve geçmiş bilgisinin düzgün yayılması için kısayollar sağlayan bir otoyol kanalı kapılar tarafından inşa edilmiştir. GRU'nun mimarisi, LSTM'ninkinden daha az karmaşıktır. Unutma geçidi ve giriş geçidi tek bir güncelleme geçidinde birleştirilir. Daha az parametreye sahip olan GRU'da ara bilgileri depolamak için ayrı bir "hücre" yoktur. Birçok sıralı öğrenme görevinde GRU'nun yaygın kullanımı ile bellek veya hesaplama zamanından tasarruf sağlanmıştır [7,8,9,28-31]. Türk dili ile ilgili çalışmalar da RNN'lerin ve LSTM'lerin konuşma tanımada zamanla giriş özelliklerindeki değişikliklere duyarlı DNN'ye göre avantajlı olduğunu ve daha başarılı sonuçlar verdiğini göstermektedir [32-34].

Açık konuşma tanıma araçları olan Kaldi ASR araç kutusu, Daniel Povey [35] tarafından hazırlanmıştır. Bu araç çalışmamızda, Kaldi, Türk dilinin ASR eğitimi ve kod çözme işlemlerini, dil modelleri ve akustik modeller oluşturan DBN, LSTM ve GRU'yu kullanarak gerçekleştirmektedir.

Konuşmayı İşaret diline çeviren çaışmalar incelendiğinde, Fakat ne yazık ki, Türk İşaret Dili (TİD) üzerine yapılan çalışmalar oldukça sınırlıdır. Diğer dillerde yapılan çalışmalar incelendiğinde, bir kısmı konuşmayı işaret diline çevirmeyi hedeflerken, bir kısmı da yazılı metni işaret diline çevirmeyi hedeflemiştir.

Konuşmayı işaret diline çeviren çalışmalardan, [36], İngilizceyi İngiliz İşaret Diline, [37], İngilizce'yi Malezya İşaret Diline, [38], İngilizce'yi Amerkan İşaret Diline, [39], İspanyolca'yı İspanyol İşaret Diline çevirmektedir.

Yazılı metni işaret diline çeviren uygulamalar, kural tabanlı , kural tabanlı olmayan ve örnek tabanlı bilgisayarlı çevirme yöntemini kullanan, İngilizce, Almanca yazılı dillerinden İngiliz, Amerikan ve Almanca İşaret dillerine geçiş amacıyla gerçekleştirilmiştir. Amerikan İşaret Dili [40,41] başta olmak üzere, İspanyol İşaret Dili [42] ve Arap İşaret Dili [43] gibi birçok farklı işaret dili üzerine literatürde bilgisayarlı çevirme çalışması bulmak mümkündür. TİD üzerine yapılan çalışmalar incelendiğinde, , [44]'de Türkçe ile TİD arasında çift yönlü metinsel bir makine çeviri sistemi geliştirmiştir. [45,46]'da Türkçe cümlenin İşaret diline çevrilebilmesi için ara metni elde edilmiştir. Ara metin, TİD metnine dönüştürülmüştür. [47]'de örnek ve kural tabanlı yöntemler harmanlanarak ve özgün bir eşleşme algoritması eklenerek doğru eşleştirilen videonun gösterilmesi hedeflenmiştir. Boğaziçi Üniversitesi [48], Türk Dil Kurumu [49] ve Milli Eğitim Bakanlığı'nın [50] geliştirdikleri sözlük uygulamaları da bulunmaktadır.

Çalişmamızda TİD metninin elde edilmesi için [47]'deki algoritma ve kurallardan yararlanılmış ve giriş metninin cümle bazında anlamını kaybetmeden çevrilmesi hedeflenmiştir.

1.3. Konunun Amacı

Bu çalışmanın amacı, literatür taramasında rastlanmayan Türkçe Konuşmayı Türk İşaret Diline çeviren bir sistem tasarlamaktır. Türkçe Konuşma Tanıma adımında, sistematik bir Derin Öğrenme tabanlı ASR tasarımı önerilerek, Fonem Hata Oranı (PER) ve WER üzerindeki etkiyi azaltarak ASR performansını iyileştirmektir. Çalışmalarımızda, 2005 yılında Linguistic Data Consortium (LDC) tarafından kabul edilen standart bir derlem olan "Turkish Microphone Speech v1.0" önerilen sistemi gerçekleştirmek için kullanılmıştır. Derlem önceki çalışmalarda [21], [22], [23], [51] kullanılmıştır. Benzer çalışmalarda [24],[25],[52], yazarlar tarafından sağlanan veri tabanları kullanılmış ve başka

çalışmalarda kullanılmamıştır. Veri tabanları standart değildir ve LDC tarafından onaylanmamıştır. Standart veri setinin kullanılması, uygulanan yöntemin doğruluğunu onaylar ve çalışmayı aynı veri setini kullanan önceki yayınlarla karşılaştırma fırsatı sunar. Örneğin, çalışmamızın GMM tabanlı tanınmasının sonucu [21] 'de elde edilen sonuca çok benzemektedir.

Daha önceki çalışmalarda ise alt kelime tabanlı dil modeli, eğitim için kullanılmaktadır. Türkçe'nin fonem tabanlı bir dil olması nedeniyle, Türkçe'deki tüm ses birimleri de bir alt sözcük olarak eğitilir ve sözlüğe girilir. Alt kelime veya kelime tanımanın başarısız olması durumunda, fonem tabanlı tanıma, tanınmayan kelime veya alt kelimenin ses birimi bileşenlerini bulur. Fonem bileşenleri birleştirilir ve tanınmayan kelime oluşturulabilir.

[25] ve [52] 'da, Derin Öğrenme tabanlı ASR sistemleri yalnızca kelime tabanlı ASR'yi destekler ve önerilen sistemlerin performansı yalnızca Kelime Hata Oranı (WER) ölçüsü ile ölçülür. Çalışmamızda uygulanan yöntemin performansı da Fonem Hata Oranı (PER) metriği ile de ölçülmüştür. Türk Dili için fonem temelli ASR ile ilgili önceki çalışmalarımız [17], [18] daha önce 2016 ve 2017 yıllarında yayınlanmıştır. Türkçe fonem temelli bir dil olduğu için, kelime dağarcığı dışı (OOV) kelimelerin yardımıyla tanınan ses birimi bileşenleri. Literatür taramasında Türkçe için derin öğrenme temelli herhangi bir fonem tanıyıcıya rastlanmamıştır.

Eğitim ve test için, gizli katmanların özellikler tarafından önceden eğitildiği DBN kullanılır. Benzer çalışmalar dikkate alınarak referanslarda standart (geleneksel) sıralı DNN ve Zaman Gecikmeli Sinir Ağı (TDNN) yapıları uygulanmıştır [25,52]. Referans [25]'te güncel ASR uygulamalarında DBN için uygulanmaya başlandığını bildirmektedir. Çalışmamızın sonuçları, DBN'nin doğruluğunun önceki DNN ve TDNN tabanlı çalışmalara göre% 1.5 daha yüksek olduğunu göstermektedir [10].

Çalışmamızda, önerilen DBN, LSTM ve GRU tabanlı ASR sistemlerinin performansları geleneksel tanıma yöntemi, aynı derlemi kullanan GMM tabanlı sistemler ile karşılaştırılmıştır. ASR sistemleri "Türkçe Mikrofon Konuşma Derlemi (ODTÜ 1.0)" veritabanına uygulanmıştır [21]. Önerilen sistemde, eğitim için Alt Kelime tabanlı LM, GMM-HMM ve Derin öğrenme tabanlı ASR kodlaması için Kaldi kullanılmıştır [35]. Performans ölçümleriyle ilgili olarak, uygulanan yöntemlerin tanıma oranları aynı veri seti kullanılarak önceki çalışmalarla [21,22] karşılaştırılmıştır.

Konuşma Tanıma adımında performansa katkısı bulunan güncel Derin Öğrenme metodları kullanılmış olup, performansta artış olduğu kullanılan standart veri setine yer vermiş diğer çalışmalar ile doğrulanmıştır. Derin Öğrenme Metoduna yer veren Türkçe Konuşma Tanıma alanındaki çalışmaların performansları araştırmacıların kendi hazırlayıp kaydettikleri veri tabanları test edilmiştir.

Türkçe Konuşmanın Türk İşaret Diline çevrildiği çalışmaya literatürde rastlanmamış olup, bu konuda yapılacak çalışmalara öncü olacak niteliktedir.

2. TÜRK DİLİNİN MORFOLOJİSİ

Türkçenin sondan eklemeli bir morfolojisi vardır. Birkaç son ekin eklenmesi, birçok yeni kelimeyi tek bir kökten türetebilir. Önek Türkçede kullanılmaz. Aşağıdaki örnekler, sıralı sözlü ve nominal çekimleri göstermektedir. Nominal bükülme daha az karmaşıktır [53] (Bkz. Tablo 2.1).

Tablo 2.1. Fiil ve İsim çekim örnekleri

İsim Çekimi	ev-im-de-ki-ler-den
Fiil Çekimi	yap-tır-ma-yabil-iyor-du-k

Sondan eklemeli dillerde bir alt-kelimenin ardından başka bir alt-kelime eklemek mümkündür. Her alt-kelime zaman, durum, anlaşma gibi belli bir morfolojik bilgiyi taşır. Sondan eklemeli dillerin bu özelliği, aynı köke sahip fakat farklı son ekler almış çok sayıda kelimenin oluşturduğu geniş kelime dağarcıklarına yol açar. Sözlükte çok fazla kelime bulunduğundan, modellenmemiş çok sayıda delime dağarcığı dışında sözcük olacaktır[54]. Daha büyük sözlük boyutu, kelime tabanlı konuşma tanımada kod çözücünün hızını düşürür [55]. Yüksek dağarcık dışı kelime sayısı nedeniyle, İngilizce için kullanılan konuşma tanıma yöntemleri, Türkçe için düşük tanıma sonuçları vermektedir.

Türkçenin DM açısından diğer bir önemli özelliği serbest kelime sırasıdır. Özne-Nesne-Fiil kelime sıralaması Türkçe'de tipik bir özelliktir, ancak bazı söylem koşullarında, başka sıralamalar da mümkündür. Cümleler anlamları değişmeden farklı kelime sıralamaları ile yeniden oluşturulabilir ancak, n-gram DM'in karmaşıklıkları artmaktadır [55]. Sonuç olarak, Türkçe LVCSR'de kapsama problemini çözmek için kelimelerden farklı alt birimler kullanılmalı ve DM parametrelerini güvenilir bir şekilde eğitmek için büyük miktarda eğitim verisi kullanılmalıdır.

2.1. Türkçe'de sesbirimler

Sesbirimlerin simgesel olarak ifade edilmesi sonucu oluşan simgeler fonem (phoneme) olarak adlandırılır. Türkçede her bir sesbirim alfabetik bir simge ile ifade

edilebildiğinden harfler aynı zamanda fonem olarak da adlandırılabilir. Bu nedenle Türkçe fonem tabanlı veya fonetik bir dildir. Türkçe'de 29 harf, dolayısıyla 29 fonem vardır[17].

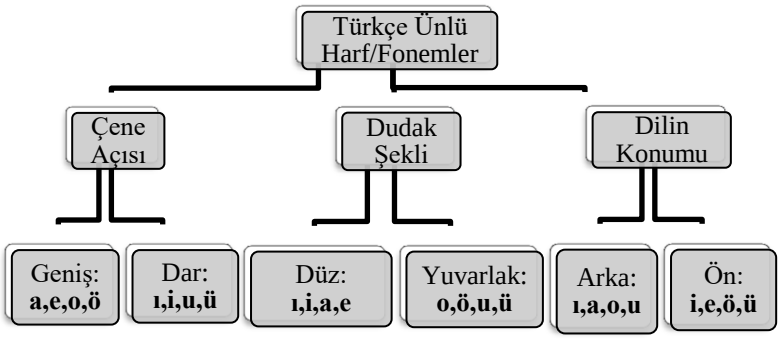
Fonemler, ses tellerinin titreşimine göre ve ünlü, ünsüz olma durumuna göre sınıflandırılır. Ötümlü fonemler ses tellerinde titreşim meydana getirir. Ötümsüz fonemlerde ses tellerinde titreşim olmaz. Ünlü fonemlerin tümü ötümlü olup, ünsüz fonemler ise ötümlü ve ötümsüz olmak üzere ikiye ayrılır [17].

Tablo 2.2. Ünsüz Fonemlerin Sınıflandırılması

Ötümlü Ünsüzler	b,d,g,v,z,j,c,l,r,m,n,y
Ötümsüz Ünsüzler	p,t,k,f,s,s,ç,h

Türkçe, Japonca veya Fince gibi fonemik bir diller grubundadır. Yazı dilinde her ses birimi bir harf ile sembolize edilir. Başka bir deyişle, yazılı metin ve telaffuz tam olarak eşleşir. Bununla birlikte, bazı ünsüzlerin ve ünlülerin ses yolunda nerede üretildiklerine bağlı olarak farklı sesleri vardır. [56].

Oluşumları sırasında nefes kanalının ağız kısmının dişler, dil veya dudaklar tarafından engellenmediği seslere ünlü denir. Türkçede ünlüler çene açısı, dudak şekli ve dilin konumuna göre üç grupta sınıflandırılabilir [57] (Bkz. Şekil 2.1, Tablo 2.3).



Şekil 2.1. Türkçe Ünlü harf/fonem Sınıflandırması

Nefesin dişler, dil veya dudaklar tarafından bloke edildiği ve bir hece oluşturmak için bir sesli harfle birleştirilebilen temel konuşma seslerine ünsüzler denir. Ünsüzler ses tellerinin durumuna göre sesli ve sessiz olmak üzere iki gruba ayrılabilir. Sesli ünsüzler, ses tellerini titreştirerek yapılan ünsüz seslerdir. Sessiz ünsüzlerin oluşumu sırasında ses

telleri titreşmez. Ünsüzler ayrıca çıktı türleri ve çıktı konumlarına göre sınıflandırılabilir [57] (Bkz. Tablo 2.4).

Tablo 2.3 Türkçe Ünlü Sınıflandırma

Türkçe Ünlüler	*Çene Açısı*	Geniş	**a,e,o,ö**
		Dar	**ı,i,u,ü**
	Dudak Şekli	Düz	**a,e,ı,i**
		Yuvarlak	**o,ö,u,ü**
	Dilin Durumu	Arka	**a,ı,o,u**
		Ön	**e,i,ö,ü**

2.2. Türkçe için kullanılan Dil Modelleme Tipleri

Bir Dil Modeli (LM), metin örneklerine dayalı olarak kelime var olma olasılığını elde eder ve verilen kelimelerin zincirinde aşağıdaki kelimeyi tahmin edebilen olasılıksal modeller geliştirir. Geniş bir kelime listesi ve bunların gerçekleşme olasılıklarını içerir. Daha büyük modeller cümleleri veya paragrafları tahmin edebilir. Önerilen sistem için alt kelime tabanlı model tercih edilmiştir [10].

ASR problemlerinde en sık kullanılan Dil modelleri şunlardır:

1. Kelime Tabanlı
2. Alt Kelime Tabanlı

Kelime Tabanlı Model modelleri, İngilizce gibi analitik ve izole edilmiş diller için kullanılırken, Alt Kelime Tabanlı Model, Türkçe ve Fince gibi sonlandırıcı dilleri modellemek için kullanılır.

Türkçe sondan eklemeli bir dildir. Kelimenin köküne birçok son ek eklenerek yeni kelimeler türetilebilir. Önek Türkçede kullanılmaz. LM açısından Türkçe'nin diğer önemli özelliği serbest kelime sırasıdır. Özne-Nesne-Fiil kelime sıralaması Türkçede tipik bir özelliktir, ancak bazı durumlarda başka tür sıralamalar da mümkündür. Cümleler anlamlarını değiştirmeden farklı kelime dizileri ile yeniden yapılandırılabilir, ancak n-gram LM'nin karmaşıklığı artmaktadır [55].

Tablo 2.4 Türkçe'de Ünsüz Sınıflandırması

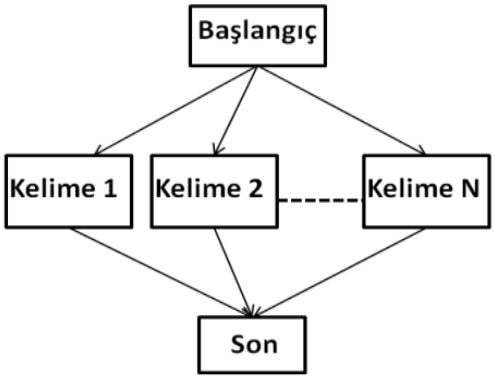
Türkçe Ünsüzler	*Çıkış Biçimi*	Patlamalı	**b,d,g,p,t,k**
		Geniz	**m,n**
		Çarpmalı	**r**
		Yan Daralma	**l**
		Sürtünücü	**c,ç,f,h,j,s,ş,v,y,z**
	Çıkış Yerleri	Çift Dudak	**b,p,m**
		Dudak-Diş	**f,v**
		Dil ucu – Diş ardı	**d,t**
		Dil ucu – Diş eti	**n,r,s,z**
		Dil-Ön damak	**c,ç,j,ş,y**
		Dil ucu – Ön damak	**l**
		Dil-Art damak	**g,k**
		Gırtlak	**h**
	Ses Tellerinin Durumu	Ötümlü	**b, c, d, g, j, l, m, n, r, v, y, z**
		Ötümsüz	**ç, f, h, k, p, s, ş, t**

Sonuç olarak, Türkçe ASR'de kapsam problemini çözmek için alt kelimeler kelimelerden farklı kullanılmalı ve LM parametrelerini güvenilir bir şekilde eğitmek için büyük miktarda eğitim verisi kullanılmalıdır.

Sonlandırıcı dillerde başka bir alt sözcük ve ardından bir alt sözcük eklemek mümkündür. Her alt kelime, zaman, durum ve anlaşma gibi belirli morfolojik bilgileri taşır. Sondan eklemeli dillerin bu özelliği, aynı köke sahip ancak farklı soneklere sahip birçok kelimeden oluşan geniş bir kelime dağarcığına yol açar. Sözlükte çok fazla kelime olduğundan, sözlükten modellenmemiş birçok kelime çıkacaktır [56]. Daha büyük sözlük boyutu, kelime tabanlı konuşma tanımada kod çözücünün hızını azaltır [55]. Kelime dışı kelime sayısının çok olması nedeniyle, İngilizce için kullanılan konuşma tanıma yöntemleri, Türkçe için düşük tanıma başarı sonuçları vermektedir.

2.2.1 Kelime Tabanlı Model

Kelime tabanlı model, kelimeleri tanıma birimleri olarak kullanan en temel LM yaklaşımıdır. Kelimeler, konuşma tanıma için sözlük girişleri olarak seçilir ve LM olasılıkları, eğitim derleminden kelimelerin birim olarak kullanılması ile çıkarılır [58]. Sözcük tabanlı sistemin yapısı Şekil 2.2'de gösterilmiştir. Sözcük tabanlı LM, sözcük başına düşük alt sözcüklerle analitik ve izole dilleri modellerken tercih edilir. Örneğin İngilizce ve Mandarin Çincesi bu gruptadır.

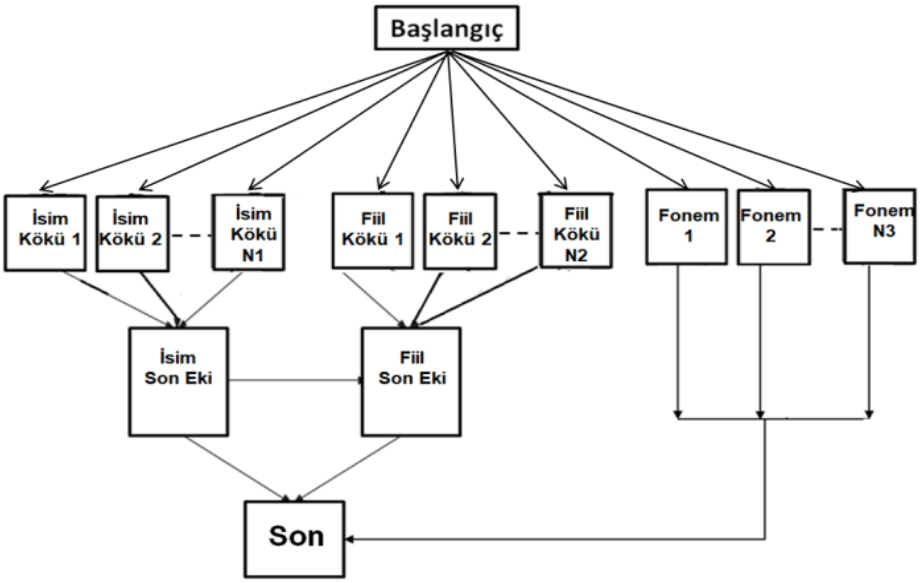


Şekil 2.2. Kelime tabanlı model

2.2.2 Alt Kelime Tabanlı Model

Sub-Word (morfem) tabanlı LM, sondan eklemeli dillerin konuşma tanıma sistemlerinde kelime tanıma başarısını geliştirir. Alt kelime tabanlı LM, konuşma tanımanın temel birimleri olarak alt kelimeleri kullanır. Şekil 2.3'de görüldüğü gibi kelimeler, Türkçe'nin yazım kurallarına ve morfolojisine uygun olarak kök ve eklerden oluşmaktadır. Bu yapı dikkate alınarak modelleme yapılır. Alt kelimeler arasındaki geçişler bigram olasılıkları ile ağırlıklandırılmıştır [10].

Fonetik kurallar, kök ve son ekler arasında bağlantı kurmak için kullanılır. Son ses birimi ve kökün son sesli harfi, belirli bir kökü takip edebilecek ekleri belirler [10].



Şekil 2.3. Alt-Kelime tabanlı model

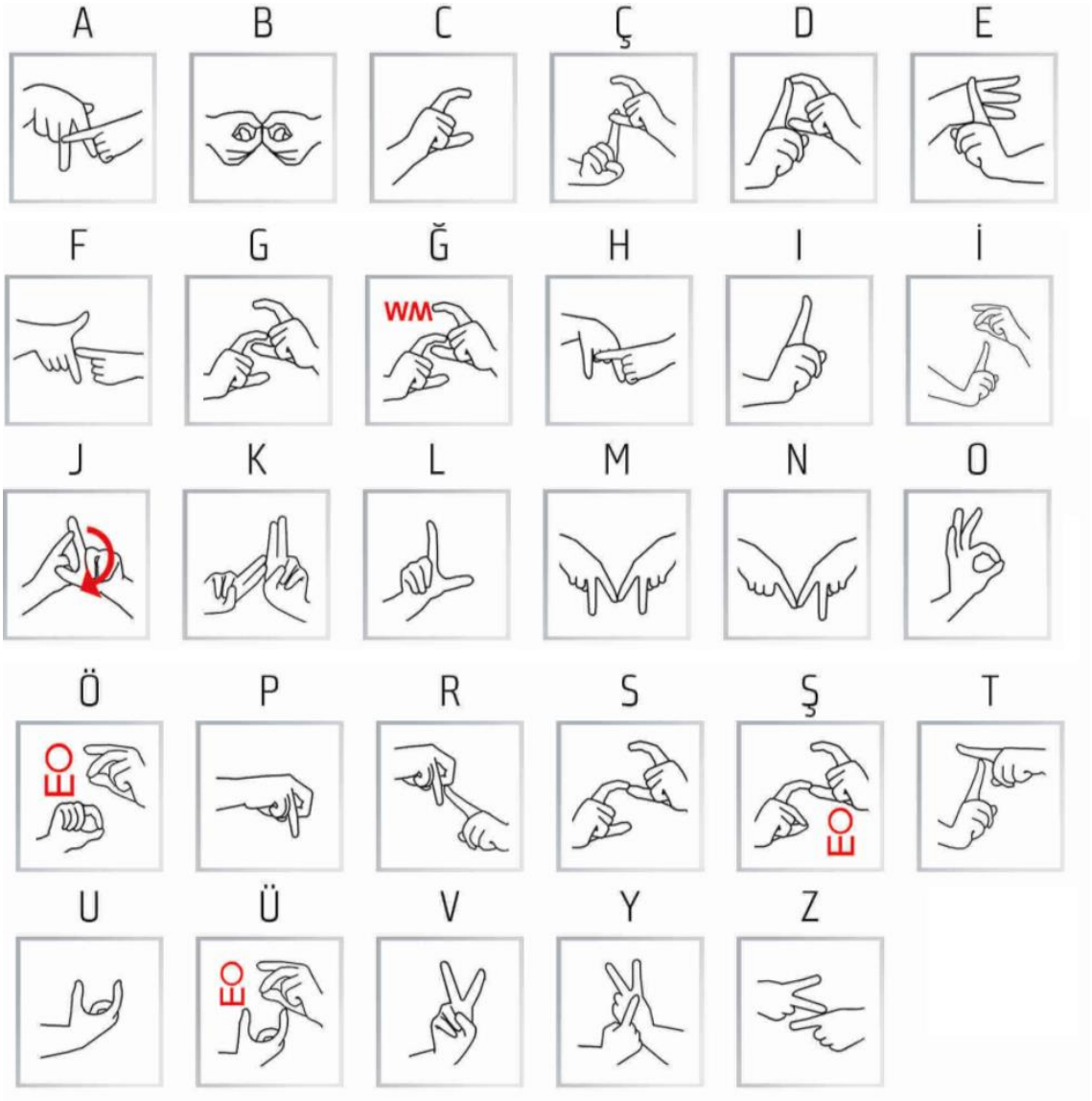
3. TÜRK İŞARET DİLİ

Türk İşaret Dili (TİD), Türkiye'de kullanılan işaret dilidir. Bazı bölgesel farklılıklar bulunmasına rağmen, TİD'in tüm Türkiye'de kullanıldığı bilinmektedir. TİD'in farklı bir işaret dilinden türetildiği ya da tarihte farklı bir işaret dilinin etkisinde kaldığı yönünde düşünmek için yeterli kanıt yoktur. Fakat TİD, İngiliz İşaret Dili ile benzerlikler göstermektedir. TİD tarihi, Osmanlı dönemine kadar uzanır. [47]

17. Yüzyılda işaret dili sadece işitme engelliler tarafından değil, duyabilen insanlar tarafından da kullanılmaya başlanmıştır. Osmanlı Devleti'nde ilk işaret dili okulu padişah 2. Abdülhamit tarafından 1902'de açılmıştır. Bu okulda Osmanlı İşaret Dili'nin sözel dil ile beraber kullanıldığı bilinmektedir. Her ne kadar o yıllarda kullanılan Osmanlı İşaret Dili'nin günümüzde kullanılan Türk İşaret Dili'nin alt yapısını oluşturduğu düşünülse de, Osmanlı İşaret Dili'nde kullanılan alfabe ile TİD alfabesi oldukça farklıdır.[47]

3.1. TİD'in Özellikleri

TİD, genel olarak kelimelerle ifade edilen bir dildir. TİD'de 750 civarında kelime tespit edilmiştir. Bu kelimeler değişik kategorilerle sınıflandırılmıştır. Bu kategoriler; alfabe, sayılar, zamanla ilgili kavramlar, görsel kavramlar, hayvanlar, meslekler, yer isimleri, zamirler ve anlatımlar olarak belirtilmiştir. Genelde TİD belirlenmiş olan bu kelimeler üzerinden ifade edilmektedir. Fakat kelimelerin yeterli olmadığı durumlarda ya da özel isimlerin geçtiği yerlerde harf işaretleri ile heceleme yapılmaktadır. Türkçe'de yer alan harflerin hepsinin TİD'de bir karşılığı vardır. Tüm harfler tek tek el hareketleri ile ifade edilebilmektedir. Bu da TİD'e el hareketlerinden oluşan bir alfabe kazandırmaktadır[47]. Şekil 3.1'de TİD Alfabesi gösterilmiştir.



Şekil 3.1 TİD Alfabesi [50]

3.2. TİD'in Türkçe ile arasındaki farklar

TİD ve konuşulan Türkçe arasında önemli farklılıklardan birisi sondan ekleme özelliğidir. Türkçe sondan eklemeli bir dildir. Fakat TİD'nin dil bilgisi ile ilgili özelliklerine baktığımızda sondan eklemeli olma özelliğini göstermemektedir. Türkçe'de, özellikle yüklemlerde, cümlenin anlamına katkıda bulunan birçok cümle öğesi ek şeklinde kodlanabilir. Mesela Türkçe'de sık olarak zamirlerin yüklem içinde ek olarak kodlandığı görülür. Fakat TİD'de zamirler yüklem üzerine kodlanamaz. Örnek vermek gerekirse,

“çalışıyorum” cümlesi içerisinde zamirin yükleme ek olarak kodlandığı görülür. Fakat aynı cümlenin TİD karşılığı “ben + çalışmak” ya da “ben + çalışmak + ben” şeklindedir. Örnekte de görüldüğü gibi, TİD’de zamirler yükleme ek olarak kodlanmak yerine ayrı bir kelime olarak cümleye eklenmiştir.[47]

TİD ve konuşulan Türkçe ile arasındaki farklılıklardan biri de zaman kavramının cümle içinde belirtilme şeklidir. Türkçe’de zaman kavramı genellikle yükleme ek olarak kodlanır. Fakat TİD’de bu durum farklılık gösterir. Cümledeki zaman kavramının “önce”, “sonra” gibi zaman anlamı içeren sözcüklerin yanı sıra “tamam” ve “bitti” gibi tamamlanmışlık anlamı içeren sözcükler tarafından katıldığı görülür. Örnek vermek gerekirse, “geliyorum” ile “geleceğim” cümlelerinin TİD çevrimleri aynıdır. Fakat cümle içine eklenen zaman anlamı içeren sözcük ve ya kelime öbekleri ile zaman bilgisi cümleye eklenebilir.[47]

TİD’de olumsuzluk anlamı genelde başı yukarı doğru kaldırarak, kaşları yukarı doğru kalkık vaziyette, tek eli ya da her iki eli havaya kaldırarak verilir. Bu hareket aynı zamanda “değil” anlamına da gelmektedir. TİD’de soru sorma yöntemleri de dilbilimsel olarak Türkçe ‘de olduğundan farklıdır. [47]

4. KONUŞMA TANIMA PROBLEMİ VE KULLANILAN GELENEKSEL YÖNTEMLER

Genel olarak, ASR sisteminin amacı konuşma verilerini almak ve seslendirilenleri cümlelere veya kelimelere çevirmektir. Tahmin işlevselliği, özellik çıkarma ve dil analizi adımlarıyla gerçekleştirilir. Özellik çıkarıcı, konuşma sinyalinden akustik özellik dizilerini alır. Özelliklerdeki ses birimlerinin olasılıkları hesaplanır ve sesbirimleri tahmin eden sınıflandırıcıya aktarılır. Tahmin yapılırken iki temel modelleme kullanılır. Bunlar Akustik Modelleme ve Dil Modellemesidir (LM). Fonem olasılıkları, akustik modeli temsil eden LM ve HMM ile birlikte konuşmayı tanımak için kullanılır. LM yardımı ile kelimeler ve cümleler tahmin edilir.

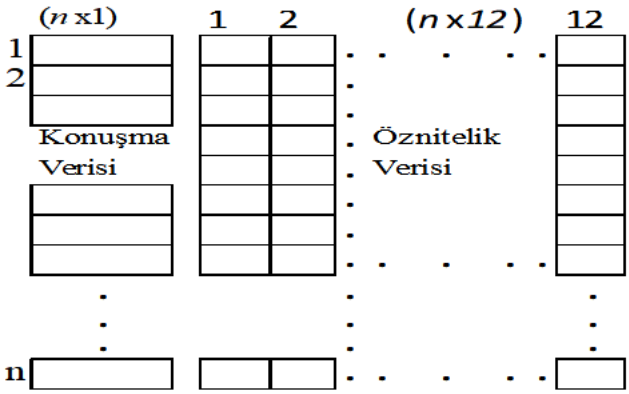
ASR bir sınıflandırma ve örüntü tanıma problemidir. Bu alanda klasik yöntemlerin yanı sıra makine öğrenmesine dayalı uygulamalar da kullanılmaktadır. ASR'de klasik yöntemleri uygulamak için, dilden bağımsız olarak aşağıdaki adımlar uygulanır:

1. Öznitelik Çıkarma
2. Özellik Sınıflandırması
3. Akustik Modelleme
4. Dil Modelleme (LM)

4.1. Öznitelik Çıkarma

Konuşma işaretindeki ses bilgisinin özniteliklerinin bulunabilmesi için en çok başvurulan yöntemler, Doğrusal Öngörü Kepstral Katsayıları (LPCC) ve Mel Frekansı Kepstrum Katsayılarıdır (MFCC). Eğitim aşamasında, öncelikle, öznitelikler çıkarılır. Öznitelikler elde edildikten sonra sınıflandırma düğümünün eğitim aşamasına geçilir. Sınıflandırmanın sonucu olarak, sesli ifadedeki ses bileşenleri fonem dizisine çevrilir[17].

Hazırlanan veri seti matris yapısı Şekil 4.1’de gösterilmektedir. Öznitelik matrisi n satırdan oluşmaktadır. Bu sayı, öznitelik adımında ayarlanan pencere genişliği ve ses parçasının süresine göre değişir. Kolon sayısı ise öznitelik katsayılarını ifade eder[17].



Şekil 4.1 Öznitelik matris yapısı

4.1.1 Doğrusal Öngörü Kepstral Katsayıları (LPCC)

Doğrusal Öngörü Kepstral Katsayılar (LPCC) yöntemi, temel olarak LPC katsayılarının Fourier dönüşümü ile kepstral katsayılara dönüştürülmesi olarak tanımlanabilir[17].

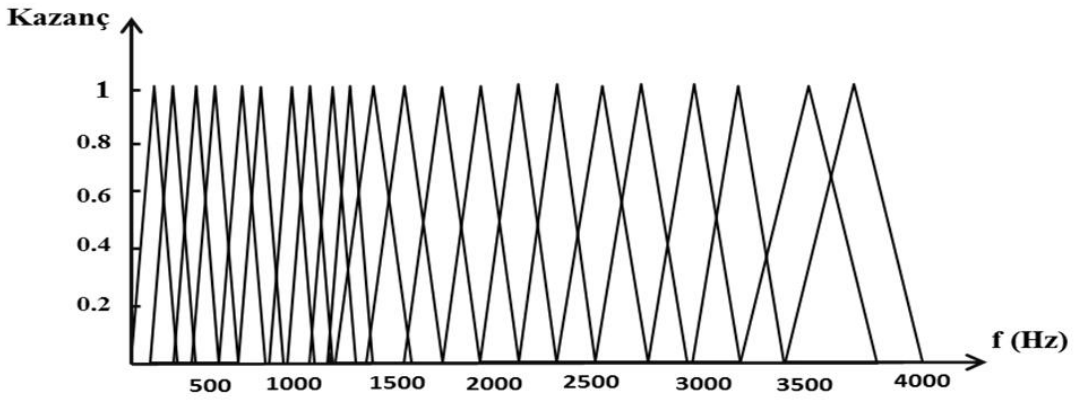
Doğrusal Öngörü Kodlama yönteminde, n zamanda verilen s(n) ses örnekleri, önceki p tane ses örneğinden yaklaşık olarak Eşitlik 4.1'deki gibi elde edilebilir.

$$S(n) = \sum_{i=1}^{p} a_i s(n-i) \qquad (4.1)$$

Burada p, LPC kodlayıcının derecesi, $a_1, a_2, \ldots, a_p$ ise LPC katsayıları olarak ifade edilmektedir[17].

4.1.2 Mel Frekans Kepstrum Katsayıları(MFCC)

MFCC insan işitme algılamasına dayanmaktadır[59]. Buna göre iki tip filtrelemeden bahsetmek gerekir. Filtre dizileri, 1 kHz'nin altında doğrusal aralıklı, 1 kHz'nin üzerinde logaritmik aralıklıdır. Bu filtreleme sayesinde konuşma işaretindeki önemli fonetik özellikler yakalanır [60]. Şekil 4.2'de Mel filtre dizisi gösterilmektedir.



Şekil 4.2. Mel Filtre Dizisi

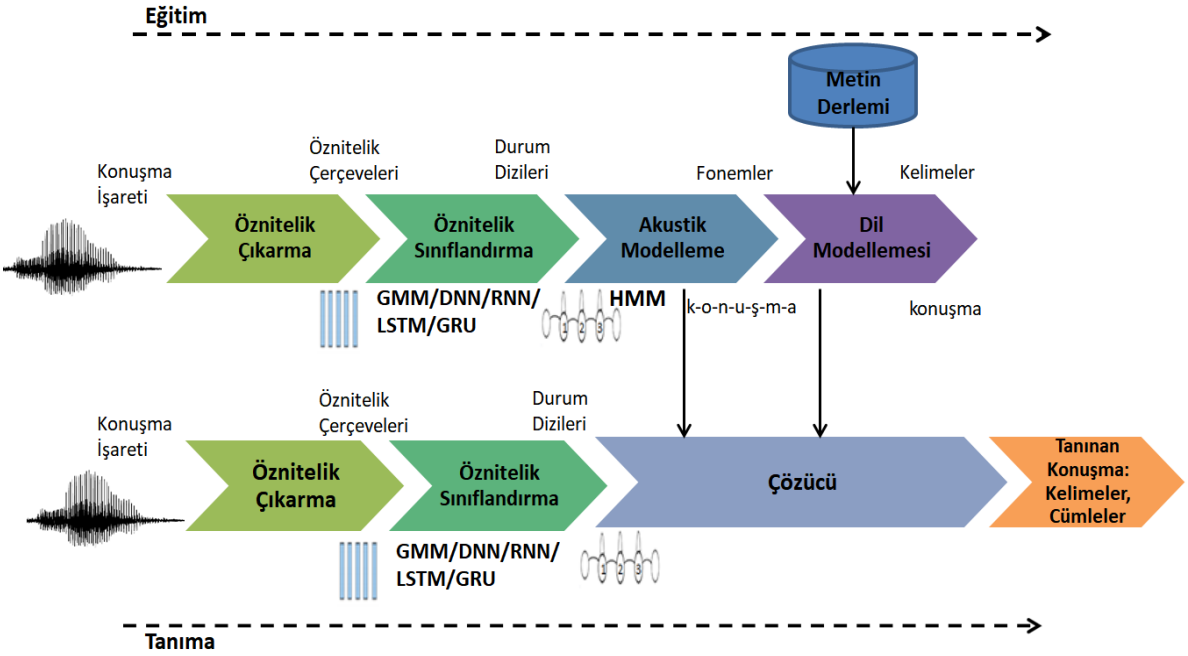
Mel ölçeği, frekans ölçeği olarak, Eşitlik 4.2'deki gibi ifade edilir[59].

$$Mel(f) = 2595xlog\left(1 + \frac{f}{700}\right) \qquad (4.2)$$

Sinyalin Fourier dönüşümü alınır ve Mel filtre dizisinden geçirilir. Bu sayede konuşma tanımada önemli rol oynayacak frekans bileşenleri ayırt edilmeye çalışılır.

4.2. Öznitelik Sınıflandırması

Öznitelik çıkarma adımından sonra, elde edilen katsayılara Gauss olasılık fonksiyonlarından oluşan Gaussian Karışım Modelleri (GMM) uygulanır. GMM, HMM Tabanlı Tanıyıcıda kullanılacak durumların dağılımını modellemek için kullanılır. GMM-HMM Sınıflandırıcısının bu yapısı Şekil 4.3'de gösterilmektedir.



Şekil 4.3. ASR Sistemi

4.2.1 Gauss Karışım Modelleri (GMM)

Konuşma kaydındaki ses birimleri, GMM-HMM (Saklı Markov Modelleri) modeli kullanılarak tahmin edilir. Daha sonra söylenen kelime veya sürekli kelimeler belirlenir [20].

Spektral şekil, GMM'de θm ve $P\ (\omega_m)$ bileşen ağırlıklarına sahip bir M bileşen karışım modeli ile temsil edilebilir. Karışım modeli Eşitlik 4.3'de ifade edilebilir.,

$$p(x|\theta) = \sum_{m=1}^{M} P(\omega_m) p(x|\omega_m, \theta_m) \qquad (4.3)$$

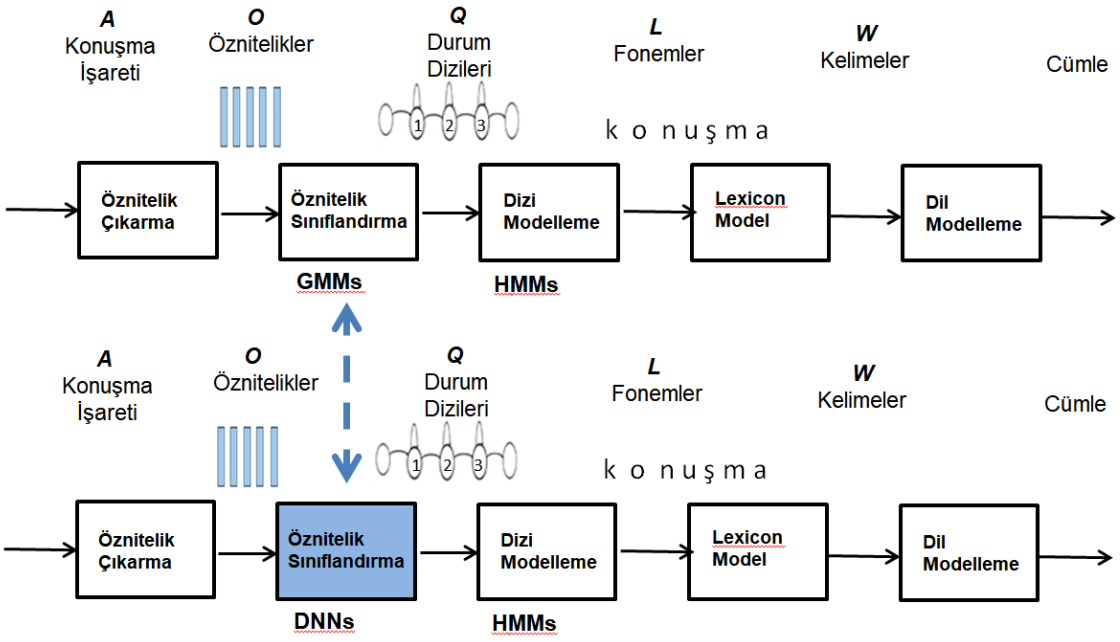
$p(x|\omega_m, \theta_m), m$ ve θ_m bileşen parametrelerinin birincil olasılığıdır. Burada karışım bileşenleri Gauss'tur ve Eşitlik 4.4'teki gibi ifade edilir.

$$p(x|\omega_m, \theta_m) = \frac{1}{\sqrt{2\pi\sigma_m^2}} exp\left[-\frac{(x-\mu_m)^2}{2\sigma_m^2}\right] \qquad (4.4)$$

μ_m, ortalama, σ_m, ω_m bileşenlerinin standart sapmasıdır. Spektral gösterime dayalı sürekli bir PDF oluşturmak, konuşmadan bir GMM hesaplamanın ilk adımıdır. Optimum GMM parametreleri, GMM ile spektral PDF arasındaki mesafenin en aza indirilmesi ile elde edilir[61].

Standart HMM'ler için en yaygın olarak kullanılan uzantı, durum-çıktı dağılımlarının karışım modelidir. HMM Tabanlı Tanıyıcıda, Gauss dağılımı, gözlemlenen öznitelik vektörlerinin tek modlu ve simetrik olduğu varsayılarak durum çıktı dağılımının modellenmesi için kullanılır[61].

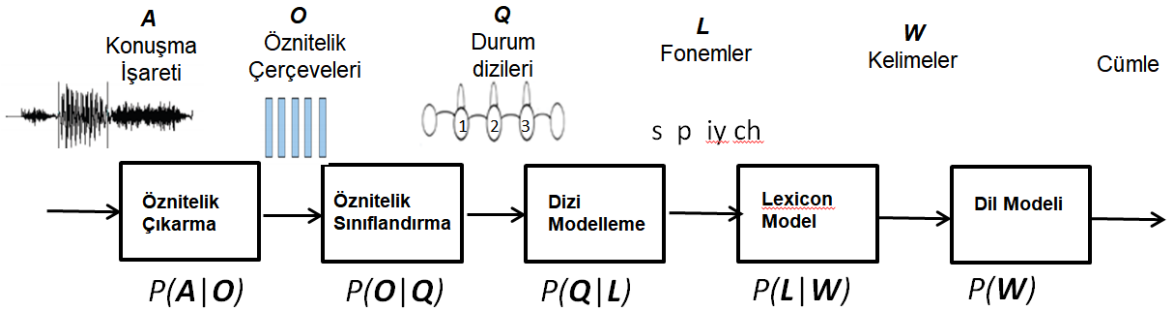
Uygulamada, konuşmacı, cinsiyet ve aksan farklılıkları verilerde birden fazla mod yaratma eğilimindedir. Tek Gauss durumdan çıktıya dağılımını çok modlu ve asimetrik verileri modelleyebilen bir GMM ile değiştirmek bu sorunu çözer (Bkz. Şekil 4.4).



Şekil 4.4. ASR Sistemi GMM ve DNN Çözümleri

4.3. Akustik Modelleme

Çoğunlukla kullanılan ve bilinen ASR stratejileri istatistiksel tabanlı yöntemlerdir[62]. Bir istatistiksel konuşma tanıyıcının blok diyagramı Şekil 4.5'de gösterilmektedir.



Şekil 4.5 İstatistiksel ASR sistemi

ASR'nin amacı konuşma sesini metne dönüştürmektir. Bu prosedür, aşağıdaki gibi istatistiksel olarak ifade edilebilir.

$O = (o_1, o_2, \ldots, o_n)$ (Konuşma vektörlerinin dizisi, o_i, i zamanındaki vektör) akustik gözlemler kümesidir ve $W = (w_1, w_2, \ldots, w_n)$ kelimelerin dizisidir. Maksimum olasılık şu şekilde hesaplanabilir:

$$\widehat{W} = arg_w max P(W|O) = arg_w max \frac{P(W)P(O|W)}{P(O)} \qquad (4.6)$$

Eşitlik (4.7) Bayes kuralını kullanır ve en olası kelime sırasını belirtir. $P(O)$, konuşma ifadesinin olasılığını temsil eder. W dizisinden bağımsızdır ve ihmal edilebilir. Böylece, Eşitlik (4.6) aşağıda gösterildiği gibi basitleştirilmiştir [62]:

$$\widehat{W} = arg_w max P(W)P(O|W) \qquad (4.7)$$

Eşitlik (4.7) iki ana faktör içerir. Bunlar, Kelime dizisi olasılığı $P(W)$ ve $P(O \mid W)$, kelime dizisi için akustik verilerin olasılığıdır. $P(W)$ değeri LM'ye bağlıdır ve $P(O \mid W)$ akustik modele göre hesaplanır. Her iki model de ayrı ayrı oluşturulabilir, ancak bir konuşma sinyalini tanırken birlikte çalışırlar. Akustik modellemenin temeli HMM'ler tarafından temsil edilmektedir.

4.3.1 ASR için Saklı Markov Modelleri (HMM)

GMM fonksiyonları ve durum geçiş olasılıkları, HMM'de sınıflandırma için kullanılır. HMM'ler stokastik sonlu durum makineleri olup, ASR uygulamalarında akustik modeller ve LM'ler oluşturur [63]. Geçişlerle ilgili bir dizi durum içerirler (Bkz. Şekil 4.5). Markov süreci "gizli" olarak adlandırılır çünkü gözlemci durum dizisini doğrudan

göremez. Her durum için Olasılık Yoğunluk Fonksiyonundan (PDF) oluşturulan bir konuşma vektör dizisi gözlemlenir.

Şekil 4.6'de verildiği gibi, bir HMM olasılıkları bir durum dizisine atar. Sonuç olarak, aşağıdaki parametreler bir HMM'yi karakterize eder: [62]

- bir dizi durum $S = (s_1, s_2, \ldots, s_N)$ t anında durum: q_t

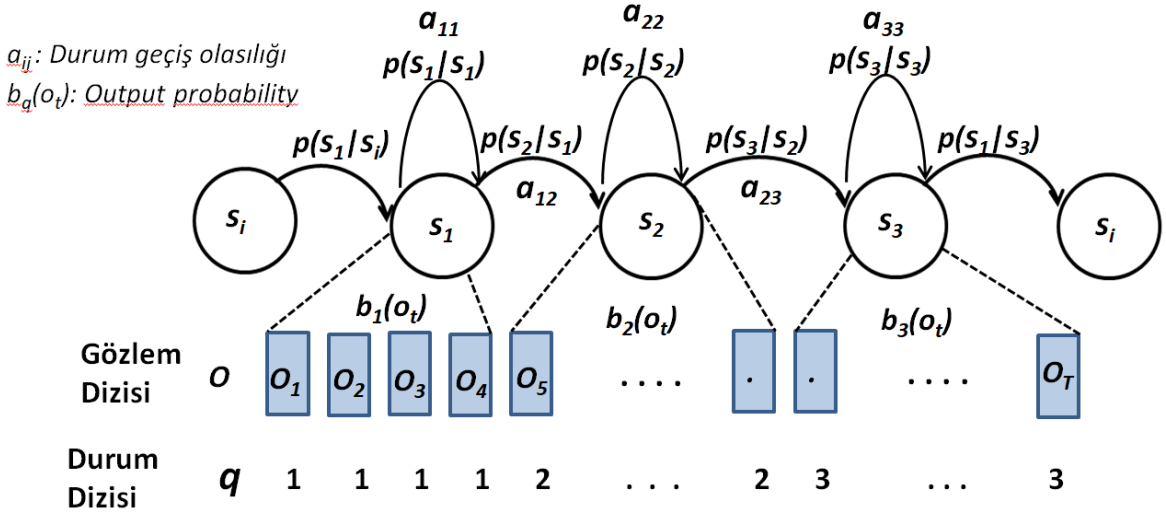
- Durumlar geçiş olasılıkları:

$A = (a_{11}, a_{12}, \ldots, a_{NN})$, her a_{ij} i durumundan j durumuna geçiş olasılığını temsil eder;

- gözlem olasılıkları, her $B = b_i(o_t)$ i durumundan oluşturulan bir o_t gözleminin olasılığını tanımlar;

- ilk durum dağılımları:

$$\pi = \{\pi_i = P[q_1 = s_i],\ i = 1, \ldots, N\}. \quad (4.8)$$



Şekil 4.6 HMM gösterimi

Bir HMM, $\lambda = (A, B, \Pi)$ ile gösterilir. HMM'ler, konuşma sinyalinin küçük zaman dilimlerinde kararlı olduğunu varsayar. HMM'ler konuşma sinyali değişkenliğini iyi yönetebilirler ve konuşma modellemede iyidirler. Konuşma tanıma modellerinde konuşma, soldan sağa sıralı durumlara göre modellenir.

ASR sistemleri için istatistiksel yöntemler genellikle HMM'yi akustik bir model olarak kullanır. GMM, genellikle HMM için akustik özellik vektörlerinin olasılığını hesaplarken kullanılır. GMM aşamasından sonra elde edilen durumların durum geçiş olasılıkları tahmin edilmektedir. Kelime, burada oluşturulan akustik modellerin karşılaştırılmasıyla tahmin edilir.

Cümleleri oluşturan kelimelerin tanınması için kelimeyi oluşturan ses bileşenlerinin veya ses birimlerinin akustik olarak modellenmesi gerekir. Akustik modellemede her bir fonemin olasılığı telaffuz veya konuşma sırasında GMM kullanılarak hesaplanır. Böylece, konuşma sinyalinin zaman içindeki değişimi ve spektral çeşitliliği modellenmiştir.

Ses tanımadaki en küçük akustik birim fonemdir. Sese dayalı modelleme, 3-durumlu HMM ile oluşturulmuştur. Tek durumlu (monofon) modelleme, diğer fonemlerden bağımsız bir ses tanıma sistemi sağlar. Üç durumlu modellemede (triphone), her bir ses birimi komşu birimlerle (sağ ve sol) modellenir. Bu model kullanılarak ses segmentlerindeki akustik farklılıkların ve düzensizliklerin sınıflandırma üzerindeki olumsuz etkileri azaltılmış ve tanıma başarısı artırılmıştır. HMM'ler, konuşma sinyalinin küçük zaman aralıklarında kararlı olduğunu varsayar. HMM'ler konuşma sinyali değişkenliğini iyi yönetebilir ve konuşma modellemede iyidir[62].

4.4. Dil Modeli (LM)

LM, metin örneklerinden yola çıkarak, kelimenin gerçekleşme olasılığını öğrenir ve verilen kelimelerin sırasına göre bir sonraki kelimeyi tahmin edebilen olasılıksal modeller geliştirir. Oluşması için çok sayıda kelime ve olasılık içerir. Daha büyük modeller cümleleri veya paragrafları tahmin edebilir. Tahmin, Bayes denklemi kullanılarak gerçekleşir. İlgili Bayes denklemi Eşitlik 4.9'da verilmiştir.

$$P(W/A) = \frac{P(A/W)P(W)}{P(A)} \tag{4.9}$$

Hipotez cümlesi *W* ile temsil edilir, olası cümlelerle akustik uyumu en yüksek olasılığa sahip cümledir. *P (W | A),* elde edilen akustik verilere *(A)* göre veri tabanındaki cümlenin *(W)* akustik sıra ile uyumlu olma olasılığını temsil eder. Eşitliğin ikinci bölümünde Bayes kuralı uygulanmaktadır. *W* cümlesinin oluşma olasılığı *P (W)* 'dir ve

LM'ye göre hesaplanır. Akustik dizinin cümlede oluşma olasılığı *P (A | W)* 'dir. *P (A), W*'den bağımsız olduğu için kısaltılmıştır [64].

4.5 Ses Derlemi- Konuşma Veritabanı

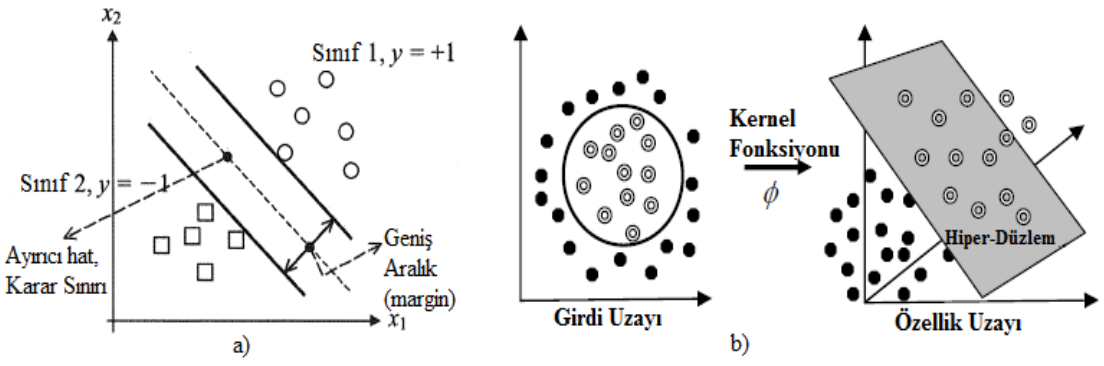
ASR uygulaması, bir bilgisayar veri tabanında depolanan yazılı veya sözlü dil metinlerinin bir koleksiyonu olan bir derleme ihtiyaç duyar. Derlemdeki yazılı metinler gazete, kitap veya dergilerden alınır. Sözlü derlem, konuşma dilinin transkriptlerini içerir.

5. KONUŞMA TANIMADA KULLANILAN YENİ NESİL YÖNTEMLER

Bilgisayar teknolojisinin gelişimi ve grafik kartlarının (GPU) hesaplama gücüne destek vermesi ile birlikte, Gauss karışım Modellerinin konuşma tanıma uygulamalarında kullanımı yerine Yapay Sinir Ağları, Destek Vektör Makinaları ve Derin Öğrenme kullanımına yer veren sistemlerin performansları incelenmiştir.

5.1. Destek Vektör Makinaları (SVM)

SVM, doğrusal sınıflandırma yaparken, iki sınıf arasında ayrım yapılacak en geniş ayırma aralığını bulur ve bu aralık arasından geçen sınır çizgisi sınıflandırmayı gerçekleştirir[65]. Bu en geniş aralığı oluşturma en iyileme problemidir. Bu sınıflandırıcıya en yakın özellik vektörlerine destek vektörler olarak adlandırılır[16]. Bu en geniş aralık çözümü, Destek Vektör Makinalarının gürültülü ortamlarda diğer doğrusal olmayan sınıflandırıcılar karşısında avantaj sağlar. SVM aynı zamanda doğrusal olarak ayrılamayan sınıfları da ayırt edebilir[65]. Doğrusal olarak sınıflandırılamayan problemlerde ise çekirdek (kernel) fonksiyonlar, örnek olarak Radyal Tabanlı Fonksiyonlar (Radial Basis Functions) kullanılarak verilerin boyutu artırılır. Böylece, yeni veri uzayında doğrusal olmayan problem doğrusal biçimde sınıflandırılabilir hale gelir. Şekil 5.1'de SVM sınıflandırma örnekleri gösterilmiştir.



Şekil 5.1 a) SVM ile doğrusal sınıflandırma [66]. b) Kernel fonksiyonu kullanımı[67].

5.2 Deep Neural Networks (DNN)

Derin öğrenme, YSA'nın geliştirilmiş mimarisidir. Desen ve özellikler hakkında bilgi edinmek için birden fazla gizli katman kullanılır. Örneğin videolar, resimler ve konuşma gibi birçok karmaşık sinyal modeli, birçok katman aracılığıyla başarıyla öğrenilir. Bu katmanlar, doğrusal olmayan işleme işlevlerine sahip düğümlere sahiptir. Ön eğitim adımları, milyonlarca düğüme sahip büyük ağların eğitilmesine izin verir.
ASR uygulamalarında, Derin Öğrenme Yaklaşımı GMM'in yerini almıştır. DNN'ler GMM tabanlı sistemlerden daha iyi çalışır[20]. (Bkz. Şekil 4.3). Birçok gizli katmana sahip DNN'lerin, çeşitli konuşma tanıma kriterlerinde GMM'lerden daha iyi performans gösterdiği gösterilmiştir. Verilerdeki desenleri tespit etmek için gizli katmanlar kullanılır. Daha fazla katman, daha karmaşık veriler üzerinde çalışmayı sağlar [27].

Doğrusal olmayan aktivasyon fonksiyonları ile DNN, rastgele doğrusal olmayan bir fonksiyonu modelleyebilir (girişlerden çıkışlara projeksiyon). Her gizli birimde, *j*, alttaki katmandan gelen toplam girdi, x_j, skaler duruma (y_j) eşlenir ve yukarıdaki katmana gönderilir.

$$x_j = b_j + \sum_i y_i w_{ij} \quad (5.1) \qquad y_j = logistic(x_j) = \frac{1}{1 + e^{-x_j}} \quad (5.2)$$

b_j, *j* birimi sapmasıdır, *i*, aşağıdaki katmandaki birimlerin bir indeksidir ve w_{ij}, aşağıdaki katmanda *i*'den *j*'ye olan bağın ağırlığıdır.

Çok sınıflı sınıflandırma için, bir sınıf olasılığı p_j elde etmek için "softmax" doğrusallığı kullanılır.

$$p_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)} \quad (5.3)$$

burada *k*, sınıfların [27] tamamında bir indekstir.

DNN ağlarının eğitim aşamasında, gerçek ve hedef çıktılar arasındaki uyumsuzluğun türevi geri yayılır. Başlangıç ağırlıkları küçük rasgele değerlere ayarlanabilir. Genel olarak, DNN'nin Derin Boltzmann Makinesi (DBM) veya DBN olarak önceden eğitilmesi, başlangıç için daha iyi bir yoldur, daha sonra kayıt örnekleri ince ayar yapmak için kullanılır [68].

Son yıllarda bilgisayar teknolojisinin gelişmesi ve eğitim süresinin kısalması Derin Öğrenmeye dayalı sistemlerin geliştirilmesini sağlamıştır. Gelişmiş bir çok katmanlı YSA olarak Derin Öğrenme algoritmaları, ASR dahil olmak üzere birçok sorunu çözmede performansı artırmıştır. Derin Öğrenme yaklaşımı GMM'nin ASR performansından daha iyi performans gösterdi. DNN'ler, akustik model eğitiminde yaygın olarak uygulanmıştır ve istatistiksel yöntemlerden daha iyi performans göstermiştir.

Derin Öğrenme Yaklaşımı, konuşma tanıma uygulamalarında Gauss Karışımı adımının yerini başarıyla almıştır [27]. DNN ve GMM, her bir ses segmentini temsil eden HMM için durum bilgisi sağlar. DNN, HMM'ye fonemler arasındaki farkları daha iyi temsil eden daha fazla durum bilgisi sağlar. Böylece, daha büyük ses verisi ve kelime tanıma başarısı daha yüksektir. DNN-HMM sisteminin yapısı Şekil 4.3'de gösterilmektedir.

Derin Öğrenme, YSA'nın bir uzantısıdır. Desenler ve özellikler hakkında bilgi edinmek için birden çok gizli katman kullanılır. Video, görüntü ve konuşma gibi birçok karmaşık sinyal modeli, birçok katman aracılığıyla başarıyla öğrenilir. Bu katmanlar, doğrusal olmayan işleme işlevlerine sahip düğümlere sahiptir. Derin öğrenme yöntemlerinden biri olan Recurent Neural Networks (RNN), konuşma tanıma uygulamalarında her iki özellik sınıflandırma adımını gerçekleştirir (Şekil 4.3).

5.2.1. Üretken Ön eğitim:

Üretim öncesi eğitimde, öznitelik dedektörleri başlangıçta sınıflar arasında ayrım yapmak için öznitelik dedektörleri tasarlamak yerine giriş verilerindeki yapıyı modellemek için tasarlanmıştır [27]. İlk olarak, öznitelik dedektörlerinin bir katmanı öznitelik verileriyle eğitilir, ardından eğitilen öznitelik algılayıcıları bir sonraki katmanın eğitimi için veri görevi görür. Çok sayıda öznitelik detektörü katmanı, bu üretken "ön eğitim" ile ayırt edici bir "ince ayar" aşaması için iyi bir şekilde hazırlanmıştır. Geri yayılım sırasında, eğitim öncesi bulunan ağırlıklar DNN tarafından biraz ayarlanır.

Özel bir Boltzman Makinesi olan Kısıtlı Boltzman Makinesi (RBM), gizli bir birim katmanından ve gizli - gizli veya görünür-görünür bağlantıları olmayan görünür bir birim katmanından oluşur. Gizli birimler ile görünür birimler arasındaki bağlantılar simetrik ve

yönsüzdür. Gizli ve görünür birimlerin değeri genellikle stokastik ikili birimlerdir. (Olasılığa bağlı olarak 1 veya 0) [69].

RBM yapısı Şekil 5.2'de gösterilmektedir. Model, bir enerji fonksiyonunda [70] bir dizi *(v, h)* değeri için aşağıdaki gibi tanımlanabilir:

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\sum_{i \epsilon \text{visible}} a_i v_i - \sum_{j \epsilon \text{hidden}} b_j h_j - \sum_{i,j} v_i h_j w_{ij} \qquad (5.4)$$

h_j ve v_i, sırasıyla görünür birim i ve gizli birim j'nin ikili durumlarıdır; w_{ij}, h_j ve v_i arasındaki ağırlıktır; b_j ve a_i, sırasıyla h_j ve v_i'nin öndeğerleridir. v ve h üzerindeki ortak dağılım şu şekilde verilmiştir:

$$P(\mathbf{v}, \mathbf{h}; \theta) = \frac{1}{Y} \exp\big(-E(\mathbf{v}, \mathbf{h}; \theta)\big) \qquad (5.5)$$

burada Y, şu şekilde verilen bir bölüm işlevidir:

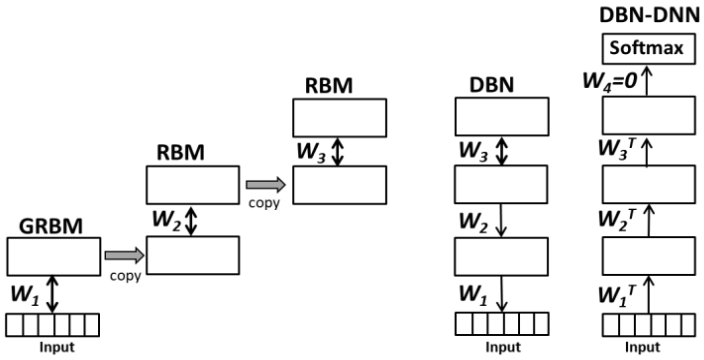
$$Y = \Sigma_v \Sigma_h e^{\{-E(v,h;\theta)\}} \qquad (5.6)$$

Kontrastif diverjans (CD) algoritmasını kullanarak maksimum olasılığın öğrenilmesi, RBM model parametrelerini tahmin etmek için eğitim verilerini kullanır θ [71].

ASR uygulamalarında DBN-DNN, özelliklere göre HMM'nin her durumu için posterior olasılık üretir. Kelimeler, heceler, telefonlar, telefon durumları, alt telefonlar vb. Gibi konuşma birimleri HMM veya HMM durumlarına göre modellenir [72].

5.2.2. Derin İnanç Ağları (DBN):

DBN, görüntü ve ses sınıflandırmasında birçok önemli başarı elde etmiştir. DBN'nin amacı, büyük miktarda etiketlenmemiş veriyi denetimsiz bir şekilde eğiterek tipik veri özelliklerini öğrenmektir. Yığınlanmış RBM'ler bir DBN [73] oluşturur.



Şekil 5.2 Derin İnanç Ağları [27]

Üç gizli katmana sahip bir DBN (Bkz. Şekil 5.2) oluşturmak ve bunu önceden eğitilmiş bir DBN-DNN'ye dönüştürmek için ilk olarak bir Gaussian-Bernoulli RBM (GRBM), gerçek değerli akustik katsayı çerçevelerini modellemek için eğitilir[27]. Bir RBM'yi eğitmek için, GRBM'nin ikili gizli birimlerinin durumları olan verilere ihtiyaç duyulur. Bu işlem tekrarlanarak istenildiği kadar gizli katman oluşturulur. Düşük seviyeli RBM'lerin yönsüz bağlantıları, yukarıdan aşağıya, yönlendirilmiş bağlantılarla değiştirilir. Böylece, RBM yığını, tek bir üretken modele dönüştürülür ve bir DBN oluşturur. Her HMM'nin olası her durumunu içeren bir "softmax" çıktı katmanı eklenir. Böylece, ön eğitimi yapılmış bir DBN-DNN oluşturulur. DBN-DNN daha sonra ayrıca eğitilir. Bu şekilde, giriş penceresinin HMM durumları tahmin edilir.

Konuşmanın özellikleri, MFCC'ler, bir GRBM'e verilerek Gauss gürültülü doğrusal değişkenler tarafından modellenir; RBM enerji işlevi şu şekilde verilir:

$$E(\boldsymbol{v},\boldsymbol{h}) = \sum_{i \epsilon \text{vis}} \frac{(v_i - a_i)^2}{2\sigma_i^2} - \sum_{j \epsilon \text{hid}} b_j h_j - \sum_{i,j} \frac{v_i}{\sigma_i} h_j w_{ij} \qquad (5.7)$$

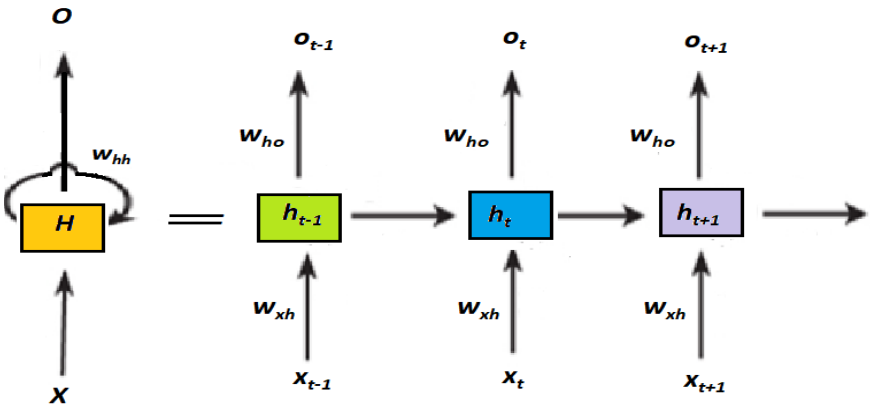
b_j ve a_i'nin sırasıyla h_j ve v_i'nin öndeğerleri, σ_i, görünür birim i için Gauss gürültü standart sapmasının değeridir[27].

5.2.3. Tekrarlayan Sinir Ağı (RNN)

Özyinelemeli yapısı nedeniyle, RNN, konuşma tanıma gibi zamanla değişen problemlerin çözümü için uygundur. RNN'ler, gizli bir tekrarlayan duruma sahip olarak geleneksel ileri besleme konseptini genişletir. Gizli yinelenen durumun aktivasyonu

öncekine bağlıdır. Böylece, geleneksel nöron ağlarından farklı olarak, konuşma tanıma için önemli bir değerlendirme olan süreçlerdeki zamanlama bilgileri kaydedilir [74].

RNN'ler, dizi tahmin problemleriyle çalışmak üzere tasarlanmıştır. Metin, konuşma verileri ve regresyon tahmin problemlerinin sınıflandırılması için kullanılır. RNN ve genişltilmesi Şekil 5.3'de gösterilmektedir. Bir N-kelimeden oluşan bir cümleyi temsil eden bir RNN düşündüğümüzde, bu ağ aynı zamanda her kelime için bir katmana sahip N-katmanlı Sinir Ağı olarak da adlandırılabilir.



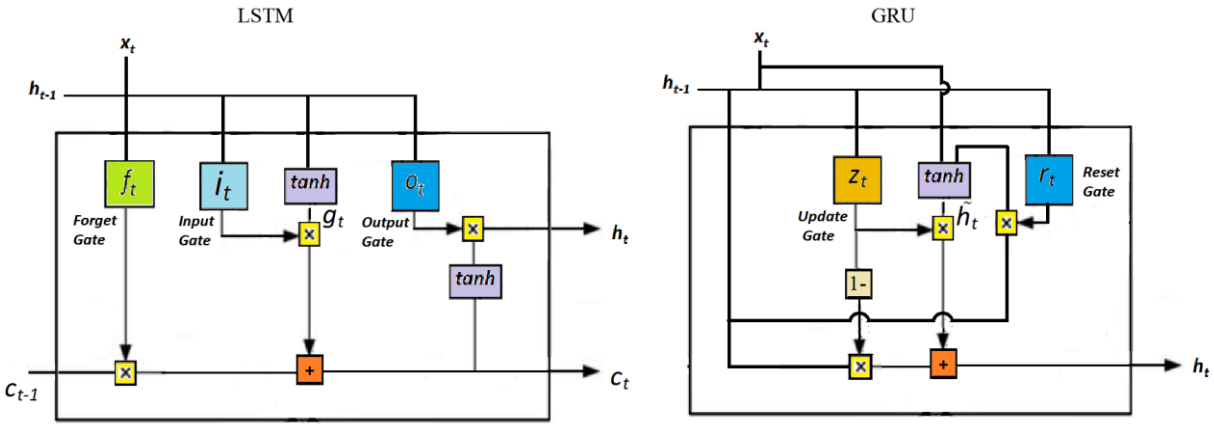
Şekil 5.3. Tekrarlayan Sinir Ağının Açılması

Bununla birlikte, RNN'lerin eğitimi, patlayan ve kaybolan gradyanlar nedeniyle karmaşık olabilir ve bu, uzun vadeli bağımlılıkları öğrenmeyi engelleyebilir. RNN'lerin temel fikri, çeşitli zaman adımlarında bilgi akışını daha iyi kontrol etmek için bir kapı mekanizmasının tanıtılmasıdır. Geçitli RNN'lerin yardımıyla, yok olan gradyan problemleri, gradyanlarla birden fazla zamansal adımın atlanabildiği etkili "kısayollar" oluşturarak hafifletilir. LSTM'ler, kapılı RNN'ler grubunda en popüler olanlardır. Makine öğrenimi görevlerinde, özellikle konuşma tanıma, son teknoloji performans genellikle LSTM'ler tarafından elde edilir. LSTM mimarisinde giriş, çıkış ve unutma kapıları bellek hücrelerini kontrol eder. Etkililiklerine rağmen, böylesine karmaşık bir kapı mekanizması, aşırı karmaşık bir modelle sonuçlanabilir. Diğer bir konu da hesaplama verimliliğinin RNN'ler için çok önemli olması ve alternatif mimarilerin geliştirilmeye çalışılmış olmasıdır [31].

5.2.4. Uzun Kısa Süreli Bellek (LSTM) Sinir Ağları

En çok kullanılan RNN türlerinden biri Uzun Kısa Süreli Bellek (LSTM) Sinir Ağıdır. Uzun vadeli bağımlılık sorununu çözmek için tasarlanmıştır [75]. Zaman içinde bağımlılıkları daha iyi modellemek ve Kaybolan Gradyan Problemini çözmek için kullanılır.

Tekrarlayan hücrede, Şekil 5.4'te gösterildiği gibi, yalnızca bir sinir ağı kapısı değil, etkileşimli üç kapı vardır. Giriş, unutma ve çıkış portları, LSTM'nin davranışını tanımlar. Kapıların açık veya kapalı olmasına bağlı olarak bilgiler hücrede saklanabilir veya okunabilir. Önceki hücre değerleri, unutma geçidi ile çarpılır. Böylece sıfırlama işlevi gerçekleştirilir. Kapılar aldıkları sinyalin gücüne ve ağırlığına göre üzerlerindeki bilgileri geçirir veya bloke eder. LSTM'deki ağırlıklar özyinelemeli sinir ağı öğrenmesine göre ayarlanır [76].



Şekil 5.4. LSTM ve GRU Hücre Yapıları

LSTM katmanına ilişkin vektör hesaplaması aşağıda verilmiştir. [32]:

$$i_t = \sigma(w_{xi}x_t + w_{hi}h_{t-1} + b_i) \quad (5.8)$$

$$f_t = \sigma(w_{xf}x_t + w_{hf}h_{t-1} + b_f) \quad (5.9)$$

$$g_t = tanh(w_{xg}x_t + w_{hg}h_{t-1} + b_g) \quad (5.10)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (5.11)$$

$$o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + b_o) \quad (5.12)$$

$$h_t = o_t \odot \tanh(c_t) \quad (5.13)$$

Genel olarak RNN'ler ve özellikle LSTM'ler, genellikle doğal dil işleme adı verilen kelime ve paragraf dizileriyle çalışmakta başarılı olmuştur.

5.2.5. Geçitli Tekrarlayan Birimler (GRU):

Son zamanlarda, yalnızca iki çarpımsal kapıya dayanan Geçitli Tekrarlayan Birim (GRU) olarak adlandırılan yeni bir modelin tasarımı, LSTM'leri basitleştirmek için dikkate değer bir girişim olmuştur. Unutma geçidini ve giriş kapısını tek bir güncelleme geçidinde birleştirir. GRU, ara bilgileri depolamak için ayrı bir "hücreye" sahip değildir ve Şekil 5.4'te gösterildiği gibi bellek akışını kontrol eden bir sıfırlama geçidi ve bir güncelleme geçidine sahiptir. Bu nedenle GRU, LSTM'lerden biraz daha az parametreye sahiptir. Basitliği nedeniyle, GRU'lar hafızayı veya hesaplama süresinden kazandırır ve birçok sıralı öğrenme görevinde yaygın olarak kullanılmaktadır [8,9]. Özellikle, standart GRU mimarisi aşağıdaki eşitliklerle tanımlanır, burada r_t ve z_t sırasıyla sıfırlama ve güncelleme kapılarının vektörleridir, h_t ise mevcut zaman çerçevesi t için durum vektörünü temsil eder.

$$z_t = \sigma(w_z x_t + u_z h_{t-1} + b_z) \quad (5.14)$$

$$r_t = \sigma(w_r x_t + u_r h_{t-1} + b_r) \quad (5.15)$$

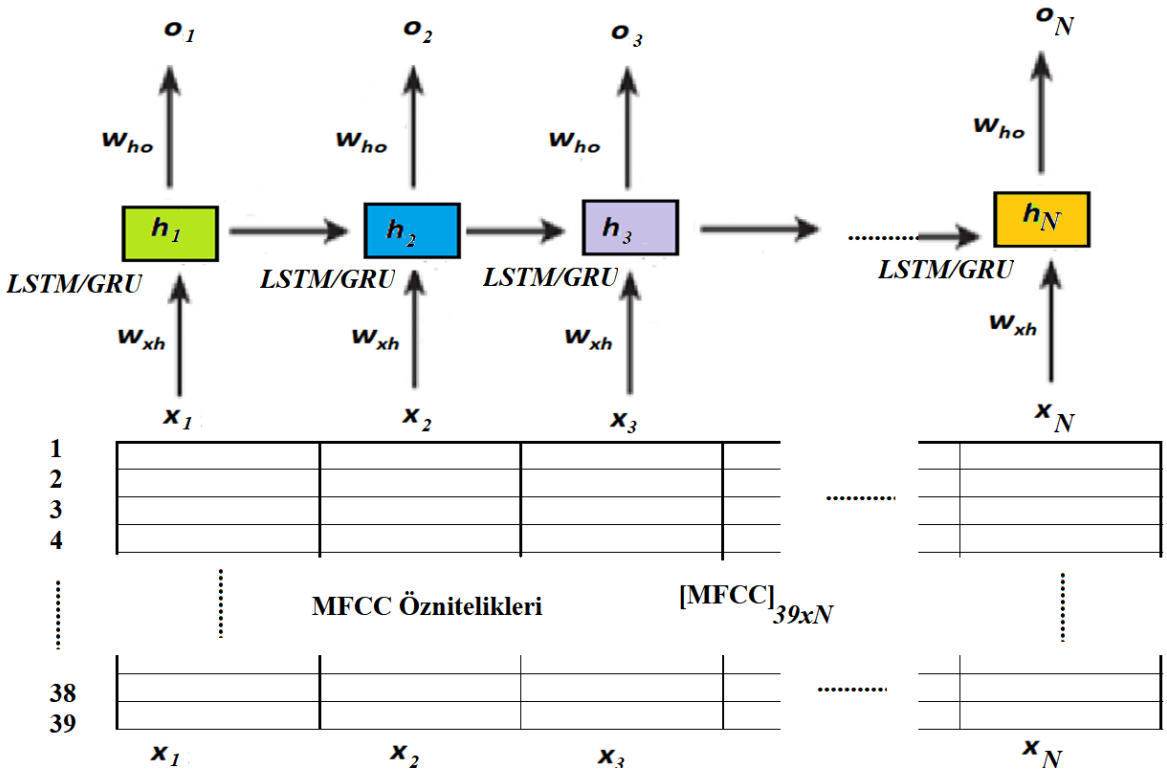
$$\tilde{h}_t = tanh(w_h x_t + u_h(h_{t-1} \odot r_t) + b_h) \quad (5.16)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t \quad (5.17)$$

"ʘ" işlevi, Eleman bazlı çarpımları ifade eder. Her iki kapının aktivasyon fonksiyonları lojistik sigmoid σ'dır. Bu şekilde, z_t ve r_t, 0 ve 1 arasında değişen değerler alacak şekilde sınırlandırılır. Aday durum $\tilde{h_t}$, hiperbolik bir tanjantla işlenir. Mevcut girdi vektörü x_t (örneğin, konuşma özellikleri) ağı besler ve matrisler, w_z, w_r, w_h (ileri besleme bağlantıları) ve u_z, u_r, u_h (tekrarlayan ağırlıklar) modelin parametrelerini temsil eder. Eğitilebilir öndeğer vektörleri, b_z, b_r ve b_h, dir[31].

5.2.6 Konuşma Tanımada MFCC Özniteliklerinin LSTM ve GRU'ya Uygulanması

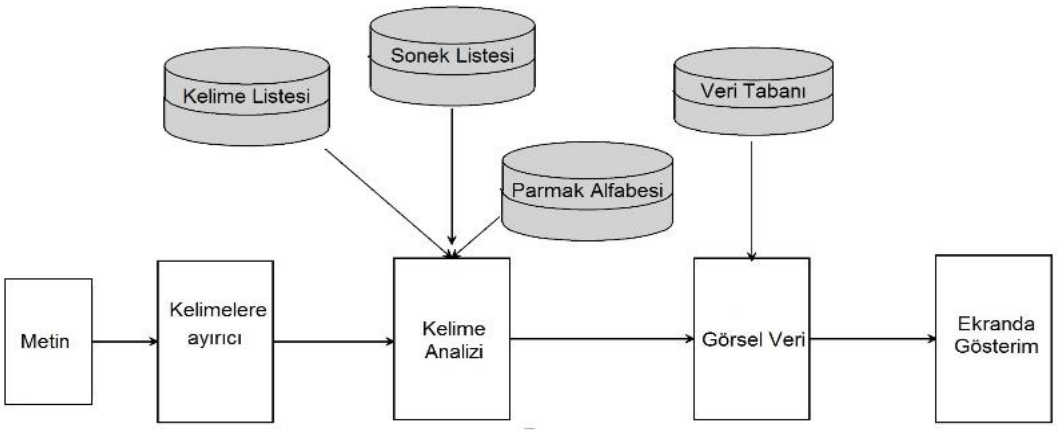
Önerilen LSTM ve GRU tabanlı Konuşma Tanıma Sisteminde, MFCC öznitelik dizileri sinir ağı girilerine uygulanır. MFCC öznitelikleri, sinyalin 10 ms'lik bir örtüşme ile 25 ms'lik çerçevelere bölünmesiyle hesaplanır. Bu çerçeveler, 12 MFCC parametresi ve çerçeve enerjisinden oluşan 13 parametre ile temsil edilir. Parametrelerin birinci ve ikinci türevleri de dahil edilerek (13 statik + türev + ikinci türev) her t anında 39 adet MFCC katsayısı hesaplanır. 39xN boyutlu MFCC matrisi Her t zaman adımında $X = \{x_1, \dots, x_t, \dots, x_N\}$ LSTM/GRU girişlerine uygulanır.



Şekil 5.5. MFCC Özniteliklerinin LSTM ve GRU'ya giriş olarak uygulanması

5.2.7 Yazılı Metinden Türk İşaret Diline Dönüştürme

Şekil 5.6'da da görüldüğü gibi, yapılan çalışmada kullanılan yöntemde metin kelimelere ayrılarak incelenir. Daha sonra kelimeler analiz edilir ve doğru görsel veri ile bulunur. Son aşamada da görsel veri ekranda gösterilir.



Şekil 5.6. Yazılı Metinden Türk İşaret Diline Dönüştürme[47]

Cümlelerin kelimelere ayrılarak işlenmesinde Türkçe'nin sondan eklemeli bir dil olmasının önemi büyüktür. Konuşma tanıma adımında elde edilen metindeki kelimeler, aralarındaki boşluktan faydalanarak ayrılırlar. Türkçe kelimelerin sonundaki eklerin çoğu TİD'de karşılık bulmamaktadır. Bunlar göz önünde bulundurulduğunda cümleyi kelimeler bazında ele almak, kelimeleri iyi analiz etmek gereklidir.

Kelime analizi yapılırken kelimenin öncelikle direkt olarak veri tabanındaki kelimelerden bir tanesi ile eşleşip eşleşmediği kontrol edilir. Direkt eşleşmesi beklenen kelimeler herhangi bir ek almadan saklandıkları için veri tabanında "kök" klasörü altında bulunurlar. Direkt eşleşen bu kelimeler sistemde "kök" olarak etiketlenirler. TİD'de kelime sayısı oldukça düşüktür ve Türkçe'deki her kelimenin bir TİD karşılığı bulunmamaktadır Veri tabanında olmayan, eşleştirilemeyen ve işlenemeyen kelimeler için kelime harf bileşenlerine ayrılıp işaret dili parmak alfabesi harfleri ile ifade edilecektir[47].

Eğer kelime kök ya da özel isim değilse, kelimenin işlenmesi faydalı olacaktır. İşleme sırasında kelimenin sonek alıp almadığına, sonek varsa bu sonekin TİD için anlamlı olup olmadığına bakılması gerekir. Kelimenin soneklerden ayrılıp kökünün tespit edilmesi, veri tabanındaki kelimelerle karşılaştırıp eşleştirme yapabilmek açısından önemlidir. Veri

tabanındaki kelimeler ile de direkt eşleşme sağlamadığı için kök olarak nitelendirilemez. Bu durumda sistem bu kelimeyi işlenecek olarak nitelendirir ve bu şekilde etiketler.

Tablo 5.1. Kelime, Kök ve İşlemler

Kelimeler	**Kök**	**İşlenecek**
onlar	onlar	
yarın	yarın	
şehir	şehir	
dışına		dışına
çıkıyorlar		çıkıyorlar

Sistem işlenmesi gereken bir kelimenin öncelikle son ekini kontrol eder ve kelimeden ayırır. Kelimenin ek ayrıldıktan sonraki hali gövde olarak adlandırılır. Soneki tespit etmek için sistem kelimeyi sondan incelemeye başlar. Tespit edilen sonekin anlamlı olup olmadığını anlamak önemlidir. Veri tabanında ayrı bir klasör altında anlamlı olabilecek sonekler yer alır. Anlamlı olmasından kasıt, bahsi geçen sonekin TİD tarafında bir anlamsal karşılığının olmasıdır. Eğer tespit edilen sonek anlamlıysa, kelimenin kökü tespit edildikten sonra sonek kökün sonuna sanki farklı bir kelimeymiş gibi eklenir. Örneğin, "çalışıyorum" sözcüğünde "-yorum" ekinin TİD tarafındaki anlamsal karşılığı birinci tekil şahıs anlamı katıyor olmasıdır. Bu durumda bu kelimenin TİD karşılığı aslında "çalışmak + ben" ya da "ben + çalışmak" şeklinde olacaktır. Bu kelime soneki bulunmak üzere incelenmeye başlandığında son harfinden başlanarak mümkün olan bütün olasılıklar ek olabilecek şekilde değerlendirilir. Tüm olasılıklar ayrı bir dosya altında kayıtlı bulunan, anlamlı sonekler ile karşılaştırılır. Bu soneklerden bir veya daha fazlası ile eşleşme yapılabilir. Fakat bu eşleşmelerden en çok harf ile olanı dikkate alınacaktır. Eğer "çalışıyorum" kelimesinin sonek analizinde "-yorum" ekinin haricinde "-ıyorum" ya da daha fazla harfli bir ek de eşleşme sağlasaydı "-yorum" yerine dikkate alınacaktı. Fakat bu kelimede en yüksek harfli eşleşme, Tablo 5.2'de gösterildiği gibi "-yorum" ekinde sağlandığı için "-yorum" sonek olarak alınmıştır[47].

Tablo 5.2. Kelime ve Sonek analizi

İşlenecek Kelime	Sonek Analizi
okuyorum	m
	um
	rum
	orum
	yorum
	uyorum
	kuyorum
	okuyorum

Analizde karşılaştırılan diğer olasılıklardan da sonek olanlar çıkabilir. Fakat önemli olan ilgili sonekin TİD tarafında anlamlı olarak nitelendirilmiş olmasıdır. Anlamlı olabilecek sonekler ayrı bir dosya altında tutulmaktadır. Böylece anlamsız olan ekler eşleşme sağlamayarak elenecektir.

Kelimenin soneki olup olmadığı kontrol edildikten sonra kelime sonekinden ayrılmış olarak bir sonraki aşamaya geçer. Bu aşamada kelime kök halinde bulunabilir. Eğer ekten ayrıldıktan sonra kelime kök haline gelirse bu bir sonraki aşama için en iyi durum olarak nitelendirilebilir. Çünkü kök halindeki bir kelimenin veri tabanında olup olmadığını tespit etmek daha kolaydır. Fakat her zaman sonek ayrıldıktan sonra kelime kök halinde olmayabilir. Veya kelimenin soneki anlamlı değilse sonek tespit edilememiş ve kelime cümlede bulunduğu ekli hali ile bulunuyor olabilir. Bu durumda kelimenin veri tabanında olup olmadığını tespit etmek için kelimenin kökünün tespit edilmesi gerekir.

Tasarlanan eşleştirme algoritması ilgili kelimeyi veri tabanında kök olarak kayıtlı olan tüm kelimelerle karşılaştırır. Her karşılaştırma sonunda bir uzaklık değeri üretilir. Bu uzaklık, ilgili kelimenin veri tabanındaki kelime ile olan uzaklığını göstermektedir. Bu uzaklık değeri gövde halindeki kelimeden kök halinin çekilebilmesi için önemlidir. Yapılan çalışmada bu uzaklık değerinin hesaplanması için kullanılan uzaklık formülü (Eşitlik 5.18) dinamik programlama (DP) algoritmasında kullanılan uzaklık formülünden esinlenilerek türetilmiştir[47].

$$u = \frac{2f+e}{2a+2f+e} \tag{5.18}$$

Tablo 5.3. Uzaklık Parametreleri ve Açıklamaları

Uzaklık Parametreleri	Açıklama
u	Uzaklık
f	İki kelime arasındaki farklı harf sayısı
e	İşlenen kelimede karşılaştırılan kelimeye göre fazladan kaç harf bulunduğunu gösterir
a	İki kelime arasındaki sıralı olarak eşleşen harf sayısını gösterir

Eşitlik 5.18'de gösterildiği gibi, iki kelime arasındaki uzaklık (u) kelimelerin harf bazında karşılaştırılmasıyla elde edilir. Bu karşılaştırmada önce karşılaştırılacak kelimenin veri tabanındaki kelimeden boyut olarak büyük olup olmadığı kontrol edilir. Eğer karşılaştırma yapılacak kelimenin boyutu büyük ya da eşit ise uzaklık formülü uygulanabilir. Eğer kelime veri tabanındaki kelimeden büyük ise, iki kelime arasındaki harf sayısı farklılığı formülde "e" ile gösterilir. İki kelime arasında eşleşen harf sayısı (a) ve farklı olan harf sayısının (f) tespit edilmesi önemlidir. Farklı olan harfler ve fazladan bulunan harfler (e) kelimenin farklılığını belirleyecektir. Fakat kelimeler arasındaki farklı harflerin bu uzaklıktaki etkisi, fazladan bulunan harflere göre daha fazladır. Bu da formülde katsayı kullanılarak belirtilmiştir. Bu uzaklık değeri kelimedeki harf sayısına göre normalleştirilir. Sonuçta ortaya çıkan uzaklık değeri 0 ile 1 arasında değişecektir. Tamamen eşleşen bir kelimenin uzaklığı 0 çıkarken, tamamen farklı olan bir kelimenin uzaklığının 1 olması beklenmektedir. Sistem veri tabanındaki kelimeler ile ilgili kelimenin uzaklıklarını karşılaştırır ve en düşük uzaklık değerine sahip olan veri tabanındaki kelime ile eşleştirme yapılır.

Eğer eşleşme olmayacaksa sistem ilgili kelimenin özel isimlerde olduğu gibi harfbazında parmakla heceler. Bu durum söz konusu ise, öncesinde kullanıcıya kelimenin veri tabanında bulunamadığı ve parmakla heceleme yapılacağına dair bilgilendirme yapılır. Eğer eşleşme yapılacaksa, bulunan kök ve eğer varsa sonek birleştirilir[47].

6. DENEYLER VE ANALİZLER

Bu çalışmada, Türkçe konuşma tanıyıcı ve işaret diline çevirici geliştirilmiştir. Önerilen sistem kelime kelimelerini tanır ve kelime dağarcığı dışında kalan kelimelerin fonetik bileşenlerini bulur.

Sisteme alt kelime tabanlı LM uygulanmıştır. Türkçe'nin her bir ses birimi modelde bir alt sözcük olarak modellenmiştir. Alt kelime tabanlı LM, kelime dağarcığındaki aşırı büyümeyi önlemek için sondan eklemeli diller için yaygın olarak kullanılmaktadır. Önerilen LSTM ve GRU tabanlı ASR sisteminin performansı, geleneksel tanıma yöntemi olan GMM tabanlı HMM ile karşılaştırılmıştır. Performans ölçümleriyle ilgili olarak, Türk dilinin tanınma oranı yazarların önceki çalışmaları ve kullanılan derleme yer veren çalışmalara göre iyileştirilmiştir.

Deneysel çalışmada kullanılan derlem, "Türkçe Mikrofon Konuşma v1.0" 120 konuşmacının seslendirdiği 40 cümleden ve Türk alfabesinde 29 harfe karşılık gelen 38 ses biriminden oluşmaktadır [18].

Deneylerde GMM, DBN, LSTM ve GRU tabanlı ASR sistemleri eğitilmiş ve test edilmiştir. Analizler Fonem Hata Oranı (PER) ve Kelime Hata Oranı (WER) kriterlerine göre, konuşmacıdan bağımsız durumlar için yapılmıştır.

WER ve PER oranları aşağıdaki aşağıdaki eşitliklerle hesaplanır:

$$WER = \frac{D_w + S_w + I_w}{N_w} * 100 \tag{6.1}$$

D_w*:* Referans cümleyi elde etmek için yapılması gereken kelime silme işlemi sayısı,
S_w*:* Referans cümleyi elde etmek için yapılması gereken kelime değiştirme işlemi sayısı,
I_w*:* Referans cümleyi elde etmek için yapılması gereken kelime ekleme işlemi sayısı,
N_w*:* Referans cümledeki kelime sayısıdır.

$$PER = \frac{D_p + S_p + I_p}{N_p} * 100 \quad (6.2)$$

D_p: Referans kelimeyi elde etmek için yapılması gereken fonem silme işlemi sayısı,

S_p: Referans kelimeyi elde etmek için yapılması gereken fonem değiştirme işlemi sayısı,

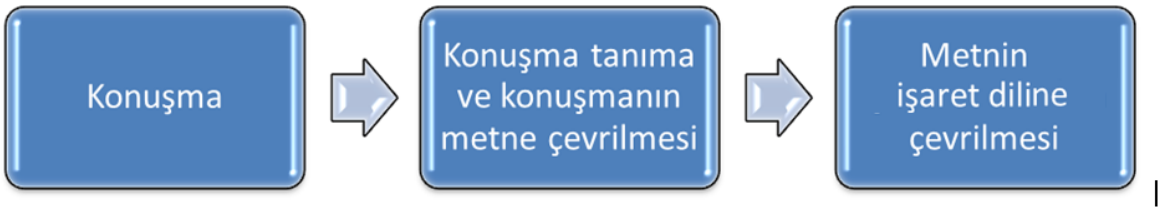
I_p: Referans kelimeyi elde etmek için yapılması gereken fonem ekleme işlemi sayısı,

N_p: Referans kelimedeki fonem (harf) sayısıdır.

Önerilen çalışmada, ASR algoritmaları için açık kaynak Kaldi kodları kullanılmıştır. Eğitim ve doğrulama için 120 kişiye ait ses kayıtları kullanılmıştır.

6.1. Sistem Mimarisi

Bu tezde gerçekleştirmek istediğimiz sistemin ana blok yapısı aşağıdaki gibidir.



Şekil 6.1: Sistem Ana blok yapısı

Şekil 6.2: Konuşma Tanıma ve Konuşmanın Metne Çevrilmesi

Yukarıdaki şema bu tezde kullanılacak olan Konuşma Tanıma adımını anlatmaktadır. Sesli ifadelerin bir mikrofon aracılığıyla örneksel sinyallere dönüştürülmesi, sayısallaştırılması, sayısallaştırılan bu sinyallerin gerekirse filtrelenmesi, etiketlenmesi

(örneğin sesler, fonemler, sözcükler olarak) ve tanıma işlemlerine taban oluşturacak parametrik yapılar ya da yalın modellerle ifade edilen biçimlere dönüştürülmesi gerekmektedir

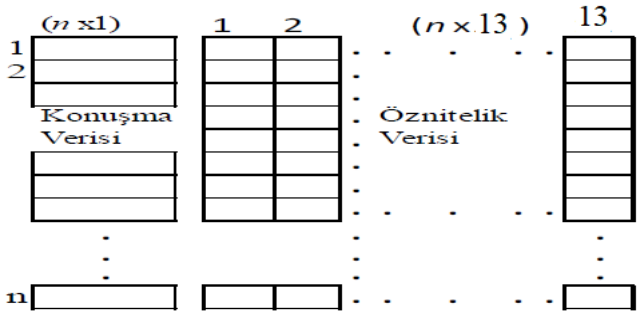
Metnin İşaret Dili videolarına Çevrilmesi aşamasında Programdan elde edilen çıkış kodlarına göre de hangi kelimenin konuşulduğu anlaşılmaya çalışılır. Konuşma tanıma adımı ile metne çevrilen ses bilgileri bu adımda işaret dilini temsil eden videolara çevrilir.

Şekil 6.3: İşaret Dili Videosu

6.2. Öznitelik Seçimi

Öznitelik matrisleri, yaygın olarak kullanılan MFCC ve LPCC yöntemleri kullanılarak oluşturulur. Söz konusu öznitelik belirleme yöntemlerinin performansları, Konuşma kaydındaki fonem bileşenlerini saptamak için tasarlanan, 8 katmanlı multiclass SVM Sınıf Uygulaması ile ölçülmüştür.

Kullanılan veri setine ait özniteliklerin boyutlandırması Şekil 6.4'de gösterilmektedir. Öznitelik matrisi n satırdan oluşup, satır sayısı, ses parçasının uzunluğuna ve öznitelik çıkarımı adımında seçilen pencere boyutuna göre değişkenlik gösterir. Kolonlar ise öznitelik katsayılarını göstermektedir.



Şekil 6.4 Öznitelik matrisi

Ses kayıtları 3 farklı kişiden alınmış olup, 16 bit ve 16kHz olarak kaydedilmiştir. Bu ses kayıtlarındaki hece ve fonemler, ses düzenleme yazılımı yardımı ile kesilip, birleştirilmiştir. Birleştirilen kayıtlar, her bir foneme ilişkin ".wav" uzantılı dosya olarak hazırlanıp, saklanmıştır.

28 adet harf/fonem ve sessizlik 29 sınıfta temsil edildiğinde, sınıfların tek sınıflandırıcıda ayrıştırılmasındaki zor olmaktadır. Bu sebeple, önce fonem tipi saptanmakta sonra fonem tipine ilişkin sınıflandırıcı kullanılarak fonem belirlenmektedir. Sınıflandırmaları sağlayan SVM katmanları ile ilgili bilgiler Tablo 6.1'de verilmektedir.

Tablo 6.1. Fonem Sınıflandırıcılar

Fonem Türü Sınıflandırıcılar					**Fonem Sınıflandırıcılar**		
SVM_1	**SVM_2**	**SVM_3**	**SVM_4**	**SVM_5**	**SVM_6**	**SVM_7**	**SVM_8**
Sesli Kısım, Sessiz Kısım	Ünlü, Ünsüz	Ünlü, Ünsüz_1	Ünlü, Ünsüz_2	Ünsüz_1, Ünsüz_2	a, e, ı, i, o, ö, u, ü	b, c, d, g, j, l, m, n, r, v, y, z	ç, f, h, k, p, s, ş, t

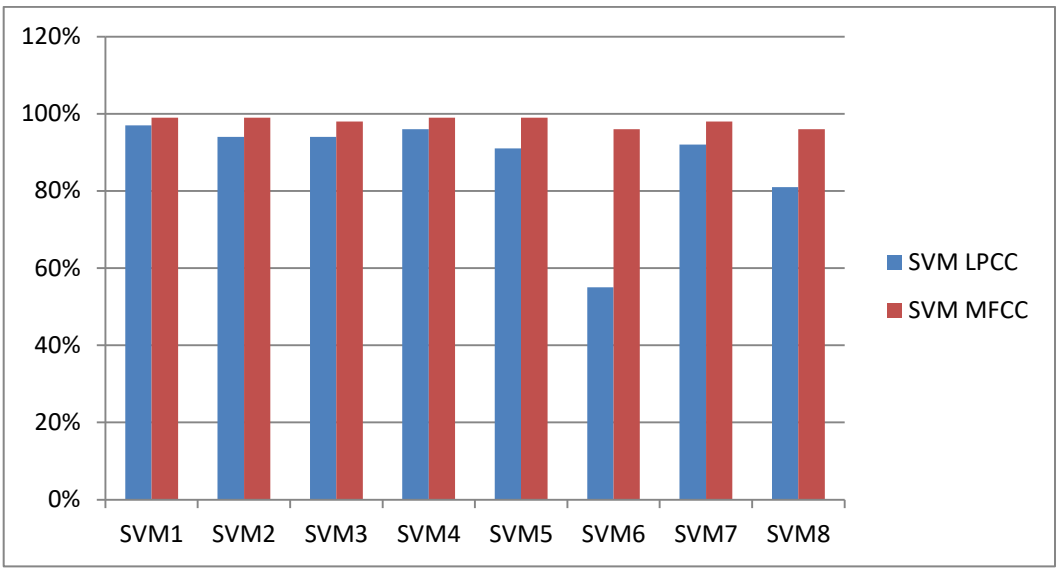
Aşağıda, sınıflandırma algoritmasının sözde kod (pseudocode) ile yazımı bulunmaktadır.

```
Input: oz; // öznitelik
u←length(oz); // öznitelik uzunluğu
        for i←1 to (u) do
          s←SVM1(öznitelik(i));
                while (s=sesli) do
                        ft1← SVM2(öznitelik(i));
                        ft2← SVM3(öznitelik(i));
                        ft3← SVM4(öznitelik(i));
                        ft4← SVM5(öznitelik(i));
                                if(ft1=ünlü and ft2 = ünlü and ft3=ünlü) then
                                        tanınan fonem(i) ← SVM6(öznitelik(i));
                                end if
                                if(ft1=ünsüz and ft2 = ünsüz1 and ft4=ünsüz1)
                                        tanınan fonem(i) ← SVM7(öznitelik(i));
                                end if
                                if(ft1= ünsüz and ft3 = ünsüz2 and ft4=ünsüz2)
                                        tanınan fonem(i) ←SVM8(öznitelik(i))
                                end if
                end while
        end for
```

SVM sınıflandırıcıların LPCC ve MFCC öznitelikleri ile sınıflandırma başarıları değerlendirilmiştir. Sınıflandırma Başarısını ifade eden oran Eşitlik 6.3'de verilmiştir. Test sonuçlarına göre, hangi özniteliğin kullanılacağına karar verebilmek için SVM katmanlarındaki fonem sınıflandırma başarıları incelenmiştir.

$$Başarı\ oranı = \frac{\text{Doğru olarak sınıflandırılan örnek sayısı}}{Toplam\ örnek\ Sayısı} \qquad (6.3)$$

Uygulanan yöntemlerin başarı karşılaştırılması Şekil 6.5'de gösterilmiştir. LPCC kullanıldığında sınıflandırma başarısı %88 ölçülüp, MFCC kullanıldığında %98 ölçülmüştür. Buna göre, SVM-MFCC sistemi en başarılı sistem olarak gözlenmiştir. Bu nedenle ilerleyen çalışma ve geliştirmelerde MFCC öznitelikleri kullanılmıştır.



Şekil 6.5 Sınıflandırma Başarı Grafikleri

6.3 Konuşma Tanımada Ses Kitaplığı ve Derlemin Önemi

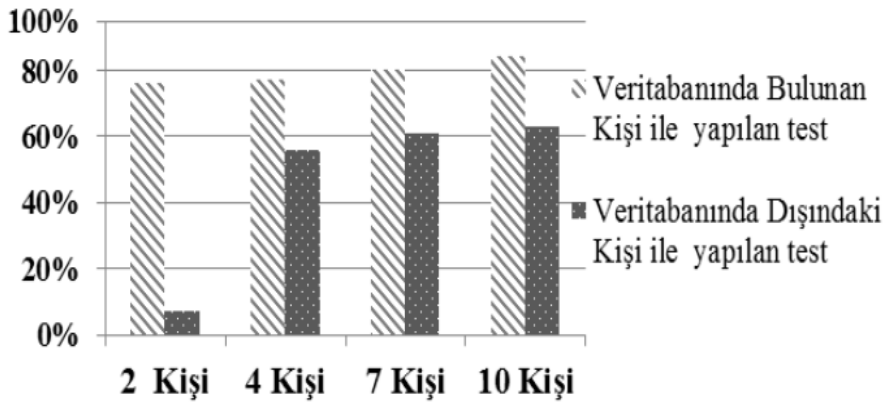
Bir sonraki çalışmada[18] önerdiğimiz sınıflandırma katmanlarının bilgileri Tablo 6.2'de verilmiştir. Görüldüğü gibi katman sayıları artırılmış ve katman başına düşen ayırt edilecek sınıf sayısı azaltılmıştır. Bu sayede fonem ayırt etme gücünde artış görülmesi planlanmıştır. Sistemde 10 kişiye ait fonem parçalarından bir eğitim seti oluşturulmuştur. Sistem eğitildikten sonra, bir test veri seti uygulanarak ürettiği fonem çıktılarının doğru sınıflandırılma başarısı hesaplanmıştır. Eğitim setindeki kişi çeşitliliği ve veri miktarı attıkça, sınıflandırma başarısının arttığı gözlemlenmiştir. Şekil 6.6'deki grafikte sınıflandırma başarısının kişi sayısına göre değişimi gösterilmektedir.

Sistemin kelime tanıma performansı, Denklem (6.4)'de verilen Kelime Doğruluk Oranı "Word Accuracy (WAR)"nölçütüne göre hesaplanmıştır. WAR hesaplamasına göre, sınıflandırma sonucu, elde edilen fonemler ile sistem veri tabanındaki kelime kalıpları karşılaştırılmış ve kelime tanıma başarısı belirlenmiştir.

$$WAR = \frac{Algılanan\ Doğru\ Kelime\ Sayısı}{Tüm\ Kelimeler} \tag{6.4}$$

Tablo 6.2. SVM Fonem Sınıflayıcılar.

	Öznitelik Verisi	
SVM1	**SL***	**SS***
SVM2	**Ünlü/Ünsüz**	
SVM3	**(a,e)/(ı,i)/(o,ö)/(u,ü)**	
SVM4	a/e	
SVM5	ı/i	
SVM6	o/ö	
SVM7	u/ü	
SVM8	**(b,d,g,p,t,k)/(c,ç,f,h,j,s,ş,v,y,z)/(m,n)/r/l**	
SVM9	**(b,p)/(g,k)/(d,t)**	
SVM10	b/p	
SVM11	g/k	
SVM12	d/t	
SVM13	**(c,ç) / (f,v) / (j,ş) / (z,s) / (h,y)**	
SVM14	c/ç	
SVM15	f/v	
SVM16	j/ş	
SVM17	z/s	
SVM18	h/y	
SVM19	m/n	



Şekil 6.6 Veritabanında uygulanan çeşitlilik sonucu performans

Önerilen sistemde, veri setindeki“fonem”ler kullanılarak sınıflandırma aşaması uygulandı ve sınıflandırma başarısı analiz edildi. Sistemin eğitim setindeki kişi sayısı 2’den 10’a artarken, veri içinde bilgileri bulunan kişi için fonem tanıma başarısı %75’ten %85’e yükselmiştir. Aynı sayıda kişiler için, eğitimde olmayan kişi için tanıma başarısının %8’den %63’e yükseldiği görülmüştür. Bu çalışmadan elde edilen sonuçlara göre, kullanılacak konuşma derlemindeki verinin, kişi çeşitliliğine sahip olması, net ve kaliteli ses kayıtlarını barındırması gerektiği anlaşılmıştır. Ayrıca kelime ve fonem modellerinin performansa katkısı olabilmesi ses birimlerin sınırlarının hassasiyetle tespit edilip ayırılması gerekmektedir. Bu nedenle, “Türk Mikrofon Konuşması v1.0” adlı konuşma derlemi çalışmaların devamında kullanılmıştır.

6.4. Ses Kitaplığı

Çalışmamızda kullanılan “Türk Mikrofon Konuşması v1.0” adlı ses derlemi, Orta Doğu Teknik Üniversitesi (ODTÜ) Elektrik-Elektronik Mühendisliği Bölümünden 193 konuşmacıdan (104 erkek, 89 kadın) toplanmıştır. 40 farklı cümle ile metinlerin her konuşmacı tarafından okunması ve toplam 2482 cümle kaydedilmesinden oluşur [21]. TIMIT derlemini oluşturmak için kullanılan yöntem budur. Her cümle konuşmacılar tarafından bir kez söylenir. "Orta Doğu Teknik Üniversitesi Türk Mikrofonu Konuşma v.1.0" adlı derlem, 2005 yılında Linguistic Data Consortium (LDC) tarafından kabul edilmiştir[21,77-79]. İlgili derlemi kullanarak analizlerde eğitim ve doğrula için 120 kişiye ait ses kayıtları kullanılmaktadır.

Türk dilinde Ü, Ö, İ gibi Latin olmayan bazı ünlüler ve Ç, Ş, Ğ gibi Latince olmayan ünsüzler bulunmaktadır. Bu özel harfler aşağıdaki gibi sembolize edilmiştir (Tablo 6.3'e bakınız) ve bu özel harfleri Kaldi konuşma tanıma araç kutusuna eklemek mümkün olmuştur.

Tablo 6.3. Kaldi'deki özel Türk harfleri

Özel Türkçe harfler	Kaldi'deki sembolleri
Ç	CH
Ğ	GH
İ	IY
Ö	OE
Ş	SH
Ü	UE

Türkçe konuşma tanıma konusunda "Türkçe Mikrofon Konuşma Derlemi" kullanılarak çalışmalar yapılmıştır. Bunlar [21], [22], [23] ve [15]'tir. İlk çalışmada, en iyi telefon tanıma oranı% 70,8'dir (PER =% 29,2). İkinci çalışmada kelime tanıma oranı% 78,54'tür (WER =% 21,46). Üçüncü çalışmada kelime tanıma oranı% 67,12'dir (WER =% 32,88). Dördüncü çalışmada, sınıflandırma için Ortak Vektör Yaklaşımı kullanılmıştır. Sesli fonem tanıma oranı% 48,75 ve ünsüz fonem tanıma oranı % 53,02'dir.

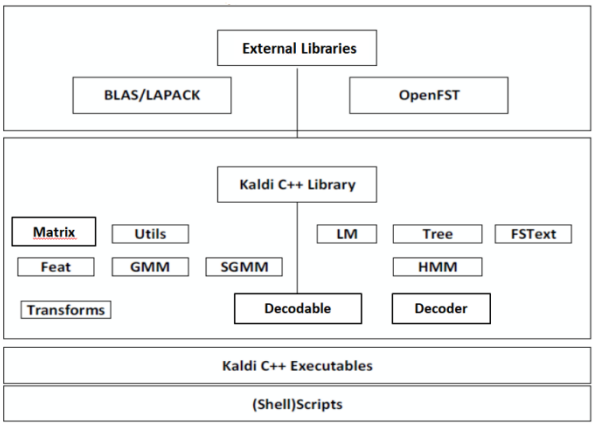
6.5. Derin Öğrenme Tekniklerinin Konuşma Tanımada Kullanılması

SVM, bazı ünsüz fonemlerin (b,d,g,p,t,k gibi) sınırları kesin olarak ayırt edilemediği için ve eğitim setine de sınırları tam olarak ayırt edilmiş veri konulamadığı için ünsüz ayırt etmede başarılı değildir. Bu sebeple büyük veri setleri kullanıldığında performansında düşme görülmüştür. Fonem ayırt etme ve dolayısıyla kelime tanıma başarısı düşmektedir. Ayrıca SVM'ler konuşma işareti gibi zamanla birlikte değişime uğrayan verilerin ayrıştırılmasında üst performans sağlayamamışlardır. Sabit değişime uğramayan verilerin sınıflandırmalarında başarılıdırlar.

Son yıllarda bilgisayar teknolojisinin gelişmesi ve eğitim süresinin kısalması Derin Öğrenmeye dayalı sistemlerin geliştirilmesi ve ASR uygulamalarında GMM'den daha iyi performans göstermesi sebepleriyle DNN'ler, akustik model eğitiminde yaygın olarak uygulanmıştır. Performansları istatistiksel yöntemlerden daha üst seviyededir. Çalışmaların devamında Derin Örenme yaklaşımı kullanılmasına karar verilmiştir. Derin Öğrenme, Kaldi ASR araç kutusu kullanılarak gerçekleştirilmiştir.

6.5.1. Kaldi ASR Araç Kutusu

Kaldi ASR araç kutusu, Daniel Povey tarafından kurulmuş bir açık konuşma tanıma aracıdır. HMM Toolkit'in (HTK) çeşitli talimatları entegre edildi ve ardından DNN modeli tanıtıldı. Sistem çerçevesi Şekil 6.7'da gösterilmektedir [35]. Kaldi konuşma tanıma araç seti eğitim ve kod çözme işlemlerini gerçekleştirir. Bir monofon sistemi, konuşma tanıyıcı için bir triphone HMM oluşturur. Triphone sistemi, monofon sisteminden türetilen hizalamaların temelini kullanır. Bu nedenle, GMM sistemi tarafından sağlanan hizalamalar derin öğrenme sistemini eğitmek için kullanılır [72]. Kaldi, bina akustik modelleri ve LM'ler sağlar. Kaldi, C ++ programlama dilinde yazılmıştır ve açık kaynaklı bir konuşma tanıma araç setidir. Kaldi araç seti birkaç kabuk komut dosyası ve C ++ çalıştırılabilir dosyası içerir. Kodların anlaşılması kolay, çok modern ve esnektir. Bu hem Linux hem de Microsoft Windows işletim sistemlerinde mevcuttur [80].



Şekil 6.7. Kaldi sistem çerçevesi [35]

Özellik çıkarma, etiket / hizalama hesaplama ve kod çözme Kaldi ile yapılmaktadır. Akustik modelleme GMM, DNN, RNN, LSTM ve GRU araç kitleri ile gerçekleştirilir. GMM ve Derin Öğrenme araç takımları tarafından her çerçeve için üretilen son olasılıklar, önceki olasılıklarına göre normalleştirilir. HMM tabanlı bir kod çözücü, elde edilen olasılıkları işler ve son olarak akustik, sözlük ve dil modeli bilgilerini birleştirdikten sonra kelimelerin sırasını tahmin eder [30]. Dil Modelleme için SRILM Toolkit for Kaldi kullanılır.

6.5.2. Akustik verilerin hazırlanması

Akustik verilerin Kaldi ASR eğitiminden önce hazırlanması gerekir. Eğitim ve test için kullanılan "Türkçe Mikrofon Konuşma" derlem verileri veri klasörüne yerleştirilmiştir (Bkz. Tablo 6.4). Analizlerde 120 kişiye ait ses kayıtları kullanılmaktadır. Eğitimde 120 kişi tarafından seslendirilen 4000 (120p x 40) cümle kullanılmaktadır. Ses verileri hazırlandıktan sonra, ilgili dil ve akustik modeller oluşturulur. Ardından üç dosya, wav.scp, metin ve utt2spk manuel olarak oluşturulur. Bu dosyalar ayrıca yazılı komut dosyaları kullanılarak da oluşturulabilir.

Tablo 6.4. "veri" klasörünün içeriği

Dosyalar	**Açıklamalar**
wav.scp	Konuşmacı ses dosyalarına erişen yol bilgisi
Text	Konuşmacıların konuşma içerikleri
utt2spk	Sesli fadelerin hangi konuşmacıya ait olduğunu belirtir.
spk2gender	Konuşmacı cinsiyeti
Corpus	ASR sistemindeki her bir ifade (konuşmacı başına 40 cümle)

MFCC özellikleri, Kaldi çalıştırdıktan sonra feats.scp dosyasında çıkarılır ve saklanır. Feats.scp, wav.scp, utt2spk, text ve spk2utt adlı dosyalar data / train klasörüne yerleştirilir. Derlem veri / yerel klasöre yerleştirilir (Bkz.Tablo 6.4)

6.5.3. Dil verilerinin hazırlanması

Data / local dizinde "dict" adlı bir klasör oluşturulur. Sözlük ve ses birimleri ile ilgili dosyalar klasörde oluşturulur (Bkz. Tablo 6.5).

Tablo 6.5. "dict" klasörünün içeriği

Dosyalar	**Açıklama**
lexicon.txt	Kelimelerin fonem bileşenlerine ayrılması
nonsilence_phones.txt	Dildeki fonemler (39 fonem(sesbirim)- Türkçe)
silence_phones.txt	Sessizği temsil eden fonemler

"Lexicon.txt" dosyası, sözlüğümüzdeki her kelimeyi fonem/harf çevirileriyle birlikte içerir. bazı kelime yapılan fonem çevirileri Tablo 6.6'da gösterilmektedir. “Nonsilence_phones.txt” dosyasında 38 sesbiriminin tamamı Türkçe bulunmaktadır. "Silence_phones.txt" dosyası sessiz fonemleri listeler.

Tablo 6.6. Derlemdeki bazı kelime kökleri, son ekler ve fonem bileşenleri

Kök/Son ek	**Fonem Bileşenleri - Lexicon**
SÖZLEŞME	S OE Z L EE SH M EE
NİN	NN IY NN
YAP	Y AA P
ILMASINI	I LL M AA S I NN I
KİM	K IY M
İSTİYOR	IY S T IY Y O RH
BUNDA	B U NN D AA
KÖŞE	K OE SH EE
GÖRÜN	G OE RR UE NN
NÜR	NN UE RH
ŞEKİL	SH EE K IY L
DE	D EE
KES	K E S
İLME	IY L M EE
MİŞ	M IY SH

6.5.4 Kullanılan Test Yöntemi

Derin Öğrenme tabanlı sistemlerin başarı analizlerinde, k-katlamalı çapraz doğrulama yöntemi (k-fold cross-validation) kullanılmıştır. Toplam örneklem kümesinin k adet kısma bölünmesinden oluşur; burada (k - 1) bölümleri eğitim alt kümesini oluşturmak için kullanılır ve kalan bölüm test alt kümesini oluşturmak için kullanılır. Bu

bölümlemeden sonra, tüm bölümler bir test alt kümesi olarak kullanılana kadar, öğrenme süreci k kez tekrarlanır. k mevcut örneklerin toplam sayısı ile bağlantılıdır ve genellikle 5 ile 10 arasında tanımlanır. Her aday topolojinin global performansı, k denemelerinin her birindeki bireysel performansların ortalamasının değerlendirilmesiyle elde edilir[81].

Çalişmalarımızda, 120 kişiden oluşan veri setimiz 6 (k=6) adet bölüme ayrılmıştır. 5 adet bölüm eğitim alt kümesi, 1 adet bölüm de test alt kümesi için kullanılır. Öğrenme süreci 6 kez tekrarlanmıştır.

6.5.5. Öznitelik Çıkarılması

Özellik çıkarma adımı, sinyalin 10 ms'lik bir örtüşme ile 25 ms'lik çerçevelere bölünmesini ve özellik katsayılarının tahmin edilmesini içerir. Bu çerçeveler, 12 MFCC parametresi ve çerçeve enerjisinden oluşan 13 parametre ile temsil edilir. Deneysel aktivite, farklı akustik özellikler, yani 39 MFCC (13 statik + türev + ikinci türev) dikkate alınarak gerçekleştirilir.

6.5.6. DBN Modeli Eğitimi

Kaldi'de kod çözme ve eğitim için shell ve Perl skriptleri kullanılır. DBN, LSTM ve GRU sınıflandırma modelleri, Kaldi tarafından eğitilip HMM modeli ile birlikte kullanılmıştır.

Dil Modelleme için “SRILM Toolkit for Kaldi” kullanılır. Eğitim ve Kod Çözme üç adımda yapılır:

Adım 1: GMM-HMM model train klasöründeki verileri kullanılarak eğitilmiş üç ses birimleri (üçlü-ses) dayalı ve Test klasöründe verilerle tarafından çözülür.

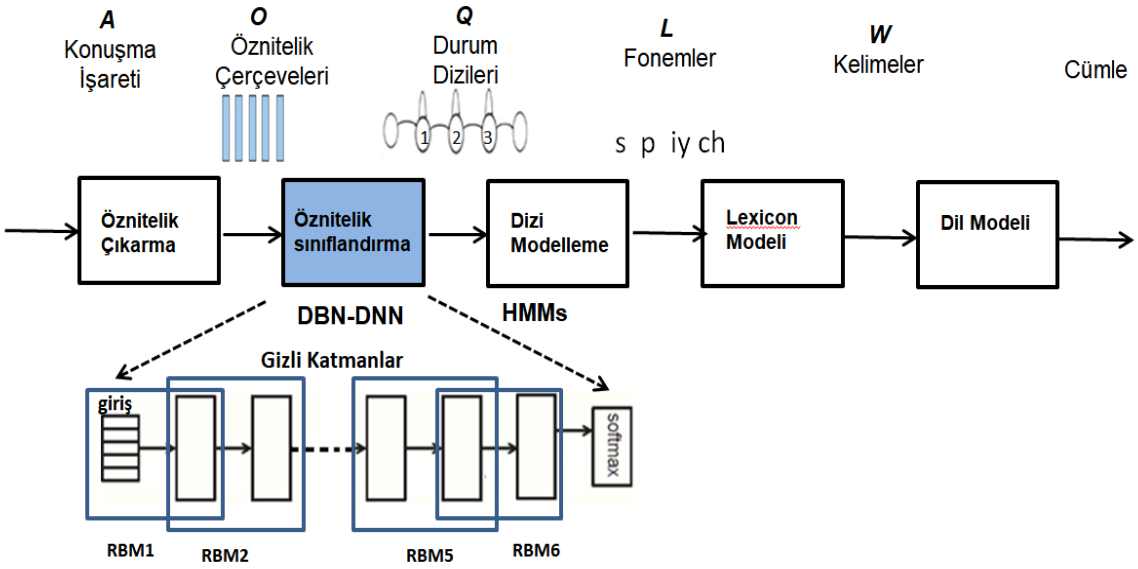
Adım 2: Triphone tabanlı DBN-HMM modeli eğitilir ve kodu çözülür.

DBN, 6 adet RBM'den oluşan bir yığınla oluşturulur (Bkz. Şekil 6.8). DBN-HMM'de 1024 boyuta sahip gizli birimlerde sigmoid aktivasyon fonksiyonu kullanılır (Bkz. Tablo 6.7).

Tablo 6.7. DBN-DNN Parametreleri

RBM Sayısı	Saklı Katman	Öğrenme Hızı
6	Sigmoid	Initial: $8x10^{-3}$
	dim:1024	Final: $7.81x10^{-6}$

Benzer uygulamalarda sigmoid için optimum başlangıç öğrenme oranı 8x10-3'tür. Eğitim bittiğinde nihai öğrenme oranı 7,81x10-6'dır (Bkz. Tablo 6.7).



Şekil 6.8. Sistem Mimarisi

Çapraz doğrulama tekniği eğitim setinin dışında yeni veri karşı DBN-HMM modeli sağlam yapmak için uygulanmaktadır. Bu şekilde ezber yapması, aşırı öğrenmesi (overfitting] önlenir. Veri kümesi, eğitim ve test alt kümelerine ayrılmıştır. Her yinelemede, test verileri alt kümesi değiştirilir ve sistem yeniden eğitilir.

Eğitim ve kod çözme işleminden sonra aşağıdaki PER değerlerini elde ettik. GMM-HMM ve DBN-MMM tanıyıcıları Kaldi üzerine çalıştırılmıştır. (bakınız Tablo 6.8)

Tablo 6.8 Fonem Hata oranları (PERs)

Model	PER
GMM-HMM	%30,64
DBN-HMM	%24,80

GMM-HMM PER değeri aynı derlem kullanıldığı biçimiyle, önceki çalışma [21]'deki değer ile tutarlıdır.

İkinci olarak, kelime tanıma performansı test edilmiştir. Sınıflandırıcıların WER değerleri Tablo 6.9'da verilmiştir.

Tablo 6.9 Kelime Hata oranları (WER'ler)

Model	**WER**
GMM-HMM	%17,21
DBN-HMM	%13,04

6.7. LSTM ve GRU Modeli Eğitimi

LSTM / GRU deneylerinde konuşma işaretindeki fonem hedeflerini tahmin etmek için eğitilir.

Alt kelime tabanlı LM eğitim için kullanılmıştır. Türkçe fonem temelli bir dil olduğu için, Türkçe'deki tüm ses birimleri de alt sözcük olarak eğitilir ve sözlüğe girilir. Böylelikle kelime dağarcığında bulunmayan kelimelerin fonem bileşenleri de tanınabilir.

Kaldi, GMM-HMM ve Derin Öğrenme tabanlı ASR kodlaması için de kullanılmıştır. GMM-HMM tabanlı sistemde, Türkçe için Kelime Tabanlı ve Alt Kelime Tabanlı LM uygulandığında Kelime Hata Oranı (WER) incelenmiş ve sonuçlar Tablo 6.10'de verilmiştir.

Tablo 6.10. Kelime Tabanlı ve Alt Kelime Tabanlı LM'lerin WER Karşılaştırması

Sistem	LM	WER
GMM-HMM	Kelime Tabanlı	%18,67
GMM-HMM	Alt-Kelime Tabanlı	%17,21

Kullanılan iki LM'nin performansları incelendiğinde, Türkçe gibi sondan eklemeli dil grubu için daha başarılı sonuçlar veren Alt Kelime tabanlı LM'nin yaptığı katkı doğrulanmaktadır. Bu model, çalışmanın sonraki adımlarında kullanılmıştır.

Önerilen çalışmada, Türkçe Mikrofon Konuşma Derlemini (ODTÜ 1.0) kullanan LSTM tabanlı tanıyıcı ile [21] 'de aynı derlemi kullanan GMM-HMM tanıyıcının performansları Fonem Hata Oranına (PER) göre karşılaştırılmıştır.

DNN gibi ileri beslemeli YSA'lar, ASR sorunu için önemli olan önceki durumların bilgilerini taşıyamaz veya işleyemez. RNN tipleri olan LSTM ve GRU'da kapıların açık veya kapalı olmasına göre söz konusu bilgiler hücre içinde saklanabilir veya okunabilir.

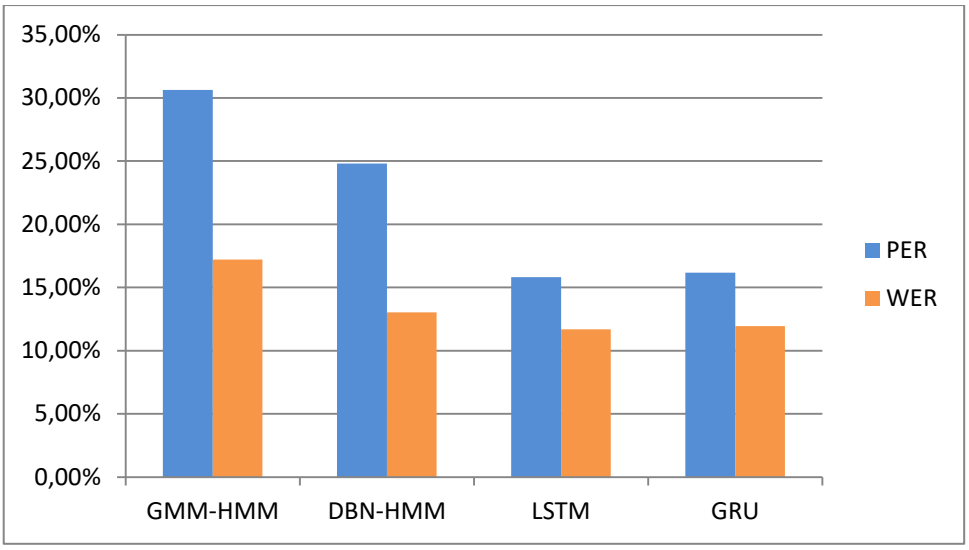
Kaldi üzerinde çalışan GMM-HMM, DBN, LSTM ve GRU konuşma tanıyıcılarının PER değerleri Şekil 6.9'da verilmiştir. Elde ettiğimiz GMM-HMM PER değeri% 30,64'tür. Bu değer, aynı derlemin kullanıldığı önceki çalışmanın [21] 'deki değeriyle tutarlıdır.

İki RNN tipi model eğitilmiştir (LSTM, GRU). LSTM ve GRU eğitimi için kullanılan aşağıdaki parametreler Tablo 6.11'de gösterilmektedir.

Tablo 6.11. LSTM ve GRU Parametreleri

Dropout Oranı	%20
Eğitim ve Test Batch Boyutu	8
Aktivasyon Fonksiyonu	tanh
Saklı Katmandaki Nöron Sayısı	550
Optimum ε	1e-8
Öğrenme Hızı	0.0004

LSTM tabanlı konuşma tanıyıcıyı aynı derlemede çalıştırdığımızda,% 15,82'lik bir PER değeri ve iyileştirilmiş tanıma performansı elde ettik. Kelime tanıma performansı incelendiğinde GMM-HMM Kelime Hata Oranı (WER)% 17,21 olarak ölçülmüştür. Önerilen LSTM tabanlı yapıda, WER performansı artmış ve değer Şekil 6.8'de görüldüğü gibi GMM-HMM'den daha düşük olan% 11,69'e düşmüştür. LSTM, GMM ile modellemeye göre daha başarılı sonuçlar vermiştir. Ek olarak, aynı kurulum GRU ile gerçekleştirildiğinde, PER ve WER sonuçları sırasıyla% 16,17 ve% 11,94 olarak elde edilmiştir.



Şekil 6.9 Uygulanan yöntemler için PER ve WER Değerleri

Hesaplama süresi LSTM için 73.518 saniye ve GRU için 61.020 saniyedir. GRU'nun eğitim süresi, LSTM'e göre daha kısadır. Buradan elde edildiği gibi, GRU'da% 17 zamandan tasarruf edilir ve WER sonuçları LSTM'ye yakındır (Tablo 6.12).

Tablo 6.12. LSTM ve GRU Hesaplama Süresi Karşılaştırması

	Hesaplama Süresi (s)	PER (%)	WER (%)
LSTM	73.518	15,82	11,69
GRU	61.020	16,17	11,94

Uygulama ve karşılaştırma sonuçları, tanıma performansının PER ve WER kriterlerine göre arttığını göstermektedir. Önerilen çalışmada, Türkçe Mikrofon Konuşma Derlemi (ODTÜ 1.0) kullanan LSTM, GRU ve DBN tabanlı tanıyıcıların performansı, aynı derlemi kullanan GMM-HMM tanıyıcıları ile karşılaştırılmıştır. [21,22], Fonem Hata Oranı (PER) ve WER'e göre karşılaştırılmıştır (Tablo 6.13-6.14).

Tablo 6.13. Fonem Hata Oranına (PER) göre karşılaştırma

Konuşma Tanıyıcı	Derlem	PER
SONIC Konuşma Tanıyıcı [21]	METU 1.0	%29,3
Kaldi Konuşma Tanıyıcı (GMM-HMM)	METU 1.0	%30,64
Kaldi Konuşma Tanıyıcı (DBN-HMM)	METU 1.0	%24,80
Kaldi Konuşma Tanıyıcı (LSTM)	METU 1.0	%15,82
Kaldi Konuşma Tanıyıcı (GRU)	METU 1.0	%16.17

Tablo 6.14. Kelime Hata Oranına (WER) göre karşılaştırma

Konuşma Tanıyıcı	Derlem	WER
HTK Konuşma Tanıyıcı (GMM-HMM) [22]	METU 1.0	%21,46
Kaldi Konuşma Tanıyıcı (GMM-HMM)	METU 1.0	%17,21
Kaldi Konuşma Tanıyıcı (DBN-HMM)	METU 1.0	%13,04
Kaldi Konuşma Tanıyıcı (LSTM)	METU 1.0	%11,69
Kaldi Konuşma Tanıyıcı (GRU)	METU 1.0	%11,94

6.8. Konuşmanın Türk İşaret Diline (TİD) Çevrilmesi

Tanıma gerçekleştikten sonra İlgili işaret dili video klibi video kütüphanesinden bulunur ve oynatılarak görsel bildiri yapılmış olur. İşaret dili video kütüphanesi, kelimelerin karşılık geldiği .avi formatlı kısa süreli video kliplerden meydana gelmektedir.

Şekil 6.10 Kaydedilen “merhaba” ifadesi için tanıma gerçekleştikten sonra gösterilen işaret dili video klibi [49]

TİD grameri Türk dili gramerinden farklıdır. TİD’de son ek kullanımı yoktur. TİD’de özne ve fiil her defasında iki ayrı sözcük ile belirtilir. TİD’de bu iki ayrı iki ayrı sözcük kullanımı İngilizce ile benzerdir ve zamir ve fiil iki ayrı sözcük olarak kullanılır. Fiil değişik zamirler ile birleştiği zaman yapısı değişmez.

Tablo 6.15. Türk İşaret Dili, Türk Dili ve İngilizce gramer örnekleri

İngilizce	**Türk Dili**	**Türk İşaret Dili**
I work	Çalışırım	Ben Çalışmak
You work	Çalışırsın	Sen Çalışmak
They work	Çalışırlar	Onlar Çalışmak

Şekil 6.11 “çalışmak” ifadesinin işaret dilinde karşılığı[49]

TİD’de temel olumsuz işaret DEĞİL anlamlı işarettir. Bu işaret genellikle cümlenin sonunda olumsuz hale getirdiği yüklemin hemen ardında görülür. Aynı cümle içinde

“DEĞİL”’in ardından başka bir işaretin geldiği çok nadir olarak görülür. Fakat zamirler olumsuzluk bildiren sözcüğün ardından gelebilir.

Tablo 6.16. Türk İşaret Dilinde olumsuz cümleler

Türk Dili	**Türk İşaret Dili**
Çocuklar Anlamıyorlar	Çocuklar Anlamak Değil
Görüşmedim	Görüşmek Değil Ben

Şekil 6.12 “değil” ifadesinin işaret dilinde karşılığı [49].

Yukarıdaki hususlar dikkate alınarak, gerekli İşaret Dili videoları seçilerek, Linux işletim sistemi komutları ile kelimeler karşılık gelen videolar oynatılarak, Cümle ve kelimelerin Türk İşaret Dili’ndeki karşılıkları görüntülenmiş olur.

7. SONUÇLARIN İNCELENMESİ

Önerilen Sistemde, ilk önce konuşma metne çevrilmekte olup, daha sonra metinden Türk İşaret Diline çevrim gerçekleşmektedir.

Çalışmalarımızda ilk önce, Konuşma Tanıma uygulamalarındaki ilk adım olan öznitelik çıkarma adımı ile ilgili, hangi özniteliğin kullanımı ile ilgili karar verebilmek için, bir Destek Vektör Makinaları tabanlı Konuşma fonem sınıflandırıcı geliştirilmiş olup, kullanımı en yaygın olan MFCC ve LPCC özniteliklerin performansları karşılaştırılmıştır. Fonem sınıflandırıcıda bulunan DVM katmanlarındaki sınıflandırma performansları incelendiğinde LPCC kullanıldığında sınıflandırma başarı oranı %88 ölçülüp, MFCC ile %98 ölçülmüştür. Buna göre, MFCC öznitelikleriyle performans artışını gözlemlemiş olup, müteakip çalışma ve geliştirmelerde MFCC öznitelikleri kullanılmıştır.

Destek Vektör Makineleri tabanlı sistem üzerinde, veri tabanındaki veri çeşitlililiği ve kişi sayısı artırılarak performans sonuçları tekrar incelenmiştir. Önerilen sistemde, veri setindeki“fonem”ler kullanılarak sınıflandırma aşaması uygulandı ve sınıflandırma başarısı analiz edildi. Sistemin eğitim setindeki kişi sayısı 2’den 10’a artarken, veri içinde bilgileri bulunan kişi için fonem tanıma başarısı %75’ten %85’e yükselmiştir. Aynı sayıda kişiler için, eğitimde olmayan kişi için tanıma başarısının %8’den %63’e yükseldiği görülmüştür. Bu çalışmadan elde edilen sonuçlara göre, kullanılacak konuşma derlemindeki verinin, kişi çeşitliliğine sahip olması, net ve kaliteli ses kayıtlarını barındırması gerektiği anlaşılmıştır.

Çalışmaların devamında, Eğitim ve test için LDC tarafından kabul edilen "Türkçe Mikrofon Konuşma Derlemi v1.0" kullanılıp, ayrıca zamanla durağan olmayan konuşma işaretini ayrıştırmada daha yetenekli olan Derin Öğrenme teknikleri kullanılarak devam edilmiştir. Türk dili için DBN-HMM tabanlı ASR sistemi geliştirilmiş olup, GMM-HMM tabanlı ASR tanıma yöntemi ile karşılaştırılmıştır. Önerilen sistemde GMM, Derin Öğrenme mimarisi ile değiştirilmiştir. DBN yöntemin tanıma performansı, [18] 'de uygulanan GMM-HMM yöntemi ile karşılaştırılmıştır. Analizlerde 120 kişiye ait ses kayıtları kullanılmaktadır. Eğitim için 100 kişinin ses kayıt kayıtları, test için 20 kişiye ait kayıtlar kullanıldı. [18] 'de, aynı derlem tarafından eğitilmiş GMM-HMM konuşma tanıyıcı yer almaktadır. PER% 29,3 olarak ölçülmüştür. Önerilen çalışmada, GMM-HMM

mimarisini kullanan Kaldi konuşma tanıyıcının PER'i% 30,64 olarak ölçülmüştür. Burada sonuçların benzer ve tutarlı olduğunu görebiliriz.

DBN-HMM konuşma tanıyıcıyı aynı derlem eğittiğimizde, PER değeri % 24,8'e gerilemiş olup, iyileştirilmiş bir tanıma performansı elde etmiş olduk. Kelime tanıyıcı dikkate alındığında GMM-HMM'nin WER değeri% 17,21 olarak ölçülmüştür. Önerilen Derin Öğrenme tabanlı DBN-HMM yapıda, WER, GMM-HMM'den daha düşük olan% 13,04'tür.

Benzer bir çalışmada [23], hem GMM-HMM hem de DNN-HMM sistemleri, yazarlar tarafından kaydedilen ve hazırlanan veritabanı tarafından eğitilmektedir. Sistemlerin WER'i sırasıyla% 17,40 ve% 14,65 olarak ölçülmüştür. Elde edilen sonuçlar sunulan çalışmaya benzer. İki çalışma arasındaki farklar kullanılan konuşma derlem ve uygulanan Derin Öğrenme yöntemidir. Çalışmamızda Feed Forward Sinir Ağı yerine DBN kullanılmıştır. [23] 'te önerildiği gibi, ASR uygulamaları için DBN uygulanması tercih edilecektir. Çalışmamızın sonuçları, DBN kullanımının sistem üzerinde yaklaşık% 1,5 daha yüksek performans etkisine sahip olduğunu göstermektedir. Önerilen çalışmada, araştırmanın [23] aksine, standart bir veri seti olan ve LDC tarafından kabul edilen "Türkçe Mikrofon Konuşması v1.0" derleminden yararlanılmıştır. Diğer araştırmacılar tarafından erişilebilen onaylanmış bir veri setinin kullanılması, aynı veri tabanını kullanan yayınlanmış çalışmaların erişilebilirliğini, karşılaştırılabilirliğini ve değerlendirilebilirliğini artırır.

Türkçe için ASR'nin kalitesini ve performansını iyileştirmek için DBN tabanlı bir Türkçe ses birimi ve konuşma tanıyıcı önermiştir. Önerilen sistem, OOV kelimelerinin sistem kelime haznesindeki ve ses birimi bileşenlerini tanıdı. Sondan eklemeli diller için yaygın olarak kullanılan alt kelime (morfem) LM tercih edilir. Türk dilinin her ses birimi bir alt sözcük olarak modellenmiştir. Türkçenin ses temelli bir dil olduğu düşünüldüğünde, sözlükte yer almayan veya tanınamayan kelimelerin harf bileşenlerini birleştirerek yaklaşık bir sonuca ulaşmak istenir. Önerilen DBN tabanlı sistemin performansı, aynı veri setini kullanan geleneksel tanıma yöntemi olan GMM tabanlı HMM ile karşılaştırılır. Her iki mimaride de HMM, dizi ve dil modellemesi için kullanılır. Uygulama ve karşılaştırma sonuçları, tanıma performansının PER ve WER kriterleri dikkate alındığında arttığını göstermektedir. Ayrıca bu sonuçlardan Derin Öğrenme modellerinin daha derin ve daha

doğru özellik olasılıkları üretebildiği ve konuşma tanıyıcının daha ayırt edici bir yeteneğe sahip olduğu görülmektedir.

Türkçenin morfolojik yapısı göz önüne alındığında, LSTM Derin Öğrenme tekniği, Türk ASR sistemlerinin performansını artırmak için son çalışmalarda uygulanmıştır. Bu çalışmada, önerilen yöntemlerin performansı, iyi bilinen veri seti ve derlem kullanılarak klasik yöntemle karşılaştırılmıştır. WER kriterlerine göre karşılaştırıldığında, LSTM'nin performansı ortak derlem kullanıldığı GMM-HMM ve DBN-HMM yöntemlerinden daha yüksektir. Çalışmanın sonuçları incelendiğinde, LSTM tabanlı sistemin ASR probleminde ses tanıma ve kelime tanıma başarısını artırdığı görülmektedir. Derin Öğrenme modelleri, daha derin ve daha kesin özellik olanakları üretebilir ve konuşma tanımada daha belirgin bir yeteneğe sahip olabilir. GRU ağları, LSTM üzerinden hesaplama süresinden tasarruf sağlar ve birçok gizli katmana sahip daha derin ağlarda kullanılabilir. Performansı artırmak için daha büyük bir veri seti kullanılabilir veya RNN'leri birleştiren hibrit yöntemler uygulanabilir.

Genellikle sorulan bir soru da , "Apple-Siri" ve "Google Sppech Engine" gibi akıllı cihazlarda bulunan sesli asistanların doğruluk performansının nasıl yüksek olduğudur. Bu uygulamalar, her an sesinizin akustik modelini konuşma sesiniz üzerinden olasılık dağılımına dönüştürmek için bir Derin Sinir Ağlarını (DNN) kullanır. Bu uygulamaların ASR uygulamaları ilgili şirketlerin ağlarına ait bulutta bulunmakdır. Bulut sunucuları, ASR tarafından kullanılan akustik modellere büyük depolama tesisleri ve güncellemeler sağlayabilir. Sistemlerin doğruluğu, toplanması ve hazırlanması oldukça pahalı olan büyük veri kümeleri kullanılarak elde edilir. Dil modelleri genellikle çok büyük metinler üzerinde eğitilir. Büyük veri setlerini depolama ve işleme gücü sağlayan teknolojik altyapıya sahip olması, söz konusu asistanların karmaşık akustik ve dil modellerini işlemesine olanak tanır. Bu nedenle, bu ses asistanlarının performansı, sınırlı işlemci kaynaklarını ve veri kümelerini kullanan araştırma çalışmalarından daha yüksektir.

Çalışmada, Otomatik Konuşma Tanıma Uygulamalarında son yıllarda bilgisayar teknolojisinin gelişimi ile kullanılan Derin Öğrenme teknikleri Türkçe Konuşma Tanıma uygulaması için kullanılmış olup, performansı LDC tarafından onaylanmış standart bir konuşma derlemi ile test edilmiştir. Ayrıca Türkçe için uygulamasına literatürde rastlanmayan Türkçe Konuşmayı İşaret Diline Çeviren bir uygulama elde edilmiştir.

KAYNAKLAR

[1] M. Varjokallio, M. Kurimo, S. Virpioja, “Learning a Subword Vocabulary Based on Unigram Likelihood”, IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 7-12, 2013.

[2] M. Varjokallio, M. Kurimo, “A Word – Level Token – Passing Decoder for Subword N-gram LVCSR”, IEEE Spoken Language Technology Workshop (SLT), pp. 495-500, 2014.

[3] P. Smit, , S. R., Gangireddy, S. Enarvi, S. Virpioja, M. Kurimo, “Character-Based Units for Unlimited Vocabulary Continuous Speech Recognition”, IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 149-156, 2017.

[4] P. Mihajlik, Z. Tüske, B. Tárjan, B. Németh, T. Fegyó, “Improved Recognition of Spontaneous Hungarian Speech-Morphological and Acoustic Modeling Techniques for a Less Resourced Task”, IEEE Transactions On Audio, Speech, And Language Processing, Vol. 18, No. 6, pp. 1588-1600,August, 2010.

[5] E. Arısoy M. Saraclar, “Language Modelling Approaches for Turkish Large Vocabulary Continuous Speech Recognition Based on Lattice Rescoring”, 14th Signal Processing and Communications Applications, IEEE, 2006

[6] T. Aksungurlu S. Parlak H. Sak M. Saraçlar, “Comparison of Language Modelling Approaches for Turkish Broadcast News”, 16th Signal Processing, Communication and Applications Conference, IEEE, 2008

[7] N. Shewalkar, D. Nyavanandi, S. A. Ludwig, “Performance Evaluation of Deep Neural Networks Applied to Speech Recognition: RNN, LSTM and GRU”, Journal of Artificial Intelligence and Soft Computing Research, 9(4), pp. 235-245, 2019.

[8] J. Kang W. Zhang, J. Liu, “Gated Recurrent Units Based Hybrid Acoustic Models for

Robust Speech Recognition", 10th International Symposium on Chinese Spoken Language Processing (ISCSLP), Conference, Tianjin, 2016.

[9] H. Dridi, K. Ouni, "Towards Robust Combined Deep Architecture for Speech Recognition : Experiments on TIMIT", International Journal of Advanced Computer Science and Applications (IJACSA), 11(4), pp. 525-534, 2020.

[10] B. Tombaloğlu H. Erdem "Deep Learning Based Automatic Speech Recognition for Turkish", Sakarya University Journal of Science, 24(4), pp. 725 – 739, 2020.

[11] U: Kimanuka, O. Buyuk, "Turkish Speech Recognition Based On Deep Neural Networks" . Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 22, pp. 319-329, 2018.

[12] A. Graves, A. R. Mohamed, G. Hinton, "Speech Recognition with Deep Recurrent Neural Networks", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Conference, Vancouver, pp. 6645- 6649, 2013.

[13] Siri Team, "Hey Siri: An On-device DNN-powered Voice Trigger for Apple's Personal Assistant", machinelearning.apple.com, https://machinelearning.apple.com /research/hey-siri, (Accessed: Jan. 7, 2021).

[14] F. Beaufays, "The Neural Networks Behind Google Voice Transcription", ai.googleblog.com, https://ai.googleblog.com/2015/08/ the-neural-networks-behind-google-voice.html, (Accessed: Jan. 7, 2021).

[15] S. Keser, R. Edizkan, "Altuzay Sınıflama ile Sesbirim Temelli Türkçe Yalıtık Kelime Tanıma", SIU 2009.

[16] O. Eray, "Destek Vektör Makineleri ile Ses Tanıma Uygulaması", Pamukkale Üniversitesi, Yüksek Lisans Tezi, 2008.

[17] B. Tombaloğlu, H. Erdem, “Development of a SVM-MFCC Based Turkish Speech Recognition System”, 24th Signal Processing and Communication Application Conference (SIU), IEEE, 2016.

[18] B. Tombaloğlu, H. Erdem, “A SVM based speech to text converter for Turkish language”, 25th Signal Processing and Communication Application Conference (SIU), IEEE, 2017.

[19] E. Arısoy, “Devoloping an Automatic Transcription and Retrieval system for Spoken Lectures in Turkish”, 25th Signal Processing and Communications Applications Conference (SIU), IEEE, 2017

[20] A. Dhankar, “Study of deep Learning and CMU Sphinx in Automatic Speech Recognition”, International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2296-2301, 2017.

[21] O. Salor, B.L. Pellom, T. Çiloğlu, Demirekler, M., “Turkish speech corpora and recognition tools developed by porting SONIC: (Towards multilingual speech recognition)”, Computer Speech and Language, Elsevier, 21, pp. 580–593, 2007.

[22] A. O. Bayer, T. Çiloglu, M. T. Yondem, “Investigation of Different Language Models for Turkish Speech Recognition”, 14th Signal Processing and Communications Applications, IEEE, 2006.

[23] D. Susman, S. Köprü, A. Yazıcı, “Turkish Large Vocabulary Continuous Speech Recognition By Using Limited Audio Corpus”, 20th Signal Processing and Communications Applications Conference (SIU), IEEE, 2012

[24] E. Arısoy M. Saraclar “Compositional Neural Network Language Models for Agglutinative Languages”, Interspeech, San Francisco, USA, pp. 3494-3498, 2016.

[25] O. Büyük U. A. Kimanuka, “Turkish Speech Recognition Based on Deep Neural Networks”, Süleyman Demirel University Journal of Natural and Applied Sciences Volume 22, Special Issue, pp. 319-329, 2018.

[26] O. Büyük, ‘A New Database for Turkish Speech Recognition on Mobile Devices and Initial Speech Recognition Results Using The Database”, Pamukkale University Journal of Engineering Sciences Volume 24-2, pp. 180-184, 2018

[27] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, B. Kingsbury, “Deep Neural Networks for Acoustic Modelling in Speech Recognition”, IEEE Signal Processing Magazine, 29(6), pp. 82-97, 2012.

[28] A. Graves, N. Jaitly, “Towards End to End Speech Recognition with Recurrent Neural Networks”, 31st International Conference on Machine Learning, Conference, Beijing, pp. 1764-1772, 2014.

[29] K. Huang, A. Hussain, Q., Wang, R., Zhang, “Deep Learning: Fundamentals, Theory and Applications”, Springer, Edinburg, 2019.

[30] M. Ravanelli, T. Parcollet, Y. Bengio, “The Pytorch-Kaldi Speech Recognition Toolkit”, 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Conference, Brighton, 2018.

[31] M. Ravanelli, P. Brakel, M. Omologo, Y. Bengio, “Light Gated Recurrent Units for Speech Recognition”, IEEE Journal Of Emerging Topics In Computational Intelligence, 2(2), pp. 92-102, 2018.

[32] G. Işık, H. Artuner, “Turkish Dialect Recognition In Terms Of Prosodic By Long Short-Term Memory Neural Networks”, Journal of the Faculty of Engineering and Architecture of Gazi University, 35(1), pp. 213-224, 2020.

[33] R. Arslan, S. N. Barışçı, “The Effect of Different Optimization Techniques on End-to-End Turkish Speech Recognition Systems that use Connectionist Temporal Classification”, 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Conference, Turkey, 2018.

[34] E. Arisoy, M. Saraclar, “Multi-Stream Long Short-Term Memory Neural Network Language Model”, 16th Annual Conference of the International-Speech-Communication-Association (INTERSPEECH 2015), Conference, Dresden, pp. 1413-1417, 2015.

[35] W. Ruan, Z. Gan, B. Liu, Y. Guo, “An Improved Tibetan Lhasa Speech Recognition Method Based on Deep Neural Network”, 10th International Conference on Intelligent Computation Technology and Automation, IEEE, pp. 303-306, 2017.

[36] Sisi Team, “IBM Research Demonstrates Innovative "Speech to Sign Language' Translation System”, https://www-03.ibm.com, https://www-03.ibm.com/ press/ us/en/pressrelease/22316.wss, (Accessed: Jan. 7, 2021).

[37] O. M. Foong, T. J. Low, W.W. La, “V2S: Voice to Sign Language Translation System for Malaysian Deaf People” in: Badioze Zaman H., Robinson P., Petrou M., Olivier P., Schröder H., Shih T.K. (eds) Visual Informatics: Bridging Research and Practice. IVIC 2009. Lecture Notes in Computer Science, vol 5857. Springer, Berlin, Heidelberg, 2009.

[38] Mind Rockets Inc, “Mimix Sign Language Translator”, apps.apple.com, https://apps.apple.com/ca/app/mimix-sign-language-translator/id1156035569, (Accessed: Jan. 7, 2021).

[39] R. San-Segundo, R. Barra-Chicote, L. F. D'Haro, J.M. Montero, “A Spanish Speech to Sign Language Translation System for Assisting Deaf-Mute People.” INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, September 17-21, 2006

[40] M. Boulares, M. Jemni, “Toward an Example-Based Machine Translation from Written Text to ASL Using Virtual Agent Animation”, International Journal of Computer Science Issues (IJCSI);, Vol. 9, Issue 1, pp. 379-388, January, 2012.

[41] A. Othman, M. Jemni, “Statistical Sign Language Machine Translation: from English written text to American Sign Language”, Gloss International Journal of Computer Science Issues (IJCSI), Vol. 8, Issue 5, No 3, September, 2011.

[42] R. San-Segundo, J. M. Montero, R. Córdoba, V. Sama, F. Fernández, L. F. D’Haro, V. López-Ludeña, D. Sánchez, A. García, “Design, Development and Field Evaluation of a Spanish into Sign Language Translation System”, Pattern Analysis and Applications, Volume 15, Issue 2, pp 203-224, May, 2012.

[43] A. Almohimeed, M. Wald, R.I. Damper, “Arabic Text to Arabic Sign Language Translation System for the Deaf and Hearing-Impaired Community”, in: EMNLP 2011: The Second Workshop on Speech and Language Processing for Assistive Technologies (SLPAT), Edinburgh, UK, Scotland, 2011, pp. 101-109.

[44] M. Selçuk Şimşek “Türkçe ile Türk İşaret Dili arasında örneğe dayalı makine çeviri sistemi”, Yüksek Lisans Tezi, Bilgisayar Mühendisliği, Hacettepe Üniversitesi, 2016.

[45] C. Eryiğit, “Yazılı Türkçe dilinden Türk İşaret Diline (tid) Makine Çevirisi Sistemi”, Doktora Tezi, Bilgisayar Mühendisliği, İstanbul Teknik Üniversitesi, 2017.

[46] M. F. Karaca, “Üç Boyutlu Sanal Model ile Türk İşaret Dili Simülasyonu”, Doktora Tezi, Bilgisayar Mühendisliği, Karabük Üniversitesi, 2018.

[47] M. Yasan, “Türkçe Metni Türk İşaret Diline Dönüştürme, Yüksek Lisans Tezi, Başkent Üniversitesi, Ankara, 2014.

[48] “Türk İşaret Dili Kaynak Sitesi”, Boğaziçi Üniversitesi Bilgisayar Mühendisliği

Bölümü, İstanbul, http://www.cmpe.boun.edu.tr/tid/ (Erişim Tarihi: 29.01.2021).

[49] "Türk İşaret Dili Sözlüğü", Türk Dil Kurumu, Ankara, https://www.tdk.gov.tr/ icerik/ basindan/turk-isaret-dili-sozlugu/, (Erişim Tarihi: 29.01.2021).

[50] "Türk İşaret Dili Sözlüğü", MEB Özel Eğitim ve Rehberlik Hizmetleri Genel Müdürlüğü, Ankara, http://orgm.meb.gov.tr/alt_sayfalar/duyurular/1.pdf, http://tid.meb.gov.tr/parmak-alfabesi/ (Erişim Tarihi: 29.01.2021)

[51] S. Keser, R. Edizkan, "Phoneme-Based Isolated Turkish Word Recognition With Subspace Classifier", 17th Signal Processing and Communications Applications Conference , IEEE, 2009.

[52] B. Asefisaray, A. Haznedaroğlu , M., Erden, L., M., Arslan, "Transfer Learning for Automatic Speech Recognition Systems", 26th Signal Processing and Communications Applications Conference (SIU), 2018

[53] E. Arısoy M. Saraclar, "Lattice Extension and Vocabulary Adaptation for Turkish LVCSR", IEEE Transactıons on Audio, Speech and Language Processıng, vol. 17, no. 1, 2009.

[54] V. Tunalı, "A Speaker Dependent Large Vocabulary Isolated Word Speech Recognition System for Turkish", Msc. Thesis, Marmara University, 2005.

[55] O. Büyük, "Sub-Word Language Modelling for Turkish Speech Recognition", Msc. Thesis, Sabanci University, 2005.

[56] Ö. Salor, B. Pellom, T., Çiloğlu, K. Hacıoğlu, and M., Demirekler, "On Developing New Text and Audio Corpora and Speech Recognition Tools for the Turkish Language", ICSLP-2002: Inter. Conf. On Spoken Language Processing, Denver, Colorado USA, pp. 349–352, 2002.

[57] İ. Ergenç, “Konuşma Dili ve Türkçenin söyleyiş sözlüğü”, Multilingual, Istanbul, 2002, p. 486.

[58] E. Arısoy, L. M., Arslan, “Turkish Dictating System for Broadcast News Applications”, 13th European Signal Processing, Conference, Antalya, (2005).

[59] S. Kasartova, “Metinden Bağımsız Konuşmacı Tanıma Sistemlerinin İncelenmesi ve Gerçekleştirilmesi”, Yüksek Lisans Tezi, Ankara Üniversitesi, 2011.

[60] L. Muda, M. Begam I. Elamvazuthi, “Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques”, Journal of Computing, 2(3), pp. 138-143, 2010.

[61] M. N. Stuttle, “A Gaussian Mixture Model Spectral Representation for Speech Recognition”, Ph.D. Thesis, Cambridge University, 2003.

[62] D. Schiopu, “Using Statistical Methods in a Speech Recognition System for Romanian Language”, 12th IFAC Conference on Programmable Devices and Embedded Systems, Velke Karlovice, Czech Republic, pp. 99-103, 25-27 September 2013.

[63] E. Köklükaya, İ. Coşkun, "Endüktif Öğrenmeyi Kullanarak Konuşmayı Tanıma". Sakarya University Journal of Science, vol.7, pp. 87-94, 2003

[64] C. Aksoylar, S. Mutluergil, H., Erdoğan “Bir Konuşma Tanıma Sisteminin Anatomisi”, IEEE 17th Signal Processing and Communications Applications, Conference, Antalya, pp. 512-515, 2009.

[65] C. Kurian, S. A. Firoz, K. Balakrishnan, “Isolated Malayam Digit Recognition Using SVM”, ICCCCT’10, p:692-695, 2010.

[66] V. Kecman, “Learning and Soft Computing” MIT Press, 2001.

[67] A. Khan, M. Farhan, A. Ali, “Speech Recognition: Increasing Efficiency of Support Vector Machines”, International Journal of Computer Application, Volume 35, No:7, 2011.

[68] M. R. Alam, M. Bennamoun, R. Togneri, F. Sohel “Deep Neural Networks for Mobile Person Recognition with Audio-Visual Signals”, Mobile Biometrics, 2017, p. 97-129

[69] A. C. Banumathi, Dr. E. Chandra, “Deep Learning Architectures, Algorithms for Speech Recognition: An Overview”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 7, Issue 1, pp. 213-220 January, 2017.

[70] S. M. Siniscalchi, T. Svendsen, C. Lee, “An Artificial Neural Network Approach to Automatic Speech Processing”, Neurocomputing, Elsevier, Vol. 140, pp. 326-338. 2014.

[71] R. V. Sharan, T. J. Moir, “An Overview of Applications and Advancements in Automatic Sound Recognition”, Neurocomputing, Elsevier, Vol. 200, pp. 22-34, 2016.

[72] R. Sustika, A. R. Yuliani, E. Zaenudin, H. F. Pardede, “On Comparison of Deep Learning Architectures for Distant Speech Recognition”", 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE), IEEE, 2017.

[73] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F.E. Alsaadi, “A Survey of Deep Neural Network Architectures and Their Applications”, Neurocomputing, Elsevier, Vol. 234, pp. 533-541, 2017.

[74] Y. Guan, Z. Yuan, G. Sun, J. Cong, “Fpga-Based Accelerator for Long Short-Term Memory Recurrent Neural Networks”, 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), Conference, Chiba, 629–634, 2017.

[75] S. Hochreiter, J. Schmidhuber, “Long Short-Term Memory”, Natural Computation, 9(8): 1735-1780, 1997.

[76] A. Graves, J. Schmidhuberab, “Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures”, International Joint Conference on Neural Networks (IJCNN), Conference, Montreal, pp. 602-610, 2005.

[77] R. S. Arslan, N., Barışçı, “A Detailed Survey of Turkish Automatic Speech Recognition”, Turkish Journal of Electrical Engineering & Computer Sciences, 28: pp. 3253-3269, 2020.

[78] E. Arısoy, M. Saraclar, “Turkish Speech Recognition”, Turkish Natural Language Processing, Springer, 2018.

[79] H. Polat, S. Oyucu, “Building a Speech and Text Corpus of Turkish: Large Corpus Collection with Initial Speech Recognition Results”, Symmetry, 12(2): 290, 2020.

[80] G T. Yadava, H S. Jayanna, “Creating Language and Acoustic Models using Kaldi to Build An Automatic Speech Recognition System for Kannada Language”, 2nd IEEE International Conference On Recent Trends in Electronics Information and Communication Technology (RTEICT), India, IEEE, pp. 161-165, May 19-20, 2017,

[81] I., Nunes Silva, D., Hernane Spatti, R., Andrade Flauzino, , L.H.B., Liboni, S.F. dos Reis Alves, “Artificial Neural Networks A Practical Course”, Springer International Publishing, Switzerland, 2017.