İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

DOĞAL DİL İŞLEME İLE TÜRKÇE YAZIM HATALARININ DENETLENMESİ

YÜKSEK LİSANS TEZİ

Müh. Aynur DELİBAŞ

Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ

Programı : BİLGİSAYAR MÜHENDİSLİĞİ

HAZİRAN 2008

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

DOĞAL DİL İŞLEME İLE TÜRKÇE YAZIM HATALARININ DENETLENMESİ

YÜKSEK LİSANS TEZİ

Müh. Aynur DELİBAŞ
504051504

Tezin Enstitüye Verildiği Tarih : 5 Mayıs 2008
Tezin Savunulduğu Tarih : 11 Haziran 2008

Tez Danışmanı : Prof.Dr. Eşref ADALI

Diğer Jüri Üyeleri Doç.Dr. Zehra ÇATALTEPE (İ.T.Ü.)

Yrd.Doç.Dr. Banu DİRİ (Y.T.Ü.)

HAZİRAN 2008

ÖNSÖZ

Tez çalışmam boyunca her zaman bana destek olan ve vakit ayıran tez danışmanım Prof. Dr. Eşref Adalı'ya sonsuz teşekkürlerimi sunarım. Kendisi gibi işine değer veren ve işini çok seven bir hoca ile çalışmış olmanın bana çok şey kattığını vurgulamak isterim.

Tez çalışmam süresince vermiş olduğu teknik desteklerinden ötürü Türk Dil Kurumu Sözlük Uzmanı Sayın Belgin Aksu' ya teşekkürlerimi iletirim.

Hayatımın her aşamasında bana destek olan ve inanan, beni yaptığım her işte yüreklendiren ve başarıya ulaşmamı sağlayan aileme bana güvendikleri ve her zaman yanımda oldukları için çok teşekkür ediyorum. Ayrıca tez çalışmam süresince göstermiş olduğu sabır ve anlayışı için sevgili Kamil Yıldırgan'a teşekkürü bir borç bilirim.

Mayıs 2008 **Aynur DELİBAŞ**

İÇİNDEKİLER

KISALTMALAR **v**
TABLO LİSTESİ **vi**
ŞEKİL LİSTESİ **vii**
ÖZET **ix**
SUMMARY **x**

1. GİRİŞ **1**
1.1 Doğal Dilin Temel Özellikleri 1
1.2 Doğal Dil İşlemenin Temel Elemanları 2
1.3 Doğal Dil İşlemenin Uygulama Alanları 4
1.4 Türkçe Dili ve Biçim Birimsel İncelenmesi 6
1.5 Benzer Çalışmalar 7
1.6 Tezin Amacı, Kapsamı ve Katkısı 12
1.7 Tezin Bölümleri 13

2. TÜRKÇE'NİN SES BİLGİSİ YAPISI **15**
2.1 Türkçe'nin Ses Özellikleri 15
2.2 Türkçe'de Sesler 17
2.2.1 Sesli Harfler 17
2.2.2 Sessiz Harfler 18
2.3 Türkçe'de Ses Olayları 18
2.3.1 Türkçe'de Bulunan ve Bulunmayan Sesler 19
2.3.2 Sesli Uyumları 19
2.3.3 Sessiz Uyumu 21
2.3.4 Sessiz Yumuşaması 22
2.3.5 Sessiz Benzeşmesi 23
2.3.6 Ses Değişmeleri 24
2.3.7 Sesliler Çatışmasının Giderilmesi 24
2.3.8 Ses Düşmeleri 24
2.4 Heceler 25
2.4.1 Türkçe'de Hece Yapısı 26
2.5 Sözcükler 27
2.6 Kökler ve Ekler 28
2.6.1 Anlamlı Birimler 28

2.6.2 Anlamsız veya Görevli Birimler 28

3. GELİŞTİRİLEN YÖNTEMLER VE ALGORİTMALAR 30
3.1 Sözlük Yapısı 30
3.2 Heceleme Algoritması 33
3.3 Kök Bulma Algoritması 35
3.3.1 Yönlü Döngüsüz Kelime Çizgesi Yapısı 36
3.3.2 Yönlü Döngüsüz Kelime Çizgesi Oluşturma 37
3.3.3 Ağaç Üzerinde Kök Bulma 39
3.4 Sözcüğün Biçim Birimsel Çözümlenmesi 40
3.4.1 Ekler 40
3.4.2 Ayrıştırma Algoritması 41
3.5 Türkçe Denetimi 43
3.5.1 Sözcük Başı Sessiz Denetimi 44
3.5.2 Sözcük Sonu Sessiz Denetimi 44
3.5.3 Sözcük İçi Sessiz Denetimi 46
3.5.4 Sesli Uyumları 48
3.5.5 Sessiz Uyumu 50
3.6 Ses Bilgisi Açısından Sözcük Denetimi 51
3.6.1 Sessiz Yumuşaması Denetimi 52
3.6.2 Sesli Düşmesi Denetimi 53
3.7 Sözcük Önerme 56
3.7.1 Sözlükte Bulunan Sözcükler İçin Önerme 56
3.7.2 Sözlükte Bulunmayan Sözcükler İçin Önerme 57

4. TEZİN BAŞARIMI VE SINAMA SONUÇLARI 60
4.1 Bilerek Hata Oluşturma Yöntemiyle Yazılımın Sınanması 60
4.1.1 Sınama Verisi Üzerinde Hata Oluşturulması 60
4.1.2 Sınama Sonuçları 61
4.1.3 Sistemin Başarımının Ölçülmesi 61
4.2 Sınama İşleminin Farklı Sınama Verileri Üzerinde Gerçekleştirilmesi 63
4.2.1 Sınama Sonuçları 63
4.2.2 Sistemin Bulduğu Hataların Dağılımı 65
4.2.3 Sistemin Başarımının Ölçülmesi 66

5. SONUÇLAR VE ÖNERİLER 67

KAYNAKLAR 68

EK A. TÜRKÇE SÖZCÜKLERE EKLENEN ÇEKİM EKLERİ 71

EK B. TÜRKÇE SÖZCÜKLERE EKLENEN YAPIM EKLERİ **73**

ÖZGEÇMİŞ **78**

KISALTMALAR

DDİ	: Doğal Dil İşleme
DLD	: Damerau-Levenshtein Distance
DAWG	: Directed Acyclic Word Graph
NLP	: Natural Language Processing
TDK	: Türk Dil Kurumu
SS	: Sert Sessiz Harfler
YSB	: Sert Karşılığı Bulunan Yumuşak Sessiz Harfler
YSM	: Sert Karşılığı Bulunmayan Yumuşak Sessiz Harfler

TABLO LİSTESİ

		Sayfa No
Tablo 1.1	"çiçekleri" kelimesi için biçim birimsel çözümleme sonuçları.....	6
Tablo 1.2	"gider" kelimesi için sözcük türü sonuçları.....	7
Tablo 2.1	Harflerin Türkçe'deki kullanım sıklıkları.....	16
Tablo 2.2	Dillerin sesli ve sessiz harf kullanım oranları.....	16
Tablo 2.3	Türkçe'deki sesli harflerin sınıflandırılması.....	18
Tablo 2.4	Türkçe'deki sessiz harflerin sınıflandırılması.....	18
Tablo 2.5	Türkçe'de sessiz uyumu için sessiz harflerin sınıflandırılması.....	21
Tablo 2.6	Türkçe'de sözcük ve hece sonunda çift sessiz kuralı.....	22
Tablo 2.7	Türkçe'de isim soylu sözcüklere eklenebilecek çekim ekleri.....	29
Tablo 2.8	Türkçe'de eylem soylu sözcüklere eklenebilecek çekim ekleri.....	29
Tablo 4.1	Yazılım sınama sonuçları.....	61
Tablo 4.2	Makaleler için yapılan sınama sonuçları……..	64
Tablo 4.3	Masallar için yapılan sınama sonuçları……	64
Tablo 4.4	Elektronik kitaplar için yapılan sınama sonuçları……	64
Tablo 4.5	Tez dokümanı için yapılan sınama sonuçları……...	65
Tablo 4.6	Sistemin bulduğu hataların dağılımı…….	65

ŞEKİL LİSTESİ

Sayfa No

Şekil 1.1 : Doğal dil işleme sistemlerinin genel blok diyagramı 2
Şekil 1.2 : Ayrıştırma ağacı örneği..... 3
Şekil 2.1 : Türkçe' de seslilerin dizilişini gösteren sonlu durum makinesi.... 20
Şekil 2.2 : Türkçe' de sessizlerin dizilişini gösteren sonlu durum makinesi.. 22
Şekil 2.3 : Türkçe' de bir sesin boğumlanması..... 26
Şekil 2.4 : Türkçe hece yapısında ses iniş ve çıkışları..... 27
Şekil 3.1 : Heceleme algoritması akış diyagramı..... 35
Şekil 3.2 : "BAL" ve "BAS" sözcükleri için yönlü döngüsüz kelime çizgesi.36
Şekil 3.3 : "BOL" ve "KOL" sözcükleri için yönlü döngüsüz kelime çizgesi37
Şekil 3.4 : "BAL" ve "SAL" sözcükleri için ağaç oluşumu……... 37
Şekil 3.5 : "balta" sözcüğünün ağaca yerleştirilmesi……. 38
Şekil 3.6 : Sözcük sonlarının ortak düğümlere bağlanması……... 38
Şekil 3.7 : Ağaç üzerinde kök bulma……... 39
Şekil 3.8 : Ayrıştırma algoritması akış diyagramı ……..... 42
Şekil 3.9 : Türkçe denetimi akış diyagramı…… 43
Şekil 3.10 : Sözcük başı sessiz harf sayısı denetimi için akış diyagramı……. 44
Şekil 3.11 : Sözcük sonu sessiz harf sayısı denetimi için akış diyagramı…… 46
Şekil 3.12 : Sözcük içindeki sessiz harf sayısı denetimi için akış diyagramı... 47
Şekil 3.13 : Büyük sesli uyumu akış diyagramı……... 49
Şekil 3.14 : Küçük sesli uyumu akış diyagramı……... 50
Şekil 3.15 : Sessiz uyumu akış diyagramı…… 51
Şekil 3.16 : Sessiz yumuşaması algoritması akış diyagramı…… 53
Şekil 3.17 : Sesli düşmesi algoritması akış diyagramı……. 55
Şekil 3.18 : Sözcük önerme algoritması akış diyagramı..... 56
Şekil 3.19 : DLD algoritması sözde kod gösterilimi…… 59
Şekil 4.1 : Karmaşıklık matrisi……... 62

DOĞAL DİL İŞLEME İLE TÜRKÇE YAZIM HATALARININ DENETLENMESİ

ÖZET

Doğal dil işleme, ana işlevi, doğal bir dili çözümleme, anlama, yorumlama ve üretme olan bilgisayar sistemlerinin tasarımını ve gerçekleştirilmesini konu alan bir bilim ve mühendislik alanıdır. DDİ, yapay zeka, biçimsel diller kuramı, kuramsal dilbilim ve bilgisayar destekli dilbilim gibi çok değişik alanlarda geliştirilmiş kuram, yöntem ve teknolojileri bir araya getirir. 1960'lı yıllarda yapay zekanın bir alt alanı olarak görülen bu konu, araştırmacıların ve gerçekleştirilen uygulamaların elde ettiği başarılar sonucunda artık bilgisayar bilimlerinin konusu olarak kabul edilmektedir. DDİ alanındaki araştırmalarda temel amaçlar şu şekildedir:

- Doğal dillerin işlev ve yapısını daha iyi anlamak
- Bilgisayarlar ile insanlar arasında arabirim olarak doğal dil kullanmak ve bilgisayar ile insan arasındaki iletişimi kolaylaştırmak
- Bilgisayar ile dil çevirisi yapmak

Doğal dillerin yapısının anlaşılması için doğal dilin ayrıntılı bir çözümlemesinin yapılması ve matematiğinin çıkarılması gerekmektedir. Bu nedenle, dilin kuralları mühendisçe bir yaklaşımla ele alınmakta ve dilin genel yapısı, kuralları ve aykırılıkları ortaya çıkarılmaktadır.

Bu tez çalışmasında, Türk dilinin matematiğinin çıkarılması ve biçimbirimsel analizinin yapılması üzerinde çalışılmıştır. Tez çalışmasındaki hedef, girilen bir Türkçe metnin yazım hatalarının ayıklanması ve düzeltilmesidir. Bunun için, girilen metnin Türkçe'nin ses ve dilbilgisi kurallarına uygun olup olmadığının araştırması yapılmıştır. Tez kapsamında girilen metindeki tümcelerde bulunan her kelimenin köklerinin bulunması, eklerinin ayrıştırılması, kelimenin doğru mu yanlış mı yazılmış olduğuna karar verilmesi, yanlış yazılmış kelimeler için sözcük önerilmesi ve Türkçe olmayan yabancı kelimeler için sözlüğe ekleme yapılması gerçeklenmiştir.

TURKISH SPELL CHECK WITH NATURAL LANGUAGE PROCESSING

SUMMARY

Natural Language Processing is a science and engineering area which designes and applies computer systems that are used in parsing, understanding, processing and producing of natural languages. NLP covers lots of theories, methods and technologies that are developed in different areas such as artificial intelligence, theory of morphological languages, theoritical linguistics and computer supported linguistics. This science area which was a sub topic of artificial intelligence in 1960s, is now a part of computer science with the success of researchers and applications succeeded. Following items shows the fundamental purposes of researches in NLP area:

- Understanding the structure and functionality of natural languages
- Using natural languages as interface between people and computers; so that making the communication better between people and computers
- Translating words from one language to another using computers

In order to understand the structure of a natural language, the natural language needs to be analysed and its mathematical structure needs to be clarified. As a result of this situation, the rules of language is analysed with a view of engineer and the general structure, rules and exceptions of language are defined.

In this study, it was worked on to clarify the mathematical model of Turkish language and to analyse its morphological structure. The purpose of this thesis is to make spell check for a given Turkish text and to correct the faulty situations. For this purpose, Turkish text is examined in order to check if it obeys the rules of Turkish grammer or not. It was worked on findind roots of each word, parsing suffixes from roots, checking words in order to decide whether they are written truly or not, making word suggestion for the words which are not written truly and adding words which do not belong to Turkish language to a newly created lexicon by user.

1. GİRİŞ

İnsanlar arasındaki haberleşmede en büyük faktör kuşkusuz dildir. Doğal dil işleme çalışmaları dil faktörünün insan-bilgisayar haberleşmesinde en etkili biçimde kullanılmasını amaçlar.

1.1 Doğal Dilin Temel Özellikleri

Bütün doğal dil işleme çalışmaları uygulamada farklılıklar göstermesine rağmen temelde aynı adımlardan geçer. Doğal dil işleme alanındaki çalışmalar dört ana başlık altında incelenebilir:

Sesbilim: Harflerin seslerini ve bunların dil içinde nasıl kullanıldığını inceler. Tüm dillerin bir alfabesi vardır ve her harfin sesi diğerlerinden farklıdır. Sesbilimde amaç, konuşulan dili yazılı dile çevirmektir. Sesler sözcükler haline getirilmeye çalışılır.

Biçim birim: Bu aşamada sözcükler tek başlarına ele alınırlar ve dilin kurallarına uygun şekilde sözcüğün yapısı incelenir. Bu incelemenin sonunda herbir sözcüğün her parçasının çözümlemesi yapılmış olur. Ekler, kökler, bunlara ilişkin kurallar ve bu yapıların sınıflandırılmaları biçimbirim kapsamında ele alınır.

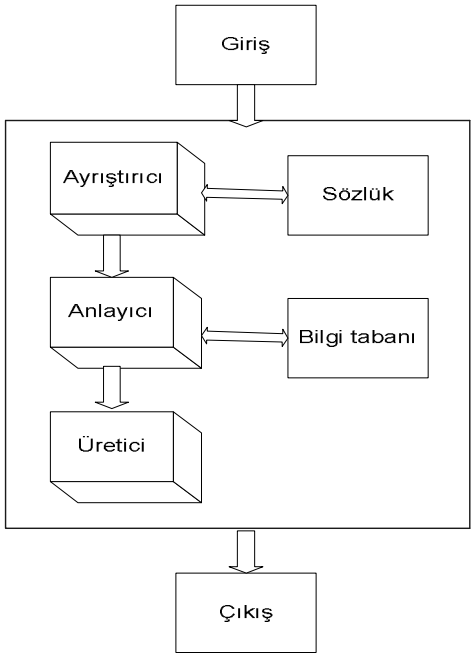
Sözdizimi: Sözcüklerin cümle oluşturmak için ne şekilde sıralanmaları gerektiğini inceler. Bu aşamada analizi tamamlanmış olan sözcükler, dilin daha büyük elemanları olan tümceleri ve metinleri oluşturacak şekilde birleştirilirler.

Anlambilim: Dilin gerçek dünyayla iletişim kurmasını sağlar. Cümle yapısının anlaşılması ve bunun sonucunda eyleme geçilmesi bu aşamada olur. Dilde sözcüklerin dizilişlerinin cümlelere kazandırdığı anlamların incelenmesi ve bu yolla anlam kazandırılması temel işlevdir. Bilgisayarın matematiksel mantığıyla çok bağdaşmayan bu işlem, birçok olasılık göz önüne alınarak matematiksel yaklaşımla çözülmeye çalışılır.

1.2 Doğal Dil İşlemenin Temel Elemanları

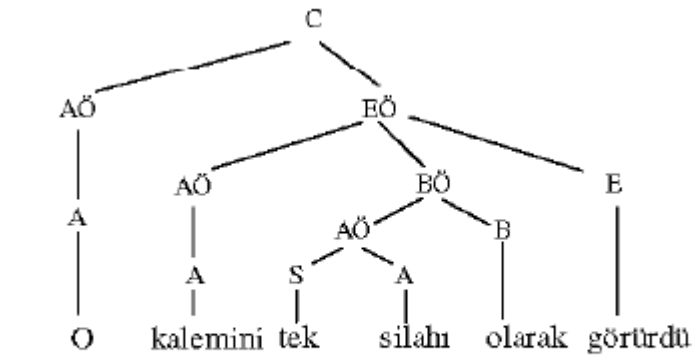
Doğal dil işlemede amaç, bilgisayar ile doğal dilde iletişimin sağlanması olduğu için bilgisayarın doğal dil kurallarını öğrenmesi gerekmektedir. Bunun için bilgisayar genel bir sözlüğe ve bu sözlüğü kullanabilmek için çeşitli algoritmalara ihtiyaç duyar. Bilgisayarın dil ile ilgili genel bilgilerin yanında, ihtiyaç duyduğu ve dilin genel yapısından bağımsız olarak algılanması gereken alana ya da göreve özel bir bilgi tabanına da gereksinimi vardır.

Doğal dil işleme sisteminde genel olarak beş temel eleman bulunur. Bunlar, ayrıştırıcı (parser), sözlük (lexicon), anlayıcı, bilgi tabanı ve üreticidir. Şekil 1.1 de bu beş elemanın birbirleriyle etkileşimi görülmektedir [1].



Şekil 1.1 Doğal dil işleme sistemlerinin genel blok diyagramı [1]

Ayrıştırıcı, doğal dil işlemenin en temel elemanıdır. Ayrıştırıcı verilen cümleyi sözdizimsel olarak analiz eder ve ayrıştırıcı ağacını oluşturur. Ayrıştırma alanında en yaygın tanınan yaklaşımlardan biri, öbek yapısal gramerlerdir. Bu yaklaşım Chomsky'nin üretimsel dönüşümlü dilbilgisi kuramına dayanır. Tümceleri öbeklere bölerek öbeklemeyi hedeflemektedir. Bu yaklaşıma göre dilin temel ve kurucu birimi tümcedir. Tümce, ad öbeği ve eylem öbeği olmak üzere iki temel yapıdan oluşur. Bu öbekler de kendi içlerinde daha küçük öbeklere bölünürler. Şekil 1.1 de örnek bir tümcenin bu yaklaşıma göre gösterimi verilmiştir [2].



C:tümce, AÖ:Ad Öbeği, EÖ: Eylem Öbeği, BÖ: Belirteç Öbeği,
S:Sıfat, A:Ad, E:Eylem, B:Belirteç

Şekil 1.2 Ayrıştırma ağacı örneği [2]

Ayrıştırma işleminden sonra görevleri belirlenen kelimeler, anlamsal analiz işleminden geçirilerek giriş tümcesine göre bir çıkış tümcesi oluşturulur.

Sözlük, program tarafından tanınması istenen tüm sözcükleri içinde barındıran bir yapıdır. Ayrıştırıcı, sözlük ile sözdizimsel analiz yaparak çalışır. Sözlük, her kelimenin doğal dil işleme sistemi tarafından tanınması istenen kök ve anlamlarını içerir [1].

Sözlük üzerinde yapılan işlemler temel olarak dört bölümde incelenebilir:

Jeton seçme (Tokanizasyon): Giriş cümlesi belirli jetonlar kullanılarak bölümlenir. Bu jetonlar genellikle kelimeler ve noktalama işaretleridir. Ancak jetonların seçiminde dilin yapısı dikkate alınmalıdır. Örneğin Türkçe'de boşluk jeton olarak seçilebilirken Japonca'da seçilemez. Çünkü Japonca'da kelimeler arasında boşluk bulunmamaktadır [1].

Köksel analiz: Bu aşamada kelime hecelerine bölünür ve böylece kelimenin köküne ulaşılır. Hecelere bölünerek köksel analiz yapılması Türkçe ve Fince gibi sondan eklemeli dillerde büyük önem taşımaktadır [1].

Sözlüğe bakma: Bulunan kök kelime sözlükte aranır ve ne anlama geldiği bilgisine ulaşılır.

Hata dönüşümü: Aranan kök sözlük içerisinde bulunamazsa bir hata söz konusudur. Bu hatalar genellikle, kelimenin yanlış hecelenmesinden, özel isimlerin doğru tespit edilememesinden ya da kelimenin yazım biçiminden kaynaklanır [1].

Doğal dil işleme sistemlerinde kullanılan sözlüklerdeki kelime sayıları dikkate alınacak olursa, sözlüğün oldukça büyük ve karmaşık bir yapı olduğu ve sözlük oluşturmanın büyük bir yatırım ve zaman gerektirdiği görülür.

Anlayıcı bilgi tabanı ile birlikte cümlenin ne anlama geldiğini tespit etmeye çalışır. *Bilgi tabanı* kavramsal olarak genel bilgi tabanı ve görev bağımlı bilgi tabanı olmak üzere iki alt öğeden oluşur. Anlayıcının temel görevi oluşturulan ayrıştırıcı ağacının bilgi tabanındaki karşılığını bulmaktır. Anlayıcı girilen cümleye uygun cevabı hazırlar [1].

Doğal dil işleme alanında kullanılan en temel *üretici* sistem, belli kelime ve cümleler için depolanmış belli kalıpların kullanıcıya gösterilmesidir.

1.3 Doğal Dil İşlemenin Uygulama Alanları

Doğal dil işlemenin yaygın olarak kullanıldığı uygulama alanları şunlardır:

Çeviri: Bilgisayarların gelişim süreci içerisinde bilgisayarlardan en fazla beklenen işlerden biri de bir dilden diğer dile çeviri yapılmasıydı. Zamanla böyle bir işin doğal dil işleme sistemlerinin kullanılmasıyla gerçeklenebileceği anlaşılmıştır. Bir dilden diğer bil dile bilgisayarla çeviri yapılmasındaki en önemli nokta, verilen cümlenin iyi anlaşılması ve anlam kaybına uğramadan diğer dilde ifade edilmesidir. Montreal Üniversitesi'nde geliştirilen ve TAUM-METEO adı verilen sistem bu tür yazılımlara örnektir. Bu yazılım İngilizceden Fransızcaya resmi raporların çevirisinin yapılmasında kullanılmıştır. Doğruluk oranı oldukça yüksektir. Bu oranın yüksek olmasının nedeni, resmi raporlarda kullanılan dilin ve formatın formalize edilmiş olmasıdır. Bu tür sistemlere bir diğer örnek ise SPANAM adı verilen bir yazılımdır. İspanyolcadan İngilizceye çeviri yapan bu yazılım, anlaşılır bir çeviri yapsa da dilbilgisi yönünden doğru bir çözüm sunamamaktadır; bu nedenle de doğruluk oranı TAUM-METEO ya göre oldukça düşüktür [1].

Görüldüğü gibi, bilgisayarla bir dilden başka bir dile çeviri yapmak henüz tam anlamıyla gerçeklenememiştir. Bunun en büyük nedeni de, doğal dilin yapısının matematiksel mantıkla uyuşmamasıdır. Bu nedenle gerçeklenen çeviri sistemlerinde de kısıtlı dil kullanılmıştır. Kısıtlı dil, kullanılabilecek kelimelerin ve dilbilgisi yapılarının belirlenmesiyle oluşturulur. Dilin bu şekilde kısıtlanarak kullanılması çeviri işlemlerindeki karmaşıklığı azaltırken doğruluğu artırır. Avrupadakiler başta olmak üzere birçok hükümet, bu kısıtlandırılmış dilin uluslararası ticarette kullanılması için kurallar getirmeye başlamışlardır.

Dilbilgisi analizi: Doğal dil işleme sistemlerinin bir diğer uygulama alanı olan dilbilgisi analizi sistemlerinde girilen yazı öncelikle yazım hatalarına karşı denetlenir. Metin içerisinde yanlış kelime ya da deyim kullanılmışsa bunların düzeltilmesi için önerilerde bulunur. Yanlış kullanılmamış kelimelerin yerine daha uygun kelimeler varsa bu kelimeler için de önerilerde bulunur. MS Word ve WordPerfect programları bu sistemlere örnek olarak gösterilebilir. Bazı dilbilgisi analiz sistemleri metnin tüm ayrıştırma ağacını çıkartabilmektedir. Bu tür sistemlere UNIX üzerinde çalıştırılmak üzere Bell laboratuvarlarında geliştirilen Writer's Workbench ile IBM sistemleri üzerinde çalışan EPISTLE örnek olarak verilebilir [1].

Veritabanı Yönetimi: Doğal dil işleme sistemleri, üzerinde büyük ve geniş veri dosyalarının depolandığı ve depolanan bu dosyalar üzerinde çeşitli işlemlerin gerçekleştirildiği veritabanlarının yönetimlerinde de kullanılırlar. Bu kullanımların en ilginçlerinden biri NASA tarafından coğrafik veritabanı üzerinde kullanılan ve LUNAR adı verilen sistemdir. Bu sistem %75-%80 doğruluk oranlarında çalışmaktaydı. Bu çalışmanın bir benzeri yine coğrafik veritabanı üzerinde 1983 yılında geliştirilen CHAT adı verilen sistemdir. Veritabanı yönetiminde kullanılan bir diğer sistem ise IBM işletim sistemleri üzerinde çalışan INTELLECT adı verilen sistemdir. Bu yazılım kullanıcıya veritabanı üzerinde doğal dili kullanarak sorgulama yapıp rapor hazırlamasını sağlar [1].

Belge Yönetimi: Belge yönetim sistemleri çok fazla sayıda belgenin bulunduğu ortamlarda, verilen anahtar kelimelere göre belgeler arasında araştırma yaparak belgelerin özetini hazırlayabilirler. Bu işlem dört aşamada gerçekleşir. İlk aşamada uygun doküman bulunur; ikinci aşamada doküman sınıflandırılır; üçüncü aşamada istenen veri çıkarılır ve son aşamada bu veriden doğal dilde cevap üretilir. SCISOR isimli belge yönetim yazılımı bu tür yazılımlara bir örnektir [1].

Konuşma Tanıma: Doğal dil işlemeyle ilgili bir yapay zeka alanıdır. Yapay zeka yazılımları bilgisayar sistemine sesli giriş yapabilecek şekilde tasarlanabilirler. Sesli giriş sonrasında elde edilen girdi doğal dil işleme sisteminde kullanılabilir ve sistemin vereceği cevap sesli şekilde kullanıcıya iletilebilir. Örneğin, teknolojinin bugün gelmiş olduğu noktada ABD, Almanya ve Japonya'daki araştırmacılar telefon ile konuşan iki kişinin konuşmalarını anında tanıyıp karşısındaki kişinin diline çeviren, onun anlayabileceği konuşmayı üreten sistemlerin prototiplerini gösterebilmişlerdir. Ancak bu gibi sistemlerin günlük hayatta yaygın olarak kullanılması için aradan bir süre daha geçmesi gerekecektir [3].

Doğal dil işleme sistemlerinin hayatımıza girmesi hiç şüphesiz yaşamımız oldukça kolaylaştıracaktır. Ancak, doğal dil işleme sistemlerinin eklendiği yazılımların daha

karmaşık, daha maliyetli oldukları ve daha fazla belleğe ihtiyaç duydukları da bilinen bir gerçektir.

1.4 Türkçe Dili ve Biçim Birimsel İncelenmesi

Türkçe dili Ural-Altay dil ailesinin bir üyesidir. Yapısal olarak Türkçe'nin doğal dil işlemede ilginç sorunlar içeren özellikleri vardır. Sözcük yapısı ve üretimi açısından Türkçe bitişken bir dildir. Bu açıdan Türkçe, Fince ve Macarca'ya benzemektedir. Bu gibi dillerde sözcükler bir kök sözcüğe eklenen biçibirimlerden oluşurlar. Bu biçibirimler eklendikleri kök veya gövdenin anlamını, sözcük türünü veya sözdizimsel işlevini değiştirebilirler. Bu nedenle, Türkçe'de bu şekilde kurulan bir sözcük ile ifade edilen kavram, bazen başka bir dilde ancak bir cümle ile ifade edilebilir [3].

Türkçe'de bir sözcüğün ekler yardımı ile dönüştürülebileceği farklı sözcük sayısı kuramsal olarak sonsuzdur. Bu duruma abartılı bir örnek olarak "Osmanlılaştıramadıklarımızdanmışsınızcasına" kelimesi verilebilir [2].

Türkçe'nin hemen her dilbilimi ders kitabında incelenmesinin nedeni, dilin bir takım dilbilgisel olgularda tipik bir örnek oluşturmasıdır; örneğin ses uyumu, bitişken sözcük yapısı, sözdizimindeki serbestlik ve öbek yapılarında tamlayanların her zaman tamlanandan önce gelmesi gibi [3].

Türkçe'nin sözcük yapısı, köklere yapım ve çekim eklerinin sonek olarak eklenmesine dayanır. Çekim soneklerinin bolluğu, yapım soneklerinin çok üretken olması ve soneklerin sözdizime olan doğrudan etkisi bilgisayarlı biçim birim çözümlemesinde ilginç sonuçlar ortaya çıkarmaktadır. Örneğin, çiçekleri kelimesi için olası çözümleme sonuçları Tablo 1.1 deki gibidir:

Tablo 1.1 "çiçekleri" kelimesi için biçim birimsel çözümleme sonuçları

Çözümleme:	**Örnek kullanım:**
Kök + çoğul + 3.tekil-iyelik	Bunlar Mehmet'in çiçekleri.
Kök + 3.çoğul-iyelik	Arkadaşlarımın çiçeklerini sulamalıyım.
Kök + çoğul + durum	Ayşe çiçekleri suladı.

Çözümlemedeki bu gibi belirsizlikler bazen öbek ve cümle düzeyinde incelemeyle giderilebilir. Örneğin, "Mehmetlerin çiçekleri soldu" cümlesinde çiçekleri sözcüğü öznedeki iyelik sonekinden dolayı kök + 3.çoğul-iyelik olarak çözümlenir. Sözdizim düzeyinde inceleme her zaman belirsizlikleri gideremeyebilir. Örneğin "çiçekleri solmuş" cümlesi "onun çiçekleri solmuş" ya da "onların çiçekleri solmuş" olarak anlaşılabilir [3].

Biçim birimsel çözümlemedeki belirsizlikler sözcük türlerinin bulunmasında da sorunlar çıkarabilir. "gider" kelimesi için olası sözcük türleri Tablo 1.2 de görülmektedir [3].

Tablo 1.2 'gider' kelimesi için sözcük türü sonuçları

Çözümleme:	**Örnek kullanım:**
Ad	Bu gider tablosu çok detaylı hazırlanmış.
Eylem + zaman	Ayşe her gün okula gider.

Çekim soneklerinin varlığı durumunda bu belirsizlik giderilebilir; örneğin "giderlerim" sözcüğünde kök adıl olmak durumundadır, çünkü ad çekimi almıştır.

Örneklerden de görüleceği gibi Türkçe, ayrıştırma alanında İngilizce gibi üzerinde yoğun olarak çalışılmış birçok dilden farklı özellikler taşımaktadır. Bu niteliği ile benzer özellikler gösteren bir sınıf dilin temsilcisi olarak görülebilir. Bu dillere örnek olarak diğer Türki diller, Fince, Macarca, Japonca ve Korece gösterilebilir [3].

Günümüzde doğal dil işleme alanındaki çalışmaların büyük bir bölümü, bilim ve iş alanında her yerde geçerli bir dil olması açısından İngilizce üzerinde yapılmaktadır. Türkçe dili üzerinde doğal dil işleme alanında teorik olarak çalışmalar yapılmış olsa da pratikte yapılmış olan çalışmaların sayısı oldukça sınırlı kalmaktadır. Bu nedenle Türkçe'de araştırma alt yapısı oluşturmak için daha çok çalışma yapılması gerekmektedir.

1.5 Benzer Çalışmalar

Bu bölümde ilk olarak yazım denetimi konusunda Türkçe dışındaki dillerde yapılmış olan çalışmalar ve gerçeklenen yazılımlarla ilgili bilgilendirme yapılmıştır. Sonrasında ise Türkçe üzerine yapılan çalışmalar incelenmiştir.

Yazım denetimi konusunda ana bilgisayarlar (mainframe) üzerinde gerçeklenen ilk çalışmalar 1970'li yılların sonuna rastlar. Kişisel bilgisayarlar için geliştirilen yazım denetleme yazılımları ise 1980 yılında ortaya çıkmıştır ve bunu 1981 yılında IBM PC nin bu alanda oluşturduğu yazılım paketleri izlemiştir. Bu dönemde Maria-Mariani, Soft-Art, Microlytics, Proximity, Circle Noetics ve Reference Software gibi geliştiricilerin ürettiği yazılımlar son kullanıcı ürünleri olarak hızla yazılım pazarında yerini almıştır. Ancak bu yazılımlar pazarda kısa süreli olarak yer almışlardır. 1980'li yılların ortalarında WordStar ve WordPerfect gibi geliştiriciler kendi üretmiş oldukları yazılım paketlerinin içine yukarıda adları verilen üreticilerin geliştirdikleri yazım denetimi paketlerini de eklemişler ve başta İngilizce olmak üzere Avrupa dillerinin çoğunu kapsayacak şekilde yazım denetimi konusunda ilerleme kaydedilmesini sağlamışlardır. Ancak Macarca ve Fince gibi yapısı Avrupa dillerinden farklı olan dillerde bu anlamda çok fazla gelişme sağlanamamıştır.

İlk oluşturulan yazım denetimi yazılımlarında sadece yanlış yazılmış olan sözcükler bulunuyordu; ancak bu sözcükler için öneri yapılmıyordu. Günümüzde ise geliştirilen yazılımların çoğunda yanlış yazılmış sözcükler için öneri yapılmaktadır. Hatta bazı yazılımlar dilbilgisi hatalarının bir kısmını da yakalayabilmektedir.

Yazım denetimi konusunda geliştirilen yazılımların çoğu başta İngilizce olmak üzere Avrupa dillerinde gerçeklenmiştir. Bu dillerde sözcükler ya çok az sayıda ek almakta ya da hiç ek almamaktadır. Örneğin; İngilizcede sözcüklerin çok büyük bir kısmı cümlede yazıldıkları şekilde sözlükte yer almaktadır; dolayısıyla sözcüklerin biçim birimsel çözümlemesi oldukça basite indirgenmiş olmaktadır. Oysaki Türkçe'de sözcükler çok sayıda ek alabildikleri için ve ekler sözcüklerin sonuna ulandıkları için biçim birimsel çözümlemenin yapılması oldukça karmaşık bir iş halini almaktadır. Bu nedenle İngilizce ve benzeri dillerde geliştirilen yazım denetimi araçlarının sayısı oldukça fazla olmasına rağmen; Türkçe için birkaç araç dışında örnek bulmak mümkün değildir. Çünkü Türkçe çok farklı kurallara ve sorunlara sahip bir dildir. Eklerin yanlış sıralarla köke ulanması ya da sesli uyumuna uygun olmayan ekin köke ulanması bu gibi durumlara örnek olarak verilebilir.

Alman dili için Gabriele Kodydek tarafından yapılmış olan sözcük çözümleme çalışmasında Alman dilindeki en küçük anlamlı parçalar atom olarak nitelendirilmiştir. Gerçeklenen sözcük çözümleme sistemi iki ana kısımdan oluşturulmuştur. Bunlar atom tablosu ve özyineleyici çözülme (recursive decomposition algorithm) algoritmasıdır. Yaklaşık olarak 6000 atom içeren atom tablosu Almanca'daki tüm sözcükleri içermektedir. Özyineleyici çözülme algoritması Alman dilinin kurallarına uygun olarak sözcükleri atomlarına ayırır. Bu

çalışmada atomlar işlevlerine göre önek, gövde ya da ek olarak sınıflandırılmışlardır. Her atom kendi sınıfına ait davranışa göre atom tablosuna yerleştirilir. Örneğin; "end" gövde olarak kullanılabilir ve "enden" halini alır ya da ek olarak kullanılarak "gehend" halini alır. Özyineleyici çözülme algoritması, girilen sözcüğün atom tablosundaki tüm alt sözcüklerini (substring) bulmaya çalışır ve Alman dilinin dilbilgisi kurallarına göre bulunan tüm atomları birleştirir. Bu sözcük çözümleme çalışmasında gerçekleştirilen yazım denetimi kısıtlıdır. Eğer girilen sözcük, sözcük çözümleme düzeneğine göre atomlarına ayrılamıyorsa ya yanlış yazılmış demektir ya da atom tablosunda bulunmayan atomları içermektedir. Örneğin; girilen isim yabancı sözcük ise ya da coğrafik bir isim ise atom tablosunda bulunmaz. Bu durumda atom, atom tablosuna eklenebilir. Gerçeklenen bu sistem tüm yazım hatalarını bulamamaktadır; çünkü sistem bazı durumlarda yanlış yazılmış olan sözcükler için (sözcükler anlamsız olmasına rağmen) çözümleme yapabilmektedir [4].

Fransızca için geliştirilen yazım denetimi aracı olan FipsOrtho, Fransızca öğrenenlere yönelik olarak geliştirilmiş olan bir yazılımdır. Bu yazılımda yanlış yazılmış olan sözcükleri bulmak ve düzeltmek için kullanılan yöntemler alpha-code yöntemi, ses bilimsel inceleme yöntemi ve ad-hoc kuralları yöntemidir. Alpha-code yönteminde sözcükte bulunan harfler vurgularına göre tekrar sıralanırlar ve tekrarlanan harfler bire indirgenir. Aynı alpha-code çözümlemesine sahip olan sözcükler benzerdir. Hatalı olan sözcük için alpha-code çözümlemesi yapılır ve bu alpha-code çözümlemesine uygun olan diğer sözcükler bulunur. Bulunan sözcükler arasında Levenshtein-Damerau edit distance hesaplaması yapılır ve uzaklığı en küçük olan sözcük ya da sözcükler doğru sözcük olarak önerilir. Örneğin; "ennumération" sözcüğün Fransızcadaki doğru yazılımı "énumération" şeklindedir. Bu sözcük için alpha-code "mnrtaeiou" şeklindedir ve bu alpha-code çözümlemesine sahip olan diğer sözcükler içinden aranan sözcüğe en yakın uzaklığa sahip olan sözcük seçilir. Ses bilimsel inceleme yönteminde girilen sözcüğü söyleyiş şekline dönüştüren bir uzman (expert) sistem kullanılmıştır. Sözlükte bu söyleniş şekline sahip olan sözcükler arasında arama yapılır ve bulunan sonuçlar öneri olarak sunulur. Örneğin; girilen sözcük "puit" ise öncelikle bu sözcüğün söyleniş biçimi bulunur ve bu da "/pyi/" şeklindedir. Bu söyleniş biçimine uygun olan "puis" ve "puits" sözcükleri öneri olarak sunulur. Üçüncü yöntem olan ad-hoc kuralı ise biçim birimsel hataya sahip olan sözcüklerin bulunmasında kullanılır; ancak kullanım alanı oldukça sınırlıdır. Bu yöntemde sözcük sonlarına gelen eklerin bazılarının doğru yazılıp yazılmadığı denetlenir. Örneğin; "-ails" ve "-als" eklerinin yerine çoğul eki olarak "-aux" eki getirilir. "générals" sözcüğü için üretilen öneri "généraux" şeklindedir [5].

Hascheck(Hrvatski akademski spelling checker) Hırvat dili için yazılan ilk yazım denetimi aracıdır. Hascheck öğrenme algoritmasına dayalı bir yapıya sahiptir. Yanlış yazılmış olan sözcüklerin değerlendirilmesinde bulanık mantık ve yapay zeka yöntemlerini (n-gram) kullanır. Bu çalışma kapsamında kullanılan sözlük İngilizce-Hırvatça derleminden türetilmiştir ve yaklaşık bir milyon sözcük içermektedir [6,7].

Hindistanda geliştirilen yazım denetimi aracı olan tamil yazım denetleyicisi Tamil dili için geliştirilmiştir. Bu dil Hindistan'ın güneyinde kullanılan bir dildir ve biçim birimsel açıdan zengin bir yapıya sahiptir. Gerçeklenen bu yazılımda girilen metindeki sözcükler sırayla sözlükte aranır. Sözlükte bulunamayan sözcük için hata düzeltme yöntemi uygulanır. Geliştirilen yazım denetleyicisi, sözcük çözümleme, sözcüklerin yazım denetimini yapma ve yanlış yazılmış sözcükler için öneride bulunma işlemlerini gerçekleştirmektedir. Girilen metindeki her sözcük biçim birimsel çözümleme ile parçalara ayrılır. Parçalara ayrılamayan sözcükler yanlış yazılmış demektir ve bu sözcükler için hata düzeltme işlemi yapılır. Eğer sözcük doğru bir şekilde kök ve eklerine ayrılabiliyorsa, kök sözcük sözlükte aranır; sözlükte bulunamayan sözcükler için o sözcüğe en yakın olan sözcükler önerilir [8].

Türkçe için geliştirilen az sayıdaki yazım denetimi araçlarından biri Ayşin Solak ve Kemal Oflazer tarafından gerçeklenmiştir [9,10]. Gerçeklenen bu yazılımın kullanmış olduğu sözlük 23.000 sözcük içermektedir. Sözlükteki her bir kök sözcük sözcüğe ait özellikleri simgeleyen bayraklarla işaretlenmiştir. Sözlükteki herhangi bir sözcük 64 farklı bayrak ile işaretlenebilir. Ancak geliştirilen yazılımda bu bayrakların 41 tanesi kullanılmıştır. Kök bulma algoritmasının temeli sözlükte arama yapma işlemine dayanmaktadır. Bu algoritmaya göre sözcük sözlükte aranır ve bulunursa hiç ek almamış demektir; dolayısıyla çözümlemeye gerek yoktur. Sözcüğün sözlükte bulunamaması durumunda sözcüğün sağından bir harf çıkarılır ve sözcüğün kalan kısmı sözlükte aranır. Bu işlem sözcük sözlükte bulunana kadar devam eder. Eğer sözcüğün ilk harfine gelinmişse; ancak kök sözlükte bulunamamışsa sözcük yanlış yazılmış demektir.

Harf çıkarılarak sözlükte bulunan en uzun sözcük her zaman aranan kök olmayabilir. Bu durumda en son sözlükte bulunan kök üzerinden devam edilerek yeni kök sözlükte aranır. Örneğin; "yapıldın" sözcüğü için ilk olarak "yapı" isim soylu kökü bulunur. Ancak sözcüğün kalanına bakıldığında eklerin isim soylu sözcüklere eklenebilecek ekler olmadığı görülür ve dolayısıyla "yapı" sözcüğünden bir harf daha çıkarılarak sözlükte arama yapılır ve "yap" eylem soylu kökü bulunur [9,10].

Kök bulma işlemi kök sözcük değişime uğradığı durumlarda sorunlara yol açmaktadır. Örneğin, "oğlumuz" sözcüğü için kök "oğul" dur ve köke birinci çoğul

şahıs eki ulandığında kök sözcük değişime uğrayarak "oğl" haline dönüşmektedir. Ancak "oğl" sözlükte bulunamayacağı için burada bir denetleme işlemi yapılır. Son iki harf sessiz ve ilk harf sesli olduğu için kök sözcüğün değişime uğramış olma olasılığı yüksektir. "ğ" ile "l" harfleri arasına sesli uyumuna uyacak şekilde "ı" ve "u" harfleri getirilerek yeni aday kökler oluşturulur ve sözlükte aranır. "oğul" kökü sözlükte bulunur ve bu kökün sesli düşmesi bayrağına sahip olup olmadığına bakılır. Kök sesli düşmesi bayrağına sahip olduğu için aranan kök bulunmuş demektir.

Kök sözcüğün değişmesi durumu için bir başka örnek ise sessiz yumuşaması durumunun oluşmasıdır. Örneğin; "tabağım" sözcüğü için sözlükte aranan kök "tabağ" olacaktır; ancak bu sözcük sözlükte bulunmadığı için burada da bir denetleme yapmak gereklidir. Ek sesli harf ile başladığı için ve son harf (ğ) b, c, d, g, ğ harflerinden biri olduğu için sessiz yumuşaması durumunun olması söz konusu olabilir. Sözcük "-loğ" eki ile bitmediği için, son harften bir önceki harf "n" harfi olmadığı için "ğ" harfinin yerine "k" harfi konur ve "tabak" sözcüğü sözlükte bulunur. Sözcük sessiz yumuşaması bayrağına sahip olduğundan dolayı aranan kök bulunmuş demektir [9,10].

Oflazer ve Solak'ın bu çalışmalarında sesli uyumu ile ilgili denetimler de yer almaktadır. Ayrıca ekler sınıflandırılmış ve bu sayede çözümlemenin daha kolay hale gelmesi sağlanmıştır. Diğer yazım denetimi araçlarında olduğu gibi, geliştirilen bu yazılımda da sözcük bazında denetleme yapılmış; cümle bazında denetleme yapılmamıştır. Ayrıca yanlış yazılmış olan sözcükler için öneri yapma sistemi mevcut değildir [9,10].

Türkçe'de yazım hatalarının düzeltilmesi ile ilgili bir çalışma Kemal Oflazer ve Cemalettin Güzey tarafından yapılmıştır. Bu çalışma iki seviyeli bir biçimbirimsel incelemeye ve dinamik programlama tabanlı bir arama algoritmasına dayanmaktadır. Yazım hatalarının düzeltilmesi ile ilgili olarak yapılan bu çalışmada, sorun iki ayrı ana başlık altında değerlendirilmiştir. İlk olarak yapılan işlem, yanlış yazılmış sözcük için sözlükten tüm aday köklerin bulunmasıdır. Burada zor olan nokta, köklerin yanlış yazılma sonucunda mı yoksa ses değişimlerine uğrayarak mı değişmiş olduğuna karar vermektir. İkinci adım ise sorunun esas kısmını oluşturmaktadır; bu da bulunan tüm aday köklerden olası tüm sözcüklerin türetilmesidir. Bunu gerçeklemek için de edit distance metric olarak bilinen iki sözcük arasındaki uzaklığı bulma yolundan yararlanılmıştır. Bu çalışmada iki sözcüğün birbirine ne kadar benzer olduğunu bulmak için "q gram" yöntemi kullanılmıştır. Yapılan sınama sonuçlarına göre yanlış yazılmış olan bir sözcüğün düzeltilmesi %95 başarıyla gerçekleştirilmektedir [11,12].

Türkçe'de yanlış yazılmış sözcüklerin bulunması ile ilgili bir başka çalışma Rıfat Aşliyan, Korhan Günel ve Tatyana Yakhno tarafından gerçekleştirilmiştir. Bu çalışmanın amacı girilen bir Türkçe metindeki sözcüklerin doğru yazılıp yazılmadığına karar vermektir. Sistem girilen Türkçe metindeki sözcükleri giriş olarak almakta ve "n-gram frequency" yöntemini kullanarak olasılık dağılımı hesabı yapmaktadır. Eğer bir sözcüğün olasılık dağılımı sıfır ise o sözcüğün yanlış yazılmış olduğuna karar verilir. Sistemi sınamak için aynı sözcükleri içeren metinlerden oluşan iki farklı veritabanı hazırlanmıştır. Bunlardan biri 685 tane yanlış yazılmış; diğeri ise 685 tane doğru yazılmış sözcük içermektedir. Sistemin yanlış yazılmış olan sözcükleri bulma başarımı %97 olarak verilmiştir [13].

1.6 Tezin Amacı, Kapsamı ve Katkısı

Yazım hatalarının denetlenmesi ve düzeltilmesi konusunda yapılmış çalışmalara bakıldığında, bu çalışmaların çoğunun Hint-Avrupa dil ailesi üzerindeki dillerde özellikle de İngilizce üzerinde yapıldığı görülmektedir. Ancak, Ural-Altay dil ailesi üyesi olan Türkçe üzerinde yapılan çalışmaların da sayısı giderek artmaktadır. Hint-Avrupa dillerinin yapısı ile Ural-Altay dillerinin yapısı biririnden farklı olduğu için İngilizce için ya da diğer Hint-Avrupa dilleri için yapılmış olan çalışmalar Türkçe için anlam ifade etmemektedir.

Bu tez çalışmasının amacı, Türkçe metinlerdeki yazım hatalarının bulunmasını ve bulunan hataların en yüksek başarımla düzeltilmesini gerçeklemektir. Bu amaçla;

- Türkçe metinlerdeki yazım hatalarının denetlenmesi,
- Türkçe metinlerdeki yabancı sözcüklerin bulunması,
- Bulunan yabancı sözcükler için Türkçe karşılıklarının önerilmesi,
- Yanlış yazılmış sözcükler için sözcük önerilmesi,
- Bulunan yazım hatalarının düzeltilmesi üzerinde çalışılmıştır.

Türkçe metinlerdeki yazım hatalarının denetlenmesi ve yabancı sözcüklerin bulunması için yapılan çalışmalarda sözcüklerin Türkçe ses bilgisi kurallarına uygun olup olmadığının sınaması yapılmıştır. Bu sınamanın yapılabilmesi için ilk olarak heceleme algoritması yardımıyla sözcüklerin hecelenebilir olup olmadığının denetimini yapmak gereklidir. Bu denetimi geçemeyen sözcükler Türkçe hece yapısına uygun olmadıkları için doğrudan elenirler.

Türkçe metinlerdeki yabancı sözcüklerin bulunması için sözcüklerin Türkçe ses bilgisi ve hece yapısı kurallarına uygunluklarının denetlenmesi yeterlidir. Ancak yazım hatalarının bulunabilmesi için biçim birimsel çözümlemeye ihtiyaç duyulmaktadır. Bu nedenle tez kapsamında gerçeklenen biçim birimsel çözümleme sayesinde sözcükler kök ve eklerine ayrılarak yazım hataları üzerinde denetleme yapılması gerçekleştirilmiştir. Yazım hatasının kök sözcükten mi yoksa kök sözcüğe ulanan eklerden mi olduğunu anlamak için biçim birimsel çözümlemeye ihtiyaç vardır. Biçim birimsel çözümlemesi yapılan sözcükte hata bulunması durumunda hatalı olan parçanın (kök ya da ek) en yüksek başarımla düzeltilmesi üzerinde çalışılmıştır. Bu alanda yapılan çalışmalar için bilinen algoritmaların yanısıra açık kaynak kodlu yazılım ve araçlardan da faydalanılmıştır.

Tez kapsamında yapılan iş biçim birimsel olarak yanlış yazılmış olan sözcüklerin bulunması ve düzeltilmesidir. Bundan sonra yapılması gereken ise girilen Türkçe metinlerdeki anlamsal hataların bulunup bulunmadığını denetlemektir. Ancak bunun için sözcüklerin tek başına biçim birimsel çözümlemesini yapmak yeterli olmayacaktır; cümlelerin anlamsal açıdan çözümlenmesi de gerekecektir.

1.7 Tezin Bölümleri

Toplam altı bölümden oluşan tezin içeriği aşağıda açıklanmıştır:

- *Bölüm 2 - Türkçe' nin Ses Bilgisi Yapısı*

  Bu bölümde Türk dilinin tez konusuyla ilgili olan özellikleri anlatılmıştır. Dolayısıyla bu bölüm Türkçe'nin ses bilgisi ile ilgili yapısı ve kurallarına ayrılmıştır. Bu konuda [14] ve [15] kaynaklarının yanısıra TDK web sayfası da kaynak olarak kullanılmıştır.

- *Bölüm 3 - Geliştirilen Yazılım ve Algoritmalar*

  Bu bölümde tez kapsamında geliştirilen algoritmaların ve kullanılan yöntemlerin tanıtımı yapılmıştır. Bölüm 3.1 tez kapsamında kullanılan sözlüğün yapısıyla ilgili bilgi içermektedir. Bu bölümde sözlüğün içindeki sözcüklerin etiketlenmeden sözlük oluşturulup oluşturulamayacağı ile ilgili olarak yapılan çalışmalara da değinilmiştir. Bölüm 3.2 heceleme algoritmasını kapsamaktadır. Bölüm 3.3'de sözcük köklerinin bulunması ile ilgili olarak kullanılan yöntem ve algoritmaların tanıtımı yapılmıştır. Bölüm 3.4'de sözcüğün biçim birimsel çözümlemesinin yapılmasında izlenen yol anlatılmıştır. Bölüm 3.5'de Türkçe denetimi yapılması için kullanılan ses

bilgisi kuralları için geliştirilen algoritmaların tanıtımı yapılmıştır. Bölüm 3.6 ses bilgisi açısından sözcük denetimine ayrılmıştır. Son olarak Bölüm 3.7'de sözcük önerme işleminin hangi durumlarda yapıldığı ve bu işlem için hangi yöntemin izlendiği anlatılmıştır.

- *Bölüm 4 – Tezin Başarımı ve Sınama Sonuçları*

  Bu bölümde geliştirilen yazılımın başarımı sınanmış; sınama ölçütleri ve sonuçları verilmiştir. Ayrıca sınama yöntemi ile ilgili detaylı bilgiye ve sınamanın hangi sınama verileriyle yapıldığına da yer verilmiştir.

- *Bölüm 5 – Sonuç*

  Bu bölümde tez çalışmasının sonucuyla ilgili özet bilgi verilmiştir. Ayrıca gelecek araştırmalar için öneriler sunulmuştur.

2. TÜRKÇE'NİN SES BİLGİSİ YAPISI

Bütün dillerin dayandığı temel unsur sestir. Diller, insanların birbirleriyle anlaşmak için çıkardıkları seslerin birleşmesinden oluşmuştur. Bu nedenle ses bilgisi, dilbilimin asıl çalışma alanını oluşturur.

Ses bilgisi (fonetik), bir dilin seslerini, ses değişmelerini (birleşme, türeme, düşme vb.) ve bütün bu değişmelerin tarih boyunca geçtiği evreleri inceler. Bir dilin ses özelliklerinin belli kalıplara ve kurallara oturtulması ses bilgisi ile gerçekleştirmiştir.

Ses genel anlamda, kulağımızla duyduğumuz dış izlenimlerin tümüne verilen addır. Ancak burada ses bilgisinin konusu olan ve dile malzeme teşkil eden ses; insan sesidir. Onu doğadaki diğer seslerden ayırmak gerekir.

İnsan sesi, vücut içinden üflenerek ağıza, dudaklara kadar gelip konuşma şekline dönüşürken uzun bir yol katetmekte, değişik boşluklardan geçmektedir. Bu yol ve boşluklardaki pek çok etki ile farklı seslerin çıkması sağlanmakta ve konuşma gerçekleşmektedir.

En küçük dil birliklerinden en büyük birlik olan cümleye kadar dili oluşturan tüm şekillerde ses denilen unsurlar bulunur. Dolayısıyla ses dilin malzemesine verilen addır.

Biçim bilgisi bir dilin sözcük yapılarını, türeme yollarını, köklerini, eklerini, çekim biçimlerini inceleyen dilbilgisi koludur. Her sözcüğün bir ses yapısı ve bir biçimi vardır.

2.1 Türkçe'nin Ses Özellikleri

Türkçe dilinin ses açısından temel özellikleri şöyle sıralanabilir:

Türkçe'de sesli harfler diğer dillere oranla daha çok kullanılır. 1972 yılında Ankara'da düzenlenen I. Türk Dili Bilimsel Kurultayı'nda bu konuda yapılmış bir çalışmaya ilişkin bildiri sunulmuştur. Nevzat M. Yusuf tarafından gerçekleştirilen bu

çalışma, "Çağdaş Türk Dilinin Fonolojisi Üzerine İstatistik Notlar" adı altında Türkçe'deki seslerin kullanım sıklığıyla ilgili istatistiksel sonuçlara ulaşır.

En seçkin Türk roman ve öykülerinden 21 tanesi ve o tarihlerde çok revaçta olan süreli yayınların 7 tanesi üzerinde, hem düz yazı metinler hem de şiirlerden seçilen metinlerle bu araştırma yapılmıştır. Bu araştırmanın sonucu, Tablo 1.3 de görülmektedir. Harflerin kullanım oranı soldan sağa doğru azalmaktadır.

Tablo 2.1 Harflerin Türkçe'deki kullanım sıklıkları

En sık rastlanan sesli harfler	a, i, e, ı
Orta sıklıkta rastlanan sesli harfler	u, o ,ü
En az rastlanan sesli harfler	ö
En sık rastlanan sessiz harfler	r, n, l, k ,d, y
Orta sıklıkta rastlanan sessiz harfler	m, t, s, ş, z, h
En az rastlanan sessiz harfler	g, t, ğ, v, c, p, f, j

Bu sonuçlardan tüm sesli ve sessiz harflerin kullanımı ile ilgili olarak bulunan oran ise sesli harfler için % 43,36, sessiz harfler için %56,64 tür. Buradan hareketle Türkçe'deki sesli harflerin kullanılma oranını oldukça yüksek olduğu söylenebilir. Diğer dillerde yapılan araştırmalarda bulunan değerler Tablo 1.4 de görülmektedir.

Tablo 2.2 Dillerin sesli ve sessiz harf kullanım oranları

DİLLER	SESLİ HARFLER	SESSİZ HARFLER
Romence	49,4	49,74
İtalyanca	47,74	52,27
Fransızca	43,42	56,58
Türkçe	43,36	56,64
Macarca	41,91	58,09
İsveççe	41	59
Çekçe	40,73	59,13
Almanca	38,86	61,14
İngilizce	37,4	62,6

Türkçe'yi diğer dillerden ayıran bir başka belirgin özellik de sesli uyumu olarak adlandırılan ses eğilimleridir. Altay dillerinin ve bazı Ural dillerinin tipik bir özelliği olan sesli uyumu, Türkçe'de çok eski zamanlardan beri varlığını sürdürmüştür. Hatta günümüz Türkçesinde sesli uyumuna uymayan bazı sözcüklerin bile eskiden bu uyumu sağladıkları bilinmektedir. Örneğin kardeş sözcüğü sesli uyumuna uymazken,

bu sözcüğün eski hali olan karındaş sözcüğü sesli uyumuna uymaktadır. O halde sesli uyumuna uymayan sözcükler ya zaman içinde değişime uğramıştır ya da dilimize yabancı dillerden girmişlerdir.

Sessiz uyumu da Türkçe'nin önemli özelliklerinden biridir. Sessiz uyumunda kök ve gövdelere getirilen eklerdeki sessiz harflerin, kök ve gövdedeki sessiz harflerle biçimi, çıkış yerleri, söylenişte ağzın aldığı şekiller bakımından birbiriyle uygunluğu söz konusudur.

Ses açısından bir başka önemli özellik de sözcük başında birden çok sessiz harfin yan yana bulunamayışıdır. Bu özellik tüm Altay dillerinde vardır. Türkçe'ye yabancı dilden girmiş olan sözcükler de bu sorunu çözmek üzere iki sessiz harf arasına bir sesli harf konur. Yazı diline bu şekilde geçmemiş olsa da kelimelerin söylenişi esnasında mutlaka sesli harf konulur. Örneğin stadyum sıtadyum, tren tiren şeklinde söylenir.

Türkçe sözcük birleşmelerine de yatkın bir dildir. Bu durum tek bir sözcüğün yetmediği durumlarda birden çok sözcüğün bir kavramı anlatmak üzere bir araya gelmesiyle ortaya çıkar. Örneğin dil ve bilim kelimeleri birleşerek dilbilim sözcüğünü oluşturmuştur.

2.2 Türkçe'de Sesler

Türkçe'de sesler sesliler ve sessizler olmak üzere ikiye ayrılırlar. Sesliler alt çenenin ve dudakların aldığı duruma göre biçimlenip çıkan seslerdir. Sesliler ses yolunda hiçbir engele uğramadan çıkan ve ses tellerini titreştiren seslerdir.

Sessizler belirli boğumlanma noktası bulunan seslerdir. Ses tellerini titreştirmezler. Sessizlerin başlı başına sesleri yoktur; yanında bir sesli bulunmak zorundadır.

2.2.1 Sesli Harfler

Sesli harfler ses yolunda hiçbir engele uğramadan ses organlarının ortaklaşa işlemesiyle oluşan seslerdir. Türkçe'de sekiz tane sesli harf bulunmaktadır.

Türkçe'de sesliler dilin, alt çenenin ve dudakların aldıkları duruma göre üçe ayrılırlar. Dilin aldığı duruma göre kalın ya da ince, alt çenenin aldığı duruma göre dar ya da geniş, dudakların aldığı duruma göre ise yuvarlak ya da düz olurlar. Tablo 2.3 de Türkçe'deki sesli harflerin sınıflandırılması görülmektedir.

Tablo 2.3 Türkçe'deki sesli harflerin sınıflandırılması

	Düz		Yuvarlak	
	Geniş	Dar	Geniş	Dar
Kalın	**a**	**ı**	**o**	**u**
İnce	**e**	**i**	**ö**	**ü**

2.2.2 Sessiz Harfler

Sessiz harfler meydana gelişleri sırasında ses yolunda bazı engellerin etkisi altında kalırlar. Ses organları daralma, gevşeme veya kapanma hareketleriyle bu seslerin şekillenmesini sağlar. Türkçe'de yirmi bir tane sessiz harf bulunmaktadır.

Sessiz harfler ses tellerinin titreşmesine göre sert ya da yumuşak, ses yolunun tıkalı ya da açık oluşuna göre sürekli ya da süreksiz, sesin çıktığı yere göre dudak, damak, diş ya da gırtlak sessizleri olarak ayrılırlar. Tablo 2.4 de Türkçe'deki sessiz harflerin sınıflandırılması görülmektedir.

Tablo 2.4 Türkçe'deki sessiz harflerin sınıflandırılması

	Sert		Yumuşak	
	Sürekli	Süreksiz	Sürekli	Süreksiz
Dudak	**f**	**p**	**m, v**	**b**
Diş	**s, ş**	**ç, t**	**j, l, n, r, z**	**c, d**
Damak		**k**	**ğ, y**	**g**
Gırtlak	**h**			

2.3 Türkçe'de Ses Olayları

Türkçe'de tüm kurallar seslere ilişkin organların hareketlerine uygun yapıdadır, herhangi bir zorlanma durumunda, sözcük içindeki sesler değişime uğrar. Bu da Türkçe'nin seslerinin kurallarını izlemeyi kolaylaştırmaktadır. Sessizlerin yanyana gelme, eklerin ulanma ve yabancı sözcükleri uyarlamada bu kurallara sıkıca uyulur. Bu araştırmada bu durum dikkate değer görülmüş ve seslerin kaynakları ile sözcüklerde bulundukları yerler arasında bağlar bulunduğu belirlenmiştir.

2.3.1 Türkçe'de Bulunan ve Bulunmayan Sesler

Türkçe bir sözcüğün sonunda süreksiz yumuşak sessiz (*b, c, d, g*) bulunmaz, böyle sesler süreksiz sert sessizlere (*p, ç, t, k*) dönüşür. "ilac" değil "ilaç", "kitab" değil "kitap" kullanılır.

Türkçe bir sözcüğün başında "c, ğ, l, m, n, r, v, z" sesleri bulunmaz.

- c sesi Türkçe'de sadece doğa taklidi sözcüklerde kullanılır. "caymak" ve "cici" sözcükleri bu kuralın istisnalarıdır. "cıvıl", "cızırtı", "cıyaklamak" bu kurala örnek olarak verilebilecek sözcüklerdir.

- ğ sesi sözcük başında hiç bulunmaz.

- l sesi Türkçe'de sadece doğa taklidi sözcüklerin başında bulunur. "löp", "lokur" sözcükleri bu kurala örnek olarak verilebilir.

- m sesi Türkçe sözcüklerin başında ancak bozulmuş tekrarlarda ve doğa taklidi sözcüklerin başında bulunur. "masmavi", "mosmor", "miyavlamak" sözcükleri bu kurala örnek olarak verilebilir.

- n sesi Türkçe sözcüklerin başında "ne" ile yapılmış birleşik sözcükler ve doğa taklidi sözcükler dışında kullanılmaz. "neden", "niçin", "nasıl" sözcükleri bu kurala örnek olarak verilebilir.

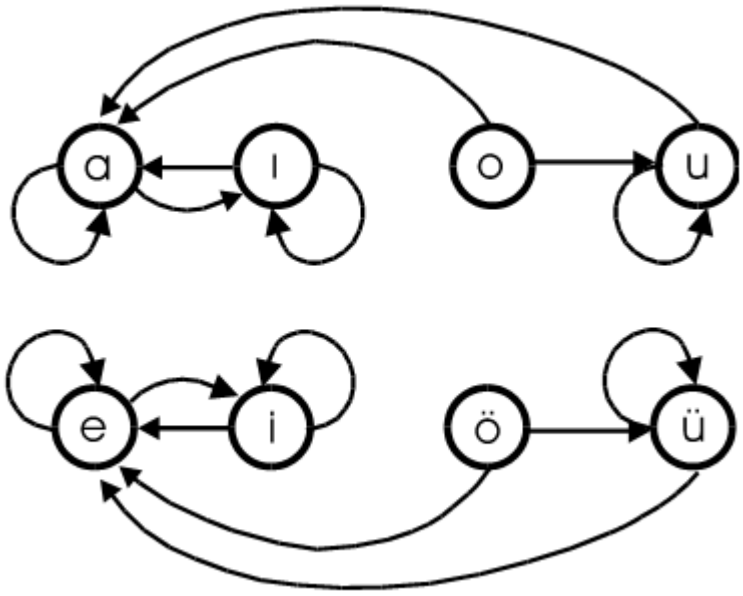
- r sesi sadece doğa taklidi sözcüklerde başta kullanılır. "rap" sözcüğü bu kurala örnek olarak verilebilir.

- v sesi türkçe sözcüklerin başında ancak doğa taklidi sözcüklerde ve aslında b olan, fakat sonradan değişikliğe uğrayan sözcüklerde kullanılır. "vızırtı", "birmek(vermek)" sözcükleri bu kurala örnek olarak verilebilir.

- z sesi sadece doğa taklidi sözcüklerin başında kullanılır. "zırıltı", "zonklamak" sözcükleri bu kurala örnek olarak verilebilir.

2.3.2 Sesli Uyumları

Türkçe'de bir sözcük içinde seslerin dizilişi belli kurallar bağlıdır. Bu kurallara ilişkin sonlu durum makinesi Şekil 2.1 de görülmektedir.



Şekil 2.1 Türkçe'de seslilerin dizilişini gösteren sonlu durum makinesi

Kural olarak; ince seslilerden sonra ince sesliler, kalın seslilerden sonra kalın sesliler gelir; buna *büyük sesli uyumu* denir. İkinci bir kural da; bir sözcüğün ilk hecesinin seslisi düz seslilerden biri ise sonraki hecelerin seslileri de düz sesli olur; ilk hecesinin seslisi yuvarlak seslilerden biri ise sonraki hecelerin seslileri ya dar yuvarlak ya da düz geniş seslilerden biri olur. Bu kurala da *küçük sesli uyumu* denir. Sözcüklere eklenen yapım ve çekim ekleri de bu kurala uyarlar.

Türkçe sözcüklerde sesli dizilişindeki uyum kuralları incelendiğinde bu sözcükleri söylerken ağzın şeklinin hep yatay ya da hep yuvarlak kaldığı görülmektedir. Dolayısıyla Türkçe'de bu kuralların koyulmasındaki hedefin sözcüklerin kolay söylenebilmesini sağlamak ve bu sayede konuşurken çok fazla enerji harcamadan, sözcükleri kolaylıkla ağızdan çıkarabilmek olduğunu söylemek mümkündür. Bunun yanısıra sesli uyumlarıyla ilgili kurallar konuşulan dilin kulağa hoş gelmesini sağlamakta ve Türkçe'ye kulağa hoş gelen bir dil olma özelliği kazandırmaktadır.

Sesli uyumu kurallarına uymayan durumlar şu şekilde sıralanabilir:

- Birleşik sözcüklerde küçük sesli uyumu aranmaz. Bu sözcüklerde büyük sesli uyumu da olmayabilir. "karasinek", "gecekondu", "vatansever" sözcükleri bu duruma örnek olarak verilebilir.

- Yabancı dillerden günümüze geçmiş birçok sözcük büyük ünlü uyumu kuralına uymaz. "demokrasi", "kitap" sözcükleri bu duruma örnek olarak verilebilir. Bu kelimelerin sonuna ek getirilirken en son hecedeki sesli harfle

uyum sağlayacak şekilde getirilir. Ancak sonu "ol" ya da "alp" ile biten yabancı sözcüklere getirilen ekler bu kurala uymaz ve ince sesli içerirler. Bu duruma örnek olarak "kalp" ve "gol" sözcükleri verilebilir.

- **-**yor, -ken , -ki, -leyin, -imtrak, -gil ekleri sesli uyumuna uymazlar.

2.3.3 Sessiz Uyumu

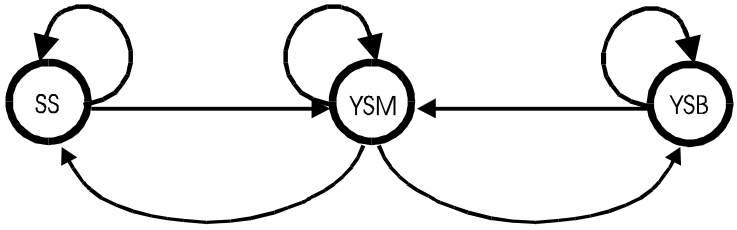
Türkçe'de sessiz uyumu kuralı sözcüklerde yan yana gelen sessizlerin ton bakımından birbirine uygun olması esasına dayanır. Bu kurala göre, sert sessizlerden sonra sert sessiz veya sert karşılığı bulunmayan yumuşak sessiz gelebilir. "ç - f - h - k - p - s - ş – t" harflerinden sonra "ç - f - h - k - p - s - ş - t - l - m - n - r – y" harfleri gelmelidir. Sert karşılığı bulunmayan yumuşak sessizlerden sonra tüm sessizler gelebilir. "l - m - n - r – y" harflerinden sonra bütün sessiz harfler gelebilir. Sert karşılığı bulunan yumuşak sessizlerden sonra yumuşak sessizler gelebilir. "b - c - d - g - ğ - j - v – z" harflerinden sonra "b - c - d - g - ğ - j - v - z - l - m - n - r – y" harfleri gelmelidir.

Türkçe'de sessiz uyumu için sessizlerin nasıl sınıflandırıldığı Tablo 2.5 de görülmektedir.

Tablo 2.5 Türkçe'de sessiz uyumu için sessiz harflerin sınıflandırılması

Sert Sessizler (SS)	**ç, f, h, k, p, s, ş, t**
Sert Karşılığı Bulunmayan Yumuşak Sessizler (YSM)	**l, m, n, r, y**
Sert Karşılığı Bulunan Yumuşak Sessizler (YSB)	**b, c, d, g, ğ, j, v, z**

Şekil 2.2 ise sessiz uyumu için sonlu durum makinesine göstermektedir.



Şekil 2.2 Türkçe'de sessizlerin dizilişini gösteren sonlu durum makinesi

Sessiz uyumuna örnek olarak "toprak" sözcüğü verilebilir. p sert sessiz, r ise sert karşılığı bulunmayan yumuşak sessiz olduğundan kurala uyar. Yabancı kökenli sözcükler de bu kurala uymaya zorlanırlar. Örneğin, "takdim" yerine "taktim", "müsbet" yerine "müspet" kullanılır.

Sessiz uyumu kuralını destekleyen ikinci bir kural da Türkçe'de birleşik sözcük ve yer adları dışında n ve b sessizlerinin yan yana bulunamaması kuralıdır. Bu nedenle "anbar" yerine "ambar", "çarşanba" yerine "çarşamba" kullanılmaktadır. Ancak "İstanbul" sözcüğü görüldüğü gibi bu kuralın dışında kalmaktadır.

Türkçe'de bir diğer sessiz uyum kuralı, sözcük ya da hece sonunda bulunabilecek çift sessiz ile ilgilidir. Türkçe sözcüklerin başında birden fazla sessiz harf bulunmaz. Bunun yanısıra Türkçe sözcüklerin sonunda üç sessiz yan yana bulunamaz. Sözük içerisinde de üç sessiz yan yana bulunamaz. Türkçe'de sözcük ve hece sonlarında bulunabilecek sessiz çiftlerine ilişkin kural Tablo 2.6 da görülmektedir. Bu kurala örnek olarak "ölç", "kalp", "kıskanç" sözcükleri verilebilir.

Tablo 2.6 Türkçe'de sözcük ve hece sonunda çift sessiz kuralı

l	**ç, k, p, t**
n	**ç, k, t**
r	**ç, k, p, s, t**
s	**t**
ş	**t**

2.3.4 Sessiz Yumuşaması

Süreksiz sert sessizlerle (*p, ç, t, k*) biten sözcüklere sesli harflerle başlayan bir ek ulandığında sert sessizler yumuşar ve (*b, c, d, g*) 'ye dönüşür. Bu kural uygulandığında gözlenen ses dönüşümleri ve örnekleri aşağıda verilmiştir.

- *p*→*b* : dolap→dolabı, kitap→kitaba gibi
- *ç*→*c* : taç→tacı, yamaç→yamacı gibi
- *t*→*d* : git→gidiş, seyret→seyreden gibi
- *t*→*dd* : cet→ceddimiz gibi

- *t*→*t* : sepet→sepeti, at→atı gibi
- *t*→*tt* : set→setti, hat→hattı gibi
- *k*→*k* : gecik→gecikiş, aşk→aşka gibi
- *k*→*kk* : hak→hakkı gibi
- *k*→*ğ* : sokak→sokağı, açık→açığa gibi
- *k*→*g* : renk→rengi gibi
- *k* sessizi iki sesli arasına düşerse *k, kk, ğ* olur. Aksi halde *g*'ye dönüşür. Eylemlerin sonundaki *k* sessizi *g*'ye, adların sonundaki *k* sessizi *ğ*'ye dönüşür. Ancak bazı aykırı durumlar söz konusu olabilir:
- Tek heceli sözcüklerin çoğu bu kurala uymazlar.

  iç, ok, sap, et, kaç, kırk, dik, yat, ak, kork, üç, kat gibi.
- Bazı yabancı sözcükler bu kurala uymazlar.

  aşk, gayret, ehemmiyet, merhamet, hukuk, millet, not gibi.

2.3.5 Sessiz Benzeşmesi

Sert sessizlerden biri ile biten bir sözcüğe yumuşak sessizlerden biri ile başlayan bir ek ulandığında yumuşak sessiz sertleşir. Örneğin kebap+ci kebapçı olurken, kümes+de kümeste olur. Ancak özel isimlere gelen eklerde bu kural söz konusu değildir.

Türkçe bir sözcüğün sonunda süreksiz yumuşak sessiz bulunmaz; bu sesler süreksiz sert sessizlere dönüşür.

- *b* sesi sözcük sonunda bulunmaz. Eski devirlerde kullanılanlar da değiştirilmiştir.
  eb→ev, sab→sav gibi.
- Sözcük ve hece sonunda bulunan *d* sesi günümüzde *t* ya da *y*'ye dönüşmüştür.
  kod→koy gibi.
- Hece ve sözcük sonunda bulunan *g* sesleri günümüzde yerini *k, ğ, y* ve *v* sessizlerinden birine bırakmıştır.

döğ→döğ→döv, öğ→öğ→öv, beğ→bey gibi.

Türkçe olmayan bazı sözcüklere de bu kural uygulanmıştır. ilac→ilaç, kitab→kitap, aheng→ahenk, derd→dert gibi.

Bu kurala aykırı durumlar da görülmektedir.

- Anlam karışıklığına sebep olacak bazı sözcüklerde bu değişiklik yapılmamıştır. ad-at, od-ot, sac-saç, kalb-kalp, harb-harp, haç-hac gibi.
- Sadece yabancı dilden geçmiş sözcüklerin sonunda *g* ünsüzü bulunabilir. diyalog, radyolog, arkeolog gibi

2.3.6 Ses Değişmeleri

Sonu geniş düz (*a, e*) bir sesli ile biten eylem kök veya gövdelerine *-en, -ecek, -erek, -e, -yor* eklerinden biri getirilirse eylem kök ya da gövdesinin sonundaki geniş düz sesli dar düz (*ı, i*) sesliye dönüşür. Bu dar düz sesli iki yuvarlak sesli arasına düşerse dar yuvarlağa (*u, ü*) dönüşür. Örneğin anla+yor anlıyor, ağla+yor ağlıyor, dinle+yor dinliyor şekline dönüşür.

2.3.7 Sesliler Çatışmasının Giderilmesi

Dilimizde bileşik sözcükler dışında iki seslinin yan yana gelmesine izin verilmez. Bu kural iki şekilde uygulanmaktadır :

- Bir sözcük sonu sesli ile biter ve o sözcüğe yine sesli ile başlayan bir ek ulanırsa sözcük ile ekin arasına '*n*', '*s*', '*y*', '*ş*' harflerinden birisi girer. Buna ses türemesi, kaynaşma denir.

  kapı+i→kapıyı, bahçe+e→bahçeye, yedi+er→yedişer gibi.

- Sonu sesli ile biten bir sözcüğe *-(H)dH, -(H)mHş, -(H)sA, -(H)ken* ek eylemleri veya ile ilgeci ulanır ve birleşik yazılırsa bu eklerin "*I*" leri *-y*'ye dönüşür.

  kapı+ile→kapıyla, yazı+imiş→yazıymış gibi.

2.3.8 Ses Düşmeleri

Birinci hecesi açık (sesli harfle biten) ikinci hecesi kapalı (sessiz harfle biten) bazı iki heceli sözcüklerde sesliyle başlayan bir yapım veya çekim ekinin sözcüğün orta

hecesinin seslisini düşürmesi olayıdır. Bu ses olayının meydana geldiği durumları aşağıdaki gibi sınıflandırabiliriz.

- Organ adları: alın, ağız, burun, omuz, bağır, göğüs, karın, boyun. burun+i→burnu gibi.
- Akraba adları: kayın, oğul. oğul+i→oğlu gibi.
- Yabancı soylu sözcüklerden bazıları: fikir, zikir, şükür, asır, sabır, akıl, usul, gönül. gönül+i→gönlü gibi.
- Bazı eylem kökleri: devir-, kıvır-, çevir-, ayır-, sıyır-, kavur-, savur-, yayıl-. kavrul+ul→kavrul gibi.
- Bu eylem köklerine -i, -ik, -im türetme ekleri veya -il edilgenlik eklerinden biri getirildiğinde bu kurala uyarlar. yayıl+im→yaylım, devir+i→devri, sıyır+ık→sıyrık gibi

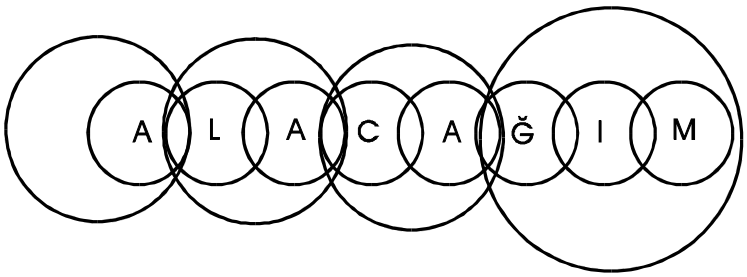
Başka bazı nedenlerden dolayı da ses düşmeleri görülebilir. Bunlar aşağıda örnekleri ile açıklanmıştır:

- Sonu sesli harflerden biri ile biten sıfat köklerine isimden eylem yapma eki -*Al* ulanırsa yapım ekinin seslisi düşer. ince+el→incel, kısa+al→kısal, doğru+el→doğrul gibi.
- Bu yapım eki sonu "*k*" ile biten sıfat köklerine ulanırsa kendi seslisi dışında sıfattaki "*k*" sessizini de düşürür. alçak+el→alçal, yüksek+el→yüksek, küçük+el→küçül gibi.
- Sonu sesli harflerden biri ile biten sıfat köklerine addan eylem yapma eki *-Ar* ulanırsa sıfat kökünün son sesi düşer. sarı+ar→sarar-, kara+ar→karar- gibi.
- Sonu sessiz ile biten sıfat köklerine addan eylem yapma eki *-Ar* ulanırsa sıfat kökünün son sesi ve sonrası (genellikle '*l*')düşer. kızıl+er→kızar-, yeşil+er→yeşer- gibi.

2.4 Heceler

Seslerin öbekleşmesiyle heceler ve sözcükler oluşur. Bir solukta çıkan bir tek ya da bileşik sese *hece* denir. Her ses hecenin bir öğesidir. Hecelerin anlamlı olması gerekmez nitelikleri bir solukta söylenebilmeleridir.

Bir sesin boğumlanmasında üç basamak olduğu gibi bir hecenin boğumlanmasında da üç basamak vardır. Türkçe'de her hecede bir tek sesli bulunur. Her sesli kendinden önceki sessizi kendi hecesi içine alır. Bir sesin boğumlanması Şekil 2.3 te gösterilmiştir.



Şekil 2.3 Türkçe'de bir sesin boğumlanması

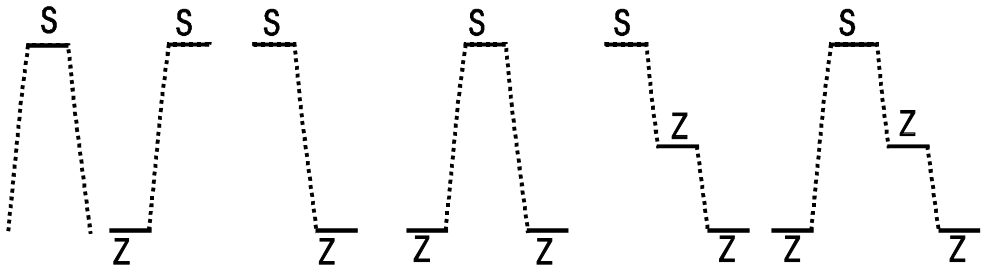
2.4.1 Türkçe'de Hece Yapısı

Türkçe'de altı temel hece yapısı bulunmaktadır. Ses kuralları bu hece yapılarına izin vermektedir. Buna göre Türkçe hecelerin içinde en az bir, en çok dört ses bulunabilir. Hece yapıları aşağıda verilmiştir. (S : sesli ve Z : sessiz harfi göstermektedir.)

- Bir sesliden meydana gelen hece yapısı (S) : o
- Bir sesli ve bir sessizden meydana gelen hece yapısı (SZ) : al
- Bir sessiz ve bir sesliden meydana gelen hece yapısı (ZS) : ye
- Bir sesli, bir sessiz ve bir sesliden meydana gelen hece yapısı (SZS) : gök
- Bir sesli ve iki sessizden meydana gelen hece yapısı (SZZ) : alt
- Bir sessiz, bir sesli ve iki sessizden meydana gelen hece yapısı (ZSZZ) : Türk

Bir sesli ile biten hecelere açık hece sessiz ile biten hecelere ise kapalı hece denir. Türkçe'de tek heceli sözcükler de bu altı biçimden birine uymak zorundadır. Tek heceli kök sözcüklerin çoğu birer kapalı hecedir. Al, vur, sev, at, kuş, aş gibi.

Şekil 2.4 te, hecelerdeki ses iniş ve çıkışları modellenmiştir. Şekilden görüldüğü gibi, Türkçe'de doruğun solunda ancak bir, sağında ise en çok iki sessiz bulunur.



Şekil 2.4 Türkçe hece yapısında ses iniş ve çıkışları

Türkçe'ye yabancı dillerden girmiş olan sözcükler yukarıda belirtilen hece yapılarına uymazlar. Bu duruma örnek olarak gram, Slovak, tren sözcükleri verilebilir.

Türkçe sözcüklerde hece birleştirme kuralları da vardır. Bu kuralları şu şekilde sıralamak mümkündür [16]:

- Sessiz ile biten bir heceye sessiz ile başlayan bir ek geliyorsa o zaman ikinci hecede "ğ" yoktur.
- Aynı hecede sonda iki sessiz harf varsa ve sondan ikinci sessiz harf "l" harfi ise sonuncu harf "ç,k,p,t" harflerinden biridir.
- Aynı hecede sonda iki sessiz harf varsa ve sondan ikinci sessiz harf "r" harfi ise sonuncu harf "ç,k,p,s,t" harflerinden biridir.
- Aynı hecede sonda iki sessiz harf varsa ve sondan ikinci sessiz harf "s" harfi ise sonuncu harf "t" harfidir.
- Aynı hecede sonda iki sessiz harf varsa ve sondan ikinci sessiz harf "ş" harfi ise sonuncu harf "t" harfidir.

2.5 Sözcükler

Anlamı ya da cümlede bir dilbilgisi görevi bulunan bir ya da birkaç heceli ses işaretidir. Sözcükte anlam aramak şarttır, örneğin "O / bugün / geliyor" cümlesinde "o" tek sesli bir hece olmasına karşın anlamı olduğu için sözcüktür. İlgeçlerin ve bağlaçların tek başlarına anlamları yoktur ancak onların anlamları sözcükler arasında ya da cümleler içindeki dilbilgisi görevlerine göre sezilir.

2.6 Kökler ve Ekler

Türkçe sözcük türetme açısından oldukça üretken bir dildir. Bitişken bir dil oluşu, eklerin kabarık sayısı ve eklerin ulandıkları sözcük çeşidine göre sözcüğe çok farklı anlamlar katabilmesi Türkçe'nin çözümlenmesini karmaşıklaştırmaktadır. Ancak diğer yandan ona bir esneklik de getirmektedir. Türkçe sözcüklerde biçim yapısı bakımından iki birim bulunmaktadır. Bunlar anlamlı birimler ve anlamsız veya görevli birimlerdir.

2.6.1 Anlamlı Birimler

Anlamlı birimler *kök* veya *gövde* olabilir. Kök, bir sözcüğün anlamı ve yapısı bozulmadan, daha küçük parçalara ayrılamayan birimidir. Kökler bir dilde önceden var olan sonradan uydurulmayan ses birlikleridir. Türetmenin temel taşlarıdırlar.

Öztürkçe'de, köklerin genellikle birer heceli olduğu bilinmektedir. Bununla birlikte günümüz Türkçe'sinde iki ve üç heceli kökler de bulunmaktadır. Ancak üç heceli sözcüklerin sayısı oldukça azdır.

Kökler anlamları bakımından isim soylu ve eylem soylu olmak üzere iki temel sınıfa ayrılırlar. İsim soylu kökler bir nesne ya da nitelik ismi olan ya da bir ilgi, duygu bildiren köklerdir. İsim soylu kökler isim, sıfat, adıl, belirteç, ilgeç, bağlaç, ünlem gibi sözcüklerin kökleridir. "göl", "yeşil", "güzel", "sen" kökleri isim soylu köklere örnek olarak verilebilir. Eylem soylu kökler ise bir hareketin ismi olan köklerdir. "geç-", "yıka-" kökleri de eylem soylu köklere örnek olarak verilebilir.

2.6.2 Anlamsız veya Görevli Birimler

Anlamsız birimler eklerdir. Bunlar sözcük türetmekte ya da sözcüklere durum, ilgi, nicelik, kip, zaman ve kişi ile ilgili birer anlam katmakla görevli dilbilgisi öğeleridir. Ekler genellikle büyük ve küçük sesli uyumuna uyarlar ve ses olaylarına (sert sessizlerin yumuşaması ve yumuşak sessizlerin sertleşmesi) duyarlıdırlar.

Türkçe sözcüklere ulanabilecek ekler çekim ve yapım ekleri olarak iki temel sınıfa ayrılabilir. Çekim ekleri sözcüklere durum, ilgi, nicelik, kip, zaman ve kişi ile ilgili özellik katmakla görevlidirler. Çekim eklerinin isim soylu ve eylem soylu sözcüklere ulanışları iki ayrı model uyarınca gerçekleşir. Bu iki model çekim eklerinin isme mi eyleme mi ulandığına bağlı olarak eklerin hangi sırayla uygulanabileceğini belirler. Tablo 2-7 ve Tablo 2-8 de sırasıyla isim ve eylem soylu sözcüklere eklenebilecek çekim eklerinin düzeni görülmektedir.

Tablo 2.7 Türkçe'de isim soylu sözcüklere eklenebilecek çekim ekleri

İSİM MODELİ					
isim kökü	çoğul eki	iyelik eki	durum eki	ilgi eki	isim eylem eki

Tablo 2.8 Türkçe'de eylem soylu sözcüklere eklenebilecek çekim ekleri

EYLEM MODELİ					
eylem kökü	olumsuzluk eki	bileşik eylem eki	Asıl zaman eki	ikinci zaman eki	kişi eki

Türkçe'de bulunan tüm çekim ekleri Ek-A da yer almaktadır.

Yapım ekleri sözcük türetmeye yarayan eklerdir. Sözcüğün anlamını ve çeşidini tümüyle değiştirebilirler. Türkçe'de bulunan tüm yapım ekleri Ek-B de yer almaktadır.

3. GELİŞTİRİLEN YÖNTEMLER VE ALGORİTMALAR

Tez kapsamında geliştirilen yöntemler ve algoritmalar aşağıdaki gibi sıralanabilir:

- Sözlük oluşturma
- Heceleme algoritması
- Kök bulma algoritması
- Sözcüğün biçim birimsel çözümlenmesi
- Türkçe denetimi
- Ses bilgisi açısından sözcük denetimi
- Sözcük önerme

Bu yöntemler ve algoritmalarla ilgili detaylı açıklamalar bölüm içinde verilmiştir.

3.1 Sözlük Yapısı

Tez kapsamında oluşturulan sözlük güncel TDK sözlüğünde yer alan sözcüklerin yanısıra özel isimleri de içinde bulunduran bir text dosyasıdır. Bu text dosyasında sözcükler kök halinde bulunurlar.

Türkçe kurallı bir dil olmasına rağmen aykırı durumları da içeren bir dildir. Bu nedenle kullanılan sözlükte aykırı durumların oluşturacağı hataları ortadan kaldırmak için sözcüklerin etiketlenmesi yoluna gidilmiştir. Etiketlemede izlenen yol şu şekildedir:

- Sözcük özel isim ise bu durumu belirten etiket eklenir.
- Sözcük sessiz yumuşaması kuralına göre değişime uğruyorsa bu durumu belirten etiket verilir.

- Sözcük ses düşmeleri kuralına göre değişime uğruyorsa bu durumu belirten etiket verilir.

Sözlükteki kök sözcük yukarıda belirtilen özel durumlardan birini içeriyorsa kök sözcüğün yanına bu özel durum da yazılır. Böylece sözcük etiketlenmiş olur.

Tez kapsamında gerçeklenen sözlükte yukarıda belirtilen ses değişimlerinin yakalanabilmesi ve bu ses değişimlerinden dolayı oluşan yazım hatalarının düzeltilebilmesinin sözcükleri etiketlemeden gerçeklenip gerçeklenemeyeceği üzerinde çalışmalar yapılmıştır. Bu kapsamda TDK'nın güncel sözlüğü içerisinde bulunan ve ses düşmesi, sessiz yumuşaması gibi Türkçe ses olaylarından etkilenen kelimeler incelenmiştir. Bu inceleme yapılırken özellikle [17] kaynağından oldukça yararlanılmıştır.

Sessiz yumuşaması kuralı ele alınacak olursa, "k" harfi ile biten sözcüklere sesli ile başlayan bir ek getirildiğinde "k" harfinin yerine "g", "ğ", "k" ya da "kk" harfleri gelmektedir. Örneğin ismin "-i" hal eki aşk kelimesine ulanınca kelime aşkı, sokak kelimesine ulanınca kelime sokağı, ahenk kelimesine ulanınca kelime ahengi, hak kelimesine ulanınca kelime hakkı halini almaktadır. Görüldüğü gibi "k" harfi için dört farklı durum söz konusu olmaktadır. Benzer durum "t" harfi için de geçerlidir. Bu aykırı durumların kaynağının ne olabileceği üzerinde yapılan çalışmalar sonucunda varılan nokta şu olmuştur: Kökeni doğu dillerine dayanan sözcüklerde genellikle yumuşama olurken, batı dillerinden Türkçe'ye girmiş olan sözcüklerde ise yumuşamaya daha az rastlanmaktadır. Örneğin, "kitap" sözcüğünün kökeni Arapçadır ve özgün hali "kitab" şeklindedir. Bu sözcüğe sesli ile başlayan ek ulandığında sözcük özgün haline geri dönmekte ve ek özgün haline ulanmaktadır. Benzer şekilde Arapça kökenli olan hak ve cet sözcükleri de sesli ile başlayan ek aldığında özgün halleri olan hakk ve cedd hallerine geri dönmektedirler. Bunun yanısıra ıslahat, sefahat, tatbikat gibi Arapça kökenli sözcüklerin özgün halleri de şu an kullanılan halleriyle aynı olduğu için bu sözcükler herhangi bir değişime uğramamaktadırlar. Dolayısıyla buradan da görüleceği gibi doğu kökenli dillerden Türkçe'ye geçmiş olan sözcüklerin pek çoğunda sözcük, sessiz yumuşaması kuralına uymak için özgün haline geri dönmektedir.

Yabancı dillerden Türkçe'ye girmiş olan sözcükler üzerinde de benzer çalışma yapılmıştır. Örneğin Fransızca'dan dilimize girmiş olan akrobat, bürokrat, demokrat sözcükleri sesli ile başlayan bir ek aldığında değişime uğramazken yine Fransızca'dan dilimize girmiş olan piramit sözcüğü sesli ile başlayan bir ek aldığında değişime uğramakta ve sözcük sonundaki "t" harfinin yerini "d" harfi almaktadır. Benzer şekilde Fransızca kökenli olan bank sözcüğünde yumuşama söz konusu

değilken, müzik sözcüğünde yumuşama olmaktadır. Bu noktada yabancı dillerden Türkçe'ye girmiş olan sözcüklerin bir kısmında yumuşama gerçeklenmekte, bir kısmında gerçeklenmemektedir. Dolayısıyla bu çalışmadan anlaşılmıştır ki, her ne kadar doğu kökenli sözcükler yumuşama açısından genel bir tavır sergileseler de batı kökenli sözcüklerde yumuşama açısından belli bir kural oluşturulamamıştır. Dolayısıyla hem yazım yanlışlarını tespit etme ve düzeltmedeki hata oranını en aza indirgemek açısından hem de sözcükleri kökenlerine göre ayırmanın tezin konusu kapsamı dışında kalacağı düşünüldüğünden, kullanılan sözlük içindeki sözcükler yumuşama kuralı için etiketlenmişlerdir.

Ses düşmeleri açısından da TDK sözlüğündeki sözcükler üzerinde çalışma yapılmıştır. Ses düşmesi kuralının sessiz yumuşaması kuralına göre daha kurallı bir yapıya sahip olduğu söylenebilir. Ancak burada da aykırı durumlar söz konusudur. Örneğin, "beniz" sözcüğü sesli ile başlayan bir ek aldığında ikinci hecedeki "i" seslisi düşmektedir. Ancak "deniz" sözcüğüne bakılacak olursa değişen tek harf sözcük başındaki harf olmasına rağmen, bu sözcükte ses düşmesi söz konusu olmamaktadır. Dolayısıyla sözlükteki kelimelerin ses düşmesine göre etiketlenmesi de bu aykırı durumların ortaya çıkaracağı hataları en aza indirgemeyi sağlayacak ve başarımı artıracaktır.

Tez kapsamında gerçeklenen sözlük yukarıda anlatılan özellikler dikkate alınarak oluşturulmuştur. Bu sözlüğün dışında kullanıcının girmiş olduğu cümle ya da metinlerdeki yabancı kökenli sözcüklerin yerine Türkçe sözcükler önermek amacıyla bir text dosyası daha oluşturulmuştur. Bu dosyada dilimize yabancı dillerden girmiş olan ancak Türkçe karşılığı bulunan sözcüklerin Türkçe karşılıkları ilgili sözcüğün yanına yazılmıştır. Örneğin arkeoloji sözcüğü dilimize Fransızca'dan girmiştir; bu kelimenin yerine önerilen sözcük "kazı bilimi" dir. Benzer şekilde agresif sözcüğü yerine de "saldırgan" sözcüğü önerilebilir. Burada amaç, Türkçe karşılıkları bulunan sözcüklerin yabancı dillerdeki hallerinin kullanımının yaygınlaşmasını önlemeye çalışmak ve dilimize yerleşmesini engellemektir.

Kullanıcının girmiş olduğu cümle ya da metinler Türkçe sözlükte bulunmayan sözcük ya da sözcükler içeriyorsa, bu durumda kullanıcı isterse bu sözcükleri sözlüğe ekleme yetkisine sahiptir. Bu amaçla bir yabancı kelimeler dosyası oluşturulmuştur. Bu dosya da diğer dosyalar gibi bir text dosyasıdır. Örneğin, "Bugün National Geographic'de bir belgesel izledim." cümlesindeki National ve Geographic sözcükleri Türkçe olmadığından kullanıcıya uyarı verilecektir ve kullanıcı isterse yabancı kelimeler dosyasına bu kelimeleri ekleyebilecek ve böylece kendi sözlüğünü oluşturmuş olacaktır.

3.2 Heceleme Algoritması

Heceleme algoritmasının çıkarılması biçim birimsel analizin ilk aşamasını oluşturur. Sözcüğün hecelenmesi, sonraki aşamalar olan sözcük kökünün bulunması ve sözcüğün biçim birimsel analizinin yapılabilmesi için gerekli olan bir aşamadır. Türkçe'de ekler ve kökler arasındaki ses olayları hece boyutunda gerçekleştiği için hecelemenin önemi oldukça büyüktür.

Türkçe'de altı temel hece yapısı bulunmaktadır. Ses kuralları sadece bu hece yapılarına izin vermektedir. Bu kurallara göre Türkçe bir hecenin içinde en az bir, en fazla dört ses bulunabilir. Türkçe'deki hece modelleri aşağıda verilmiştir. Burada S harfi sesli harfi, Z harfi ise sessiz harfi simgelemektedir.

- Bir sesliden oluşan model (S) : o
- Bir sesli ve bir sessizden oluşan model (SZ) : al
- Bir sessiz ve bir sesliden oluşan model (ZS) : ye
- Bir sesli, bir sessiz ve bir sesliden oluşan model (SZS) : gök
- Bir sesli ve iki sessizden oluşan model (SZZ) : alt
- Bir sessiz, bir sesli ve iki sessizden oluşan model (ZSZZ) : Türk

Türkçe'deki bu hece modelleri sayesinde hecelemenin yapısal olarak modellenmesi ve yazılımsal olarak gerçeklenmesi mümkündür. Bu gerçeklemenin dışında kalan bazı aykırı durumlar ise yabancı kökenli sözcüklerden kaynaklanmaktadır.

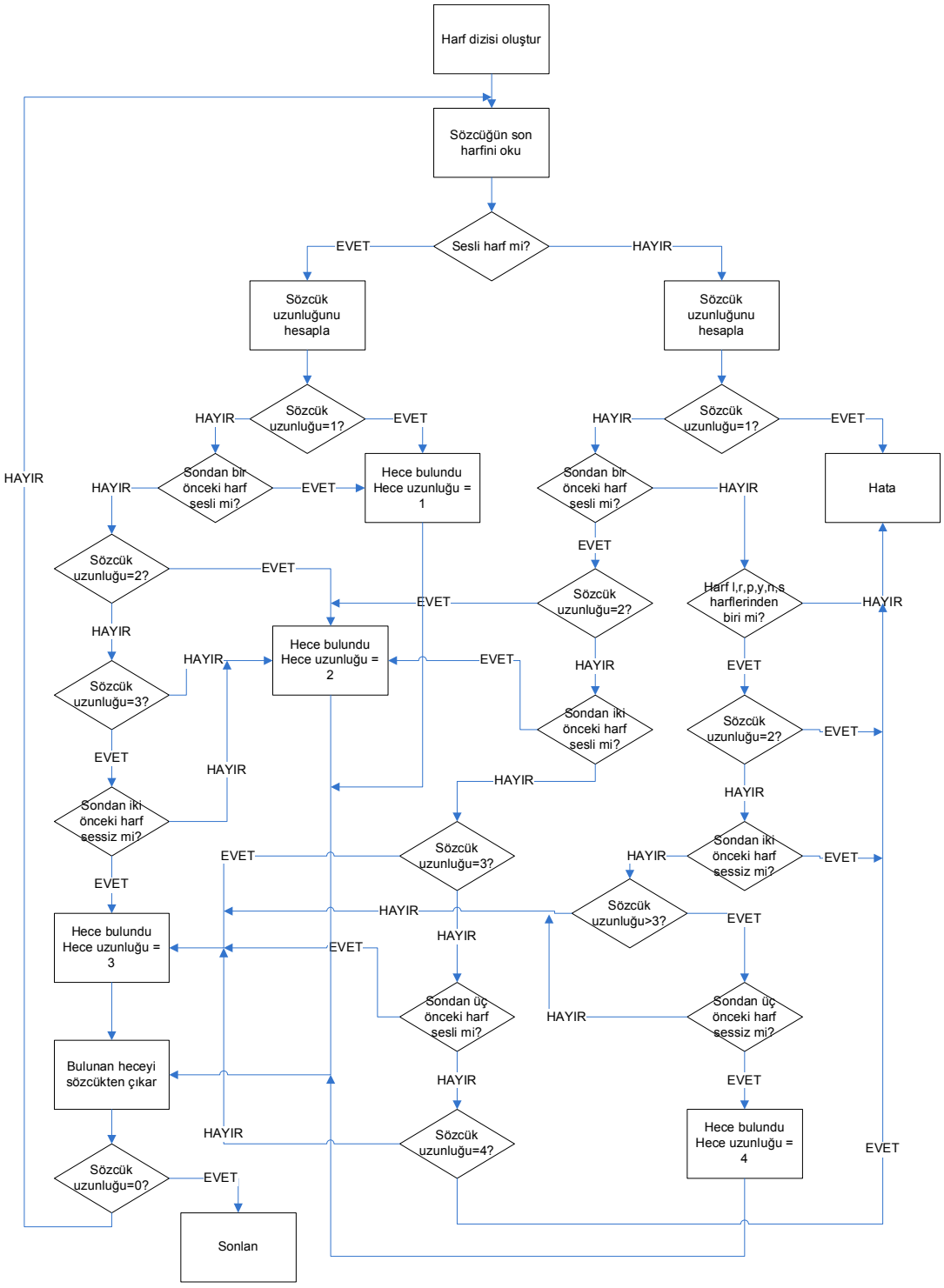
Tez kapsamında gerçeklenen heceleme algoritmasının temeli yukarıda anlatılan Türkçe hece modelleri esasına dayanmaktadır. Buna göre gerçeklenen algoritmada sözcük sondan başa doğru analiz edilmekte ve bu şekilde heceleme yapılmaktadır [18,19].

Öncelikle kelimenin sonundaki harf okunur ve bu harfin sesli mi yoksa sessiz mi olduğu denetlenir. Eğer son harf sessiz ise kelime uzunluğu denetlenir. Kelime uzunluğu bir ise bu hata durumudur; çünkü Türkçe hece modelleri esasına göre Türkçe'de heceler sadece bir sessiz harften oluşamazlar. Kelime bu hata denetimini geçerse sondan bir önceki harf ele alınır. Eğer sondan bir önceki harf sesli ise, bu durumda kelime uzunluğu denetlenir. Eğer kelime uzunluğu iki ise (örneğin; al) ya da sondan iki önceki harf sesli ise (örneğin; saat, zaaf) hece bulunmuş demektir. Eğer kelime uzunluğu üç(örneğin; yat, kat) ise ya da sondan üç önceki harf sesli ise

(örneğin; kitap, kalem) hece bulunmuş demektir. Bu koşullar sağlanmıyorsa sondan dört önceki harfe bakılır ve bu harf sesli ise (örneğin, kezzap) hece bulunmuş demektir. Bu koşulların dışındaki durumlarda hata oluşturulur. Buraya kadar anlatılanlar sondan bir önceki harfin sesli olması durumunda izlenecek yolu göstermektedir. Sondan bir önceki harf de son harf gibi sessiz ise ve bu harf "l, r, y, p, n, s" harflerinden biriyse bu durumda hece Türkçe hece modellerinden SZZ ya da ZSZZ modellerinden birine sahip demektir. Kelime uzunluğu 2 ise ya da sondan iki önceki harf de sessiz ise hata oluşturulur. Çünkü Türkçe hece modellerine göre bir hece iki sessiz (ZZ) ya da üç sessizden (ZZZ) oluşamaz. Bu denetim sayesinde yabancı dillerden dilimize girmiş olan ve sözcük başında ya da ortasında birden fazla sessiz harfi yan yana bulunduran sözcüklerin hecelenmesine izin verilmez. Kelime uzunluğu üçten büyük ve sondan üç önceki harf sessiz ise (örneğin; kalp, Türk, ebeveyn) hece bulunmuş demektir.

Son harfin sesli olması durumunda ise uygulanan yöntem şu şekildedir: Kelime uzunluğu bir ise (örneğin; o) hece bulunmuş demektir. Sondan bir önceki harf sessiz ise ve kelime uzunluğu iki ise (örneğin; ye, ya) hece bulunmuş demektir. Sondan iki önceki harf sessiz ve kelime uzunluğu üç ise hece bulunmuş demektir. Tüm bu denetimlerin dışında kalan durumlar için hata üretilir ve heceleme yapılamaz.

Yukarıda da belirtildiği gibi heceleme kelime sonundan başına doğru yapılmaktadır. Dolayısıyla ilk hece bulunduktan sonra bulunan hece sözcükten çıkarılmakta ve anlatılan denetimlerin hepsi sözcükte kalan diğer heceler için de aynen gerçekleştirilmektedir. Her hece bulunduğunda, bulunan hece dışında sözcükte kalan diğer heceler için aynı işlemler tekrarlanmaktadır ve bu işlem sözcük uzunluğu sıfırlanana kadar devam etmektedir. Böylece sözcükteki tüm heceler bulunabilmektedir. Şekil 3.1 heceleme algoritmasına ait akış diyagramını göstermektedir.



Şekil 3.1 Heceleme algoritması akış diyagramı

3.3 Kök Bulma Algoritması

Tez kapsamında kullanılan kök bulma algoritması ağaç veri yapısı ile gerçeklenmektedir. Sözlükten okunan tüm kökler bu ağaca yerleştirilir. Kök ağacı

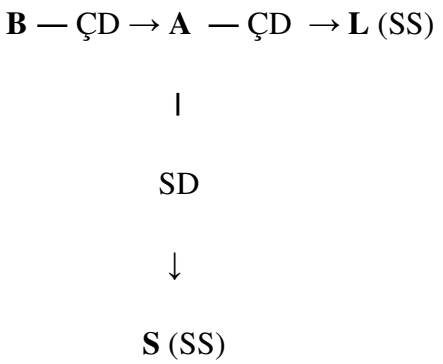
Yönlü Döngüsüz Kelime Çizgesi (DAWG) yapısına sahiptir. Sözlükten okunan köklerin kök ağacına yerleştirilmesinde Yönlü Döngüsüz Kelime Çizgesi oluşturma algoritması kullanıldığından dolayı bu bölümde ilk olarak Yönlü Döngüsüz Kelime Çizgesi yapısı ile ilgili bilgilendirme yapılmış ve ardından da ağaç üzerinde aranan bir kökün nasıl bulunabileceği anlatılmıştır.

3.3.1 Yönlü Döngüsüz Kelime Çizgesi Yapısı

Yönlü döngüsüz kelime çizgesi hızlı kelime aramalarına olanak tanıyan bir veri yapısıdır. Çizgeye giriş noktası aranan kelimenin başlangıç harfidir. Her düğüm bir harfi temsil eder. Çizgede ilerlerken karşılaşılan harf ile aranan harfin aynı olmaması durumunda bir düğümden diğer düğümlere geçiş yapılabilir.

Çizgenin yönlü bir çizge olmasının nedeni, herhangi iki düğüm arasında sadece özel ve belirli bir yolda ilerlenilebilmesidir. Diğer bir deyişle, A düğümünden B düğümüne gidilebilir; ancak B düğümünden A düğümüne gidilemez. Döngüsüz olmasının nedeni ise çizge üzerinde herhangi bir yolun başlangıç ve bitiş düğümünün aynı olmamasından kaynaklanmaktadır. Bir başka deyişle, çizge üzerinde A->B->C->A şeklinde bir yol bulunması söz konusu değildir [20].

Şekil 3.1 "bal", "bas" sözcüklerinin yönlü döngüsüz kelime çizgesi içindeki yerleşimi göstermektedir. Şekillerdeki ÇD çocuk düğüm, SD sonraki düğüm, SS ise sözcük sonu anlamına gelmektedir.

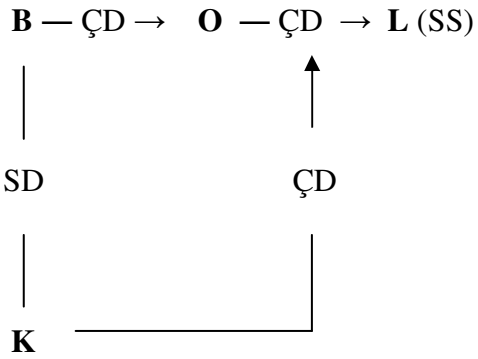


Şekil 3.2 "BAL" ve "BAS"sözcükleri için yönlü döngüsüz kelime çizgesi

Şekil 3.2 deki yönlü döngüsüz kelime çizgesinde BAS sözcüğünün aranması durumunda giriş noktasından (B harfi) arama başlayacaktır. B harfi aradığımız sözcüğün ilk harfi olduğundan dolayı, bundan sonra yapılacak iş B harfinin çocuk düğümüne gitmektir. BAS sözcüğü için aradığımız ikinci harf A harfidir ve bu harf çizge üzerinde B harfinin çocuk düğümü olan A harfi ile aynı olduğundan bu kez de

A harfinin çocuk düğümüne bakılacaktır. BAS sözcüğü için aranılması gereken sıradaki harf S harfidir. Ancak bu harf, çizge üzerinde A harfinin çocuk düğümü olan L harfinden farklı olduğu için L harfinin çocuk düğümüne bakılmayacak ve bir sonraki düğüme (S) geçilecektir. Bu düğüm aranılan harf olan S harfini göstermektedir. Aranılan kelime içindeki tüm harfler bulunduğuna göre, bundan sonra yapılması gereken sözcük sonuna gelinip gelinmediğinin denetlenmesidir.

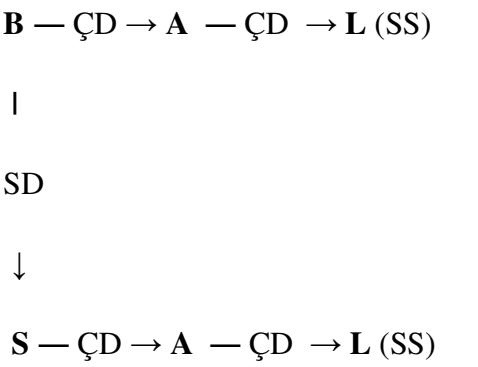
Yönlü döngüsüz kelime çizgelerinde aynı harfle biten sözcüklerin aynı düğümde sonlanması işlemi gerçekleştirilebilir. Şekil 3.3 deki "bol" ve "kol" sözcüklerinin yönlü döngüsüz kelime çizgesindeki yerleşimi bu duruma örnek olarak gösterilebilir.



Şekil 3.3 "BOL" ve "KOL"sözcükleri için yönlü döngüsüz kelime çizgesi

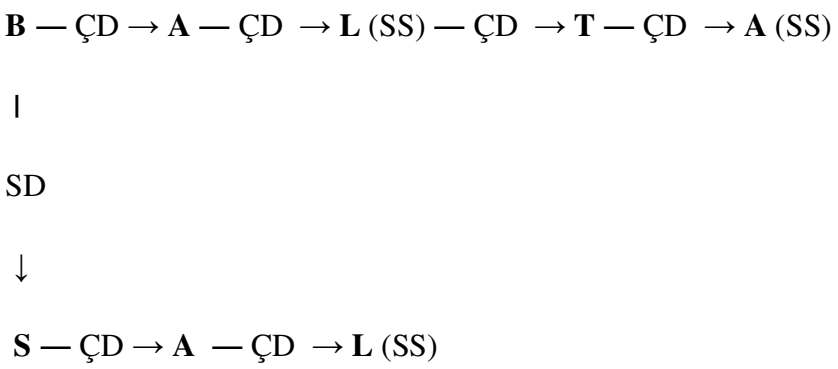
3.3.2 Yönlü Döngüsüz Kelime Çizgesi Oluşturma

Yönlü döngüsüz kelime çizgesi oluşturmak için ilk olarak ağaç yaratılır. Ağaçtaki yapraklardan biri sözcük sonunu temsil eder. Ağaçta birbirinin aynı olan çok sayıda yaprak bulunabilir. Şekil 3.4 "bal" ve "sal" sözcüklerinin ağaçtaki dizilişi görülmektedir.



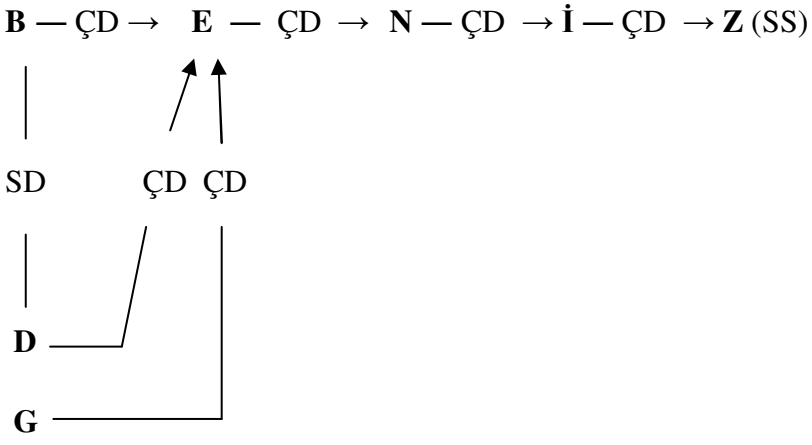
Şekil 3.4 "BAL" ve "SAL"sözcükleri için ağaç oluşumu

Şekil 3.4 de görülen ağaca balta sözcüğü eklenmek istendiğinde, L harfine gelene kadar ağaç üzerinde arama yapılır. L harfine gelindiğinde bu düğümün çocuğu olmadığı görülür ve T düğümü son düğüm olan L düğümüne, A düğümü de son eklenen T düğümüne eklenir. Sonuçta oluşan durum Şekil 3.5 deki gibidir.



Şekil 3.5 "balta" sözcüğünün ağaca yerleştirilmesi

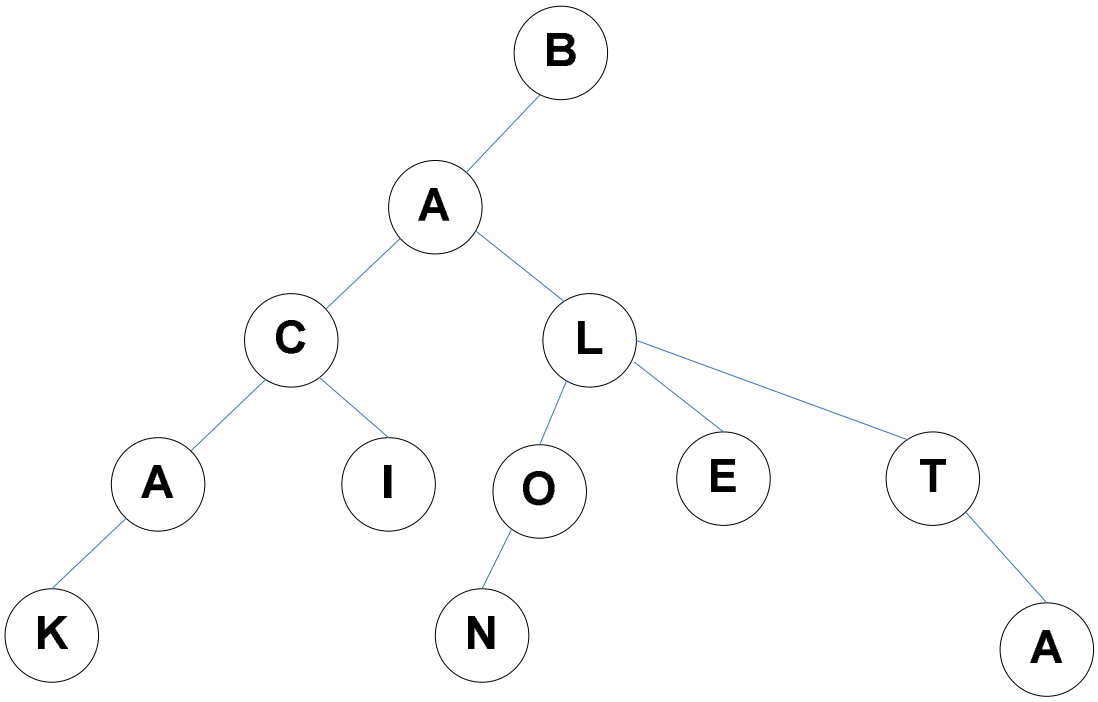
Şekil 3.5 den de görüleceği gibi ağaca bu şekilde sözcük eklemek sözcüklerin başlangıçları için ortak düğümler olmasını sağlarken sözcük sonları her sözcük için farklı olmaktadır. Çizgenin boyutunu daha sınırlı hale getirmek için sözcük sonlarının ortak olan kısımları bulunmalı ve birleştirilmelidir. Bunu gerçekleştirmek için yaprak düğümden yola çıkılır. İki yaprak düğüm aynı ise bunlar birleştirilebilir. İki yaprak düğümün aynı olması için bu düğümlerin aynı harfi taşımaları tek başına yeterli değildir; bir sonraki düğümleri ya da çocuk düğümleri varsa bunları da aynı olması gereklidir. Şekil 3.6 de görülen çizge bu duruma örnek olarak verilebilir.



Şekil 3.6 Sözcük sonlarının ortak düğümlere bağlanması

3.3.3 Ağaç Üzerinde Kök Bulma

Ağaç üzerinde bir kökü bulmak için ağacın başından itibaren kökü oluşturan harfleri temsil eden düğümleri izlemek gerekir. Şekil 3.7 de görülen ağaç üzerinde örneğin balo kökü bulunmak istendiğinde, ilk olarak B başlangıç düğümünden A düğümüne, A düğümünden L düğümüne ve oradan da O düğümüne geçmek gerekir.



Şekil 3.7 Ağaç üzerinde kök bulma

Ağaç üzerinde kök bulma işlemi bir veri yapısı olan ağacın düğümlerinin üzerinde gezilmesi ve aranan düğümle üzerinde bulunulan düğümün karşılaştırılması şeklinde gerçekleştirilmektedir. Dolayısıyla kök bulma algoritmasının özü ağaca yerleştirilmiş olan kökler üzerinde yapılan ağaç işlemleridir. Girilen kelimenin ilk harfinden başlanarak ağaç üzerinde ilerlenilir. İlerlenecek yer kalmayana ya da kelime bitene kadar ağaç üzerinde ilerleme işlemine devam edilir. Ağaç üzerinde ilerlerken karşılaşılan tüm kökler aday olarak toplanır. Örneğin; balon kelimesi için ağaç üzerinde yürürken karşılaşılan bal, bale ve balon köklerinin üçü de aday kökler olarak bulunur.

Ses değişimleri (sessiz yumuşaması, ses düşmesi vb.) nedeniyle değişime uğrayan sözcüklerin ağaca eklenmesi sırasında hem sözcüğün özgün hali hem de değişime uğrayan hali ağaca eklendiği için kök bulma işlemi sırasında herhangi bir sorunla karşılaşılmaz. Örneğin, şarap sözcüğü hem şarap hem de şarab olarak ağaca

eklenmektedir; ancak bu iki düğümün de işaret ettiği kök aynıdır ve bu kök şarap köküdür. Böylece sessiz yumuşamasına uğramış olarak aranan sözcüğün kökünün de ağaçta bulunması sağlanmış olur.

3.4 Sözcüğün Biçim Birimsel Çözümlenmesi

Sözcüğün biçim birimsel çözümlenmesinin yapısını daha iyi anlamak için öncelikle ekler hakkında bilgi vermenin faydalı olacağı düşünülmüştür. Kökler hakkında gerekli bilgi Bölüm 3.1 de sözlük oluşturma aşamasında anlatıldığı için bu bölümde ekler ile ilgili bilgilendirme yapılmış, daha sonra da biçim birimsel çözümleme işleminde kullanılan ayrıştırıcının yapısı anlatılmıştır.

3.4.1 Ekler

Ekler ve eklerin sözcük kök veya gövdelerine ulanışlarında oluşturdukları desenlerin incelenmesi biçim birimsel çözümleme açısından oldukça önemlidir. Eklerin sınıflandırılmasında en önemli aşama yapım ve çekim eklerinin hangi sıralarla birbirlerini izlediklerini belirleyebilmektir. Türkçe dilbilgisi kurallarına göre çekim ekleri yapım eklerinden sonra gelir. Dolayısıyla bir kez çekim eki alan sözcük, daha sonra sadece çekim eki alabilir. Çekim eklerinin birbirlerini hangi sırayla takip edecekleri de kurallara bağlıdır ve bu kurallar Tablo 2-7 ve Tablo 2-8 de verilmiştir. Yapım ekleri için bu şekilde oluşturulmuş net kuralların varlığından bahsetmek henüz mümkün değildir.

Tüm Türkî dillerde ekler dilin çekirdeğini oluştururlar ve bu nedenle eklerin yapısının anlaşılması biçim birimsel inceleme açısından oldukça önemlidir. Şu an için Türkçe'de eklerin isimlendirilmesinde ve gösterilmesinde kullanılan standart bir yöntem mevcut olmamakla beraber bu bölümde verilecek olan örnekler sıkça kullanılan yöntemler baz alınarak verilmiştir. Örneğin çoğul eki olan "-lar" ekinin gösterilimi "-lAr" şeklindedir. Burada küçük harfle yazılmış olan "l" ve "r" harfleri eke doğrudan eklenileceğini gösterirken, büyük harfle yazılmış olan "A" harfi ekleneceği sözcüğün son seslisine göre değişebileceğini ifade etmektedir. Ekin ulanacağı sözcüğün son seslisi kalın sesli ise bu harf "a", ince sesli ise "e" halini alacaktır.

Türkçe'de ekler için özel bazı durumlar söz konusudur. Örneğin, Türkçe'de şimdiki zaman eki olan "-(H)yor" eki ulandığı eylemin son harfi sesli harf ise bu harfin düşmesine neden olur. Örneğin; "ara-" eylem köküne şimdiki zaman eki ulandığında oluşan eylem "ara-yor" şeklinde değil, "ar-ıyor" şeklindedir. Benzer şekilde

eylemleri edilgen yapan ekler de ulanacakları eylemlerin son sessizlerine bağlı olarak değişebilmektedirler. Örneğin; "gel-" eylemine edilgenlik eki ulandığında eylem "gel-in-mek" halini alırken "kes-" eylemi "kes-il-mek" haline dönüşmektedir.

Tez kapsamında kullanılan ek dosyası içinde ekler kümelere ayrılarak sınıflandırılmıştır. Örneğin, ismin hal ekleri için bir küme oluşturulmuş ve bu kümeye dahil olan tüm ekler (-i, -e, -de, -den) bu kümenin içine konulmuştur. Ayrıca çekim eklerinin birbirlerini hangi sırayla takip edebilecekleri ile ilgili kurallar göz önüne alınarak ekler arasında ilişkilendirme yapılmıştır. Bir ekten sonra gelebilecek tüm olası ekler, ilgili ekle ilişkilendirilmiş ve bu sayede eklerin ayrıştırılması daha kolay çözümlenebilir hale getirilmiştir. Bunun yanısıra, yukarıda da belirtildiği gibi bazı eklerin özel durumlara sahip olması nedeniyle bu eklere ilgili özel durumları belirten etiketler verilmiştir.

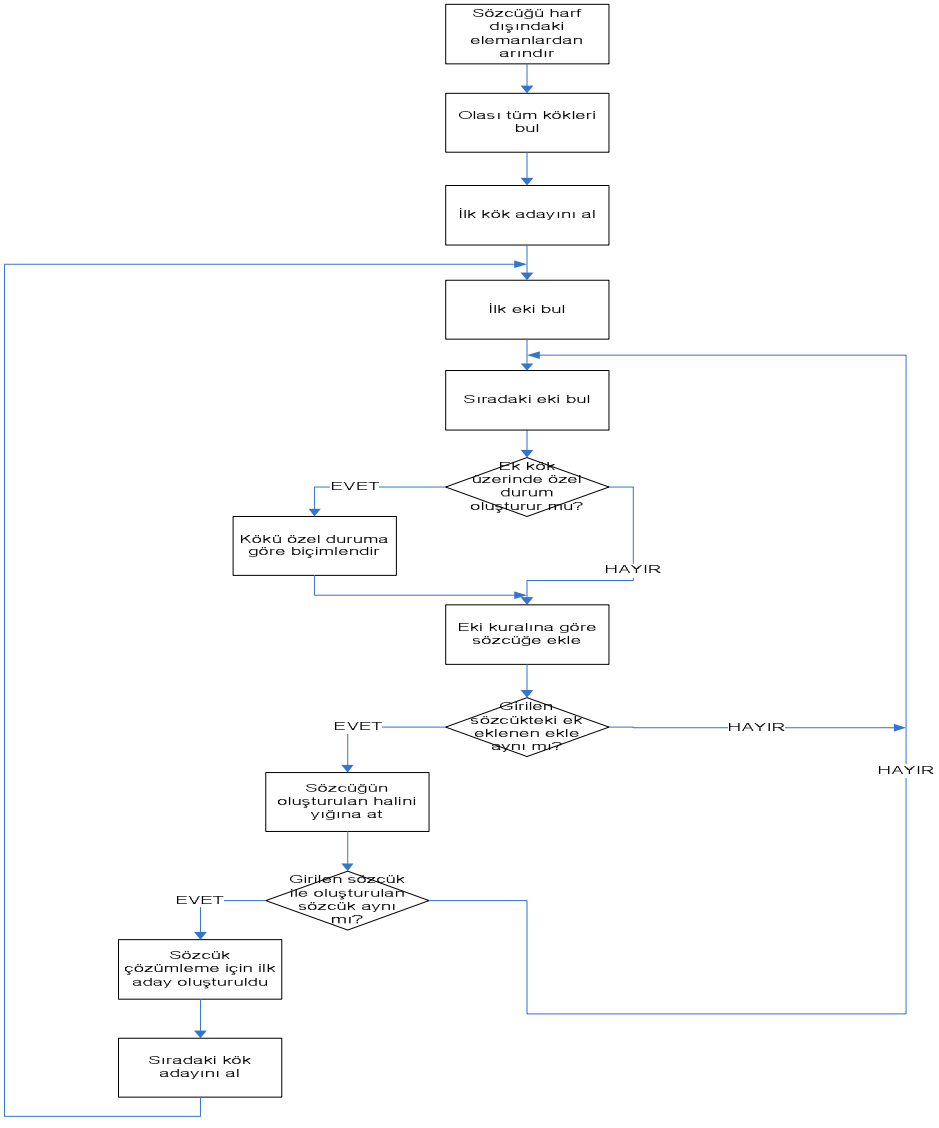
3.4.2 Ayrıştırma Algoritması

Biçim birimsel çözümlemede kullanılan ayrıştırma algoritması girilen bir sözcüğe ait olası tüm kök ve ek çözümlerini bulmak için kullanılır. Girilen bir sözcüğün biçim birimsel çözümlemesinin yapılmasında ilk adım, girilen sözcüğü noktalama işaretleri, kısa çizgi işareti gibi harf dışındaki elemanlardan ayrıştırmaktır. Bu sayede geriye kalan elemanlar sadece kök ve ekler olur.

Harf dışındaki elemanların ayrıştırılmasından sonra yapılan işlem, olası tüm köklerin kök ağacı üzerinde gezilerek bulunmasıdır. Kök bulma algoritması Bölüm 3.3.3 de ele alındığı için burada tekrar anlatılmayacaktır.

Olası tüm kökler bulunduktan sonra eklerin bulunması işlemi başlar. Bunun için kök sözcüğün almış olduğu ilk eke bakılır. Bölüm 3.4.1 de belirtildiği gibi ilk ek kendinden sonra gelebilecek eklerle ilgili bilgi verdiği için sırayla bu ekler ele alınır ve ekin kök sözcük üzerinde özel bir durum oluşturup oluşturmadığına bakılır. Örneğin ek sesli ile başlıyorsa ve kök sözcük sesli düşmesi özel durumuna sahipse bu durumda ek, kök sözcük üzerinde özel bir durum oluşturur. Eklerin kök sözcük üzerinde özel durumların oluşmasına neden olması durumda, kök sözcük özel duruma uygun şekle dönüştürülür. Kök sözcük uygun şekle dönüştürüldükten sonra sıra ekin köke ulanmasına gelir. Burada ekin özelliğine göre köke ulama işlemi gerçekleştirilir. Örneğin; kök sözcük elma ise ve ulanacak ek "-lAr" eki ise ilk olarak sözcüğe "l" harfi ulanır. Daha sonra sözcüğün son seslisinin kalın mı ince mi olduğuna bakılır. Elma kök sözcüğü için son sesli kalın olduğundan ulanacak ek kalın olacak şekilde biçimlendirilir. Bu durumda kök sözcük "elmala" halini alır. Son olarak da "r" harfi sözcüğe ulanır ve böylece "-lAr" ekinin sözcüğe ulanması işlemi

bitmiş olur. İlk ek sözcüğe ulandıktan sonra girilen sözcük ile oluşturulan sözcüğün aynı olup olmadığı denetlenir. Sözcükler aynı ise ulanacak ek kalmamış demektir. Aynı değilse girilen sözcük daha fazla sayıda ek içeriyor demektir; bu durumda oluşturulan sözcük bir yığında saklanır ve ek ulama işlemine devam edilir. Bir sonraki eki bulmak için önceki ekle ilişkilendirilmiş olan eklere bakılır. Bu noktada çekim eklerinin hangi sırayla birbirlerini takip edebileceği kurallarının var olması oldukça işe yaramaktadır. Bir önceki örnekten devam edilecek olursa, girilen sözcük "elmalarım" sözcüğü olmak üzere ilk ek "-lAr" eki sözcüğe eklendikten sonra geride kalan "-ım" eki için "-lAr" ekinden sonra gelebilecek ekler arasında arama yapılır ve bu ekin birinci tekil şahıs iyelik eki olduğu arama sonucunda ortaya çıkar. Bu ek de sözcüğe ulandıktan sonra giriş sözcüğü ile oluşturulan sözcüğün aynı olup olmadığına bakılır ve sözcükler aynı olacağı için çözümleme işlemi gerçekleştirilmiş olur. Şekil 3.8 ayrıştırma algoritmasına ilişkin akış diyagramını göstermektedir.

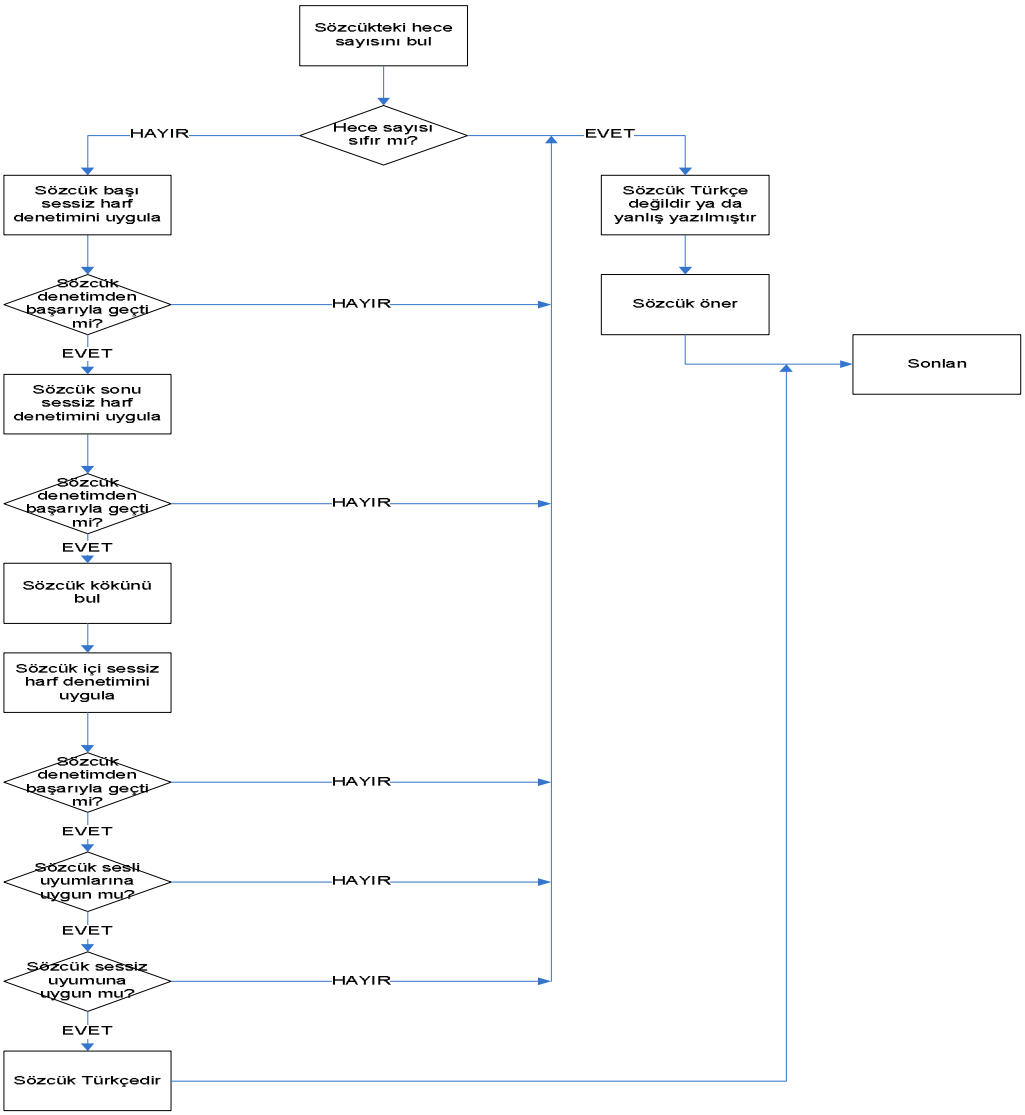


Şekil 3.8 Ayrıştırma algoritması akış diyagramı

3.5 Türkçe Denetimi

Sözcük için Türkçe denetimi yapılmasında amaç girilen sözcüğün Türkçe bir sözcük olup olmadığını denetlemektir. Bunu gerçekleyebilmek için öncelikle sözcüğün hecelenebilir olup olmadığının denetlenmesi gerekir. Eğer girilen sözcük hecelenemiyorsa Türkçe olmadığı kararına varılır.

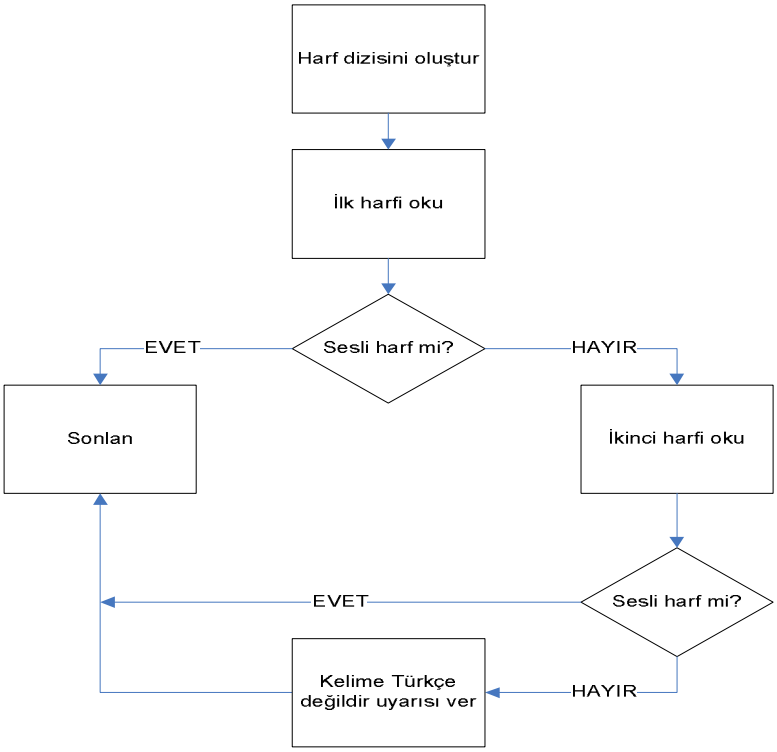
Hecelenebilen sözcüklerde Türkçe sözcüklerde bulunması gereken kuralların denetimi yapılır. Bölüm 2 de detaylı bir şekilde anlatılan bu kuralların geliştirilmesinde izlenen yollar ve gerçeklenen algoritmalar bu bölümde anlatılmaktadır. Bu kurallardan en az bir tanesini sağlamayan sözcük Türkçe bir sözcük değildir ya da dilimize yabancı dillerden girmiş bir sözcüktür. Şekil 3.9 Türkçe denetimine ilişkin genel akış diyagramını göstermektedir.



Şekil 3.9 Türkçe denetimi akış diyagramı

3.5.1 Sözcük Başı Sessiz Denetimi

Türkçe bir sözcüğün başında birden fazla sessiz harf bulunamaz. Dolayısıyla ilk olarak yapılan denetim sözcük başındaki sessiz harf sayısını bulma amacına yöneliktir. Bu aşamada yapılan iş sözcük başındaki sessiz sayısının birden fazla olup olmadığının bulunmasıdır. Eğer bu sayı birden fazla ise sözcüğün Türkçe olmadığına karar verilir. Örneğin, strateji, tren gibi sözcükler dilimize yabancı dillerden girmiş olan sözcüklerdir ve yapılan bu denetim ile bu sözcüklerin Türkçe olmadığı kolay bir şekilde ortaya çıkarılır. Şekil 3.10 sözcük başındaki sessiz harf sayısının denetimine ilişkin akış diyagramını göstermektedir.



Şekil 3.10 Sözcük başı sessiz harf sayısı denetimi için akış diyagramı

3.5.2 Sözcük Sonu Sessiz Denetimi

Sözcük sonundaki sessiz harflerle ilgili Türkçe'de bulunan kurallar şu şekilde özetlenebilir:

- Türkçe sözcüklerin sonunda b, c, d, g harfleri bulunamaz.
- Türkçe bir sözcüğün sonunda ikiden fazla sayıda sessiz harf bulunamaz.
- Türkçe bir sözcük sonunda çift sessiz bulunabilir. Bu durum için kurallar şu şekildedir:

- Son harfi “ç, k, p, t” olan sözcüklerde bir önceki harf sessiz harf ise bu harf “l” harfi olabilir. Örneğin, ölç, kalp gibi.

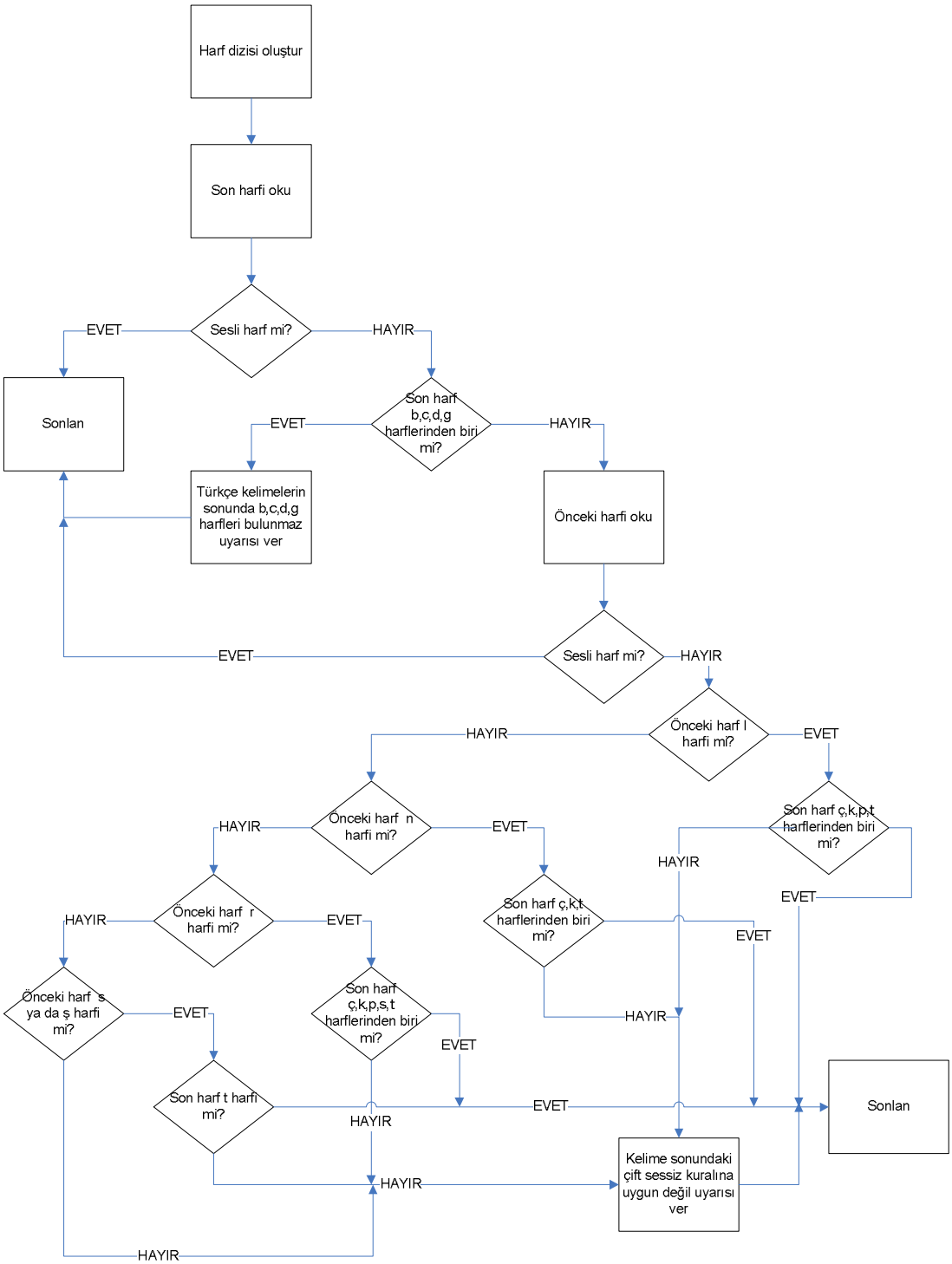
- Son harfi “ç, k, t” olan sözcüklerde bir önceki harf sessiz harf ise bu harf “n” harfi olabilir. Örneğin; kıskanç, iğrenç gibi.

- Son harfi “ç, k, p, s, t” olan sözcüklerde bir önceki harf sessiz harf ise bu harf “r” olabilir. Örneğin; bark, Türk gibi.

- Son harfi “t” olan sözcüklerde bir önceki harf sessiz harf ise bu harf “s” olabilir. Örneğin; dost gibi.

- Son harfi “t” olan sözcüklerde bir önceki harf sessiz harf ise bu harf “ş” olabilir. Örneğin; hişt gibi.

Yukarıda sıralanan kurallardan yola çıkılarak sözcük sonundaki sessiz harf denetlenir. Eğer bu harf b, c, d, g harflerinden biri ise sözcüğün Türkçe olmadığına karar verilir. Eğer sözcük bu kuralı sağlıyorsa, bu durumda sözcük sonundaki sessiz harf sayısına bakılır. Sözcük sonundaki sessiz harf sayısı ikiden büyük ise sözcüğün Türkçe olmadığına karar verilir. Sözcük sonundaki sessiz harf sayısı iki ise sondan bir önceki harfe bakılır. Bu harf “l” ise ve son harf “ç, k, p, t” harflerinden biri değilse sözcük Türkçe değildir. Benzer şekilde sondan bir önceki harf “n” ise ve son harf “ç, k, t” harflerinden biri değilse sözcük Türkçe değildir. Sondan bir önceki harf “r” ise ve son harf “ç, k, p, t, s” harflerinden biri değilse sözcük Türkçe değildir. Sondan bir önceki harf “s” ya da “ş” ise ve son harf “t” değilse sözcük Türkçe değildir. Şekil 3.11 sözcük sonundaki sessiz harf sayısının denetimine ilişkin akış diyagramını göstermektedir.

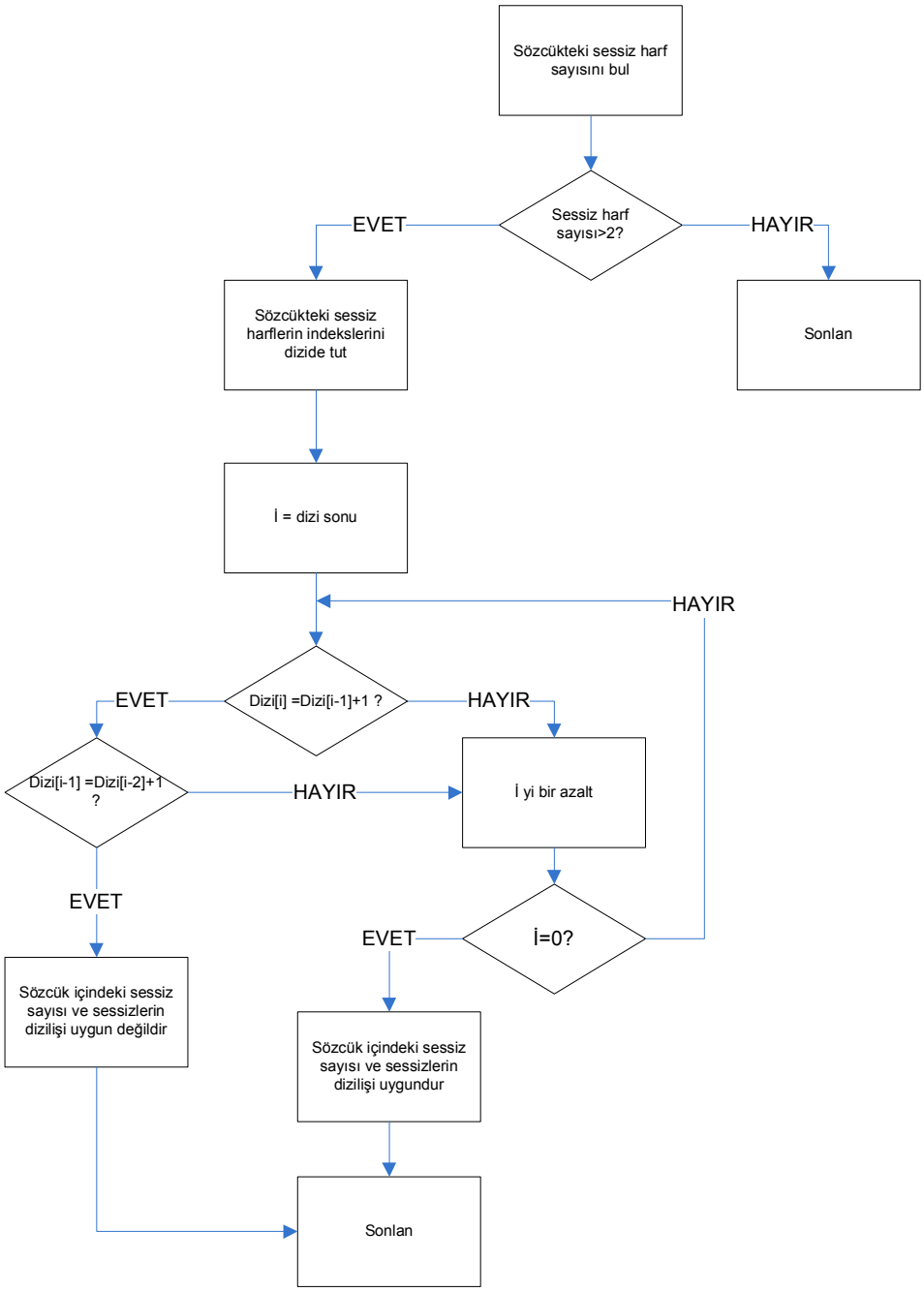


Şekil 3.11 Sözcük sonu sessiz harf sayısı denetimi için akış diyagramı

3.5.3 Sözcük İçi Sessiz Denetimi

Türkçe bir sözcüğün içinde üç ya da daha fazla sayıda sessiz harf yan yana bulunamaz. Bu kuraldan yola çıkılarak sözcük içinde yan yana bulunan sessiz harf sayısının denetimi yapılır. Bunun için girilen sözcük ilk harfinden son harfine kadar

taranır ve sözcük içindeki yan yana bulunan sessiz harf sayısı hesaplanır. Eğer bu sayı üçe eşit ya da üçten büyük ise sözcük Türkçe değildir. Şekil 3.12 sözcük içinde yan yana gelebilecek sessiz sayısının denetimine ilişkin akış diyagramını göstermektedir.



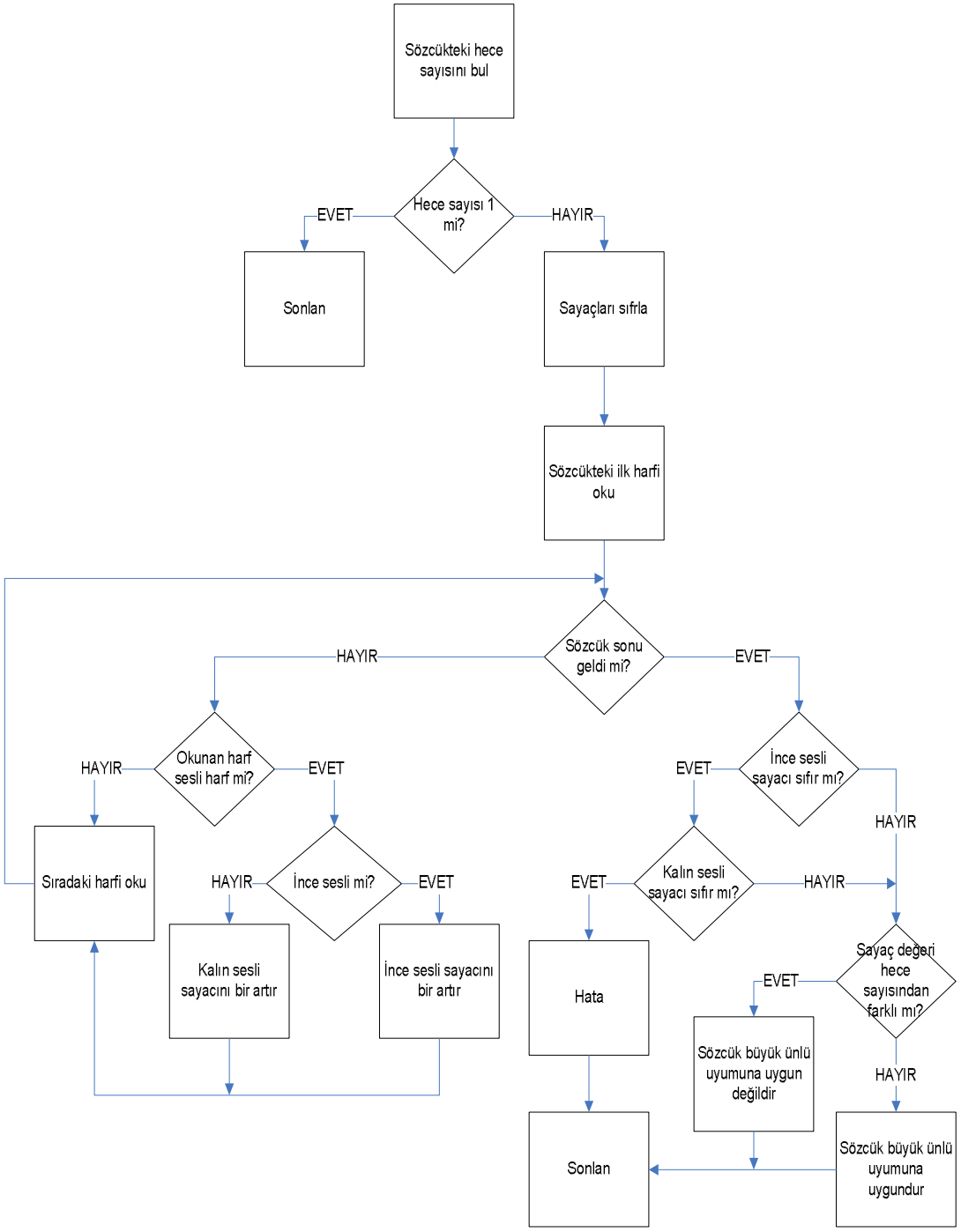
Şekil 3.12 Sözcük içindeki sessiz harf sayısı denetimi için akış diyagramı

3.5.4 Sesli Uyumları

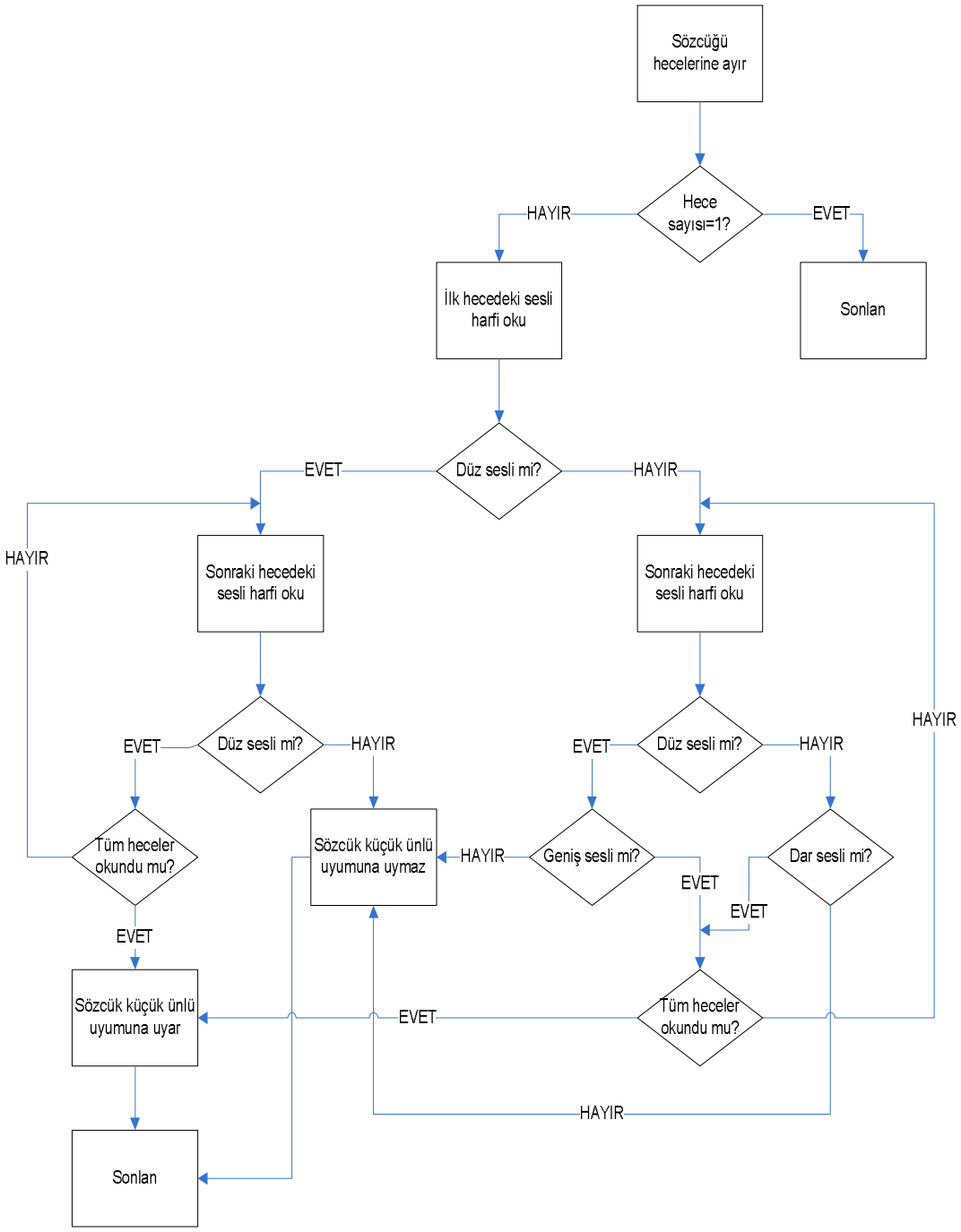
Türkçe kelimelerde uyulması gereken en önemli özelliklerden biri de sesli uyumu kuralıdır. Şekil 2.1 de çizilen sonlu durum makinesi temel alınarak sesli uyumu algoritması çıkarılmıştır. Bu algoritma gereğince ilk olarak sözcük heceleme algoritması ile hecelerine ayrılır. Burada amaç sözcük içindeki sesli harf sayısını bulmaktır. Bölüm 3.2 de anlatılan hece modellerine göre Türkçe bir hecede mutlaka bir sesli harf bulunmak zorundadır. Dolayısıyla Türkçe bir sözcükteki hece sayısı aynı zamanda sözcük içindeki sesli harf sayısını da verir.

Sesli uyumu algoritmasında izlenen yol şu şekildedir. Hece indekslerinden yararlanılarak sözcük içindeki sesli harflerin indekslerine erişilir ve ilk sesli harf okunur. İlk harf kalın sesli ise sonraki hecelerdeki tüm sesli harflerin de kalın sesli olup olmadığı denetlenir. Benzer şekilde ilk harf ince sesli ise sonraki hecelerdeki tüm sesli harflerin ince sesli olup olmadığı denetlenir. Eğer sözcük bu kurala uymuyorsa Türkçe değildir ya da dilimize yabancı dillerden girmiştir. Sözcük bu kurallara uygunsa sözcüğün küçük sesli uyumuna uyup uymadığı denetlenir. Sözcüğün ilk sesli harfi düz seslilerden biri ise sonraki hecelerdeki sesli harfler de düz sesli olmalıdır. Sözcüğün ilk sesli harfi yuvarlak seslilerden biri ise sonraki hecelerdeki sesli harfler ya dar yuvarlak ya da düz geniş olmalıdır. Bu kurala uymayan sözcükler Türkçe değildir ya da dilimize yabancı dillerden girmişlerdir.

Şekil 3.13 büyük sesli uyumu için, Şekil 3.14 ise küçük sesli uyumu için akış diyagramını göstermektedir.



Şekil 3.13 Büyük sesli uyumu akış diyagramı

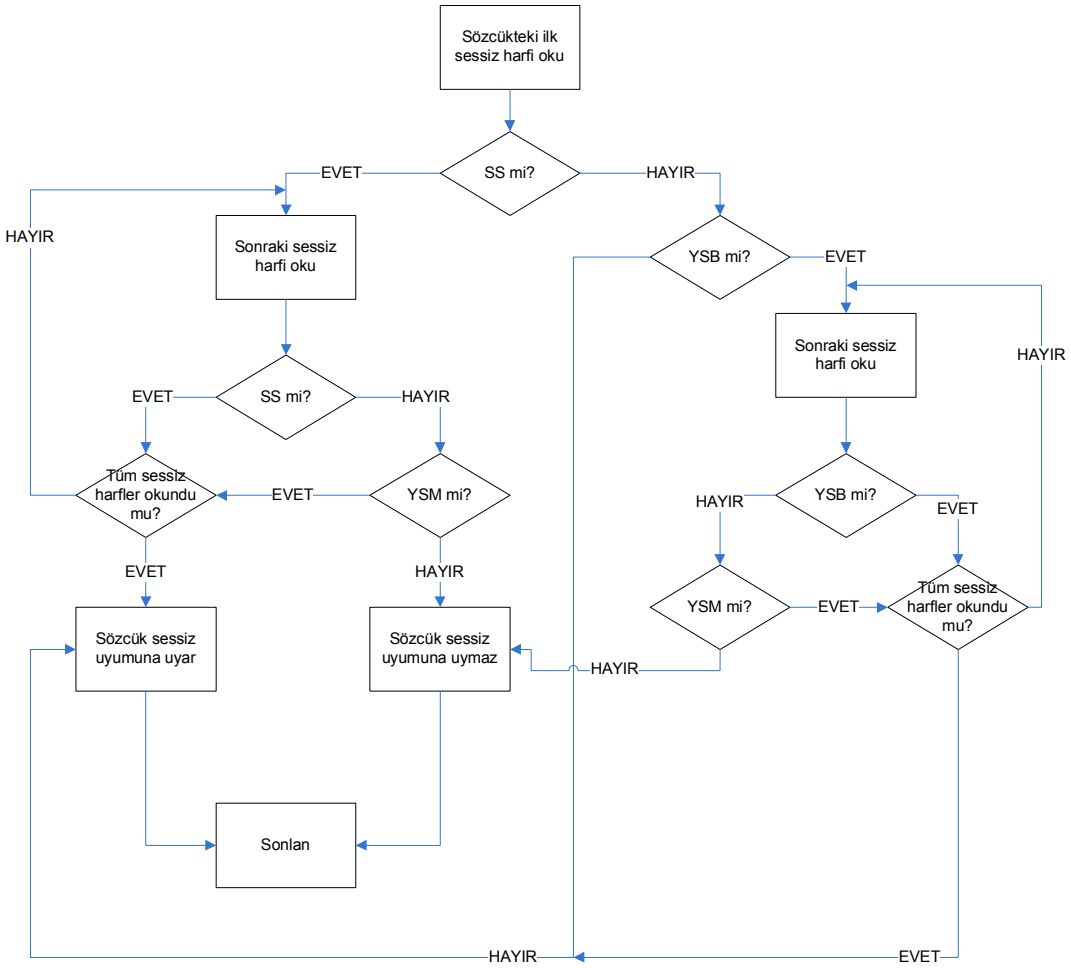


Şekil 3.14 Küçük sesli uyumu akış diyagramı

3.5.5 Sessiz Uyumu

Türkçe sözcüklerde sessizler açısından da bir uyum söz konusudur. Şekil 2.2 deki sonlu durum makinesinden yola çıkılarak sessiz uyumu algoritması gerçeklenmiştir. Buna göre sözcük içindeki sessiz harfler sırasıyla taranmakta ve ele alınan sessiz harfin bir sonraki sessiz harf ile uyumlu olup olmadığının denetimi yapılmaktadır. Buna göre ele alınan sessiz harf sert sessizlerden biriyse ve bir sonraki sessiz harf de

sert sessiz harflerden ya da sert karşılığı bulunmayan yumuşak sessiz harflerden biri değilse sözcük Türkçe değildir ya da yabancı kökenli bir sözcüktür. Ele alınan sessiz harf sert karşılığı bulunan yumuşak sessizlerden biri ise ve bir sonraki sessiz harf sert sessiz ise sözcük türkçe değildir ya da yabancı dillerden dilimize girmiş bir sözcüktür. Şekil 3.15 sessiz uyumu için akış diyagramını göstermektedir.



Şekil 3.15 Sessiz uyumu akış diyagramı

3.6 Ses Bilgisi Açısından Sözcük Denetimi

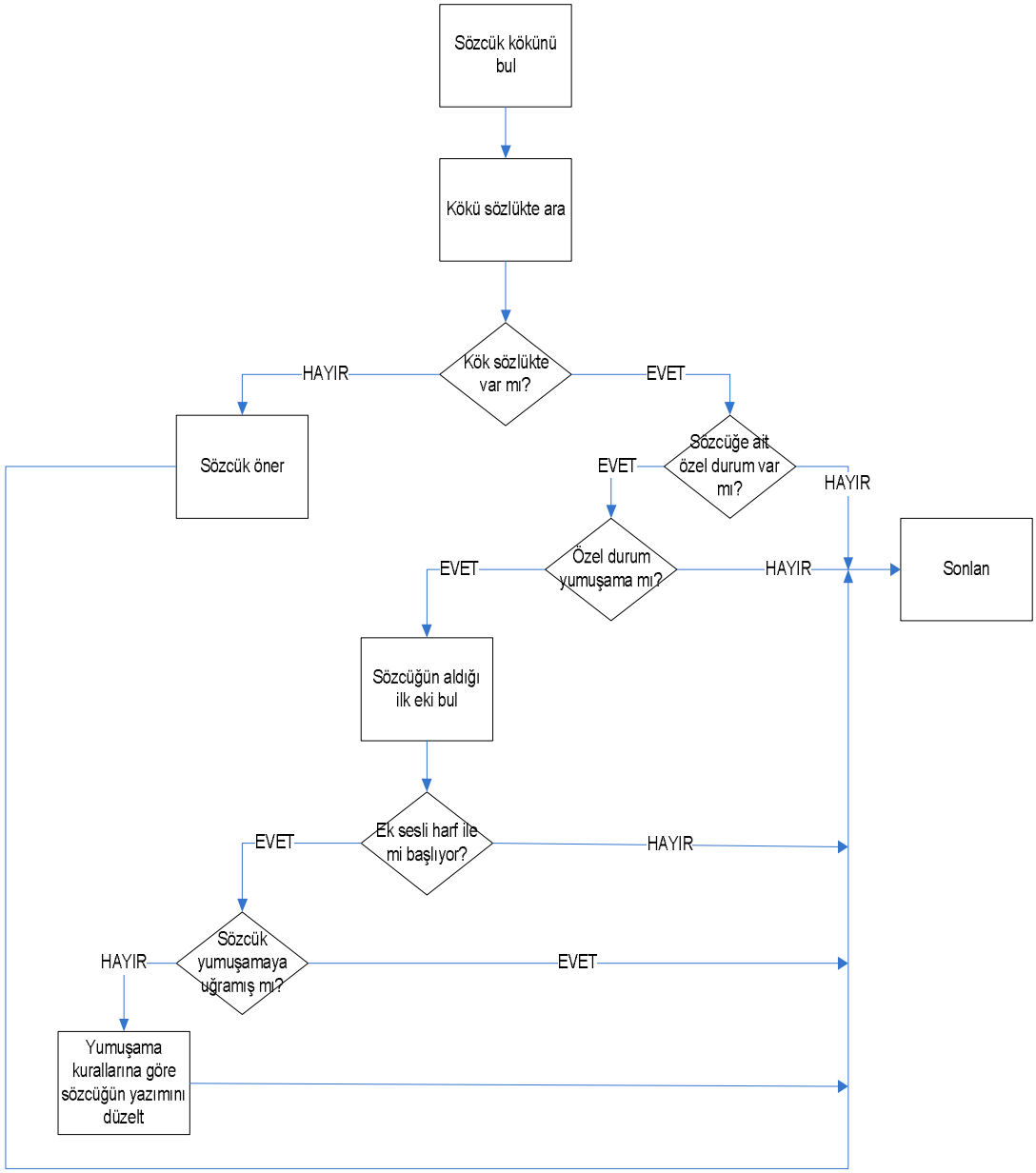
Ses bilgisi açısından Türkçe denetimi için yapılması gereken denetimlerin dışında kalan denetimler sözcük denetleme bölümünde ele alınmaktadır. Bunlar sessiz yumuşaması ve sesli düşmesi durumları için yapılan denetimlerden oluşur.

3.6.1 Sessiz Yumuşaması Denetimi

Bu bölümde yapılan denetim sesli ile başlayan bir ek aldığında sessiz yumuşamasına uğraması gereken bir sözcüğün bu değişime uğrayıp uğramadığının denetlenmesi şeklindedir. Bölüm 3.1 de belirtildiği gibi tez kapsamında kullanılan sözlük içerisinde kelimeler etiketlendiği için bu denetimin yapılmasında sözlükte bulunan etiketlerden yararlanılmaktadır.

Sessiz yumuşaması denetimi için geliştirilen algoritma şu şekilde çalışmaktadır: Girilen sözcüğün ilk olarak kökü bulunur ve bulunan kök sözlükte aranır. Sözlükte bu kök sözcüğe ait etiketlenmiş özel bir durum olup olmadığının denetimi yapılır. Eğer sözcük için özel durum olarak yumuşama etiketi bulunmuyorsa, bu sözcük için sessiz yumuşaması durumu söz konusu değildir. Aksi durumda ise sözcük sessiz yumuşamasına uğrayan özelliğe sahip olduğundan dolayı sözcüğün almış olduğu eke bakılır. Sözcük birden fazla ek almış olabilir, ancak sessiz yumuşaması için bakılması gereken ek sözcüğün almış olduğu ilk ektir. İlk ek sesli harf ile başlayan bir ek değilse denetlenmesi gereken durum oluşmamış demektir. Ancak ilk ek sesli harf ile başlayan bir ek ise bu durumda sözcüğün sessiz yumuşaması kurallarına uygun bir şekilde yumuşamaya uğrayıp uğramadığının denetimi yapılır. Bu denetim sonucuna göre eğer yumuşama Türkçe ses bilgisi kurallarına uygun olarak gerçekleştirilmişse hata durumu oluşmamıştır. Hata durumunun oluştuğu durumda ise Türkçe ses bilgisi kurallarına uygun olarak sessiz yumuşaması işlemi gerçekleştirilir ve böylece yanlış yazılmış olan sözcük düzeltilmiş olur.

Sessiz yumuşaması algoritmasının işleyişi şu örnekle daha iyi açıklanabilir: Girilen sözcük “kitapımdakiler” sözcüğü olmak üzere yukarıda anlatılmış olan işlemler adım adım yürütülür. Öncelikle sözcüğün kökü olan “kitap” sözcüğü bulunur. Bu kök sözlükte aranır ve yumuşama etiketine sahip olduğu bulunur. Yumuşama özel durumuna sahip olduğundan dolayı sözcüğe eklenen ilk ekin sesli harf ile başlayıp başlamadığı denetlenir. Bu ek (-ım) sesli harf ile başladığından dolayı sözcük sonundaki p harfinin Türkçe ses bilgisi kuralları gereğince sessiz yumuşamasına uğraması gerekmektedir. Ancak girilen sözcüğün bu kurala uymadığı için sözcüğün yanlış yazılmış olduğuna karar verilir ve sözcük sonundaki “p” harfinin yerine “b” harfi getirilerek sözcüğün sessiz yumuşaması kuralına uygun olarak doğru yazılması sağlanır. Şekil 3.16 sessiz yumuşaması denetimi için akış diyagramını göstermektedir.



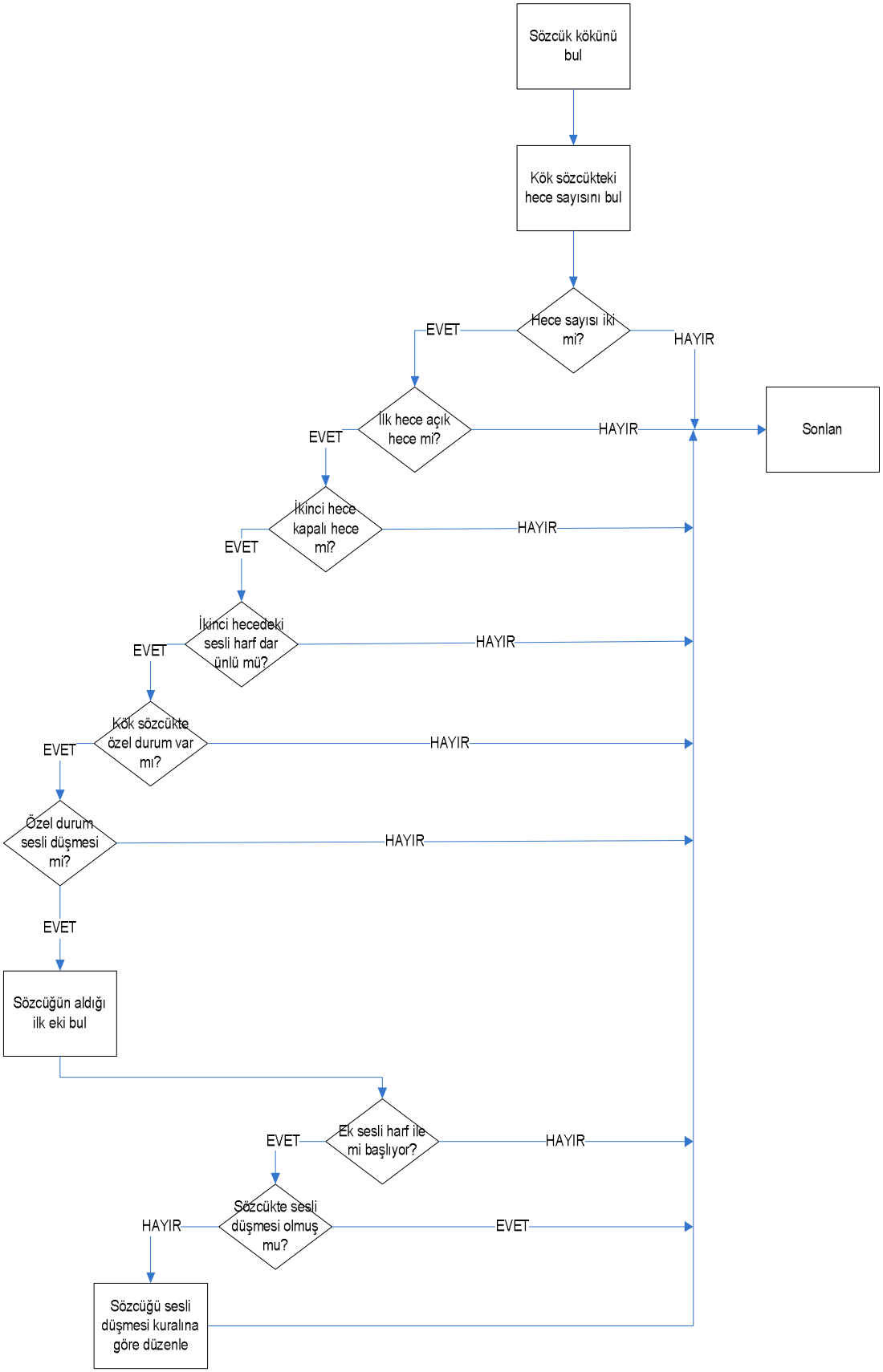
Şekil 3.16 Sessiz yumuşaması algoritması akış diyagramı

3.6.2 Sesli Düşmesi Denetimi

Bu bölümde yapılan denetim sesli ile başlayan bir ek aldığında sesli düşmesine uğraması gereken bir sözcüğün bu değişime uğrayıp uğramadığının denetlenmesi şeklindedir.

Türkçe'de sesli düşmesi iki heceli sözcüklerde görülen bir ses değişimi olduğu için bu denetim için gerçeklenen algoritmada ilk olarak sözcük içindeki hece sayısı bulunur. Hece sayısı iki ise ilk ve ikinci hecelerin açık ya da kapalı hece olup olmadığının denetimi yapılır. Eğer ilk hece açık, ikinci hece kapalı hece ise sözcükte

ses düşmesi durumu olabilir. Türkçe'de sesli düşmesi ikinci hecesinde dar ünlü bulunan sözcüklerde olduğu için bundan sonraki aşamada ikinci hecedeki sesli harfin dar ünlü olup olmadığının denetimi yapılır. Eğer ikinci hecedeki sesli harf dar ünlü ise sözcük kökü alınır ve sözlükte ünlü düşmesi etiketine sahip olup olmadığının denetimi yapılır. Bu denetimin yapılmasındaki amaç sözcüğün ünlü düşmesine kesin olarak uğrayıp uğramayacağını belirleyebilmektir. Bundan önceki adımlarda yapılan denetimler sözcüğün çok büyük bir oranla ünlü düşmesine uğrayabileceğini gösterir; ancak aykırı durumlar söz konusu olduğundan dolayı kesin olarak söyleyemez. Örneğin, "beniz" sözcüğü iki heceli ve ikinci hecesinde dar ünlü bulunduran bir sözcüktür ve sesli ile başlayan bir ek aldığında ikinci hecedeki sesli harf düşer. Oysaki "deniz" sözcüğünde durum farklıdır. Bu sözcük de iki heceli olmasına ve ikinci hecesinde dar ünlü barındırmasına rağmen sesli düşmesi kuralından etkilenmez. Dolayısıyla bunun gibi aykırı durumlarda oluşabilecek sorunları ortadan kaldırmak için sözcük kökünün sözlükteki etiketine bakılması yoluna başvurulur. Sözlükte aranan kök sözcüğe ait sesli düşmesi etiketi varsa, sözcüğün bu kurala uygun olarak yazılıp yazılmadığı denetlenir. Kurala uygun olarak yazılmayan sözcüklerin ikinci hecesindeki sesli harf silinir ve böylece yanlış yazılmış olan sözcük Türkçe sesli düşmesi kuralına uygun olarak düzeltilmiş olur. Şekil 3.17 sesli düşmesi denetimine ilişkin akış diyagramını göstermektedir.



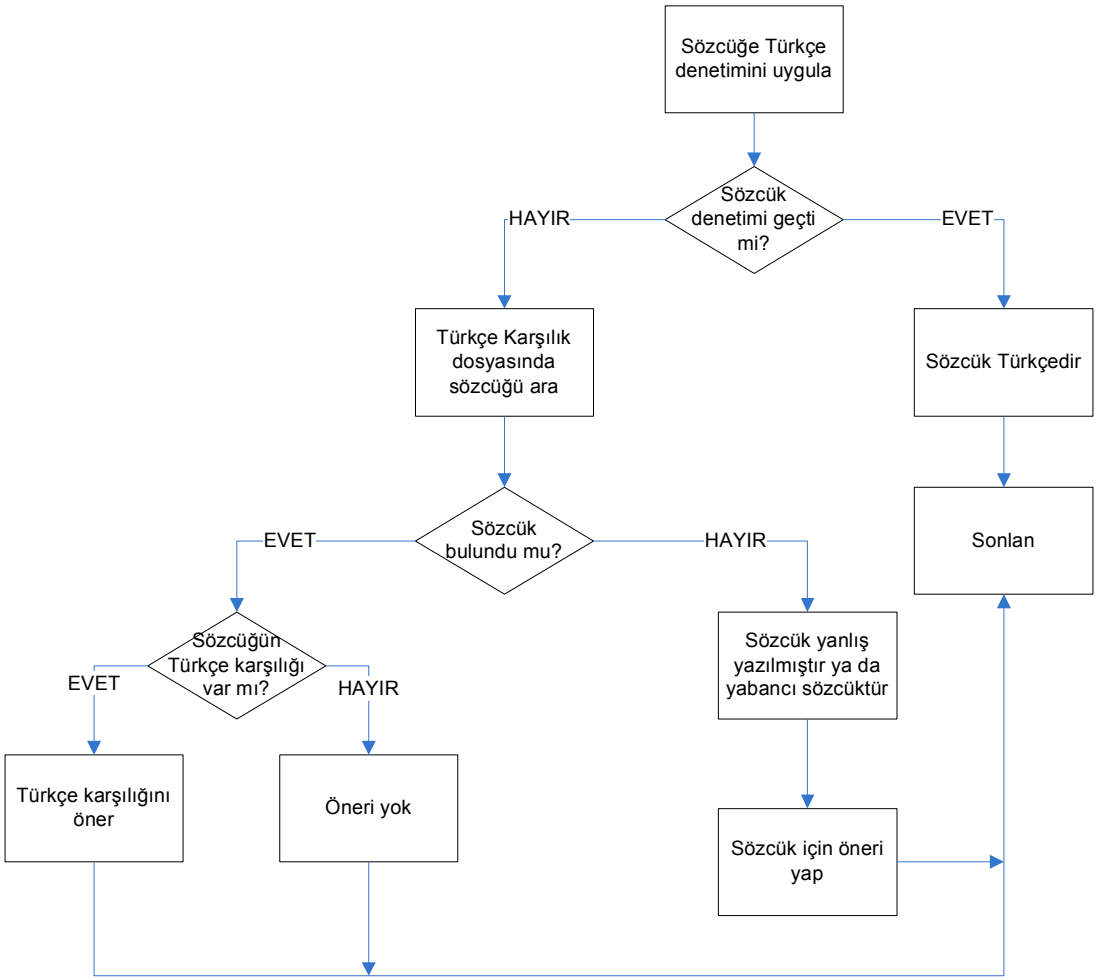
Şekil 3.17 Sesli düşmesi algoritması akış diyagramı

3.7 Sözcük Önerme

Tez kapsamında sözcük önerme işlemi iki farklı durum için yapılmaktadır:

- Sözlükte bulunan sözcükler için önerme
- Sözlükte bulunamayan sözcükler için önerme

Şekil 3.18 sözcük önerme durumuna ilişkin genel akış diyagramını göstermektedir.



Şekil 3.18 Sözcük önerme algoritması akış diyagramı

3.7.1 Sözlükte Bulunan Sözcükler İçin Önerme

Sözlükte bulunan sözcükler için önerme işlemi, Türkçe'ye yabancı dillerden girmiş sözcükler için gerçekleştirilen bir işlemdir. Bu sözcükler TDK güncel sözlüğü temel alınarak oluşturulan ve tez kapsamında kullanılan sözlük içerisinde bulunmaktadır. Ancak bu sözcüklerin Türkçe'de zaten var olan karşılıkları bulunduğundan dolayı

sözcük önerme işlemi yapılırken ilgili sözcüklerin Türkçe karşılıkları kullanıcıya önerilir. Burada amaçlanan dilimize özellikle batı dillerinden girmiş olan sözcüklerin yerine var olan Türkçe karşılıklarının kullanılmasını teşvik etmek ve bu sayede Türkçe karşılıkları bulunan yabancı sözcüklerin özellikle gündelik hayatta kullanımının azaltılarak dilimize yerleşmesinin önlenmesine yardımcı olmaktır. Bu amaçla oluşturulan "Türkçe Karşılık" sözlüğünde Türkçe karşılığı bulunan sözcüklerin yanına "-" işareti konulduktan sonra var olan Türkçe karşılıkları yazılmıştır. Örneğin agresif sözcüğünün sözlükteki görünümü "agresif – saldırgan" şeklindedir.

Önerme işleminin yapılabilmesi için öncelikle girilen sözcüğün Türkçe olmadığının ya da Türkçe'ye yabancı dillerden girmiş bir sözcük olduğunun tespit edilmesi gerekir. Bunun için de Bölüm 3.5 de anlatılan Türkçe denetimi yapılır. Sözcük bu kapsamdaki denetimlerden en az birine uymuyorsa, sözcük kökü Türkçe Karşılık sözlüğünde aranır ve bulunduğunda "-" işaretinin yanında yazan sözcük okunur ve kullanıcıya öneri olarak sunulur. Aranan sözcük bulunamazsa öneri yapılmaz.

Önerme işleminde amaç, ilk paragrafta da belirtildiği gibi Türkçe' ye başka dillerden girmiş olup Türkçe karşılığı bulunan sözcükler için öneri yapmaktır. Örneğin; "analiz" sözcüğü dilimize Fransızcadan girmiş bir sözcüktür ve bu sözcüğün yerine "çözümleme" sözcüğü önerilebilir. Ancak "kitap" sözcüğü için durum farklıdır. Dilimize Arapça'dan girmiş olan bu sözcük Türkçe denetimi esnasında sesli uyumuna uygun olmadığından dolayı Türkçe Karşılık sözlüğünde aranır; fakat bu sözcük dilimize yerleşmiş bir sözcük olduğundan ve bu sözcüğün yerini tutan başka bir sözcük dilimizde bulunmadığından dolayı "kitap" sözcüğü için öneri yapılmaz. Dolayısıyla bu örneklerden de anlaşılacağı üzere, Türkçe denetimini geçemeyen her sözcük için öneri yapılması söz konusu değildir; sadece Türkçe karşılığı bulunan sözcükler için öneri yapılır.

3.7.2 Sözlükte Bulunmayan Sözcükler İçin Önerme

Sözlükte bulunmayan sözcükler için önerme işlemi kullanıcının sözcüğü yanlış yazmış olma ihtimali göz önüne alınarak gerçeklenmiştir. Ancak sözcük doğru yazılmış ise ve sözlükte yer almıyor ise bu sözcüğün Türkçe bir sözcük olmadığı anlamına gelir. Bunun için ilk olarak girilen sözcüğün Türkçe karakterler dışında bir karakter içerip içermediği kontrol edilir. Örneğin x ve w karakterlerini içeren sözcükler Türkçe olamayacağı için bu sözcükler için öneri yapılmaz ve uyarı verilir. Bu karakterleri içermeyen ve sözlükte bulunmayan sözcükler için öneri yapılır.

Sözcük önermede kullanılan kök bulma algoritması Bölüm 3.3.3 de anlatılan kök bulma algoritmasına oldukça benzerdir. Bu kök bulma algoritmasında ağaç üzerinde ilerlerken karşılaşılan tüm aday kökler toplanır. Giriş kelimesi ile ağaç üzerinde ilerlerken oluşturulan kök sözcükler arasında harf uyuşmaması durumu yoktur. Ancak sözcük önerme amacıyla kullanılan kök bulma algoritmasının temeli, girilen sözcük ile ağaç üzerinde ilerlerken oluşturulan sözcük arasındaki farklı harf sayısının en az olmasına dayanır. Dolayısıyla burada hata mesafesinin(farklı harf sayısının) tanımlanan değerden az olması durumunu sağlayan tüm aday kökler ağaç üzerinde ilerlerken toplanır. Örneğin hata mesafesi bir olacak şekilde tanımlanmış ise girilen sözcük ile ağaç üzerinde ilerlerken oluşturulan sözcük arasındaki farklı harf sayısı bir olan tüm sözcükler aday olarak alınır ve kullanıcıya önerilir. Bunu gerçeklemek amacıyla kullanılan algoritma "Damerau-Levenshtein Edit Distance" algoritmasıdır.

Damerau-Levenshtein Edit Distance algoritması Damerau ve Levenshtein' in yazım hataları üzerinde çalışırken oluşturdukları bir algoritmadır. Bu algoritmanın yaygın olarak kullanıldığı alanlar, yazım hatalarının denetimi, konuşma tanıma ve DNA çözümlemesi alanlarıdır.

Damerau-Levenshtein uzaklığı algoritması bir katarı diğer bir katara çevirebilmek için gerekli olan işlem sayısını hesaplayan bir yöntemdir. Burada işlemden kastedilen bir harf ekleme, bir harf çıkarma, bir harfin yerine başka bir harf koyma ya da yan yana olan iki harfin sırasını değiştirmektir. Bir katarı diğerine çevirebilmek için bu dört işlem arasında çeşitli sayıda çözümlemeler olabilir. En az sayıda işlem yaparak oluşturulan çözüm iki sözcük arasındaki uzaklığı verir [21]. Buradan hareketle, x ve y karşılaştırılacak iki katar olmak üzere, Damerau-Levenshtein uzaklığı şu şekilde formülleştirilebilir:

$$DLD(x,y) = \min(i)\ (\#E(i) + \#Ç(i) + \#S(i) + \#T(i))$$

Bu formülde #E(i) harf ekleme sayısını, #Ç(i) harf çıkarma sayısını, #S(i) bir harfin yerine başka bir harf koyma sayısını ve #T(i) yan yana olan iki harfin sırasını değiştirme sayısını göstermektedir. Şekil 3.19 Damerau-Levenshtein Edit Distance algoritmasının sözde kod şeklindeki gösterilimi görülmektedir. Burada str1 ve str2 aralarındaki DLD uzaklığı bulunacak katarları ve lenStr1 ve lenStr2 bu katarların uzunluklarını göstermektedir.

```
int DamerauLevenshteinDistance(char str1[1..lenStr1], char str2[1..lenStr2])
   // d is a table with lenStr1+1 rows and lenStr2+1 columns
   declare int d[0..lenStr1, 0..lenStr2]
   // i and j are used to iterate over str1 and str2
   declare int i, j, cost

   for i from 0 to lenStr1
       d[i, 0] := i
   for j from 1 to lenStr2
       d[0, j] := j

   for i from 1 to lenStr1
       for j from 1 to lenStr2
           if str1[i] = str2[j] then cost := 0
                                else cost := 1
           d[i, j] := minimum(
                                d[i-1, j  ] + 1,     // deletion
                                d[i  , j-1] + 1,     // insertion
                                d[i-1, j-1] + cost   // substitution
                              )
           if(i > 1 and j > 1 and str1[i] = str2[j-1] and str1[i-1] = str2[j]) then
               d[i, j] := minimum(
                                d[i, j],
                                d[i-2, j-2] + cost   // transposition
                               )


   return d[lenStr1, lenStr2]
```

Şekil 3.19 DLD algoritması sözde kod gösterilimi [22]

DLD algoritmasının kullanımını bir örnekle açıklamak yararlı olacaktır. str1 = “OT” ve str2 = “OST” olmak üzere bu iki katar arasındaki DLD uzaklığı;

DLD (str1 , str2) = 1 dir. Çünkü str1 katarının str2 katarına dönüşmesi için gereken işlem bir karakter çıkarma işlemidir.

str1 = “OST” ve str2 = “TO” olmak üzere bu iki katar arasındaki DLD uzaklığı;

DLD (str1 , str2) = 3 tür. Çünkü str1 katarından str2 katarını elde etmek için bir çıkarma işlemi yapılmalıdır ve buna ek olarak yan yana olan iki harfin sırası değiştirilmelidir. Örneklerden de görüldüğü gibi katarlar arasındaki farklılıklar çoğaldıkça DLD uzaklığı artar.

4. TEZİN BAŞARIMI VE SINAMA SONUÇLARI

Tezin başarımını ölçmek amacıyla yapılan sınama işlemi için farklı kaynaklar kullanılmıştır. Bu kaynakların oluşturulmasında e-kitaplar, makaleler, masallar ve bu tez dokümanı kullanılmıştır.

Sınama işlemi iki farklı şekilde gerçekleştirilmiştir. İlk sınama işlemi için seçilen kaynak üzerinde bilerek hatalar oluşturulmuş ve sistemin bu hataların ne kadarını algıladığı ölçülmeye çalışılmıştır. Bu sınama işlemi Bölüm 4.1 de detaylı olarak anlatılmıştır ve sınama sonuçlarına göre sistemin başarımı hesaplanmıştır.

İkinci sınama işleminde ise farklı kaynaklardan oluşan sınama verisi üzerinde herhangi bir değişiklik yapmadan sınama işlemi gerçekleştirilmiştir ve sistemin başarımı hesaplanmıştır. Bu kısım Bölüm 4.2 de ele alınmıştır.

4.1 Bilerek Hata Oluşturma Yöntemiyle Yazılımın Sınanması

Bu sınama yönteminde seçilen kaynak üzerinde bilerek hatalar oluşturulmuştur. Yazılımın sınanması için Prof. Dr. Eşref Adalı'nın "Mikroişlemciler Mikrobilgisayarlar" kitabı seçilmiştir. Bu kaynağın kullanılmasının nedeni, kaynağın Türkçe açısından doğruluğuna güvenilebilir bir kaynak olmasıdır. Ayrıca kaynağın İngilizce terimler içeriyor olması da bir diğer tercih nedenidir.

4.1.1 Sınama Verisi Üzerinde Hata Oluşturulması

Sistem, sınama verisi olarak text halindeki dosyaları giriş olarak kabul etmektedir. Bu nedenle, ilk olarak sınama verisi olan kaynak txt uzantılı dosya şekline dönüştürülmüştür. Bundan sonraki aşama ise sınama verisi üzerinde hatalar oluşturmaktır. Oluşturulan hatalar;

- Kök sözcüğün yanlış yazılması
- Eklerin kök sözcüğe yanlış ulanması
- Sözcüklerdeki Türkçe ses kurallarına uygunluğun bozulması

şeklinde gerçeklenmiştir. Oluşturulan hatalara örnek olarak, "buyrukun", "gerekmektirmaktaydı", "uclarına", "bilgsayarlarda" sözcükleri verilebilir.

Bu şekilde oluşturulan hataları içeren sınama verisi sisteme girdi olarak verilmiş ve sistemin başarımı ölçülmüştür.

4.1.2 Sınama Sonuçları

Sınama verisi üzerinde toplam 304 tane sözcükte hata oluşturulmuştur. Oluşturulan hatalara ilişkin istatistiki veri Tablo 4.1 de verilmiştir. Tablodan da görüleceği gibi 304 sözcük üzerinde oluşturulan hataların 281 tanesi sistem tarafından algılanmıştır. Sistem hatalı olduğunu algıladığı 281 sözcüğün 268 tanesi için öneri üretebilmiştir. Sistemin üretmiş olduğu 268 önerinin 253 tanesi gerçekte yazılmak istenen sözcüğü de içermektedir.

Tablo 4.1 Yazılım sınama sonuçları

	Oluşturulan hata sayısı	Sistemin algılayabildiği hata sayısı	Sistemin öneri ürettiği hatalı sözcük sayısı	Üretilen önerilerin gerçek sözcüğü içeren miktarı
Sözcük sayısı	304	281	268	253

Tablodan elde edilen verilerle, sistemin hatalı sözcükleri bulmada %92, hatalı sözcükler için öneri yapmada %95, yapılan önerilerin gerçek sözcüğü içermesinde %94 başarıma sahip olduğu söylenebilir.

Sistem bilerek oluşturulan 304 hatanın dışında 309 sözcüğü daha hatalı olarak yorumlamıştır. Bu 309 sözcükten 166 sı gerçekten yanlış yazılmış sözcüklerdir. 107 tanesi İngilizce sözcüktür. Kalan 36 sözcüğün 20 tanesi TDK sözlüğünde yer almamaktadır (bipolar, osilatör, diyot vb). 16 sözcük ise sistemin biçim birimsel çözümleme hatalarından kaynaklanarak bulmuş olduğu hatalı sözcükleri kapsamaktadır.

4.1.3 Sistemin Başarımının Ölçülmesi

Başarım ölçümü elle işaretlenmiş olan sınama verisi üzerinde gerçekleştirilmiştir. Başarım ölçümünde kullanılan ölçütler şunlardır:

- Anma (Recall)

- Keskinlik (Precision)
- F ölçüsü (F measure)

Anma ve keskinlik ölçütlerinin hesaplanması Şekil 4.1 deki karmaşıklık matrisi üzerinden gösterilebilir.

	ÖNGÖRÜLEN SINIF		
DOĞRU SINIF		+1	-1
	+1	GD	YY
	-1	YD	GY

Şekil 4.1 Karmaşıklık matrisi

Matris üzerindeki ifadelerin açılımı şu şekildedir:

GD: Gerçek Doğru

YY: Yanlış Yanlış

YD: Yanlış Doğru

GY: Gerçek Yanlış

Bu ifadelerden hareketle, anma ve keskinlik ölçütleri şu şekilde elde edilir:

Anma (Doğru Pozitif Oranı) = Doğru sınıflandırılmış pozitif örnek sayısı / Pozitif örneklerin sayısı = GD / (GD + YY)

Keskinlik = Doğru sınıflandırılmış pozitif örnek sayısı / Pozitif sınıflandırılmış örnek sayısı = GD / (GD + YD)

F ölçüsü ise anma ve keskinlik değerlerinin harmonik ortalaması alınarak elde edilir.
F ölçüsü = 2 * Anma * Keskinlik / (Anma + Keskinlik)

F ölçüsü 0 – 1 aralığında değer alır. F ölçüsünün değer aralığının akademik notlandırma şeklinde olduğu söylenebilir. Buna göre 0.9 – 1 aralığındaki değerler mükemmel (A), 0.8 – 0.9 aralığındaki değerler iyi (B), 0.7 – 0,8 aralığındaki değerler orta (C), 0.6 – 0.7 aralığındaki değerler zayıf (D) ve 0.6 – 0 aralığındaki değerler başarısız (F) olarak nitelendirilebilir.

Sistemin başarımının ölçülmesi için ilk olarak anma ve keskinlik değerleri hesaplanmıştır. Daha sonra ise anma ve keskinlik değerlerinden F ölçüsünün değeri elde edilmiştir.

Sınama verisinde bilerek oluşturulan hatalı sözcük sayısı 304 tanedir. Sınama verisinde var olan toplam hatalı sözcük sayısı 470 tanedir. Sistemin bulmuş olduğu toplam hatalı sözcük sayısı ise 590 tanedir. Bulunan hataların içinde gerçekten hatalı olan sözcük sayısı 447 tanedir. Bu verilerden hareketle, Anma değeri 447 / 470 ≈ 0.951 ve Keskinlik değeri 447 / 590 ≈ 0.757 olarak hesaplanır. F ölçüsü ise 2 * 0.951 * 0.757 / (0.951 + 0.757) ≈ 0.842 olarak bulunur. Bulunan F ölçüsü değeri yukarıda belirtilen değer aralıklarından "iyi" aralığına karşılık gelmektedir; dolayısıyla bu sonuçtan hareketle sistemin başarımının yüksek olduğunu söylemek mümkündür.

4.2 Sınama İşleminin Farklı Sınama Verileri Üzerinde Gerçekleştirilmesi

Sınama işleminin farklı kaynaklar üzerindeki başarımını ölçmek için sınama işlemi farklı türden kaynaklar üzerinde de gerçekleştirilmiştir. Bu amaçla toplanan sınama kaynakları arasında, makaleler, elektronik kitaplar, masallar ve bu tez dokumanı yer almaktadır. Makaleler iki farklı gazetenin yedi yazarına ait yazılardan oluşacak şekilde seçilmiştir. Her bir yazara ait ortalama beş makale sınama verisi olarak seçilmiştir. Masallar internet ortamından alınmıştır ve toplam on bir masal sınama verisine dahil edilmiştir. Elektronik kitap olarak Can Dündar'ın "Büyülü Fener", Elif Şafak'ın "Pinhan", Erdal Atabek'in "Kırmızı Işıkta Yürümek", Reşat Nuri Güntekin'in "Yaprak Dökümü", Dan Brown'ın "Dijital Kale" adlı eserleri sınama verisi olarak seçilmiştir.

Sınama verisi olarak seçilen bu kaynaklar üzerinde hiçbir elle işaretleme yapılmamış; bilerek hata yaratılmamıştır. Çünkü bu bölümde yapılan sınama işlemindeki amaç, seçilen eserlerde var olan yazım hatalarının denetimini yapmaktır. Bilerek hata oluşturarak sınama yapma işlemi Bölüm 4.1 de ele alınıp incelendiği için bu bölümde tekrar ele alınmamıştır.

4.2.1 Sınama Sonuçları

Her bir kaynak için yapılan sınama işlemlerinin sonuçları MS Word ile karşılaştırmalı olarak verilmiştir. Bu sayede yazılımın başarımı hakkında bilgi sahibi olunabileceği düşünülmüştür.

Makaleler için yapılan sınama işleminin sonuçları MS Word ile karşılaştırmalı olarak Tablo 4.2 de verilmiştir.

Tablo 4.2 Makaleler için yapılan sınama sonuçları

	Makalelerde bulunan gerçek hatalı sözcük sayısı	Doğru olarak bulunan yanlış sözcük sayısı	Yanlış olarak bulunan yanlış sözcük sayısı
Yazılımın sonuçları	62	59	28
MS Word sonuçları	62	45	16

Tablo 4.3 masallar üzerinde yapılan sınama sonuçlarını MS Word ile karşılaştırmalı olarak göstermektedir.

Tablo 4.3 Masallar için yapılan sınama sonuçları

	Masallarda bulunan gerçek hatalı sözcük sayısı	Doğru olarak bulunan yanlış sözcük sayısı	Yanlış olarak bulunan yanlış sözcük sayısı
Yazılımın sonuçları	61	59	16
MS Word sonuçları	61	57	7

Tablo 4.4 elektronik kitaplar üzerinde yapılan sınama sonuçlarını MS Word ile karşılaştırmalı olarak göstermektedir.

Tablo 4.4 Elektronik kitaplar için yapılan sınama sonuçları

	E-kitaplarda bulunan gerçek hatalı sözcük sayısı	Doğru olarak bulunan yanlış sözcük sayısı	Yanlış olarak bulunan yanlış sözcük sayısı
Yazılımın sonuçları	621	608	143
MS Word sonuçları	621	586	96

Tablo 4.5 ise bu tez dokümanının sınama verisi olarak kullanılması sonucu ortaya çıkan sınama sonuçlarını göstermektedir.

Tablo 4.5 Tez dokümanı için yapılan sınama sonuçları

	Tez dokümanında bulunan gerçek hatalı sözcük sayısı	Doğru olarak bulunan yanlış sözcük sayısı	Yanlış olarak bulunan yanlış sözcük sayısı
Yazılımın sonuçları	31	27	11
MS Word sonuçları	31	24	4

Tablolardaki sonuçlardan görüldüğü gibi, tez kapsamında gerçeklenen hata bulma yöntemi MS Word ile kıyaslandığında daha başarılıdır; ancak yanlış olmadığı halde sistemin yanlış olarak algıladığı sözcük sayısının da fazla olduğu tablolardan görülmektedir.

4.2.2 Sistemin Bulduğu Hataların Dağılımı

Sistemin doğru olarak algılayabildiği hatalı sözcüklerin dağılımı Tablo 4.6 da görülmektedir.

Tablo 4.6 Sistemin bulduğu hataların dağılımı

	Kök sözcüğün yanlış yazılması durumu	Ekin köke yanlış ulanması durumu	Ayrı yazılması gereken sözcüklerin bitişik yazılması durumu
Sözcük sayısı	297	167	289

Sistemin doğru olarak algılayabildiği yazım hatalarının %39.44 ü kök sözcüğün yanlış yazılmasından kaynanklanan hatalardan oluşmaktadır. Bu duruma örnek olarak "sayfasının" yerine "safyasının" yazılmış olması, "pencereden" yerine "pncereden" yazılmış olması, "gibi" yerine "giib" yazılmış olması, "çocuklar" yerine "çoçuklar" yazılmış olması verilebilir.

Sistemin doğru olarak algılayabildiği yazım hatalarının %22.17 si eklerin köke yanlış ulanması sonucunda oluşan hatalardır. Bu duruma örnek olarak "hafifçe" yerine "hafifce" yazılması, "arıyor" yerine "arayor" yazılması, "fincanıyla" yerine "fincanıyle" yazılması, "tasarlarken" yerine "tasarlerken" yazılması verilebilir.

Sistemin doğru olarak algılayabildiği yazım hatalarının %38.37 si ayrı yazılması gereken sözcüklerin bitişik yazılmasından kaynaklanan hatalardır. Bu duruma örnek

olarak “can sıkıntısından” yerine “cansıkıntısından” yazlması, “çekip çevrilmesinin” yerine “çekipçevrilmesinin” yazılması, “ön yargı” yerine “önyargı” yazılması verilebilir.

4.2.3 Sistemin Başarımının Ölçülmesi

Sistemin başarımın ölçülmesi için Bölüm 4.1.1 deki tablolarda görülen sınama sonuçları kullanılmıştır. Sınamada kullanılan farklı kaynaklar üzerindeki toplam hatalı sözcük sayısı 775 tanedir. Sistemin doğru olarak algılayabildiği hatalı sözcük sayısı 753 tanedir. Sistemin hatalı olmadığı halde hatalı olarak algıladığı toplam sözcük sayısı 198 tanedir. Bu verilerden hareketle, Anma değeri 753 / 775 ≈ 0.971 ve Keskinlik değeri 753 / 951 ≈ 0.79 olarak hesaplanır. F ölçüsü ise 2 * 0.971 * 0.79 / (0.971 + 0.79) ≈ 0.87 olarak bulunur. Bulunan değer “iyi” değer aralığına krşılık düşmektedir, dolayısıyla sistemin başarımının farklı kaynaklar üzerinde de iyi sonuç verdiği söylenebilir.

5. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında, Türkçe metinlerdeki yazım hatalarının denetlenmesini ve düzeltilmesini gerçekleştiren bir yazılım geliştirilmiştir. Geliştirilen yazılım yanlış yazılmış sözcükler için öneri yapmayı da sağlamaktadır.

Yazım hatalarının denetlenmesi amacıyla, girilen metindeki her bir sözcüğün öncelikle hecelenebilir olup olmadığı denetlenir. Bu amaçla geliştirilen heceleme algoritması, her sözcük için çalıştırılır. Türkçe hece yapılarına uygun olmayan sözcükler için bu algoritma çözüm üretemeyeceğinden sözcüğün hecelenmesi gerçekleştirilemez. Hecelenemeyen sözcükler doğrudan yabancı sözcük olarak algılanır ve sistem kullanıcıya kendi oluşturacağı bir sözlüğe bu yabancı sözcükleri ekleme olanağı verir.

Hecelenebilen sözcüklerin Türkçe ses kurallarına uygun olup olmadığının denetimi yapılır. Türkçe ses kurallarından en az birine uygun olmayan sözcüğün Türkçe olmama olasılığı yüksektir. Bu sözcükler için sözlükte arama yapılır. Sözlükte bulunan sözcükler biçim birimsel açıdan çözümlemesi yapılabilen sözcüklerdir. Bu sözcükler Türkçe'ye yabancı dillerden girmiş sözcüklerdir. Sözlükte Türkçe karşılıkları bulunan sözcükler için var olan Türkçe karşılıkları önerilir.

Biçim birimsel çözümlemesi yapılamayan sözcükler Türkçe değildir ya da yanlış yazılmıştır. Bu sözcükler için öneri yapılır. Öneri yapmak için kullanılan algoritmada Damerau-Levenshtein Edit Distance yöntemi kullanılır. Aranan sözcüğe en yakın uzaklığa sahip olan sözcük ya da sözcükler öneri olarak kullanıcıya sunulur.

Girilen metinler için sessiz yumuşaması ve sesli düşmesi durumları ile ilgili denetimler de yapılmaktadır. Bu değişimlere uğraması gerektiği halde uğramamış olan sözcükler için düzeltme işlemi gerçekleştirilir ve sözcük olması gereken hale dönüştürülür.

Bu tez çalışması kapsamında girilen bir Türkçe metindeki yazım hatalarının denetlenmesi için heceleme yöntemi, Türkçe ses bilgisi kurallarına uygunluğun denetlenmesi yöntemi ve biçim birimsel çözümleme yöntemi kullanılmıştır. Amaç, sözcük bazında yazım denetimi yapmaktır. Gelecek çalışmalar için öneriler cümle bazında anlamsal denetim yapmaktır.

KAYNAKLAR

[1] Doğal Dil İşleme,

http://www.elektrik.gen.tr/content/view/104/61/, 12 Nisan 2008.

[2] **Eryiğit, G.,** 2006. Türkçe'nin Bağlılık Ayrıştırması, *Doktora Tezi*, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.

[3] **Bozşahin, H. C. and Oflazer, K.,** 1994**.** Turkish Natural Language Processing Initiative: An Overview, In *Proceedings of Third Turkish Symposium On Artificial Intelligence,* Middle East Technical University, Ankara, Turkey.

[4] **Kodydek, G.**, 2000. A Word Analysis System for German Hyphenation, Full Text Search, and Spell Checking, With Regard To The Latest Reform Of German Orthography, In *Proceedings of Third International Workshop On Text, Speech and Dialogu,* Brno, Czech Republic, September 2000, p. 39-44.

[5] **Starlander, M. and Popescu-Belis, A.**, 2002. Corpus-based Evaluation Of A French Spelling and Grammer Checker, In *Proceedings of Third International Conference On Language Resources And Evaluation,* Las Palmas, Canary Islands, Spain, p. 268-274.

[6] **Dembitz, S., Knezevich, P. and Sokele, M.**, 2004. Developing A Spell Checker As An Expert System, *Journal Of Computing And Information Technology*, 04, 285-291.

[7] **Dembitz, S., Knezevich, P. and Sokele, M.**, 1998. Hascheck – the Croatian Academic Spelling Checker, In *Proceedings of 18th Annual International Conference of The British Computer Society Specialist Group on Expert System,* Cambridge, UK, December 1998, p. 184-198.

[8] **Dhanabalan, T., Parthasarathi, R and Geetha, T. V.,** 2003. Tamil Spell Checker, *Tamil Internet 2003*, Chennai, Tamilnadu, India.

[9] **Oflazer, K. ve Solak, A.**, 1992. Parsing Agglutinative Word Structures And Its Application to Spelling Checking for Turkish, In *Proceedings of the 15th International Conference On Computational Linguistics,* Nantes, France, August 23-28, p. 39-45.

[10] **Solak, A.**, 1991. Design And Implementation of A Spelling Checker For Turkish, *M.S. Thesis*, Bilkent University, Ankara.

[11] **Güzey, C. ve Oflazer, K.**, 1994. Spelling Correction in Agglutinative Languages, *Bilkent University Department Of Computer Engineering and Information Systems Technical Report*, BU-CEIS-94-01, Ankara, Turkey.

[12] **Oflazer, K.**, 1993. Two-level Description Of Turkish Morphology, In *Proceedings of the Sixth Conference Of The Europen Chapter Of The Assotiation For Computational Linguistics,* Utrecht, Netherlands, April 1993.

[13] Detecting Misspelled Words in Turkish Text Using Syllable n-Gram Frequencies,

http://www.springerlink.com/content/440vt41p20550n28/, 01 Mayıs 2008.

[14] **Banguoğlu, T.**, 2000. Türkçe'nin Grameri, Türk Dil Kurumu Yayınları:528, Ankara.

[15] **Hengirmen, M.**, 2002. Türkçe Dilbilgisi, Engin Yayıncılık, Ankara.

[16] **Adalı, E ve Büyükkuşçu, İ.**, 2006. Heceleme Yöntemiyle Kök Sözcük Üretme, *Türkiye Bilişim Vakfı Bilgisayar Bilimleri Ve Mühendisliği Dergisi*, 02, 25-29.

[17] **Aksu, B., Gemalmaz, E. ve Tariktaroğlu, A.**, 2004. Türkçe Sözlük'ün Ters Dizimi, Türk Dil Kurumu Yayınları:845, Ankara.

[18] CL ile Türkçe Sözcük Heceleme Programı,
http://mlblog.osdir.com/lisp.region.turkey/2006-12/index.shtml,10 Nisan 2008.

[19] Zemberek Kütüphanesi,

https://zemberek.dev.java.net/java_kullanim.html, 20 Şubat 2008

[20] Directed Acyclic Word Graph,

http://www.wutka.com/dawg.html, 28 Nisan 2008.

[21] **Gregory, V. B.,** 2007. Spelling-Error Tolerant, Order-Independent Pass-Phrases via The Damerau-Levenshtein String-Edit Distance, *2007 Australasian Information Security Workshop Privacy Enhancing Technologies*, Ballarat, Australia.

[22] **Levenshtein Distance,** http://en.wikipedia.org/wiki/Levenshtein_distance

EK A. TÜRKÇE SÖZCÜKLERE EKLENEN ÇEKİM EKLERİ

İsim Soylu Sözcüklere Eklenen Çekim Ekleri	
Ek Tanımı	**Örnek**
Çoğul eki	okullar
1. tekil şahıs iyelik eki	okulum
2. tekil şahıs iyelik eki	okulun
3. tekil şahıs iyelik eki	okulu
1. çoğul şahıs iyelik eki	okulumuz
2. çoğul şahıs iyelik eki	okulunuz
3. çoğul şahıs iyelik eki	okulları
-i hal eki	okulu
-e hal eki	okula
-de hal eki	okulda
-den hal eki	okuldan
Tamlama eki	okulun
Araç-Birliktelik eki	okulla
İlgi eki	okuldaki

Eylem Soylu Sözcüklere Eklenen Çekim Ekleri	
Ek Tanımı	**Örnek**
Olumsuzluk eki	gelme
Geniş zaman eki	gelir
Şimdiki zaman eki	geliyor
Dili geçmiş zaman eki	geldi
Mişli geçmiş zaman eki	gelmiş
Gelecek zaman eki	gelecek
İstek kipi eki	geleyim
Dilek-şart kipi eki	gelse
Gereklilik kipi eki	gelmeli
1. şahıs eki	geliyorum
2. şahıs eki	geliyorsun
3. şahıs eki	geliyor
1. çoğul şahıs eki	geliyoruz
2. çoğul şahıs eki	geliyorsunuz
3. çoğul şahıs eki	geliyorlar
Yeterlik eki	gelebil
Tezlik eki	gidiver
Sürerlik eki	gidedur, yapagör, duyagel, alıkoy, şaşakal
Yaklaşma eki	düşeyaz

EK B. TÜRKÇE SÖZCÜKLERE EKLENEN YAPIM EKLERİ

İsimden İsim Yapan Ekler	
Ek	**Örnek**
-lık	gözlük, kulaklık
-lı	ağaçlı, şehirli
-sız	susuz, parasız
-cü	yolcu, simitçi
-ce	Türkçe, İngilizce
-daş	yoldaş, vatandaş
-üncü	üçüncü, beşinci
-ımsı	acımsı, ekşimsi
-cil	bencil, insancıl
-şın	sarışın
-sal	kadınsal
-ıt	yaşıt
-cağız	kızcağız, çocukcağız
-cık	azıcık, küçücük
-tı	cıvıltı, horultu

İsimden Eylem Yapan Ekler	
Ek	**Örnek**
-la	sula, uğurla
-al	azal, daral
-l	sıvril
-a	kana, yaşa
-ar	morar, yaşar
-da	fısılda, horulda
-at	gözet
-ık	gecik
-ımsa	benimse, küçümse
-sa	susa, önemse

Eylemden İsim Yapan Ekler	
Ek	**Örnek**
-ca	düşünce, eğlence
-ecek	giyecek, silecek
-ak	kaçak, durak
-ga	bölge, süpürge
-gan	çalışkan, kaygan
-gı	sevgi, askı
-gıç	dalgıç, bilgiç
-gın	yorgun, bitkin
-ı	yazı, koku
-ıcı	yapıcı, satıcı
-ık	kesik, göçük
-ım	seçim, ölüm
-ın	yığın, ekin
-ınç	sevinç, gülünç
-ıntı	kesinti, döküntü
-ır	gelir, okur
-ış	yürüyüş, oturuş
-ıt	yakıt, ölçüt
-tı	belirti, kızartı
-a	sapa
-ağan	durağan, olağan

-alak	yatalak
-amak	basamak, kaçamak
-anak	seçenek, görenek
-aç	sayaç, süreç
-gaç	utangaç, süzgeç
-maca	bulmaca, çekmece
-man	danışman, seçmen
-mık	kıymık, kusmuk
-ma	dokuma, görüşme
-mak	yemek, çakmak

Eylemden Eylem Yapan Ekler	
Ek	**Örnek**
-dır	güldür, koştur
-ala	kovala, silkele
-er	gider, çıkar
-ımsa	anımsa, gülümse
-ın	gezin, sevin
-ır	kaçır, içir
-ış	görüş, gülüş
-ıt	korkut, üşüt
-ıl	kırıl, satıl

ÖZGEÇMİŞ

Aynur Delibaş, 18 Aralık 1982 tarihinde Bursa'da dünyaya gelmiştir. İstanbul Teknik Üniversitesi Elektrik-Elektronik Fakültesi Bilgisayar Mühendisliği Bölümü'nde 2000 yılında başladığı lisans eğitimini 2005 yılında tamamlayarak Bilgisayar Mühendisi ünvanını almıştır. 2005 yılında çalışmaya başlamış olduğu Siemens A.Ş. bünyesinde halen AR-GE mühendisi olarak çalışmaktadır.