BAŞKENT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

BELİRLİ BİR SÖZCÜĞÜN SESLENDİRİLMİŞ TÜRKÇE METİNLER İÇİNDE BULUNMASI

ELÇİN ERKİN

YÜKSEK LİSANS TEZİ

2010

BELİRLİ BİR SÖZCÜĞÜN SESLENDİRİLMİŞ TÜRKÇE METİNLER İÇİNDE BULUNMASI

DETECTING A WORD IN TURKISH AUDIO TEXTS

ELÇİN ERKİN

Başkent Üniversitesi

Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin

ELEKTRİK-ELEKTRONİK Mühendisliği Anabilim Dalı İçin Öngördüğü

YÜKSEK LİSANS TEZİ

olarak hazırlanmıştır.

2010

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma, jürimiz tarafından **ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI 'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Başkan Prof. Dr. Murat Emin AKATA

Üye Yrd. Doç. Dr. Mustafa Doğan

Üye Yrd. Doç. Dr. Hasan Oğul

ONAY

Bu tez ...../...../2010 tarihinde, yukarıdaki jüri üyeleri tarafından kabul edilmiştir.

..../..../2010

Prof.Dr. Emin AKATA

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

Bu tezi yazmamı sağlayan
Annem'e...

TEŞEKKÜR

Tez çalışmalarım esnasında bana destek olan, yardımlarını esirgemeyen, yol gösteren ve en önemlisi başarabileceğime olan inançlarını kaybetmeyen değerli danışmanım, hocalarım, ailem ve arkadaşlarıma teşekkürü bir borç biliyorum.

Danışmanım Prof. Dr. Emin Akata'ya beni tez öğrencisi olarak seçtiği ve böyle özel bir alanda çalışma şansı tanıdığı, yapıcı yaklaşımlarıyla yol gösterdiği ve geçtiğim aşamalarla yakından ilgilendiği için,

Dr. Mustafa Doğan'a, yoğun çalışma temposuna rağmen, bana zaman ayırdığı, yaratıcı mühendislik yaklaşımları ve engin bilgi birikimini her zaman çözüm odaklı kullanarak en stresli zamanlarımda bana destek olduğu için,

Boğaziçi Üniversitesi'nden Dr. Murat Saraçlar ve BUSIM grubundan, Haşim Sak ve Doğan Can'a, tez çalışmalarında ihtiyacım olan kaynakları ve değerli birimlerini paylaştıkları, farklı fikirleri ile yeni yöntemler deneme ışık tuttukları için,

Uykusuz gecelerde beni yalnız bırakmayan, bitmeyen sevgileri ile her zaman en büyük destekçim olan ailem ve eşi bulunmaz bir arkadaş, kardeş, her şeyim ve 'alter ego'm Beril'ime,

Sonsuz sabrıyla beni dinleyen, sıkıntılı günlerime mutluluk katan Emrah Onur Toprak'a teşekkürlerimi sunuyorum.

Yaşamımda bu değerli insanlar olmasa bu tezin bitmiş olabileceğini hayal dahi edemiyorum.

ÖZ

BELİRLİ BİR SÖZCÜĞÜN SESLENDİRİLMİS UZUN TÜRKÇE METİNLER İÇİNDE BULUNMASI

Elçin Erkin

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Bilişim teknolojilerinin hızla ilerlediği günümüzde, insan – bilgisayar etkileşimi de oldukça yaygın hale gelmiştir. Bu etkileşimde kullanılan klavye, fare gibi çevresel birimlere kıyasla, ses ve konuşma, kullanıcıya kendini daha rahat ifade etme olanağı sağlamaktadır. Oldukça geniş bir yelpazede yaşama geçirilebilecek alana sahip olan konuşma tanıma uygulamalarından birisi ve bu tezin de konusu Sözcük Tanımadır. Sözcük tanıma, seslendirilmiş bir metin içerisinde, sorgu olarak verilen sözcüklerin varlığının ve bulundukları yerin tespiti olarak tanımlanabilir. Uygulama alanları arasında içerik tarama ve güvenlik uygulamaları sayılabilir.

Bu tez kapsamında gerçekleştirilen çalışma, seslendirilmiş bir Türkçe metinde sözcük tespitine dair yöntemlerin denenmesi ve sonuçların incelenmesidir. Bu amaçla ses kayıtları toplanmış, bu kayıtlar tanıma uygulamasında kullanılacak şekilde segmentlere ayrılmış, transkripsiyonları çıkarılmış ve eğitilmiştir. Eğitim açık kaynak kodlu konuşma işleme araçları kullanılarak gerçekleştirilmiştir. Dil ve akustik modeller geliştirilirmiş, tanıma sonuçları incelenmiştir. Sonuçlar bir tanıma uygulaması ile değerlendirilmiştir.

ANAHTAR SÖZCÜKLER: Konuşma işleme, konuşma tanıma, sözcük saptama, HMM, HTK

Danışman: Prof Dr Emin AKATA, Başkent Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü.

ABSTRACT

DETECTING A WORD IN TURKISH AUDIO TEXTS

Elçin Erkin

Başkent University Institute of Science

Department of Electrical-Electronics Engineering

The human – computer interaction becomes very common and a major part of our life with the emerging developments in the information technology. As speech is considered to be a more natural interface than keyboard or mouse, speech driven communication is also becaming more of an issue. Speech recognition has a broad application area. One of those applications which is also the subject of this thesis is Word Identification. The purpose of word identification can be summarized as detecting a query word in an audio text. It is mainly used in information retrieval and security areas.

The work done in this thesis can be summarized as making acoustic and language models, training speech data and making evaluations from the results.

KEYWORDS: Speech recognition, Word Identification, HMM, HTK

Advisor: Prof. Dr Emin AKATA, Başkent Üniversitesi, Electrical – Electronics Engineering

İÇİNDEKİLER

ŞEKİLLER LİSTESİ

SİMGELER VE KISALTMALAR LİSTESİ

S	Durum uzayı
q_t	t anındaki durum
t	Anlık zaman gösterimi
V	Gözlem uzayı
A	Durum geçiş matrisi
N	Durum sayısı
M	Gözlem sayısı
a_{ij}	i durumundan j durumuna geçme olasılığı
B	Gözlemlenen sembollerin olasılıksal dağılım seti
v_i	Tekil gözlemler
s_i	Tekil durumlar
o_t	Gözlemlenen sembol
π	Başlangıç durumu
O	Gözlem dizisi
q	Durum dizisi
λ	Parametre modeli
β	Kısmi gözlem dizisinin i durumunda t zamanında λ modeli ile ortaya çıkma olasılığı
δt	Viterbi değişkeni
ψ1	Maksimizasyon kriteri
ξ	Si durumunun t zamanında ve Sj durumunun t+1 zamanında gözlemlenme olasılığı
$P(w_1...w_n)$	Söz dizisinin gözlemlenme olasılığı
$C(w_i...w_n)$	Söz dizisinin görülme miktarı
I	Kafes indeksi
J	Nod indeksi

S	Başlangıç nodu
E	bitiş nodu
a	akustik model olasılığı
l	dil modeli olasılığı
SRLIM	SRI language modelling
ISR	Isolated Speech Recognition
CSR	Continuous Speech Recognition
ASR	Automatic Speech Recognition
LVCSR	Large Vocabulary Continuous Speech Recognition
HMM	Hidden Markov Model
HTK	Hidden Markov Model Toolkit
IBM	International Business Machines
AT & T	American Telephone and Telegraph
DTW	Dynamic Time Warping
DP	Dynamic Programming
ANN	Artificial Neural Networks
MFCC	Mel Frequency Cepstrum Coefficients
FSM	Finite State Machine
WFSN	Weighted Finite State Network
WFST	Weighted Finite State Transducers
WER	Word Error Rate
FSM	Finite State Machine
FOM	Figure of Merit
ROC	Receiver Operating Characteristics
LPC	Linear Predictive Coding
FFT	Fast Fourier Transform
DCT	Discrete Cosine Transform

LM	Language Modelling
BOUN	Boğaziçi Üniversitesi
BUSIM	Boğaziçi University Speech and Image Processing Group
STD	Spoken Term Detection
STDEval	Spoken Term Detection Evaluation
NIST	National Institute of Standards and Technology
RTTM	Rich Transcription Time Mark
ECF	Experiment Control File
DET	Detection Error Tradeoff
ATWV	Term Weighted Value

1. GİRİŞ

Konuşma günlük yaşantımızda en sık kullandığımız iletişim araçlarından biridir. 21. yüzyılda hızla ilerleyen teknoloji insanlar arası konuşmanın yanı sıra makinelerle de iletişim içinde olmamızı gerektirmekte ve bunu hayatımızı kolaylaştıracak uygulamalar olarak bize sunmaktadır. İnsan makine etkileşiminde en önemli adımlardan birisi konuşma tanıma uygulamalarını içermektedir. Komut alan, buna göre istenen çıktıyı veren cihazlar, telefon aracılığıyla gerçekleştirilen rezervasyon yaptırma veya bilgi edinme uygulamaları, akıllı ev/ofis otomasyonları ve bunun yanı sıra savunma sanayisinde dinleme ve tespit işlemleri gibi birçok oluşum bize daha rahat ve güvenli bir yaşam ortamı sağlamak için geliştirilmektedir.

Yapılan çalışmalarla bu alanda oldukça büyük ilerlemeler kat edilse de yüksek başarı düzeyinde bir tanıma henüz gerçekleştirilememiştir. Bu durumun sebeplerinden biri makineler tarafından gerçekleştirilmesi beklenen tanıma işleminin disiplinler arası doğasının karmaşıklığıdır. [1] Basit bir tanıma işleminde ihtiyaç duyulan farklı disiplinlerdeki uygulamalar şu şekildedir;

i. Sinyal işleme
ii. Akustik
iii. Örüntü tanıma
iv. Haberleşme ve bilişim teorisi
v. Linguistik
vi. Fizyoloji
vii. Bilgisayar bilimleri
viii. Psikoloji

Bu nedenle konuşma tanıma disiplinler arası bir alan olarak kabul edilip başarılı bir uygulama geliştirebilmek için bütün alanlarda başarılı bir analizin gerçekleşmesi gerekmektedir.

1.1 Amaç ve Kapsam

Bu tez kapsamında incelenecek olan konu ses dosyası olarak saklanmış Türkçe okuma metinlerinde belirli bir sözcüğün ve bu sözcükten türetilmiş sözcüklerin nerelerde geçtiğinin bulunmasıdır. Bu bağlamda öncelikle mevcut konuşma tanıma teknolojilerinin gelişimi incelenmiş, yeni teknikler araştırılmıştır. Yaygın olarak kullanılmakta olan konuşma tanıma araçlarından Cambridge Üniversitesi tarafından geliştirilmiş olan Hidden Markov Model Toolkit (HTK) yapılan çalışmalarda araç olarak tercih edilmiştir. HTK ile eğitilen ve test edilen kayıtlar üzerinde modellemeler denenmiş ve optimal bir yöntem aranmıştır.

1.2 Geçmiş Çalışmalar

Konuşma tanıma çalışmaları ilk olarak 1952 senesinde Bell Laboratuarları'nda Davis, Biddulph ve Balashek'in çalışmaları ile başlamıştır. Bir rakamın sesli harflerinin formant frekanslarının ölçümü ile sonradan Yalıtılmış Konuşma Tanıma (Isolated Speech Recognition - ISR) olarak tanımlanan, ilk kelime tanıma sistemini gerçekleştirilmiştir.

Aynı yıllarda RCA Laboratuarları'nda Olson ve Belar [2] bir konuşmacının söylediği 10 kelimeyi tanıyan bir sistemi ve MIT Lincoln Laboratuarları'nda ise Forgie ve Forgie [3] konuşmacı bağımsız 10 kelimeyi tanıyan bir sistemi hayata geçirmişlerdir. Bu sistemler de Bell Laboratuarları'ndakiler gibi bir analog filtre kümesi ile kelime içinde geçen sesli harfin spektral ölçümlerine göre tanıma yapmaktadır. RCA Laboratuarları'nda gerçekleştirilen uygulamanın ayırt edici özelliği ise fonem dizisinde tanıma gerçekleştirirken, dizilerin istatistiksel bilgilerine de seçim tespit aşamasına da yer verilmesi olmuştur.

Sürekli Konuşma Tanıma (Continuous Speech Recognition - CSR) konusunda yapılmış ilk çalışmalar 1960'lı yılların başında Japonya'da çeşitli laboratuarlarda gerçekleşmiştir. Burada konuşma tanıma işlemi için farklı sistem donanımları inşa edilebilmiştir. Bunlar arasında en dikkat çekici olanları Suzuki ve Nakata tarafından Tokyo Radio Research Laboratuarları'nda gerçekleştirilen sesli harf

tanıma [4] ve Sakai ve Doshita tarafından Kyoto Üniversite'sinde tasarlanan fonem tanıma birimleridir [5].

Konuşma tanıma çalışmalarında 1960'li yıllarda önemli kavramlar öne sürülmüştür. İstatistiksel söz dizimi (statistical syntax) kullanımı bu yıllarda University College in England'dan Fry ve Denes'in 4 sesli ve 9 sessiz harf tanıyan sistemleri ile ağırlık kazanmıştır [6]. Martin ve ekibi tarafından RCA Laboratuarları'nda ses sinyalindeki çeşitlilik (nonuniformity) sorununun önüne geçmek için zaman normalleştirmesi (time normalization) yöntemleri geliştirilmiştir. Bu yöntem ile tanıma istatistiklerini oldukça yükselten bir aşama olarak konuşmanın başlama ve bitiş noktalarının tespiti gerçekleşmiştir [7]. Aynı dönemde Sovyetler Birliği'nde Vintsyuk, konuşma tanımada "Time-Aligning of Speech Utterances" için olan Dinamik Programlama'yı (Dynamic Programming - DP) öne sürmüştür [8]. Bu çalışma Batı'da fazla yankı bulamadıysa da sonraki yıllarda Rabiner ve Myers tarafından geliştirilecek olan Dinamik Zaman Eğrilmesi (Dynamic Time Warping - DTW) ve Ney tarafından geliştirilecek bu algoritmanın sadeleştirilmiş versiyonu gibi uygulamaların öncüsü olarak kabul edilmiştir. Bu dönemin son satir başı ise Reddy tarafından fonemlerin dinamik takibinin (Dynamic Tracking of Phonemes) Sürekli Konuşma Tanıma (Continuous Speech Recognition - CSR) alanında kullanılması olmuştur. Bu uygulama daha sonra Carnegie Mellon Üniversitesi'nin bu alanda çok önemli bir başarısı olarak varlığını sürdürecektir.

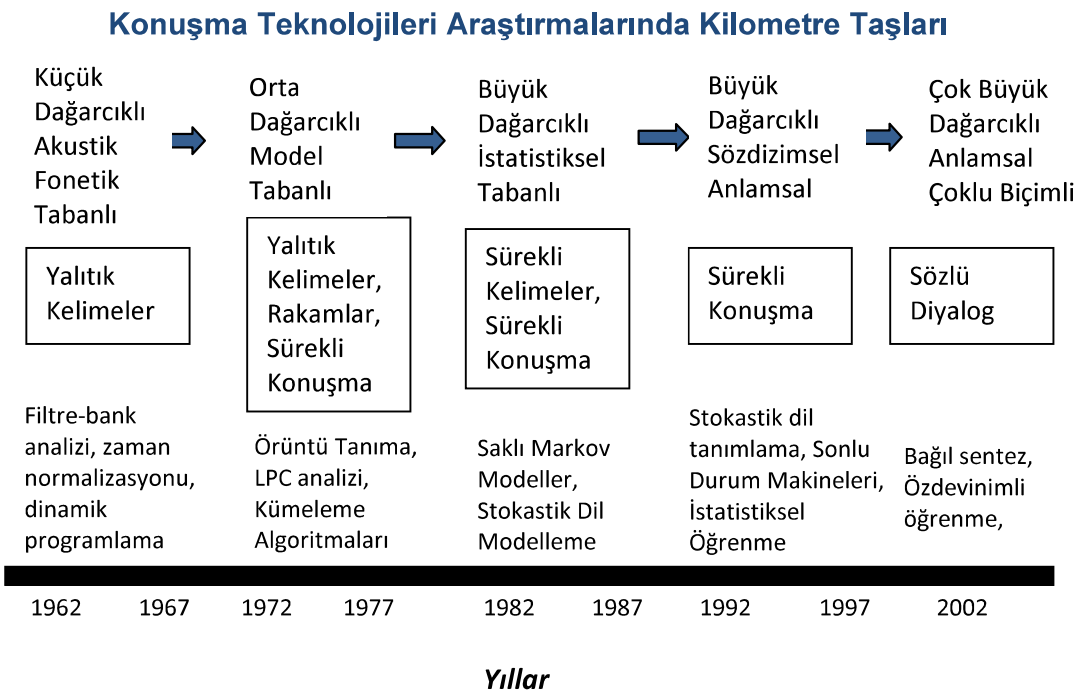
1970'li yıllarda konuşma tanımada birçok dönüm noktası yaşanmıştır. Elektronik ve bilgisayar teknolojilerindeki ilerlemeler, bilgisayarların güçlü işlemcileri bu alanda da ilerlemeyi beraberinde getirmiştir. Rusya'dan Velichko ve Zagoruyko'nun Örüntü Tanıma (Pattern Recognition) işlemini konuşma alanına katarken, Japonya'dan Sakoe ve Chiba dinamik programlamanın konuşma tanıma çalışmalarına başarılı şekilde uygulanabilirliğini göstermiştir. ABD'den Itakura ise düşük bit hızında konuşma kodlama uygulamalarında kullanılan "Linear Predictive Coding"'in (LPC) konuşma tanımada da uygulanabilirliğini göstermiştir.

Bu dönemde IBM, AT&T gibi birçok büyük firma araştırma grupları kurarak bu alanda deneyler yürütmeye başlamıştır. Konuşma tanımayı konuşmacı bağımsız hale getirmek, yalıtılmış kelime tanımanın yanı sıra sürekli kelime tanıma da bu dönemde araştırmaların en ön plandaki konularından biri olmuştur. Bunların yanı sıra bu yıllarda ABD Gelişmiş Savunma Araştırma Projeleri Kuruluşu'nun (DARPA) da konuşma tanıma konusunda araştırma-geliştirme çalışmalarına ağırlık vermesi bu alanda büyük bir atılımın gerçekleşmesine ön ayak olmuştur. Grafiksel arama konsepti bu dönemde yapılmış katkılarından biridir. Bu konseptte giriş sinyali parametrik analiz sonrası segmente edildikten sonra fonem kalıp eşleştirme yöntemi ile konuşma tanıma gerçekleşmiştir. Bu sistemin konuşma tanıma uygulamalarına etkisi 1990'li yılların sonunda Ağırlıklı Son Durum Ağı (Weighted Finite State Network - WFSN) kullanımını yaygınlaştırmak olacaktır.

Konuşma tanıma çalışmaları 1980'li yıllarda olan gelişmelerle yeni bir yöne sürüklenmiştir. Model tabanlı (template-based) çalışmalardan istatistiksel modelleme yöntemlerine geçiş ve Saklı Markov Modeli (Hidden Markov Model - HMM) yaklaşımı, bu konuda önemli bir aşama olmuştur. Markov Modelleri uzun yıllardır bilinmekte olsa da konuşma tanıma alanında uygulanması göreceli olarak yenidir ve fakat bugüne kadar kullanılmış bütün yöntemler arasında en efektif sonuçları vermektedir. 1980'li yıllarda öne sürülen bir diğer çalışma ise Yapay Sinir Ağları (Artificial Neural Networks - ANNs) uygulamalarını konuşma tanıma alanında kullanmak olmuştur. Geniş Dağarcıklı Sürekli Konuşma Tanıma (Large Vocabulary Continuous Speech Recognition - LVCSR) konusunda gelişmeler bu teknolojilerin ışığında sürdürülmüştür.

1990'li yıllarda ise örüntü tanıma konusunda yenilikçi gelişmeler yaşanmıştır. Bayes yöntemleri çerçevesinde ilerleyen örüntü tanıma teknikleri tanıma hatasını minimize eden bir optimizasyon problemine dönüşmüştür. En uyumlu tanıma sonucu yerine hata payı en düşük sonuca odaklanarak geliştirilen sistemler oldukça başarılı sonuçlar vermiştir.

Bütün bu gelişmelerin ışığında konuşma tanıma teknolojileri günlük hayatın bir parçası haline gelmiş ve bu alana verilen önem giderek artmıştır.



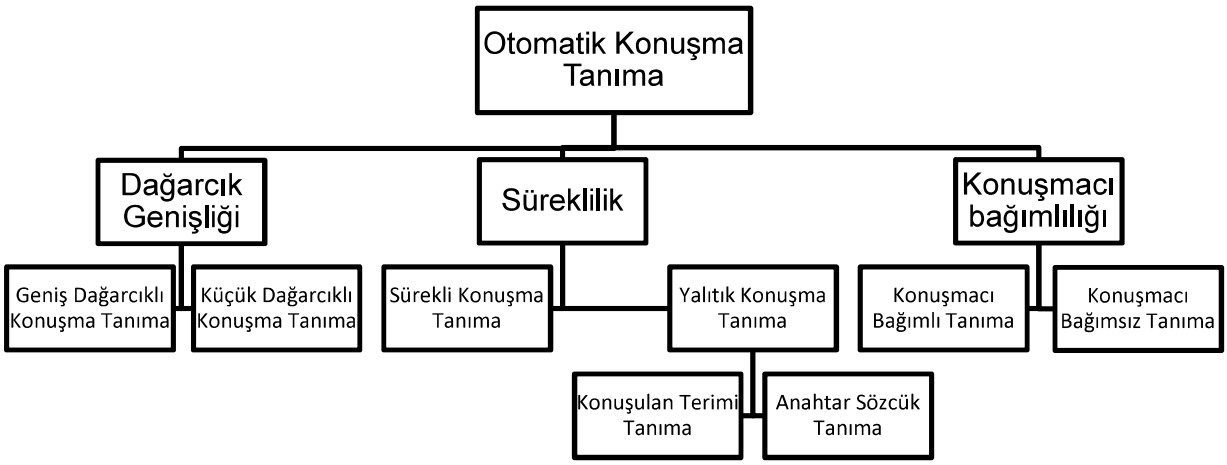
Şekil 1.1 Konuşma tanıma teknolojilerinin geçmiş 48 yıldaki gelişimi

2. OTOMATİK KONUŞMA TANIMA

Otomatik konuşma tanıma (Automatic Speech Recognition - ASR) verilen bir ses sinyalinin karşılık geldiği sözcüğü tanımak için en uygun istatistiksel, akustik ve dil modelleri arasında değerlendirme ve seçim yapılması olarak tariflenebilir.

Konuşma tanıma şu alt başlıklarda incelenebilir;

- Dağarcık genişliği: Geniş dağarcıklı (large vocabulary) ve küçük dağarcıklı (small vocabulary) sistemler
- Süreklilik ve yalıtıklık: Sürekli konuşma tanıma (continuous speech recognition) ve yalıtık konuşma tanıma (isolated speech recognition)
- Konuşmacı bağımlılığı: Konuşmacı bağımlı (speaker dependent) ve konuşmacı bağımsız (speaker independent) sistemler



Şekil 2.1 Konuşma tanıma çalışmalarının özelleşme alanları

Geniş dağarcıklı sistemler 5000-10000 kelime arası bir kelime haznesine sahip olup, çağrı merkezleri ve dikte gibi uygulamalarda tercih edilmektedirler. Küçük dağarcıklı sistemler 500 kelime civarında bir hazneye sahip olup, otomatik arama, sesli komut sistemleri gibi uygulamalarda kullanılmaktadırlar.

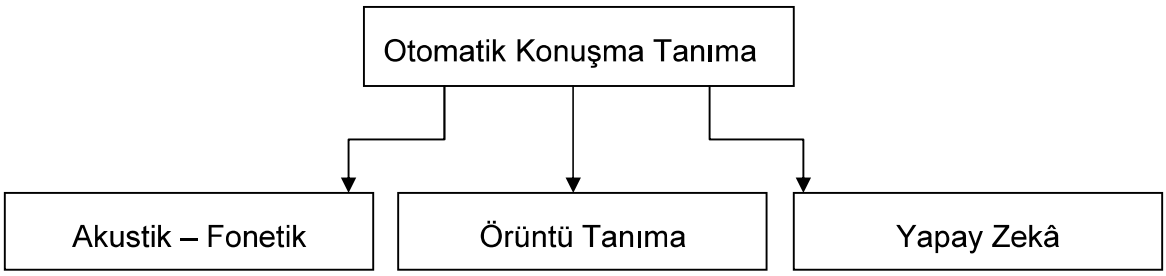
Sınırlı sayıda kullanıcı tarafından eğitilen sistemler, bu kullanıcıların konuşmalarını tanımaya daha eğilimli olup, farklı kullanıcılar için daha yüksek hata oranları verebilmektedirler.

Yalıtık konuşma sistemlerinin uygulama alanları arasında konuşulan sözcüğü tespit etmek ve bir ses kaydı içerisinden aranan anahtar kelimenin geçtiği segmentleri döndürmek üzere geliştirilmiş yöntemler vardır. İstihbarat uygulamaları ve sesli kayıt içerisinde sözcük arama gibi uygulamalarda kullanılırlar.

Bu tez kapsamında incelenecek konu, sesli bir metin içerisinden sözcüklerin tespitidir. Sözcük bulma olarak adlandırılan bu uygulamada, sürekli bir konuşma içerisinde, sorgu olarak sisteme verilen sözcükleri tespit etmek üzere geliştirilmiş yöntemler incelenmiş ve bunların test metinlerinde uygulamaları değerlendirilmiştir.

Konuşma tanıma alanında şu yaklaşımlar öne çıkmıştır.

1. Akustik – fonetik yaklaşım
2. Örüntü tanıma yaklaşımı
3. Yapay zekâ yaklaşımı



Şekil 2.2 Konuşma tanıma uygulamalarında farklı yöntemler

2.1 Otomatik Konuşma Tanımada Yöntemler

2.1.1 Akustik – fonetik yaklaşım

Bu yaklaşım şu önerme üzerinden geliştirilmiştir; konuşulan dilde, sonlu ve ayırt edici özelliklere sahip fonetik birimler bulunmaktadır ve bu birimler ses sinyalinin bir takım özellikleri ile karakterize edilmiştir. Her ne kadar konuşmacı bağımlılığı ve çevreden gelen ekstra gürültü bu özelleştirmeyi zorlaştırsa da bir takım kurallar çıkarmak ve bunları uygulamalarda kullanmak mümkündür. Bu nedenle bir takım yöntemler geliştirilmiştir. İlk adım segmentasyon ve etiketlemedir. İkinci adım, ilk adımda çıkarılmış etiketlerden anlamlı bir söz bütünü oluşturmaktır. Bu adımda oluşturulan sözcükler söz dizimsel (syntactic) ve anlamsal (semantic) açıdan bir anlam ifade ettiği durumda tanıma işlemi gerçekleştirilmiş olur. [1]

2.1.2 Örüntü tanıma yaklaşımı

Bu yaklaşımda ses sinyali önceki durumda olduğu gibi segmentlere ayrılmadan bir bütün olarak ele alınır. İki adımlı bir yöntem uygulanır. Önce konuşma örüntüleri eğitilir. Daha sonra örüntüler, örüntü karşılaştırma yöntemi ile tanınır. Burada eğitim ile bir veritabanı oluşturulur. Tanıma prensibi şudur: Yeterli miktarda örüntü tipi eğitim seti içerisinde verildiği takdirde örüntünün akustik özellikleri gerekli şekilde karakterize edilmiş olacağından gelen bilinmeyen konuşma örüntüler arası eşleştirme ile sorunsuz olarak tanınabilir. [1]

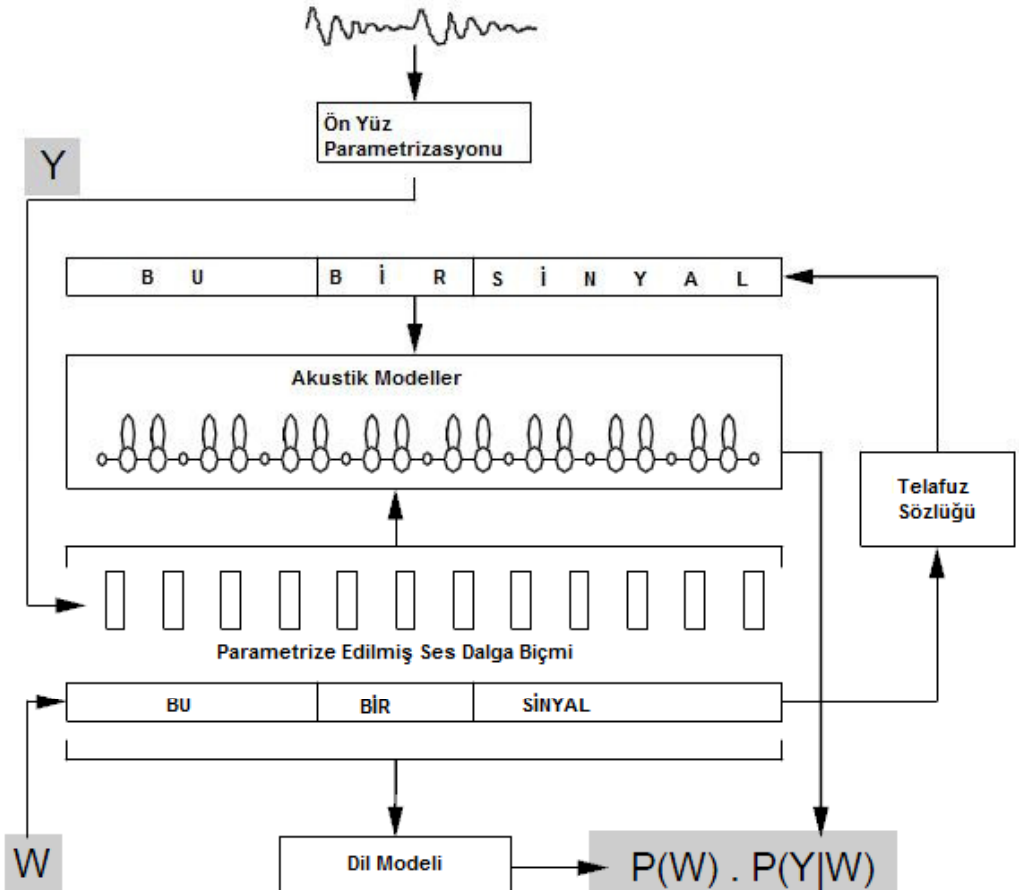
2.1.2 Yapay zekâ yaklaşımı

Yapay zekâ yaklaşımı yukarıda tariflenen iki uygulamanın bir karışımı olarak nitelendirilebilir. Eldeki akustik özellikler üzerinde bir insan zekâsının görme, analiz etme ve karar verme adımlarını mekanize etmek gibi bir yöntem izler. Yapay sinir ağları uygulamaları ile öğrenme ve yeni gelen veriye göre adapte olma gibi kavramlar da hayata geçirilmeye çalışılır. [1]

Tezin geri kalanında baz alınacak yöntem örüntü eşleme yöntemi olacaktır. Ve metot şu alt başlıklarla incelenecektir;

- Akustik model
- Dil modeli

Bu iki modelin birleşiminden oluşan sistem Şekil 2.3'te verilmiştir. Bu şekilde girdi olarak verilen ses sinyali ön yüz parametrizasyonu işleminden geçmektedir. Bu işlemin detayları 2.1.1 bölümünde verilmiştir. Şekilde Y olarak ifade edilen bu bileşen konuşmanın akustik özelliklerini temsil etmektedir. Girdi ses sinyalinin yazılı karşılığı olan "bu bir sinyal" transkripsiyonu sistemin W bileşenini oluşturmaktadır. Dil modeli oluşturmada değerlendirilen bu cümle için gerçekleştirilen işlemler bölüm 2.3'te detaylarıyla açıklanacak olan $P(W)$ yani dil modeli olasılığını sisteme katmaktadır. Bölüm 2.1.2'de açıklanan HMM modellemesi ile oluşturulan akustik modeller, verilen sözcüğün o model ile ortaya çıkma olasılığı olan $P(Y|W)$ bileşenini oluşturmaktadır. Bu şekilde gösterilen $P(W)P(Y|W)$ ifadesi, Y akustik sinyali verildiğinde, en olası W sözcük dizilimini bulma durumunu ve konuşma tanıma uygulamalarında temel olarak kabul edilen bir olasılığı ifade etmektedir.



Şekil 2.3 Akustik model ve dil modelinin oluşturduğu tanıma sistemi

2.2 Akustik Model

Şekil 2.3'teki tasarımı gerçekleştirmede ilk adım bir akustik model oluşturmaktır. Burada akustik model oluştururken konuşmacı, içerik ve çevresel etken çeşitliliğin olduğu verileri kullanmak daha dengeli ve sağlam bir model dağılımı sağlayacaktır. İnsan algısının duyarlılığına ulaşmaktan uzak olsa dahi, modeller eğitim verisi artırılarak yüksek başarı sağlar duruma getirilebilmektedir.

2.2.1 Ön yüz parametrizasyonu ve özellik çıkarma

Zaman ve frekans bileşenleriyle ifade edilebilen ses sinyali 10 ms'lik birimlerde durağan özellik göstermektedir. Bu nedenle girdi sinyali 10 ms'lik pencerelere bölünmektedir. Ancak sesin çabuk değişiklik gösteren özelliklerini de kaçırmamak için çerçeveler örtüşen bir sıklıkla oluşturulmaktadır. Burada genellikle 10ms'lik örtüşmelerle 25 ms'lik aralıklı pencereleme gerçekleştirilir. Her pencere için bir pencereleme (windowing) fonksiyonu uygulanır. Genellikle Hamming fonksiyonu tercih edilir. Bu sayede keskin geçişler ve çerçeveler arası oluşabilecek boşluklar yumuşatılmış olur.

Ön yüz parametrizasyonunda çeşitli özellikler kullanılabilmektedir. Bunlardan bazıları doğrusal tahminsel katsayılar (Linear Predictive Coefficients - LPC), mel frekans kepstrum katsayıları (Mel Frequency Cepstrum Coefficients - MFCC) ve LP tabanlı kepstrum katsayılarıdır. Tez çalışmaları kapsamında kullanılan HTK aracı, yöntem olarak MFCC özelliklerinin çıkarılmasını kullanmaktadır. [24]

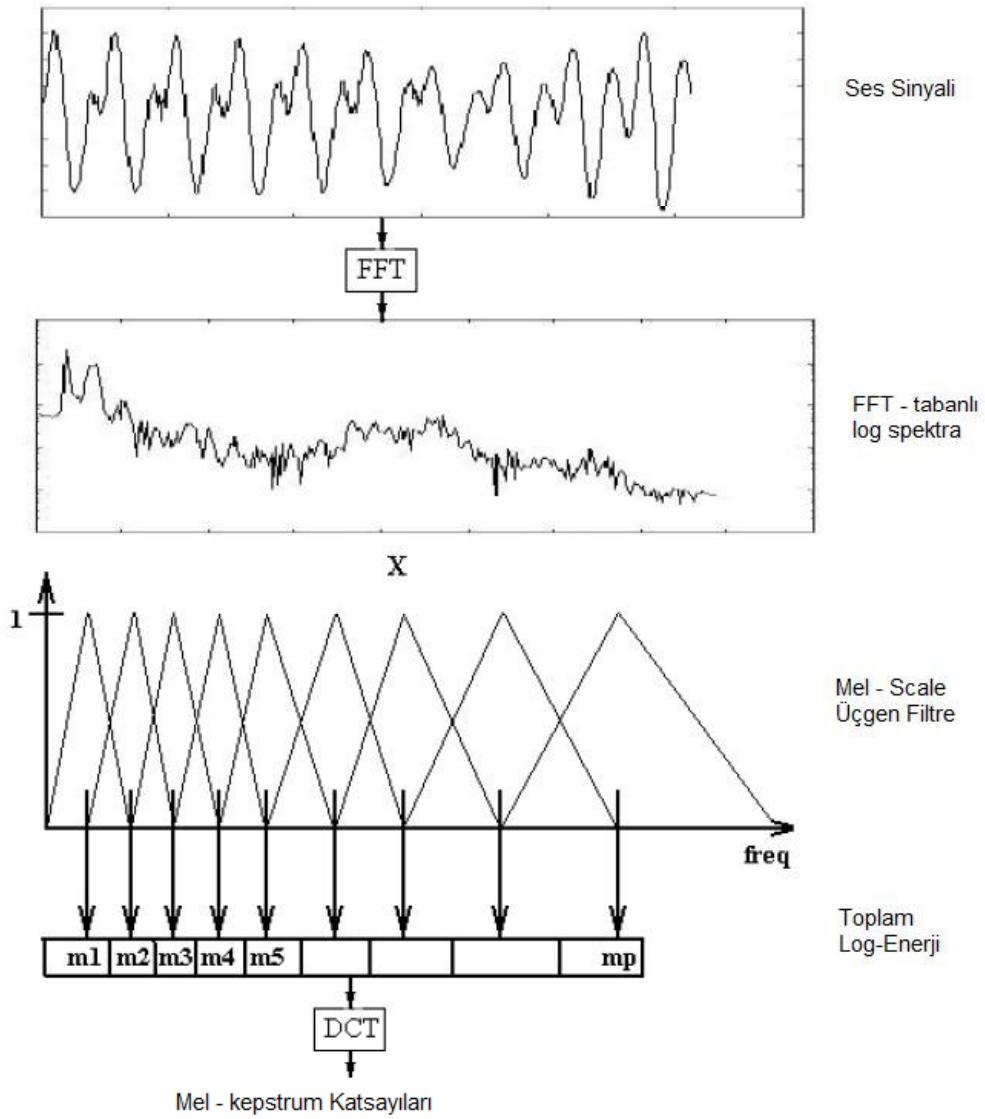
İmge analizinde piksel değerlerini simgeleyen özellikler, metin analizinde ilgili sözcük ile karşılaşma sıklığını gösterebilmektedir. Konuşma işlemede ise mel frekans kepstrum (MFC) olarak adlandırılan özellikler, ses sinyalinin, frekansın doğrusal olmayan mel ölçeğinin logaritmik güç spektrumunun doğrusal kosinüs dönüşümünün kısa süreli güç spektrumudur (short-term power spectrum).

MFCC olarak adlandırılan Mel Frekans Kepstrum Katsayıları, MFC'yi oluşturan değerlerin toplamıdır. Bunlar bir ses kaydının kepstral gösteriminden elde edilirler.

Mel ölçeği, düşük frekans aralığında (<1000Hz) lineer, yüksek frekansta (>1000Hz) ise logaritmik özellik gösteren bir ölçektir. Bu da insanın işitme sistemi benzeri bir yapıdır. MFCC'ler şu şekilde türetilmiştir;

1. Pencereleme fonksiyonu ile uygun genişlikteki çerçeveler alınır.
2. Bu çerçeve içerisindeki sinyalin Fourier Dönüşümü alınarak spektral domain'e geçilir.
3. Bu spektrumun logaritması alınarak log ölçeğine geçilir.

4. Üst üste binen üçgen pencereler ile önceki adımdaki log spektrumun güç değeri çıkarılır.
5. Mel kepstral katsayıların hesaplanması için ters Fourier Dönüşümü alınmak istenir fakat dizinin simetrik özelliğe sahip olması nedeni ile kesikli kosinüs dönüşümü hesaplanır.
6. Bu sayede MFCC'ler ortaya çıkmış olan spektrumun genlik değeri olarak bulunur.



Şekil 2.4 MFCC tabanlı örüntü eşleyici

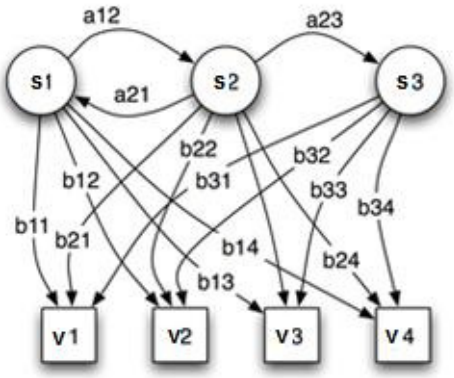
Örüntü tanımada özellik vektörü, bir objenin sayısal özellikleri gösteren n-boyutlu bir vektördür. Örüntü tanımanın yanı sıra yine otomatik konuşma tanıma konusunun alt başlıklarından biri olan özdevinimli öğrenme (machine learning) uygulamalarındaki birçok algoritmada da kullanılan özellik vektörleri istatistiksel analiz yapmayı sağlar ve gerçekleştirilecek işlemleri kolaylaştırır.

2.2.2 Saklı Markov modeller

Konuşma tanıma alanında geçmiş yıllarda Dinamik Programlama (DP), Dinamik Zaman Eğrilmesi (DTW), Saklı Markov Modelleri (HMMs) ve Yapay Sinir Ağları (ANN) gibi çeşitli yöntemler kullanılmıştır. Bunlar arasında günümüz uygulamalarında en çok tercih edilen yöntem HMM yöntemidir.

Saklı Markov Modelleri, mevcut durumları bilinmeyen fakat çıktıları gözlemlenebilen istatistiksel modellerdir. Burada her durumun olasılıksal bir dağılımı mevcuttur. Bu sayede ortaya çıkan sıralı çıktılar, durumların o sıradaki değerleri ile ilgili bilgi taşırlar.

Aşağıdaki sekil HMM modellerin olasılıksal durumlarının bir ifadesidir;



Şekil 2.5 HMM modeli için olasılık şeması

Burada s ile tanımlanan yuvarlaklar durumu, v ile belirtilen kareler çıktıyı, a ile tanımlanan oklar mevcut durumdan bir diğerine geçişi, b ile tanımlanan oklar ise verilen durumdaki çıktı olasılıklarını temsil etmektedir.

HMM modeller bir Sonlu Durum Makinesi (Finite State Machine - FSM) olarak ifade edilebilir. Model her kesikli t anında bir durumdan diğerine geçiş yapar. Durum

$$S = \{s_1, s_2 \dots s_N\} \tag{2.1}$$

ile ifade edilir. Burada N durum sayısını ifade etmektedir.

Bu geçişin sonunda yeni bir çıktı elde edilir. Bu çıktıya gözlem denmektedir. Gözlem;

$$V = \{v_1 , v_2 \dots v_M\} \tag{2.2}$$

ile ifade edilir. Burada ise M gözlem sayısını ifade etmektedir.

Yukarıda bahsedilen bir durumdan diğerine geçiş, Durum Geçiş Matrisi (State Transition Matrix) olarak adlandırılan $A = \{a_{ij}\}$ matrisi ile gösterilmektedir.

$$a_{ij} = P\,(q_t+1 = j \mid q_t = i)\ 1 \le i, j \le N \tag{2.3}$$

q_t t anındaki durumu, $q_t + 1$ ise t + 1 anındaki durumu ifade etmektedir. Eğer j durumu i durumundan geçişin gerçekleşemeyeceği bir durum ise $a_{ij} = 0$'dir.

Gözlemlenen sembollerin olasılıksal dağılım setini gösteren $B = \{b_j(o_t)\}$ matrisi ise su şekilde ifade edilmektedir.

$$b_j(o_t) = P\,(o_t \mid q_t = j);\ 1 \le t \le T \tag{2.4}$$

Bu denklem j durumunda, gözlemlenen çıktı değerlerini vermektedir.

Yapılan gözlemlerde başlangıç durumu $\pi = \{\pi_i\}$ ile ifade edilir. Başlangıç durum dağılım notasyonu su şekildedir;

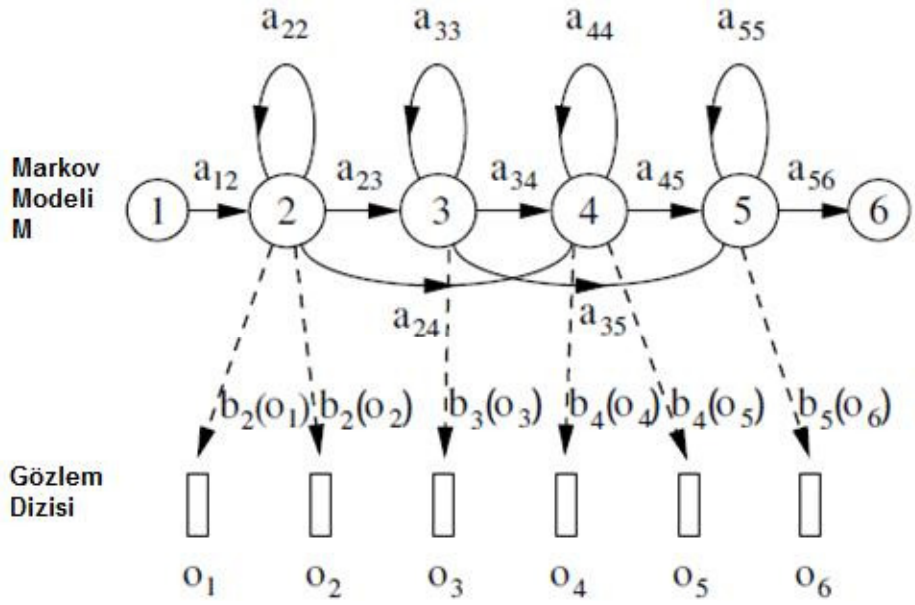
$$\pi_i = P\,(q_1 = i);\ 1 \le i \le N \tag{2.5}$$

Durum ve gözlem dizilerini oluşturmak bize olasılık hesaplarında gerekli olacaktır. Gözlem dizisi $O = (o_1, o_2 \dots o_T)$ ve durum dizisi de $q = (q_1, q_2 \dots q_T)$ şeklinde ifade edilebilir.

Yukarıdaki parametreler setini tek bir denklem ile ifade etmek istersek, şu denklemi kullanabiliriz;

$$\lambda = (A, B, \pi) \tag{2.6}$$

Bu parametre modeli verilen bir O gözlem seti için olasılık hesabında P(O|λ) notasyon kolaylığı sağlamaktadır.



Şekil 2.6 Durum geçiş ve gözlem olasılıkları şeması

Konuşma tanıma uygulamalarında HMM modeller ile çözüm getirilmek istenen 3 problem vardır. Bunlar;

1. Verilmiş bir gözlem dizisi O = (o_1, o_2 … o_T) ve bilinen parametre modeli λ = (A, B, π) varken bu gözlem setinin elde edilme olasılığı *P*(O|λ)'nin nasıl hesaplanacağı.
2. Yine verilmiş bir gözlem seti O = (o_1, o_2 … o_T) ve bilinen parametre modeli λ = (A, B, π) varken bu gözlemlerin elde edilebilmesi için oluşmuş olması gereken optimal durum seti q = (q_1, q_2 … q_T)'in nasıl belirleneceği
3. *P*(O|λ) olasılığını maksimize edecek model parametrelerinin nasıl ayarlanacağıdır.

Bu problemlerin çözümlerinde şu yöntemler kullanılmıştır; [10]

2.2.2.1 Forward-Backward algoritması

Birinci problem verilmiş modelin yapılan gözlemlerle uyumluluğunu da ölçmektedir ve mevcut gözlemler için en uygun modelin seçilmesinde de işe yarar. Çözümü şu şekilde ele alınmıştır;

T sayıdaki bütün durum dizilerini numaralandırabiliriz. Bir tanesi aşağıdaki dizi olsun;

$$q = (q_1 q_2 \ldots . q_T) \tag{2.7}$$

Burada q_1 başlangıç durumunu göstermektedir. Yukarıda durum denklemi verilen gözlem dizisinin olasılığı şu şekildedir;

$$P(\mathrm{O} \mid \mathrm{q}, \lambda) = \prod_{t=1}^{T} P(o_t | q_t, \lambda) \tag{2.8}$$

değerlerin yerine konması ile yukarıdaki denklem aşağıdaki şekle dönüştü;

$$P(O \mid q, \lambda) = bq_1(o_1).bq_2(o_2) \ldots bq_T(o_T) \tag{2.9}$$

q durum denkleminin değeri de yerine yazıldığı durumda

$$P(q \mid \lambda) = \pi\, q_1 a\, q_1 q_2 a\, q_2 q_3 \ldots a q_T - 2 q_T \tag{2.10}$$

O ile q'nun birleşik olasılıkları yani, ikisinin ayni anda oluşma olasılığı yukarıdaki ifadelerin çarpımı ile elde edilir.

$$P(O, q \mid \lambda) = P(O \mid q, \lambda).P(q \mid \lambda) \tag{2.11}$$

O gözlem setinin olasılığı ise yukarıdaki ifadenin bütün olası q durum dizileri üzerinden toplamı ile ifade edilebilir.

$$P(\mathrm{O} \mid \lambda) = \sum_{\text{all q}} P(\mathrm{O} \mid \mathrm{q}, \lambda).P(\mathrm{q} \mid \lambda) \tag{2.13}$$

$$= \sum_{\mathrm{q1}, \mathrm{q_2} \ldots \mathrm{q_t}} \pi\, \mathrm{q_1}\, \mathrm{bq_1}(\mathrm{o_1}) \mathrm{aq_1 q_2}\, \mathrm{bq_2}(\mathrm{o_2}) \ldots\, \mathrm{a_{qT-1qT}}\, \mathrm{bq_T}(\mathrm{o_T}))$$

Burada t = 1 zamanında q_1 durumunda π q_1 olasılığı ile bulunuyoruz ve o_1 sembolünü b_{q1} (o_1) olasılığı ile üretiyoruz. Zaman t'den t + 1'e geçtiğinde q_1 durumundan q_2 durumuna a_{q1q2} olasılığı ile geçmiş olup o_2 sembolünü b_{q2} (o_2) olasılığı ile üretiyoruz. Bu işlem son geçişi T zamanında q_{T-1} 'den q_T durumuna gerçekleştirip o_T olasılığını üretene kadar tekrarlıyor. Ancak bu olasılığı hesaplamak $2TN^T$ işlem gerektireceğinden bu hesaplamada daha sadeleştirilmiş bir yöntem olan Forward-Backward Procedure kullanılmaktadır.

Forward Procedure şu şekilde ifade edilmektedir;

$$\alpha t(i) = P\,(o_1 o_2 \ldots o_t, q_t = i \mid \lambda) \tag{2.14}$$

Bu denklem kısmi gözlem dizisinin i durumunda ve t zamanında verilen λ modeli ile ortaya çıkma olasılığıdır. Su şekilde çözülebilir;

1. Başlangıç durumu

$$\alpha_1 = P\,\pi_i\, b_i\,(o_1) \tag{2.15}$$

2. Tümevarım

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \left(\alpha_t(i)\alpha_{ij}\right)\right] b_j(o_{t+1}) \tag{2.16}$$

3. Bitiş durumu

$$P\,(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{2.17}$$

Bu yinelemeli çözümden de görüleceği üzere *P* (O | λ) olasılığının hesaplanması N^2T işlem gerektirmektedir.

Backward Procedure ise şu şekilde ifade edilebilir;

$$\beta t\,(i) = P\,(o_{t+1} o_{t+2} \ldots o_T \mid q_t = i, \lambda) \tag{2.18}$$

Bu denklem kısmi gözlem dizisinin i durumunda t zamanında ve λ modeli ile ortaya çıkma olasılığıdır. Tümevarımsal olarak şu şekilde çözülebilir;

1. Başlangıç durumu

$$\beta_t(i) = 1, \tag{2.19}$$

2. Tümevarım

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \tag{2.20}$$

2.2.2.2 Viterbi algoritması

İkinci problem optimal durum dizisinin tahmin edilmesi konusundadır. Birinci yöntemdeki tek bir çözümün aksine verilmiş gözlem dizisi ile durum dizisinin tahmin etmenin çeşitli yolları vardır. Birçok Optimality Criteria tanımlanabilir. Örneğin tekil olarak en yüksek olasılığa sahip q değerlerini seçmek bir yöntemdir. Burada tercih edilecek yöntem $P(q \mid O, \lambda)$ değerini maksimize edecek durum dizisini bulmak olacak. Bu Bayes' Rule'a [10] göre $P(q, O \mid \lambda)$ olasılığını maksimize etmek demek olur. Çözüm bir çeşit dinamik programla yöntemi olan Viterbi algoritması ile elde edilir. [18] Viterbi değişkeni $\delta_t(i)$ şu şekilde tanımlanır;

$$\delta_t(i) = P(q_t = i \mid O, \lambda) \tag{2.21}$$

Burada $q = (q_1 q_2 \ldots q_T)$ verilen $O = (o_1 o_2 \ldots o_T)$ dizisi için en yüksek olasılığa sahip durum dizisidir. $\delta_t(i)$'nin özyinelemeli versiyonu şu şekilde yazılabilir;

$$\delta_t(i) = \max_{q_1, q_2 \ldots q_{t-1}} P\left[q_1 q_2 \ldots q_{t-1}, q_t = i, o_1 o_2 \ldots o_t \mid \lambda\right] \tag{2.22}$$

1. Başlangıç durumu

$$\delta t(i) = \pi i \; bt\,(o1) \tag{2.23}$$

$$\psi_1(i) = 0 \tag{2.24}$$

2. Özyineleme

$$\delta t(j) = \max_{1<i<N_1}[\delta_{t-1}(i)a_{ij}]b_j(o_t) \quad (2.25)$$

$$\psi_t(j) = arg\max_{1<i<N_1}[\delta_{t-1}(i)a_{ij}] \quad (2.26)$$

3. Bitiş

$$P^* = \max_{1<i<N}[\delta T(i)\,] \quad (2.27)$$

$$q^*_T = arg\max_{1<i<N_1}[\delta_T(i)] \quad (2.28)$$

4. Yol (durum dizisi) gerileme

$$q^*t = \psi_t + 1\,(q^*t + 1) \quad (2.29)$$

Forward Procedure'den farkı, oradaki toplama adımı yerine Viterbi Algoritması'nda maksimizasyon kriterinin kullanılmasıdır.

2.2.2.3 Baum – Welch algoritması

Üçüncü ve en zor problem ise parametre tahmini üzerinedir. A, B, ve π model parametreleri bir optimizasyon kriteri sağlamak üzere tahmin edilmektedir. Optimizasyon kriteri ise $P(O \mid \lambda)$'yu maksimize etmek üzerine geliştirilmektedir. Bunun için kullanılan yöntem Baum-Welch metodu olarak adlandırılmıştır [17].

Bu yöntemde gözlem sembolü olasılık dağılımı B = {bj(k)}'nin modellenmesi bağlı karışım modellemesi (tied mixture modelling- semicontinuous olarak da adlandırılır) ile gerçekleştirilmiştir. Bu modelde bütün akustik uzay, Sürekli Yoğunluk Kod kitabı (Continuous Density Codebook) ile ifade edilmiştir. Akustik uzay bağımsız Gaussian yoğunluk ifadeleri olarak ele alınmıştır. Çıktı ortalamaları (mean) ve kovaryansları (covariances) bir kod kitabına kaydedilmiştir [20]. Her tanıma birimindeki olasılık yoğunluk fonksiyonu kod kitabında bulunan yoğunlukların bir karışımı olarak ifade edilmiştir. Bu yöntem yoğunlukların olasılıklarını modellerken işlem yoğunluğunu düşük tutabilmiştir.

Öncelikle verilmiş bir t zamanında S_i durumunda ve t+1 zamanında S_j durumunda olma olasılığını ele alalım;

$$\xi t\,(i,j) \;=\; P\,(q_t \;=\; S_i\,, q_{t+1} \;=\; S_j \mid O, \lambda) \tag{2.30}$$

Bir önceki bölümde açıklanmış Forward-Backward değişkenler kullanılarak bu formül şu şekilde de ifade edilebilir;

$$\xi t(i,j) = \frac{\alpha_t(i)\alpha_{ij}\beta_{t+1}(j)b_j(o_{t+1})}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)\alpha_{ij}\beta_{t+1}(j)b_j(o_{t+1})} \tag{2.31}$$

İkinci değişken posteriori olaslık olarak adlandırılan

$$\gamma = \sum_{j=1}^{N}\xi t(i,j) \tag{2.32}$$

değişkenidir.

Baum-Welch'te gerçekleştirilcek bir sonraki adım olan *P*(O | λ) maksimizasyonu ile HMM modellerin güncellenmesi için yukarıdaki iki değişkeni (2.31) ve (2.32) kullanılabilir. Yeni a ve b değerleri şu şekilde hesaplanmaktadır;

$$\mathrm{a}'_{\mathrm{ij}} = \frac{\sum_{\mathrm{t-1}}^{\mathrm{T}}\xi_{\mathrm{t}}(\mathrm{i,j})}{\sum_{\mathrm{t-1}}^{\mathrm{T}}\gamma_{\mathrm{t}}(\mathrm{i})} \tag{2.33}$$

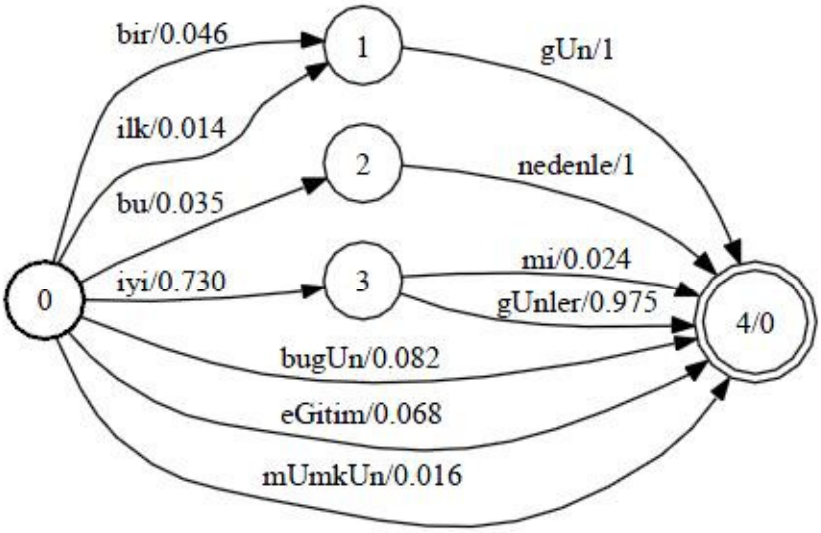
$$\mathrm{b}'_{\mathrm{ij}} = \frac{\sum_{\mathrm{t-1}}^{\mathrm{T}}\gamma_{\mathrm{t}}(\mathrm{i,j})}{\sum_{\mathrm{t-1}}^{\mathrm{T}}\gamma_{\mathrm{t}}(\mathrm{i})} \tag{2.34}$$

2.2.3 Ağırlıklı sonlu durum çeviricileri

Ağırlıklı Son Durum Çeviricileri (Weighted Finite State Transducers – WFST) konuşma tanımada HMM modellerinde, seslendirme sözlükleri ve alternatif tanıma çıktılarında açıklayıcı bir gösterim sağlamaktadır. Ağırlıklı determinizasyon ve minimizasyon algoritmaları için optimal bir zaman ve yer düzeni sağlayarak konuşma tanıma çıktı dağılımlarını düzenlemektedir.

Otomatik konuşma tanıma sistemleri, kafes formatında çıktı veren tanıma sonuçları içinde bir arama gerçekleştirmektedir. Burada en iyi tek sonuç (one-best) çıktısı, kafesin düğümlerinin ağırlık değerlerinden yapılan hesaplamalar ile elde edilebilir. Aramayı hızlandırmak için bütün düğümlere indeksler verilir.

"İyi günler" söz dizimi için örnek bir tanıma çıktı kafesi şu şekilde gösterilmektedir;



Şekil 2.7 Örnek bir tanıma kafesi

Burada 0 durumunda tanıma işlemi başlar. Bulunan kelimeler, kollarda transkripsiyonları ve ağırlıklı olasılık değerleri ile ifade edilirler. 0 düğümünden çıkan kollarda en yüksek olasılığın "iyi" kelimesine ait olduğu görülmektedir. Diğer olasılıklar, "one-best" tanıma işlemine değer katamayacak derecede küçüktürler. 3. düğüm, tekrar iki kola ayrılır. Burada ise "günler" kelimesi daha yüksek bir değere sahiptir. Bu şekilde elde edilen bir tanıma çıktısı "iyi günler" tanıma sonucunu döndürmektedir.

Ağırlıklı son durum çeviricileri ve konuşma tanıma uygulamalarındaki çalışmalar hakkında daha detaylı bilgi için [19] incelenebilir.

2.3 Dil Modeli

Dil modeli oluşturmak, söz dizilerinin ardışık olasılıklarını belirlemektir. Dil modeli konuşma tanıma, el yazısı tanıma, makine çevrimi ve imla düzenleme gibi çeşitli alanlarda işe yaramaktadır. En yaygın olarak kullanılan dil modelleri "Katz-smoothed trigram model" gibi sade modellerdir. Fakat bu basit modeller üzerinde çeşitli geliştirmeler uygulanmıştır. Bu geliştirmeler "caching", "clustering", "higher-order *n*-grams", "skipping models" ve "sentence-mixture models"'i içermektedir. Bütün bunlara aşağıda kısaca değinilecektedir. Daha karmaşık modeller üzerinde fazla durulmamış, herhangi bir geliştirme gerçekleştirilmemiştir.

Dil modellemede söz konusu olan bir çelişki şudur: Birbirinden bağımsız olarak yüksek başarı yüzdesiyle sonuç veren iki ayrı model bir arada kullanıldığında artan bir başarı göstermeyebilir. Bazı modeller ise birleştirildiğinde daha iyi performans göstermektedir.

Dil modeli oluşturmanın hedefi olan $\mathrm{w1}\ .\ .\ .\mathrm{w}n$, söz dizilimlerinin görülme olasılığı $P(\mathrm{w1}\ .\ .\ .\mathrm{w}n)$'nin hesaplaması şu şekilde yapılmaktadır;

$$P(\mathrm{w1}\ .\ .\ .\mathrm{w}i) = P(\mathrm{w1}) \times P(\mathrm{w2}|\mathrm{w1}) \times \cdots \times P(\mathrm{w}i\,|\mathrm{w1}\ .\ .\ .\mathrm{w}i{-}1)$$

Burada ilk kelimeden sonra gelen her kelimenin olasılığı, kendinden önce gelen kelimeye bağlı olarak hesaplanmaktadır. Fakat uzun sözcük dizilimleri için $P(\mathrm{w}i\,|\mathrm{w1}\ .\ .\ .\mathrm{w}i{-}1)$ olasılığını hesaplamak oldukça karmaşık bir işlem olacağından, i değerini optimal bir değerde sabitlenmektedir. Bu modele n-gram dil modeli denilir. Buradaki n yani i gerçek yaşamdaki birçok uygulamada 3 olarak tercih edilmektedir. Bunun sebebi daha yüksek dereceli modellerin işlem karmaşıklığını artırması ve daha düşük dereceli modellerin ise istatistiksel anlamda yeterli bilgiyi içermemesidir [9]. Modelin ismi trigram dil modeli olarak verilmiştir. Kendinden önceki 2 kelimenin olasılık değerlerine bağlı olan bu olasılık şu şekilde ifade edilmektedir;

$$P(\mathrm{w}_i\,|\mathrm{w}_1\ .\ .\ .\mathrm{w}_{i-1}) = P(\mathrm{w}_i\,|\mathrm{w}_{i-2}\mathrm{w}_{i-1})$$

Dil modeli oluşturmada Türkçe sondan eklemeli yapısı ile dezavantajlı bir dildir. Yeterince geniş olmayan bir kelime sözlüğü, aynı kökten türemiş bir kelimenin birçok farklı varyasyonunu kaçıracaktır. Sözlük dışı (out of vocabulary) olarak adlandırılan bu kelimeler tanıma sonuçlarını olumsuz etkileyecektir. Örneğin aynı büyüklükteki Türkçe ve İngilizce kelime sözlükleri üzerinden gerçekleştirilen bir tanımada Türkçe'de sözlük dışı kelime oranı %10 iken İngilizce'de %1 ile sınırlı kalabilmektedir. Bu durumun önüne geçmek için izlenen bir yol, dil modeli oluştururken kelimenin alt birimlerini değerlendirmek olur. Dilbilimsel ve istatistiksel açıdan kelime şu alt birimler olarak incelenebilir;

- Fonem
- Hece
- Morfem
- Grammatik kök
- İstatistiksel kök

2.3.1 Dil modelleme teknikleri

Dil modeli oluşturmadaki temel amaç $w_i w_{i+1 \ldots} w_n$ sözcük diziliminin olasılığı olan $P(w_i w_{i+1 \ldots} w_n)$'yi hesaplamaktır. Bu alanda yapılan çalışmalar arka arkaya gelen bütün sözcükleri ele almak yerine 3 adet sözcüğün optimal sonuç verdiğini göstermiştir. [9] Trigram dil modeli olasılıkları ise sözcüklerin eğitim korpusundaki sayılarından yola çıkılarak hesaplanmaktadır. $C(w_{i-2} w_{i-1} w_i)$ ve $C(w_{i-2} w_{i-1})$ değerleri $w_{i-2} w_{i-1} w_i$ sözcüklerinin eğitim korpusunda geçme sayısını ifade ediyor. Buradan şu çıkarım yapılabilir,

$$P(w_i \mid w_{i-2} w_{i-1}) \approx \frac{C(w_{i-2} w_{i-1} w_i)}{C(w_{i-2} w_{i-1})}$$

Burada w_i kelimesinin trigram olasılığını ifade eden $P(w_i \mid w_{i-2} w_{i-1})$, kabaca bu kelimelerin eğitim korpusunda geçme sayısının kendisinden önce gelen iki kelimenin geçme sayısına oranı olarak ifade edilmektedir.

Genel anlamda bu modelin verimsiz olduğu bir konu vardır: Eğitim korpusunda hiç geçmemiş olabilecek birçok ardışık kelime dizisi, aslında kullanım esnasında birçok kere tekrarlayabilir. Örneğin "yarın akşam yemekte" tamlaması eğitim korpusunda hiç geçmemiş olabilir. Bu sebeple *C(yemekte|yarın akşam)* sayısı 0 olabilir. Fakat "yarın akşam" tamlaması farklı birçok şekilde kullanılabilir. *P(yarın akşam yemekte)* = 0 olasılığa sahip olması, bu tamlamayı içerebilecek başka kelime dizilimlerinin de azımsanmasına sebep olur. Hatta dil modelinden gelen katsayının 0 olması durumunda, akustik model olasılığı da değer kaybedeceğinden, durum veri kaybı ile sonuçlanabilir. Bu sorunun önüne geçmek için çeşitli yöntemler geliştirilmiştir.

Aşağıda açıklanacak olan yöntemler ve uygulamaları hakkında daha detaylı bilgi [9]'dan edinilebilir.

2.3.1.1 Yumuşatma metodu

Sıfır olasılık probleminin önüne geçmek için kullanılan metotlardan biri yumuşatma (smoothing) metodudur. Bunun için örneğin "yemekte" kelimesinin olasılığı benzer kalıpta kullanılabilecek "televizyonda" gibi alternatif sözcükler üzerinde yumuşatılarak dağıtılabilir. Bu konuda yapılan çalışmalar arasında, şu teknikler öne çıkmıştır; Katz smoothing ve Jelinek–Mercer smoothing (bu teknik aynı zamanda silinen eklenti olarak da adlandırılabilir). Bilinen en verimli yumuşatma tekniği ise Kneser–Ney yumuşatma tekniğidir. Trigram modellerin derecesini arttırmak ile yumuşatma işlemi arasında bir bağlantı görülmüştür. 4-gram ya da 5-gram uygulamalarda da Kneser-Ney tekniğinin, Katz'a kıyasla yüksek bir performans artırımı yaşadığı ispatlanmıştır. [9]

Sıfır olasılık sorununun önüne geçmek için yumuşatma metodu ile geliştirilmiş ve türevleri arasında önce çıkmış bir algoritma da "back-off smoothing" olarak adlandırılmıştır. Bu yöntemde olasılığı sıfır olarak belirlenmiş n-gramlara, uni-gram olasılıklarıyla bağlantılı olarak sıfır olmayan bir olasılık atanmıştır.

2.3.1.2 Atlama metodu

Uygulanan başka bir yöntem atlama (skipping) yöntemi olarak adlandırılır. Bu yöntemde kelimeden önce gelen iki kelime yerine farklı bir bağlam değerlendirilir. Örneğin $P(w_i | w_{i-2} w_{i-1})$, değerini hesaplamak yerine $P(w_i | w_{i-3} w_{i-2})$ değeri hesaplanır. Bu model tek başına çok iyi çalışmazken, standart modelle birleştirildiğinde iyileşmeler sağladığı görülmüştür.

2.3.1.3 Kümeleme metodu

Kümeleme (clustering) modeli kelimeler arasındaki benzerliklerden yola çıkarak olasılıkları dağıtmak üzere kurulmuştur. Örneğin "yarın akşam yemekte" ve "yarın akşam televizyonda" tamlamalarından sonra gelebilecek "yarın akşam toplantıda" tamlaması da "yarın akşam" kalıbının ardından gelmesi ve içerik açısından benzerlik göstermesi sebebiyle bir katsayıya sahip olur.

2.3.1.4 Önbellek metodu

Önbelleğe alma (caching) modeli daha önceden kullanılan bir kelimenin tekrar kullanılması ihtimali üzerinde durarak buna göre değerlendirme yaparlar. Uygulaması kolay olan bu model sözcük-hata olasılığına (word-error rate) büyük iyileştirme getirmezler. Bu modeller için trigram dil modeli unigram dil modeline kıyasla oldukça büyük fayda sağlar.

2.3.1.5 Cümle karışımı metodu

İncelenen son model cümle karışımları (sentence mixture) modelidir. Bu model cümle tiplerini tek bir tip olarak kabul etmek yerine farklı şekillerde modellerler. Bugüne kadar geliştirilen uygulamalarda ortalama 4-8 farklı cümle modeli kullanılmıştır ancak son yapılan çalışmalar göstermiştir ki, bu sayıyı artırmak ile daha ciddi iyileştirmeler sağlanabilmektedir.

2.3.2 Dil modelleme araçları

İstatistiksel dil modelleme için çeşitli yazılım paketleri oluşturulmuştur. Bunlar uygulaması kolay algoritma paketleri olarak sunulmuştur. Öne çıkan uygulamalardan bir tanesi Carnegie Mellon – Cambrigde üniversiteleri tarafından geliştirilen LM (Language Modelling) Toolkit aracıdır. Konuşma işleme toplulukları tarafından araştırmalarda sıkça kullanılmıştır.

Bu tez kapsamında kullanılmış olan araç ise SRI Language Modelling (SRLIM) Toolkit olarak adlandırılan dil modelleme aracıdır. LM Toolkit'e (Language Modelling) kullanıcı arayüzü ve bir takım gelişmiş fonksiyonların eklenmesi ile oluşturulmuştur. Bu araç, C++ dilinde yazılmış kütüphanelerin ve çalıştırılabilir programları ile yardımcı kodların derlenmesi ile oluşturulmuş olup, konuşma tanıma ve diğer uygulamalar için istatistiksel dil modelleri oluşturma ve bunlar ile deneyler yapma olasılığı sağlayan bir araçtır. SRLIM ticari olmayan amaçlar için herkes tarafından kullanıma açıktır. Araç n-gram modellerden oluşturulacak çeşitli dil modellerini, farklı metotları ve buna benzer amaçlarla kelime kafeslerinin ve n-best listelerin istatistiksel işaretleme ve manipülasyonlarını desteklemektedir.

Eğitim verisinden bir model oluşturulması ve bu modeldeki kelime dizilim olasılıklarının hesaplanması bu araç ile gerçekleştirilmiştir.

2.4 Otomatik konuşma tanıma sonuçlarının değerlendirilmesi

Konuşma tanıma sonuçlarının değerlendirilmesinde çeşitli yöntemler ele alınabilir. Bunlardan bazıları

- Sözcük Hata Oranı (Word Error Rate - WER)
- Hassasiyet – Geri Çağırma(Precision-Recall) Grafikleri
- Merit Figürü (Figure of Merit -FOM) hesaplaması
- Alıcı Operasyon Karakteristiği (Receiver Operating Characteristics - ROC) Eğrisi

Otomatik konuşma tanıma için Sözcük Hata Oranı (WER) standart bir değerlendirme metriğidir.

Şu formül ile hesaplanır;

$$\text{WER} = 100 * \frac{\text{eklemeler} + \text{değiştirmeler} + \text{silmeler}}{\text{referans kelime sayısı}}$$

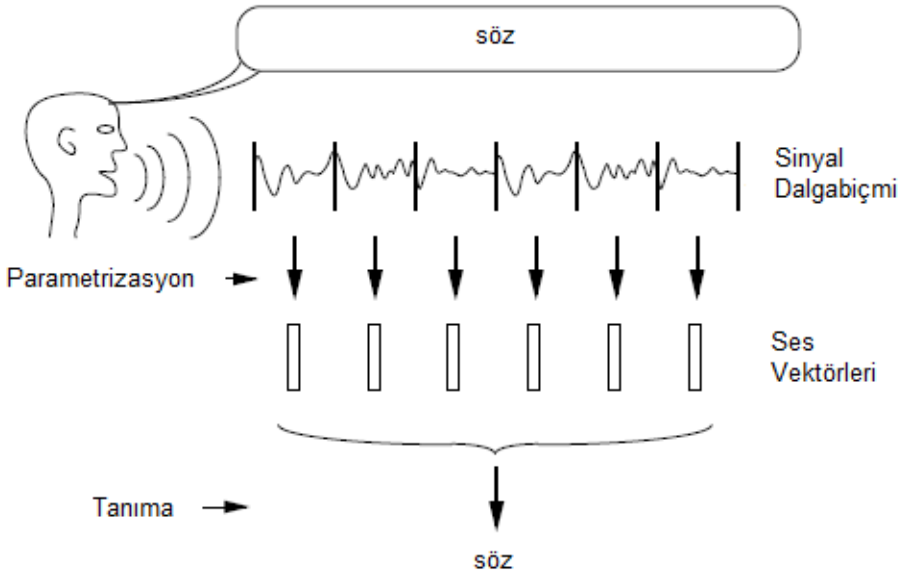
Burada pay terimi kelime bazında hipotez kelime ile referans transkripsiyonu arasındaki minimum değiştirme mesafesini (minimum edit distance) ifade etmektedir. Payda terimi ise bu işlemin ne kadar kelime arasından yapıldığını belirtmektedir.

3. HTK ARACI

Konuşma tanıma alanında birçok uygulama geliştirilmiştir. Bunlar arasından öne çıkanlar Cambrigde Üniversitesi tarafından geliştirilen Hidden Markov Model Toolkit (HTK), AT&T Research tarafından geliştirilen AT&T Decoder ve Carnegie Mellon Üniversitesi tarafından geliştirilen SPHINX'tir.

HTK (Hidden Markov Model Toolkit), konuşma tanıma sistemlerinde kullanılabilecek Hidden Markov Modeller yaratmak üzere Cambridge Üniversitesi Konuşma Grubu tarafından tasarlanmış bir araçtır. Açık kaynak kodlu olup, geliştirmelere ve eklemelere uygundur. Bu tezin hazırlanma döneminde en güncel versiyonu 3.1 numaralı versiyondur. Yeni versiyonlar geliştirildikçe, çalışma grubu tarafından yayınlanmaktadır.

HTK ile tanıma dört aşamada gerçekleştirilmektedir. Önce HTK'in eğitim araçları kullanılarak, verilen ses kayıtları ve bunların transkripsiyonları için oluşturulacak modeller çıkartılır. Daha sonra verilen bilinmeyen ses verisi, HTK'in tanıma araçları kullanılarak tanınır.



Şekil 3.1 HTK ile konuşma tanıma şeması

3.1 Çalışma Prensibi

HTK fonksiyon kütüphaneleri gerçekleştirilecek işlemler için gerekli araçları bulundurur.

HTK'in konuşma tanıma işleminde gerçekleştirdiği 4 aşama mevcuttur;

1. Veri hazırlama

Tanıma eğitimi için kullanılacak ses verisi ve bunun transkripsiyonu, istenen parametrik forma çevrilip, uygun formatta fonem ve kelime etiketleri oluşturulmalıdır. Ses kayıtları HTK ile veya herhangi başka bir kaynaktan edinilecek veri ile de oluşturulabilir.

2. Verileri eğitme

Birçok HMM eğitim modeli HTK ile gerçekleştirilebilir. Önceki bölümde açıklanan Baum-Welch prosedürü, Viterbi kodlama, N-best listelerinin hazırlanması ve HMM'lerin düzenlenebilmesi ile kullanıcıya akustik model eğitmede esneklik sağlar. Aynı zamanda dil modeli de eğitimi de HTK tarafından desteklenir. N-gram dil modelleri yaratmak için ilgili kütüphaneler mevcuttur.

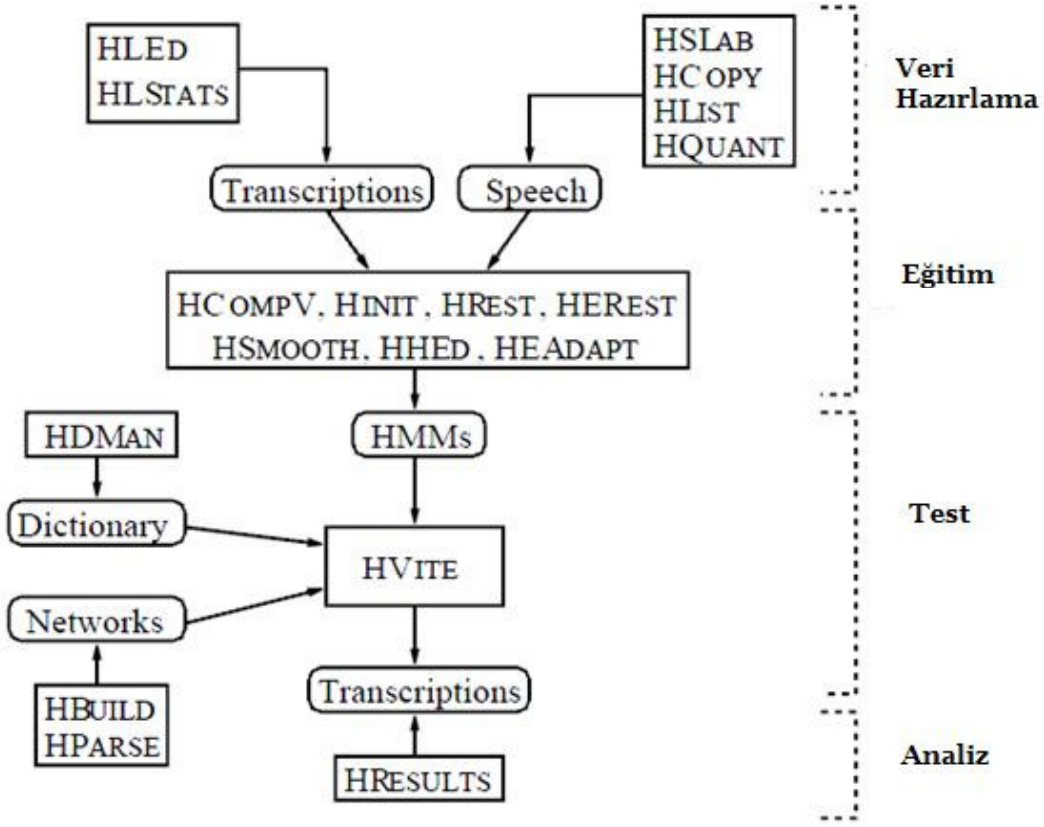
3. Tanıma

HTK'in sunduğu tanıma fonksiyonu dil modeli ve oluşturulan kafesler ile tanıma sağlar. İki temel fonksiyon HVite ve HDecode bu konuda kullanılmaktadırlar. Her ikisi de bir HMM Model seti ve dil modeli/sözlük kullanarak MLF (Master Label Format) veya SLF (Standart Lattice Format) formatında kafesler oluştururlar.

4. Analiz

Tanıma sonuçlarını analiz etmek için HResults komutu ile değerlendirmeler yapılabilir.

Bu işlemler gerçekleştirilirken kullanılan fonksiyonları özetleyen tablo aşağıdaki gibidir;



Şekil 3.2 HTK'in tanıma işlemlerini gerçekleştiren fonksiyon şeması

3.2 HTK ile Konuşma Tanıma

Bu tezde HTK ile gerçekleştirilen işlemler şu adımlardan oluşmuştur.

3.2.1 Gramer dosyası oluşturmak

Öncelikle gerçekleştirilecek tanıma işleminin niteliğine göre bir gramer oluşturulur. Bu gramer HTK'in sunduğu fonksiyonlardan HParse ile Şekil3.4'teki gibi bir kelime ağına (word network) dönüştürülür. Bu kelime ağı HTK'nin Standart Kafes Formatı (Standard Lattice Format – SLF) olarak tanımladığı

formatta yaratılır. Bu format tanıma esnasında kelimeler arası geçişleri tanımlamaktadır.

Sözcük tanıma uygulaması için örnek bir gramer aşağıdaki gibi tanımlanabilir.

```
$key =  gUreSmek;
$sil = sil;

(
[sil] (<[$key]>) [sil]
)
```

Şekil 3.3 Anahtar sözcük bulma grameri

Burada kullanılan format, HTK'nın tanımladığı gramer dil modelidir. Köşeli parantezler opsiyonel cümle parçalarını, açılı parantezler birden çok kere tekrarlayabilecek cümle parçalarını ifade etmektedir. Yani sessizlik-kelime-sessizlik şeklinde tanımlanan gramer içinde, kelime sayısı birden fazla olabilmekte ve sessizlik kısımları opsiyonel olarak yer bulmaktadır.

```
VERSION=1.0
N=5    L=6
I=0    W=!NULL
I=1    W=!NULL
I=2    W=sil
I=3    W=gUreSmek
I=4    W=sil
J=0     S=4     E=1
J=1     S=0     E=2
J=2     S=2     E=2
J=3     S=2     E=3
J=4     S=3     E=4
J=5     S=4     E=4
```

Şekil 3.4 Gramer dosyası ile hazırlanan kelime ağı

Bu şekilde ifade edilen kelime ağı, ses dosyasının içerisinde, bir kafes formatına çevrildiğinde elde edilen düğüm ve kolları göstermektedir. I düğümlerin indeksini, W kelimeyi, J kolların indeksini, S başlangıç nodunu, E ise varılan nodu göstermektedir.

3.2.2 Sözlük oluşturmak

İkinci adım tanıma sözlüğünün oluşturulmasıdır. Sözlük eğitilen verinin transkripsiyonudur aynı zamanda. Bu adım akustik model oluşturmada kullanılacağı için, tanıma sözlüğünün büyüklüğü ve fonetik olarak dengeli olması modelin dayanıklılığını arttırır. Sözlük, kayıtta geçen kelimenin transkripsiyonu yanı sıra seslendirilmesini de içermelidir. İngilizce tanıma gerçekleştiren uygulamalarda, hecelerin seslendirilmesi farklı karşılıklar bulabilirken, Türkçe yazıldığı gibi okunan bir dil olduğu için, kayıtlar, harf harf ayrılmış sözcükler olarak seslendirme sözlüğünde yer alır.

```
aGIrlIGI                    a G1 I1 r l I1 G1 I1
aGIzlarIndan             a G1 I1 z l a r I1 n d a n
aGlamaya                    a G1 l a m a y a

ABLE                        ey b ax l
ABNORMAL                 ae b n ow r m ax l
ABOARD                      ax b ow r d
```

Şekil 3.5 Türkçe ve İngilizce Sözlük örneği (lexicon)

Yukarıdaki şekilde HTK uygulaması için uygun formatta bir sözlük örneği verilmiştir. Türkçe sesli harfler uygulamaya, “harf+1” gösterimi ile tanıtılmıştır. İngilizce harfler ise okunuşuna uygun harf kombinasyonlar ile verilmiştir.

3.2.3 Ses kayıtlarının hazırlanması

Eğitim için kullanılacak ses kayıtları HTK ile oluşturulabilir. Bu işlem için hazır fonksiyonlar HSLab ve HSGen'dir. Başka bir kaynaktan, örneğin haber kayıtlarından veya çağrı merkezi görüşmelerinden edinilen veriler ile veya herhangi bir ses kayıt programı vasıtası ile de eğitim kayıt sesi oluşturulabilir.

3.2.4 Transkripsiyon dosyalarının oluşturulması

Bir HMM setini eğitmek için her eğitim dosyasının ilgili bir fonem düzeyinde transkripsiyonu olmalıdır. Kelime boyutundaki sözcükler doğrudan işlenemediği için Ana Etiket Dosyası (Master Label File – MLF) formatında transkripsiyon dosyalarını oluşturmak gereklidir.

```
#!MLF!#
"out/lattices/test.lab"
gUreSmek
.
"out/lattices/test.lab"
gUreS
tutmak

#!MLF!#
"*/sample1.lab"
sil
G
U
R
E
S
M
E
K
sil
.
```

Şekil 3.6 Kelime ve harf düzeyinde mlf dosyası örnekleri

Bu formatlar HTK uygulamasının dil modeli eğitimi ve tanıma değerlendirmelerinde kullandığı düzende verilmiştir. Sözcük ve harf bazında tanıma yapmayı sağlamaktadır.

3.2.5 Özellikli vektörleri çıkarmak

Veri hazırlama kısmının beşinci adımı ses dosyasındaki veriyi özellikli vektörlere dönüştürmektir. HTK Toolkit yukarıda daha detaylı olarak açıklanmış olan LPC ve FFT tabanlı analizleri desteklemektedir. Hazırlanan uygulamada FFT tabanlı log spektra'dan türetilmiş MFCC özellikleri kullanılacaktır. HTK'in HCopy fonksiyonu ile bu vektörler oluşturulmaktadır. Ancak öncelikle verilmesi gereken konfigürasyon aşağıdaki Şekil 3.7'de gösterilmiştir.

```
# Coding parameters
TARGETKIND = MFCC_0
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
ENORMALISE = F
```

Şekil 3.7. MFCC vektörlerini çıkarmak için kullanılan konfigürasyon dosyası

Burada çerçeve boyutunun ses sinyalinin özelliklerinde anlatılmış olduğu gibi 10ms olacağı, çıktının sıkıştırılmış formatta olacağı ve çıktıda 12 MFCC katsayısının bulunacağı gibi bir takım yapılandırmalar verilmiştir.

Veri hazırlama aşaması tamamlandıktan sonra, eğitim adımlarına geçilmektedir.

3.2.6 Mono fon HMM'lerin oluşturulması

Hazırlanan verileri eğitme kısmında öncelikle tekil-Gaussian mono fon olarak eğitilmiş bir HMM tariflenecektir. Bunun için prototip HMM dosyası hazırlanacaktır. HCompV komutu ile MFCC değerlerinde elde edilen hesaplamalar prototip dosyayı doldurur. Daha sonra bu modele sessizlik ve duraksama (silence – short pause) modelleri eklenerek, HREst komutu ile HMM değerleri yeniden oluşturulur. Son olarak HVite komutu yine klasördeki veriler bir referans dosyaya göre tekrar hizalanarak eğitim tamamlanır. Ortaya Şekil 3.8'deki gibi bir hmm klasörü çıkar.

```
~o
<STREAMINFO> 1 25
<VECSIZE> 25<NULLD><MFCC_D_N_Z_0><DIAGC>
~s "silst"
<MEAN> 25
 -1.687015e+01 4.438992e+00 -1.217539e+01 5.298560e+00 -8.477816e+00 5.718115e+00 -1.242172e+00
5.931939e+00 -1.560962e+00 1.708615e+00 2.976593e-01 -3.560771e+00 3.767588e-03 4.825946e-03
2.334869e-03 8.491258e-03 1.306933e-02 7.783043e-03 -1.500273e-02 -1.426337e-02 -2.347082e-02
-1.524391e-02 1.973551e-02 1.741393e-03 4.329497e-04
<VARIANCE> 25
 2.365789e+00 2.600042e+00 2.509712e+00 3.098348e+00 5.057958e+00 4.573861e+00 6.189445e+00
6.955873e+00 1.070578e+01 6.251605e+00 2.518376e+00 3.287338e+00 3.852637e-02 6.814048e-02
9.819699e-02 1.892491e-01 2.634540e-01 2.486979e-01 2.780010e-01 3.884485e-01 8.194297e-01
2.750033e-01 2.037393e-01 2.758864e-01 1.447709e-02
<GCONST> 3.955250e+01
~h "b"
<BEGINHMM>
<NUMSTATES> 5
<STATE> 2
<MEAN> 25
```

Şekil 3.8 Akustik model için eğitilen örnek mono fon HMM dosyası

3.2.7 Trifon HMM'lerin oluşturulması

Eğitim esnasında oluşturulan mono fon seslendirme sözlüğü, harflerin 3'lü gruplar halinde birbirine bağlanması ile trifon haline getirilmek istenmektedir. Trifon ile tanıma yapmak doğruluk oranını artırmaktadır, çünkü tanıma işleminin içerisine bağlam bilgisi de girmektedir. Tek bir harf aramak yerine, 3 harfin birleşiminden oluşan bir model aramak, doğruluk oranına olumlu bir katkı sağlamaktadır.

```
a-n+a
a-n+b
a-n+c
a-n+d
a-n+e
a-n+f
a-n+g
a-n+h
a-n+i
a-n+j
a-n+k
a-n+l
a-n+m
a-n+n
a-n+o
a-n+p
```

Şekil 3.9 Akustik model eğitiminde oluşturulmuş örnek bir trifon gösterimi

Yukarıdaki şekilde “n” harfinin trifon modelleme ile gösteriminin bir parçası verilmektedir. “n” harfinin öncesi ve sonrasında görülebilecek harflerin olasılık bilgileri ilgili kombinasyonla sisteme eklenmekte ve böylelikle tanıma yüzdelerini yükseltmek ihtimali sağlanmaktadır.

3.2.8 Tanıma sonuçlarının değerlendirilmesi

Veriler ve eğitilmiş modellerle olan işlemler tamamlandıktan sonra performans değerlendirmesi yapılır. Tanıma işlemi gerçekleştirdikten sonra HResults komutu ile tanıma sonuçları referans dosyalarla karşılaştırılarak doğru sonuçlar, hatalı sonuçlar ve ekleme, değiştirme, silme istatistikleri incelenebilir.

HTK toolkit ile geliştirilecek daha kapsamlı uygulamalar için veya fonksiyonların farklı kullanımları için tanımlar ve açıklamalı örnekler, kurulum dosyaları ile birlikte indirilen HTKBook’ta mevcuttur. [12]

4. VERİTABANI

4.1 Boğaziçi Üniversitesi Haber Kayıtları

Temel tanıma sistemi için Boğaziçi Üniversitesi tarafından toplanan Türkçe Haber Kayıtları verileri kullanılmıştır. BOUN News DB olarak adlandırılan bu veriler 2006 yılında CNN Türk, NTV, TRT1, TRT2 ve VoA radyo kanalından toparlanmıştır. Ayrıca TRT2'nin İşitme Engelliler için Haber Bülteni kayıtları da kullanılmıştır.

Kayıtlar televizyon ve radyodan otomatik bir kaydedici aracılığıyla kaydedilmiştir. Haber dışı verileri içerek kayıtlar elenmiş, kalitesi düşük yayınlar kapsama alınmamıştır. Geri kalan kayıtlar, segmente edilmiş, transkripsiyonları hazırlanmış ve düzenlenmiştir.

4.1.1 Segmente etmek

Haber kayıtlarını oluşturan verilerin çoğu elle segmente edilmiştir. Açıklamalarına göre sınıflandırılmış ve şu şekilde etiketlenmiştir; f0 (duru konuşma), f1 (spontane konuşma), f2 (telefon konuşması), f3 (arka fon), f4 (düşük kalitede ses).

4.1.2 Transkripsiyon

Segmente edilmiş ses kayıtlarının transkripsiyonları da elle gerçekleştirilmiştir. Veritabanında yaklaşık 277 saatlik konuşma kaydı mevcuttur.

Channel	f0	f1	f2	f3	f4	fx	Total
CNN	25.41	9.77	3.38	10.46	42.60	1.66	93.28
NTV	20.34	5.04	3.07	8.92	50.20	2.09	89.66
TRT2	5.57	1.74	0.17	3.32	9.16	0.26	20.22
IE	11.87	0	0	0	0	0	11.87
TRT1	1.16	1.37	0	0.36	2.66	0.14	5.69
VoA	36.46	1.49	7.98	6.21	4.63	0.15	56.92
Total	100.81	19.41	14.60	29.27	109.25	4.30	277.64

Tablo 4.1 Farklı kanallardan alınmış kayıtların süreleri

Test için kullanılan örnek bir haber kaydı aşağıdaki gibidir;

“Vergi iadesi tarih oldu, çalışanlar da artık vergi iadesi için fiş toplamayacak. Vergi iadesi yerine asgari geçim indirimi uygulamasını getiren yasa resmi gazetede yayımlanarak yürürlüğe girdi. Yasaya göre asgari geçim indirimi brüt asgari Ücretin yarısı olarak belirlenecek ve bir ocak iki bin sekizden itibaren elde edilen gelirler için geçerli olacak.”

4.2 Başkent Üniversitesi Ses Kayıtları

Başkent Üniversitesi ses veri tabanı için eğitim ve test amaçlı hazırlanan kayıtlar, Ses İşleme ve Konuşma Tanıma Komitesi (Sound Processing & Voice Recognition Committee) tarafından onaylanmış bir metinden alınmıştır. Bu metinde Türkçe dil yapısı için karakteristik özellik taşıyan “z, ş, p” harflerini içeren sözcükler bol miktarda kullanılmıştır.

Kayıtlar Audacity programı ile 16kHz örnekleme frekansında, 16 bit PCM formatında kaydedilmiştir.

Metinlerin transkripsiyonu elle hazırlanmış, kelime ve seslendirme sözlükleri, metin içerisinde geçen cümlelerin işlenmesi ile elle elde edilmiştir.

Eğitim için 154 adet örnek cümle kullanılmıştır.

Test için seçilen segment aşağıdaki cümle olmuştur;

“Öyle kung fu gösterileri ile başpehlivan olunmaz. Pazu göstererek de adam korkutamazsınız. Er meydanında belli olur kimin başpehlivan olduğu. Erkekçe güreşerek kazanılır başpehlivanlık. El kol sallayarak pazularını şişirerek değil. Yüreği yiyorsa senin güreşçinin, çıkar er meydanında benim pehlivanımın karşısına ve kıran kırana erkekçe güreş tutarlar. Biri diğerini tuş edene kadar güreşirler. Biz de görürüz hangisinin pazuları daha güçlüymüş ve hangisi gerçek baş pehlivanmış.”

5.GERÇEKLEŞTİRİLEN ÇALIŞMALAR

5.1 Sistem Hazırlıkları

Tezde amaç sesli bir metin içerisinde, belirtilen bir sözcüğün ve türevlerinin geçişlerinin tespiti olarak belirlenmiştir. Otomatik konuşma tanımanın genel tanımından varılan özelleşmede, bu çalışmanın, yalıtılmış konuşma konu başlığı altındaki sözcük bulma olarak tanımlandığı önceki bölümlerde tartışılmıştır. Bu işlemi gerçekleştirmek için örüntü tanıma metotlarından yararlanılmıştır.

Çalışma platformu olarak çoklu görev gerçekleştirmedeki hızı ve kararlı sistem özellikleri nedeniyle Linux tercih edilmiştir. Dağıtım olarak tez çalışmalarının başladığı dönem içerisindeki en güncel sürümlü Ubuntu kullanılmıştır. Yapılan uygulamaları Windows ortamında da gerçekleştirmek mümkündür, fakat Linux kullanım kolaylığının yanı sıra beraber, veri aktarımı için Boğaziçi Üniversitesi'nin uzak terminal bağlantılarını gerçekleştirmede de uyumluluk sağlamıştır.

Uygulamanın bir kısmının geliştirilmesi HTK Toolkit ile gerçekleştirilmiştir. Bu bağlamda Linux için HTK toolkit kurulumu yapılmıştır. HTK'in standart versiyonunda bulunmayıp, kullanılması gerekli olan ekstra kütüphaneler de program web sitesinden indirilip kuruluma eklenmiştir.

Sonuçların değerlendirilmesi STDEval Tool ile gerçekleştirilmiştir. NIST'in Konuşma tanıma çalışmaları grup web sitesinden, STDEval kurulumu indirilip uygulama çalışır hale getirilmiştir.

Uygulamada kullanılan ses kayıtları Audacity programı ile yapılmıştır. Bu programın da Linux için son versiyonu indirilmiştir.

İndirilen bütün programlar ve uygulamalar açık kaynak kodlu olup, hepsi ticari amaçlar için olmayan kullanımlarda bütün kullanıcılara açıktır. Güncel adresleri [21], [22], ve [23]'te belirtilmiştir.

5.2 Verilerin Hazırlanması

Çalışma esnasında, eğitim, model yaratma, test ve analiz aşamalarında kullanılan verilerin bir kısmını Boğaziçi Üniversitesi konuşma işleme grubu tarafından hazırlanan BOUN News verilerinden oluşmuştur. Yapılan modeller ayrıca Başkent Üniversitesi ses işleme çalışmaları için hazırlanan veri tabanı ile de test edilmiştir. Ses kayıtları Audacity programı ile 16KHz örnekleme hızında 16bit PCM olarak alınmıştır.

5.2.1 Dil modeli eğitimi

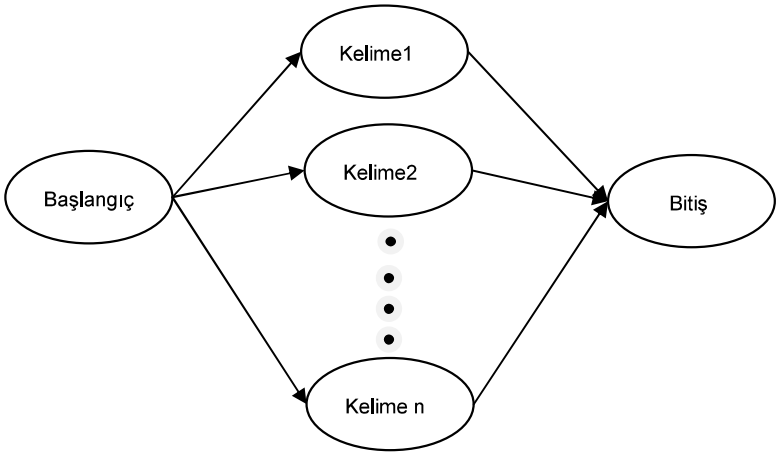
Başkent Üniversitesi verileri için model eğitimi sırasında ihtiyaç duyulan veri transkripsiyonları ve seslendirme sözlüğü elle hazırlanmıştır. Seslendirme sözlüğünde, değerlendirme aşamalarında eşleniklik sağlanması açısından BOUN News tarafından da kullanılmış olan aşağıdaki notasyon tercih edilmiştir.

```
acI                          a c I1
acImak                     a c I1 m a k
acImasIz                 a c I1 m a s I1 z
acIrIm                     a c I1 r I1 m
adam                         a d a m
adamakIllI             a d a m a k I1 l l I1
```

Şekil 5.1 Seslendirme sözlüğü örneği

Türkçenin içerdiği özel sesli harfler ‘büyük harf’ + 1 olarak ifade edilmiştir. Bu bağlamdı ‘ç’ harfi ‘C1’, ‘ı’ harfi ‘I1’ rakamları ile temsil edilmişlerdir.

Dil modeli eğitiminde, uygulamanın sözcük tanıma üzerine geliştirilmiş yapısı nedeniyle, aşağıdaki gramer kullanılmıştır;



Şekil 5.2 Anahtar sözcük tanıma grameri

Dil modeli eğitiminde 3.2. bölümde tarif edilen adımlar gerçekleştirilerek istenen model elde edilmiştir.

5.2.2 Akustik model eğitimi

Akustik model eğitimi için HTK aracı kullanılmış ve tariflenen adımlar izlenmiştir. Ses kayıtlarından MFCC özellik vektörleri çıkarıldıktan sonra mono fon ve trifon değerleri elde edilmiştir.

Verilerin işlenmesi ve gerekli referans dosyalarının oluşturulması için HTK aracının da çalışma olan Perl kodları hazırlanmıştır. Aşağıda HTK aracı ile akustik model eğitimi için kullanılan bir kod parçası örneği bulunmaktadır;

```
#create phone level transcriptions
HLEd -l '*' -d $workdir/pron_dict.txt -i $workdir/phones0.mlf mkphones0.led $workdir/trainwords.mlf

#Extract MFCC features
find $data_train_dir -iname '*.wav' > $workdir/train.list
perl -e 'open($FILE, $ARGV[0]."/train.list"); while (<$FILE>) {chomp(); $name = $_; $name =~ s/\.wav/\.mfc/; $name =~ s/\//
_/g; print $_, " $ARGV[0]/mfc/$name\n";}' $workdir > $workdir/codetr.scp
HCopy -C config -C configwav -S $workdir/codetr.scp

#create a new version of proto hmm with global means and variances
perl -ne 'chomp; @f=split(/\s+/); print $f[1], "\n";' $workdir/codetr.scp > $workdir/train.scp
HCompV -C config -f 0.01 -m -S $workdir/train.scp -M $workdir/hmm0 proto

#create monophone hmm definitions
echo "~o <VecSize> 39 <MFCC_0_D_A>" > $workdir/hmm0/macros
cat $workdir/hmm0/vFloors >> $workdir/hmm0/macros
create_hmm_defs.pl $workdir/hmm0/proto monophones0 > $workdir/hmm0/hmmdefs

#train monophone hmms
HERest -m 1 -C config -I $workdir/phones0.mlf -t 250.0 150.0 1000.0 -S $workdir/train.scp -H $workdir/hmm0/macros -H
$workdir/hmm0/hmmdefs -M $workdir/hmm1 monophones0
HERest -m 1 -C config -I $workdir/phones0.mlf -t 250.0 150.0 1000.0 -S $workdir/train.scp -H $workdir/hmm1/macros -H
$workdir/hmm1/hmmdefs -M $workdir/hmm2 monophones0
HERest -m 1 -C config -I $workdir/phones0.mlf -t 250.0 150.0 1000.0 -S $workdir/train.scp -H $workdir/hmm2/macros -H
$workdir/hmm2/hmmdefs -M $workdir/hmm3 monophones0

# Fix the silence model and add in our short pause Z, see HTKBook 3.2.2.
duplicate_silence.pl $workdir/hmm3/hmmdefs > $workdir/hmm4/hmmdefs
```

Şekil 5.3 Örnek bir Perl kodu parçası

5.2.3 Zoraki hizalama

Başkent Üniversitesi verisi işlenirken kullanılan bir yöntem zoraki hizalama (forced alingment) olarak adlandırılmaktadır. Bu işlem ile ses kaydında geçen sözcüklerin metin karşılıklarının kayıt içerisinde hangi zamanlarda geçtiklerinin sisteme öğretilmesi gerçekleştirilmiştir. Ses kayıtları ve metnin o kısmına karşılık gelen transkripsiyonlar uygulamaya tek tek girdi olarak verilerek, hizalanmış bir tanıma dosyası elde edilmektedir. [29]

Bu işlemin sonunda değerlendirme için gerekli referans dosyaları çıkartılmıştır. Recout.mlf dosyası bunlardan biridir.

```
#!MLF!#
"*/sample1.rec"
0 104800000 gUreSmek -82012.843750
.
"*/sample2.rec"
0 1500000 gUreS -2021.755615
1500000 52700000 tutmak -40719.648438
.
```

Şekil 5.4 Zoraki hizalama ile elde edilmiş bir recout dosyası

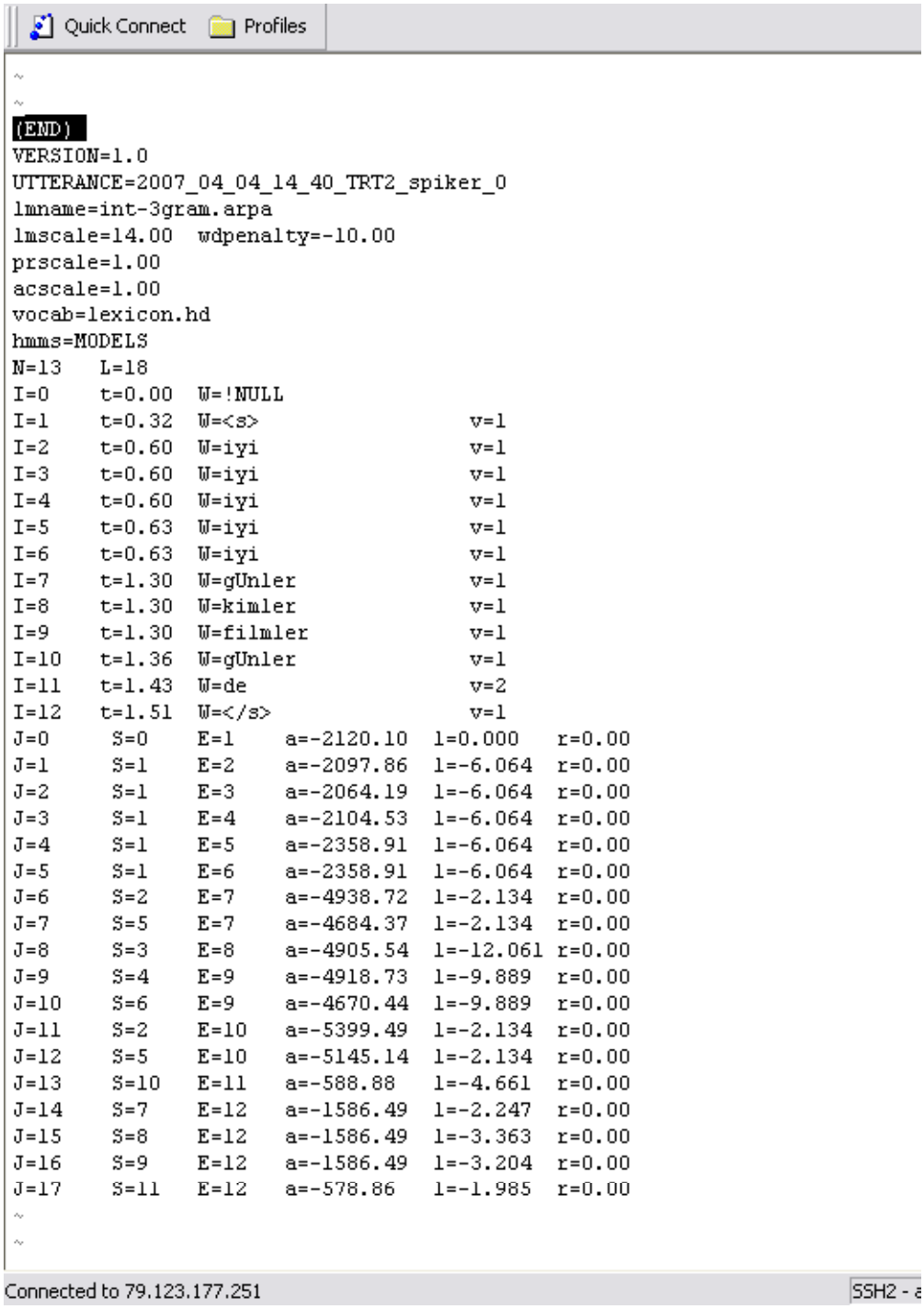
Burada yapılan işlem, ses kaydı ile bu kaydın transkripsiyonunu içeren bir gramer'i HTK'in HVite fonksiyonu ile hizalayarak kayıt içerisinde, kelimenin hangi sürede geçtiğinin belirlenmesidir. Örneğin 'güreşmek' ve 'güreş tutmak' kelimelerinin ses kaydı ve ilgili gramer dosyaları birlikte sisteme verilir ve bu

kayıtlar içerisinde, karşılık gelen sözcüklerin hangi zamanlarda geçtiği nanosaniye cinsinden belirlenir. İleriki aşamada tanıma sonuçları bu verilere göre değerlendirilecektir.

5.3 Tanıma çıktıları

Hazırlanan modeller ile test verisi, tanıma sonuçlarını almak için HTK uygulaması ile Viterbi algoritmasından geçirilir. HTK uygulamasının HVite komutu ile gerçekleştirilen bu işlem tanıma sonuçları kafes adı verilen dosyalar içerisinde oluşur. Sistem aldığı her ses kaydı için özellik vektörlerini kendi veritabanındaki bilgiler ile karşılaştırarak, örüntü tanıma yöntemi sonucu, en olası çıktıları listelemektedir. Bu sonuçlara dil modeli ve akustik modelden gelen katsayılar ile ağırlıklı olasılık değerleri vermektedir.

Ağırlıklı son durum çeviricileri ile modellenen bu veriler Şekil 5.4'te gösterildiği gibi dosyalar oluşturur;



Şekil 5.5 HTK ile oluşturulmuş örnek bir tanıma çıktısı

Burada I harfi, düğümlerin indeksini, t harfi tanınan parçanın başlangıç anını, W ise bulunan sözcüğü gösterir. Bu ekran Şekil2.7'deki kafesin HTK çıktısı olarak gösterimidir. Çıktının alt tarafındaki bölüm, geçiş olasılıklarını ifade eder. J durumunda S ile gösterilen başlangıç indeksine sahip sözcük, E ile gösterilen

bitiş indeksine geçiş yapmıştır. Bu geçişin olasılığı a akustik model olasılığı ve l dil modeli olasılığı çarpımları ile ifade edilir. Örneğin J=6 satırı incelendiğinde 2 numaralı indeksle gösterilen 'iyi' sözcüğünün, 7 numaralı indeksle gösterilen 'günler' sözcüğüne geçiş yaptığı görülür.

Bu şekilde oluşturulan, test kaydı miktarındaki kafes, tanıma sonuçlarının değerlendirilmesi için STD Eval aracına gönderilir.

5.4 Tanıma Sonuçlarının Değerlendirilmesi

Değerlendirme yapmak için çeşitli yöntemler uygulanabilir. HTK'in sunduğu HResults komutu ile kelime hata oranları alınabilir veya farklı uygulamalar gerçekleştirilebilir. Bu tez kapsamında sonuçların değerlendirilmesi için STD Eval Toolkit olarak adlandırılan araç kullanılmıştır.

5.4.1 STD Eval aracı

Amerika Birleşik Devletleri'ne bağlı bir ajans olan National Institute of Standards and Technology'ye (NIST) bağlı çalışan bir grup olan Information Technology Laboratory – Multimodal Information Group, konuşma tanıma değerlendirmesi (Speech Recognition Evaluation) çalışmalarının araştırma geliştirme sonuçlarını ve konuşma verilerinden bilgi elde etme teknolojilerindeki gelişmeleri ortak bir tabanda toplamak ve bu sonuçlara bir standart getirmek amacıyla 2006 yılında bir değerlendirme girişiminde bulunmuştur. Bu girişime Söylenen Sözcüğün Tespiti (Spoken Term Detection – STD) adı verilmiş ve teknik gelişimde ortaklaşa bir araştırma aktivitesi ile şu hedefler belirlenmiştir;

- Bu alandaki yeni fikirleri keşfetmek
- Teknolojinin gelişimine bu fikirler ışığında katkıda bulunmak
- Yapılan çalışmaların performanslarını ölçmek
- Bir platform oluşturarak, topluluk içinde elde edilen sonuçların değerlendirilmesini sağlamak

NIST 2006 yılında iki günlük bir atölye çalışması düzenleyerek, elde edilen sonuçların değerlendirilmesini sağlamıştır.

Çalışmada hedef, belirli bir terimin verilmiş konuşma kaydı içerisinde saptanması ve elde edilen sonuçların değerlendirilmesidir.

STD Eval Değerlendirme aracının kullanımı için bir kılavuz hazırlanmıştır [13]. Buna göre uygulamaya verilecek tanıma sonuçları için belirli formatlarda referans dosyalarına ihtiyaç vardır. Önceki bölümde bahsedilen referans dokümanlarından biri RTTM (rich transcription time mark) diğeri ise ECF (Experiment Control File) dosyasıdır.

```
<ecf source_signal_duration="548522.800" version="1">
<excerpt audio_filename="audio/sample100.sph" channel="1" tbeg="0.000"\
 dur="6981.000" language="turkish" source_type="baskent"/>
<excerpt audio_filename="audio/sample101.sph" channel="1" tbeg="0.000"\
 dur="4193.000" language="turkish" source_type="baskent"/>
<excerpt audio_filename="audio/sample102.sph" channel="1" tbeg="0.000"\
 dur="11113.000" language="turkish" source_type="baskent"/>
```

Şekil 5.6 STD Eval uygulaması tarafından referans olarak kullanılan .rttm dosyası örneği

```
SPKR-INFO       sample3 1       <NA>    <NA>    <NA>    unknown sample3
 <NA>
SPEAKER sample3 1       <NA>    <NA>    <NA>    <NA>    sample3 <NA>
NON-LEX sample3 1       0       1.33    sil     other   <NA>    <NA>
LEXEME  sample3 1       1.33    2.24    gUreS   lex     <NA>    <NA>
LEXEME  sample3 1       3.57    1.4     etmek   lex     <NA>    <NA>
NON-LEX sample3 1       4.97    0.089   sil     other   <NA>    <NA>
LEXEME  sample3 1       4.97    0.0899  sil     lex     <NA>    <NA>
```

Şekil 5.7 STD Eval uygulaması tarafından referans olarak kullanılan .ecf dosyası örneği

Bu dosyalar değerlendirme sistemine referans olarak verilip, girdi olarak gelen tanıma çıktılarının sonuçlarının değerlendirilmesini sağlamaktadır. .rttm uzantılı dosya her bir kayıt için kayıt ismi ile başlangıç bitiş sürelerini, dil bilgisini ve kaynak tipini belirtmektedir. Bu çalışma kapsamında tek dil ve tek tip kaynak çıktısı kullanılmıştır. .ecf uzantılı dosya ise her bir kayıt içerisindeki sözcüklerin ve sessizlik geçişlerinin başlangıç ve bitiş sürelerini belirtmektedir. <NA> olarak görülen alanlar, girilmesi zorunlu olmayıp, değerlendirme için kullanılabilecek ek parametreleri göstermektedir.

5.4.2 Tanıma sonuçları

Tez kapsamında iki farklı veri grubu ile eğitimler, modellemeler ve testler yapılmıştır.

İlk veri grubu olan Boğaziçi Üniversitesi'nin BUSIM grubu tarafından toparlanmış kayıtları, geniş bir sözlüğe sahiptir. Yaklaşık 300 saat kadar süren veri, oldukça dengeli ve çeşitlilik sağlayan bir model oluşturmuştur. Başkent Üniversitesi veritabanında bulunan kayıtlarının tamamı ise eğitilememiştir. Veri miktarı göreceli olarak daha kısa kalmış, 154 cümle ile eğitilmiş model, dengeli bir değerlendirme yaratmak için istenen çeşitliliği vermemiştir. Elde bulunan daha uzun süreli kayıtları eğitmek ve modellemek bu tezin hedeflerinin ve kapsamının ötesinde olmuştur. Bu nedenle temel sistemde test için kullanılacak model olarak haber verilerinden oluşan BOUN News modeli belirlenmiştir.

Tanıma testleri için bir sorgu listesi oluşturulmuştur. Anahtar kelime olarak verilen sorgu sözcükleri, sesli metin içerisinde tespit edilmek istenmiştir. Metnin tamamını kapsayan bir tanıma yüzdesi elde edilmek istendiğinden seçilen metin içerisinde geçen bütün sözcükler sorguda anahtar kelime olarak verilmiştir. Böylelikle tanıma gerçekleştiğinde bütün metin değerlendirilmiş ve bütün sözcükleri arayıp tespit eden çıktılarının elde edilmesi sağlanmıştır.

Yapılan tanıma çalışması sonucunda BOUN News kayıtları arasından seçilen bir parça ile elde edilen sonuç oldukça yüksek bir tanıma yüzdesine sahip olmuştur. STD Eval aracı ile gerçekleştirilen değerlendirme sonucunda ağırlıklı terim tanıma oranı %91 olarak bulunmuştur. Bu, sorgu olarak verilen sözcüklerin %91'ini doğru olarak tanıdığı anlamına gelmektedir.

İkinci adım olarak Başkent Üniversitesi ses kayıtları ile testler yapılmıştır. Burada tanıma sonucu göreceli olarak düşük çıkmıştır. Yapılan tanıma sonucu tespit oranı %65 olarak elde edilmiştir. Yani verilen sorgu sözcüklerinin %65'i doğru olarak tanınmıştır. Bu durumun değerlendirmesi Sonuçlar bölümünde yapılmaktadır.

6.SONUÇLAR

Tez kapsamında ses sinyalin özellikleri incelenmiş, sinyal üzerinde konuşma işleme için gerekli özelliklerin çıkarılmıştır. Konuşma tanıma için geliştirilmiş uygulamalar taranmış ve seçilen bir araç üzerinden konuşma işleme çalışmaları gerçekleştirilmiştir. Tanıma yapmak için ihtiyaç duyulan akustik ve dil modeli eğitimleri tamamlanarak, tanıma sonuçları değerlendirilmiştir.

Gerçekleştirilen çalışmaların sonucunda elde edilen değerlendirmelerde Başkent Üniversitesi veritabanı ile gerçekleştirilen anahtar sözcük tanıma sonuçları, BOUN News ile karşılaştırıldığında göreceli olarak düşük çıkmıştır. Sonuçların bu şekilde olmasında çeşitli etkenler vardır.

Birinci etken, Başkent Üniversitesi verilerindeki konunun tamamen farklı bir bağlama sahip olmasıdır. Türkçenin ses özelliklerini öne çıkaran metin parçalarından içerik güreş sporu ile ilgili olup, haber kayıtlarında fazlaca yer bulmuş bir konu olmamaktadır.

Bir başka etken bütün sözcüklerin sözlükte bulunmaması olasılığıdır. Türkçenin sondan eklemeli yapısı nedeniyle, bir kökten sınırsız sayıda sözcük üretilebilmektedir. Bu sözcüklerin tamamının dil modeli içerisinde modellenmesi hem işlem karmaşıklığı hem de veri fazlalığı açısından oldukça verimsizdir. Ancak sözlükte bulunmayan sözcükler ise sözlük dışı (out of vocabulary - OOV) olarak değerlendirileceği ve tanıma sonuçlarında yer alamayacağından, tanıma yüzdesini düşürmekte ancak gerçek bir performans sonucunu ortaya koymamaktadır. Bu durumun önüne geçmek bir “trade-off” yapmayı gerektirmektedir. Aynı köke sahip çok fazla sayıda sözcük üretilebileceğinden hepsini hazırlanan modelde bulundurmak maliyetli olmaktadır.

Geliştirme önerilerinden biri kelimelerin morfolojik analizinin yapılmasıdır. Bu alanda Kemal Oflazer tarafından geliştirilen bir uygulama ile kelimeler morfolojik köklerine yüksek yüzdeyle ayrılmaktadır. Morfolojik köklerine göre ayrılmış kelimeler, tanıma sözlüğünde bu şekilde yer alarak, ekli sözcüklerin tanınmasına katkı sağlamaktadır. Boğaziçi Üniversitesi’nde geliştirilen Morphologic Parser, bu sistemin, tanıma sistemine entegrasyonunu sağlamıştır.

[20] makalesinde, geliştirilen bir uygulamanın tanıma sonuçlarına etkileri ve yükselttiği başarı grafiği açıklanmıştır.

Bir diğer etken olarak kullanıcı bağımlılığından söz edebiliriz. Haber kayıtları çeşitli sunucunun sesleri alınarak oluşturulmuş ve mümkün olduğunca varyasyon yaratılmaya çalışılmıştır. Ancak tamamen farklı bir konuşmacı, aksan ve telaffuz ile gerçekleştirilmiş kayıt modeli eğiten kişinin verdiği test kaydından düşük bir yüzde çıkarabilir sonucuna varabiliriz.

Sonuncu etken olarak ise, kayıt kalitesi tartışılabilir. Ev ortamında mikrofonla alınmış kayıtlar ile stüdyoda alınmış profesyonel kayıtların tanıma sonuçlarına bir derecede etkisi olabilecektir.

Gelecek çalışmalar sözcük tanıma uygulaması ile morfolojik ayırıcıyı birlikte kullanarak, daha geniş bir veri seti üzerinde yapılacak bir çalışma ve analiz ile elde edilen sonuçların geliştirilmesi üzerine gerçekleştirilebilir.

7. KAYNAKLAR

[1] L. Rabiner, B. H. Huang, Fundamentals of Speech Recognition, Prentice Hall, 1993

[2] H. F. Olson and H. Belar, Phonetic Typewriter, J. Acoust. Soc. Am., Vol. 28, No. 6, pp. 1072-1081, 1956.

[3] J. W. Forgie and C. D. Forgie, Results Obtained from a Vowel Recognition Computer Program, J. Acoust. Soc. Am., Vol. 31, No. 11, pp. 1480-1489, 1959.

[4] J. Suzuki and K. Nakata, Recognition of Japanese Vowels—Preliminary to the Recognition of Speech, J. Radio Res. Lab, Vol. 37, No. 8, pp. 193-212, 1961.

[5] K. Nagata, Y. Kato, and S. Chiba, Spoken Digit Recognizer for Japanese Language, NEC Res. Develop., No. 6, 1963.

[6] D. B. Fry and P. Denes, The Design and Operation of the Mechanical Speech Recognizer at University College London, J. British Inst. Radio Engr., Vol. 19, No. 4, pp. 211-229, 1959.

[7] T. B. Martin, A. L. Nelson, and H. J. Zadell, Speech Recognition by Feature Abstraction Techniques, Tech. Report AL-TDR-64-176, Air Force Avionics Lab, 1964.

[8] T. K. Vintsyuk, Speech Discrimination by Dynamic Programming, Kibernetika, Vol. 4, No. 2, pp. 81-88, Jan.-Feb. 1968.

[9] J. T. Goodman, A Bit of Progress in Language Modeling, Machine Learning and Applied Statistics Group, Microsoft Research, One Microsoft Way, Redmond, WA 98052, U.S.A., 2001

[10] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, IEEE Fellow, 1989

[11] B.H. Juang, L. R. Rabiner, Automatic Speech Recognition – A Brief History of the Technology Development, Georgia Institute of Technology, Atlanta, Rutgers University and the University of California, Santa Barbara

[12] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, P. Woodland, The HTK Book, 2009

[13] The Spoken Term Detection (STD) 2006 Evaluation Plan, 2006

[14] A. Haznedaroğlu, L. M. Arslan, O. Büyük, M. Erden, Çağrı Merkezi Görüşmelerine Yönelik Türkçe Geniş Dağarcıklı Sürekli Konuşma Tanıma Sistemi, Boğaziçi Üniversitesi, Bebek, İstanbul, Türkiye, 2010

[15] A. Stolcke, SRILM —AN EXTENSIBLE LANGUAGE MODELING TOOLKIT, Speech Technology and Research Laboratory SRI International, CA, U.S.A., 2006

[16] S. Young, Large Vocabulary Continuous Speech Recognition: a Review, Cambridge University Engineering Department, 1996

[17] H. Sak, M. Saraçlar, T. Güngör, Integrating Morphology into Automatic Speech Recognition, Boğaziçi University, 2009

[18] X. L. Aubert, A Brief Overview Of Decoding Techniques For Large Vocabulary Continuous Speech Recognition, Philips Research Laboratories, Aachen, Germany, 2006

[19] M. Mohri, F. Pereira, M. Riley, Weighted Finite-State Transducers in Speech Recognition, AT&T Labs – Research, , NJ USA, Computer and Information Science Dept., University of Pennsylvania, PA USA, 2009

[20] H. Sak, M. Saraçlar, T. Güngör, Integrating Morphology into Automatic Speech Recognition, Boğaziçi Üniversitesi, 2009

[21] http://htk.eng.cam.ac.uk/download.shtml

[22] http://www.itl.nist.gov/iad/mig//tools/

[23] http://audacity.sourceforge.net/download/linux

[24] http://en.wikipedia.org

[25] D. Can, Indexation, Retrieval & Decision Techniques for Spoken Term Detection, , Bogaziçi University, 2006

[26] S. Parlak, Speech Retrieval For Turkish Broadcast News, Boğaziçi University, 2006

[27] M. A. Çömez, Large Vocabulary Continuous Speech Recognition for Turkish using HTK, ODTÜ, 2003

[28] K. Çarkı, T. Schultz, Turkish LVCSR: Towards Better Speech Recognition For Agglutinative Languages

[29]http://www.isip.piconepress.com/projects/speech/software/tutorials/production/fundamentals/v1.0/section_04/s04_04_p01.html