



YAPAY SİNİR AĞI İLE SESLİ-SESSİZ HARFLERİN TESPİTİ VE HECE

ÖLÇÜSÜNDE SES-METİN SENKRONİZASYONU

HALİL İBRAHİM BAYAT

YÜKSEK LİSANS TEZİ

MEKATRONİK MÜHENDİSLİĞİ

DR. ÖĞR. ÜYESİ MÜRSEL ÖNDER

Haziran - 2020
Her hakkı saklıdır

T.C.
TOKAT GAZİOSMANPAŞA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MEKATRONİK MÜHENDİSLİĞİ

MEKATRONİK MÜHENDİSLİĞİ

YAPAY SİNİR AĞI İLE SESLİ-SESSİZ HARFLERİN TESPİTİ VE HECE
ÖLÇÜSÜNDE SES-METİN SENKRONİZASYONU

HALİL İBRAHİM BAYAT

TOKAT
Haziran - 2020

Her hakkı saklıdır

Halil İbrahim Bayat tarafından hazırlanan “Yapay Sinir Ağı İle Sesli-Sessiz Harflerin Tespiti ve Hece Ölçüsünde Ses-Metin Senkronizasyonu” adlı tez çalışmasının savunma sınavı ...**25 HAZİRAN 2020** tarihinde yapılmış olup aşağıda verilen Jüri tarafından Oy Birliği / Oy Çokluğu ile Tokat Gaziosmanpaşa Üniversitesi Fen Bilimleri Enstitüsü **MEKATRONİK MÜHENDİSLİĞİ** Anabilim dalında da **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Danışman
DR. ÖĞR. ÜYESİ MÜRSEL ÖNDER

.....

Üye
Doç. Dr. Mahmut HEKİM

.....

Üye
Dr. Öğr. Üyesi Timur DÜZENLİ

.....

ONAY

Prof. Dr. Çetin ÇEKİÇ
Fen Bilimleri Enstitüsü Müdürü
---/---/2020

TEZ BEYANI

Tez yazım kurallarına uygun olarak hazırlanan bu tezin yazılmasında bilimsel ahlak kurallarına uyulduğunu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezin içerdiği yenilik ve sonuçların başka bir yerden alınmadığını, kullanılan verilerde herhangi bir tahrifat yapılmadığını, tezin herhangi bir kısmının bu üniversite veya başka bir üniversitedeki başka bir tez çalışması olarak sunulmadığını beyan ederim.

HALİL İBRAHİM BAYAT

25 Haziran 2020

ÖZET

YÜKSEK LİSANS TEZİ

YAPAY SİNİR AĞI İLE SESLİ-SESSİZ HARFLERİN TESPİTİ VE HECE ÖLÇÜSÜNDE SES-METİN SENKRONİZASYONU

HALİL İBRAHİM BAYAT

TOKAT GAZİOSMANPAŞA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI

TEZ DANIŞMANI: DR. ÖĞR. ÜYESİ MÜRSEL ÖNDER

Bu tez kapsamında, bir kelime içerisindeki sesli harfleri sessizlerden ayırt edilmesiyle hece ölçüsünde metin ses senkronizasyonu yapılması amaçlanmıştır. Geliştirilen bu yöntem kapsamında sınıflandırıcı olarak iki farklı yapay sinir ağı yapısı kullanılmış, ses özelliklerinin çıkarılmasında Mel-frekans cepstrum katsayıları tercih edilmiştir. Süreç içerisinde kullanılmak üzere katman özyinelemeli sinir ağı ile örüntü tanıma ağı çalışma için seçilmiştir. Konuşmacı bağımlı olarak geliştirilen bu ses tanıma sisteminde ilk aşamasında sesli ve sessiz harflerden ayrı ayrı örnekler alınmış, bu örneklerin özellikleri çıkarılarak ağlar eğitilmiştir. İkinci aşamada en iyi performansı veren ağ yapısı test edilerek saptanmıştır. Katman özyinelemeli sinir ağı sesli-sesiz sınıflandırmada ortalama %97.43 başarı sağlarken, örüntü tanıma ağı sesli harflerde tanımda %98.82 sessiz harflerde %97.27 ortalamada ise %98.04 başarı göstermiştir. Üçüncü aşamada seçilmiş olan bu ağa 10 adet kelime vektörlerinden oluşan sesler ayrı ayrı verilip kelimeler içerisindeki sesli harfler başarılı bir şekilde tespit edilmesi sağlanmıştır. Dördüncü adımda oluşturulan arayüzde bu kelimeler, geliştirilen heceleme algoritması ile hecelerine ayrılmış, metin ve bu metne ait ses hece ölçüsünde eş zamanlı olarak vurgulanmıştır. Sonuç olarak örüntü tanıma ağları sesli harfleri tanımda başarılı bir performans göstermiş ve sesli harflerin metin içerisinde tespit edilmesiyle metin-ses senkronizasyonunun yapılabileceği gösterilmiştir.

2020, 81 SAYFA

ANAHTAR KELİMELER: Yapay sinir ağları, Sesli-sessiz harf tespiti, Ses-metin senkronizasyonu,

ABSTRACT

MASTER THESIS

IDENTIFICATION OF VOWEL-NON VOWEL LETTER WITH ARTIFICIAL NEURAL NETWORK AND SOUND-TEXT SYNCHRONIZATION AT SYLLABLE LEVEL

HALİL İBRAHİM BAYAT

**TOKAT GAZİOSMANPAŞA UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES**

DEPARTMENT OF MECHATRONICS ENGINEERING

SUPERVISOR:ASST. PROF. DR., MURSEL ONDER

In the context of this thesis, it was aimed to synchronize text-voice in syllable-size by discriminating vowels from non-vowel in a word. Within the scope of this developed method, two different artificial neural network models were used as classifier, and Mel-frequency cepstrum coefficients were chosen for voice feature extraction. Pattern recognition network and Layer-recurrent network were used for this process. In this speech recognition system, which is created as Speaker-depended, vowel and consonant samples were taken at the first stage, and the neural networks were trained by features extracting of these samples. In the second stage, the network, which gave the best performance, was determined by testing. While the layer-recurrent neural network achieved an average of %97.43 success in the classification of vowel-consonant, the pattern recognition network achieved %98.82 in identification of vowels, %97.27 in consonants and %98.04 on average. In the third stage, in which the network showing the best performance was chosen, the sounds consisting of 10-word vectors were supplied separately to the network and the vowels in the words were successfully detected. In the fourth step, these words are separated into syllables with the developed syllable algorithm in which created in an interface program, the text and voice files that corresponding to these words are simultaneously emphasized in the syllable size. As a result, pattern recognition networks show a successful performance in the recognition of vowels, and it was also shown that text-voice synchronization can be done by detecting vowels in the text.

2020, 81 PAGE

KEYWORDS: Vowel-nonvowel identification, Artificial neural network, Sound-text synchronization

ÖNSÖZ

Bu tezin hazırlanmasında öncelikle benden desteğini hiçbir zaman esirgemeyen, gerek fikirleri gerekse manevi desteğini sürekli yanımda hissettiğim değerli hocam ve danışmanım Dr. Öğr. Üyesi Mürsel ÖNDER'e en içten teşekkürlerimi sunarım.

Hayatım boyunca beni her durum ve şart altında destekleyen, eğitim ve öğretim hayatım boyunca üzerimde ilgisini ve motivasyonunu daima hissettiğim annem sn. Birten BAYAT ve bana her türlü imkanı sağlayan babam sn. Alaaddin BAYAT'a sonsuz teşekkürlerimi sunarım.

Ayrıca araştırma ve çalışmalarım boyunca, benden tecrübesini ve ilgisini esirgemeyen değerli hocam sn. Alper GÜLBE'ye teşekkürü bir borç bilirim.

Son olarak bu süreç boyunca benden manevi desteklerini esirgemeyen değerli arkadaşlarıma ve kardeşime teşekkürlerimi sunarım.

HALİL İBRAHİM BAYAT

25 Haziran 2020

İÇİNDEKİLER

Sayfa

ÖZET	i
ABSTRACT.....	ii
ÖNSÖZ	iii
İÇİNDEKİLER	iv
SİMGE VE KISALTMALAR	vi
ŞEKİL LİSTESİ.....	vii
ÇİZELGE LİSTESİ.....	viii
1. GİRİŞ	1
1.1 Neden Hece Ayrımı?.....	1
1.2 Türkçe'deki Harfler	2
2. KAYNAK ÖZETLERİ	4
3. MATERYAL ve YÖNTEM.....	11
3.1 Yapay Sinir Ağları.....	11
3.1.1 Nöron (<i>Neuron</i>)	12
3.1.2 Algılayıcı (<i>Perceptron</i>)	13
3.1.3 Aktivasyon fonksiyonları	13
3.1.4 Katmanlar (<i>Layers</i>).....	17
3.1.5 Öğrenme metotları.....	18
3.1.6 YSA tipleri	19
3.1.7 Derin sinir ağları.....	22
3.1.8 YSA avantajları.....	23
3.1.9 YSA dezavantajları.....	24
3.2 Ses İşleme ve Özellik Çıkarma	24
3.2.1 MFCC	25
3.3 Ses Tanıma Sistemleri Hakkında Temel Kavramlar.....	32
3.4 Materyal ve Çalışma Çerçevesi	34
3.4.1 Katman-özyinelemeli Sinir Ağı.....	39
3.4.2 Örüntü tanıma ağı	40
3.4.3 Hedef (<i>Target</i>) matrislerinin oluşturulması.....	41

3.4.4 Ağların eğitimi ve testler	42
3.4.5 Arayüz programının oluşturulması	46
4. BULGULAR.....	53
5. TARTIŞMA ve SONUÇ	70
KAYNAKLAR	72
EKLER	76
ÖZGEÇMİŞ	81



SİMGELER ve KISALTMALAR

Simgeler	Açıklama
C_n	Mel-frekans katsayıları
S_k	Mel filtrelerinin k. filtrenin kuvveti
w_i	Ağırlık
$ x(k) $	DFT parametrelerinin genliği
x_i	YSA girdisi
B	Bias
E	Euler sabiti
$h(n)$	Hamming pencereleme fonksiyonu
$Mel(f)$	Mel-frekans dönüşümü
N	DFT derecesi
$f(x_i)$	Softmax fonksiyonu
$x(k)$	DFT parametreleri
$x(n)$	Ses sinyali
$y(x)$	YSA çıktısı

Kısaltmalar	Açıklama
AKD	Ayrık kosünüs dönüşümü
ÇKA	Çok katmanlı algılayıcı
DFT	Ayrık Fourier dönüşümü (<i>Discrete Fourier transform</i>)
DSA	Derin sinir ağları
İBSA	İleri beslemeli sinir ağları
KÖSA	Katman özyinelemeli sinir ağları
LPC	Doğrusal Tahminli Kodlama (<i>Linear predictive coding</i>)
MFC	Mel-frekans cepstrum
MFCC	Mel-frekans cepstrum katsayıları
ÖH	Özdüzenleyici haritalar
ÖSA	Özyinelemeli sinir ağları
ÖTA	Örüntü tanıma ağları
PLP	Algısal lineer tahmin (<i>Perceptual linear prediction</i>)
RTFA	Radyal temelli fonksiyon ağları
TIMIT	Teksas belgeler (<i>Instruments</i>) / Massachusetts Teknoloji Enstitüsü
YSA	Yapay sinir ağları

ŞEKİL LİSTESİ

<u>Şekil</u>	<u>Sayfa</u>
Şekil 2.1 Parlaktuna ve arkadaşlarının oluşturduğu sinir ağı yapısı	5
Şekil 2.2 Srigos ve arkadaşlarının oluşturduğu sınıflandırıcı modeli	6
Şekil 3.1 Yapay sinir ağlarının(ileri baslemeli) şematik gösterimi	3
Şekil 3.2 Biyolojik nöron ve kısımları	12
Şekil 3.3 Tekli bir algılayıcının şematik gösterimi	12
Şekil 3.4 Lineer aktivaston fonksiyon grafiği	14
Şekil 3.5 Sigmoid fonksiyon grafiği	15
Şekil 3.6 Tanjant hiperbolik Fonksiyon	16
Şekil 3.7 ÖSA yapısı	20
Şekil 3.8 Elman Ağları şematik gösterimi.....	21
Şekil 3.9 Jordan Ağları şematik gösterimi.....	22
Şekil 3.10 MFC Katsayılarının elde edilme sürecini gösteren diyagram	26
Şekil 3.11 Kayan çerçeveleme işleminin gösterimi.....	27
Şekil 3.12 Hamming pencereleme işlemi öncesi ve sonrasını gösteren grafik	28
Şekil 3.13 Her bir harfin oluşturulduğu veri setlerinin şematik gösterimi	37
Şekil 3.14 Bütün ses verilerinin tek bir matriste birleştirildiği <i>Ham_Giris_Verisi</i> matrisinin gösterimi	37
Şekil 3.15 KÖSA 'nın modelinin şematik gösterimi	39
Şekil 3.16 Matlab' da ÖTA yapısının şematik gösterimi	40
Şekil 3.17 İkili sınıf için oluşturulan hedef matrisi... ..	41
Şekil 3.18 Dokuzlu sınıf için oluşturulan hedef matrisi	42
Şekil 3.19 ÖTA eğitim performans grafiği.....	43
Şekil 3.20 KÖSA eğitim performans grafiği	44
Şekil 3.21 Hecelere ayırma algoritmasının akış şeması	48
Şekil 3.22 Arayüz programı çalışma akış diyagramı.....	49
Şekil 3.23 Arayüz programı genel görüntüsü	50
Şekil 3.24 Arayüz programında hecelenmiş olarak gösterilen <i>merhaba</i> metni.....	51

Şekil 3.25 Saptanan sesli harf zamanlarının ses okunmasıyla ekranda..... vurgulanması.....	52
Şekil 4.1 Sesli-Sessiz testi sonuçlarının istatistiksel değerlendirmesi	60
Şekil 4.2 Ağa gönderilen “a” harfinin sonuç matrisinin görsel çıktısı	60
Şekil 4.3 Ağa gönderilen “t” harfinin sonuç matrisinin görsel çıktısı.....	61
Şekil 4.4 Dokuzlu sınıf içi ağa gönderilmiş “i” harfinin çıktısı	62
Şekil 4.5 Ağa gönderilmiş “y” harfinin çıktısı	63
Şekil 4.6 Bütün sesli harflerin sıralı olduğu ses dosyası grafiği.....	64
Şekil 4.7 Bütün sesli harflerin bulunduğu dosyanın ağ çıktısı	64
Şekil 4.8 (b), “merhaba” kelimesini içersindeki sesli-sesiz harf ayrımını gösteren grafik, (a) zaman tanım kümesinde bu ses dosyasının grafiği	65
Şekil 4.9 (b), “dünya” kelimesini içersindeki sesli-sesiz harf ayrımını gösteren grafik, (a) bu sese ayit zaman tanım kümesindeki grafiği	66
Şekil 4.10 “merhaba” kelimesine ayit çıkış matrisi üzerindeki sesli harflerin yerleri	67
Şekil 4.11 “merhaba dünya” kelimesinin sesli-sessiz sınıflandırma grafiği	68
Şekil 7.1 “istanbul” kelimesinin ağ çıkış matrisi grafiği	76
Şekil 7.2 “günaydın” kelimesinin ağ çıkış matrisi grafiği.....	76
Şekil 7.3 “Türkiye” kelimesinin ağ çıkış matrisi grafiği	77
Şekil 7.4 “cumhuriyet” kelimesinin ağ çıkış matrisi grafiği	77
Şekil 7.5 “bilim” kelimesinin ağ çıkış matrisi grafiği	78
Şekil 7.6 “millet” kelimesinin ağ çıkış matrisi grafiği	78
Şekil 7.7 “özgürlük” kelimesinin ağ çıkış matrisi grafiği	79
Şekil 7.8 “vatan” kelimesinin ağ çıkış matrisi grafiği	79
Şekil 7.9 “ay yıldız” kelimesinin ağ çıkış matrisi grafiği.....	80

ÇİZELGE LİSTESİ

<u>Çizelge</u>	<u>Sayfa</u>
Çizelge 1.1 Türkçe 'deki alfabede bulunan sesli ve sessiz harfler	2
Çizelge 1.2 Türkçedeki ünlü harflerin kendi arasında sınıflandırılması	2
Çizelge 1.3 Türkçedeki sessiz harflerin kendi içerisinde sınıflandırılması	3
Çizelge 2.1 Sürekli sesli ifade sinyali ile farklı heceler için farklı özellikteki nöron ağları ile kullanımı.....	7
Çizelge 2.2 Farklı sayıda ayrık heceler için farklı özellikteki nöron ağları ile kullanımı	7
Çizelge 3.1 Üçgen band geçiren filtrelerin frekans eşikleri ve sayısı	31
Çizelge 3.2 Sesli ve sessiz harflerden alınan ses kayıtlarının örnek sayısı ve süre aralıkları	36
Çizelge 3.3 Özellik çıkarımında kullanılan parametreler	38
Çizelge 4.1 Tek katman için ÖTA nöron sayısı değişimi ve performans parametreleri	54
Çizelge 4.2 İki katman için ÖTA nöron sayısı değişimi ve performans parametreleri	54
Çizelge 4.3 Üç katman için ÖTA nöron sayısı değişimi ve performans parametreleri	55
Çizelge 4.4 Tek katman için KÖSA nöron sayısı değişimi ve performans parametreleri	56
Çizelge 4.5 İki katman için KÖSA nöron sayısı değişimi ve performans parametreleri	56
Çizelge 4.6 Üç katman için KÖSA nöron sayısı değişimi ve performans parametreleri	57
Çizelge 4.7 Seçilen ÖTA için 10-20 arası MFC katsayılarının performans parametreleri	58
Çizelge 4.8 Seçilen KÖSA için 10-20 arası MFC katsayılarının performans parametreleri	58
Çizelge 4.9 Farklı çerçeve uzunluklarına göre ÖTA yapısının test sonuçları	59
Çizelge 4.10 Ağa gönderilmiş “a” sesine ayit sonuç matrisi (ilk 7 çerçeve).	63
Çizelge 4.11 Ağa gönderilmiş “t” sesine ayit sonuç matrisi	61
Çizelge 4.12 Ağin “i” harfine ayit çıkış matrisi	61
Çizelge 4.13 Ağin “i” harfine ayit çıkış matrisi	62

Çizelge 4.14 Çalışmada kullanılan kelime ve kelime grupları	68
--	-----------



1.GİRİŞ

Bilindiği üzere günlük yaşantımızda izlediğimiz görsellerde (TV, sinema, karaoke) sesin(*konusma*) daha iyi anlaşılabilmesi için o sese ait altyazılar kullanılmaktadır. Bu alt yazılar ve bunların zamanlamaları belli programlar vasıtası ile el yordamıyla yani manuel olarak ayarlanmaktadır.

Bu tezdeki çalışmada bu işlem için yapay sinir ağları (YSA) ile metin içerisindeki sesli harflerin zamanları saptanarak hece ölçüsünde metin ve bu metne ait sesin senkron bir şekilde gösterimine dayanan bir yöntem sunulmuştur. Burada bu yöntem geliştirilirken Türkçe'nin dilbilimsel yapısı göz önünde bulundurulmuştur. Bir dilde cümleler kelimelerden, kelimeler hecelerden, ve hecelerde harflerden oluşmaktadır. Buradaki motivasyon eğer bir metin kendini oluşturan hecelerin ölçüsünde eş zamanlı gösterimi yapılırsa bu o kelimenin vurgulamasında daha hassas ve verimli bir gösterim meydana getirecektir. Bu da ayrıca Türkçe'nin öğrenilmesi ve duyma engelli insanların sese ait metni takip etmesinde faydalı olacağı düşünülmektedir (Yalçın, 2006).

1.1 Neden Hece Ayrımı?

Bu tez çalışmasında hece boyutunda bir ses-metin eş zamanlı vurgulama yöntemi uygulanmıştır. Bilindiği üzere Türkçe eklemeli diller ailesine mensuptur. Kelimeler bir kökten oluşur ve bu kelime köküne bir takım ekler gelerek kelimenin anlamını, yapısını ve de durumunu değiştirebilir. Bu durumda kelime ve bu kelimeyi oluşturan heceler ayrı bir önem kazanmaktadır. Çünkü Türkçede her hece bir sesli harf etrafında oluşur. Eğer bu hecelerin içerisindeki sesli harfler saptanır ise hecenin seslendirildiği an ve dolayısıyla kelimeyi oluşturan hecelerin seslendirme anları yüksek bir hassasiyetle vurgulanmış olacaktır (Önder, Basım aşamasında). Bu durum aynı zamanda bu dilin öğrenilmesinde, eğitim amaçlı sesli harflerin sessiz harfler içerisindeki hece oluşumu bağlamında önemli bir fayda getireceği düşünülmektedir.

1.2 Türkçe'deki Harfler

Türkçe de sesleri ifade etmek için kullanılan Latin alfabesinde 29 harf bulunmaktadır. Çizelge 1.1'de Türkçedeki sesli ve sessiz harfler verilmiştir. Bu harfler ünlü ve ünsüz harfler olarak temelde iki farklı sınıfa ayrılarak incelenir.

Çizelge 1.1 Türkçe 'deki alfabede bulunan sesli ve sessiz harfler.

Sesli Harfler	A	E	I	İ	O	Ö	U	Ü
Sessiz Harfler	B	C	Ç	D	F	G	Ğ	H
	J	K	L	M	N	P	R	S
	Ş	T	V	Y	Z			

Çiğerlerdeki havanın çıkarılması esnasında, gırtlaktan sonra havanın belirli bir yerde tıkanması, hava yolundaki çeşitli daralmalar, sürtünmeler, ses tellerinin titreşme durumu, dilin dişlere değip değmemesi, dudakların konumu gibi durumlar seslerin niteliğini belirlemede kullanılan önemli etmenlerdir (*Demir ve Yılmaz, 2011*). Bu gibi ayrımlar ile sesli ve sessiz harfler de kendi aralarında bir takım sınıflandırmaya tabidir. Bu sınıflandırmalar Çizelge 1.2 ve Çizelge 1.3' de ana hatlarıyla gösterilmiştir.

Çizelge 1.2 Türkçedeki ünlü harflerin kendi arasında sınıflandırılması (*Demir ve Yılmaz 2011*).

	ART		ÖN	
	DÜZ	YUVARLAK	DÜZ	YUVARLAK
GENİŞ	A	O	E	Ö
DAR	I	U	İ	Ü

Çizelge 1.3 Türkçedeki sessiz harflerin kendi içerisindeki sınıflandırılması(Demir ve Yılmaz, 2011).

	ÖTÜMLÜ			ÖTÜMSÜZ	
	SÜREKLİ		SÜREKSİZ	SÜREKLİ SÜREKSİZ	
	Akıcı	Sızıcı	Patlayıcı	Sızıcı	Patlayıcı
Çift-Dudak	M		B		P
Diş-Dudak		V		F	
Diş	N	Z	D	S	T
Diş-Damak		J	C	Ş	Ç
Ön-Damak	L		G		
Art-Damak		Ğ	G		K
Gırtlak				H	K
Çarpmalı	R				
Yarı Ünlü	Y				

Bu tezin oluşturulmasındaki temel hipotez Türkçedeki sesli ve sessiz harflerin bir ses tanıma sistemi ile öğrenilmesinin ardından bu sisteme bir kelime vektörü sunulduğunda bu vektör içerisindeki sesli ve sessiz harfleri ayırt edilmesi sayesinde sesli harflerin kelime içerisindeki zamanlarını tespit etmektir. Bu zaman tespiti ile ses metin eş zamanlılığı mümkün olacaktır.

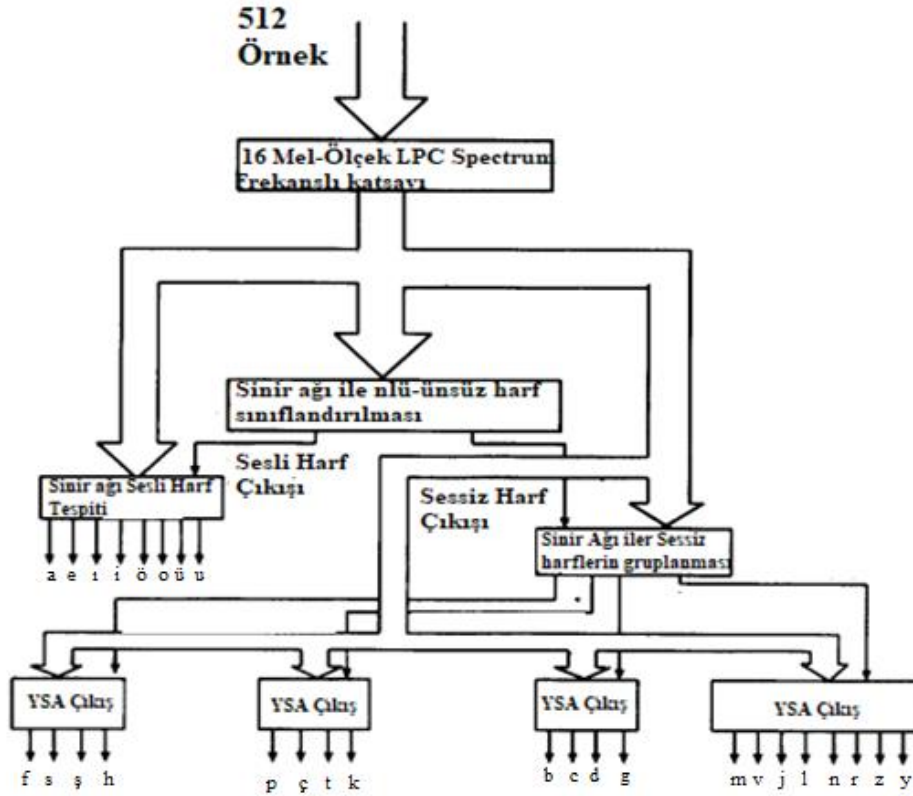
Bu kapsamda “Kaynak Özetleri” kısmında literatürde yapılmış benzer çalışmalardan bahsedilmiş, “Materyal ve Yöntem” kısmında ise YSA ve ses işleme ile ilgili genel bilgilerin yer aldığı, çalışmada kullanılan materyal özelliklerinden bahsedilip yapılan çalışmalar özetlenmiştir. “Bulgular” kısmında ise yapılmış çalışmalar detaylandırılarak sunulmuş ve çıkarımlardan bahsedilmiştir. “Tartışma ve Sonuç” kısmında ise genel değerlendirmelerden ve önerilerden oluşmaktadır.

2.KAYNAK ÖZETLERİ

Bu kısımda literatürde YSA ile yapılmış bazı ses tanıma çalışmalarına yer verilmiştir. YSA'nın son çeyrek asrı aşan popülerliğinden dolayı literatürde bu konu ile ilgili sayısız çalışma mevcuttur. Bu çalışmalarda görüleceği üzere, farklı YSA türleri ve ses sinyallerinin özelliklerini çıkarmak için farklı yöntemler denendiği saptanmış bunların sonuçları değerlendirilmiştir.

Cosi ve arkadaşları İngilizce sesli harf alt bileşenli konuşmacıdan bağımsız bir fonem (*phoneme*) sınıflandırma sistemi oluşturarak bir çalışma gerçekleştirmiştir (*Cosi ve ark., 1990*). Bu sesli harf sınıflandırıcısının mimarisi bir kulak modeline ve çok katmanlı sinir ağına dayanmaktadır. Sinir ağı seslendirme yeri, dil konumu ve seslendirme tarzı özellikleri öğrenmesi için eğitilmiştir. Bu kapsamda yeni konuşmacılar üzerinde gerçekleştirilen deneylerde 10 tane İngilizce sesli harf üzerinde %95 bir tanıma oranı saptanmıştır.

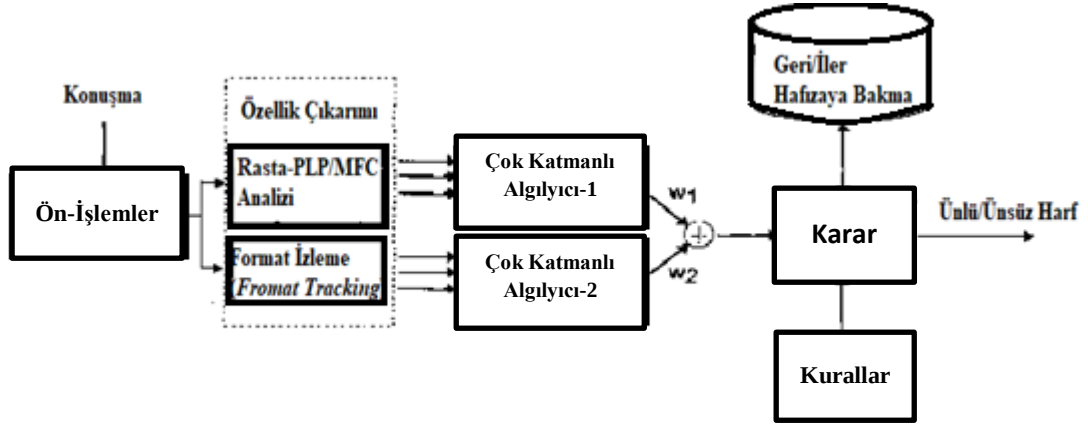
Parlak Tuna ve arkadaşları YSA ile Türkçe 'de sesli ve sessiz harfleri tanıma işlemini gerçekleştirmişlerdir (*Parlaktuna ve Ark., 1994*). Bu tanıma sistemi sesli-sessiz tanıma, sesli tanıma ve sessiz harfleri tanıma olmak üzere üç grupta ele alınmıştır. Eğitim verisi için 11 konuşmacı, test verisi için de 7 konuşmacıdan ses örnekleri temin edilmiş ve seslerin özelliklerinin çıkarılması için 16 Mel-ölçekli frekans noktası 240-4500 Hz arasında kullanılarak ses verilerinin Doğrusal Tahminli Kodlayıcı (*Linear Predictive Coding*) ile özellikleri çıkarılmıştır.



Şekil 2.1 Parlaktuna ve arkadaşlarının oluşturduğu sinir ağı yapısı (Parlaktuna, 1994)

Sesli-sessiz harfleri tanımda test verileri için ortalama, sırasıyla %84.7, %91.1; sesli harfleri tanımda ortalama %80.1; sessiz harfleri tanımda %50.7 başarı sağlanmıştır. Bu çalışmada kullanılan YSA yapısı Şekil 2.1’ de gösterilmiştir.

Srigos ve arkadaşları ikili Çok Katmanlı Algılayıcı (ÇKA) yapılışı ve Rasta-PLP(Perceptual Linear Prediction) yöntemi ile sesli-sessiz karar verici bir sınıflandırıcı uygulaması gerçekleştirmiştir (Sirigos ve ark., 1996). Rasta-PLP yöntemi ile 15 MFC (Mel-frekans kepstrum) katsayısı elde edilerek gerçekleştirilen bu uygulamanın blok diyagramı Şekil 2.2’de gösterilmiştir.



Şekil 2.2 Srigos ve arkadaşlarının oluşturduğu sınıflandırıcı modeli (Srigos ve ark., 1996).

Bu çalışmada Texas Instruments/Massachusetts Institute of Technology (TIMIT) veri tabanına ait akustik-fonetik kütüphaneden ses verileri temin edilmiştir. Bu kütüphane 10 cümlelerin 630 farklı konuşmacıya okutulmuş ses verilerinden oluşmaktadır. Bu veriler Amerika'nın 8 farklı bölgesinden insanlara aittir. Bu çalışma konuşmacıdan bağımsız bir çalışmadır. Çalışmanın eğitim ve test aşamalarındaki veriler de buradan temin edilmiştir. Eğitim ve test aşamalarında 15 MFC katsayısı kullanılmış ve %98 bir başarı elde edilmiştir.

Üstün LPC (*Linear predictive coding*) ve Ayrık Fourier Dönüşümü (*DFT*) analizi metodu ile YSA sınıflandırıcı olarak kullanılmasıyla sesli ifadeleri tanımaya çalışmıştır (Üstün, 1997). DFT spectral özellikleri 16 gruptan oluşturulmuştur. LPC için 12 katsayısı vektörü eklenerek 28 her bir çerçevenin veri gurubu ile ses verilerini işlemiş ve YSA eğitim ve test için kullanmıştır. Bu çalışmada tek kişiden alınan ses örnekleri ile eğitim ve test verisi hazırlanmıştır. YSA modeli olarak Çok Katmanlı Algılayıcı kullanılmıştır. Bu durumda %97.5 başarı sağlanmıştır.

Can yapmış olduğu araştırma kapsamında geliştirdiği sesli ifade tanıma sisteminde hece tabanlı bir sınıflandırma çalışması gerçekleştirmiştir (Can, 2007). Bu sınıflandırma işlemi için çok katmanlı algılayıcıları kullanmış ve seslerin özelliklerini MFC yöntemi kullanarak çıkarmıştır. MFC yönteminde 13 katsayıdan yararlanılmıştır. Deney, sürekli

sesli ifadeler için ve sürekli sesli ifadelere dayalı hecelerin sınırları belirlenmiş şekilde olmak üzere iki farklı şekilde yapılmıştır.

Çizelge 2.1 Sürekli sesli ifade sinyali ile farklı heceler için farklı özellikteki nöron ağları ile kullanımı.

Yöntem	ÇKA	ÇKA	ÇKA	ÇKA
Taninan farklı hece sayısı	42	42	42	55
Sınıflandırma yüzdesi	%62.5	%38.8	%65.6	%31.5
Katman sayısı	4	5	3	6

Çizelge 2.2 Farklı sayıda ayırık heceler için farklı özellikteki nöron ağları ile kullanımı

Yöntem	ÇKA	ÇKA	ÇKA	ÇKA	ÇKA	ÇKA
Taninan farklı hece sayısı	5	10	15	20	25	30
Eğitim kümesi boyu	134	249	399	537	692	833
Deneme kümesi boyu	50	66	82	88	84	85
Sınıflandırma Başarı yüzdesi	%90	%80.3	%65.8	%59.1	%48.8	%33
Ara katman sayısı	1	2	1	3	3	4

Çizelge 2.1 ve 2.2 bu iki durum için yapılmış olan test sonuçları gösterilmiştir. Tablolardan da anlaşılacağı üzere tanınması istenilen hece sayılarının artması ile performansın azlığı görülmektedir. Bu durum daha karmaşık YSA modellerinin gerekliliği şeklinde yorumlanmıştır (Can, 2007).

Dede yapay sinir ağıları kullanarak yapmış olduğu ses tanıma sistemi çalışmasında MFC yöntemi kullanarak sıfırdan dokuza (0-9) kadar olan sayıları seslendirmiş, daha sonra bu seslerin özelliklerini çıkararak üç farklı YSA modeli ile teste tabi tutmuştur (Dede, 2008). 16 MFC özellik katsayısının kullanıldığı bu çalışmada geri yayımlı ÇKA, Elman ve Olasılık Sinir Ağları ile testler yapmıştır. Sistem konuşmacı bağımlı bir sistem olup metine bağımlıdır. Yapış olduğu test sonucunda sırasıyla %98.7, %99.3 ve %100 olmak üzere başarı yakalamıştır.

Yavuz ve arkadaşı tarafından Olasılık Sinir Ağı ile Türkçedeki sesli harfleri tanıma üzerine bir çalışma gerçekleştirilmiştir (Yavuz ve Topuz, 2010). Çalışmada Olasılık Sinir Ağları tarafından Türkçedeki ünlülerin tanınması bir spektral analiz yöntemi kullanılarak gerçekleştirilmiştir. Hoparlörlerden elde edilen telefonların güç spektral yoğunluğu tahmin edilmiştir. Daha sonra ağırlıklandırılan güç spektrumu, o telefona ait güç spektral yoğunluğunun bir dizi band-geçiren filtre uygulandıktan sonra hesaplanmıştır. Bu yolla, hoparlörlerden elde edilen telefonların tahmini güç spektrumları Mel-ölçeğine yaklaşmıştır. Elde edilen Mel-ölçekli katsayıları söz konusu telefonun özellik vektörlerini oluşturmuştur. Bunun sonucunda ilgili konuşmacıya ait özellik vektörlerinden o konuşmacının veri tabanı oluşturulmuştur. Bu çalışmada 20 konuşmacı (10 kadın ve 10 erkek olmak üzere) bulunmaktadır. Konuşmacılara ait özellik vektörleri 27 katsayıdan oluşmaktadır. Çalışma konuşmacı-bağımlı bir çalışma olup testler sonucunda sesli harflerin ortalama %95 oranında doğru bir şekilde tespiti yapılmıştır.

3. MATERYAL ve YÖNTEM

3.1 Yapay Sinir Ağları

Yapay sinir ağı insan beyninin bilgiyi işleme şeklinden esinlenmiş bir örgü tanıma metodudur. Genel bir tanımla yapılması gerekirse, YSA: Basit (*genellikle uyarlanabilir*) elemanların büyük ölçekli bir birine bağlı ağları ve bu ağların hiyerarşik organizasyonları, biyolojik sinir sisteminin yaptığı gibi gerçek hayattaki nesnelerle etkileşime girmesini amaçlar (*Haykin, 1999 ; Kohonen, 1987*). İnsan beyni birçok katmanda trilyonlarca (yaklaşık olarak 10^{11}) nörondan oluştuğuna düşünülmektedir (*Sağıroğlu ve ark., 2003*). Bu nöronlar bilgi öğrenme üzerine özelleşmişlerdir. Bu nedenle, insanlar gördükleri dünyayı yorumlamada oldukça iyidirler. Bu bağlamda el yazısı tanıma basit bir örnektir. Yıllar boyunca farklı el yazılarıyla ayarlanmış, ortalama bir insan beyni farklı insanların el yazılarını kolaylıkla anlayabilecek kapasitededir. Mesele bunu bir bilgisayar programına uyarlamaya gelince, akla gelen makul yaklaşım harfleri ve karakterleri modellemek olacaktır. Fakat bunu söylemesi yapmasından daha kolay elbette, çünkü farklı kişilerin farklı el yazı varyasyonları doğru ve keskin bir model bulunmasını zorlaştırıyor. Bu sebep bilgisayarların başarı oranını insanlarınkine kıyasla oldukça düşürüyor.

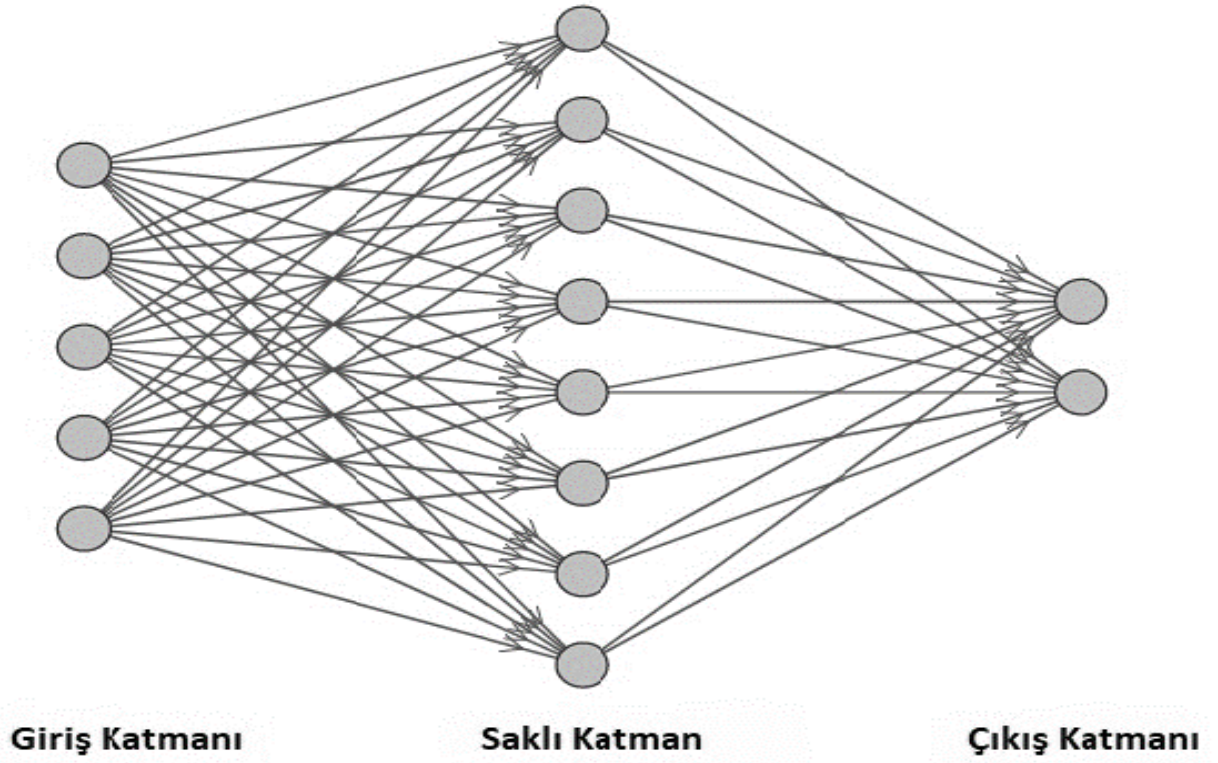
YSA'lar bu probleme yaklaşımı da insan beyninin yaklaşımı benzerdir. Her bir harfi giriş olarak (*ya da eğitim verisi*) atar, yapay nöronların giriş katmanı, saklı katmanı (*isteğe bağlı olarak katmanlar*) ve çıkış katmanını oluşturur ve daha sonra her bir eğitim verisi ile ağı ayarlamaya başlanır. Bir kayıp fonksiyonu (*Loss veya Cost Function*) tahmin edilen çıkış ile istenilen çıkış arasındaki hatayı belirlemek için seçilir. Eğitimin amacı bu kayıp fonksiyonunu minimize etmektir.

Örüntü tanımadaki yapay sinir ağlarının icadı 1940'lara kadar uzanmaktadır. YSA ile ilgili kayda değer gelişim ve araştırmaların çoğunun yakın zamanda gerçekleştirildiğini göz önünde bulundurursak bu kulağa biraz garip gelebilir. Fark edildiği üzere fikir bilgisayarların gelişinden önceye dayanmaktadır. 1943'de ilk yapay sinir hücresi nörofizyolog Warren McCulloch ve mantıkçı Walter Pitts tarafından üretilmiştir

(McCulloch, 1943). Basit bir YSA elektrik devresi kullanarak modellemişlerdir. Bilgisayarların geliştirilmesiyle birlikte ilk kuramsal sinir ağı 1950’de geliştirildi. Algılayıcı (*perceptron*) Frank Rosenblatt tarafından 1958’de oluşturuldu(Rosenblatt, 1958) ve ilk geri yayımlı algoritma Paol Verbos tarafından 1975’de bulundu(Werbos, 1974). Her iki çalışmanın detayları aşağıda ifade edilmiştir.

Gerçek şu ki başlangıçta sunduğu iyi sonuçlar ve gelecek vadetmesine rağmen YSA 1970’lerde gözden düşmüştü. YSA’nın öğrenme için hesaba dayalı gereksinimleri (*dönemin bilgisayar teknolojisini de dikkate alırsak*) mantıksız blunmuştu. Dahası YSA’yı yeterli bir şekilde eğitmek için gerekli olan eğitim verisi çoğunlukla o zamanda mevcut değildi ya da elde edilemiyordu. Diğer bir problem ise ağ parametrelerinin başlangıç değeri yakınsama süresini oldukça etkiliyordu ve ilk kullanıma hazırlanmasının nasıl gerçekleştirileceğini gösterebilecek temel bir kural yoktu. Sonuç olarak da bu alana yapılan yatırımlar azaldı ve diğer örgü tanıma metotlarına yönelim oldu, doğal olarak da YSA araştırmaları yavaşladı.

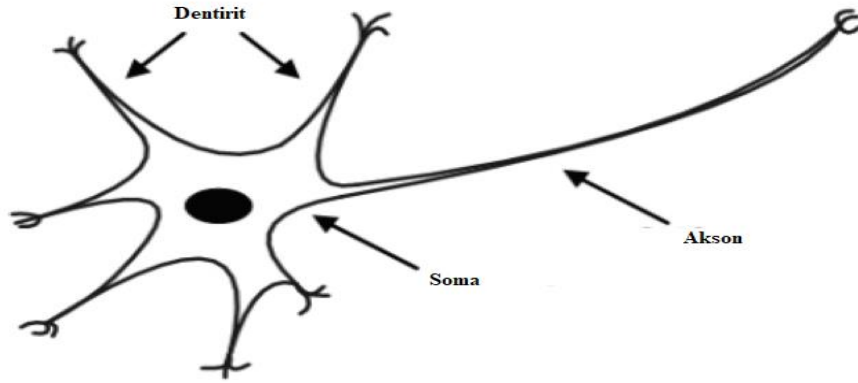
2000’li yıllarda, YSA alanındaki çalışmalar bilgisayarlı teknolojinin gelişmesi ve yeni öğrenme algoritmalarının bulunmasıyla tekrardan ilgi odağı olmaya başladı. 2006 yılında basit ilklendirme (*initialization*) problemi Derin inanç ağları (*Deep belief networks*) için tanımlanan öğrenme algoritması aracılığıyla başarılı bir şekilde ele alındı (Hinton, 2006). Bu ağ denetimsiz ön-eğitimi (*pre-training*) kullanarak ağ parametrelerini ilk kullanıma hazırladı. Bu günlerde çok katmanlı YSA’ları çok geniş eğitim verisi ile eğitilebilmekte ve en son teknolojik bulgular birçok örgü tanıma çalışmasından elde edilmiştir.



Şekil 3.1 İleri Beslemeli YSA'nın şematik gösterimi

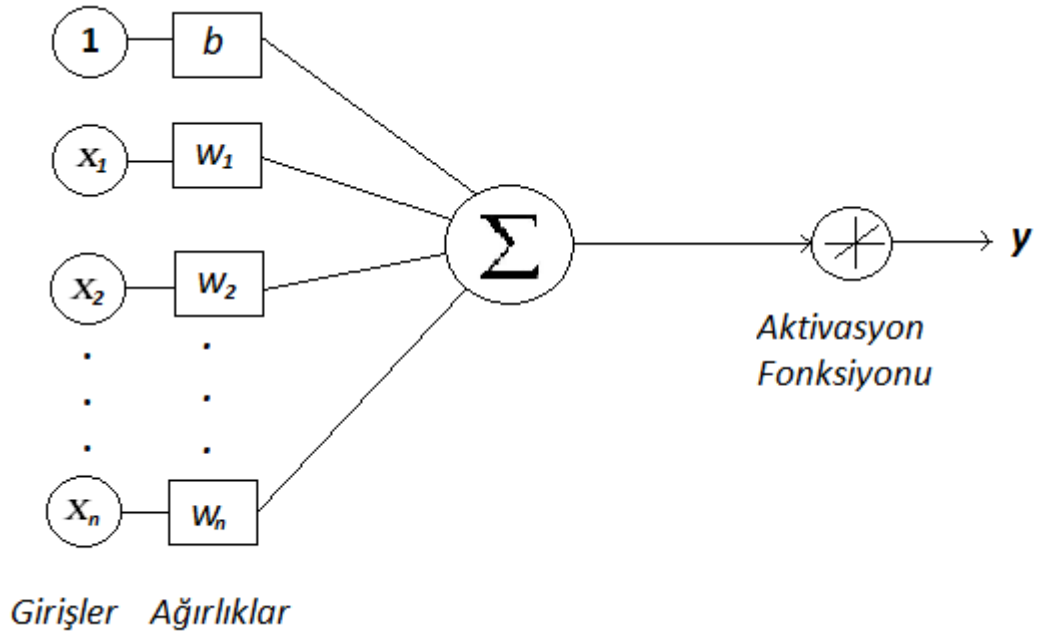
3.1.1 Nöron

Yapay nöronlar (*boğum yada birim olarak da adlandırılırlar*) YSA'ların temel elemanlarıdır. Biyolojik nöronlar sinapslar aracılığı ile bir birlerine bilgi iletirler. Bir nöronun girişindeki uyarıcı miktarına göre iletme geçip geçmeyeceği belirlenir. Şekil 3.2'de biyolojik bir nöronun kısımları ile beraber gösterilmiştir. YSA bu olayı nöronların bağlantıları arasındaki yapay bir nörona girişleri ağırlandırılarak modeller (*Krenker ve ark. 2011*).



Şekil 3.2 Biyolojik nöron ve kısımları (Krenker, 2011)

Giriş katmanındakiler hariç, her nöron bir önceki katmandaki nörondan girdi alır ve bir sonraki katmanın nöronu için bir çıkış oluşturur. Her bir giriş uygun bir ağırlık parametresiyle ağırlıklandırılıp daha sonra toplanır. Belirlenen fonksiyona bağlı olarak yapay nöron bir çıkış oluşturur. Bu fonksiyona aktivasyon fonksiyonu denir.



Şekil 3.3 Tekli bir algılayıcının şematik gösterimi

3.1.2 Algılayıcı (Perceptron)

1958 yılında Frank Rosenblatt tarafından geliştirildi. Algılayıcı (*perceptron*) yaygın olarak kullanılan bir yapay nöron biçimidir. Bir algılayıcı birden fazla girişten oluşabilir (x_i), bu girişler ağırlıklarla (w_i) çarpılarak toplanır ve *bias* (b) değerine göre çıkışa aktarılır.

Bir algılayıcının çıkış değeri aşağıdaki gibi hesaplanır;

$$y(x) = \begin{cases} 1, & \sum_{i=1}^n w_i x_i + b > 0 \\ 0, & \sum_{i=1}^n w_i x_i + b \leq 0 \end{cases} \quad (3.1)$$

Burada n , algılayıcıdaki giriş sayısını ifade etmektedir. Şekil 3.3’de ise bir algılayıcının yapısı gösterilmiştir.

3.1.3 Aktivasyon fonksiyonları

Aktivasyon fonksiyonları giriş katmanındaki girdilerin ağırlıklarla çarpılıp toplanmasının ardından net girdiyi işleyerek yapay sinir ağının bu girdiye karşılık oluşturacağı çıktıyı veya çıktıları belirler. Aktivasyon fonksiyonları nöronun çıkış değer aralığını istenilen değere göre sınırlar, bu değerler genellikle $[0,1]$ veya $[-1,1]$ arasındadır.

YSA’larda aktivasyon fonksiyonu genellikle türevi kolay alınabilen ve sürekli olan fonksiyonlardan seçilir. YSA kullanım amacına göre tek ya da çift yönlü olmak üzere farklı aktivasyon fonksiyonları kullanılabilir. Çoğunlukla kullanılan transfer fonksiyonları tanjant hiperbolik veya hiperbolik fonksiyonlardır. Lineer ve doğrusal olmayan transfer fonksiyonlarının kullanılması, YSA’nın farklı problemlere uygulanabilirliğini sağlamıştır (*Sağıroğlu ve ark., 2003*).

Doğrusal fonksiyon

Yapay sinir ağlarının çıkış katmanında kullanılırlar. Gelen girdileri direkt çıkışa aktarır. Şekil 3.4’de karakteristiği gösterilmiştir. Çoğu kez ADALINE olarak da anılan doğrusal aktivasyon fonksiyonu genellikle standart işaret işleme veya istatistiksel regresyon analizlerinde kullanılmaktadır. Formülse ifadesi ise eşitlik 3.2’de verilmiştir. Burada K sabit bir sayıyı temsil eder.

$$y = Kx \quad (3.2)$$

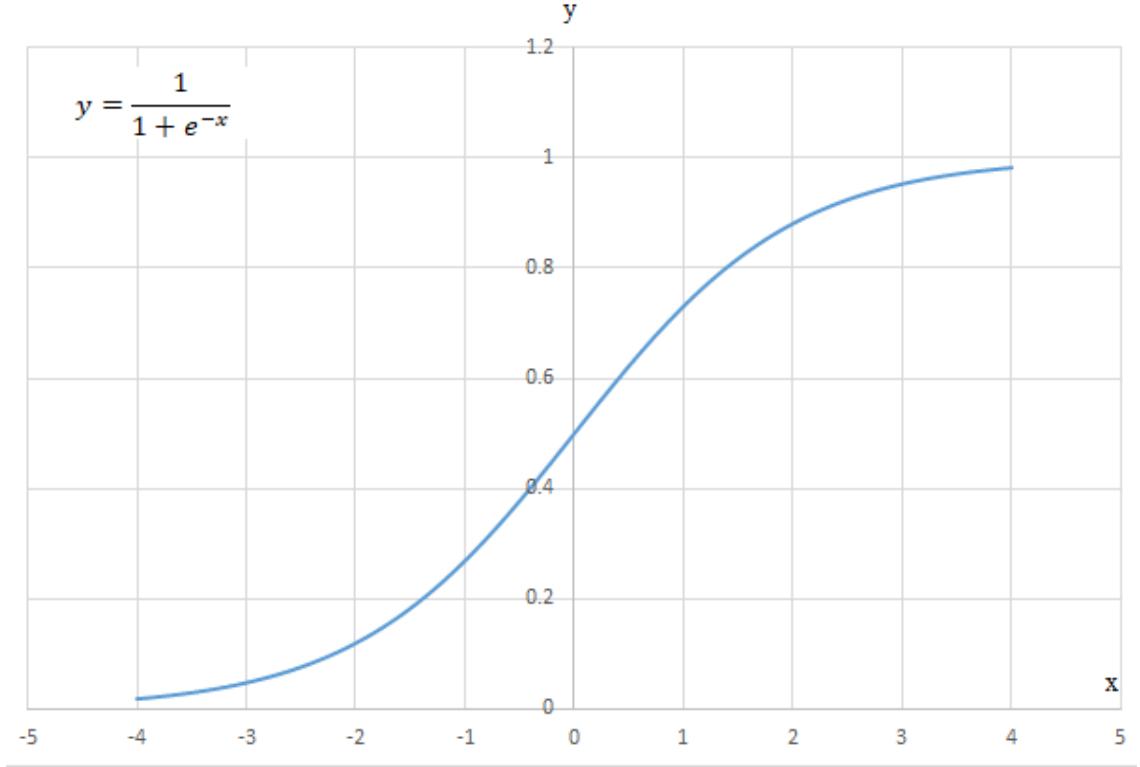


Şekil 3.4 Lineer aktivaston fonksiyon grafiği

Sigmoid

YSA uygulamalarında en çok kullanılan aktivasyon fonksiyonlarından biridir. Fonksiyonun grafiği Şekil 3.5’de gösterilmiştir. Bu fonksiyonun en aktif bölgesi 0.2 ile 0.8 arasındadır. Literatürde tek kutuplu aktivasyon fonksiyonu olarak da bilinir. Formülü eşitlik 3.3’de ifade edilmiştir.

$$y = \frac{1}{1 + e^{-x}} \quad (3,3)$$

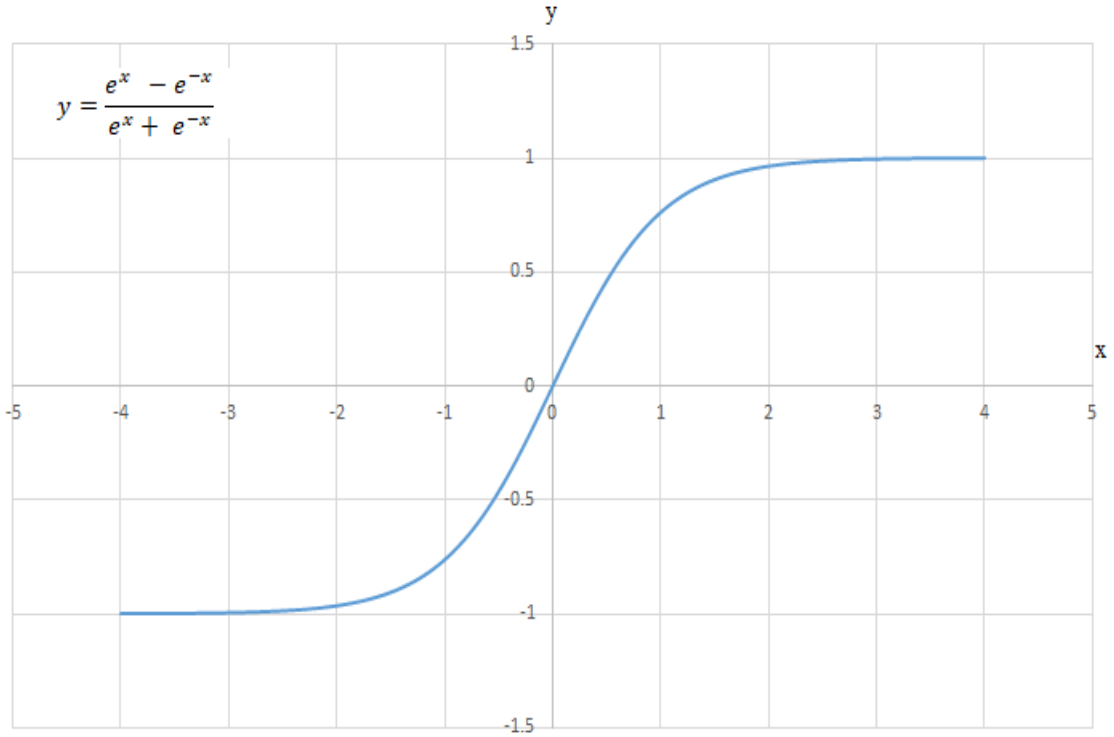


Şekil 3.5 Sigmoid fonksiyon grafiği

Tanjant hiperbolik

YSA uygulamalarında yaygın olarak kullanılan bir diğer aktivasyon fonksiyonu ise tanjant hiperbolik fonksiyonudur. Bir diğer ismi ise çift kutuplu (*bipolar*) fonksiyondur. Şekil 3.6'da karakteristiğini gösteren grafiği verilmiştir ve formül sel ifadesi eşitlik 3.4 gösterilmiştir.

$$y = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3,4)$$



Şekil 3.6 Tanjant hiperbolik Fonksiyon

Softmax

Softmax aktivasyon fonksiyonu YSA’da son zamanlarda kullanılmaya başlanan bir diğer aktivasyon fonksiyonudur. Bu aktivasyon fonksiyonu bir gerçek sayılar vektöründen olasılık dağılımını hesaplar. Softmax fonksiyonu olasılıklar toplamı 1’e eşitken 0-1 arasında değerler aralığında olan bir çıkış üretir. Bunu eşitlik 3.5’de verilen formül ile gerçekleştirir (Nwankpa ve ark., 2018).

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (3.5)$$

Softmax fonksiyonu her bir sınıfın olasılıklarının döndürüldüğü çok sınıflı modellerde kullanılır. Bu fonksiyon çoğunlukla YSA’nın çıkış katmanlarında kullanılırlar.

3.1.4 Katmanlar (Layers)

YSA katmanları gruplaşmış yapay nöronlardan oluşmaktadır. Her bir katmandaki nöronlar bir önceki ve bir sonraki katmana ait nöronla bağlantı halindedir. Yani bir nöronun girişi kendinden önceki bir nöronun çıkışı ve kendisinin çıkışı ise kendisinden sonraki bir nöronun girişi konumundadır. Fonksiyonelliği baz alınarak ifade edilmesi gerekirse üç farklı katman bulunmaktadır:

Giriş katmanı (Input layer) :

Giriş katmanı eğitim verilerinin alındığı yapay sinir ağlarının ilk katmanıdır. Giriş katmanındaki nöronlar pasif olarak ifade edilir. Eğitim verileri üzerinde herhangi bir değişiklik yapmazlar. Görevleri yalnızca kendi çıkışlarına değerleri taşımaktır. Ağırlıklar, biasler ve aktivasyon fonksiyonu bağlamında bakıldığında, giriş katmanında $w=0$ (ağırlık), $b=x$ (bias) ve $f(x)=x$ (aktivasyon fonksiyonu) (Haykin, 1999).

Saklı katman (Hidden layer) :

Saklı katman giriş ve çıkış katmanlarının arasında yer alan katmandır. Görevi ise eğitim verileri arasındaki farkı bulup tasvir etmektir. Saklı katman olarak ifade edilmesinin nedeni ise sadece çıkış ve giriş katmanı olmadığının ifade edilmesi açısından “saklı (hidden)” olarak adlandırılmıştır. Bazı YSA bir, iki veya ikiden fazla saklı katman içerebilir. İki veya ikiden fazla saklı katmana sahip YSA’lar, literatürde derin sinir ağları olarak (DSA) adlandırılırlar. Saklı katmanın tasarımı (nöron sayısı ve aktivasyon fonksiyonu) öğrenme sürecinde önemli bir rol oynar, çünkü saklı katman YSA sisteminin karmaşıklığını belirler. Saklı katman için gerekli parametreler deneme yanılma yoluyla bulunur (Krenker ve ark. 2011).

Çıkış katmanı (Output layer) :

Çıkış katmanı isminden de anlaşılacağı üzere YSA’nın en son katmanıdır. Bu katmanda bir saptama görevi için ağa verilmiş olan eğitim verilerindeki en son olasılık çıkışını veren nöronlardan oluşur. Çıkış katmanındaki nöron sayısı örneğin olası çıkışlarının sayısına

eşittir. Bu olasılık aralığı $[0,1]$ arasındadır. Yani herhangi bir tespit etme problemi için oluşturulan YSA siteminde çıkışlar bu eşik değerine göre sınırlandırılarak iki çıkış verir, yani saptama yapıldı mı (1) yoksa yapılmadı mı (0) (Öztemel, 2003).

3.1.5 Öğrenme yöntemleri

YSA'lar için öğrenme terimi eğitim verisinin işlenerek YSA çıktısının istenen çıktıya adım adım yaklaşması anlamına gelir. Bu yöntem insan beynin öğrenme yöntemlerinden esinlenilerek elde edilmiştir, örnek verecek olursak günlük yaşantımızda bir kavramı anlamak için genellikle bu kavramla ilgili örneklerle bakarak onu öğrenmeye çalışırız yani örnekler bize o kavramı anlamamıza yardımcı olur. Buradaki durum da bu bağlamda benzerlik göstermektedir. Beynin nasıl öğrendiği hakkında bilgimiz henüz başlangıç aşamasında olmasına rağmen öğrenme sürecini nasıl işlediğini temel olarak anlayabiliyoruz. Beynin aktivitelerine bağlı olarak, öğrenme sürecinde beynin sinirsel yapısı değiştiği, sinaptik bağlantıların gücünün arttığı veya azaldığı düşünülmektedir. YSA da her bir örnek için ağırlıkları ve biasleri güncelleyerek bu işlemi taklit eder. YSA için bu şekilde üç farklı öğrenme yöntemi vardır: Denetimli, Yarı-Denetimli ve Denetimsiz öğrenme (Çakır, 2014).

Denetimli öğrenme metodu her bir eğitim örneği bir etiket gurubu ile ilişkilendirilerek kullanılır. Böylece ağa verilen giriş için istenilen çıkışlar temin edilmiş olur. Her bir örnek için istenilen çıktı tahmin edilen çıktı ile karşılaştırarak bir hata ya da kayıp değeri (*Loss value*) hesaplanır. Ağ ağırlıkları ve biasleri öğrenme algoritmasına göre ayarlayarak düzenler ve böylece hata oranı minimize edilmiş olur.

Yarı denetimli öğrenme sistemleri daha çok, az miktarda etiketlenmiş veriler ve büyük miktardaki etiketlenmemiş verilerde kullanılır (Chapelle ve ark., 2006). Bu sistemler özellikle eğitim esnasında etiketlere ihtiyacın kritik olduğu ve test verisinin büyük olduğu durumlarda kullanışlıdır.

Denetimsiz öğrenmede ise, eğitim verisine dair olan etiketler ağda mevcut değildir. YSA yalnızca tahmini çıkışlardan hata oranını belirler. Ağ veriler arasındaki benzerlikleri

öğrenmeye ve onları daha etkili bir şekilde temsil etmeye çalışır. İnsanında görsel sisteminin bu şekilde çalıştığı düşünülüyor. Her ne kadar denetimsiz öğrenme metodu biyolojik perspektiften cazip gibi görünse de hata oranının saptanmasındaki yetersizliği ve modelin gerçek hayattaki problemlere uyarlaması hususunda zorluğu sebebiyle istenilmeyen bir durum oluşturmaktadır. Ayrıca olası çıkışların sayısının bilinmemesi ve eğitim verinin az olduğu durumlarda performansın çoğunlukla düşük olması yöntemin diğer bir olumsuz yönüdür. Denetimsiz öğrenme çoğunlukla kümeleme ve karışım modellerinde kullanılır (Krenker ve ark. 2011).

3.1.6 YSA tipleri

YSA'lar topolojilerine ve öğrenme algoritmalarına bağlı olarak birkaç grup adı altında sınıflandırılabilirler. YSA türleri ve bu türlerin seçimi belli görevlerde performansı önemli ölçüde etkileyebilir. Bu yüzden oldukça önemlidir. YSA'ya olan ilginin son yıllarda artması neticesinde yeni YSA türleri geliştirilmiş olup burada en yaygın kullanılan YSA türleri ele alınmıştır (Krenker ve ark. 2011).

İleri Beslemeli Sinir Ağları

İleri beslemeli sinir ağları (İBSA) en yaygın kullanılan sinir ağı topolojilerinden bir tanesidir. Bu ağlar katmanlardaki düzenlenmiş nöronlardan oluşurlar. İlk katman girdileri alırken son katman ise çıkışları verir. Aradaki katmanlar dışarıya herhangi bir çıkış üretmezler bu yüzden de saklı katman olarak ifade edilirler. Bilgi nöron bağlantıları vasıtasıyla giriş katmanından çıkış katmanına doğru ilerler.

İBSA'ları tamamen bir birine bağlı bir yapıda olabilirler. Tamamen bir birine bağlı bir İBSA, bir katmandaki her bir nöron kendinden önceki ve sonraki nöronla bağlantı halindedir. Tamamen bağlantıda olmayan İBSA'da ise belirli bağlantıları için ağırlığın sıfır olduğu düşünülebilir. Bu ağırlık sembolik olarak gösterimi Şekil 3.1'de verilmiştir.

Denetimli öğrenme İBSA'da kullanılır. Her eğitim örneği bir hedef matrisi ile ilişkilendirilir ve amaç da hedef matrisi ile tahmin edilmiş hedefler arasındaki hata oranını minimize etmektir. İBSA sınıflandırma problemlerinde sıklıkla kullanılırlar.

Öz-düzenleyici Haritalar

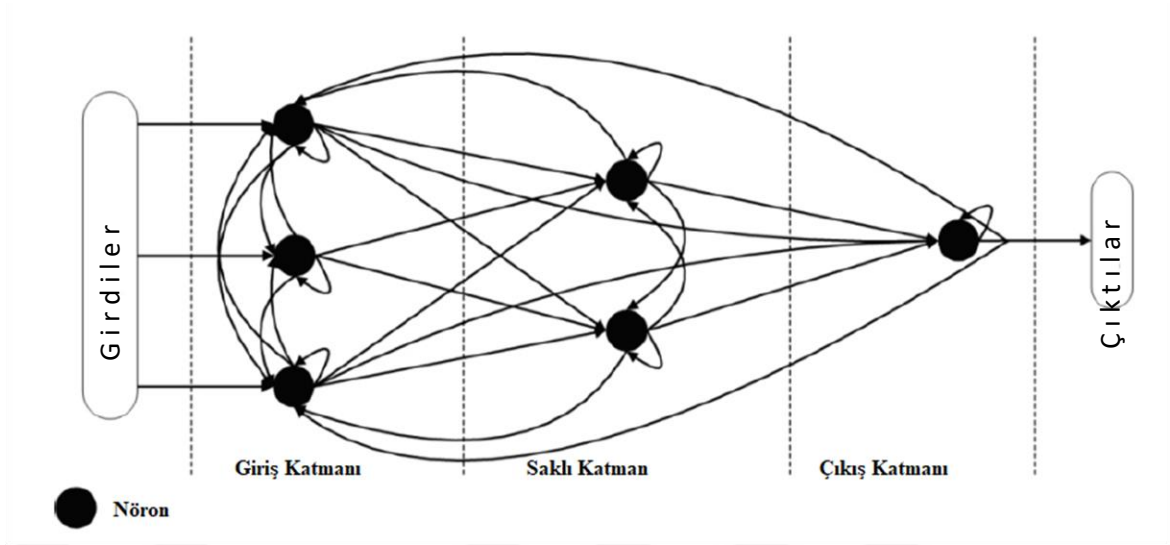
Öz-düzenleyici haritalar(ÖH), nöronları ve ağırlıkları ayarlamak için denetimsiz öğrenmeyi kullanan bir YSA türüdür. Ayrıca, Kohonen haritası olarak da adlandırılan ÖH'ler, yüksek boyutlu girdileri düşük boyutlu (*çoğunlukla iki boyutlu*) ayrı haritalara dönüştürmek için kullanılır. Özellikle kümeleme uygulamalarında kullanışlıdır.

Radyal Temelli Fonksiyon Ağları

Radyal temelli fonksiyon ağları (RTFA), biyolojik sinir hücrelerinin gözlemlenen etki-tepki davranışlarından esinlenilerek geliştirilmiştir. Bu ağ modeli ilk olarak filtreleme probleminde kullanılmıştır (Fix, 1951). Genel olarak ifade edilirse, RTFA çok değişkenli modelleme ve yakınsama problemlerinde kullanılan bir YSA modelidir (Sağıroğlu ve ark., 2003).

Öz-yinelemeli Sinir Ağları

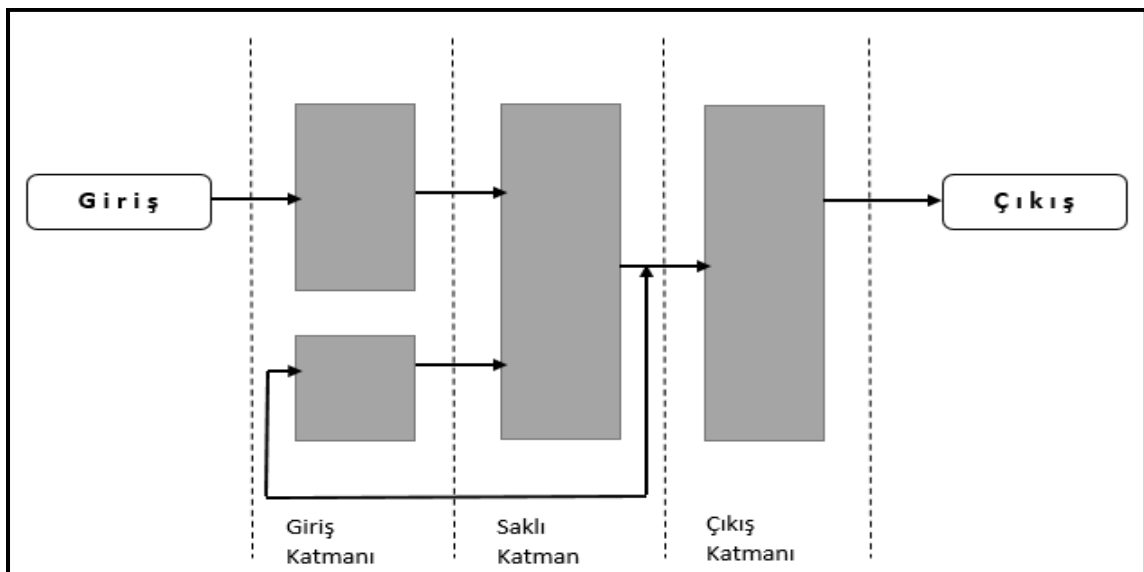
Öz-yinelemeli sinir ağları (ÖSA) nöronlarının çıktılarını bir önceki katmanın nöronuyla paylaşarak bir geri bildirim döngüsü yaratır. Bu da göstermektedir ki ÖSA girdisi olmasa dahi geçici bir süreliğine aktifliğini sürdürebilir. Bu nedenle ÖSA dinamik bir hafızaya sahip dinamik sistemlerdir. İleri beslemeli ağlardan farklı olarak bir geri bildirim biriminin bulunmasıdır. (Mikolov, 2010). Bu ÖSA insan beyninin aktivasyonlarını modellemede önemli bir avantaj getiriyor. Ancak, geri bildirimden dolayı ÖSA yapılarının karmaşıklığı bu ağları İBSA karşısında daha az tercih edilir bir konuma sokuyor. ÖSA genel olarak hesaplamalı sinir biliminde, makine öğrenmesi ve doğrusal olmayan sinyal işleme uygulamalarında kullanılırlar (Lukosevicius, 2009). Şekil 3.7'de ÖSA yapısı kabaca gösterilmiştir.



Şekil 3.7 ÖSA yapısı (Krenker, 2011)

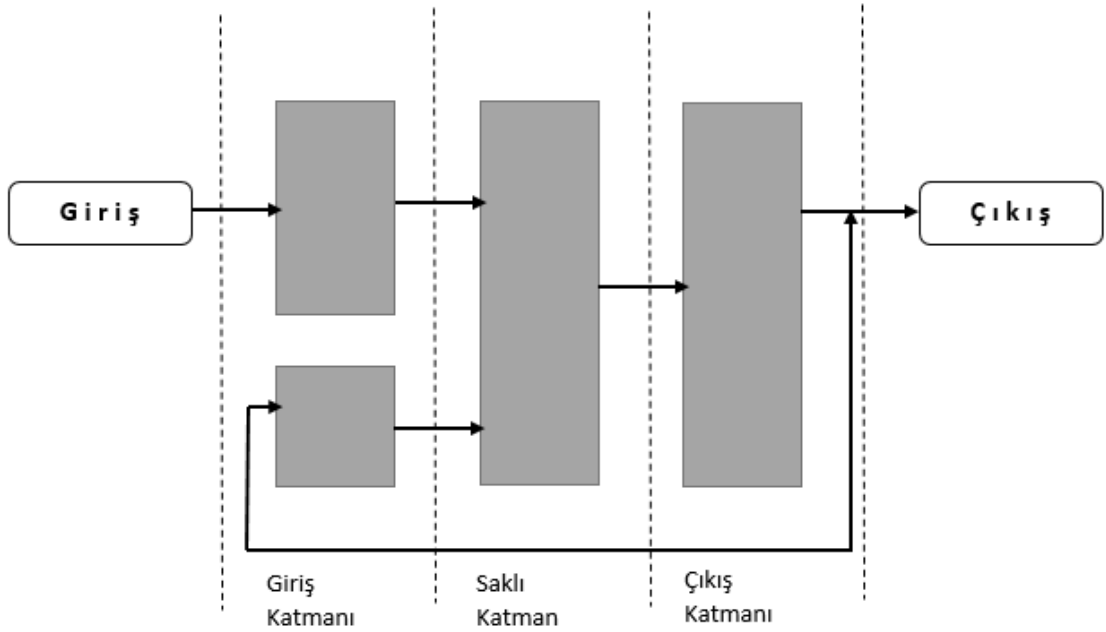
Elman ve Jordan sinir ağları

Elman ağları aynı zamanda basit öz-yinemeli ağlar olarak da ifade edilen özel bir öz-yinemeli yapay sinir ağları yapısıdır. İlk katmanında öz yineli bağlantıya sahip olması bakımından geleneksel iki katmanlı ağ yapılarından farklılık gösterir. Yapısal olarak, gizli katmandan giriş katmanına bağlam birimi olarak adlandırılan birimleri ile geri döngü sağlayan basit bir üç katmanlı yapay sinir ağıdır. Bu tipteki yapay sinir ağı, zamanla değişen örüntüleri saptamaya ve oluşturmaya izin veren bir hafızaya sahiptir.



Şekil 3.8 Elman Ağları şematik gösterimi

Şekil 3.8’de Elman ağlarının yapısal gösterimi verilmiştir. Elman ağlarında saklı katmanda sıklıkla sigmoid fonksiyonu ve çıkış katmanında ise doğrusal aktivasyon fonksiyonu kullanılır. Bu transfer fonksiyonu kombinasyonu eğer saklı katmanda yeterli nöron varsa herhangi bir transfer fonksiyonu ile yakın bir performans sergileyebilir. Elman ağlarının bilgileri depolayabilmesi zamansal ve mekansal örüntü üretebilmesini ve bunlara cevap oluşturmasını sağlayabilmektedir (Krenker, 2011).



Şekil 3.9 Jordan Ağları şematik gösterimi

Jordan ağları ve Elman ağları bezer yapılı ağlardır. Aralarındaki tek fark bağlam birimlerinin saklı katmanda değil, çıkış katmandan beslenmesidir. Şekil 3.9’da Jordan ağının yapısı gösterilmektedir.

3.1.7 Derin sinir ağları

Derin sinir ağları (DSA) literatürde iki veya daha fazla (çoğunlukla ikiden fazla) saklı katmandan oluşan YSA olarak ifade edilir (Hinton, 2006 ; Bengio, 2012). YSA genel hatları itibariyle tek katmanlı bir yapı olarak resmedilir. Ancak, eğitim verisinin yetersiz ve girişlerin yeterince özellik içermediği durumlardaki karmaşık problemlerde bu tek katman yeterli performans göstermeyebilir. Çoklu katman kullanmadaki amaç, düşük

seviye özellikler olarak tanımlanmış veriden yüksek seviye soyut özellikler bulmaktır (Bengio, 2012). Bu yüksek düzeyde soyut özellikler eğitim verisindeki bağımsız dağılımları ayırt etmede yardımcı olur.

Sığ (*shallow*) ağ yapıları yani bir veya iki saklı katmana sahip YSA'ı, girişlerinin iyi temsil edilebilmesi için genellikle çok sayıda nörona ihtiyaç duyarlar. Nöron sayısı arttıkça ağırlıklar ve biaslar gibi ağ parametrelerinin sayısı doğal olarak artacaktır ve bu da ağır bir işlem yükü oluşturur. Bu da birçok değişken içeren büyük boyutlu işlemlerin sığ ağ yapılarıyla etkili bir biçimde temsil edilememesine sebep olur (Bengio, 2011).

Derin öğrenme yapısının kullanılmasını teşvik eden bir diğer unsur da insan beynin çalışmasından ileri gelmektedir. Sinir sinyalinin vücut içerisinde görsel bir bilgiyi iletmesi esnasında, dolaştığı mesafeyi ve zamanı bulmak için bazı ölçümler yapılmıştır. Bu ölçümlerin sonucu göstermiştir ki basit bir obje tanıma işleminde bile bu işleme karışan biyolojik nöronların katman sayısı yaklaşık olarak ondur (Çakır, 2014).

İnsanoğlu içgüdüsel olarak karşılaştığı problemleri belli kısımlara bölerek çözme eğilimindedir. Örneğin bir mimar öncelikle tek katlı bir bina tasarlayarak mesleğin temellerini öğrenir daha sonra ise çok katlı karmaşık binaların yapımına yönelik çalışır, ya da bir program yazıcısı yazacağı programı önce belli başlı kısımlara ayırıp tasvir ederek genel bakıldığında karmaşık olarak görülen bir programın yazımına başlar. Benzer bir başka örnek vermek gerekirse piyano öğrenmeye çalışan biri direkt olarak Mozart'ın senfonileri ile öğrenme girişiminde bulunmaz, öncelikle daha basit melodileri çalarak kendini geliştirir ve sonrasında karmaşık eserleri çalmaya çalışır. Bu tip örnekler derin öğrenme kavramının arka planında yatan motivasyonlardan bazılarıdır. Derin öğrenme mimarileri bir seviyedeki kavramları tasvir eder ve nihai tasviri elde etmek için daha düşük seviyedeki kavramların bir derlemesini oluşturur.

Bilgi sayımsal teknoloji (computational technology) gelişmeler derin öğrenme yapılarının oluşmasında önemli bir etken oluşturmaktadır. YSA'ların çoklu saklı katmanlarla eğitilmesi, YSA'nın içerdiği parametre miktarı sebebiyle bir hesaplama zorluğu oluştuyordu. Gelişmiş paralel hesaplama teknolojileri ile bu zorluk nispeten

kolaylaştırıldı. Hali hazırda 10-15 saklı katmana sahip ağ yapıları yüksek seviyelerde bilgi işleme gücüne ihtiyaç duymaktadır (*Çakır, 2014*).

3.1.8 YSA avantajları

Sinir ağları dağınık modelleridir. Ağırlık ve nöron biasleri bağlantıları sistemin genel girdi-çıkıtı yanıtını verir. Her bir nöron oldukça karmaşık olan yapının bir parçasını oluşturur. Belirli bir giriş için bazı nöronlar fail haldeyken bazı nöronlar pasif davranış sergiler. Farklı uyarılara nasıl cevap verileceğinin öğrenilmesi işlemi YSA'ya doğrusal olmayan girdi-çıkıtı eşleştirmelerini ifade etmesine olanak sağlar.

Sinir ağları geniş çapta örgü tanıma problemlerini icra etmesi için eğitilebilirler. Ağ bir kez eğitildiğinde ağırlıklandırılmış bağlantılar işlem görmemiş girdiden bilgi edinir. Bağlantı ağırlıklarının farklı kombinasyonları, her bir girdi için farklı cevaplar oluşturarak, sinir ağının çeşitli eşleştirme problemlerini temsil etmesini sağlar.

Sinir ağlarının önemli özelliklerinden bir tanesi de girişlerinin kapsamlı bir biçimde betimlenmesine gereksinim duymamasıdır. Karmaşık girdi-çıkıtı ilişkilerini iyi temsil edebilme kabiliyetlerinden dolayı YSA'nın girişlerini kompleks dönüşümlerle ön izleme tabi tutmaya gerek yoktur. Ayrıca YSA girdi ve çıktıların benzer karakterde olmasına da ihtiyaç duymadan işlem yapabilme kabiliyetine sahiptir (*Çakır, 2014*).

3.1.9 YSA dezavantajları

YSA dezavantajları arasında belki de en önemlilerinden biri de bir işlemi yerine getirecek nöron sayısını önceden kestirilememesidir. Gerekli olan nöron sayısı deneme-yanılma yolu ile belirlenir. Saklı katmanda yer alan nöronların yetersiz sayıda olması girdi-çıkıtı arasında makul bir düzen kuramaz yani öğrenme işlemini tam anlamıyla yerine getiremez, literatürde buna yetersiz uydurma (*underrfitting*) denir (*Dede, 2008*). Öte yandan saklı katmanlarda gereğinden fazla nöron kullanılması YSA'nın belirli bir eğitim verisi üzerinde iyi çalışırken farklı bir veride başarısız olmaktadır. Bu durum ise ağ öğrenme

yapmamış, verileri ezberlemiştir şeklinde ifade edilir. Bu da aşırı uydurma (*overfitting*) olarak ifade edilir.

3.2 Ses İşleme ve Özellik Çıkarma

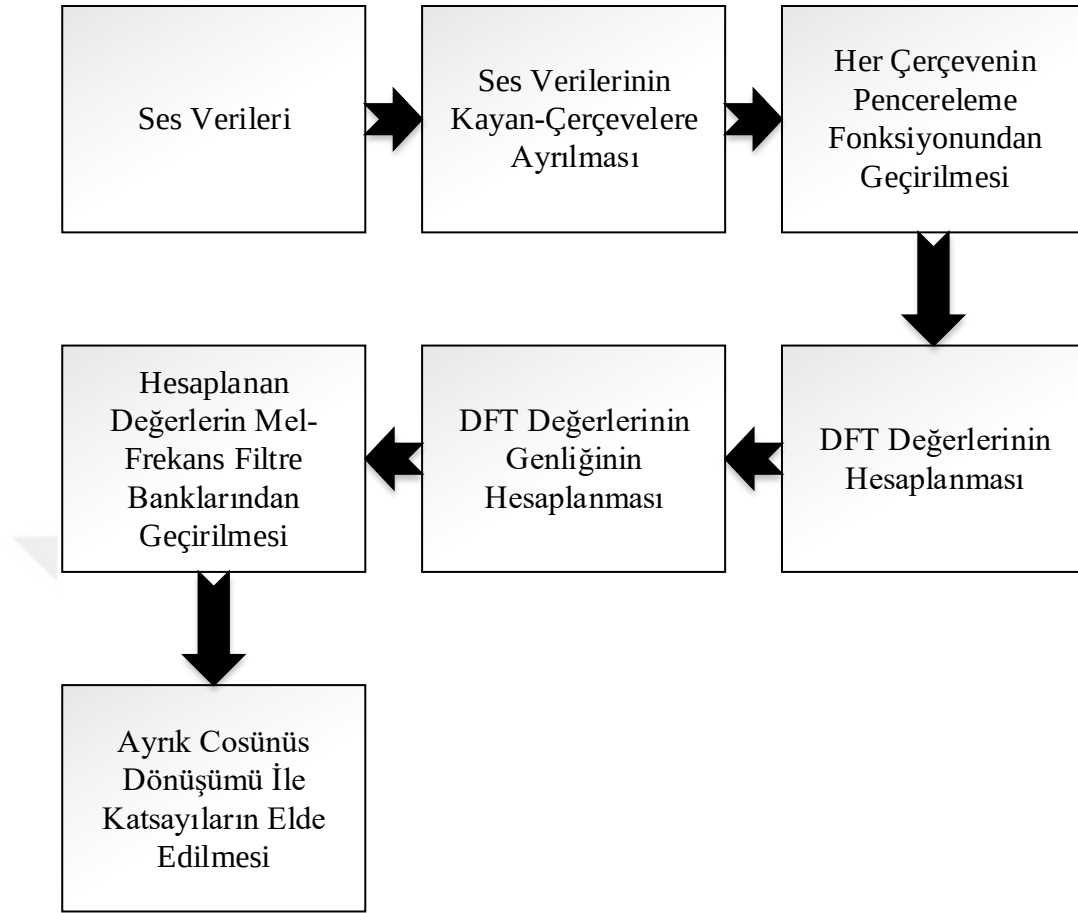
Ses işleme, ses sinyallerinin sayısallaştırarak bilgisayar ortamında sayısal metotlarla işlenmesi sürecidir. Bu süreç seslerin kaydedilip bilgisayar ortamına aktarılmasıyla başlar. Bu aşama içerisinde yapılan işlemler seslerin numerik olarak en iyi bir biçimde ifade etme çabasıdır.

Ses tanıma, son yarım yüzyılda oldukça popüler ve geniş çalışma alanı olan temel bir örüntü tanıma problemidir. Bu problem genel anlamda ses dosyalarının kayıt edilmesi işlemiyle yani sesin sayısallaştırılmasıyla başlar, kayıtların düzenlenmesi, tanınması istenilen seslerin net ve belirgin bir şekilde ortam gürültüsünden arındırılarak normalleştirilmesi gibi bir takım süreçler içerir. Ses dosyaları sürekli sinüsoydal bir dalga yapısında olup bu yapı içerisindeki belirli karakteristik özellikleri ifade edebilmek adına yeni yöntemler geliştirilmektedir. Bu yöntemlerin hepsinin ortak amacı ses dosyası içerisindeki verileri kendi içerisindeki diğer verilerden ayırıştırıp, kendine has özelliklerini ortaya koymaktır. Bunun için ses tanıma sistemlerinde ses dosyasının özelliğinin çıkarılması oldukça elzemdir (*Muda ve Mumtaj, 2010*).

3.2.1 Mel-Frekans kepstrum katsayıları (MFCC)

Mel frekans kepstrum yöntemi ses tanıma sistemlerinde kullanılan oldukça popüler ve verimli bir özellik çıkarma metodudur (*Meng, 2004*). Bu yöntem insan kulağının duyma işlemini model alınarak elde edilmiştir. İnsanın duyma süreci üzerinde yapılan çalışmalar göstermiştir ki insan kulağı düşük frekanstaki seslere karşı yüksek çözünürlüğe ve doyuma sahiptir yüksek frekanstaki seslere kıyaslandığı zaman.

MFC katsayılarının elde edilmesi süreci içerisinde uygulanan işlem basamakları Şekil 3.10' de gösterilmiştir.



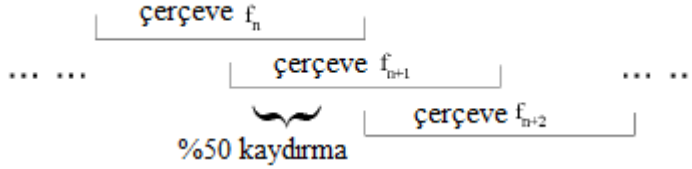
Şekil 3.10 MFC Katsayılarının elde edilme sürecini gösteren diyagram.

Çerçeveleme (Framing)

Ses verileri ve özellikle insan sesi durgun olmayan sinyallerdir. Bu sinyallerin özelliklerinin ortaya konulmasında ilk adım sinyalleri belirli bir uzunlukta anlamlı parçalara ayırma işlemidir. Bu işlem stabil olmayan sinyalin daha stabil sayılacak ve modelinin çıkarmaya elverişli hale getirme de denilebilir (Tiwari, 2010).

Eğer sinyal çerçevesi çok uzun olursa sinyal özelliği çerçeve boyunca çok fazla değişir ve bu durum zaman çözünürlüğünü olumsuz olarak etkiler. Çerçeve çok kısa olursa da kısa-bant bileşenlerin özellikleri feda edileceğinden, bu durum frekans çözünürlüğünü olumsuz etkiler (Joshi ve Cheeran, 2014). Ayrılmış çerçeve sınırlarında veri kaybını önlemek veya en aza indirmek için, mevcut sinyalin özelliklerinden mümkün olduğunca fazla yararlanmak adına kayan çerçeveleme (*overlapping*) işlemi uygulanır.

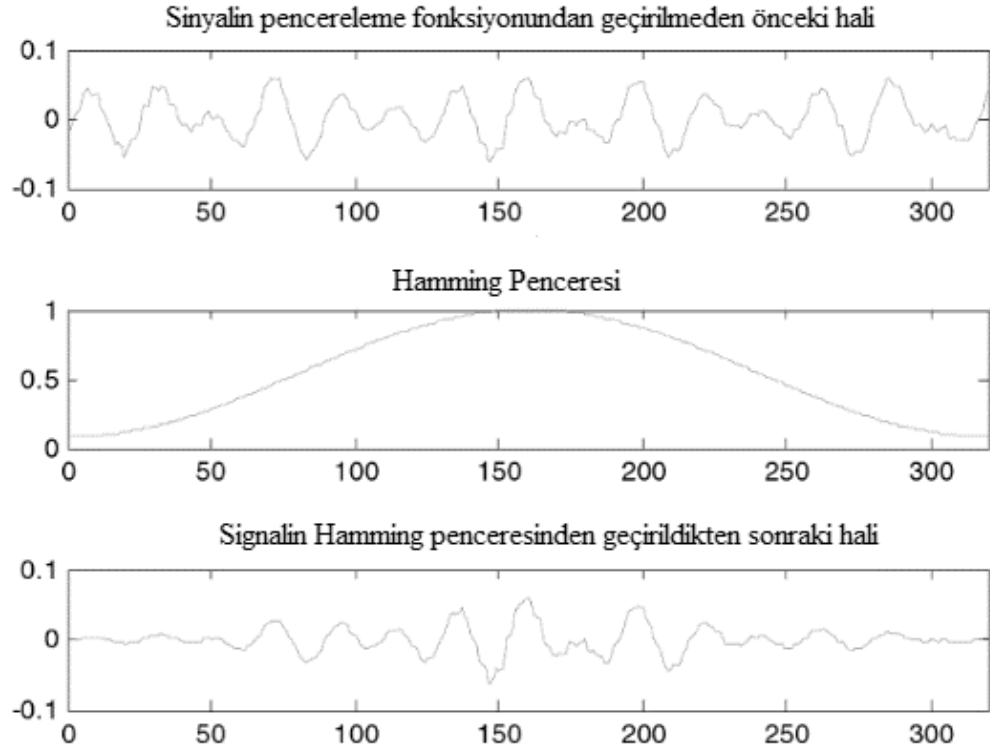
Yani “N” uzunluğunda bir çerçeve olsun (f_n), kendinden sonra gelen (f_{n+1}), aynı uzunluğa sahip diğer bir çerçeve ile aralarında belli bir oranda örtüşme sağlanması durumudur. Şekil 3.11’da bu durum ifade edilmiştir.



Şekil 3.11 Kayan çerçeveleme işleminin gösterimi (*Han ve ark., 2006*)

Pencereleme

Çerçeveleme işlemi sonrasında sinyallerin başlangıç ve son kısımları arasındaki kesinti sinyallerin frekans kümesinde istenmeyen etki bırakır. Yani sıfırdan sinyale ve sinyalden sıfıra ani bir değişim, çerçeve başlangıcı ve sonunda dolayısıyla çerçeveleme işlemi sonucunda oluşur ve bu da yüksek oranda frekans gürültüsüne sebep olur (*Han ve ark. 2006*). Bu olumsuzluğu ortadan kaldırmak için her bir çerçeve serisi bir pencere fonksiyonu ile çarpılır. Şekil 3.12’da bu durumu ifade edilmiştir. Şekilden de anlaşılabacağı üzere örnek sinyalin başlangıç ve bitiş noktalarındaki keskin geçiş pencereleme işlemi sonrasında kısmen ortadan kalkmıştır.



Şekil 3.12 Hamming pencereleme işlemi öncesi ve sonrasını gösteren grafik (Yavuz ve Topuz, 2010).

Litaratürde en yaygın kullanılan pencereleme fonksiyonu Hamming'dir. Bu fonksiyonun matematiksel ifadesi eşitlik 3.6'da verilmiştir.

$$h(n) = \begin{cases} 0.51 - 0.46 \cos\left(\frac{2\pi}{N}n\right), & 0 \leq n \leq N \\ 0, & \text{değilse} \end{cases} \quad (3.6)$$

Ayrık Fourier dönüşümü

Ayrık Fourier Dönüşümü (DFT) zaman tanım kümesinde tanımlı veriyi frekans tanım kümesine taşımayı sağlar. Eşitlik 3.7’de bu işlemi gerçekleştiren matematiksel eşitlik verilmiştir (Joshi ve Cheeran, 2014).

$$x(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, 0 \leq k \leq N-1 \quad (3.7)$$

Bu eşitlikte $x(n)$ parçalara ayrılmış (çerçevenilmiş) giriş sinyali serisinin bir penceresini ifade eder, $x(k)$ ise buna karşılık gelen DFT değerini verir.

Ses sinyali çerçevelere ayrılıp, pencereleme fonksiyonu ile çarpıldıktan sonra DFT değeri hesaplanır. Hesaplanan bu DFT değeri kompleks yapıdadır. Yani hem gerçek ($\Re(x(k))$) hem de imajinel ($\Im(x(k))$) sayılardan oluşmaktadır.

$$|x(k)| = \sqrt{\Re(x(k))^2 + \Im(x(k))^2} \quad (3.8)$$

Ses tanıma sistemleri daha çok gerçek verilerle ilgilenir. Bundan dolayı kompleks değer çoğunlukla ihmal edilir (Meng, 2004 ; Muda ve Mumtaj, 2010). Burada ise eşitlik 3.8 deki ifade ile ikili DFT değerlerinin genliği her bir çerçeve için hesaplanır. Bu sayede elimizde sadece gerçek sayılardan oluşan bir seri elde ederiz.

Mel-Frekans filtre bankası (Mel-frequency filter bank)

Bu adımda, elde edilen DFT değerlerinin Mel-Frekans filtre bankalarından geçirilmesi işlemi izah edilmiştir. İnsana aiyit konuşma sinyalleri DFT’de kullanılan doğrusal frekans ölçeğine uymaz. Bu yüzden ses perdesine eşit uzaklıkta algısal bir ölçek, yani Mel-Ölçeği olarak adlandırılan yöntem seslerin özelliklerini çıkarmak için kullanılır. Mel-Ölçeği Frekansı, insan algısını yansıtan doğrusal frekansın logaritması ile orantılıdır. Burada logaritma kullanılmasının sebebi kulağımız desibelde duyma işlemini ölçümler. Eşitlik

3.9’da doğrusal ölçekli frekansın Mel-Ölçeğini dönüşümünü sağlayan matematiksel ifade gösterilmiştir.

$$Mel(f) = 2595 \log_{10} \left(1 + \frac{f}{700} \right) \quad (3.9)$$

Üçgen bant geçiren filtreler, konuşma sinyalindeki baskın frekans bileşenleri tarafından oluşturulan spektral zarfı çıkarmak için kullanılır. Böylelikle, Mel-frekans filtreler doğrusal frekans eksenini düzensizce ve Mel-frekans eksenine düzgün olarak yerleştirilen, bunun yanı sıra düşük frekans bölgesinde fazla ve yüksek frekans bölgesinde az olacak şekilde üçgen band-geçiren (*triangular band pass*) filtrelerden oluşmaktadır (Meng, 2004). Bu filtrelerin eşik değerleri ve sayısı çizelge 3.1’de gösterilmiştir.

Çizelge 3.1 Üçgen band geçiren filtrelerin frekans eşikleri ve sayısı (Slaney, 1998)

Filtre Sırası	Band-Geçirme Eşikleri (Hz)	Filtre Sırası	Band-Geçirme Eşikleri (Hz)
1. Filtre	133-267	21. Filtre	1618-1856
2. Filtre	200-333	22. Filtre	1733-1988
3. Filtre	267-400	23. Filtre	1856-2130
4. Filtre	333-467	24. Filtre	1988-2281
5. Filtre	400-533	25. Filtre	2130-2444
6. Filtre	467-600	26. Filtre	2281-2618
7. Filtre	533-667	27. Filtre	2444-2804
8. Filtre	600-733	28. Filtre	2618-3004
9. Filtre	667-800	29. Filtre	2804-3217
10. Filtre	733-867	30. Filtre	3004-3446
11. Filtre	800-933	31. Filtre	3217-3692
12. Filtre	867-999	32. Filtre	3446-3954
13. Filtre	933-1071	33. Filtre	3692-4236
14. Filtre	999-1147	34. Filtre	3954-4537
15. Filtre	1071-1229	35. Filtre	4236-4860
16. Filtre	1147-1316	36. Filtre	4537-5206
17. Filtre	1229-1410	37. Filtre	4860-5577
18. Filtre	1316-1510	38. Filtre	5206-5973
19. Filtre	1410-1618	39. Filtre	5577-6399
20. Filtre	1510-1733	40. Filtre	5973-6854

Burada 40 adet aralarında belli bir kaydırma oranı olacak şekilde bir biri ardına yerleştirilmiş ban geçiren üçgen filtreler kullanılmıştır.

Ayrık kosinüs dönüşümü ve katsayıların elde edilmesi

Bu son adımda *Log-mel* spectrumunda bulunan verileri tekrardan zaman tanım kümesine aktarılır. Ayrık kosinüs dönüşümü (AKD) uygulanarak MFC katsayıları elde edilir. Ses sinyalinin Mel-filtre bakalarının çıktılarının logaritmasının hesaplanması sonucunda elde edilen parametreleri gerçek sayılardan oluştuğu için AKD ile kolaylıkla zaman tanım kümesine dönüşümü sağlanır.

$$C_n = \sum_{k=1}^k (\log S_k) \cos \left\{ n \left(k - \frac{1}{2} \right) \frac{\pi}{k} \right\} \quad (3.9)$$

$$n=1,2,3 \dots k-1, K=1,2,3 \dots K$$

AKD'nin matematiksel ifadesi 3.9'da verilmiştir (Tiwari, 2010). Burada n belirtilen katsayı adetini, C_n ise n tane MFC katsayısını ifade eder.

3.3 Ses Tanıma Sistemleriyle İlişkili Bazı Temel Kavramlar

Ses tanıma sistemlerinde literatürdeki çalışmalar incelendiğinde bir takım temel kavramlar göze çarpmaktadır. Bu bölümde bu kavramlardan kısaca bahsedilecektir. Bu kavramlar ses tanıma uygulamalarının değerlendirilmesi kapsamında faydalı olacağı düşünülmektedir.

Ses tanıma sistemleri, isminden de anlaşılacağı üzere sesleri belirli standartlara göre sınıflandırma işlemlerini yerine getiren uygulamalar bütünüdür. Bu sistemler genel hatları itibarıyla seslerin karakteristiğini ortaya çıkarmak için bir takım filtrelerden, özellik çıkarma metotlarından veya ses işleme algoritmalarından geçirilir ve ses verileri istenilen karakteristik özellikleri ortaya çıktığı düşünüldüğü andan sonra bir takım sınıflandırıcılar (*classifier*) aracılığı ile tanımlama işlemi gerçekleştirilir. Bu tanıma işleminden önce tabi belirli bir veri kümesi ile sisteme ses verileri tanıtılır (*eğitilir*). Genel olarak ses tanıma sistemleri ana hatlarıyla bu şekilde ifade edilebilecek işlem basamaklarından oluşmaktadır.

Ses tanıma sistemlerinin ana hatları ile; konuşmacı bağımlılığı, temel alınan ses birimi ve metne göre bir birinden ayrılabilir.

Konuşmacıya bağımlı yada bağımsız olma durumu ses tanıma sisteminde işlenilen sesin tek bir kişiden mi yoksa farklı kişilerden alınan seslerden mi oluştuğu meselesidir. Konuşmacıdan bağımsız sistemler çok fazla kişiden alınan ve işlenen ses verileriyle çalışır ve bu tür sistemlerden farklı kişilerin seslerini tanıması beklenir. Konuşmacıya bağlı sistemlerde ise konuşmacının sesinden farklı bir sesin tanınması beklenmemekte aksi durumda sistemin verimi ciddi bir ölçüde düşmektedir (*Dede, G. 2008*).

Temel alınan ses birimine göre sınıflandırmasında işlenilen sesin kelimelerden mi yoksa fonemlerden mi oluştuğu konusu ile alakalıdır. Fonem, kelimelerin seslendirmesi esnasında ağızdan çıkan seslerin değiştirilmesi ile anlamı değişen en küçük ses birimidir. Kelimeler ile çalışılan ses tanıma sistemlerinde kelimelerin işlenmesi zor ve hassas bir iş olmasına rağmen ses sistemlerinde başarı yüzdesi yüksektir. Fonem tabanlı sistemlerde ise durum bunun tam tersidir. Seslerin işlenmesi kolay fakat ses verilerinin başlangıç ve bitişlerinin ayırt edilmesi zor bir işlem olduğundan sistem performansı kelime tabanlı sistemlere göre düşüktür (*Yalçın, N. 2006*).

Ses tanıma sistemlerinde metine bağımlı ve bağımsız olma durumu, eğitim verisi ve test verisi arasındaki ilişki ile değerlendirilebilir. Burada metne dayalı bir ses tanıma sisteminden söz edildiğinde bu test verisinin eğitim verisi ile sınırlı tutulduğu anlamına gelmektedir. Yani eğitim verisinde kullanılan ses kelime ise test verisinde de bu kelimenin farklı kayıtları ile test edilir. Yani sistem elma ile armut kelimelerine ait seslerin sınıflandırılmasını için önce bu seslerin belirli bir sayıdaki kaydı eğitim için kullanılır ve sonrasında eğitime gönderilmeyen aynı objelere ait ses verileri ile test işlemi yapılır. Metinden bağımsız olma durumunda ise durumlar biraz farklıdır. Bu tür ses tanıma sistemlerinden belirli kelime vektörleri ile ağ eğitimi yapılır (veya fonem) fakat test işleminde eğitim için kullanılan seslerin farklı kombinasyonlarının da sistem tarafından tanınması durumudur. Örneğin “Kırıkkale” kelimesini ele alalım sistemin eğitiminde “kırık” ve “kale” kelimeleri ayrı bir şekilde eğitildiğini ve sisteme “kalekırık” ifadesi ile test edildiğinde sistemin bu ifadeyi tanıması beklenir.

3.4 Materyal ve Çalışma Çerçevesi

Bu tezdeki çalışmada iki YSA modeli ile sesli ve sessiz harflerden oluşan veriler kullanılarak ağ eğitilmiş ve bu testlerde ağ eğitiminde kullanılacak ses verileri MFCC metodu kullanılarak özellikleri çıkartılmıştır. Ağ eğitimi bittiğinde eğitime katılmamış sesli ve sessiz harflerden oluşan test verileri ile bir ön teste tabi tutulmuştur. Bu ön testlerde iki farklı YSA yapısı kullanılmış ve bu YSA yapıları iki farklı özellik çıkarma metodu ile değerlendirmeye tabi tutulmuştur. Bu değerlendirme sonucunda performansı yüksek olan ağlar kelimelerden oluşan ses verileri ile ayrıca bir teste sokulup içerdikleri sesli ifadelerin zamanları tespit edilmeye çalışılmıştır. Tüm bu çalışmalar *Matlab 2018a* programı vasıtasıyla yapılmıştır.

Daha sonra ses verilerinin içerisindeki tespit edilen sesli harflerin zamanları ve bu ses dosyalarının metinleri eş zamanlı olarak göstermek için *Visual Studio 2017* tümleşik geliştirme ortamında (IDE) C# programlama dilinde bir ara yüz programı hazırlanmıştır. Bu programda metin hecelerine ayrılarak, ses dosyası çalınmaya başladığında metne ait ses dosyasındaki sesli harflerin geçişlerini eş zamanlı olarak vurgulanmasını sağlamaktadır. Bu sayede her bir hece içerisindeki sesli harflerin geçişleri, ses okunduğu anda çok rahat bir şekilde gözlenebilir ve ses-metin eşzamanlılığı algılanabilir hale getirilmiştir. Programın içerisinde kullanılan hecelere ayırma algoritması bu bağlamda önemli bir yer tutmaktadır.

Tüm bu çalışmalar orta seviye bir dizüstü bilgisayar üzerinden yürütülmüştür. Çalışmalar esnasında ses kayıtları alımında *WavePad Sound Editor* programı kullanıldı. Bilgisayar ana kart üzerinde kullanılan ses kartı modeli *Realtek High Definition Audio* (sürüm: 6.00.0001.7620) ve mikrofon olarak da *JBL* marka orta seviye kulak içi kulaklık mikrofonu kullanıldı.

Ses kayıtlarının alınması sürecinde, farklı zamanlarda her bir sesli harften (*a, e, ı, i, o, ö, u, ü*) 30'ar ve her bir sessiz harften ise 26'şar farklı kayıt alınmıştır. Kayıtlar 11 025 kHz örnekleme frekansı ve 16 bit PCM parametrelerine göre kaydedilmiştir. Sesler erkek konuşmacıdan alınmıştır. Seslerin kayıt edime sürecinde ortam gürültüsü minimuma

indirgenerek sesin karakteristik özelliklerinin kayda iyi yansması için ayrıca bir çaba sarf edilmiştir. Bu sesin özellik vektörlerinin çıkarılması ve YSA tarafından iyi algılanabilmesi için oldukça önemlidir. Sesli harflerden toplamda 240 adet ses dosyası kaydı yapıldı ve bu kayıtların yaklaşık %34'ü (*her sesli harfi için 10 farklı kayıt*) ağı test etmek için farklı isimlerde adlandırılarak kaydedildi (*Test_a1, ...Test_a10, Test_o1 ...Test_o10* şeklinde). Diğer kayıtlar eğitimde kullanıldı. Aynı şekilde sessiz harflerden toplamda 540 adet ses dosyası kaydedildi ve bu seslerden yaklaşık %39'u (*her bir sessiz harften 10 farklı kayıt*) test için ayrılırken, geri kalanlar ağı eğitimi için kullanılmak üzere ayıldı (*d1, ...d16, Test_d1 ...Test_d10* gibi). Sessiz kendi başlarına anlamlı ses içermediklerinden dolayı, sessiz harfleri yakalamak için ikili heceler ve bazı kelimeler oluşturularak seslendirme işlemi yapılmış ve bunun sonucunda bu seslerden sesli harfler kesilip atılarak geride kalan sessizler, sessiz harf olarak kaydedilmiştir. Örnek olarak verecek olursak "s" harfi "-as, -es, -ıs, -is, -os, -ös, -us, üs" ve "-sa, -se, -sı, -si, -so, -sö, -su, -sü" şeklinde heceler seslendirilip buralardan sesli harfler kesilip atılarak elde edilmiştir. Bunun haricinde örneğin "p" harfinde "kitap" ve "para" kelimeleri seslendirilerek "p" harfinin kelime başına ve sonuna geldiğindeki karakteristiğinin iyi anlaşılabilmesi için bu kelimelerden kesilip alınarak kaydı yapıldı. Bu kayıtların örnek sayıları ve saniye cinsinden uzunluk aralıkları Çizelge 3.2' de gösterilmiştir. Bütün bu kayıt işlemleri farklı zamanlarda alınarak eğitim verisi ayrıca zenginleştirildi. Buradaki amaç özellikle sesli harflerden sesler alındığında seslerin bir birinden olabildiğince farklı ve zengin bir içerik oluşturabilecek karakterde kayıtlarının alınmasının istenmesidir. Bunun yanında YSA'nın sesli harf eğitim verileri daha geniş tutularak sesli harfleri tanımda performansının yüksek olması istenmiştir.

Son aşamada ise rastgele seçilen 10 adet kelimenin senkron bir şekilde vurgulanması işlemi uygulanmıştır. Örneğin seslendirilen kelimelerin içerisinde "merhaba", "Türkiye", "dünya", "cumhuriyet" gibi kelimeler yer almaktadır. Bu kelimelerin ses dosyaları daha sonra eğitilmiş olan ağı verilerek içerdikleri sesli harflerin geçiş zamanları tespit edilmeye çalışıldı.

Çizelge 3.2 Sesli ve sessiz harflerden alınan ses kayıtlarının örnek sayısı ve süre aralıkları.

HARF	ÖRNEK SAYISI ARALIĞI	UZUNLUK(Sn.)	HARF	ÖRNEK SAYISI ARALIĞI	UZUNLUK(Sn.)
A	2603-2946	0.236-0.267	M	716-3581	0.064-0.324
B	474-544	0.043-0.049	N	876-2503	0.079-0.227
C	494-1166	0.044-0.105	O	2353-3214	0.213-0.291
Ç	1243-2555	0.112-0.231	Ö	2351-2983	0.213-0.271
D	513-601	0.046-0.054	P	442-1968	0.040-0.178
E	2351-3151	0.213-0.285	R	1217-2570	0.110-0.233
F	2051-3209	0.186-0.291	S	1239-2800	0.112-0.254
G	472-858	0.043-0.077	Ş	1124-2652	0.101-0.241
Ğ	1084-2530	0.098-0.229	T	803-1476	0.072-0.133
H	574-2089	0.052-0.189	U	1985-3161	0.177-0.281
I	1995-2988	0.181-0.271	Ü	2195-3103	0.199-0.281
İ	2411-3360	0.218-0.304	V	1135-2565	0.103-0.232
J	1175-2564	0.106-0.232	Y	1046-2122	0.094-0.192
K	794-1622	0.072-0.147	Z	1355-2283	0.123-0.207
L	1007-1787	0.091-0.162			

YSA eğitiminde kullanılacak ses verileri *Matlab* programında öncelikle okutulup 440'lık(0.0399 sn.) çerçevelere (*framing*) ayrıldı. Uygun çerçeve boyutu denenerek bulunmuştur. Bu çerçeveler arasında %75'lik (330'luk örtüşme pencere boyu) örtüşme sağlanacak şekilde kaydırmalı çerçeveleme(*overlapping*) yapıldı. Kaydırmalı çerçeveleme işleminin amacı bir ses verisinden mümkün olduğunca çok veri elde etmek ve bu sayede de sesin karakteristiğini iyi bir şekilde ortaya koymaktır. Bundan sonraki adımda art arda gelen iki pencere arasında %75'lik örtüşme sağlanmış çerçeveler *input_a1*, *input_a2*, ...*input_a20* değişkenleri adı altında *Matlab* kod satırında bir matris olarak ayrı ayrı oluşturulmuştur. Daha sonra bu veri matrisleri Şekil 3.12'deki gibi bir araya getirilmiştir. Buradaki *a_set* matrisi olarak ifade edilen matrisi bir birinden farklı bütün "a" sesine ait verilerin sırasıyla bir araya getirilmesinden oluşmuştur. Bu matrisin her bir sütununun uzunluğu çerçeve uzunluğu kadar olup sütunlarının toplam (*N*) sayısı

ise bütün “a” seslerinden elde edilen çerçeve sayısı kadardır. Genel olarak ifade edecek olursak bütün “a” harfleri, belirlenmiş olan çerçeve uzunluğu kadar(440) çerçevelere bölünüp aralarında örtüşme sağlanarak çerçeve sayıları arttırılmış ve tek bir sütunlar matrisine dönüştürülmüştür(sütunlar matrisinin her bir sütunun ilerleme oranı %25, bu da yaklaşık 0.01 sn. tekabül etmektedir). Bu işlem bütün harfler için ayrı ayrı uygulanmıştır(*a_set, b_set, c_set ... z_set*).

$$a_set = \begin{pmatrix} 1. Sütün & 2. Sütün & & N. Sütün \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ 440. satır & . & & . \end{pmatrix}$$

Şekil 3.13 Her bir harfin oluşturulduğu veri setlerinin şematik gösterimi

Bundan sonraki adımda bütün bu veri setleri de bir araya getirilerek *Ham_Giris_Verisi* adı altında Şekil 3.14’ de gösterildiği gibi tek bir matrise dönüştürülmüştür. Bu sayede dağınık olan eğitim verileri tek matris altında derlenip toplanmasıyla eğitime sokmak için daha elverişli bir hale getirilmiştir. Bu işlemler *sesli_sessiz_dosya_yükle.m* Matlab dosyası olarak kaydedilmiştir.

$$Ham_Giris_Verisi = \begin{pmatrix} a_set & b_set & & z_set \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ 440. satır & . & & . \end{pmatrix}$$

Şekil 3.14 Bütün ses verilerinin tek bir matriste birleştirildiği *Ham_Giris_Verisi* matrisinin gösterimi.

Bütün ses verileri tek bir matris altında toplandıktan *MFCC* metodu kullanılarak özelliklerinin çıkarılması aşamasına geçilmiştir. Bu yöntemde kullanılan parametreler Çizelge 3.3’de gösterilmiştir. Bu yöntemlerde seçilmiş olan katsayı adetleri denenerek ve ağ performansı göz önünde bulundurularak seçilmiştir.

Tüm ses verilerinin tek bir matris adı altında toplanmasının ardından ilgili özellik çıkarım yöntemlerinin uygulanması aşamasına geçilmiştir. Bu aşamada *Ham_Giriş_Verisi* adlı matris içerisinde bulunan bütün veriler çerçeve-çerçeve ele alınıp ilgili çerçevenin Çizelge 3.3 de verilen pencereleme fonksiyonundan geçirilip ardından ilgili özellik çıkarım metodu uygulanarak katsayılar hesaplanmıştır. Bu işlemler *ozellik_cikarma.m* adı altında Matlab dosyası olarak kaydedilmiştir.

Çizelge 3.3 Özellik çıkarımında kullanılan parametreler

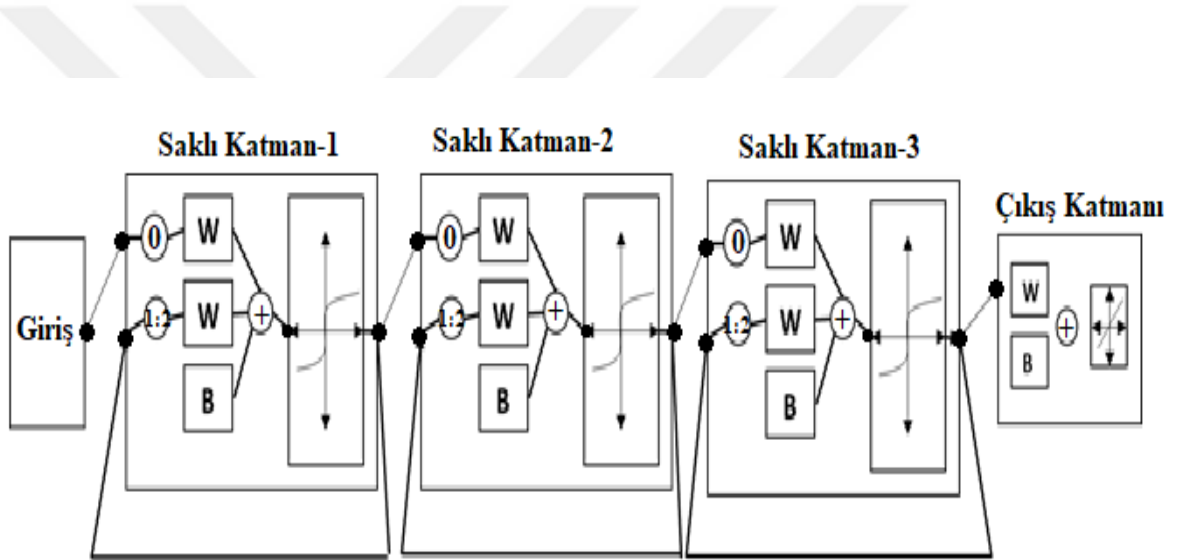
Özellik Çıkarma Metodu	MFCC
Belirlenen Katsayı Sayısı	17-19
Pencereleme Fonksiyonu	Hamming
Pencereleme Fonksiyonunun Uzunluğu	Çerçeve Boyu Kadar (440)
FFT Derecesi	512
Art Arda Gelen Çerçevelerin Örtüşme Oranı	%75
Belirlenmiş Çerçeve Boyu	440 (0.0399 sn.)

Özellik çıkarımı parametrelerinin *Ham_Giriş_Verisi* matrisine uygulanmasından sonra, ilgili ağ eğitimlerine sokulacak olan *Giris_Verisi* adlı matrise dönüşmüştür.

Bundan sonraki adımda iki farklı ağ yapısı ve bu ağlara ait parametrelerin belirlenmesi aşamasına geçilmiştir. Bu aşamada Katman-özyinelemeli sinir ağı (*Layer-Recurrent Neural Network*) ve örüntü tanıma sinir ağı (*Pattern Recognition Neural Network*) adlı iki farklı YSA kullanılmıştır.

3.4.1 Katman-özyinelemeli sinir ağı

Katman-özyinelemeli sinir ağı (KÖSA) dinamik bir YSA modelidir. Bu ağı Matlab, *Neural Network Toolbox* adı altında tanıtılmıştır. Bu ağı Elman Ağlarının gelişmiş bir versiyonudur. Elman ağları ileri beslemeli, kademe gecikmeli ek bir öz-yineleme katmanı eklenmesiyle oluşturulmuş bir YSA modelidir (Elman, 1990). Şekil 3.15’de KÖSA Matlab modelinin şematik gösterimi verilmiştir. Bu gösterim tek saklı katmanlı bir yapının gösterimidir. KÖSA modeli ileri beslemeli ağlara benzer bir yapıda olup farkı her katmanın kendisi ile ilişkilendirilmiş kademe gecikmesine sahip bir öz-yinelemeli yapıda olmasıdır.



Şekil 3.15 KÖSA'nın modelinin şematik gösterimi.

Matlab kod satırında “`net = layrecnet(1:2,neuronN,trainFcn)`” kod satırı ile KÖSA yapısı oluşturulmuştur. Bu ifade “`neuronN`” ifadesi ile katman sayısı ve bu katmanlardaki nöronlar belirlenmiştir. “`neuronN = [25,15,10]`” kod satırı ile bu belirleme işlemi gerçekleştirilmiştir. Saklı katmanların ve çıkış katmanının aktivasyon fonksiyonu aşağıda verilen ilk dört satırlardaki kod satırı ile belirlenmiştir. Buradaki ilk üç satır saklı katmanların, dördüncüsü ise çıkış katmanı fonksiyonunu belirlemektedir. Beşinci satırdaki kod satırı ile eğitim fonksiyonu belirlenmiştir. Son satırdaki ifade ile de verilerin hepsini eğitim için kullanılmasını gerçekleştirmek adına yazılmıştır.

```

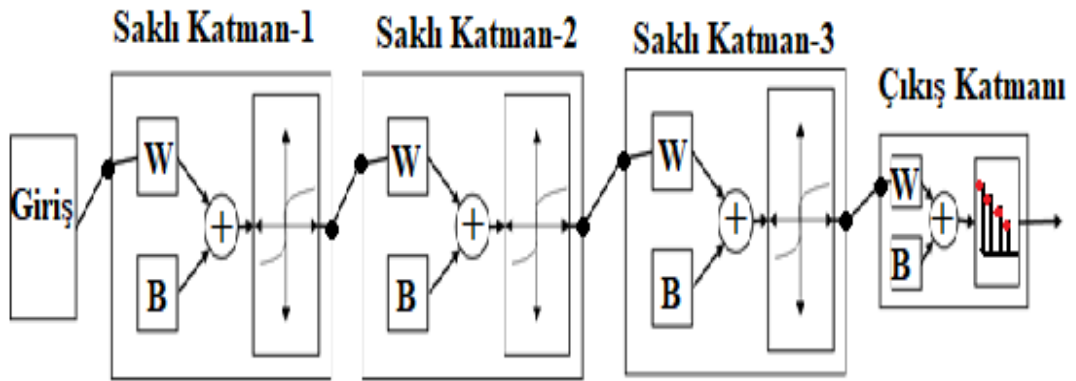
net.layers{1}.transferFcn='tansig'
net.layers{2}.transferFcn='tansig'
net.layers{3}.transferFcn='tansig'
net.layers{4}.transferFcn='purelin'
trainFcn = 'trainlm';
net.divideParam.trainRatio=100/100;

```

Burada üç saklı katmana bir KÖSA modeli tasarlanmış katmanlarda sırayla [25, 15, 10] adet nöron kullanılmıştır. Bu eğitimler sırasında nöron sayıları ve katman sayılarıyla beraber eğitim için kullanılan eğitim fonksiyonu, denenerek tespit edilmiştir.

3.4.2 Örüntü tanıma ağı (Pattern recognition neural network)

Örüntü tanıma ağı (ÖTA) Matlab'ın “Neural Network Toolbox” da tanıttığı, sınıflandırma problemleri için oluşturulmuş bir ileri beslemeli ağ modelidir. Ağın şematik gösterimi Şekil 3.16'da gösterilmiştir. Bu ağ yapısının çıkış katmanında genellikle “softmax” aktivasyon fonksiyonu kullanılır, şekilde de bu ağın çıkış katmanında bunu ifade eden sembol kullanılmıştır.



Şekil 3.16 Matlab' da ÖTA yapısının şematik gösterimi.

“net = patternnet(neuron,trainFcn,performFcn)” komutu ile ağ yapısı oluşturulmuş olup “neuron”, “trainFcn”, “performFcn” parametreleri içeriği aşağıdaki satırlarda belirtilen kodlar ile faaliyete geçirilmiştir. “neuron = [40,20,10]” biçiminde ilk ara katmanda 40,

ikinci ara katmanda 20 ve üçüncü ara katmanda ise 10 tane nöron kullanılmış olduğu anlamında gelmektedir.

```
performFcn = 'crossentropy'
trainFcn = 'trainscg'
net.layers{1}.transferFcn='tansig'
net.layers{2}.transferFcn='tansig'
net.layers{3}.transferFcn='tansig'
net.layers{4}.transferFcn='softmax'
```

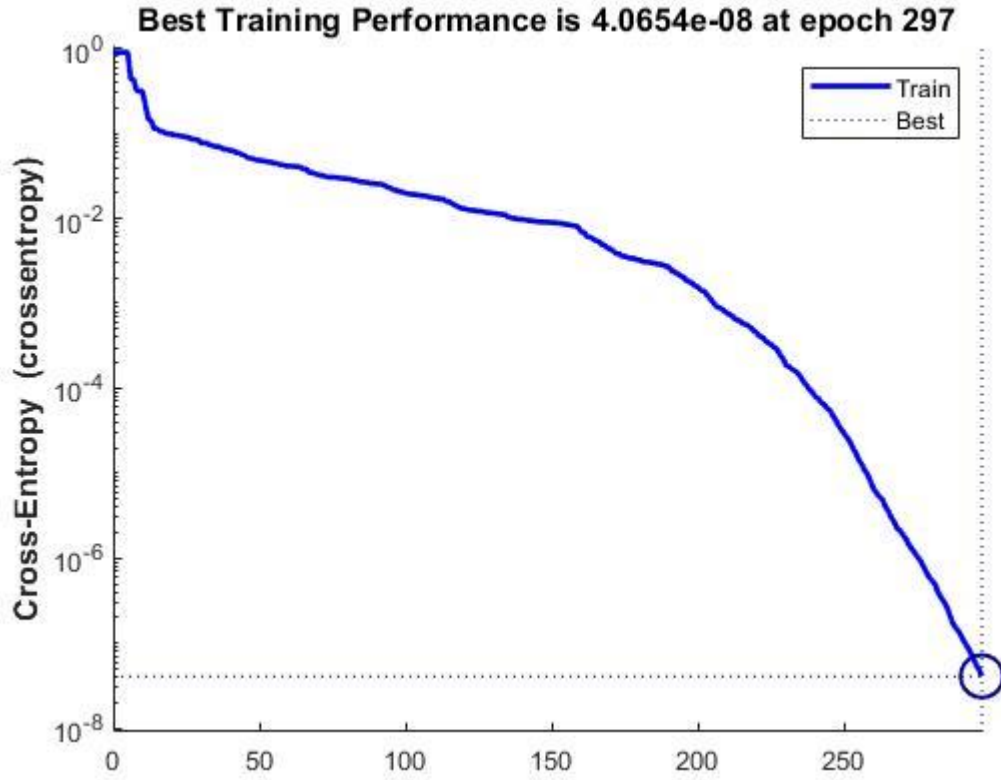
Buradaki ilk satırda yer alan kod performans fonksiyonunu, ikincisi eğitim fonksiyonunu, akabinde gelen ilk iki kod satırları her bir saklı katmanda ait transfer fonksiyonunu belirtmek için kullanılmıştır. Son satırdaki kod ile çıkış katmanının transfer fonksiyonu belirlenmiştir. Buradaki katman sayıları ve bu katmanlara ait nöron sayıları ve ayrıca transfer fonksiyonları da denenerek seçilmiştir.

3.4.3 Hedef (*Target*) matrislerinin oluşturulması

Hedef matrisi ağ yapıları oluşturulduktan sonra ağın eğitimi aşamasında kullanılan bir matristir. Bu matris ağın girdilerinin kendi arasında kaç sınıfa ayrılacağı ve bu sınıfların ilgili girdiler ile ilişkilendirmek için kullanır. Bu çalışmada ağ eğitiminde kullanılmak üzere oluşturulmuş iki farklı hedef matrisi tasarlanmıştır. Bu matrisler sesli-sessiz (2 sınıf) ve sesliler-sessiz (9 sınıf) olacak şekilde oluşturulmuştur. Buradaki birden fazla hedef matrisi dolayısıyla sınıfın kullanılmasındaki amaç ağın performansını daha net görebilmek yani ağın sesli-sessiz ayrımını, sesli harfleri kendi içerisinde ayırt edebilme kabiliyetini de gözlemleye bilmektir.

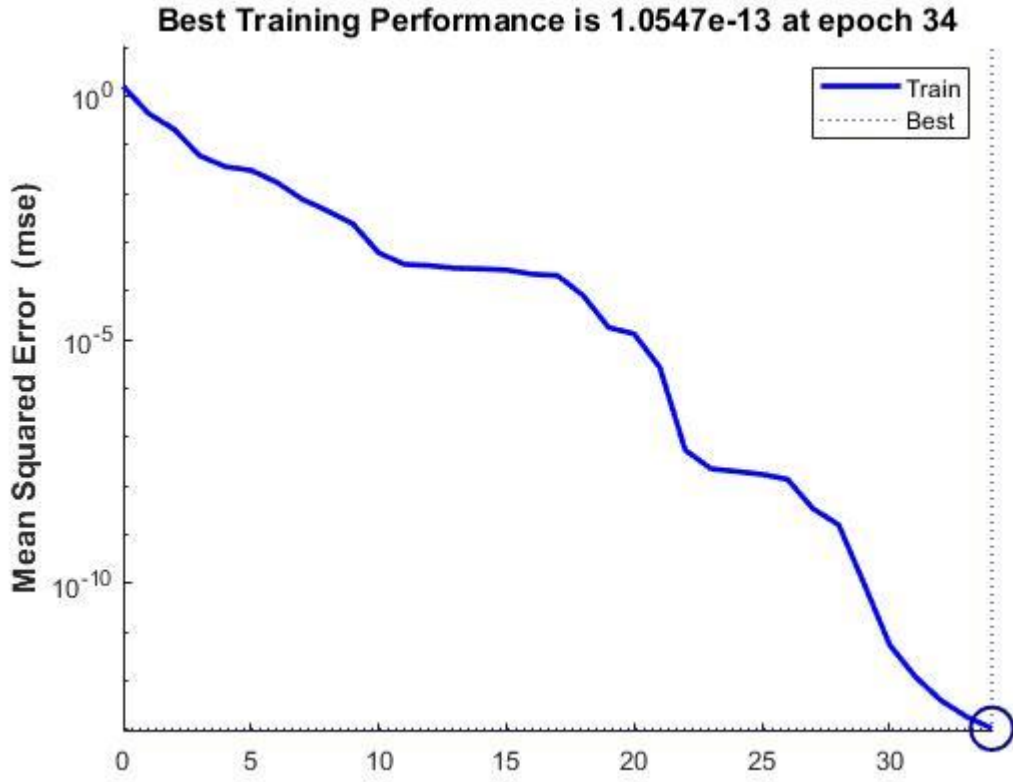
$$\text{Hedef Matrisi} = \begin{matrix} \text{Sesli Harfler} \\ \{ \text{1111111111} \dots \text{111000000000} \dots \text{000} \} \\ \{ \text{0000000000} \dots \text{000111111111} \dots \text{111} \} \\ \text{Sessiz Harfler} \end{matrix}$$

Şekil 3.17 İkili sınıf için oluşturulan hedef matrisi



Şekil 3.19 ÖTA eğitim performans grafiği

Bu kod satırlarındaki birinci satır daha önce oluşturulma aşamalarından bahsedilen iki YSA modeli için eğitimi gerçekleştiren kodu temsil etmektedir. Diğer satırdaki kod satırı ise eğitim sırasında uygulanacak yineleme sayısını belirtilmesi için kullanılmıştır. Bu sayı ağ eğitimi sırasında denenerek belirlenmiştir. Belirlenmesinde ki kıstas ise eğitim sırasında belirli bir denemeden sonra karesel hatanın (*Mean squared error*) yada çapraz entropinin (*Cross-entropy*) sıtabilleşmesi gözlemlenmiş, yani eğitim eğrisinin eğim açısı sıfıra yaklaştığı ya da sıfır olduğu andaki yineleme sayısı temel alınmıştır. Fakat bir çok eğitim daha düşük yineleme aşamasında sonlanmış. Eğitimler iki farklı eğitim fonksiyonu kullanılarak gerçekleştirilmiştir. Bunlar Matlab’da tanımlanan “trainlm” (*Levenberg-Marquardt backpropagation*) ve “trainscg” (*Scaled conjugate gradient backpropagation*) eğitim fonksiyonlarıdır.



Şekil 3.20 KÖSA eğitim performans grafiği

Eğitim işlemleri, her bir özellik çıkarma metodunun belirtilen YSA yapıları ile eğitilip, bu eğitimler bittikten sonra her bir ağ yapısı hazırlanan test verileri ile test edilmiştir. Buradaki test aşamasındaki uygulanan süreç eğitim verisi için de aynen uygulanmıştır. Testler aşamalı olarak gerçekleştirilmiştir. Birinci aşamada ağ her bir harften 10 adet olmak üzere 290 tane ses dosyasını öncelikle sesli-sessiz ve buna ek olarak da sadece gözlem açısından sesliler-sessiz şeklinde, yani sesli harfleri kendi içinde ayırt edilme çalışması gerçekleştirilmiştir.

Bundan sonraki test aşamasında kelime içerisindeki sesli harflerin tespiti için ağa kelime vektörlerinden oluşan veri seti verilmiş ve kelime içerisindeki sesli harflerin tespit edilmesi amacıyla testler yapılmıştır. Kelime bazlı metinler kelime grupları rasgele günlük konuşma dilinden seçilmiştir. Tanıma işleminden başarıyla geçen ağ yapılarından sonraki aşamada sesli harflerin geçtiği zamanı hesaplanmış ve bu zamanlar ve ses dosyalarına ait metinler ayrı ayrı metin belgesi olarak kaydedilmiştir.

Test işlemlerinin ilk aşamasında sesli ve sessiz harfler tek bir matris haline getirilip sessli ve sessiz olarak işaretlenmiştir. Bu aşamadan sonra eğitilmiş olan ağa gönderilerek bazı performans değerleri elde edildikten sonra bu işlem farklı ağ yapıları kullanılarak karşılaştırılmıştır (Sammur ve Webb, 2011). Bu kavramlar sırayla; Doğruluk oranı (*Correct rate*), Hata oranı (*Error rate*), Hassaslık (*Sensitivity*) ve Özel etken oranı (*Specificity*).

Doğruluk oranı

Doğruluk oranı olarak ifade edilen bu parametre test sonucunda sesli harfler ve sessiz harflerin doğru tespit etme oranını vermektedir. Formülüne etmek gerekirse;
“(Doğru tespit edilen tüm sesli harfler + Doğru tespit edilen tüm sessiz harfler) / Tüm Harfler”. Buradaki bu oranın mümkün olduğunca “1” e yakın olması istenir.

Hata oranı

Hata oranı kısaca hatalı yapılan tanımları ifade eder. “1” den doğruluk oranının çıkarılması ile elde edilir.

Hassaslık

Hassaslık, gerçek pozitif oran (*true positive rate*) olarak da adlandırılır. İkili sınıflandırma problemlerinde performansın ölçülmesinde kullanılan bir kavramdır. Bu ikili sınıflandırmada ilk sınıf pozitif, ikinci sınıf ise negatif olarak isimlendirilir (Sammur ve Webb, 2011). Buradaki çalışmada sesli harfler pozitif, sessiz harfler negatif olarak aşağıdaki gibi işaretlidir;

- Gerçek pozitif (*True positive*): Doğru bir şekilde sınıflandırılmış sesli harfler.
- Hatalı pozitif (*False positive*): Hatalı bir şekilde sesli harf olarak sınıflandırılmış sessiz harfler.
- Gerçek negatif (*True negative*): Doğru bir şekilde sınıflandırılmış sessiz harfler.
- Hatalı negatifler (*False negative*): Hatalı bir şekilde, sessiz harf olarak sınıflandırılmış sesli harfler.

Hassaslığı formülize edilmesi gerekirse:

Hassaslık = Gerçek pozitiflerin sayısı / (Gerçek pozitiflerin sayısı + Hatalı negatiflerin sayısı)

Özel etken oranı

Özel etken oranı (*Specificity*) olarak ifade edilen kavram ikili sınıflandırmalarda ikinci sınıfın doğru tespit oranını verir. Yukarıda verilen tanımlar kullanılarak formülize edilecek olursa;

Özel etken oranı = Gerçek negatiflerin sayısı / (Gerçek negatiflerin sayısı + Hatalı pozitiflerin sayısı) olarak ifade edilir (*Chicco ve Jurman, 2020*).

Bu tez çalışmasının son aşamasında ise C# programlama dili kullanılarak bir arayüz programı hazırlanmış ve metin-ses senkronu ve vurgusu bu program aracılığı ile yapılmıştır. Bu programın çalışma mantığı ve içeriği sonraki başlık altında detaylı bir şekilde açıklanmıştır.

3.4.5 Arayüz Programının Oluşturulması

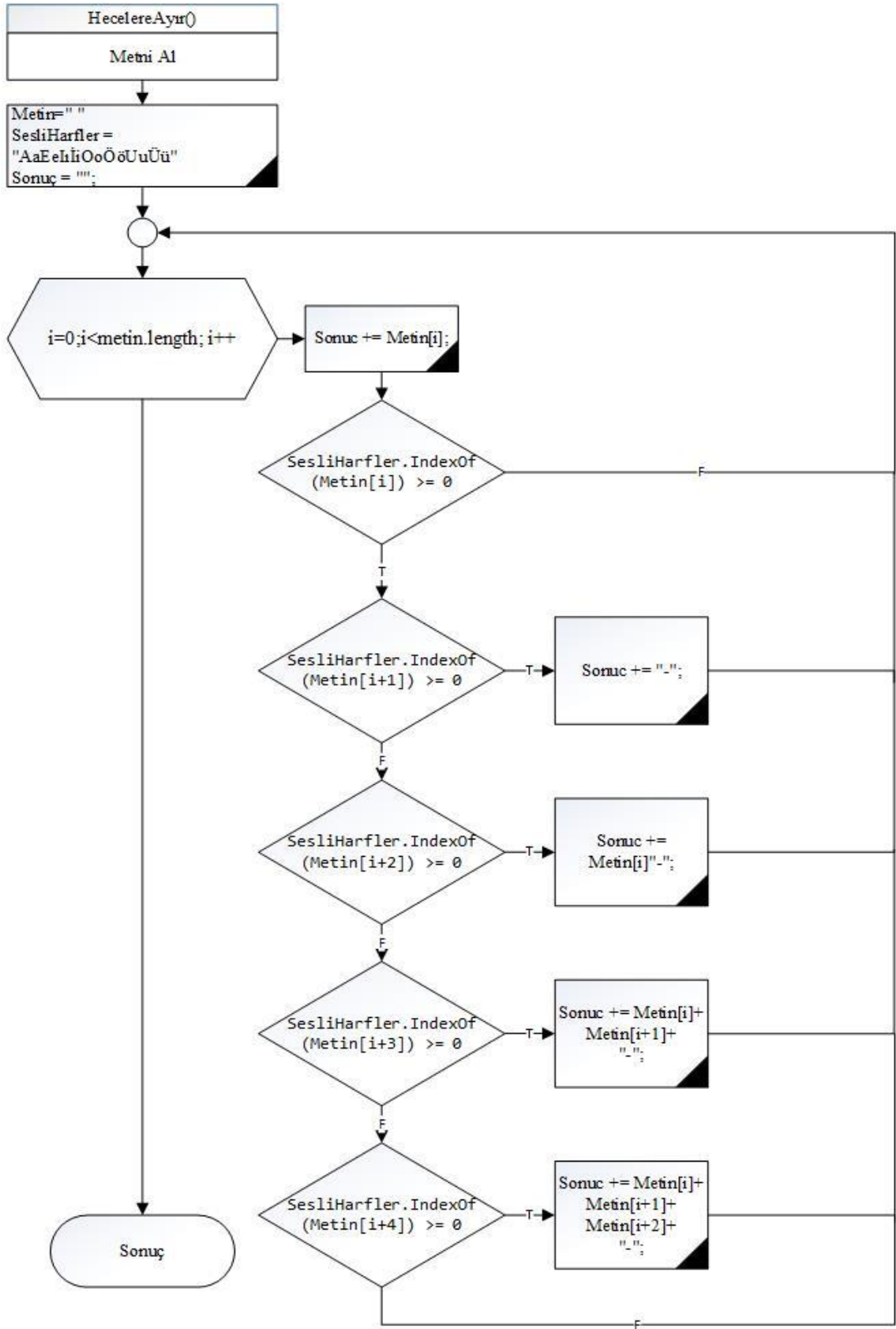
Arayüzün hazırlanmasında C# programlama dili ve programlama ortamı olarak *Visual Studio 2017* 'den (*IDE*) yararlanılmıştır. Bu programda, *Matlab* çalışmalarının sonucunda elde edilen kelimelerin ses dosyalarındaki sesli harf zamanları, sesin metin halinin kaydedildiği zengin metin belgesi okutulup ses dosyası ile beraber oynatılarak her bir harfin geçişi sırasında metin içerisinde vurgulanması işlemi gerçekleştirilmiştir.

Türkçe 'de heceler sesli harf etrafında oluşur. Bu özellikten faydalanılarak sesli ifadelerin zamanlarını daha net ve belirgin bir biçimde ortaya koymak adına programa basit bir heceleme algoritması eklenmiştir. Söz konusu olan ses dosyasına ait metin belgesi arayüz programı tarafından okunması esnasında metin hecelere ayrılarak ekranda gösterilmesi işlemi gerçekleştirilmiştir. Bu sayede sesli harfler ve bu harfler etrafında oluşan hece ve dolayısıyla kelimenin hem görsel hem de yapısal incelemesinde önemli bir fark edilebilirlik oluşturmuştur. Kullanıcı bu sayede hem sesli harflerin geçişlerini net bir

biçimde takip edebilir hem de kelimenin yapısını net bir biçimde gözlemleyebilir. Kelimelerin hecelerine ayrılmasında kullanılan algoritma başarılı bir şekilde Türkçe ‘de yer alan tek, iki, üç ve dört heceden oluşan kelime guruplarını başarılı bir şekilde hecelerine ayırma işlemini gerçekleştirmiştir. Fakat bazı yabancı kökenli olan “*sfenks*”, “*trans*”, “*krank*” gibi beş ve altı harften oluşan kelimelerin heceleme yapılamamaktadır.

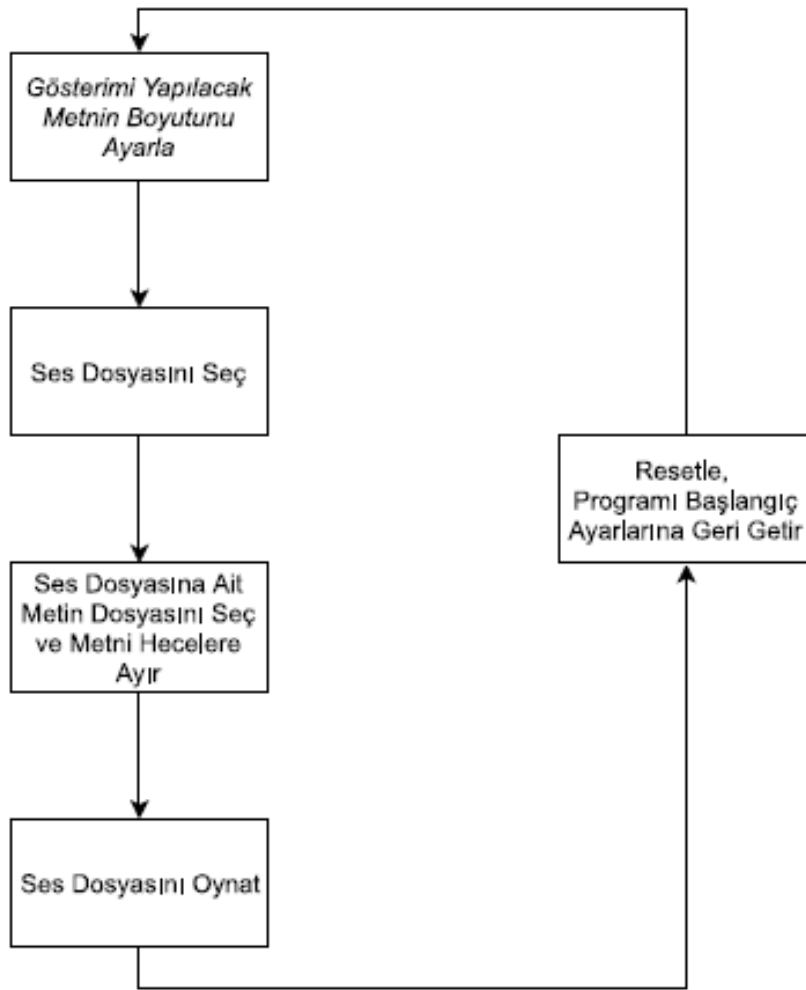
Günlük hayatımızda kullandığımız kelimeler ve bu tezdeki seslendirilmiş metin göz önünde bulundurulduğunda dört heceye kadar ki heceleme performansı yeterli görülmüştür. Çünkü dört harfe kadar olan hece boyutları Türkçe’deki kelimelerin tamamına yakını oluşturduğundan bu durum göz ardı edilmiştir. Şekil 3.21’de heceleme programının akıl şeması gösterilmiştir.

Türkçe’deki genel hece yapıları; tek heceliler V ($V=\text{ünlü}$, $C=\text{ünsüz}$) sadece sesli harften oluşanlar (*a, e, ı, i, o, ö, u, ü*), VC (*el-ma, ül-kü vb*) şeklinde gösterebileceğimiz bir sesli bir sessiz harften oluşanlar, CV sessiz-sesli harften (*ba-lık, ma-sal vb.*) oluşanlar, CVC üç harften oluşanlar (*kül, sal, dal*), yine üç harfli VCC biçiminde (*ırk, ilk*) sesli ile başlayıp sonraki iki harfli sessiz olanlar, CCV ilk iki harfi ünsüz olup son harfi sesli olanlar (*tra, bre*) ve son olarak $CVCC$ şeklinde gösterebileceğimiz dört harfli heceler (*Türk, kurt, yurt*) şeklinde özetlenebilir (*Aşlıyan ve ark., 2006*).



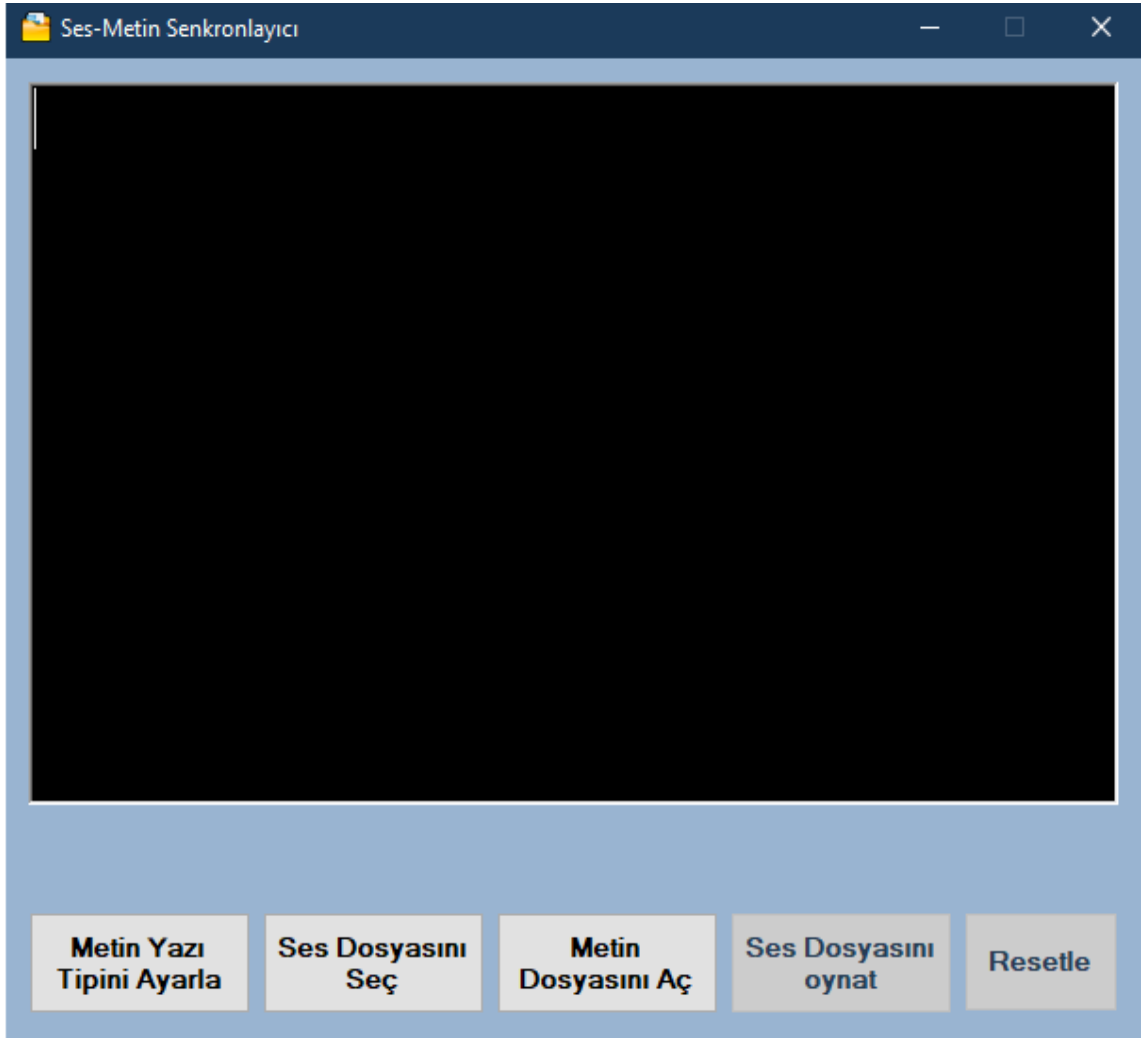
Şekil 3.21 Hecelere ayırma programının akış şeması

Programın çalışma akış diyagramı Şekil 3.22’de gösterilmiştir. Program çalıştırıldığında öncelikle metnin uzunluğuna bağlı olarak net bir şekilde gözlemlenmesi için kullanıcıdan metin boyutunu ayarlanması istenir. Sonraki adımlarda kullanıcıdan sekronlama işleminde kullanılacak ses dosyası seçilmesi işlemi ve akabinde bu ses dosyasına ait, içeriğinde metin ve bu metne ait zamanlama parametreleri olan zengin metin belgesinin seçimi sırayla gerçekleştirilmesi işlemi uygulanır. Son aşamada ize kullanıcı ses dosyasını oynatır ve sesteki değişim ile metindeki değişimi gözlemler.



Şekil 3.22 Arayüz programı çalışma akış diyagramı.

Arayüz programın ekran görüntüsü Şekil 3.23’da gösterildiği gibidir. Burada metinlerin gösterileceği ekran siya arka plana sahiptir. Metinlerin gösterilmesi için beyaz renk ve vurgulama işlemi için de sarı renk tercih edilmiştir.



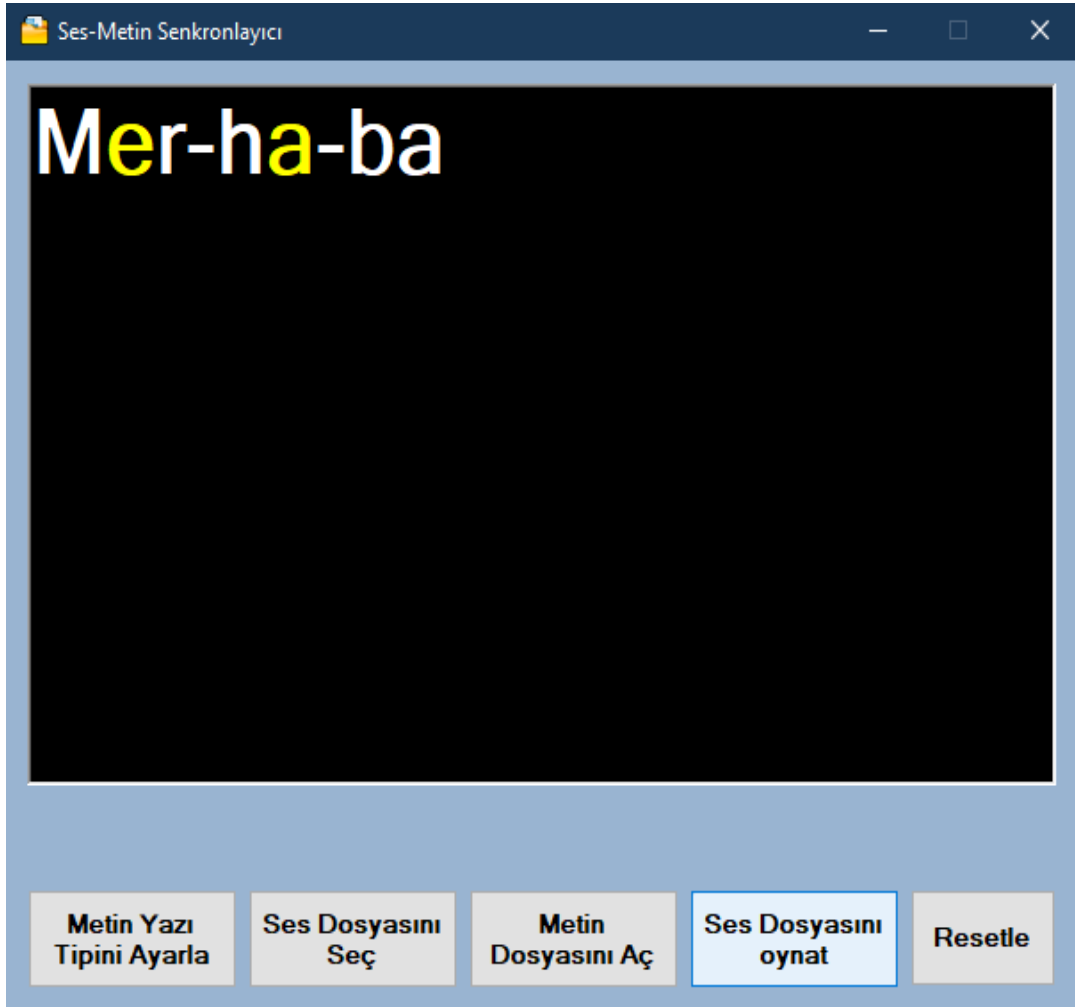
Şekil 3.23 Arayüz programı genel görüntüsü

Program çalıştırıldığında metnin ve hecelere ayırma işleminin gerçekleştirilmesi sonucu metnin aldığı son hali ifade etmesi açısından Şekil 3.24’de örnek bir gösterim sergilenmiştir. Burada her bir hecenin arasında “-” işareti konularak hecelerin bir birinden ayrılması sağlanmıştır.



Şekil 3.24 Arayüz programında hecelenmiş olarak gösterilen “Merhaba” metni.

Sonraki durum olan sesle belirlenen harfin aynı zamanda vurgulaması işlemini gösterilmesi amacıyla Şekil 3.25’ de gösterilen ekran görüntüsü vurgulamanın nasıl gerçekleştirildiği ifade etmektedir.



Şekil 3.25 Saptanan sesli harf zamanlarının ses okunmasıyla ekranda vurgulanması

Yukarıdaki şekilde de görüldüğü üzere, “*merhaba*” kelimesinin içerisindeki üç sesli harfin geçiş zamanları saptanıp ses dosyası okutulması esnasında o zamanların geçişi sırasında metin içerisindeki harfi vurgulanması işlemi gerçekleştirilmiştir. Ekran görüntüsünde ikinci harfin vurgulandığı anda ekran görüntüsü alınmıştır.

4. BULGULAR

Bu kısımda tez kapsamında yapılmış olan analizlerden adım adım bahsedilerek, elde edilmiş sonuçlar değerlendirilmiştir. Öncelikle kullanılacak YSA modellerinin parametrelerinin belirlenmesi için bir dizi testler yapılmıştır. Bu testlerin ilki YSA içinde uygun ara katman ve bu ara katmanlardaki nöron sayılarını belirlemek için yapılmıştır.

Yapılan testler ÖTA ve KÖSA yapıları için ayrı ayrı yapılarak karşılaştırılmıştır. Çalışmalar üç katmana kadar sınırlandırılmış olup belirli aralıklarla nöron sayılarını arttırarak her katman için performans parametreleri kaydedilmiş ve tablolar halinde gösterilmiştir. Performans değerlendirmesinde testler sonucunda doğruluk oranı, hata oranı, hassaslık ve özel etken oranı gibi parametrelere bakılarak kıyaslamaya gidildi. Bu aşamadan sonra elde edilen sonuçlarla birlikte her iki sinir ağının en iyi performans veren yapısı ele alınarak bunlar üzerinde MFC katsayısı değerlendirmesi yapıldı. 10-20 arasında katsayılar ile tekrardan ağ performansları kıyaslandı. Bunun sonucunda en iyi MFC katsayısı her iki ağ yapısı için belirlendi. Sonraki adımda en iyi ağ belirlendikten sonra bu ağ çerçeve uzunlukları bakımından bir teste tabi tutuldu ve en iyi çerçeve uzunluğu performansına göre belirlendi. Son adımda ise en iyi performansı veren ağ yapısına on adet kelime sokularak bu kelimelerin içerisindeki sesli harfleri ayırt etmesi işlemi uygulandı.

Tüm bu testlerde ÖTA yapısı için eğitim fonksiyonu “*trainscg*” ve KÖSA için ise “*trainlm*” seçilmiştir. Aktivasyon fonksiyonunda ise her iki yapıda “*tansig*” kullanılmıştır. ÖTA yapısında çıkış katmanı fonksiyonu “*softmax*”, KÖSA’da “*purelin*” seçilmiştir. Bu seçim denenerek belirlenmiştir. İlk aşama olan YSA için ara katman ve nöron sayısını belirleme testinde 17 MFC katsayısı kullanılmıştır. En iyi ağ belirlendikten sonra MFC katsayıları arasında da kıyaslama yapılmıştır. Bu testlerde eğitilen farklı YSA yapılarını test etmek için kullanılan 290 adet (*bütün harflerden 10 tane olmak üzere*) sesli-sessiz harf düzeneğinin hedef matrisinde ikili sınıflandırma modeli kullanılmıştır. Tüm gerçekleştirilen testlerdeki çalışmalar tek seferde ağ eğitilerek sonuçları tablolar halinde gösterilmiştir.

Çizelge 4.1 Tek katman için ÖTA nöron sayısı değişimi ve performans parametreleri

ÖRÜNTÜ TANIMA AĞI (<i>trainscg</i>)	NÖRON SAYISI	EĞİTİM PERFORMANSI (<i>Cross-Entropy</i>)	DOĞRULUK ORANI (<i>Correct Rate</i>)	HATA ORANI (<i>Error Rate</i>)	HASSASLIK (<i>Sensitivity</i>)	ÖZEL ETKEN ORANI (<i>Specificity</i>)
TEK KATMAN	10	9.3562e-07	0.9474	0.0526	0.9263	0.9707
	20	2.902e-07	0.9651	0.0349	0.9570	0.9733
	30	2.52e-07	0.9651	0.0349	0.9506	0.9804
	40	2.7563e-07	0.9709	0.0291	0.9590	0.9834
	50	2.7942e-07	0.9704	0.0296	0.9625	0.9784
	60	1.9065e-07	0.9692	0.0308	0.9624	0.9762
	70	3.04e-07	0.9645	0.0355	0.9481	0.9821

Çizelge 4.1’de tek katmanlı ÖTA’nın onarlı olarak nöron sayılarını arttırarak eğitilmiş, eğitim sonrası ikili sınıflandırma performans parametreleri gösterilmiştir. Çizelgeden de anlaşılacağı üzere ortalama doğruluk oranı %97-%94 arası değişmektedir.

Daha sonraki adımda iki katman kullanılarak eğitimler yapılmış. Bu adımda da kullanılan nöron sayıları her iki katmanda da arttırılarak değiştirip sonuçları Çizelge 4.2’de verilmiştir. Görüleceği üzere performans olarak bir birine yakın değerler dikkate çarpmaktadır.

Çizelge 4.2 İki katman için ÖTA nöron sayısı değişimi ve performans parametreleri

ÖRÜNTÜ TANIMA AĞI (<i>trainscg</i>)	NÖRON SAYISI (1. Katman, 2. Katman)	EĞİTİM PERFORMANSI (<i>Cross- Entropy</i>)	DOĞRULUK ORANI (<i>Correct Rate</i>)	HATA ORANI (<i>Error Rate</i>)	HASSASLIK (<i>Sensitivity</i>)	ÖZEL ETKEN ORANI (<i>Specificity</i>)
İKİ KATMAN	20,5	5.9388e-08	0.9631	0.0369	0.9450	0.9826
	20,10	8.6892e-08	0.9679	0.0321	0.9543	0.9822
	20,20	1.5134e-07	0.9665	0.0335	0.9537	0.9799
	30,10	8.5897e-08	0.9648	0.0352	0.9545	0.9755
	30,20	9.8748e-08	0.9723	0.0277	0.9596	0.9857
	40,10	1.0487e-07	0.9679	0.0321	0.9654	0.9703
	40,20	1.0262e-07	0.9651	0.0349	0.9501	0.9810
	40,30	7.774e-08	0.9690	0.0310	0.9619	0.9762
	50,10	1.4472e-07	0.9799	0.0201	0.9787	0.9810
	50,20	1.8098e-07	0.9726	0.0274	0.9637	0.9818
	50,30	6.0268e-08	0.9651	0.0349	0.9511	0.9799
	50,40	9.5682e-08	0.9723	0.0277	0.9704	0.9742
	50,50	4.6491e-08	0.9712	0.0288	0.9536	0.9901

Çizelge 4.3 Üç katman için ÖTA nöron sayısı değişimi ve performans parametreleri

ÖRÜNTÜ TANIMA AĞI (<i>trainscg</i>)	NÖRON SAYISI (1. Katman, 2. Katman, 3. Katman)	EĞİTİM PERFORMANSI (Cross-Entropy)	DOĞRULUK ORANI (Correct Rate)	HATA ORANI (Error Rate)	HASSASLIK (Sensitivity)	ÖZEL ETKEN ORANI (Specificity)
ÜÇ KATMAN	20,10,10	2.4129e-08	0.9628	0.0372	0.9479	0.9787
	20,20,10	4.4833e-08	0.9704	0.0296	0.9540	0.9879
	30,10,10	2.9942e-08	0.9681	0.0319	0.9553	0.9816
	30,20,10	5.7483e-08	0.9676	0.0324	0.9557	0.9800
	30,20,20	3.6177e-08	0.9718	0.0282	0.9616	0.9823
	40,10,10	1.6496e-07	0.9589	0.0411	0.9515	0.9666
	40,20,10	4.0654e-08	0.9804	0.0196	0.9730	0.9881
	40,20,20	2.8066e-08	0.9737	0.0263	0.9710	0.9764
	40,30,20	3.4884e-08	0.9740	0.0260	0.9628	0.9857
	40,30,30	2.0738e-08	0.9712	0.0288	0.9595	0.9834
	50,10,10	3.0302e-08	0.9729	0.0271	0.9627	0.9835
	50,20,10	4.2468e-08	0.9754	0.0246	0.9695	0.9814
	50,30,10	2.2659e-07	0.9653	0.0347	0.9591	0.9717
	50,30,20	5.6888e-08	0.9785	0.0215	0.9646	0.9931
	50,30,30	4.5603e-08	0.9631	0.0369	0.9479	0.9792
	50,40,10	3.5215e-08	0.9734	0.0266	0.9653	0.9818
	50,40,20	3.7714e-08	0.9692	0.0308	0.9569	0.9822
	50,40,30	4.0281e-08	0.9679	0.0321	0.9644	0.9713
	50,40,40	3.8336e-08	0.9595	0.0405	0.9490	0.9703
	50,50,10	6.0878e-08	0.9720	0.0280	0.9637	0.9807
	50,50,40	3.0435e-08	0.9687	0.0313	0.9529	0.9856

Üç katmanlı YSA modelinde yapılmış çalışmalar ÖTA yapısı için en iyi sonuçların elde edildiği çalışma olmuştur. Ortalama doğruluk oranı %98'e ulaşmıştır (40-20-10 yapısı için). Çizelge 4.3'de görüleceği üzere, önceki farklı katman-nöron sayılarında oluşturulmuş 41 YSA yapısı içerisinde [40-20-10] nöron sayısına sahip üç katmanlı YSA modeli test sonuçları tek, iki ve üç katmanlı ağ yapıları içerisinde en iyi sonucu vermiştir.

Benzer çalışma KÖSA yapısı için de uygulanmış bir katmandan başlayarak üç katmana kadar belirli nöron sayıları aralığında testler yapılmıştır. Burada KÖSA'nın dinamik yapısından ve eğitim fonksiyonundan kaynaklı bir takım sıkıntılar yaşandı. ÖTA yapısına göre bu ağda eğitim süreleri uzamış ve bilgisayarda bir takım kasmalara sebep olmuştur. Bu sebeple nöron sayılarının sayısı fazla artırılamamıştır. Bunun sebebi ise kullanılan bilgisayar hafızası ve işlemci performansı yetersiz olmasıdır.

Tek katmanlı KÖSA yapısında seçilmiş olan nöron sayıları ve eğitim sonrası yapılmış testlerin sonuçları Çizelge 4.4’de verilmiştir. Test sonucunda %93-96 arası bir doğruluk oranı vardır.

Çizelge 4.4 Tek katman için KÖSA nöron sayısı değişimi ve performans parametreleri

Katman Özyinelemeli Sinir Ağı (trainlm)	NÖRON SAYISI	EĞİTİM PERFORMANSI (Mean-Square Error)	DOĞRULUK ORANI (Correct Rate)	HATA ORANI (Error Rate)	HASSASLIK (Sensitivity)	ÖZEL ETKEN ORANI (Specificity)
TEK KATMAN	10	0.01484	0.9502	0.0498	0.9491	0.9514
	15	0.0091246	0.9388	0.0612	0.9173	0.9624
	20	0.0095811	0.9592	0.0408	0.9560	0.9624
	25	0.0071028	0.9670	0.0330	0.9623	0.9718
	30	0.0065754	0.9435	0.0565	0.9424	0.9447
	35	0.0053399	0.9575	0.0425	0.9538	0.9612

İki katmanda oluşturulan YSA modelleri ve bunlarla yapılan testler Çizelge 4.5’de gösterilmiştir. Buradaki durum tek katmanlı yapıya benzerlik göstermektedir. Sistemlerin doğruluk oranı %94-96 arasında değişmektedir.

Çizelge 4.5 İki katman için KÖSA nöron sayısı değişimi ve performans parametreleri

Katman Özyinelemeli Sinir Ağı (trainlm)	NÖRON SAYISI (1. Katman, 2. Katman)	EĞİTİM PERFORMANSI (Mean-Square Error)	DOĞRULUK ORANI (Correct Rate)	HATA ORANI (Error Rate)	HASSASLIK (Sensitivity)	ÖZEL ETKEN ORANI (Specificity)
İKİ KATMAN	10,5	3.22498e-5	0.9488	0.0512	0.9311	0.9680
	10,10	5.5396e-10	0.9628	0.0372	0.9528	0.9732
	15,5	3.6656e-11	0.9667	0.0333	0.9674	0.9660
	15,10	1.1226e-10	0.9603	0.0397	0.9419	0.9802
	20,5	1.2996e-13	0.9673	0.0327	0.9537	0.9816
	20,10	1.8323e-13	0.9595	0.0405	0.9505	0.9687
	20,15	1.1336e-11	0.9542	0.0458	0.9374	0.9722
	30,5	1.521e-11	0.9670	0.0330	0.9562	0.9783
	30,10	2.8939e-13	0.9637	0.0363	0.9559	0.9716
	30,15	8.7641e-14	0.9597	0.0403	0.9447	0.9758
	30,20	2.486e-10	0.9550	0.0450	0.9375	0.9739

Üç katmanlı KÖSA ile farklı sayıdaki nöronlarla oluşturulan ağ yapıları diğer iki ve tek katmalı yapıya göre az da olsa daha iyi bir performans göstermektedir. Çizelge 4.6’da da

görüreceği üzere KÖSA yapısı içerisindeki en iyi parametrik değerleri [20,15,10] yapısı vermiştir. Sistemin doğruluk oranı % 96.9 olarak ölçülmüştür.

Çizelge 4.6 Üç katman için KÖSA nöron sayısı değişimi ve performans parametreleri

Katman Özyinelemeli Sinir Ağı (trainlm)	NÖRON SAYISI (1. Katman, 2. Katman, 3. Katman)	EĞİTİM PERFORMANSI (Mean-Square Error)	DOĞRULUK ORANI (Correct Rate)	HATA ORANI (Error Rate)	HASSASLIK (Sensitivity)	ÖZEL ETKEN ORANI (Specificity)
ÜÇ KATMAN	10,5,5	1.6007e-14	0.9553	0.0447	0.9437	0.9674
	10,10,5	3.1416e-11	0.9586	0.0414	0.9446	0.9735
	10,10,10	2.9122e-13	0.9382	0.0618	0.9068	0.9746
	20,5,5	6.2002e-15	0.9623	0.0377	0.9543	0.9705
	20,10,5	9.2299e-12	0.9592	0.0408	0.9394	0.9807
	20,10,10	8.444e-12	0.9611	0.0389	0.9587	0.9636
	20,15,10	3.848e-12	0.9690	0.0310	0.9634	0.9746
	20,15,15	2.9957e-15	0.9572	0.0428	0.9430	0.9724
	20,20,10	6.0341e-13	0.9611	0.0389	0.9603	0.9620
	20,20,15	5.177e-14	0.9634	0.0366	0.9524	0.9749
	30,10,5	2.2712e-13	0.9670	0.0330	0.9522	0.9827
	30,10,10	4.6839e-12	0.9561	0.0439	0.9334	0.9812
	30,20,5	1.9102e-10	0.9533	0.0467	0.9344	0.9738
	30,20,10	1.8471e-11	0.9642	0.0358	0.9575	0.9711
	30,20,20	1.1853e-13	0.9665	0.0335	0.9537	0.9799
	30,30,10	1.0845e-10	0.9665	0.0335	0.9607	0.9723
	30,30,20	8.4561e-15	0.9544	0.0456	0.9355	0.9749

YSA yapılarının ara katman ve bu katmanlara ayit nöron sayılarını belirleme işlemi sonrasında en uygun MFC katsayılarını belirlemek 10-20 katsayı aralıklarında testler yapılmıştır. Buradaki testler her iki ağ için en iyi performansı gösteren yapılar seçilerek uygulanmış bulunmaktadır. Literatürdeki çalışmalarda MFC katsayıları 12-16 arasında değişmektedir (Han ve ark., 2006 ; Can, 2007 ; Dede, 2008 ; Joshi ve Cheeran, 2014). Çizelge 4.7’de ÖTA yapısı için farklı MFC katsayılarına karşılık gelen ağ performans değerleri gösterilmiştir. Buradan da görüleceği üzere en yüksek performansı 17 MFC katsayısı vermiştir.

Çizelge 4.7 Seçilen ÖTA için 10-20 arası MFC katsayılarının performans parametreleri.

ÖRÜNTÜ TANIMA AĞI (<i>trainscg</i>)	MFC KATSAYILARI	EĞİTİM PERFORMANSI (<i>Cross-Entropy</i>)	DOĞRULUK ORANI (<i>Correct Rate</i>)	HATA ORANI (<i>Error Rate</i>)	HASSASLIK (<i>Sensitivity</i>)	ÖZEL ETKEN ORANI (<i>Specificity</i>)
ÜÇ KATMAN (40-20-10)	10	3.2201e-08	0.9536	0.0464	0.9560	0.9512
	11	3.3926e-08	0.9600	0.0400	0.9536	0.9666
	12	4.2466e-08	0.9648	0.0352	0.9595	0.9701
	13	4.0485e-08	0.9637	0.0363	0.9662	0.9612
	14	7.1674e-08	0.9555	0.0445	0.9537	0.9574
	15	3.3127e-08	0.9614	0.0386	0.9650	0.9579
	16	3.0604e-08	0.9651	0.0349	0.9550	0.9755
	17	4.0654e-08	0.9804	0.0196	0.9730	0.9881
	18	3.8866e-08	0.9740	0.0260	0.9633	0.9852
	19	4.5827e-08	0.9732	0.0268	0.9602	0.9868
	20	4.1406e-06	0.9732	0.0268	0.9617	0.9851

KÖSA yapısı için tekrarlanan testlerde en iyi sonuç 19 MFC katsayısı ile elde edilmiştir. Çizelge 4.8’ de gösterildiği üzere ağın 16-19 katsayılarıdaki performansları bir birine yakın değerdedir. Buradan da anlaşılacağı üzere ÖTA yapısı daha iyi bir performans sergilemesi sebebiyle sonraki adımlarda yapılacak olan testlerde bu YSA yapısı ile çalışmalar sürdürülmüştür.

Çizelge 4.8 Seçilen KÖSA için 10-20 arası MFC katsayılarının performans parametreleri.

Katman Özyinelemeli Sinir Ağı (<i>trainlm</i>)	MFC KATSAYILARI	EĞİTİM PERFORMANSI (<i>Mean-Square Error</i>)	DOĞRULUK ORANI (<i>Correct Rate</i>)	HATA ORANI (<i>Error Rate</i>)	HASSASLIK (<i>Sensitivity</i>)	ÖZEL ETKEN ORANI (<i>Specificity</i>)
ÜÇ KATMAN (25-15-10)	10	2.8799e-12	0.9553	0.0447	0.9624	0.9484
	11	2.9343e-13	0.9561	0.0439	0.9467	0.9658
	12	8.1739e-13	0.9645	0.0355	0.9668	0.9623
	13	5.0523e-14	0.9463	0.0537	0.9554	0.9377
	14	9.9367e-14	0.9486	0.0514	0.9592	0.9384
	15	1.5955e-14	0.9555	0.0445	0.9532	0.9580
	16	1.1181e-14	0.9637	0.0363	0.9539	0.9738
	17	3.848e-12	0.9690	0.0310	0.9634	0.9746
	18	1.812e-09	0.9679	0.0321	0.9603	0.9756
	19	1.0547e-13	0.9743	0.0257	0.9684	0.9802
	20	1.471e-13	0.9595	0.0405	0.9461	0.9736

ÖTA (17 MFC katsayısı ve [40-20-10] ağ yapısıyla) performans olarak daha iyi sonuç verdiği saptandıktan sonra bu ağın çerçeve boyutlarının bir biri ile kıyaslanması amacıyla ayrı bir teste tabi tutulmuştur. Bu test ile ağın farklı çerçeve uzunlukları ile nasıl bir performans sergileyeceğini gözlemlemek amacıyla yapılmıştır.

Çizelge 4.9 Farklı çerçeve uzunluklarına göre ÖTA yapısının test sonuçları

ÖRÜNTÜ TANIMA AĞI (<i>trainscg</i>)	Çerçeve Uzunluğu	Kaydırma Oranı	MFC Katsayısı	EĞİTİM PERFORMANSI (<i>Cross-Entropy</i>)	DOĞRULUK ORANI (<i>Correct Rate</i>)	HATA ORANI (<i>Error Rate</i>)	HASSASLIK (<i>Sensitivity</i>)	ÖZEL ETKEN ORANI (<i>Specificity</i>)
Üç Katmanlı [40-20-10]	160	%75	17	2.7523e-08	0.9544	0.0456	0.9442	0.9635
	200	%75	17	4.7234e-08	0.9576	0.0424	0.9407	0.9735
	240	%75	17	2.9347e-08	0.9663	0.0337	0.9533	0.9786
	280	%75	17	3.682e-08	0.9633	0.0367	0.9476	0.9787
	320	%75	17	3.7392e-08	0.9619	0.0381	0.9507	0.9728
	360	%75	17	2.6003e-08	0.9655	0.0345	0.9572	0.9737
	400	%75	17	3.1556e-08	0.9743	0.0257	0.9660	0.9826
	440	%75	17	4.0654e-08	0.9804	0.0196	0.9730	0.9881

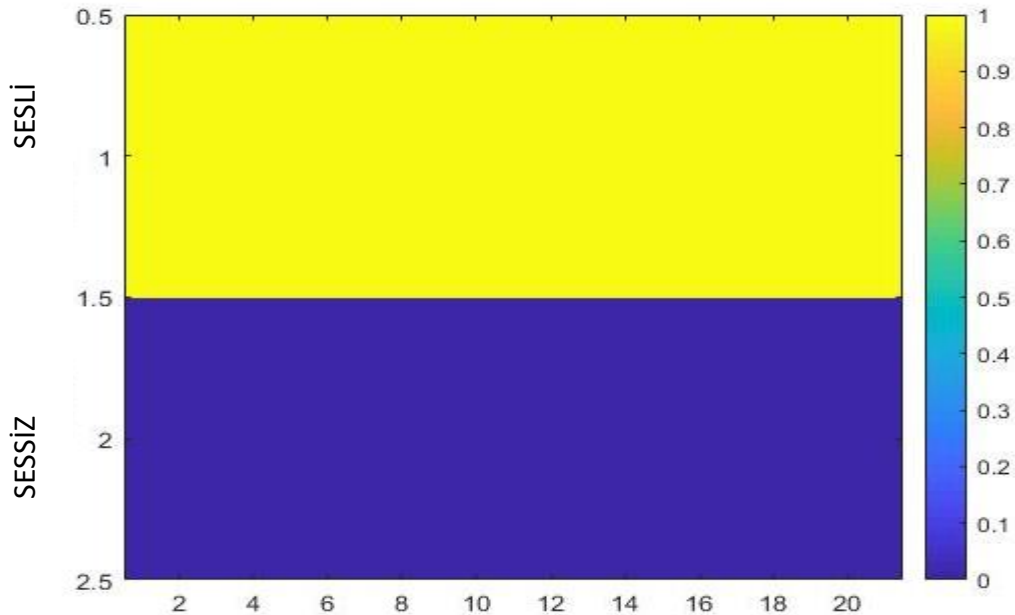
En iyi ağ modeli ve yapısı belirlendikten, bu ağın birde eğitimi ve testlerini farklı çerçeve boyutları ile değerlendirmeye tabi tutuldu. Sonuçlar başlangıçtaki öngörülerimizi destekler nitelikte çıkmıştır. Çerçeve boyutunu belirlerken başlangıçta kaydedilen sesli ve sesiz harflerin boyutlarını göz önünde bulunduruldu. Kayıtlarda bazı sessiz harflerin boyutlarının 450-600 örnekler arasında değiştiği saptandıktan sonra 440'lık çerçevelerin ideal olacağı düşünüldü, buradaki yaklaşım sessizlerin anlamlı çerçeve uzunluklarını mümkün olduğunca bozmamaktır. Yaklaşımın doğruluğu yapılmış olan son testten de anlaşılabilmektedir. Çizelge 4.9'da görüleceği üzere performansı en yüksek veren çerçeve uzunluğu 440'da elde edilmiştir. Çerçeveler 40'ar olarak arttırılmış ve kaydırma oranı ve diğer parametreler sabit tutularak test edilmiştir. 440'dan sonra artırım uygulanmamasının sebebi çerçeve uzunluğunun bazı sessiz harflerin örnek uzunluğunu geçmesidir.

Sonuç Matrisi	Sesli	Sessiz	Oran	Toplam
---------------	-------	--------	------	--------

Sesli	1763	21	%98.82	1784
Sessiz	49	1747	%97.27	1796
Oran	%97.30 (Hassaslık)	%98.81 (Özel etken oranı)	%98.04	-
Toplam	1812	1768	-	3580

Şekil 4.1 Sesli-Sessiz testi sonuçlarının istatistiksel değerlendirmesi

En iyi performansı veren YSA modeli ile yapılmış olan sesli-sessiz harf testinin sonuç matrisi Çizelge 4.10’de verilmiştir. Burada 290 adet ses 440’lık ve %75 kaydırılarak 3580 adet çerçeveye bölünüp sesli-sessiz olarak tek bir matriste toplanıp ağa sokulmuştur. Sonuç olarak %98.02’lik bir ortalama doğruluk oranı yakalanmış, bu oran sesli harflerde %98.82, sessiz harflerde %97.27 olarak hesaplanmıştır. Bu çalışmadan sonra sesli harfleri tanıma işlemini görsel olarak ifade etmek için test sonuç matrisinin grafiksel gösterimi Şekil 4.2 (“a” harfinin test sonucu) verilmiştir.

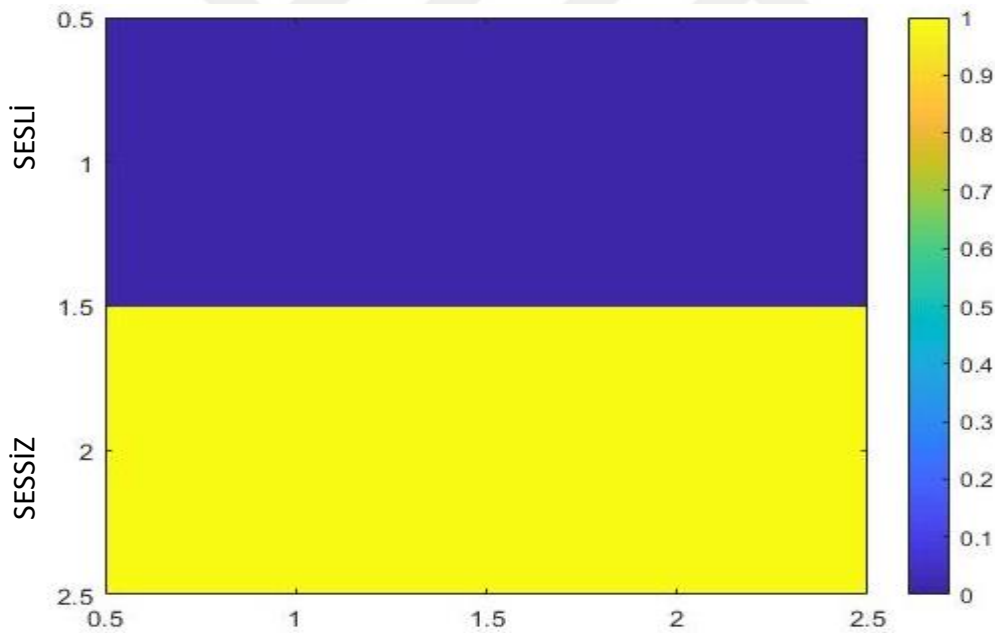


Şekil 4.2 Ağa gönderilen “a” harfinin sonuç matrisinin görsel çıktısı

Çizelge 4.10 Ağa gönderilmiş “a” sesine ait sonuç matrisi (ilk 7 çerçeve)

Sesli	1	1	1	1	1	1
Sessiz	1.08E-09	2.22E-10	2.58E-10	5.72E-10	4.69E-10	1.21E-09

Buradaki ve bundan sonraki test sonuçlarının grafiklerinde; yatay eksen çerçeve sayısını, dikeyde ise sınıfları temsil etmektedir. Örneğin ikili sınıflandırmada dikeyde iki, dokuzlu sınıflandırmada ise dikeyde 9 sınıf tablolarında ve grafiklerde gösterilmiştir. Benzer şekilde yatayda ise çerçeve sayıları belirli bir aralıkta vurgulanmaktadır. Çerçevelerin çokluğu ve azlığı sestense değişim göstermektedir. Grafiklerdeki renkler(*grafik sağında ölçeklendirilmiş değerler gösterilmiştir*); sarı “1” (%100) mavi ise “0” (%0) saptama oranını temsil etmektedir. Buna benzer bir diğer çalışmada ağa sessiz harf gönderilerek sonuçları gözlemlenmiştir. Örnek olarak “t” harfinin ağ çıktı görseli Şekil 4.3’de verilmiştir.



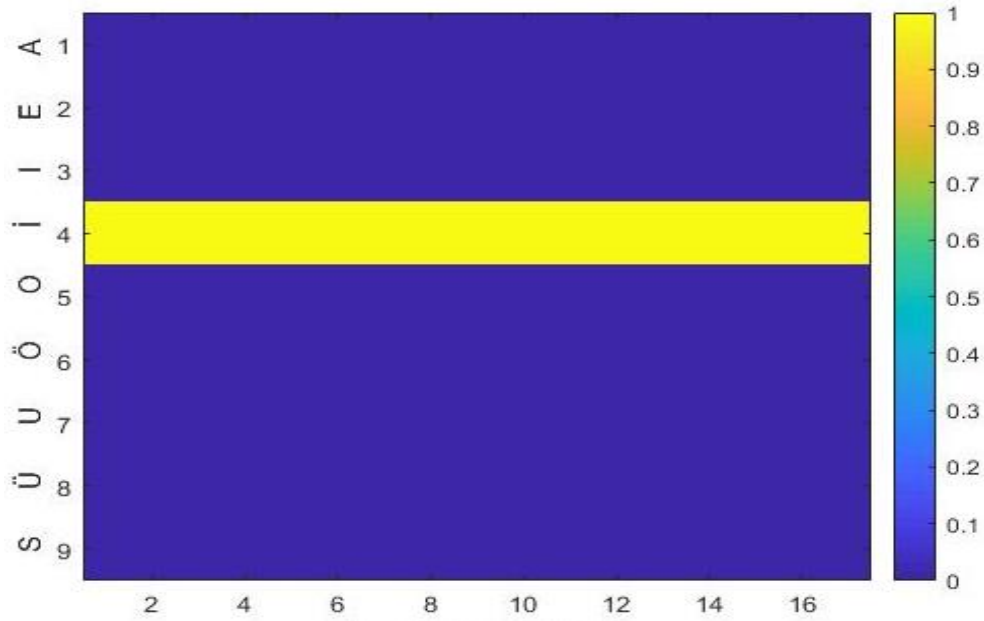
Şekil 4.3 Ağa gönderilen “t” harfinin sonuç matrisinin görsel çıktısı

Buradaki çıkış matrisinin uzunluğu ağa gönderilmiş olan harfin çerçeve sayısına göre değişmektedir. Çizelge 4.11’den anlaşılacağı üzere “t” harfinin iki çerçeveden oluştuğu gözlemlenmektedir.

Çizelge 4.11 Ağa gönderilmiş “t” sesine ait sonuç matrisi

Sesli	2.94E-11	4.96E-12
Sessiz	1	1

İkili sınıflandırma bu şekilde gözlemlendikten sonra ağı sesli harfleri kendi içerisinde sınıflandırmasını net bir şekilde görmek için ayrıca bir test işlemi uygulanmıştır. Bu teste 9’lu sınıflandırma yöntemi kullanılarak ağa gönderilen farklı sesli harflerin tanınması görselleştirilmiştir.



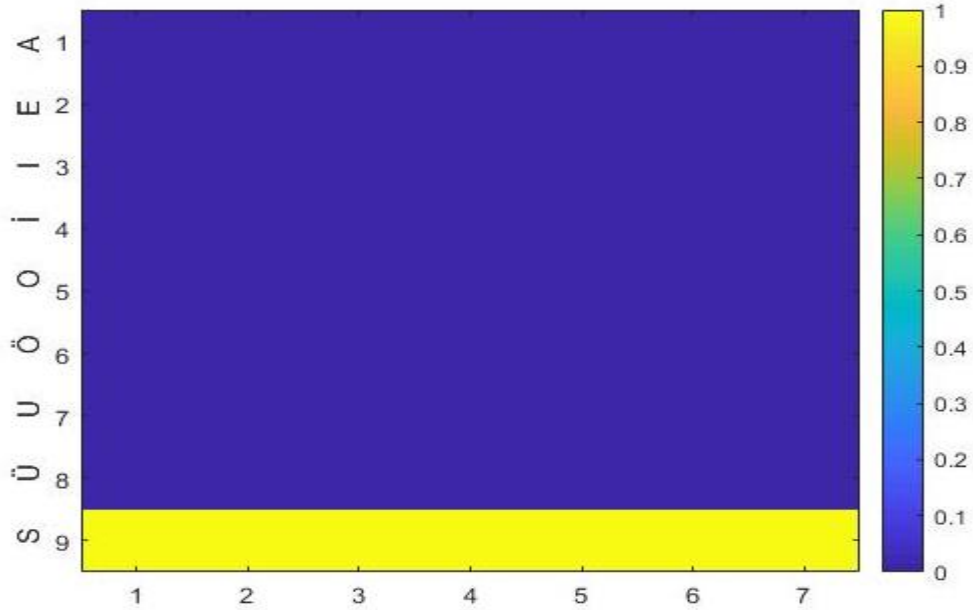
Şekil 4.4 Dokuzlu sınıf içi ağa gönderilmiş “i” harfinin çıktısı

Çizelge 4.12’de ağa gönderilmiş olan “i” harfinin çıkış matrisi verilmiştir. Şekil 4.4’de ise bu matrisin görselini ifade etmektedir. Şekilde yataydaki 2-16 arası rakamlar çerçeve sayısını ifade etmekte, dikeydekiler ise sınıfları yani harfleri göstermektedir. “S” ile gösterilen satır sessiz harflere ait sınıfı ifade etmektedir. Burada “i” harfinin saptanılan çerçevelerdeki kesinlik göze çarpmaktadır. 0-1 arası yapılan sınıflandırmada doğrudan 1 olarak ağ harfi tanımıştır.

Çizelge 4.12 Ağı “i” harfine ait çıkış matrisi (ilk 7 çerçeve)

A	2.84E-10	3.09E-10	4.13E-10	9.97E-10	1.85E-09	1.18E-09
E	6.32E-07	3.26E-07	2.48E-07	1.29E-07	1.75E-07	1.93E-07
I	3.14E-07	5.82E-07	8.69E-07	2.37E-06	2.49E-06	1.84E-06
İ	1	1	1	1	1	1
O	2.73E-06	4.86E-06	6.38E-06	1.23E-05	1.01E-05	8.89E-06
Ö	1.73E-05	1.45E-05	1.21E-05	7.44E-06	6.16E-06	7.48E-06
U	1.38E-09	1.15E-09	1.20E-09	1.86E-09	3.17E-09	2.25E-09
Ü	5.05E-08	3.00E-08	2.70E-08	2.22E-08	3.65E-08	3.32E-08
Sessiz	4.09E-10	6.63E-10	8.02E-10	7.62E-10	4.40E-10	5.77E-10

Tablolardaki bir ile ifade edilen kısımları %100 olarak tanımanın gerçekleştiği olarak yorumlarsak diğer kısımlar örneğin (Tablo 4.12’de 2.sütun, 3. satır) 5.82e-07 ifadesi 0.00000058198 ifadesinin yaklaşık değerine eşittir. Bu da sıfıra oldukça yakın bir değer olarak ifade edilebilir.



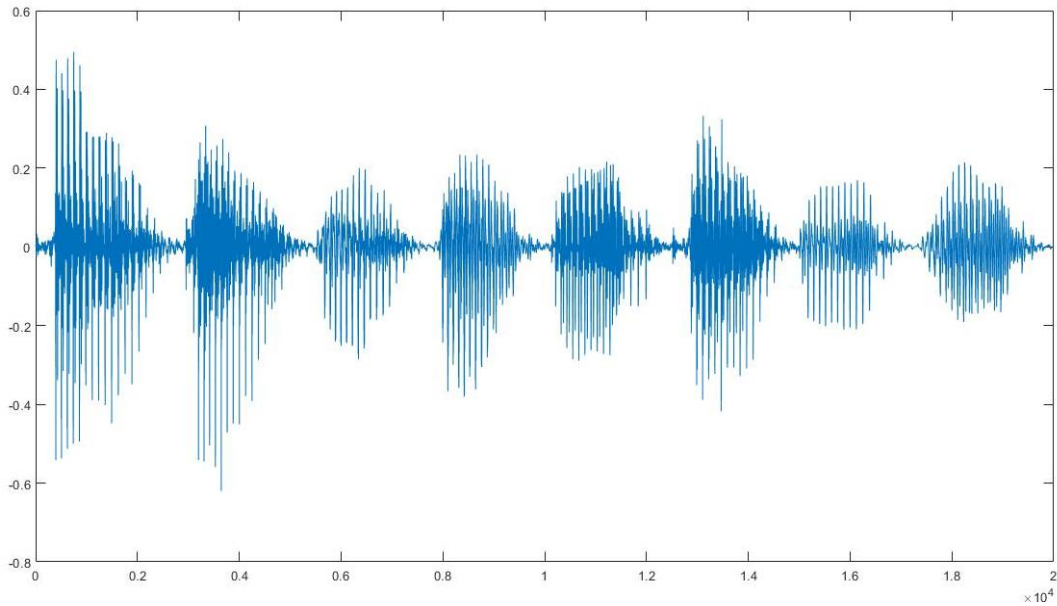
Şekil 4.5 Ağa gönderilmiş “y” harfinin çıktısı

Sonraki adımda sessiz harflerden örnek verilerek sonuçlar elde edilmiştir. Bunlardan bir tanesi “y” harfini ağa gönderilerek yapılmıştır. Sonuç matrisi Çizelge 4.13’de verilmiş ve bu matrise ait görsel Şekil 4.5’de gösterilmiştir. Buradan da anlaşılacağı üzere ağ bu harfi sessiz olarak sınıflandırılmıştır.

Çizelge 4.13 Ağın “y” harfine ait çıkış matrisi

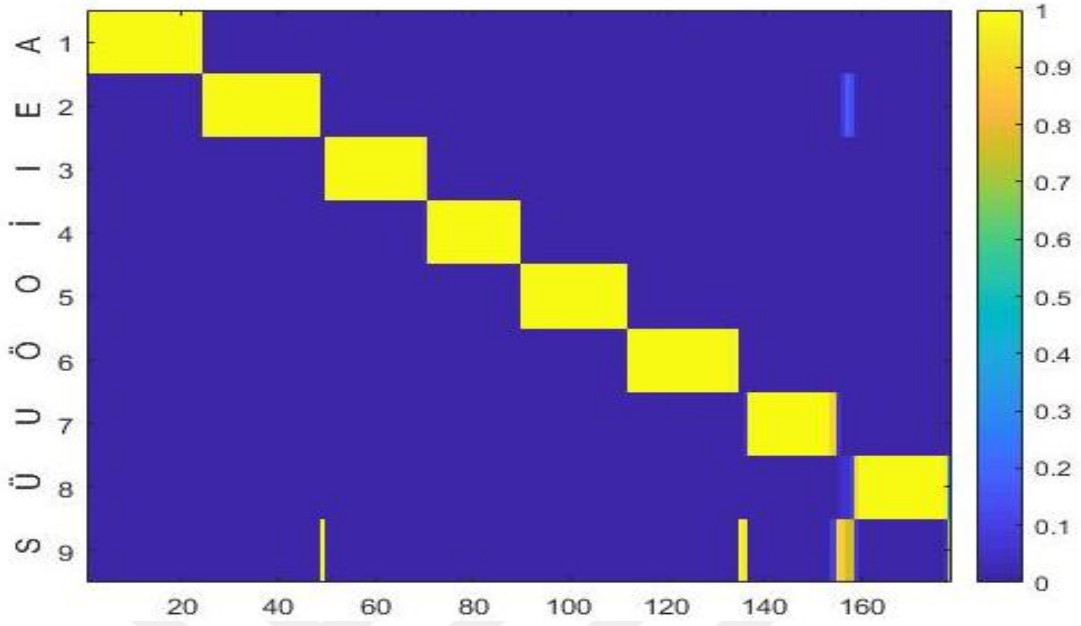
A	1.31E-10	1.38E-11	6.86E-12	1.15E-11	7.51E-12	8.43E-11
E	1.46E-09	2.49E-09	3.60E-09	2.31E-09	1.41E-08	1.57E-07
I	1.04E-08	1.56E-10	5.07E-11	9.35E-11	5.39E-11	4.48E-10
İ	8.31E-12	4.79E-13	2.88E-13	2.69E-13	1.53E-12	1.73E-10
O	1.14E-09	4.12E-10	2.98E-10	4.51E-10	3.96E-10	3.33E-09
Ö	4.17E-08	1.25E-08	1.23E-08	9.03E-09	3.78E-08	1.50E-07
U	1.73E-09	7.08E-10	5.64E-10	7.45E-10	6.16E-10	1.15E-09
Ü	7.69E-09	6.75E-10	4.47E-10	3.75E-10	1.24E-09	1.20E-08
Sessiz	1	1	1	1	1	1

Yapılmış olan diğer bir teste, ağa tek tek sesli harfleri vermek yerine sesli harflerin tümünün sırayla(a, e, ı, i, o, ö, u, ü) seslendirilmesi ile oluşan bir ses dosyasının verilmesiyle gerçekleştirilmiştir. Ses dosyasının grafiği Şekil 4.6'da verilmiştir. Burada yatay eksen sesin süresini(sn.), dikey ise genliğini ifade etmektedir.



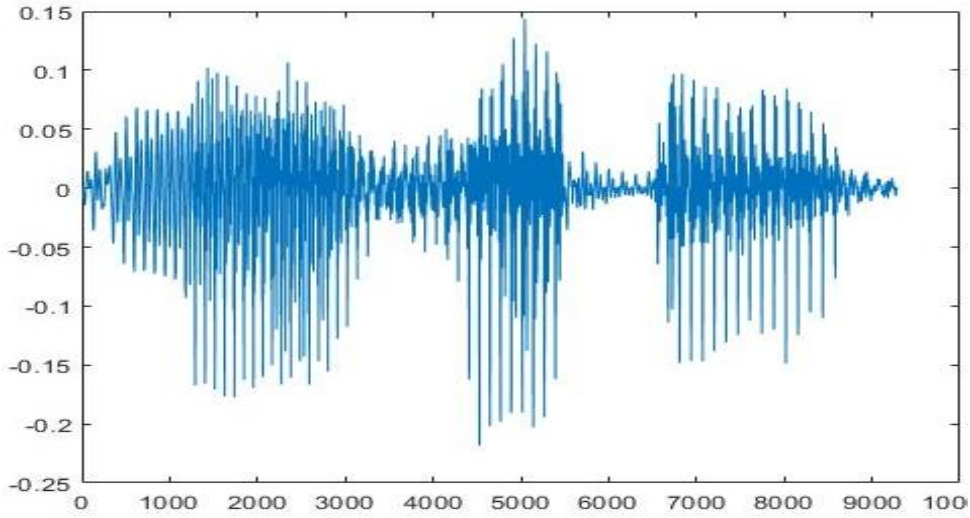
Şekil 4.6 Bütün sesli harflerin sıralı olduğu ses dosyası grafiği

Sesli harflerin bir arada bulunduğu ses dosyasının eğitilmiş olan ağa gönderilmesiyle oluşan çıkış matrisinin görsel hali Şekil 4.7'de gösterilmiştir. Burada sonuçların oldukça net olduğu görülmektedir. Görselden de anlaşılacağı üzere bazı harflerin başlangıç ve bitiş noktalarındaki bir takım çerçeveler sessiz olarak tanınmıştır. Bunun sebebi fonem tabanlı ses tanıma sistemlerdeki veri karakteristiğinin küçük boyutta olmasına bağlanmıştır.

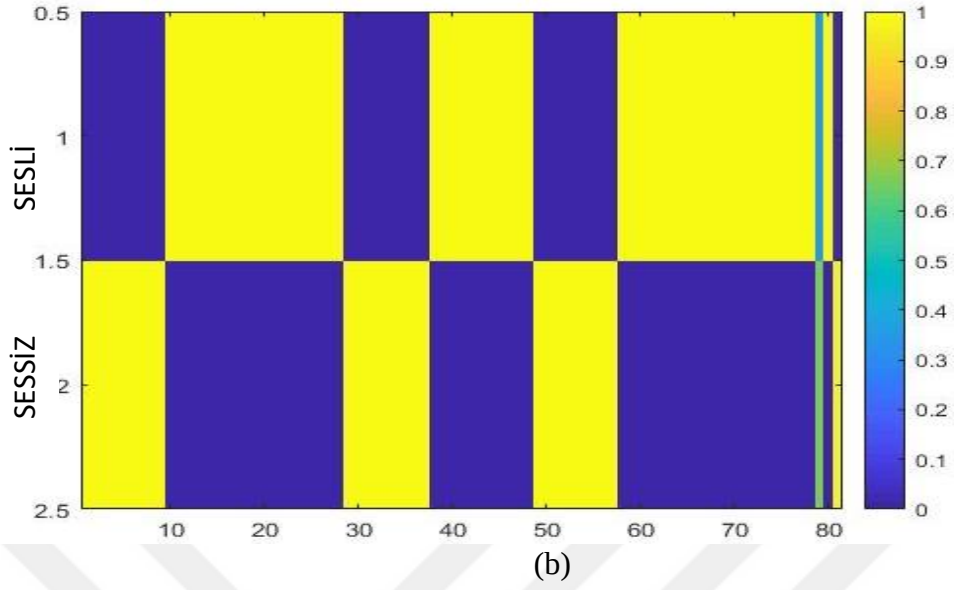


Şekil 4.7 Bütün sesli harflerin bulunduğu dosyanın ağ çıktısı

Son aşamada YSA'ya kelime vektörlerinden oluşan 10 adet ses kaydı sırayla ağı sokulup, kelime içerisindeki sesli harfleri tespit edilmesi istendi. Tespit edilen sesli harflerin geçiş zamanları hesaplanarak metin dosyasına kayıt edildi. Şekil 4.8(a) ve 4.9(b)'da "merhaba" ve "dünya" kelimelerinin ağ çıktılarının görselleri verilmiştir. Grafikten de anlaşılağı üzere metin içerisindeki sesli harfler net bir şekilde tespit edilmiştir. Sarı ile gösterilen bölgeler sesli ve sessiz harflerden birinin tanındığı yani matris çıktısının 1'e çok yakın olduğunu belirtiyor. Grafiğin sağındaki skaladan da bu anlaşılmaktadır.

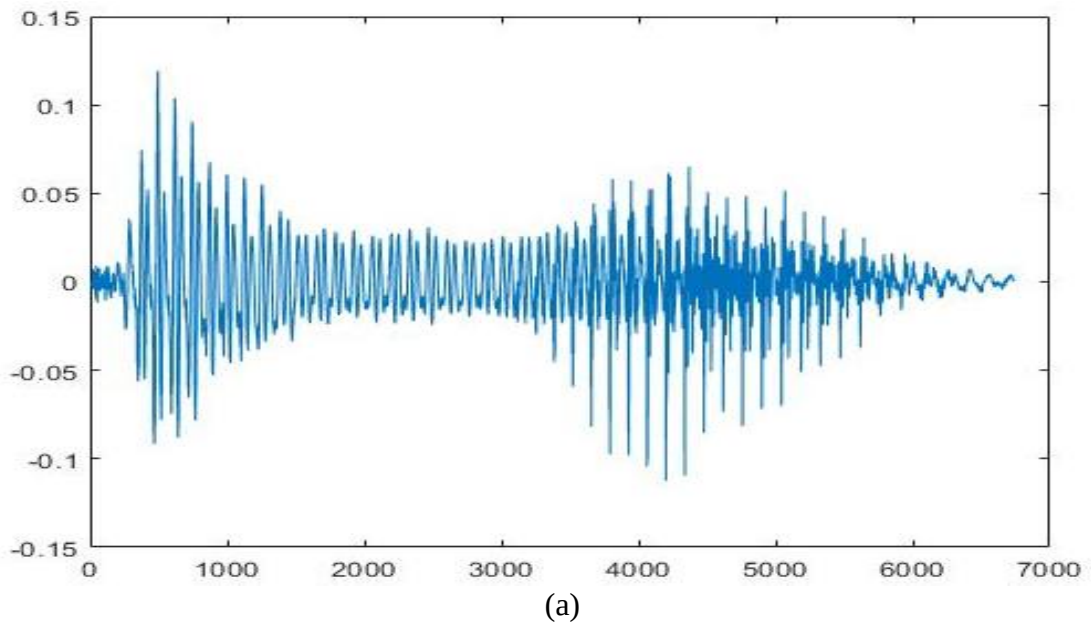


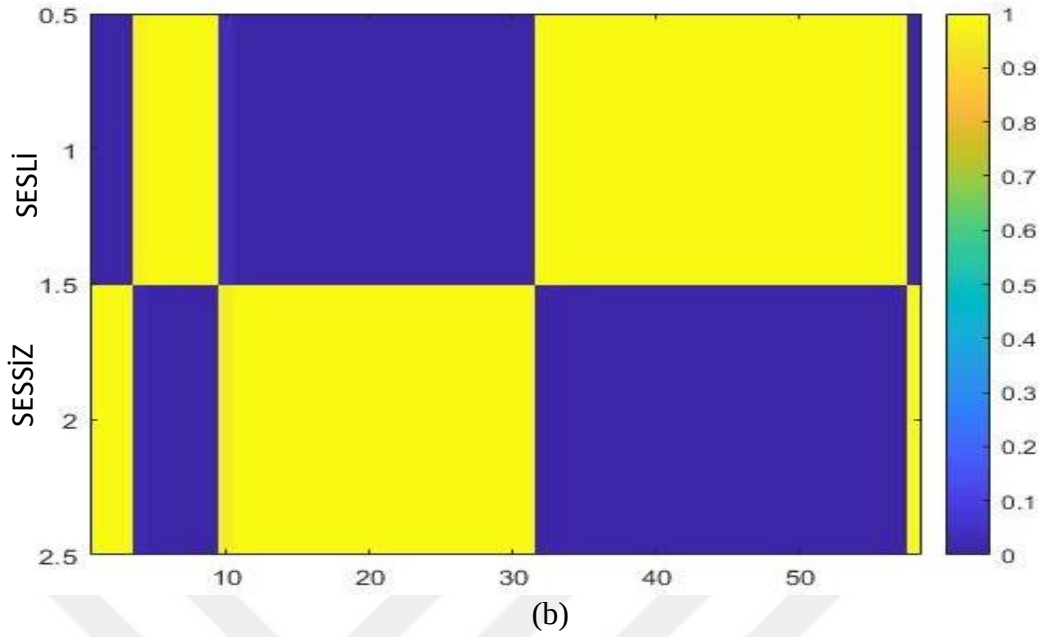
(a)



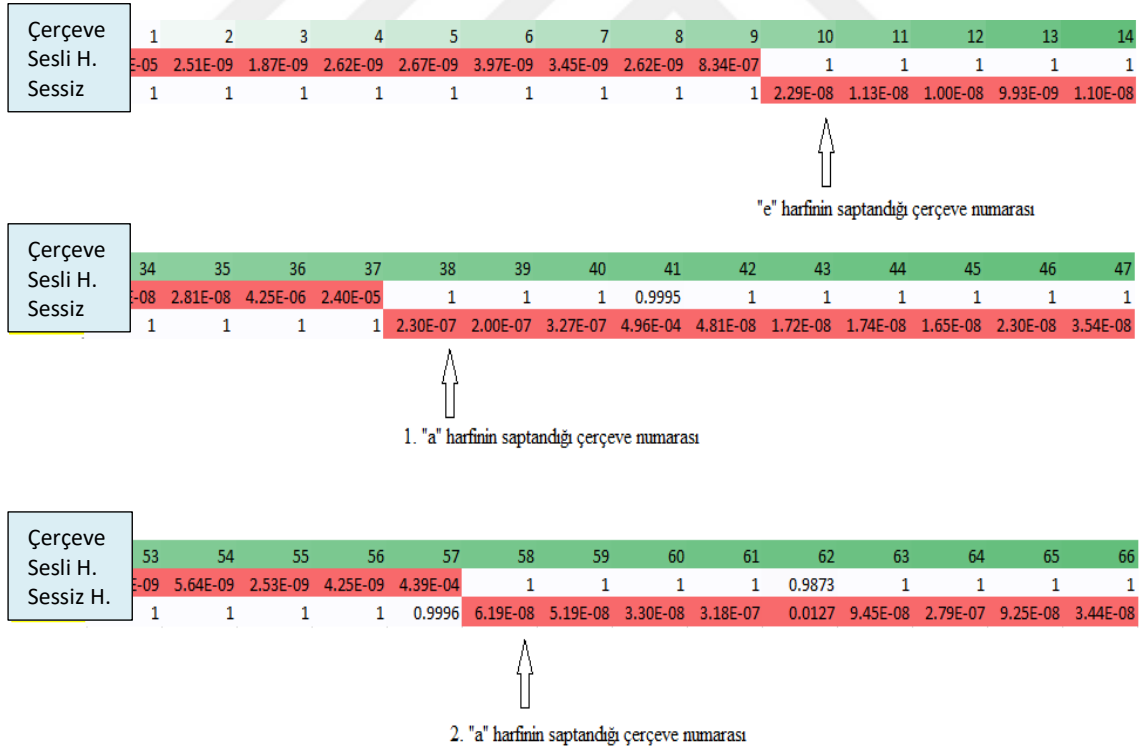
Şekil 4.8 (b), “merhaba” kelimesini içerisindeki sesli-sesiz harf ayrımını gösteren grafik, (a) zaman tanım kümesinde bu ses dosyasının grafiği.

Her iki grafikte de son çerçevelerde bir hata göze çarpmaktadır. Ses dosyası sesli harf ile bitmesine rağmen iki grafiğin son çerçevelerindeki tanımda hata göze çarpmaktadır. Bunun sebebi sesli harflerin başlangıç ve bitiş noktalarındaki çerçevelerin sessizler ile karıştırılmaya müsait olması ve yeterli karakteristik özellik barındırmaması sonucuna varıldı. Çünkü sesli harfler belirli bir salınım, kendini tekrarlayan ses karakteristiğine sahipken bu salınım sesin baş ve son kısımda etkisini yitirmektedir.





Şekil 4.9 (b) “dünya” kelimesini içerisindeki sesli-sesiz harf ayrımını gösteren grafik, (a) bu sese ait zaman tanım kümesindeki grafiği.



Şekil 4.10 “merhaba” kelimesine ait çıkış matrisi üzerindeki sesli harflerin yerleri

Sesli harflerin taptanması işleminden sonra bu sesli harflerin ses dosyasının hangi zaman içerisinde geçtiğini saptama için çıkış matrisi üzerinde bir takım hesaplamalar yapılmıştır. Şekil 4.11’de yeri tespit edilen “merhaba” kelimesinin içerisindeki harflerin yerlerin

yerleri matris üzerinde gösterilmektedir. Burada hesaplama işlemi için daha önce belirlenen çerçeve boyu ($440 \cong 0.0399$ sn.) ve kaydırma miktarı ($330 \cong 0.0299$ sn.) göz önünde bulundurularak yapılır. Örneğin “merhaba” kelimesindeki “e” harfi 10. çerçevede tespit edilmiştir. Bu çerçevenin gerçek ses dosyasındaki zaman karşılığı eşitlik 4.1’deki gibi yapılmıştır.

$$\frac{\overbrace{440}^{\text{İlk çerçeve uzunluğu (sn.)}}}{11025} + \left(\frac{\overbrace{440}^{\text{Kayma miktarı (sn.)}}}{11025} - \frac{330}{11025} \right) \times 9 = 0.1297 \text{ sn. (4.1)}$$

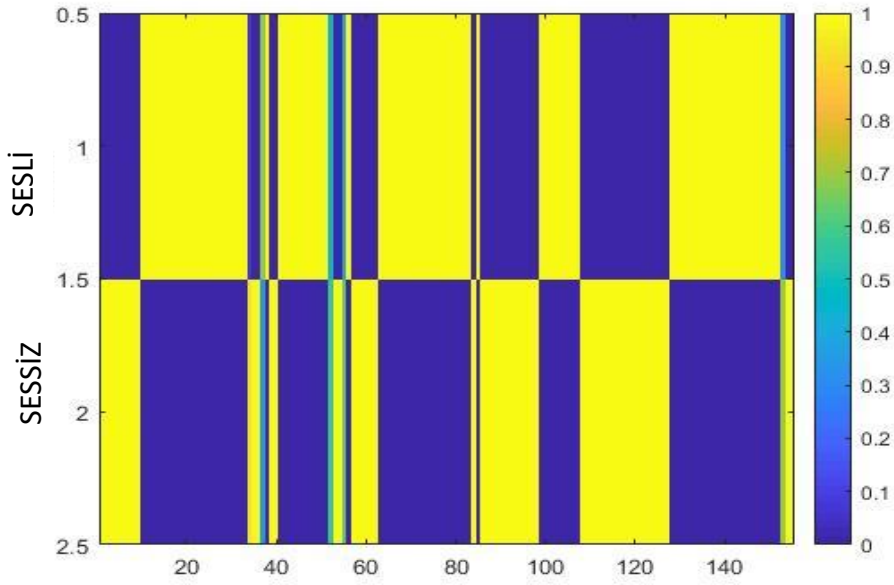
Kelime içerisindeki herf tespitinde diğer 9 kelime için de aynı yöntem uygulanarak sesli harflerin zamanları hesaplanmıştır.

Çizelge 4.14 Çalışmada kullanılan kelime ve kelime grupları

Kelime veya Kelime gurubu	Sesli harf sayısı	Sesli harflerin zamanlamaları (sn.)
Merhaba	3	0.139-0.419-0.618
Bilim	2	0.199-0.528
Dünya	2	0.099-0.329
Türkiye	3	0.126-0.529-0.752
Vatan	2	0.159-0.558
İstanbul	3	0.069-0.289-0.518
Günaydın	3	0.079-0.279-0.638
Millet	2	0.259-0.738
Cumhuriyet	4	0.089-0.628-0.937-1.167
Özgürlük	3	0.069-0.429-0.858
Ay yıldız	3	0.069-0.439-0.937
Merhaba dünya	5	0.139-0.448-0.678-1.027-1.326

Kelime çalışmalarının yanında bir de 3 farklı kelime gurubu ile çalışma yapıldı. “merhaba” ve “dünya” örnekleri incelendi. Daha sonra “merhaba dünya” adında tek bir ses dosyası üzerinden tekrar ağa gönderilerek çıktılar incelenmiştir. Şekil 4.12’de ağın

çıkış matrisinin grafiği verildi. Buradan da görüleceği üzere sesli harflerin netliği ilk duruma göre oldukça azalmış belli belirsiz çerçevelerde tanınma hataları gözlemlenmiştir.



Şekil 4.11 “merhaba dünya” kelimesinin sesli-sessiz sınıflandırma grafiği

Yapılmış olan çalışmalardan anlaşılmıştır ki kelime gruplarında harf yakalama performansı düşmektedir. Benzer durum kelimenin uzunluğu arttığında da gözlemlenmiştir. Zamanın tespiti için, buradaki hatalı tanımlanan çerçevelerin zaman tespitinde bir yanlış oluşturmaması için tespit yapıldığı çerçevelerin mümkün olduğunca uzun serilerden oluştuğu aralıklar değerlendirilmeye alındı. Çünkü hatalı tanımlana çerçeveler şekil üzerinde de fark edileceği üzere kısa aralıklıdır. Kalan diğer kelimelerin ağ çıkış matrisleri Eklerde verilmiştir.

İçerisindeki sesli harflerin zamanlamaları hesaplanan kelime (10 adet) ve kelime guruplar (2 adet) metin belgesi içerisine ilk satırda kelimeye ait metin yazısı, sonraki satırlarda her satırda bir tane olacak şekilde, zamanları ifade eden sayılar mili saniye cinsinden kaydedilmiştir. Bu kayıtlar oluşturulan bir ara yüzle okunup, heceleme ve senkronizasyon işlemi yapılmıştır.

5. TARTIŞMA ve SONUÇ

ÖTA ve KÖSA ile yapılan çalışmalarda görüleceği üzere sesli ve sessiz harfler %97-99 arası bir doğrulukta tanımlanmış ve testlerde başarılı sonuçlar vermiştir. Çalışma kapsamında ağ eğitiminde toplam 6671 (*sesli ve sessiz harfler*) çerçeve ağ eğitimlerinde kullanılmıştır. Sonraki aşamada eğitimde kullanılanlardan farklı ses kayıtları kullanılarak (*3580 anlamlı ses verisinden oluşan*) ağlar test edilip bir birleriyle kıyaslanmıştır. Bu kıyaslamaların sonucunda en iyi performansı gösteren YSA modelleri iki farklı ağ için değerlendirmeye alınmış, bu sonuçlar kapsamında ÖTA en iyi performansı göstermiştir. ÖTA yapısında üç katman kullanılmış ve derin öğrenme ile başarı elde edilmiştir. Bu sonuca varılmasında nöron sayısı, YSA saklı katman sayısı, kullanılan MFC katsayı adedi ve ses verilerinin çerçeve uzunlukları değiştirilerek yapılan testler yol göstermiştir. Ses tanıma sisteminde öncelikli hedef sesli harflerin olabildiğince yüksek başarı oranı ile tanıma işlemini gerçekleştirmesidir. Çünkü metin içerisinde hece ölçüsünde ses-kelime sekronizasyonu yapılmasında bu çok önemli bir yer tutmaktadır. Türkçede hecelerin sesli harf etrafında oluştuğunu ve bu sesli harfin yeri hecenin söylenmesi anı ile yakından ilişkilidir. Bu konunun öneminden dolayı sesli harf kayıtlarının sayısı eğitim verisinde fazladır.

Sesli-sessiz harflerle yapılan testler çok net olmasına rağmen kelime vektörleri ile yapılmış olan testler bu netlikten biraz uzak kalmaktadır. Bunun en büyük sebebi ise kelimelerle ağ test edildiğinde sistem metinden bağımsız bir çalışma kapsamına girmekte ve ağ sesleri saptamakta güçlük çekmektedir (*Dede, 2008*). Ayrıca Türkçede kelimeler içerisindeki sesli harflerin harmonisi kelimedenden kelimeye farklılık göstermekte ve konuşmacının söyleyiş biçimiyle de yakından ilişkisi bulunmaktadır (*Kılıç, 2015*). Bu sebeplerle kelimelerin içerisinde sesli veya sessiz harflerin tespitini zorlaştırmaktadır. Çalışmada kullanılan kelimeler 10 tane ile sınırlandırılmıştır. Mümkün olduğunca her harfin kelime içerisinde tespitini göstermek amacı ile rastgele kelimeler seçilmiştir. Tezin

bu kapsamında amacı sesli harflerin yakalanmasıyla hece ölçüsünde kelime-ses vurgulanmasını eş zamanlı olarak yapılabileceğini gösterilmesi olduğundan geniş bir kelime gurubu ile çalışma yapılmamıştır.

Bütün bu çalışmalar yürütülürken sistemin performansının artırılması konusunda bazı fikirler edinilmiştir. Ses tanıma sistemlerinde en önemli hususlardan biri ses kayıtları ve eğitim verisinin mümkün olduğunca geniş ve zengin tutulması meselesidir (*Sirigos ve ark., 1996*). Bu sebeple eğitim verisinin zenginliği ve ses kayıtlarının kalitesi ağ performansını önemli ölçüde etkilemektedir (*McLoughlin, 2009*). Ayrıcı kullanılacak daha karmaşık YSA modelleri ve farklı sınıflandırıcı kombinasyonlarının uyarlamaları sistem performansının geliştirilmesinde bir yol gösterici olabilir (*Vafeiadis ve ark., 2017*). Son olarak geliştirilen hibrit sistemler, yapay zeka problem çözme konusunda yeni fikirler doğmasına sebep olmaktadır (*Wang ve ark., 2006*).

Farklı sınıflandırıcı ve ses özelliklerinin kullanılması sistemin daha yüksek sonuç vermesini sağlayabilir. Böyle bir sistemin yüksek verimle çalıştırılması ileride fonem tabanlı ses tanıma, ses-yazı veya yazı-ses sistemlerinin daha verimli bir şekilde oluşturulmasının önünü de açacaktır. Böyle bir sistemin eğitimden haberleşmeye kadar bir çok alanda faydalanılabilecek bir içerik barındırmaktadır. Özetlemek gerekirse, günümüzde manuel olarak yapılan herhangi bir ses veya görselle ait metin senkronizasyonun otomatik ve yüksek hassasiyette yapılabilmesinin önü açılacaktır.

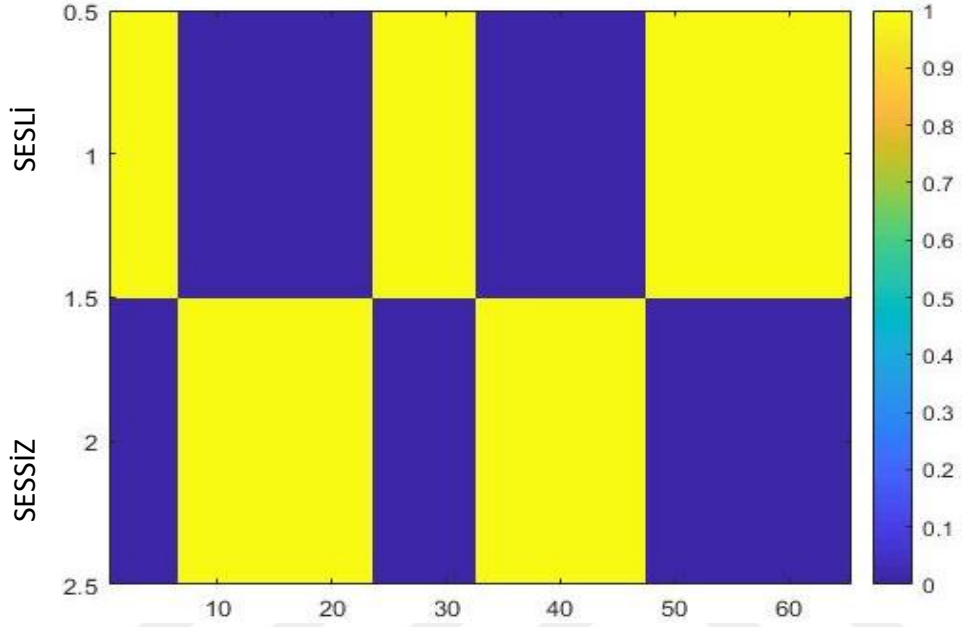
KAYNAKLAR

- Aşlıyan, R., Filiz, A. ve Günel, K., 2006. Türkçe otomatik heceleme sistemi ve hece istatistikleri, VIII Akademi Bilişim Konferansı Bilgi Teknolojileri Kongresi IV, Denizli.
- Baygün, M.K., 2006. Türkçe komutları tanıyan ses tanıma sistemi geliştirilmesi. (Yüksek lisans tezi), Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Denizli.
- Bengio, Y., 2012. Deep learning of representations for unsupervised and transfer learning. In Proceedings of ICML workshop on unsupervised and transfer learning, pp. 17-36.
- Broomhead, D. S. ve Lowe, D., 1988.variable functional interpolation and adaptive networks. Complex Systems, 3(2), pp. 231-355.
- Cakir, E., 2014. Multilabel sound event classification with neural networks. (Master thesis), Tampere University of Technology, Faculty of Computing and Electrical Engineering, Finland.
- Can, B., 2007. Bir hece-tabanlı Türkçe sesli ifade tanıma sisteminin tasarımı ve gerçekleştirimi. (Yüksek lisans tezi), Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- Chapelle, O., Schölkopf, B. ve Zien, A., 2006 . Semi-supervised learning. Cambridge:MIT press, pp. 524, London, England.
- Chicco, D. ve Jurman, G., 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score ve accuracy in binary classification evaluation. BMC Genomics, 21(6), 13 pp.
- Cosi, P., Bengua, Y. ve De Maria, R., 1990. Phonetically-based multi-layered neural networks for vowel classification. Speech Communication, 1(9), pp. 15-19.
- Çakır, M.Y., 2017.Gerçek zamanlı yüksek kalite ses tanıma. (Yüksek lisans tezi), İstanbul Sabahattin Zaim Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Dave, N., 2013. Feature extraction methods LPC, PLP and MFCC in speech recognition. International journal for advance research in engineering and technology, 4(1), 5 pp.
- Dede, G., 2008. Yapay sinir ağları ile konuşma tanıma. (Yüksek lisans tezi), Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- Delibaş, A., 2008. Doğal dil işleme ile Türkçe yazım hatalarının denetlenmesi. (Yüksek lisans tezi), İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Demir, N. ve Yılmaz, E., 2011. Türkçe Ses Bilgisi 1. Baskı. Anadolu Üniversitesi Yayını, 168s, Eskişehir.
- Demirci, D.M., 2005. Bilgisayar destekli ses tanıma sistemi tasarımı.(Yüksek lisans tezi), İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Elman, L. J., 1990. Finding structure in time. Cognitive Science, 2(14), pp. 179-211.
- Gupta, S., Jaafar, J., Ahmad, W.F.W. ve Bansal, A., 2013. Feature extraction using MFCC. An International Journal (SIPIL), Vol 4(4), pp. 101-108.

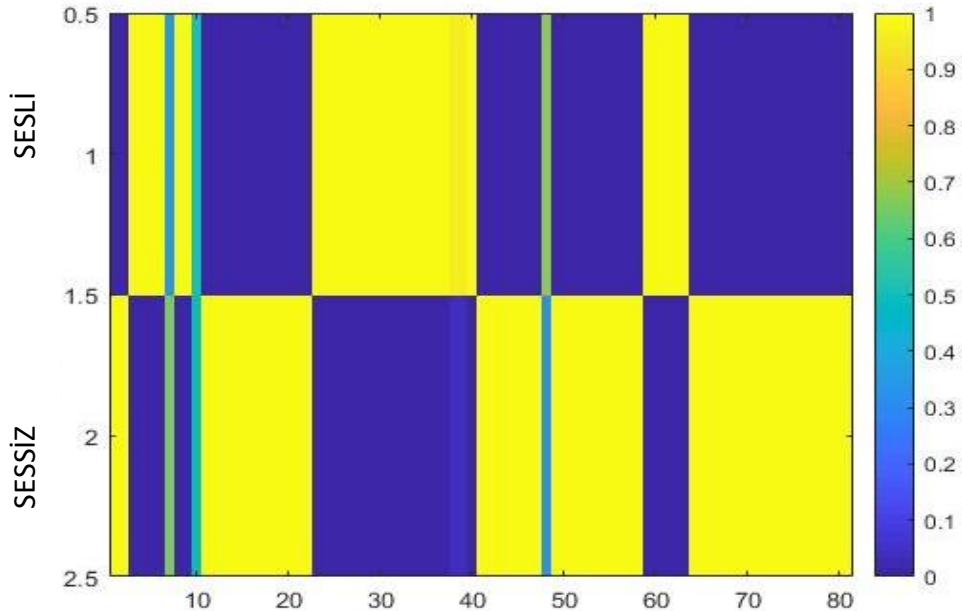
- Güloğlu, T., 2014. Speec recognition for Turkish phonology using wavelet techniques. (Master thesis), Dokuz Eylül University, Graduate School of Natural and Applied Sciences, İzmir.
- Han, W., Chan, C.F., Choy S.C. ve Pun K.P., 2006. An efficient MFCC extraction method in speech recognition, 2006 IEEE International Symposium on Circuits and Systems, Island of Kos.
- Haykin, S., 1999. Neural Networks: A Comprehensive Foundation, 2nd Edition, Prentice-Hall, pp. 823 , Ontario, Canada.
- Hinton, G., Osindero, S. ve Teh, Y. W., 2006. A fast learning algorithm for deep belief nets. Neural computation, 18(7), pp. 1527-1554.
- Joshi, C.S. ve Cheeran, A.N., 2014. Matlab based feature extraction using Mel frequency cepstrum coefficients for automatic speec recognition. International Journal of Science, Engineering and Technology Research(IJSETR), 3(6), pp. 1820-1823.
- Kılıç, E., 2015. The effects of Turkish vowel harmony in word recognition. (Master thesis), DePaul University, The Department of Psychology Collage of Science and Health, Chicago, Illinois, USA.
- Kımanuka, U., 2019. Turkish speech recognition based on deep neural networks. (Master thesis), School of Natural and Applied Sciences, Kocaeli University, Kocaeli.
- Koç, F., 2019. Spektrogram tekniği kullanılarak derin öğrenme yöntemleri ile ses tanıma. (Yüksek lisans tezi), Van Yüzüncü Yıl Üniversitesi, Fen Bilimleri Enstitüsü, Van.
- Kohonen, T., 1987. State of the art in neural computing. In Proceedings, IEEE First International Conference on Neural Networks, pp. 179-190, San Diego, USA
- Krenker, A., Bešter J. ve Kos, A., 2011. Introduction to the Artificial Neural Networks, Artificial Neural Networks - Methodological Advances and Biomedical Applications, University of Ljubljana, pp. 362, Slovenia.
- Kubat, C., 2019. Matlab yapay zeka ve mühendislik uygulamaları 4. baskı. Abaküs Kitap Yayın Dağıtım Hizmetleri, 763s, İstanbul.
- McCulloch, W. S. ve Pitts, W. 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), pp. 115-133.
- McLoughlin, L., 2009. Applied speec and audio processing with Matlab examples. Cambridge University Press, pp. 206, UK.
- Meng, Y., Lee, T., Ching, P.C. ve Zhu, Y. 2004. Speec recognition on DSP : issue on computational efficiency and performance analysis. Microprocessors and Microsystems, 30(3), pp. 155-164.
- Nwankpa, E. C., Ijomah, W., Gachagan, A. ve Marshall, S., 2018. Activation functions: comparisonds in practice and research for deep learning. arXiv, arXiv preprint arXiv:1811.03378.
- Okkan, U. Ve Dalkılıç, H. Y., 2012. Radyal tabanlı yapay sinir ağları ile kemer baraj aylık akımlarının modellenmesi. İMO Teknik Dergi, 112(23), 5957-5966s.
- Öcal, K., 2005. Otomatik konuşma tanıma algoritmalarının uygulamaları. (Yüksek lisans tezi), Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Ankara
- Önder, M., Basım Aşamasında. Kur'an-ı Kerim Eğitiminde Elmas Hece Mühendislik
- Özkaynak, E., 2013. Paralel programlama ile ses tanıma işleminin gerçekleştirilmesi. (Yüksek lisans tezi), Karabül Üniversitesi, Fen Bilimleri Enstitüsü, Karabük.
- Öztemel, E., 2003. Yapay Sinir Ağları 1. Baskı. Papatya Yayıncılık, 232s, İstanbul.

- Parlaktuna, O., Cakici, T., Tora H. and Barkana, A., 1994. Vowel and consonant recognition in Turkish using neural networks toward continuous speech recognition. Mediterranean Electrotechnical Conference, Antalya, Turkey.
- Rosenblatt, F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), pp. 386.
- Sağıroğlu, Ş., Beşdok, E. ve Erler, M., 2003. Mühendislikte yapay zeka uygulamaları-I: Yapay sinir ağları. *Ufuk Kitap Kırtasiye-Yayıncılık*, 426s, Kayseri.
- Sammut, C. ve Webb, G.I.(eds.), 2011. *Encyclopedia of Machine Learning*, Springer Science-Business Media-LLC, 1059 pp.
- Sirigos, J., V. Darsinos, N. Fakotakis ve G. Kokkinakis, 1996. Vowel-non vowel decision using neural networks and rules. *Proceedings of Third International Conference on Electronics, Circuits, and Systems*, Rodos, Greece.
- Slaney, M., 1998. *A Matlab Toolbos for Auditory Modeling Work*, version 2. Interval Research Corporation, 52 pp.
- Tezer, H.K., 2013. Gerçek zaman vema tabanlı bir sistemin gerçekleştirilmesi. (Yüksek lisans tezi), *Deniz Harb Okulu, Deniz Bilimleri ve Mühendisliği Enstitüsü, İstanbul*.
- Tiwari, V., 2010. MFCC and its application in speaker recognition. *International Journal on Emerging Technologies*, 1(1), pp. 19-22.
- Üstün, S.V., 1997. Yapay sinir ağları kullanılarak Türkçedeki sesli harflerin tanınması. (Yüksek lisan tezi), *Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul*.
- Vafeiadis, A., Kalatzis, D., Votis, K., Giakoumis, D., Tzovaras, D., Chen, L. and Hamzaoui, R., 2017, November. Acoustic scene classification: From a hybrid classifier to deep learning.
- Wang, J.C., Wang, J.F., He, K.W. ve Hsu, C.S., 2006, July. Environmental sound classification using hybrid SVM/KNN classifier and MPEG-7 audio low-level descriptor. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pp. 1731-1735
- Werbos, P. 1974. Beyond regression: New tools for prediction and analysis in the behavioral sciences. (Ph.D thesis), *Harvard University*
- Yakar, Ö., 2016. Sözcük ve hece tabanlı konuşma tanıma sistemlerinin karşılaştırılması. (Yüksek lisans tezi), *Adnan Menderes Üniversitesi, Fen Bilimleri Enstitüsü, Aydın*.
- Yalçın, N., 2006. İlköğretim birinci sınıf öğrencilerine konuşma tanıma teknolojisi yardımıyla ilkokuma yazma öğretimi. (Doktora tezi), *Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara*.
- Yavuz, E. ve Topuz, V., 2010. Recognition of Turkish vowels by probabilistic neural network using Yule-Walker AR method. *International Conference on Hybrid Artificial Intelligence Systems*, Berlin, Heidelberg, Germany.

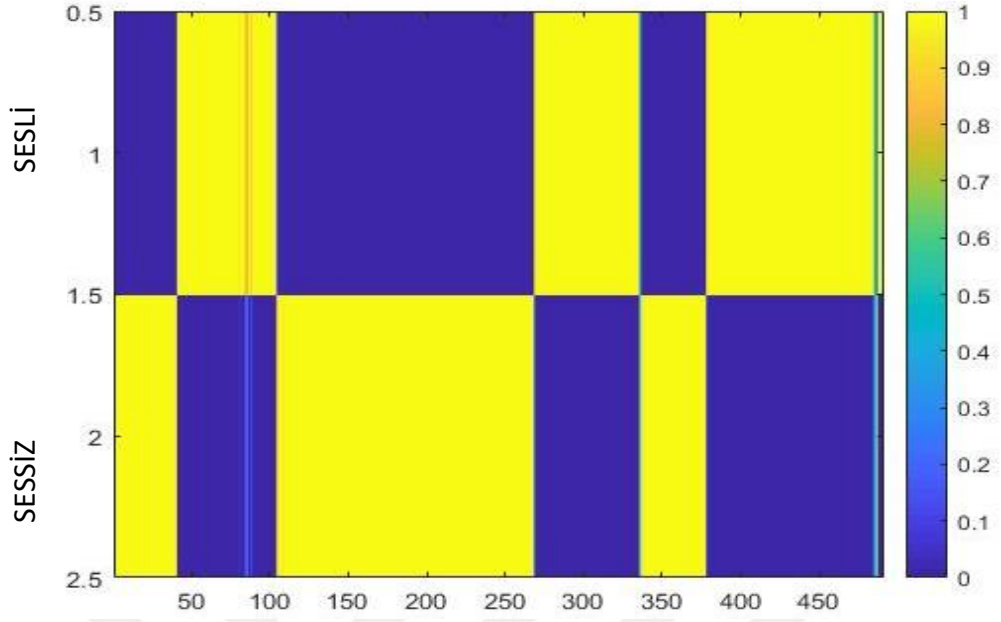
EKLER



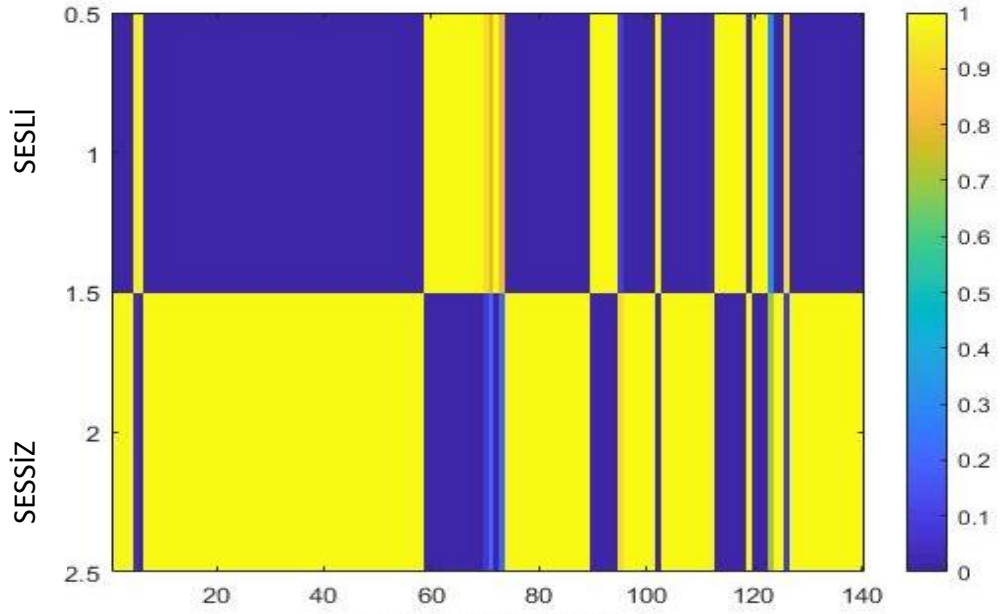
Şekil 7.1 “istanbul” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



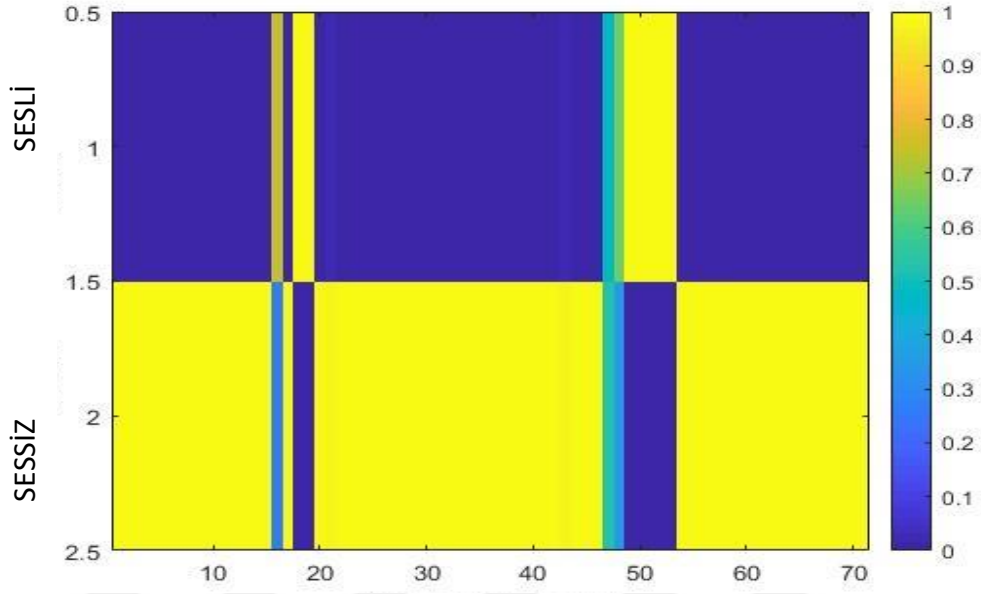
Şekil 7.2 “günaydın” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



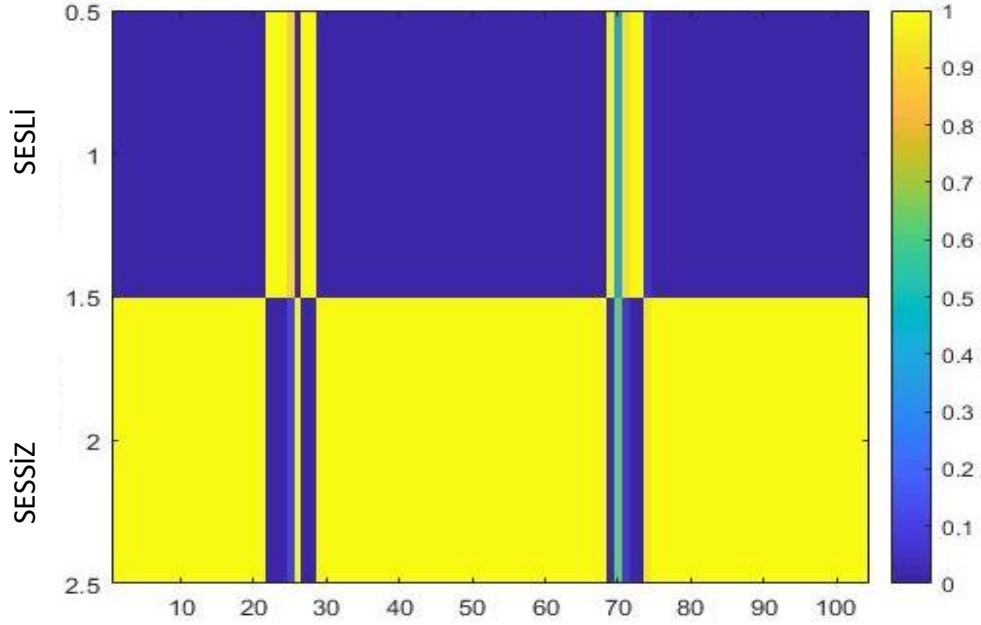
Şekil 7.3 “Türkiye” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



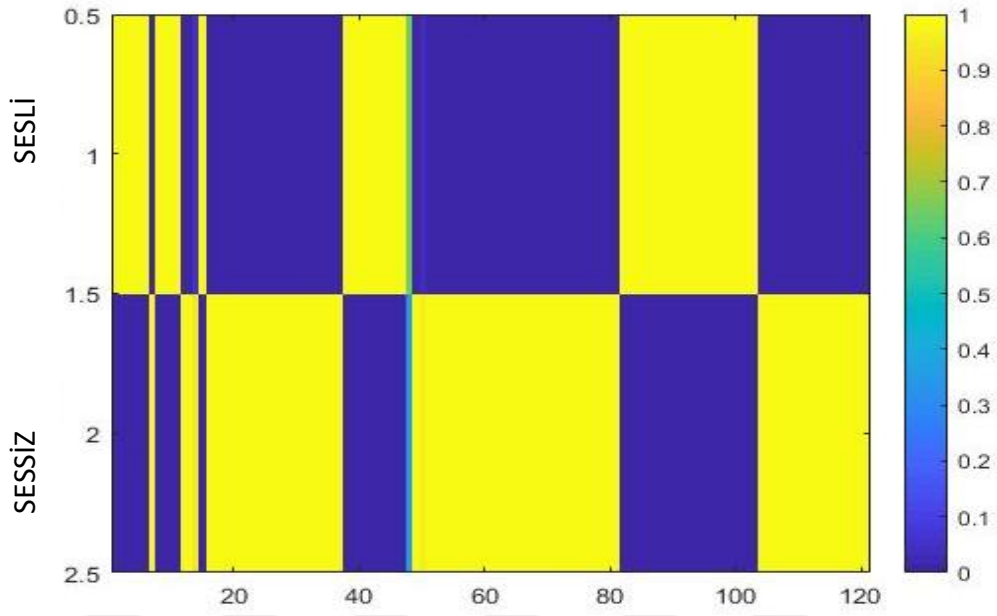
Şekil 7.4 “cumhuriyet” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



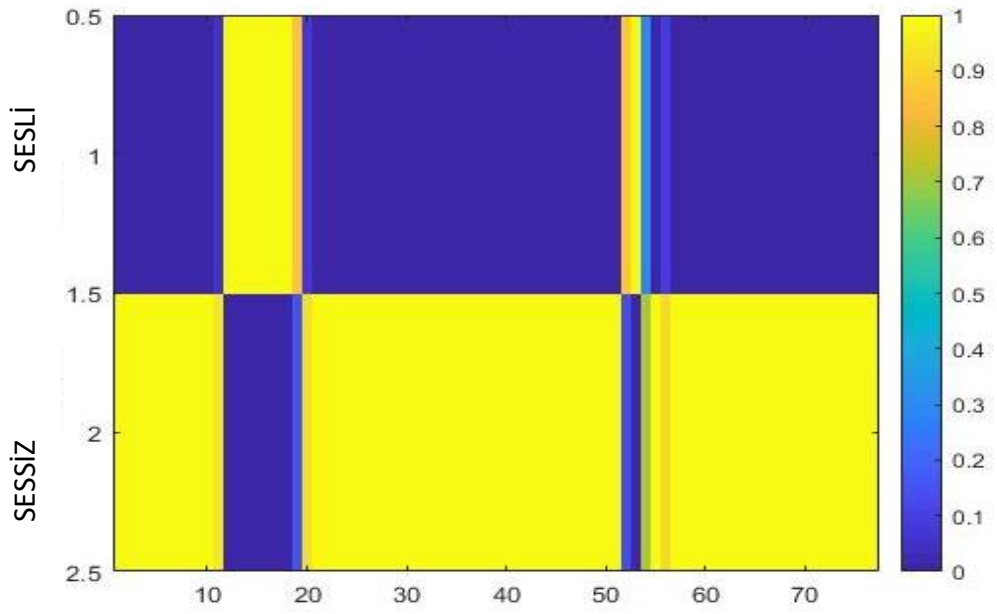
Şekil 7.5 “bilim” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



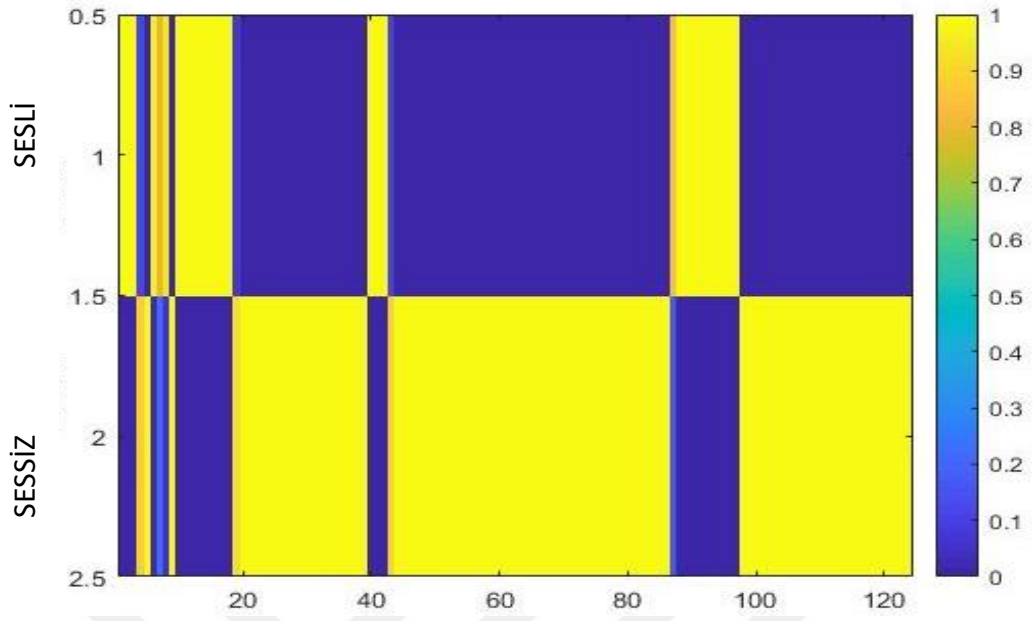
Şekil 7.6 “millet” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



Şekil 7.7 “özgürlük” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



Şekil 7.8 “vatan” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)



Şekil 7.9 “ay yıldız” kelimesinin ağ çıkış matrisi grafiği(sesli-sessiz sınıflandırma)

ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı : Halil İbrahim Bayat
Uyruğu : Türkiye Cumhuriyeti
Doğum Tarihi ve Yeri : 18.07.1991
Yabancı Dil : İngilizce (iyi)
Telefon : 541 831 7954
E-mail : halilbyt@aol.com

Eğitim Bilgileri

Derece	Alan	Eğitim Birimi	Mezuniyet
Yüksek Lisans	Mekatronik Mühendisliği Ana Bilim Dalı	Gaziosmanpaşa Üniversitesi Fen Bilimleri Enstitüsü	Haziran 2020
Lisans	Mekatronik Mühendisliği	Gaziosmanpaşa Üniversitesi	Şubat 2016
Lise	Fen Bilimleri	15 Temmuz Şehitleri Anadolu Lisesi	Haziran 2009