Journal of the Faculty of Engineering and Architecture of Gazi University 35:1 (2020) 213-224

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Turkish dialect recognition in terms of prosodic by long short-term memory neural networks

Gültekin Işık*, Harun Artuner
Computer Engineering Department, Hacettepe University, Ankara, 06800, Turkey

Highlights:

- Dialect identification using prosodic features of short speech samples
- Dialect profiling with long short-term memory neural networks
- Use of legendre polynomials in dialect recognition

Keywords:

- Turkish dialect recognition
- Long short-term memory neural networks
- Prosody
- Language model
- Legendre polynomials

Article Info:

Research Article
Received: 15.08.2018
Accepted: 15.12.2018

DOI:

10.17341/gazimmfd.453677

Correspondence:

Author: Gültekin Işık
e-mail: gultekin@cs.hacettepe.edu.tr
phone: +90 537 475 9042

Graphical/Tabular Abstract

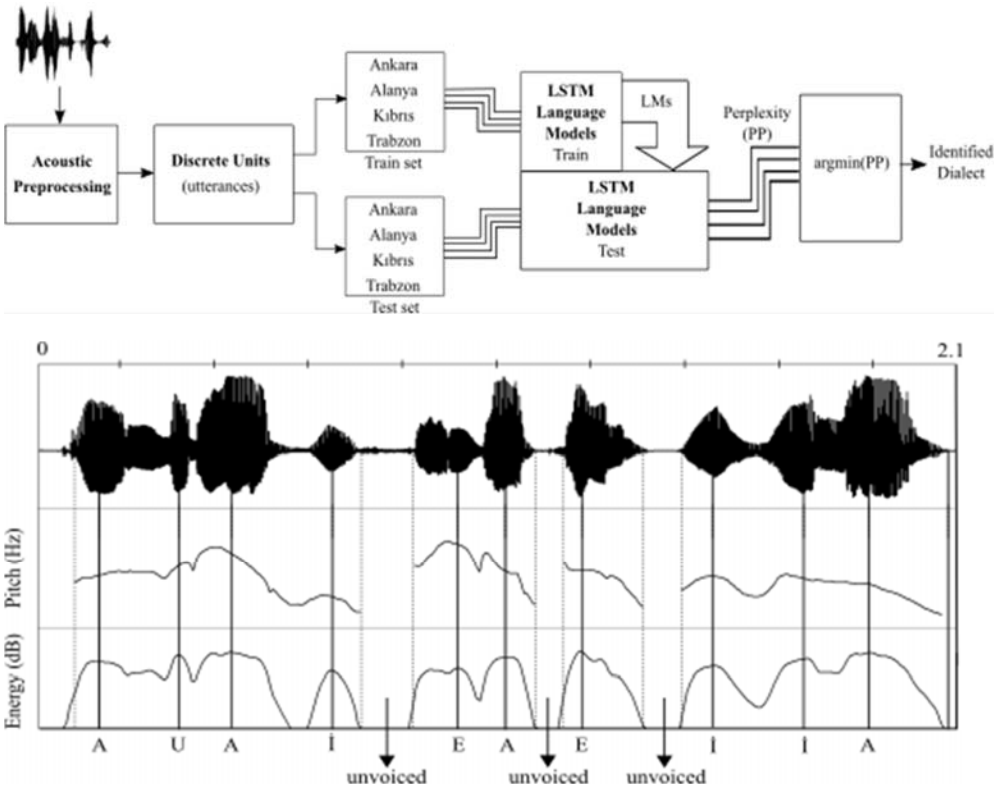


Figure A. Segmentation and identification of vowels

Purpose: The aim here is to classify the Turkish dialects by extracting their prosodic features using LSTM neural networks. For this purpose, the Legendre coefficients of the prosodic features were used in LSTM. Also the profile of each dialect was obtained using prosodic features. The results are quite satisfactory.

Theory and Methods:
LSTM neural networks are successful in modeling long-term contextual information. Here we based Mikolov's [Mikolov et al, 2010] language model and improved the Adami's [Adami, 2007] discrete units model.

Results:
It was observed that the proposed methods gave an accuracy rate of 78.7% on the Turkish dataset consisting of Ankara, Alanya, Kıbrıs and Trabzon dialects.

Conclusion:
In this study, discrete units were improved by finding the vowel identity in the syllable. LSTM neural networks are also successful in classifying Turkish dialects.

Journal of the Faculty of Engineering and Architecture of Gazi University 35:1 (2020) 213-224

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Uzun kısa-dönem bellekli sinir ağlarıyla prozodik açıdan Türkçe ağız tanıma

Gültekin Işık*, Harun Artuner
Hacettepe Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Ankara, 06800, Türkiye

ÖNEÇIKANLAR

- Kısa konuşma örneklerinin prozodik öznitelikleri kullanılarak ağız belirleme
- Uzun kısa-dönem bellekli sinir ağları ile ağız profilleme
- Legendre polinomlarının ağız tanımada kullanılması

Makale Bilgileri
Araştırma Makalesi
Geliş: 15.08.2018
Kabul: 15.12.2018

DOI:
10.17341/gazimmfd.453677

Anahtar Kelimeler:
Türkçe ağız tanıma,
uzun kısa-dönem bellekli sinir ağları,
prozodi,
dil modeli,
legendre polinomları

ÖZET

Ağızlar ait oldukları dilden bazı özellikler bakımından ayrılan ve ülkenin belli bir bölgesine özgü olan konuşma biçimleridir. Ağızlara özgü karakteristiklerin elde edilmesi ve bunlar kullanılarak ağızların tanınması, konuşma işleme alanında popüler konular arasındadır. Özellikle, büyük ölçekli konuşma tanıma sistemlerinin başarımlarını arttırmak için konuşmanın ağzının öncelikli olarak belirlenmesi istenmektedir. Diller/ağızlar birbirinden tonlama, vurgu ve ritim gibi prozodik özniteliklerle ayrılır. Bu algısal öznitelikler fiziksel düzeyde sırasıyla perde, enerji ve sürenin ölçülmesiyle elde edilmektedir. Son yıllarda, derin sinir ağlarının popüler hale gelmesiyle birlikte Uzun Kısa-Dönem Bellekli (LSTM) sinir ağları dizi sınıflandırma ve dil modelleme problemlerinde sıklıkla kullanılmaktadır. LSTM sinir ağları, uzun dönemli bağlam bilgisini modellemede başarılıdır. Bu çalışmada prozodik öznitelikler kullanılarak LSTM sinir ağları ile Türkçe ağız tanıma yapılmıştır. Burada LSTM sinir ağları hem dizi sınıflandırıcı hem de dil modelleyici olarak kullanılmıştır. Önerilen yöntemlerin Ankara, Alanya, Kıbrıs ve Trabzon ağızlarından oluşan Türkçe veri kümesi üzerinde %78,7 doğruluk oranı verdiği gözlenmiştir.

Turkish dialect recognition in terms of prosodic by long short-term memory neural networks

HIGHLIGHTS

- Dialect identification using prosodic features of short speech samples
- Dialect profiling with long short-term memory neural networks
- Use of legendre polynomials in dialect recognition

Article Info
Research Article
Received: 15.08.2018
Accepted: 15.12.2018

DOI:
10.17341/gazimmfd.453677

Keywords:
Turkish dialect recognition,
long short-term memory neural networks,
prosody,
language model,
legendre polynomials

ABSTRACT

Dialects are forms of speech, separated from languages which they belong to in terms of some characteristics and which are specific to a certain region of the country. Obtaining dialect-specific characteristics and recognition of dialects using them is among the popular topics in speech processing. In particular, the dialect of the speech is asked to be identified first in order to improve the performance of large scale speech recognition systems. Languages/dialects are distinguished from one another by prosodic features such as intonation, stress and rhythm. These perceptual features are obtained by measuring the pitch, energy and duration at the physical level, respectively. In recent years, with the increasing popularity of deep neural networks, Long Short-Term Memory (LSTM) neural networks are frequently used in sequence classification and language modeling problems. LSTM neural networks are successful in modeling long-term contextual information. In this study, Turkish dialect recognition was performed with LSTM neural networks using prosodic features. Here, LSTM neural networks were used both as sequence classifier and language modeler. It was observed that the proposed methods gave an accuracy rate of 78.7% on the Turkish dataset consisting of Ankara, Alanya, Kıbrıs and Trabzon dialects.

*Sorumlu Yazar/Corresponding Author: gultekin@cs.hacettepe.edu.tr / Tel: +90 537 475 9042

1. GİRİŞ (INTRODUCTION)

Konuşma dilinin tanınması, basitçe, sesli konuşma verisinin dilinin belirlenmesi işlemidir [1]. Konuşulan dillerin ayırt edilmesi insanlar için doğuştan gelen bir yetenektir. İnsanlar, dilleri işitme sistemindeki algısal süreçlerden geçirerek tanımaktadır. Bu yüzden insanların kullandığı bu algısal ipuçları otomatik konuşma dili tanıma çalışmalarına esin kaynağı olmuştur [2]. Genelde bilgisayar ortamında yapılan çalışmalar metin üzerinden dil tanıma üzerine olmuştur. Sesli ifadeye göre nispeten daha kolay olan metin-tabanlı dil tanıma yaklaşımı [3] dilin sözcük veya sözcük-altı birimlerine dayanır. İnsanlar üzerinde yapılan dinleme deneylerinde, dil sınıflarını belirlemek için alt düzey ve üst düzey olmak üzere iki ipucunun kullanıldığı belirlenmiştir [2]. Ses (fonetik) varlığı, fonotaktik (fonem dizimi), ritim ve tonlama gibi bilgiler, konuşma sinyalinden doğrudan elde edildikleri için bunlara alt düzey ipuçları denilmektedir [4]. Sözcükler (leksikal) ve söz dizimi (sentaks) gibi bilgiler ise üst düzey ipucu sınıfına girmektedir.

Dil tanıma, akustik yaklaşım olarak bilinen, en alt düzeyde fonemlerin spektral özniteliklerinin kullanılmasıyla başlamaktadır [5]. Bir düzey yukarı çıkıldığında, dil tanıma için fonemlerin bir dizi oluşturacak şekilde arka arkaya sıralanma kurallarını belirleyen fonotaktik yaklaşım kullanılmaktadır. Bu yaklaşımın ilk aşamasında fonem tanıyıcılar yer almaktadır. En popüler yöntemi PPRLM (Parallel Phone Recognition followed by Language Modeling, Paralel Ses Tanıma Ardından Dil Modeli) [6] yöntemidir. Ağız tanıma dil tanımanın özel bir durumudur ancak dil tanımadan daha zordur. Çünkü diller yukarıda bahsi geçen bütün düzeylerde farklılık gösterirken ağızlarda bu farklılık daha çok alt düzey özelliklerle sınırlıdır. Ağız, aynı kökten geldiği bir standart dilden belli oranda ayrılabilen yerel konuşma biçimi olarak tanımlanmaktadır [7]. Aynı şehir veya yörede doğup yaşadığı yeri çok uzun süre terk etmeyen insanların konuşma biçimleri birbirine benzemektedir. Ağızlar, ait olduğu dilden ve diğer ağızlardan; daha çok sessel (fonolojik) olmak üzere, şekilsel (morfolojik), söz varlığı (leksikal) ve çok az söz dizimi (sentaks) bakımlarından farklılık gösterir [8]. Cinsiyet, yaş gibi özelliklerin yanında ağız farklılıkları otomatik konuşma tanıma sistemlerinin performansını etkileyen önemli faktörlerdendir [9]. Bu yüzden büyük ölçekli, konuşmacıdan bağımsız otomatik konuşma tanıma sistemlerinin oluşturulabilmesi için ağız farklılıklarının ele alınması gerekir [10]. Böylece konuşulan ağzın tanınması ve konuşma tanıma sisteminin buna göre ilgili modele anahtarlanması düşünülebilir.

Türkçenin standart ağzı olarak İstanbul ağzı belirlenmiştir. Ancak Türkiye'nin çeşitli bölgelerinde konuşulan ağızlar İstanbul ağzından birçok yönüyle farklıdır. Bu farklılıklar işlenerek Türkçe konuşma tanıma sistemlerinin başarımları arttırılabilir. Ağızlar alt düzey ipuçları olan tonlama, ritim ve vurgu özniteliklerine göre diğer ağızlardan ayrılmaktadır. Bu özniteliklerin tümüne birden prozodi adı verilir. Bu öznitelikler sırasıyla temel frekans eğrisi, süre ve enerji eğrisinden türetilen parametreler kullanılarak gösterilir.

Prozodinin dil tanıma [11-13], konuşmacı tanıma [14, 15] gibi yerlerde kullanıldığı görülmektedir. Prozodik öznitelikler Legendre polinomlarıyla elde edilerek i-vektör yapılarıyla diller sınıflandırılmıştır [12]. Perde, enerji ve süre ölçümlerinin ayrık birimler haline getirilerek n-gramlarla modellenmesi ve böylece dillerin sınıflandırılması yapılmıştır [13, 16]. Arapça ağız tanıma için prozodik özniteliklerin kullanıldığı Gauss karışım modelleri (GMM, Gaussian Mixture Model) ile sınıflandırma çalışmaları bulunmaktadır [10]. Bunların yanında prozodik özniteliklerin elle çıkartıldığı ve bunlarla dillerin sınıflandırıldığı çalışmalar mevcuttur [17, 18]. Derin Sinir Ağları (DNN, Deep Neural Networks) konuşmanın doğasından gelen uzun dönemli bağımlılıkları modelleyememektedir. Bu nedenle Uzun Kısa-Dönem Bellekli Yinelemeli Sinir Ağları (LSTM RNN, Long Short-Term Memory Recurrent Neural Networks) uzun dönemli bağlam bilgisini modellemeye daha müsaittir [19]. Prozodik bilgi de böyle modellenebilecek niteliktedir. LSTM sinir ağları, derin sinir ağlarının popüler hale gelmesiyle özellikle dizi sınıflandırma ve dil modelleme problemlerinde sıklıkla kullanılmaktadır. Bu çalışmada perde ve enerji eğrileri Legendre polinomlarıyla parametrik hale getirilmiş ve polinom katsayıları öznitelik olarak kullanılmıştır. Bu öznitelikler LSTM katmanlı sinir ağıyla sınıflandırılmıştır.

LSTM sinir ağları ile geliştirilen dil modeli [20] konuşma tanıma sistemlerinde son yıllarda çokça kullanılır hale gelmiştir. LSTM ağlarıyla eğitilen dil modelleri, istatistiksel n-gram modellerden ve klasik RNN dil modellerinden [21] daha başarılı sonuçlar üretmektedir. LSTM dil modelleri sözcük ya da karakter tabanlı olarak çalışır ve bir bağlama dayalı olarak bir sonraki sözcüğün ya da karakterin tahmin edilmesini sağlar. Bu çalışmada LSTM dil modeli, ayrık birimler olarak adlandırılan prozodik öznitelikleri modellemek ve ilgili ağzın profilini çıkarmak için kullanılmıştır.

Makalenin bundan sonraki bölümleri şöyle düzenlenmiştir: Prozodik öznitelikler ve bunların elde edilmesi için gereken adımlar ikinci bölümde anlatılmış, üçüncü bölümde çalışmada kullanılan veri kümesi, LSTM sinir ağları ve bu ağlarla oluşturulan dil modelleri tanıtılmıştır. Çalışmanın dördüncü bölümünde yapılan deneyler aşamalar halinde açıklanmış ve önerilen modeller üzerinde durulmuştur. Beşinci bölümde deneylerin bulguları ve tartışma kısmına, altıncı bölümde çalışmanın sonuçlarına yer verilmiştir.

2. PROZODİK ÖZNİTELİKLER (PROSODIC FEATURES)

Konuşma olayı, bir dildeki anlamlı seslerin sıralı olarak bir araya getirilmesiyle meydana gelir. Ancak konuşma, yalnızca seslerin belli bir sırayla arka arkaya dizilmesi değil aynı zamanda doğal da olmalıdır. Bazı özellikler konuşmayı doğal hale getirir. Konuşmayı doğal hale getiren özelliklerin

tümüne prozodi adı verilmektedir. Perde değişimi, konuşmaya ayırt edilebilen melodik özellikler katar. Perdenin bu şekilde değişim göstermesiyle tonlama oluşur. Fonem ve hece düzeyindeki ses birimleri konuşmaya ritmik özellikler katmak için kısaltılıp uzatılabilir. Ayrıca, konuşmada hece veya sözcükler, diğerlerine göre vurgulu söylenerek daha belirgin hale getirilebilir [22]. Bunların yanı sıra, tonlama, ritim ve vurgu gibi prozodik öznitelikler verilen mesajın anlaşılırlığını arttırır. Sayılan bu prozodik öznitelikler, algısal düzeydeki ipuçlarıdır ve fiziksel düzeyde sırasıyla perde (f_0), süre ve enerji parametreleriyle ifade edilirler [17, 12].

Konuşmanın tonlama, ritim ve vurgu gibi prozodik öznitelikleri, konuşulan dilin kimliğine ilişkin bilgiler taşır. Konuşulan dilin belirlenmesi için yapılan dinleme deneylerinde, küçük çocukların tonlama ve ritim gibi özellikleri kullanarak karar verdikleri görülmüştür [4]. Aynı şekilde, yetişkinler de hiç aşina olmadığı diller söz konusu olduğunda, prozodi bilgilerine göre hareket etmektedir.

2.1. Prozodik Özniteliklerin Elde Edilmesi
(Obtaining of Prosodic Features)

Prozodik özniteliklerin çıkartılması için genelde 3 aşama söz konusudur [23]. Temel prozodi eğrilerinin (contour) çıkartılmasının ardından konuşma hece-benzeri birimlere ayrılır ve en sonunda bu birimler zamansal olarak modellenir. Perde ve enerji eğrilerinin çıkartılması için genelde otokorelasyon ve karekök ortalama (RMS, Root Mean Square) yöntemleri kullanılır.

2.1.1. Konuşmanın hece-benzeri birimlere ayrılması
(Segmentation of speech into syllable-like units)

Konuşmanın heceler olarak arka arkaya dizilmesi, ağzın açılıp kapanması arasında bir ritmik değişime neden olmaktadır. Bu yüzden heceler prozodik olayların merkezindedir [22]. Diller geniş anlamda; vurgu zamanlı, hece zamanlı ve mora zamanlı olarak ritmik/zamanlama özelliklerine göre ayrılmaktadır. Türkçe hece zamanlı diller grubundadır. Hece zamanlı dillerde ardışık hecelerin süresi yaklaşık olarak aynıdır. Bu nedenlerle bu çalışmada temel segment birimi olarak heceler kullanılmıştır.

Hece veya hece-benzeri birimlere ayırma işlemine segmentasyon denilmektedir. Hecenin tam bir karşılığı olmasa da içinde bir ünlü sesin olduğu birimler hece olarak kabul edilir. Hece tanımı dilden dile değiştiğinden bu birimler hece veya hece-benzeri birim olarak adlandırılmaktadır [18, 16, 10]. Konuşmanın hece veya hece-benzeri birimlere ayrılması için genelde, konuşma tanıma sistemlerinin kullanıldığı ve kullanılmadığı yöntemler söz konusudur. Konuşma tanıma sisteminin kullanılmasıyla konuşmalar fonem ve hecelerine ayrılarak segmentasyon işlemi doğal olarak yapılmaktadır [23, 16]. Bu yöntem dile büyük ölçüde bağımlı olmasına karşın segmentlere ayırma konusunda başarılıdır. Hecelerin ünlü başlangıç noktalarının tespit edilerek ayrılması [22], konuşma tanıma sisteminin kullanılmadığı yöntemlere örnek olarak verilebilir. Ayrıca konuşma sinyalinden elde edilen enerji kontöründeki vadiler kullanılarak hecelere ayırma işlemi yapılmaktadır [14, 11, 24]. Bunların yanı sıra akustik yöntemlerde uygulandığı şekliyle, sabit örtüşmeli pencereler kullanılarak da konuşmalar segmentlere ayrılmaktadır [23, 12].

2.1.2. Cümle düzeyinde modelleme (Modelling at utterance level)

Prozodik özniteliklerin çıkartılması sürecinde üçüncü aşama modellemedir. Modelleme için eğri uydurma ve profil çıkarma olarak özetlenebilecek genelde iki yaklaşım mevcuttur. Eğri uydurma yaklaşımında, segmentlerine ayrılmış perde ve enerji eğrileri ayrık kosinüs dönüşümü ile modellenerek öznitelik vektörü elde edilir [23]. Ayrıca n. dereceden Legendre polinomlarıyla da her segmentten öznitelik vektörü çıkartılmaktadır [11, 12]. Profil çıkarma yaklaşımında, her örneğin dil modeli elde edilmektedir. Perde ve enerji eğrilerinden ayrık sınıflar adı verilen birimler oluşturulmakta ve bu birimler dilin profilini çıkarmak için kullanılmaktadır [15, 16]. Genelde dil modeli oluşturmak için istatistiksel n-gram modelinden faydalanılır.

İki yaklaşımın dışında, öznitelik vektörünün elle çıkartılması da söz konusu olmaktadır. [17, 25] perde ve enerji eğrilerinden istatistik yöntemlerle özniteliklerin çıkartıldığı ilk çalışmalardandır. Ayrıca [18]'de elle çıkarılan 20 öznitelik prozodi temsili için kullanılmıştır.

2.1.2.1. Legendre polinomları ile modelleme
(Modeling with Legendre polynomials)

Dil tanımada, genelde konuşmanın kısa zamanlı kepstral öznitelikleri kullanılmaktadır. Bu özniteliklerin çıkartılması için en yaygın kullanılan yöntemlerden biri Mel Frekansı Kepstral Katsayılar (MFCC, Mel Frequency Cepstral Coefficients) yöntemidir. Legendre polinom gösterimi, MFCC gibi, gerçek verinin daha kompakt ve daha öz halini sağlamaktadır. Akustik eğrilerin modellenmesinde Legendre polinomlarının kullanılması yaygındır. Legendre polinomları ortogonal polinomlar sınıfında yer almaktadır. Ortogonal polinomlar, katsayılar arasındaki korelasyonları en aza indirgeme, yani büyüklüklerin birbirinden bağımsız olarak hesaplanması özelliğine sahiptir. Uyumdan (fitting) sonra her eğri bir modelle tanımlanmış olur. Bu model, Legendre polinomlarının toplamı olacak şekilde bir katsayılar kümesi (a_i) ile belirlenir. $f(x)$ formülü veya verisi bilinen bir fonksiyon, $P(x)$ ilgili fonksiyon noktalarından geçen model olsun. (Eş. 1)

$$f(x) = \sum_{i=0}^{N} a_i P_i(x) \quad (1)$$

Burada amaç $[-1,1]$ aralığında verilen bir $f(x)$ fonksiyonunu $n.$ dereceden $P(x)$ ile gösterilen Legendre polinomlar dizisi yardımıyla yakınsamaktır. Her bir polinomun a_i katsayısı bulunabilirse fonksiyon eğrisi üzerindeki noktalar yakınsanabilir. Fonksiyonu en iyi yakınsayan katsayıları elde etmek için en küçük kareler yöntemi ile hata miktarı bulunup düzeltme yapılmaktadır.

Polinomsal analizde, eğri ne kadar karmaşık olursa onu göstermek için o kadar çok polinom gerekir. İlk birkaç polinom fiziksel olarak yorumlanabilir: ilk polinom katsayısı a_0 ortalamayı, a_1 eğimi, a_2 parabolü, a_3 ise eğrinin dalga şeklini ifade eder.

2.1.2.2. N-gram ile modelleme (Modelling with n-gram)

Konuşma örnekleri segmentlere ayrıldıktan sonra perde (f_0) ve enerji eğrilerini tanımlayacak şekilde ayrık birimler elde edilir ve bunlar n-gramlarla modellenir [13, 15, 16]. Adami [13], perde ve enerji eğrilerini hizaladıktan sonra bunların birbirine göre durumlarını tanımlayarak ayrık sınıfları oluşturur. Bu sayede 5 ayrık sınıf oluşmaktadır: Artan perde, artan enerji (1); artan perde, azalan enerji (2); azalan perde, artan enerji (3); azalan perde, azalan enerji (4) ve ötümsüz (unvoiced) segment (5). Aynı şekilde, süresi belli bir eşik değerinden kısa olan segmentler kısa (S), diğerleri uzun (L) olmak üzere 2 sınıf belirlenir. Sonuçta, her bir örnek cümle için 5S 4S 2S 1S 3L 4L 5S 1S 2S 3S 4S 3S 2L 4S 5S gibi bir dizi ortaya çıkmaktadır. Bir dildeki bütün örnekler bu şekilde ayrık birimlerle ifade edildikten sonra her dilin n-gram modeli çıkartılır.

Bu çalışmada, konuşma örnekleri hece ortalarından segmentlere ayrıldıktan sonra her segmentteki ünlü (vowel) kimlikleri belirlenmiştir. Burada elde edilen 8 ünlü sınıfı da [15]'in ayrık sınıflarına eklenmiştir. Perde ve enerjinin birbirilerine göre artma ve azalma durumları Legendre katsayılarıyla elde edilmiştir.

3. VERİ KÜMESİ VE LSTM (DATA SET AND LSTM)

3.1. Türkçe Ağızları Veri Kümesi (Turkish Dialects Data Set)

Çalışmada Türkçe Ağızları Veri Kümesi [26] kullanılmıştır. Bugüne kadar Türkçenin ağızlarına ilişkin böyle bir veri kümesinin oluşturulmadığı bilinmektedir. Oluşturulanlar ise daha çok dil bilimcilerin çalışmalarına konu olacak niteliktedir.

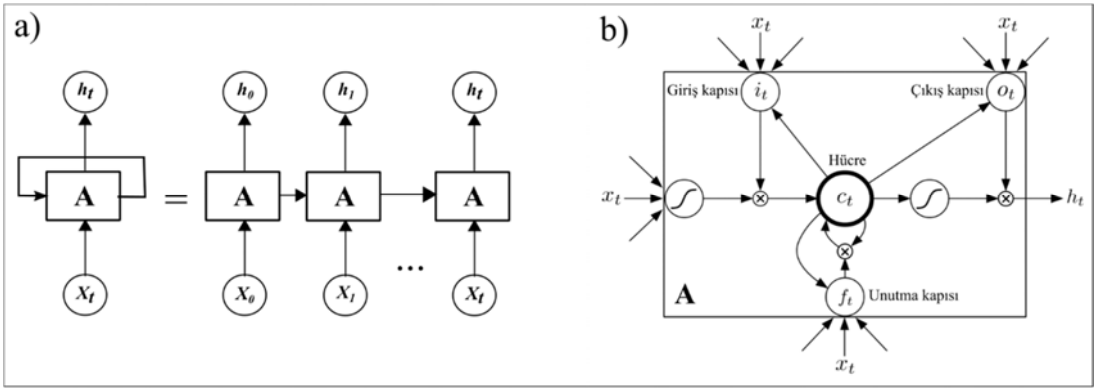
Türkçenin ağız özelliklerini inceleyen dil bilimcilerden elde edilen içeriklerin düzenlenmesiyle bir veri kümesi hazırlanmıştır. Türkçenin Ankara, Kıbrıs, Trabzon ve Alanya ağız bölgelerinden toplanan örnekler kullanılmış ve bunlar üzerinde çalışmalar yürütülmüştür. Toplanan konuşma örnekleri yüz yüze mülakat ve kaydetme yöntemiyle yapılmıştır. Bu ağızlara sahip yörelerde, ağzın özelliklerini taşıdığı düşünülen kişiler seçilmiştir. Seçilen kişilerin yaşlı olmasına, çoğunlukla bulunduğu yöreyi terk etmemiş olmasına ve eğitim seviyesinin düşük olmasına dikkat edilmiştir. Sayılan bu özellikleri sağlayan kişilerde ağza özgü seslerin bulunma ihtimalinin yüksek olduğu bilinmektedir [27].

Veri kümesi; gürültü, kanal sayısı farklılığı, örnekleme frekansı farklılığı gibi etkilerden ve sessizlik (silence) bölgelerinden arındırılmıştır. Böylece Ankara 0,8h, Kıbrıs 0,65h, Trabzon 0,55h ve Alanya 0,7h olmak üzere toplamda dört ağız bölgesi için 2,7h veri elde edilmiştir. Her bir ağız için dört kişiden konuşmalar alınmıştır. Tüm kayıtların örnekleme frekansı 16 KHz'e dönüştürülmüştür. Daha sonra konuşmalar sözcük bazında Praat yazılımıyla [28] etiketlenmiştir. Konuşma kayıtları cümlelere ayrılmış böylece her ağız bölgesi için yaklaşık 400 cümle belirlenmiştir. Her cümle yaklaşık 2-3s uzunluğundadır. Bu veri kümesi herhangi bir metne dayanmamakta ve kendiliğinden (spontane) gelişen konuşmalardan oluşmaktadır.

3.2. LSTM Sinir Ağı (LSTM Neural Network)

Yinelemeli Sinir Ağları (RNN, Recurrent Neural Networks) ardışık zaman serilerinin modellenmesini sağlayan klasik sinir ağlarının genelleştirilmiş hali olarak görülebilir. Yinelemeli sinir ağlarının hesaplama birimleri açık hale getirilerek ileri beslemeli sinir ağına dönüştürülebilmektedir. RNN'ler konuşma ve el yazısı tanıma [29], makine çevrimi [30] ve dil modelleme [21] gibi birçok ardışık yapılı probleme başarıyla uygulanmıştır.

Yinelemeli katmanın çıkış aktivasyonu, bir sonraki zaman adımında katmanın kendi girişine bağlanarak ağın verideki zamansal bağımlılıkları modellemesi sağlanır (Şekil 1a). Bu, serinin bir zaman adımını işlerken sinir ağının önceki zaman adımlarını hatırlaması demektir. Olasılık modeli açısından bakıldığında, bu yinelemeli bağlantı, modelin mevcut zaman adımındaki tahminini önceki zaman adımlarına göre



Şekil 1. a) Yinelemeli bağlantı (Recurrent connection) b) LSTM hücresi (LSTM cell)

yapmasını sağlamaktadır. RNN, türevlerin geriye doğru hesaplanması aşamasında türevin azalarak yok olmasına (vanishing gradient) veya çok yüksek değerlere çıkmasına (exploding gradient) neden olmaktadır [31]. Bu nedenle yinelemeli sinir ağlarının eğitimi pratikte zor hale gelmektedir [32]. RNN yapısında olan LSTM sinir ağları, RNN'nin bu dezavantajını ortadan kaldırmak üzere tasarlanmıştır. Bunu da ağa eklenen bellek hücreleri ve çeşitli kapılarla sağlamaktadır [33].

Şekil 1b'de LSTM'in bellekli ve kapılı yapısı görülmektedir. Her blokta yinelemeli olarak bağlanmış bir veya birden fazla bellek hücresi ve üç çarpım birimi (giriş i, çıkış o ve unutma f kapısı) vardır. Giriş kapısı girdi aktivasyonlarının bellek hücresine girişini kontrol ederken; çıkış kapısı, bellek hücresinin çıktı aktivasyonlarının ağın geri kalanına akışını kontrol eder. Unutma kapısı bellek bloğundan bellek hücresine doğru bilgi akışını kontrol ettiğinden hücrenin belleğinin silinmesini (unutmasını) sağlar. LSTM katmanının vektör hesaplama işlemleri aşağıda verilmiştir: (Eş. 2-Eş. 7)

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (2)$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \quad (3)$$

$$g_t = \tanh\left(W_{xg}x_t + W_{hg}h_{t-1} + b_g\right) \quad (4)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (5)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (6)$$

$$h_t = o_t \odot \tanh(c_t) \quad (7)$$

Yukarıdaki eşitliklerde W ağırlık matrislerini, b bias vektörlerini göstermektedir. x_t ve h_t giriş ve çıkış dizisini; i_t, f_t, o_t sırasıyla giriş, unutma ve çıkış kapılarını temsil etmektedir. g_t girişi ve önceki durumu hesaba katarak şimdiki duruma dönüştürme işlemidir. c_t hücrenin durumunu güncelleme adımıdır. $\sigma(\cdot)$ sigmoid fonksiyonu, Θ elemanlı çarpma işlemini ifade eder. t ise 1'den T'ye kadar zaman adımlarını göstermektedir. Bu hesaplamalar sonucunda, LSTM ağının çıkışında elde edilen h_t'ye softmax fonksiyonu (ϕ) Eş. 8'deki gibi uygulandığında sonsal olasılık dağılımı elde edilir:

$$y_t = \phi(h_t) \quad (8)$$

1'den T'ye kadar her zaman adımında yukarıdaki işlemler tekrar edilerek $x = (x_1, \ldots, x_T)$ giriş dizisinden $y = (y_1, \ldots, y_T)$ çıkış dizisine bir haritalama yapılmış olur. *Cross-entropy loss* fonksiyonuna göre türevler geriye yayılır. M örnek sayısını, P_{c_i} sonsal olasılıkları gösterir.(Eş. 9)

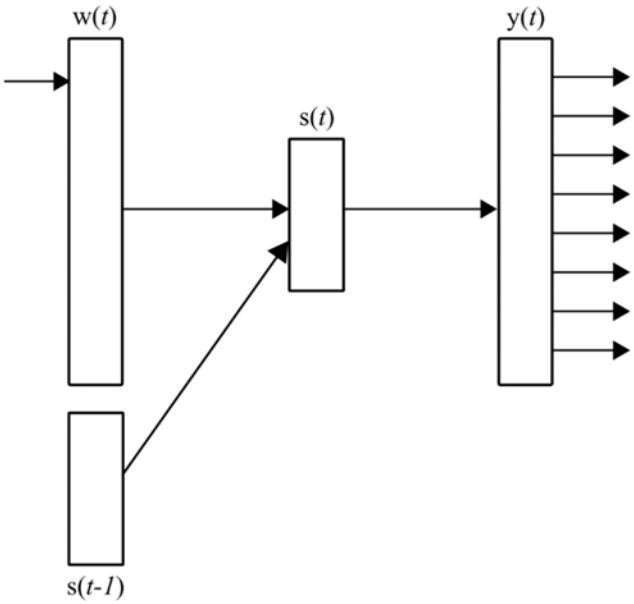
$$Loss = -\frac{1}{M}\sum_{i=1}^{M} \ln P_{c_i} \quad (9)$$

Uzun dönemli geçmiş bilgisini başarılı şekilde öğrenmesi nedeniyle LSTM sinir ağı prozodi modellemede doğal seçenek haline gelmektedir. Bu yüzden prozodik segmentlerden hesaplanan Legendre polinom katsayıları öznitelik olarak LSTM sinir ağında kullanılmıştır. LSTM'in perde ve enerji verisini kullanarak dil tanıma problemine uygulandığı bir çalışma [34] vardır. Ancak buradaki fark, segment içindeki perde ve enerji verisinin doğrudan değil Legendre katsayılarına çevrilerek sinir ağına verilmesidir. LSTM ağı Legendre özniteliklerini kullanarak ağız tanıma için çoka-bir (many-to-one) haritalama yapmaktadır.

3.3. LSTM Dil Modeli (LSTM Language Model)

Dil modeli bir cümlenin o dildeki olasılığını bulmaya yarar. Bir dilin sözcükleri veya karakterleri ile eğitilen dil modeli, cümlelere veya sözcüklere bir olasılık değeri atar. Bu olasılık değerlerine bakılarak cümlenin/sözcüğün o dil için geçerli olup olmadığı veya ne kadar geçerli olduğu anlaşılır. Birden fazla dil modeli çıkartılırsa verilen bir cümlenin/sözcüğün hangi dile ait olduğu bulunabilir. Bu durumda, cümlenin/sözcüğün ait olduğu dil modeli, diğer dil modellerine göre daha yüksek olasılık üretir.

Yinelemeli sinir ağlarının dil modelleme için kullanılması karakter veya sözcük düzeyinde olur. Şekil 2'de Mikolov [21] tarafından önerilen sistem görülmektedir. Bu mimaride RNN'in özelliğinden yararlanıp mevcut sözcükler (w) ve durum vektörü (s) kullanılarak bir sonraki sözcüğe olasılık değerleri atanır. Bunun için *one-hot* denilen, sadece ilgili sözcük pozisyonunda 1, diğer pozisyonlarda 0 olan vektörler kullanılmaktadır. Sinir ağının girişine mevcut sözcüğün *one-hot* vektörü verilirken, etiketlerini ise bir sonraki sözcüğün (hedef sözcük) *one-hot* vektörü oluşturmaktadır. Eğitim, istenen çıkışlarla sinir ağının çıkışları birbirine benzeyinceye kadar devam eder.



Şekil 2. RNN dil modeli (RNN language model).

Geçmiş sözcükleri kullanarak bir sonraki sözcük için olasılık değerlerinin hesaplanmasıyla cümlenin tamamının olasılık

dağılımı çıkartılmış olur. Böylece, olasılığın zincir kuralı (Eş. 10), RNN'nin yinelemeli yapısıyla sağlanarak dil modeli elde edilir. Burada $1:(k-1)$, indisi 1 olan sözcükten $k-1$ indisli sözcüğe kadar olan diziyi göstermektedir.

$$P(w_1, \dots, w_K) = \prod_{k=1}^{K} P\left(w_k \middle| w_{1:(k-1)}\right) \quad (10)$$

Bu çalışmada ağızların dil modeli (ağız profili veya fonolojik örüntüsü) LSTM sinir ağları ile oluşturulmuştur. Verilen bir konuşma örneği için en yüksek olasılığı üreten dil modeli, o örneğin ağız sınıfı olarak seçilmektedir.

4. SİSTEM AÇIKLAMASI (SYSTEM DESCRIPTION)

4.1. Perde ve Enerji Eğrilerinin Çıkartılması
(Extraction of Pitch and Energy Contour)

Perde ve enerji eğrilerinin elde edilmesi için Praat programının varsayılan değerleri kullanıldı. Otokorelasyon yöntemine dayalı perde izleme algoritması yardımıyla perde eğrisi, yoğunluk izleme algoritmasıyla da enerji eğrisi çıkartıldı. Enerji eğrisi, sinyalin her çerçevesine karekök ortalama yöntemi uygulanarak elde edilir. Bunların oluşturduğu en iyi eğri yolunu bulmak içinse Viterbi algoritması kullanılır.

Normalizasyon işlemi için [12]'de anlatılan yol izlendi. Perde ve enerji değerlerinin logaritması alınarak insan algılama düzeyine getirildi. Enerji değerleri maksimum değerin çıkartılmasıyla normalize edildi. Perde değerleri de ortalamanın çıkartılıp standart sapmaya bölünerek normalize edildi. Böyle yapılmasının nedeni, konuşmacılardan kaynaklanan istenmeyen değişkenliklerin azaltılmasıdır.

4.2. Segmentlere Ayırma (Segmentation)

Türkçe ağızları veri kümesi için her bir ağız bölgesinden rasgele yaklaşık 50'şer cümle seçildi ve ünlü sesler kullanılarak bu cümlelerdeki hece ortaları elle tespit edildi. Daha sonra SpeechRate betiği kullanılarak [35]'de verilen referans değerlerle otomatik olarak hece ortaları bulundu. SpeechRate betiği Praat programı için yazılmıştır. SpeechRate hece ortalarını bulurken enerji kontöründeki zirve noktalarının potansiyel hece ortası olduğunu varsayar ve ötümsüz bölgelerdeki zirve noktalarını dikkate almaz.

Otomatik olarak ve elle tespit edilen hece sayıları karşılaştırıldı ve aralarında $r = 0{,}86$ korelasyon hesaplandı. Ayrıca betik tarafından hece ortalarının bulunmasının doğruluk oranı %85 olarak hesaplandı. Sonuçta elde edilen yüksek korelasyon ve doğruluk oranı Türkçe veri kümesinde bu betiğin hece ortalarını bulmak için kullanılabileceğini göstermiştir.

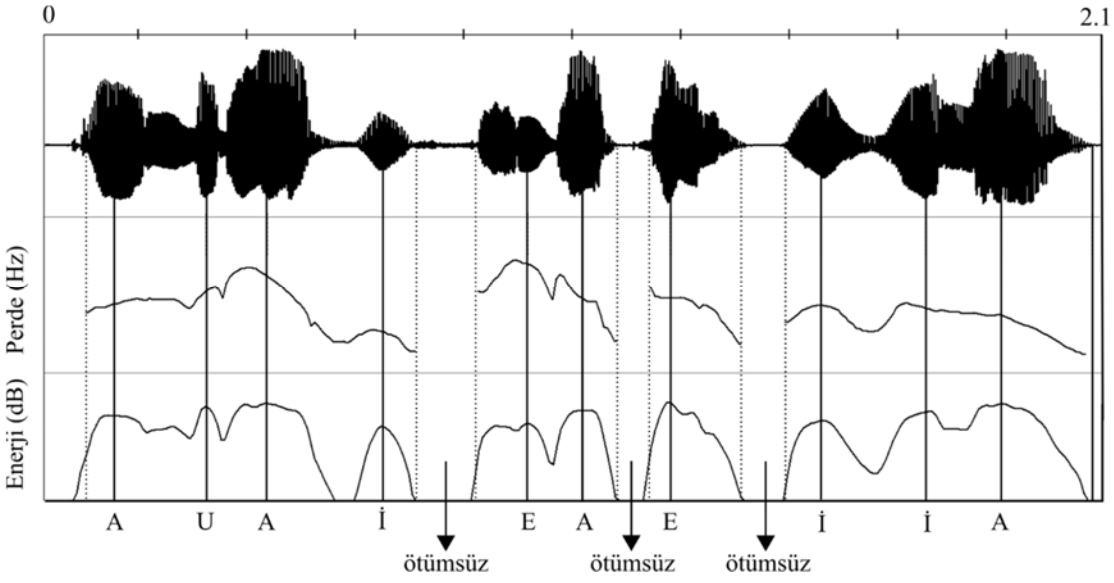
Hece ortaları bulunarak cümleler ham segmentlere ayrıldı. Ardından, sessiz bölgelerde perde kontörünün tanımsız olmasından yararlanarak cümle başı ve sonu tespit edildi. Cümle içinde, perdenin tanımsız olduğu (ötümsüz) bölgeler ise hesaba katılmadı. Böylece cümleler, üzerinde işlem yapılacak olan gerçek segmentlere ayrılmış oldu. Şekil 3'te örnek bir cümleden elde edilen perde ve enerji kontörleri ve bulunan segmentler alt alta hizalanmış şekilde görülmektedir.

4.3. Modelleme (Modeling)

Cümleler hece ortalarından segmentlere ayrıldıktan sonra modelleme aşaması gelmektedir.

4.3.1. Legendre polinomları ile modelleme
(Modelling using Legendre polynomials)

Her segmentteki perde ve enerji eğrileri, 5. dereceden Legendre polinomlarıyla yakınsandı. Verilen eğriye uyan en iyi Legendre katsayılarını hesaplamak için *numpy* kütüphanesindeki *legfit* fonksiyonu kullanıldı. Bu fonksiyon,



Şekil 3. Segmentlere ayırma ve ünlü kimliklerinin bulunması (Segmentation and identification of vowels)

verinin Legendre serisine uyumu için en küçük kareler yöntemini kullanmaktadır. Böylece normalize edilmiş gerçek perde ve enerji eğrisiyle, tahmin edilen polinom arasındaki farkı en aza indiren Legendre katsayıları hesaplanmış olur. Yakınsayan polinomun katsayıları öznitelik vektörü olarak kullanıldı. Böylece her segmentteki perde ve enerji eğrisi için 6'şar katsayı elde edildi. Her segmentin süresi, çerçeve sayısı cinsinden bulunarak toplamda 13 katsayılı öznitelik vektörü oluşturuldu. Şekil 3'teki cümlede 14 segment bulunduğundan 13×14 ebatlı öznitelik matrisi oluşturulmuştur. Her cümlenin segment sayısı farklı olduğundan matrisin boyutu değişkenlik göstermektedir.

4.3.2. Ağız profilleme (dil modeli) *(Dialect profiling (language model))*

Bu çalışmada, her segmentteki ünlü kimliği bulunarak [15]'deki ayrık birimler modeli iyileştirilmiştir. İzlenen adımlar aşağıdadır.

4.3.2.1. Ünlü kimliğinin bulunması (Identification of vowel)

SpeechRate ile hece ortaları yani ünlü seslerin yeri bulunmuştu (Şekil 3). Ünlüler Praat programı ile etiketlendi. Ünlü sesin bulunduğu çerçevenin etrafındaki $(-5, +5)$ toplam 10 çerçevenin 39 boyutlu MFCC öznitelikleri çıkartıldı. Böylece her ünlü için $39x10$ ebatlı öznitelik matrisi elde edildi. MFCC öznitelikleri şöyle çıkartıldı: Konuşma sinyaline hızlı Fourier dönüşümü uygulanarak frekans spektrumu elde edildi ve spektruma 40 kanallı Mel süzgeç bankası uygulandı. 25 ms Hamming penceresi 10 ms örtüşme süresiyle kullanıldı. 13 MFCC katsayısına ek olarak birinci ve ikinci türevlerden gelen 26 katsayı da hesaplanarak toplamda 39 katsayı elde edildi. Daha sonra ünlü seslerin öznitelik matrisleri ve etiketleri kullanılarak ünlü kimliklendirici geleneksel ileri beslemeli sinir ağı eğitildi.

İleri beslemeli sinir ağındaki katmanların düğüm sayıları şöyledir: 390-200-100-50-8. Beş katmanlı sinir ağının gizli katmanlarında sigmoid aktivasyon fonksiyonu, çıkış katmanında ise olasılıkları elde etmek için softmax fonksiyonu kullanıldı. Eğitim *cross entropy* ölçütüne göre *stochastic gradient descent* (SGD) [36] algoritmasıyla yapıldı. Test aşamasında, aynı şekilde cümleler SpeechRate betiğinden geçirilerek hece ortaları bulundu. Ünlü sesin etrafındaki 10 çerçevenin MFCC katsayıları hesaplandı ve sinir ağının girişine verildi. Böylece sinir ağı, test aşamasında %92 doğruluk oranı sağlamıştır. Bu oran, ünlü kimliğinin büyük ölçüde doğru tespit edildiğini göstermektedir. Burada her ağız için değil, bütün ağızlar için ortak bir ünlü sınıflayıcı yapılmıştır. Türkçede 8 harfe karşılık 8 ünlü fonem bulunmaktadır. Bu yüzden ileri beslemeli sinir ağının çıkış katmanı 8 sınıflıdır.

4.3.2.2. Ayrık birimlerin elde edilmesi (Obtaining discrete units)

Legendre'nin 2. polinom katsayısı bir segmentte bulunan eğrinin eğimini dolayısıyla artma-azalma özelliğini göstermektedir. Bu özellik ayrık sınıfları bulmak için kullanıldı. Perde ve enerji kontörleri Şekil 3'teki gibi hizalandığı için bunların birbirilerine göre durumlarını bulmak kolaydır. Bu çalışmada, her segmentteki ünlü kimliklerinin bulunması önerildiğinden ve ötümsüz (unvoiced) bölgelerde ünlü olmadığından ayrık sınıf sayısı 4 olarak belirlenmiştir.

Bir cümledeki segmentlerin ortalama süreleri bulunarak eşik değer olarak belirlendi. Bu eşik değerin altında kalan segmentler kısa (S), üstündekiler ise uzun (L) sınıfı olarak işaretlendi. Tanımsız perdeden (kesikli çizgi) sonra gelen ilk segment, hece ortası belirlenmiş ilk ünlüye atandı. Böylece ağız profillemenin sonunda Şekil 3'teki cümle, şu ayrık birimler haline getirilmiştir: A1S A2L U1S A4L İ4S E1S E3S A4S E3S E4L İ1S İ1L İ3L A4L.

4.4. Sınıflandırma (Classification)

4.4.1. LSTM ile sınıflandırma (Classification with LSTM)

Yukarıda elde edilen 13 boyutlu Legendre polinom katsayıları LSTM katmanlı yinelemeli sinir ağında kullanıldı. Her cümle farklı segment sayısına (N) sahip olduğundan öznitelik matrisi $13 \times N$ boyutludur. Farklı sayıdaki segmentten ötürü eğitim verileri, boyu 20 olan miniyığınlara (minibatch) ayrıldı. Böylece 3 boyutlu $(20 \times 13 \times N)$ bir veri LSTM ağına verilmektedir. Miniyığın içindeki cümleler segment sayısına göre önce sıralandı ve daha sonra sıfırla doldurma (padding) işlemi yapıldı. Cümlelerin uzunluğa göre sıralanmasının nedeni en uzun cümlenin bulunmasıdır. Kısa cümlelerin uzunluğu, en uzun cümleye sıfırla doldurma işlemi yapılarak eşitlenir. Bu sayede miniyığın içindeki uzunlukların eşit olması sağlanmıştır.

LSTM ağının giriş katmanı 13 düğümlüdür. LSTM katmanında 100 gizli düğüm vardır ve dizinin son elemanını çıkışa vermektedir. LSTM katmanından sonra 4 (ağız sayısı) düğümlü tam bağlı çıkış katmanı bulunmaktadır. Çıkış katmanında olasılıklar için softmax fonksiyonu ve *cross-entropy loss* değerini minimize etmek için SGD algoritması kullanılmıştır. Test aşamasında eğitim aşamasıyla aynı şartlar oluşturuldu. Miniyığın boyu 20 olarak belirlendi, sıralama ve sıfırla doldurma işlemi yapıldı. Veri kümesi, eğitim ve test kümeleri olarak 10-katlamalı çapraz doğrulama yöntemiyle ayrıldı. Böylece tüm veri kümesi 10 parçaya ayrıldı ve bu parçaların 9'u eğitim, 1'i de test için kullanıldı. Bu parçaların eğitim ve test olarak ayrılması işlemi 10 defa arka arkaya yapıldı ve bu 10 denemenin ortalaması alınarak sonuç skoru elde edildi. LSTM ağının kurulumu için Keras kütüphanesi [37] kullanılmıştır.

4.4.2. LSTM ile ağız profilleme yapılarak sınıflandırma *(Classification by dialect profiling with LSTM)*

Yukarıda elde edilen ayrık birimler birer sözcük olarak varsayılırsa bu sözcüklerle ilgili ağız modelini eğitmek için kullanılabilir. Ayrık birimlerin içindeki üç özellik üç ayrı

one-hot vektörle ifade edilir ve uç uca eklenirse tek bir vektör haline getirilebilir. Bu durumda oluşturulan vektöre *multi-hot* vektör denir. 8 sınıflı ünlü özelliği, 4 sınıflı perde-enerji özelliği ve 2 sınıflı süre özelliği, 14 boyutlu tek bir multi-hot vektörle ifade edilir. Örneğin A1S sözcüğünün vektör gösterimi şöyledir: 00000001|0001|01. Adami [15] ve Rouas [16] gibi çalışmalar, ayrık birimlerin her birini farklı sözcükler olarak n-gram ile modellemişlerdir. Ancak bunlar birbirinden bağımsız farklı sözcükler olarak değerlendirilirse hem vektör boyu uzar, hem de birimler içindeki özniteliklerin birbiriyle ilişkisi gözardı edilir. Örneğin ünlü sesin kimliği ile süresi arasında bir bağlantı söz konusu ise *one-hot* vektör gösterimi ve n-gram ile bu ilişki modellenemeyecektir. Hem birimler içindeki özniteliklerin, hem de birimlerin birbirileriyle ilişkisini modelleyebilmek için *one-hot* vektör yerine *multi-hot* vektör kullanılabilir.

RNN dil modelinde (Şekil 2) olduğu gibi, LSTM katmanlı sinir ağının giriş ve çıkış katmanları aynı boyuttadır. Sinir ağının girişine ve çıkışına 14 boyutlu *multi-hot* vektör verilmektedir. Örneğin yukarıda elde edilen ayrık birim cümlesi sinir ağına şöyle verilir: $T = 1$ zamanında girişe A1S vektörü verilirken çıkış etiketini A2L oluştururur. Aynı şekilde $T = 2$ zamanında girişe A2L verilirken çıkış etiketi U1S olmaktadır. Bu şekilde her bir zaman adımında bir sözcük işlenerek dizinin tamamı işleninceye kadar süreç devam eder. Bu işlem bir ağızdaki bütün eğitim verisi üzerinde tekrar edilmektedir. Bu tekrar ağın tahminleri, eğitim verileriyle tutarlı hale gelene kadar devam etmektedir. Böylece o zamana kadar olan sözcüklere göre sonraki sözcüğün tahmin edilmesi modellenerek her bir ağız için LSTM dil modeli eğitilir (Şekil 4).

LSTM ağının eğitimi *binary cross-entropy* (Eş. 11) ölçütüne göre SGD algoritmasıyla yapıldı ve türevler, zaman boyunca hatanın geriye yayılımı (BPTT, Back Propagation Through Time) [38] algoritmasıyla hesaplandı. BPTT algoritması yinelemeli sinir ağını, verilen zaman adımı sayısı kadar katmanı olan ileri beslemeli sinir ağına çevirerek türevleri hesaplar.

$$C = -\frac{1}{n}\sum_x \left(y \ln y + (1-y)\ln(1-y)\right) \tag{11}$$

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{12}$$

Burada n örnek cümle sayısıdır. *Multi-hot* vektör yapısından dolayı çıkış katmanında sigmoid aktivasyon fonksiyonu (Eş. 12) kullanıldı. Başlangıçta ağırlık matrisleri sıfıra yakın değerlerle ilklendirilerek öğrenme hızı $\propto = 0{,}1$ olarak belirlendi. Her 10 örnekten sonra ağın öğrenme kabiliyeti, doğrulama verileriyle test edildi. Bu şekilde doğrulama verisinin olasılık değerlerinin artması durumunda eğitim devam etmekte, aksi halde $\propto$ değeri yarıya düşürülmektedir. Olasılık değerinin önemli oranda artmaması halinde ise eğitim sonlandırılmaktadır. İki gizli katman ve her katmanda 50 düğüm vardır. Zaman adımı $T = 10$ olarak alındı. Bu değer, açık hale getirilen sinir ağının katman sayısına denk gelmektedir ve hafızada tutulacak sözcük sayısını gösterir. Eğitim, test ve doğrulama için konuşma örnekleri yüzde olarak 80, 10, 10 oranında ayrıldı.

Dil modellerinin başarımı *perplexity* (PP) denilen özel bir parametre ile ölçülür. İyi bir dil modeli, o dildeki bir cümleye düşük PP değeri vermelidir. Verilen bir ayrık birim cümlesi için her bir LSTM modelinin *perplexity* metriği (Eş. 13)'deki gibi hesaplanmıştır. C test verisinin entropisidir.

$$PP(C) = 2^C \tag{13}$$

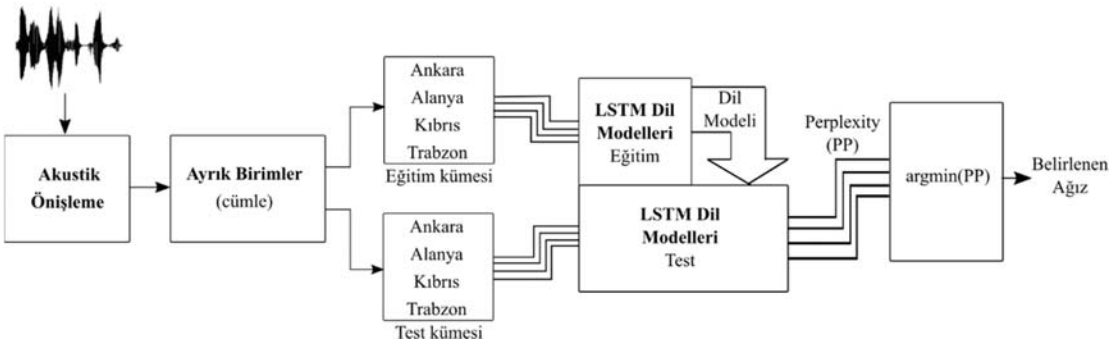
Test için ayrılan ayrık birimler, eğitilen LSTM dil modelinden geçirilerek PP değerleri üretilmektedir. Burada en düşük PP değeri veren dil modeli seçilerek ayrık birimin ait olduğu ağız sınıfı belirlenmiştir (Şekil 4).

5. SONUÇLAR VE TARTIŞMALAR
(RESULTS AND DISCUSSIONS)

Kullanılan yöntemlerin belli sürelerdeki test örneklerine göre ürettiği doğruluk oranları Tablo 1'de verilmiştir.

İkinci dereceden Legendre özniteliklerinin beşinci dereceden Legendre özniteliklerine göre daha düşük doğruluk oranı verdiği görülmektedir. Bu durum, Legendre polinom derecelerinin artmasıyla eğriyi temsil gücünün de artmasının bir sonucudur. Katsayılar arttıkça fonksiyon da eğriyi daha iyi yakınsamaktadır.

Ayrık birimlerin LSTM dil modeli ile gerçeklendiği yöntemlerin başarımları, 0,5 s ve 1 s sürelerinde Legendre katsayılarının kullanıldığı yöntemlere göre daha düşüktür.



Şekil 4. Ağızların LSTM dil modeli ile sınıflandırma mimarisi. (Classification architecture of dialects with LSTM language model)

Ayrık birimler için 0,5 s ve 1 s sürelerinin ayırıcı bilgi sağlama açısından yeterli olmadığı şeklinde yorumlanabilir. Ancak uzun test sürelerinde (3 s) bu durum tersine dönmekte ve başarım oranı artmaktadır. Dil modeli içeren yöntemlerin, n-gram ya da sinir ağı ile yapılmış dil modeli olabilir, tutarlı sonuç üretmesi için daha uzun (>3) test sürelerine ihtiyaç duymaktadır. Ayrıca 3. yöntemdeki klasik prozodik özniteliklere, ünlü kimliğinin eklenmesiyle (4) başarım oranı %76,2 olmaktadır. Ünlü kimliğinin bulunarak [10, 15, 16] sonuçlarının geliştirilmesi bu çalışmanın önerilerinden biridir. Bulunan ünlü kimlikleriyle örneğin süre arasındaki örüntü burada önerilen yöntemle modellenebilmektedir.

Tablo 1. Yöntemlerin ürettiği doğruluk oranları (%).
(Accuracy rates produced by the methods)

Yöntemler		Doğruluk oranı (%)		
		0,5 s	1 s	3 s
1	Legendre (3 katsayı) + LSTM	68,6	69,3	72,5
2	Legendre (6 katsayı) + LSTM	71,5	71,7	74,9
3	Perde, enerji, süre + LSTM Dil Modeli	68,1	70,5	75,0
4	Perde, enerji, süre + ünlü + LSTM DM	68,5	71,0	76,2
5	Perde, enerji, süre + ünlü + 3-gram DM	65,2	66,5	70,4
6	2 ve 4 birleştirilmesi	69,1	72,6	78,7

Prozodik özniteliklerin LSTM sinir ağları kullanılarak elde edilen dil modelleriyle gerçeklenmesi ve bunun sonucuna göre sınıflandırma yapılması çalışmanın bir diğer önerisidir. LSTM dil modellerinin klasik dil modellerine üstünlüğünü görmek açısından (4)'te kullanılan öznitelikler 3-gram dil modelleriyle modellenmiştir (5). Elde edilen sonuçlar sadece LSTM dil modelinin değil diğer modellerin de gerisinde kalmıştır.

En iyi yöntemlerin ürettiği olasılık değerlerinin çarpılmasıyla elde edilen en yüksek çarpım değerini veren sınıfın seçilmesine iki yöntemin birleştirilmesi (fusion) denilmektedir. Dil ve konuşmacı tanıma çalışmalarında sıklıkla bu yola başvurulmaktadır. Bu yaklaşımda, kullanılan yöntemlerin her sınıf için ürettiği olasılıklar eleman elemana çarpılır ve en yükseği veren sınıf seçilir. Altıncı yöntemde (2) ve (4)'ün sonuçları birleştirildiğinde tanıma oranı %78,7 ile en yüksek orana çıkmaktadır. Aşağıda ikinci ve dördüncü yöntemlerin 3 s test süresi için karışıklık matrisleri sırasıyla Tablo 2 ve 3'te verilmiştir.

Tablo 2. İkinci yöntemin karışıklık matrisi (%).
(Confusion matrix of the second method)

	Ankara	Alanya	Kıbrıs	Trabzon
Ankara	73,2	11,3	8,5	7,0
Alanya	9,5	75,1	10,2	5,2
Kıbrıs	8,0	9,6	74,7	7,7
Trabzon	8,1	7,5	7,8	76,6

Tablo 3. Dördüncü yöntemin karışıklık matrisi (%).
(Confusion matrix of the fourth method)

	Ankara	Alanya	Kıbrıs	Trabzon
Ankara	74,4	9,2	8,8	7,6
Alanya	8,2	75,6	10,7	5,5
Kıbrıs	7,5	9,1	76,1	7,3
Trabzon	7,8	7,1	6,4	78,7

Her iki yöntem de Trabzon ağzını diğerlerinden daha iyi ayırırken en düşük oranlar Ankara ağzı için üretilmiştir. Bu sonuç Ankara ağzının, diğer ağızlara göre, ayırt edilebilen karakteristiklere daha az sahip olduğunu gösterir. Aynı şekilde, Kıbrıs ve Alanya ağızlarının prozodik olarak birbirine diğerlerinden daha çok benzediği söylenebilir.

6. SONUÇLAR (CONCLUSIONS)

Prozodik bilginin heceler üzerinde taşındığı bilinmektedir. Segmentleme için temel birim hecedir. Heceler ise içinde bir ünlü ses barındıran birimler olarak kabul edilir. Bundan dolayı bu çalışmada, hece içindeki ünlü kimliğinin bulunması önerilmektedir.

Ayrık birimler içindeki sınıfların kendi aralarındaki ilişki *multi-hot* vektör yapılarıyla gösterilmiştir. Ayrıca ayrık birimler arasındaki uzun dönemli bağımlılıklar n-gramlar kullanılarak gösterilememektedir. Burada, uzun dönemli bağımlılığı göstermek için LSTM yinelemeli sinir ağı kullanılmıştır. LSTM sinir ağının, ağızların prozodik profilini elde etmek amacıyla kullanılması ve bu bilgiyle sınıflandırma yapılması bir ilktir. Buna ek olarak LSTM sinir ağı daha önce sınıflandırıcı olarak ham zamansal verilerle denenmiştir [34], ancak bu çalışmada giriş verisi olarak ham zamansal verilere en uygun polinom öznitelikleri kullanılmıştır.

Türkçede prozodik bilginin kullanılarak ağız tanıma daha önce yapılmadığı ve bu yönde oluşturulmuş bir başka veri kümesi olmadığı için karşılaştırma yapmak mümkün olmamaktadır. Başka dillerdeki ağızlar için yapılmış ve başka yöntemler kullanılmış çalışmaların sonuçlarıyla karşılaştırma yapmak da uygun düşmeyecektir. Ancak aynı veri kümesinde akustik özniteliklerin konvolüsyonel sinir ağlarıyla sınıflandırıldığı çalışmada [26] %83,3 doğruluk oranı yakalanmıştır. Bu oranın, bu çalışmadaki %78,7'lik doğruluk oranından yüksek olması, ağız tanımada akustik özniteliklerin prozodik özniteliklerden daha ayırt edici bilgi sağlaması ile açıklanmaktadır.

Gelecekte Türkçenin bütün ağızlarının dahil edildiği bir veri kümesinin oluşturulması planlanmaktadır. Bu sayede Türkçe ağız tanıma çalışmalarının yaygınlaşacağı ve bunun sonucunda daha iyi tanıma başarımları elde edileceği düşünülmektedir.

TEŞEKKÜR (ACKNOWLEDGMENT)

Bu çalışmada Türkçe Ağızları Veri Kümesinin oluşturulması aşamasında, işlenmemiş kayıtlarını bizimle paylaşan,

deneyimleriyle yol gösteren Prof. Dr. Nurettin Demir'e teşekkür ederiz.

KAYNAKLAR (REFERENCES)

1. Muthusamy Y. K., Barnard E. and Cole R. A., Reviewing Automatic Language Identification, IEEE Signal Process. Mag., vol. 11, no. 4, pp. 33–41, 1994.
2. Zhao J., Shu H., Zhang L., Wang X., Gong Q., and Li P., Cortical competition during language discrimination, Neuroimage, vol. 43, no. 3, pp. 624–633, 2008.
3. Kaya Y. and Ertuğrul Ö. F., A novel feature extraction approach for text-based language identification: Binary patterns, Journal of the Faculty of Engineering and Architecture of Gazi University, 31(4), 1085–1094, 2016.
4. Ramus F. and Mehler J., Language identification with suprasegmental cues: a study based on speech resynthesis., J. Acoust. Soc. Am., vol. 105, no. 1, pp. 512–21, 1999.
5. Sugiyama M., Automatic Language Recognition Using Acoustic Features, IEEE Int. Conf. Acoust. Speech, Signal Process., pp. 813–816 vol.2, 1991.
6. Zissman M. A., Comparison of four approaches to automatic language identification of telephone speech, IEEE Trans. Speech Audio Process., vol. 4, no. 1, pp. 31–44, 1996.
7. Demir N., Ağız Terimi Üzerine, Türkbilig, pp. 105–116, 2002.
8. Etman A. and Louis A. A., American dialect identification using phonotactic and prosodic features, IntelliSys 2015 - Proc. 2015 SAI Intell. Syst. Conf., pp. 963–970, 2015.
9. Huang R., Hansen J. H. L., and Angkititrakul P., Dialect/Accent Classification Using Unrestricted Audio, IEEE Trans. Audio. Speech. Lang. Processing, vol. 15, no. 2, 2007.
10. Biadsy F., Automatic Dialect and Accent Recognition and its Application to Speech Recognition, PhD Thesis,Columbia Univ., pp. 1–171, 2011.
11. Lin C. Y. and Wang H. C., Language identification using pitch contour information, ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc., vol. I, pp. 601–604, 2005.
12. Martinez D., Lleida E., Ortega A., Miguel A., Prosodic Features and Formant Modeling for an Ivector-Based Language Recognition System, 2013 Ieee Int. Conf. Acoust. Speech Signal Process., pp. 6847–6851, 2013.
13. Adami A. G. and Hermansky H., Segmentation of Speech for Speaker and Language Recognition OGI School of Science and Engineering , Oregon Health and Science University , Portland , USA International Computer Science Institute , Berkeley , California , USA, Eurospeech, pp. 1–4, 2003.
14. Dehak N., Dumouchel P., and Kenny P., Modeling prosodic features with joint factor analysis for speaker verification, IEEE Trans. Audio, Speech Lang. Process., vol. 15, no. 7, pp. 2095–2103, 2007.
15. Adami A. G., Modeling prosodic differences for speaker recognition, Speech Commun., vol. 49, no. 4, pp. 277–291, 2007.
16. Rouas J. L., Automatic prosodic variations modeling for language and dialect discrimination, IEEE Trans. Audio, Speech Lang. Process., vol. 15, no. 6, pp. 1904–1911, 2007.
17. Thyme-Gobbel A. E. and Hutchins S. E., On using prosodic cues in automatic language identification, Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96, vol. 3. 1996.
18. Ng R. W. M., Lee T., and Leung C., Spoken Language Recognition With Prosodic Features, vol. 21, no. 9, pp. 1841–1853, 2013.
19. Fernandez R., Rendel A., Ramabhadran B., and Hoory R., Prosody Contour Prediction with {Long Short-Term Memory}, Bi-Directional, Deep Recurrent Neural Networks, Proc. Interspeech, no. September, pp. 2268–2272, 2014.
20. Sundermeyer M., Schl R., and Ney H., LSTM Neural Networks for Language Modeling, Proc. Interspeech, pp. 194–197, 2012.
21. Mikolov T., Karafiat M., Burget L., Cernocky J., and Khudanpur S., Recurrent Neural Network based Language Model, Interspeech, no. September, pp. 1045–1048, 2010.
22. Mary L. and Yegnanarayana B., Extraction and representation of prosodic features for language and speaker recognition, Speech Commun., vol. 50, no. 10, pp. 782–796, 2008.
23. Kockmann M., Burget L., and Cernocky J. H., Investigations into prosodic syllable contour features for speaker recognition, in ICASSP 2010, 2010, pp. 4418–4421.
24. Ferrer L., Bratt H., Richey C., Franco H., Abrash V., and Precoda K., Classification of lexical stress using spectral and prosodic features for computer-assisted language learning systems, Speech Commun., vol. 69, pp. 31–45, 2015.
25. Foil J. T., Language identification using noisy speech, in ICASSP 1986, 1986, pp. 861–864.
26. Işık G. and Artuner H., A Dataset For Turkish Dialect Recognition and Classification with Deep Learning, in 26. IEEE Signal Processing and Communications Applications Conference (SIU), 2018.
27. Demir N., Ağız Araştırmalarında Kaynak Kişi Meselesi, Folk. Prof. Dr. Dursun Yıldırım Armağanı, p. 11, 1998.
28. Boersma P. and Weenink D., Praat: doing phonetics by computer [Computer program], 2018. [Online]. Available: http://www.praat.org/. [Accessed: 03-Feb-2018].
29. Graves A. and Jaitly N., Towards End-To-End Speech Recognition with Recurrent Neural Networks, JMLR Workshop Conf. Proc., vol. 32, no. 1, pp. 1764–1772, 2014.
30. Sutskever I., Vinyals O., and V Le Q., Sequence to sequence learning with neural networks, Adv. Neural Inf. Process. Syst., pp. 3104–3112, 2014.

31. Sak H., Senior A., and Beaufays F., Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling, Interspeech, no. September, pp. 338–342, 2014.
32. Bengio Y., Simard P., and Frasconi P., Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Networks, vol. 5, pp. 157–166, 1994.
33. Hochreiter S. and Schmidhuber J., Long Short-Term Memory, Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.
34. Cummins F., Gers F., and Schmidhuber J., Automatic discrimination among languages based on prosody alone, no. IDSIA-03-99, 1999.
35. De Jong N. H. and Wempe T., Praat script to detect syllable nuclei and measure speech rate automatically, Behav. Res. Methods, vol. 41, no. 2, pp. 385–390, 2009.
36. Bottou L., Large-Scale Machine Learning with Stochastic Gradient Descent, Proc. COMPSTAT'2010, pp. 177–186, 2010.
37. Chollet F., Keras, Github, 2015. [Online]. Available: https://github.com/fchollet/keras. [Accessed: 15-Nov-2017].
38. Williams R. J. and Peng J., An efficient gradient-based algorithm for online training of recurrent network trajectories, Neural Comput., vol. 4, pp. 491–501, 1990.