

TÜRKÇE DİLİNİN MORFOLOJİK İNCELEMESİ VE BİÇİMSEL OLARAK TANIMLANMASI

BSM 401 BİLGİSAYAR MÜHENDİSLİĞİ TASARIMI

**Egemen Zeytinci
Taşkın Temiz**

Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ

Tez Danışmanı : Prof. Dr. Nejat Yumuşak

ARALIK 2015

İÇİNDEKİLER

İçindekiler	ii
ÖZET	iii
BÖLÜM 1. GİRİŞ	1
BÖLÜM 2. BİÇİMBİLİMSEL ÇÖZÜMLEME VE TÜRKÇE’NİN BİÇİMBİLİMSEL YAPISI	2
BÖLÜM 3. BİÇİMBİLİMSEL ÇÖZÜMLEMENİN KULLANIM ALANLARI	6
BÖLÜM 4. BİÇİMBİLİMSEL ÇÖZÜMLEME ÇIKTILARI	8
4.1. Gereksiz Sonuçların Filtrelenmesi	8
4.2. Olası Çözümlerin Genişletilmesi	9
BÖLÜM 5. BİÇİMBİLİMSEL ÇÖZÜMLEME YAKLAŞIMLARI	11
BÖLÜM 6. TÜRKÇE’NİN YAPISI	14
6.1. Çekim Kümeleri	14
6.2. Eklerin Diziliş Kuralları	17
BÖLÜM 7. TÜRKÇE DİLİNİN BİÇİMSEL YAPISINDAN YARARLANILARAK YAPILAN PROGRAMLAR	36
7.1. Koşullu Rastgele Alanlar ile Türkçe Haber Metinlerinin Etiketlenmesi	36
7.1.1. Koşullu Rastgele Alanlar	36
7.1.2. Metinlerin Önişlenmesi	37
7.1.3. Metinlerin İşlenmesi	38
7.1.4. Niteliklerin Belirlenmesi	38
7.1.4.1. Kural Tabanlı Nitelikler	38
7.1.4.2. Özne ve Yer Etiketleri için Kural Tabanlı Nitelikler	39
7.1.4.3. Biçimbilimsel Nitelikler	40
7.1.4.4. Sözdizimsel Nitelikler	40
7.1.4.5. Yapısal Nitelikler	40
7.1.5. Koşullu Rastgele Alanlar’ın Kullanımı	40
7.1.6. Başarımın Ölçülmesi	41
7.2. Türkçe Dökümanlar için Kural Tabanlı Varlık İsimleri Tanımlama	42
7.2.1. Geliştirilen Sistemin Yapısı	42
7.2.2. Varlık İsimleri için Tanımlanan Kurallar	43

7.2.2.1. Tanımlama Kurallarına Örnekler	43
7.2.3. Sonuç	45
BÖLÜM 8. SONUÇ	46

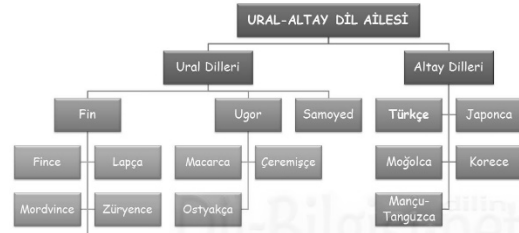
ÖZET

Anahtar kelimeler: Doğal Dil İşleme, Bilgi Çıkarımı, Koşullu Rastgele Alanlar, Varlık İsmi Tanıma, Kural Tabanlı, Türkçe, Biçimbilimsel Çözümleme, Morfoloji

Bilgisayar mühendisliği öğrencileri olarak, Türkçe dilinin yapısal açıdan (morfolojik) olarak incelemesini ve biçimsel olarak tanımlanmasını amaçladığımız bu makalede, dilimizin kurallarını ve ilginç yanlarını ele almaya çalıştık. Ayrıca Türkçe dilinin biçimsel makinelerde gerçekleştirilmesini hedefledik.

BÖLÜM 1. GİRİŞ

Türkçe, Şekil 1.1 de görüldüğü üzere Ural-Altay dil ailesinin Moğolca, Mançu-Tanguzca, Japonca ve Korece ile birlikte Altay koluna ait bir dildir.



Şekil 1.1

Günümüz Türkçesi, Türkiye, Ortadoğu ve Batı Avrupa ülkelerinde yaklaşık 60 milyon kişi tarafından anadil olarak konuşulmaktadır. Türk dilleri ailesi arasında günümüzde kullanılmayan diller de dahil olmak üzere, yaklaşık 40 dil vardır ve bu diller dünya üzerinde yaklaşık 165-200

milyon kişi tarafından anadil olarak kullanılmaktadır. Şekil 1.2 de Türk dilleri ailesi içerisindeki dillerin konuşulma oranları oranları verilmiştir.

Dil	%	Türkçe ve Türk dilleri ailesinin diğer dillerinin doğal dil işleme açısından çok ilginç zorluklar içeren bir dizi özellikleri vardır. Dilbilim açısından bakıldığında, özellikle eklemeli biçimbirim yapıları diller, ünlü uyumu veya tümce öğelerinin serbestçe yer değiştirebilmesi konuları Türkçe'nin bu özelliklere sahip bir dil olduğunu gösterir.
Türkçe	30.3	
Azerice	11.7	
Özbekçe	10.2	
Kazakça	4.3	
Uygurca	3.6	
Tatarca	2.2	
Türkmençe	1.3	
Kırgızca	1.0	
Diğerleri	35.4	

Şekil 1.2

BÖLÜM 2. BİÇİMBİLİMSSEL ÇÖZÜMLEME VE TÜRKÇE’NİN BİÇİMBİLİMSSEL YAPISI

Doğal Dil İşleme’nin en temel seviyelerinden biri olan biçimbilimsel çözümleme, bir sözcüğün yapısının bilgisayarlar tarafından otomatik olarak çözümlenmesi işlemidir. Biçimbilimsel çözümleme neticesinde, bir sözcüğün en küçük anlamlı birimleri olan morfeplerin (biçimbirimlerin) ve sözcük yapısının çözümlenmesi amaçlanır. Örnek vermek gerekirse, “çiçekler” sözcüğünün gövdesini “çiçek” kelimesinin oluşturduğunun ve bu sözcüğün çoğul eki almış bir isim olduğunun tespit edilmesi bir biçimbilimsel çözümleme işlemidir. İşlem sırasında, sözcüğü oluşturan morfeplerin birbirinden ayrılması göz önünde bulundurularak, bu işleme aynı zamanda “Biçimbilimsel Ayırıştırma” adı da verilmektedir.

Biçimbilim açısından Türkçe bitişken bir dildir, biçimbirimler bir sözcüğe deyim yerindeyse “tespih taneleri” gibi eklenirler. Türkçe’de önek kavramı yer almamaktadır. Almanca’daki gibi bir dizi isim kökünü birbirine ekleyerek beraber yazılan birleşik isimler de bulunmaz. Türkçe’de birleşik isimlerin bitişik yazılması için belli başlı bazı kurallar aranır. Ayrıca Türkçe’deki birleşik isimler, genellikle öğelerinin anlamlarından uzaklaşmış durumdadır. (Örn: Katırtırnağı, bir çiçek adıdır.)

Türkçe’de sözcükler, özel isimlere bakılmaksızın, 30.000 kadar kök sözcüğe çok üretken bir şekilde bir dizi ek ekleyerek oluşturulur. İsimler, Almanca veya Fransızca’da olduğu gibi sınıflara ayrılmaz. Sözcük dağarcığı tarihsel, coğrafi ve ekonomik birtakım nedenlerden dolayı zaman içerisinde çeşitli dillerden etkilenmiştir. Son 50-60 yıl içerisinde İngilizce ve daha öncesinde Arapça, Farsça, Yunanca, Ermenice, Fransızca, İtalyanca ve Almanca gibi dillerin oldukça etkisi altında kalmıştır.

Bir tümce içinde kullanıldığında sözcükler bir dizi çekim ve/veya yapım eki alabilir. Bu tip sözcüklerden bazıları başka dillerde bir tümceye karşılık gelebilir. (Bkz: Şekil 2.1)

yap+abil+ecek+se+k →
if we will be able to do (it)

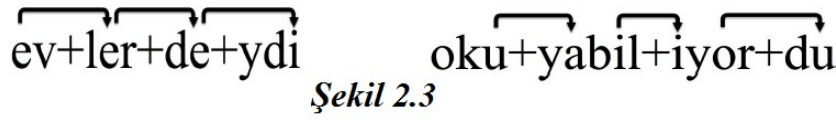
Şekil 2.1

Şekil 2.2 de gördüğümüz üzere, ilk kelimeye bitişen *-den* eki, gövdeyle uyumlu hale

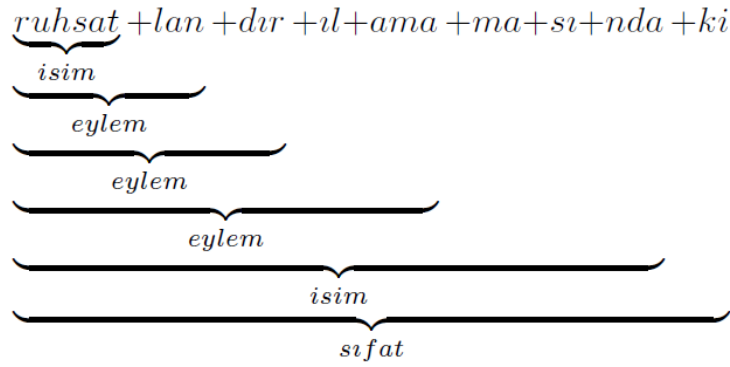
paket+ten araba+dan
Şekil 2.2

gelebilmek için ilk harfini *t* olarak değiştirir. Şekildeki ikinci kelimedede ise ekin ilk ünsüzü *d*

olarak kalır; fakat ünlü uyumunu sağlamak adına *e* harfi *a* olmak durumundadır. Ünlü uyumu adını verdiğimiz bu durum soldan sağa doğru silsile olarak gider. (Bkz: Şekil 2.3)



Türkçe sözcüklerde yapım eklerinin bulunmasına oldukça sık rastlanır. Böyle sözcükler çok karmaşık bir yapıya sahip olabilir. (Bkz: Şekil 2.4)



Ancak istatiksel olarak durum bu sözcükte olduğu kadar kötü değildir. Genelde bir derlemdeki sözcüklerde ortalama yaklaşık üç biçimbirim gözlenir. Ancak bu biraz yanıltıcıdır; çünkü yüksek sıklıkta görülen sözcükler genelde tek bir biçimbirimden oluşur.

Ayrıca sözcüklerin biçimbilimsel açıdan ortalama iki farklı yorumu vardır: 1. Kök sözcüğün sınıfının farklı olması. (Örneğin isim *ek* veya eylem *ek*) 2. Aynı yazılan biçimbirimlerin farklı anlamlara gelmesi. (Örneğin emir kipi *oku+ma* veya mastar *oku+ma*)

Sözcük	Biçimbirim	Yorum	Sözcük	Biçimbirim	Yorum
1 bir	1	4 11	kadar	1	2
2 bu	1	2 12	ama	1	3
3 da	1	1 13	gibi	1	1
4 için	1	4 14	olan	2	1
5 de	1	2 15	var	1	2
6 çok	1	1 16	ne	1	2
7 ile	1	2 17	sonra	1	2
8 en	1	2 18	ise	1	2
9 daha	1	1 19	o	1	2
10 olarak	2	1 20	ilk	1	1

Şekil 2.5

Şekil 2.5 büyük bir Türkçe

derlemdeki en sık yirmi

sözcüğü, yanlarında biçimbirim

sayısı ve farklı yorum sayısı ile

birlikte göstermektedir. Bu

rakamlardan kabaca şu

sonuçları çıkarmak mümkündür:

1) Yüksek sıklıktaki sözcüklerin

çoğunda biçimbirim sayısı bir olduğuna göre, ortalamanın üç olması için düşük sıklıktaki

sözcüklerin üçten fazla biçimbirime sahip olması gerekir. 2) Ayrıca yine yüksek sıklıktaki

kelimelerin çoğunun çok sayıda yorumu olduğu düşünülürse, ortalamanın iki yorum olması

için düşük sıklıktaki kelimelerin genelde ortalama ikiden az yorumu olması beklenir.

Türkçe sözcüklerin biçimbilimsel yapılarının vurgulanması gereken bir diğer özelliği ise, daha önce bahsedildiği gibi yapım eklerinin çok sık kullanılmasıdır.

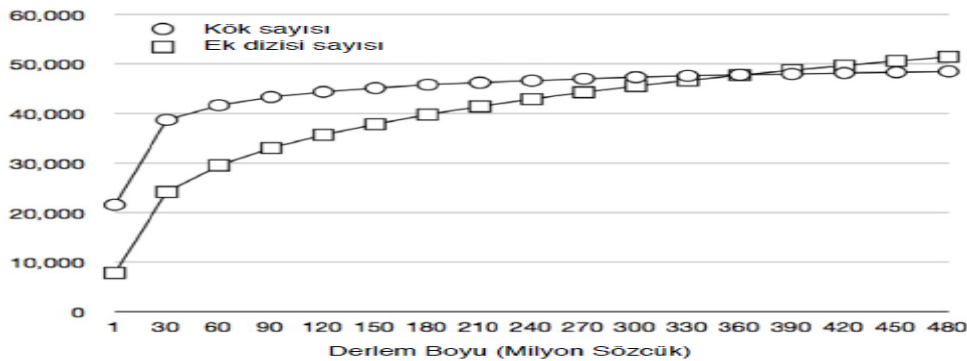
Kök	Yapım Eki	Sözcük	Toplam
masa	0	112	112
	1	4,663	4,775
	2	49,640	54,415
	3	493,975	548,390
oku	0	702	702
	1	11,366	12,068
	2	112,877	124,945
	3	1,336,266	1,461,211

Şekil 2.6

Şekil 2.6 da tek bir isim (masa) veya tek bir eylem (oku) köklerinden hiç yapım eki kullanmadan yahut 1, 2 ya da 3 yapım eki kullanarak elde edilebilecek farklı sözcüklerin sayısı görülmektedir. Tabii ki bu durum şekilde görüldüğü kadar sözcüğün kullanıyor olması anlamına gelmez; fakat bu sayılar dilin biçimbilim yapısı

açısından üretim gücünü gösterir. Tek bir eylem kökünden neredeyse 1.5 milyon değişik sözcük türetilmesi oldukça şaşırtıcıdır.

Bu üretkenliğin gerçek kaynaklardan toplanan derlemler ile desteklendiği de görülmektedir. Sak ve diğerleri, yaklaşık 500 milyon sözcükten oluşan bir haber derleminden topladıkları istatistiklerde şu gözlemlere varmışlardır: Bu derlemde yaklaşık 4.1 milyon farklı sözcük vardır ve bunların en sık geçen 50 bini derlemin %89'unu, en sık geçen 300 bini ise %97'sini kapsamaktadır. 3.5 milyon sözcük 10 defadan az geçmektedir, 2 milyon sözcük ise yalnızca bir kere geçmektedir. Derleme 490 milyon sözcükten sonra 1 milyon sözcük daha eklenince daha önceden hiç karşılaşılmamış 5.539 yeni sözcük gözlemlenmiştir. Yine aynı çalışmada bu derlemdeki sözcükler kök ve kök sonrası ek dizisi olarak ayrılır ve her bir gruptaki farklı kökler ve ek diziler sayılırsa belli bir noktadan sonra (yaklaşık 360 milyon sözcük), karşılaşılan farklı ek dizilerinin sayısı farklı köklerin sayısını geçmektedir. (Bkz: Şekil 2.7) Bu pratikte sonsuz addedilecek sözcük dağarcığı hemen hemen her türlü doğal dil işleme uygulamasında ilginç sorunlar çıkarmaktadır.



Şekil 2.7

Yazım Düzeltmesi: Yazım düzeltmesi için diğer diller için geliştirilen ve sonlu bir sözcük dağarcığı kabulüne dayanan teknikler Türkçe için uygun değildir. Önceki çalışmalarımızda, Türkçe gibi diller için sözcük dağarcığını sonlu durumlu dönüştürücü ile göstermeye dayanan ve sonlu durumlu makine çizge yapıları üzerinde çok etkin bir şekilde hataya dayanıklı yaklaşık arama yapan algoritmalar geliştirdik. (Türkçe ve Doğal Dil İşleme, Kemal Oflazer)

İşaret Kümesi Tasarımı: Türkçe sözcüklerin biçimbilimsel çözümlemesi sonucunda çıkan bilgileri Almanca ve İngilizce gibi dillerde olduğu gibi sonlu sayıda işaret ile göstermek olanaklı değildir. Şekil 2.7 de görüldüğü gibi eklerde kodlanan bilgi kurumsal ve pratik olarak sonlu bir sınır içinde değildir. Her ne kadar az sayıda sözcük sınıfı olsa da yapım ve çekim eklerinin sayısının önceden belirli bir sayıda olmaması, işaret sayısının sonlu olmasını önlemektedir.

İstatiksel Dil Modelleme: Büyük sözcük dağarcığı istatiksel dil modellemede hemen hemen her zaman veri yetersizliği probleminin yaşanmasına neden olur. Ebru Arısoy'un doktora tezinden alınan Şekil 2.8, bir konuşma tanıyıcı sisteminde kullanılan dil modeli için farklı sözcük sayısına göre bilinmeyen sözcüklerin oranı hakkında bilgi vermektedir.

Dil	Vocab.	BSO	Görüleceği gibi Türkçe ve çekimli bir dil olan
İngilizce	60K	1%	Çekçe'de %8 gibi bir bilinmeyen sözcük oranı vardır.
Türkçe	60K	8%	Türkçe gibi bitişken diller olan Fince ve Estonyaca'da
Fince	69K	15%	ise çok daha yüksek oranlar gözlemlenmiştir. Arısoy
Estonyaca	60K	10%	ayrıca dil modelleme için biçimbirimleri kullanmış ve
Çekçe	60K	8%	de yaklaşık 76 bin kök ve biçimbirim kullanarak test
			kümesi için çok çok düşük bir bilinmeyen sözcük
			oranı gözlemlemiştir.

Şekil 2.8

Sözdizim Modellemesi: Yapım ekleri sözdizim modellemesi açısından çok ilginç işlevlere sahiptirler. Bu özellikler hem öbek tabanlı modellemeler hem de bağımlılık tabanlı modellemelerde geçerlidirler.

İstatiksel Çeviri: İstatiksel dil modellemede olduğu gibi istatiksel çeviride de büyük sözcük dağarcığı veri yetersizliği problemini öne çıkarır. Bu problemi aşmak için yine biçimbilimsel yapıya dayalı çeviri yaklaşımları oldukça iyi sonuçlar elde etmişlerdir.

BÖLÜM 3. BİÇİMBİLİMSEL ÇÖZÜMLEMENİN KULLANIM ALANLARI

Doğal Dil İşleme gerektiren birçok gerçek dünya uygulamasında biçimbilimsel çözümleyicilere ihtiyaç duyulur. Örneğin Türkçe, Macarca, Almanca, Rusça gibi karmaşık biçimbilimsel yapıya sahip olan diller için arama motorları kullanılarak internette arama yapabilmek için aranan sözcüğün tüm çekimli hallerinin taranabilmesi gerekmektedir. İnternette herhangi bir arama motorunda “Rumeli Hisar Konserleri”, “Rumeli Hisarı Konseri”, “Rumeli Hisarı Konserleri” aramaları için bulunan sonuçların benzerlik göstermesi gerekmektedir. Bu tür bir uygulamada, ikircikli bir biçimbilimsel çözümlemeye ihtiyaç duyulmasa da, sözcüğün eklerinden arındırılması ve doğru sözcük gövdelerine ulaşılabilmesi gerekmektedir. Böylelikle internet sayfalarının doğru bir şekilde indislenmesi mümkün olabilmektedir.

Biçimbilimsel çözümleyicilere ihtiyaç duyulan bir diğer uygulama alanı da, doğru yazım denetimi yapan, yanlış yazılan sözcüklere yeni önerilerde bulunabilen programlardır. Örneğin metin editörlerinde, kullanıcıların düzgün sözcükler yazıp yazmadığı kontrolünün yapılması önemli bir işlemdir. Bir metin elektronik ortamda hazırlanırken, hızlı yazım nedeniyle bazı harflerin atlanması veya harflerin yerlerinin karıştırılması sık karşılaşılan durumlardır. Karmaşık sözcük yapısına sahip olmayan dillerde, önceden hazırlanmış bir sözlük kullanılarak problem şu şekilde çözülebilir:

1. Metin içerisindeki her sözcüğün sözlükte olup olmadığı kontrol edilir.
2. Eğer sözcük sözlükte yer almıyorsa, sözlükteki sözcüklerden yazılan sözcüğe en yakın olanı çeşitli algoritmalar kullanılarak bulunur ve kullanıcıya önerilir.

Ancak karmaşık sözcük yapısına sahip dillerde böyle bir sözlük hazırlanması imkansızdır. Bu nedenle hatalı yazılan sözcüklerdeki hatayı bulabilmek ve öneride bulunabilmek için biçimbilimsel bilgiye ihtiyaç duyulur.

Hata denetleme ve öneride bulunma sistemlerindeki benzer bir durum bul/değiştir işlemlerinde de ortaya çıkmaktadır. Bul/değiştir işlemleri sırasında, öncelikle aranan sözcüğün metin içerisinde bulunması daha sonra bu sözcüğün düzgün biçimde yeni sözcük ile yer değiştirmesi amaçlanmaktadır. Bu işlemin Türkçe gibi ses uyumları, harf düşmeleri içeren dillerde bir biçimbilimsel çözümleme yapılarak gerçekleştirilmesi gerekliliği ortaya çıkmaktadır. Bu gerekliliğin ilk nedeni aranan sözcüğün aldığı ekler ile farklı şekilde

görülebiliyor olmasıdır. Örneğin “oğul” sözcüğünün “kız” sözcüğü ile değiştirilmesi istendiğinde, bu sözcüğün metin içerisinde 1. Tekil kişi iyelik eki almış hallerinin (“oğlum”) de bulunabilmesi gerekmektedir. Başka bir örnek de “dolap” sözcüğü için verilebilir. Bu sözcüğün –i hal eki almış durumu “dolabı” olacaktır. Karakterleri birebir karşılaştıran basit bir arama işlemi bu sözcükleri bulamayacaktır. Benzer şekilde değiştirme işlemi aşamasında da, yeni sözcüğün orijinal sözcüğün çekimli haline benzer bir halde oluşturulması gerekmektedir. Örneğin bir metindeki “ev” sözcükleri “bina” sözcüğü ile değiştirilmek istendiğinde, öncelikle “ev” sözcüğünün bütün çekimli hallerinin bulunması, daha sonra bulunan bu sözcüklerdeki çekim eklerinin “bina” sözcüğüne eklenmesi ve yeni sözcüğün oluşturulması gerekmektedir.

Farklı iki doğal dil arasında makine çevirisi yapmayı hedefleyen bir sistemin de sözcük seviyesinde çözümleme yapması gerekebilir. Örneğin “arabaymışsınız” kelimesi, Türkçeden İngilizceye çevrilmek istendiğinde, Türkçe sözcükteki her bir biçimbirimin İngilizce karşılığı bulunmalıdır. Bu amaçla, öncelikle Türkçe sözcüğün biçimbilimsel ayrıştırması yapılmalı, daha sonra da ayrılan biçimbirimler için çeviri işlemi yapılmalıdır. (Bkz: Şekil 3.1)

Arabaymışsınız	– <i>you had been in the car</i>
araba(isim)	– <i>the car</i>
+da(İsmin –de hali)	– <i>in</i>
+ymış (-miş ekfiil eki ile eyleme dönüşme)	– <i>had been</i>
+sınız (2. çoğul kişi eki)	– <i>you</i>

Şekil 3.1

Yukarıda biçimbilimsel çözümleyicilerin kullanılabileceği bazı doğal dil işleme uygulamalarından bahsedilmiştir. Bu örnekler daha da arttırılabilir. Biçimbilimsel çözümleme, doğal dil işlemenin en temel seviyelerinden biridir ve üst düzey uygulamaların çoğunda kullanımına ihtiyaç duyulmaktadır.

BÖLÜM 4. BİÇİMBİLİMSEL ÇÖZÜMLEME ÇIKTILARI

Bu bölümde, önceki bölümde ele alınan örneklemeleri değerlendirerek şu soruya cevap arayacağız: İdeal bir biçimbilimsel çözümleyicinin ne düzeyde çözümlemeler üretmesini beklemek anlamlıdır? Önceki bölümde sözü geçen, bağlama bakarak belirsizlik giderme işlemlerinin her biri, üzerinde yoğun olarak çalışılan ayrı bir doğal dil işleme problemidir. Bu problemlere çözüm üreten algoritmaların, güvenilir ve tutarlı çözümler üreten bir biçimbilimsel çözümleyicinin çıktılarına ihtiyaçları vardır. Dolayısıyla, yukarıda örnekler ile ayrıntılı olarak anlattığımız, söylenen bir sözcüğe karşılık düşen anlamın bulunması işlemi, bir bilgisayar için tek bir doğal dil işleme adımı değildir. Bu problemin düzgün olarak çözülebilmesi için bir biçimbilimsel çözümleyiciye ve bunun sonuçları üzerinde çalışacak belirsizlik gidericilere ihtiyaç vardır.

4.1. Gereksiz Sonuçların Filtrelenmesi

İyi bir biçimbilimsel çözümleyiciden beklenen, bir sözcüğün ilgili doğal dildeki tüm olası çözümlemelerinin listelenmesidir. Ancak bu çözümlemeler bir insan tarafından incelendiğinde anlamlı ve birbirlerinden ayırt edilebilir olmalıdırlar. Bu nedenle, bazı durumlarda bazı çözümlemelerin filtrelenmesi anlamlı olacaktır. Örneğin bir biçimbilimsel çözümleyicinin “gözlük” ve “yaptırdı” sözcükleri için aşağıdaki sonuçları üretmesi mümkündür. (Bkz: Şekil 4.1.1)

gözlük	yaptırdı
1. gözlük+isim+yalınhali 2. göz+isim+yalınhali+TS+lük+isim+yalınhali 3. göz+isim+yalınhali+TS+lük+sıfat	1. yaptırdı+fiil+geçmişzaman+3.tekilkişi 2. yap+fiil+TS+tır+fiil+geçmişzaman+3.tekilkişi
TS etiketi sözcüğün türetim eki olarak başka bir sözcüğe dönüştüğü türetim sınırını ifade etmektedir.)	

Şekil 4.1.1

Tüm çözümlemeler geçerlidir. Ancak, bu sonuçlardan birinin doğru olarak seçilmesi işlemi bir insan için dahi oldukça zordur. Çünkü her iki örnekte de birinci ve ikinci çözümlemeler aynı kavramları belirtmektedirler. “Gözlük” örneğinde ilk çözümleme yalın haldeki “gözlük” sözcüğüdür. İkinci çözümleme ise “-lük” yapım eki almış “göz” sözcüğünü belirtmektedir.

Üçüncü çözümleme, -lük yapım eki ile bir isimin sıfata dönüşmesi durumunu belirtmektedir. Bu sözcüğün ilk iki çözümlemesi tamamen aynı kavramları ifade etmektedir. Benzer şekilde “Yaptırmak” fiilinin geçmiş zaman 3. Tekil kişi eki almış hali ile, “yapmak” fiilinin ettirgen çatıda geçmiş zaman 3. Tekil kişi eki almış hali aynı eylemi ifade etmektedir. Buradan yola çıkılarak ikinci çözümlemelerin gereksiz olduğu sonucuna varılabilir. Her iki durumda da sözcük, üzerine aldığı bir ek ile Türkçe sözlükte anlamı olan başka bir sözcüğe dönüşmüştür. Sözcüğün bu yapım eki ile aldığı yeni anlam o dil içerisinde tanımlıdır ve dilin sözlüğünde belirtilmiştir. Ancak eğer sözcüğün yeni aldığı anlam o dilin sözlüğünde yer almıyorsa, bu sefer ilk çözümlemeler üretilmeyecektir. Bu durumda ise ikinci çözümleme sonuçlarının gösterilmesi gerekecektir. Aşağıdaki sözcük (Şekil 4.1.2) bu duruma örnek olarak gösterilebilir. Bu örnekte, sözlükte “rujluk” diye bir sözcük yer almadığı için, bir ismin -lık yapım eki ile türemesi çözümlemesinin yapılabiliyor olması gerekmektedir.

Rujluk	Bir biçimbilimsel çözümleyici yan taraf da
ruj+isim+yalınhali+TS+luk+isim+yalın hali	gösterilen tüm çözümlemeleri yapabilmeli ancak
ruj+isim+yalınhali+TS+luk+sıfat	yaptığı çözümlemeler aynı kavramlara denk
	düşüyorsa bunlardan bazılarını sonuç

Şekil 4.1.2

listesinden elemelidir.

4.2. Olası Çözümlerin Genişletilmesi

Bazı durumlarda sözcükler cümle içerisinde farklı rollerde bulunabilirler. Örneğin “öğrencilerim” sözcüğünün çözümlemesi yapılmak istense oluşturulacak çözüm, bu sözcüğün gövdesi “öğrenci” olan birinci tekil kişi iyelik eki almış yalın halde çoğul bir isim olduğudur: (“öğrenci+isim+çoğul+1.tekil kişi iyelik+yalın hali”). Ancak bazı durumlarda bu sözcük cümle içerisinde fiil rolünde de görülebilir. Buna örnek olarak “İşte, bunlar benim öğrencilerim.” cümlesini ele alalım. Cümlede söylenmek istenen “öğrencilerimdir” sözcüğüdür. Ancak “-dir” ek fiil yapım eki düşmüş ve “öğrencilerim” denmiştir. Türkçede bu kullanımın şekli de geçerlidir. Burada, “öğrencilerim” sözcüğünün içerisinde yer aldığı bağlama bakılarak, çözümlemesi seçilmek istense, bu bağlamdaki rolü (“öğrenci+isim+çoğul+1.tekil kişi iyelik+yalın hali +TS+Fiil”), ürettiği çözümlemeler arasında yer almayacaktır.

Yukarıdaki örnekte de görüldüğü gibi, iyi bir biçimbilimsel çözümleyici, sözcüğün o dildeki kullanımına yönelik olası tüm çözümlemeleri üretmelidir. Tabii ki, her yalın haldeki isim için bir de fiil çözümlemesi üretmek karmaşıklığı arttırıcı ve gereksiz bir durum olarak görülebilir. Ancak bir cümledeki sözcüklerin doğru olarak etiketlenmesi istendiğinde bu işlem gereklidir. Biçimbilimsel çözümleyicinin seçime bağlı olarak bu ek çözümlemeleri üretmesi veya bazı önsezilerde bulunarak üretmesi anlamlı olabilir.

BÖLÜM 5. BİÇİMBİLİMSEL ÇÖZÜMLEME YAKLAŞIMLARI

Doğal Dil İşleme üzerine yapılan çalışmalar genelde istatistiksel ve kural tabanlı olmak üzere iki ana sınıfa ayrılırlar. Ancak herhangi bir yöntemi bu iki ana sınıftan sadece birine atamak çoğu zaman kolay bir işlem değildir. Bir doğal dil işleme uygulaması, genel olarak;

- Dilin yapısı ile ilgili bilginin bilgisayarda tutulması,
- Bu bilginin eldeki doğal dil verileri üzerinde kullanılması ve
- Sonuçların değerlendirilmesi

aşamalarından oluşur. Dolayısıyla, bu aşamalardan her biri istatistiksel veya kural tabanlı olabilir. Bir uygulama, bu aşamalardan biri veya daha çoğunda iki yaklaşımdan birini kullanabilir. Biçimbilimsel çözümleme yaklaşımlarını, yukarıda sıralanan ilk aşama olan dilin yapısı ile bilginin çıkarılmasına göre istatistiksel ve kural tabanlı olarak ikiye ayırabiliriz.

Dille ilgili biçimbilimsel yapının istatistiksel yöntemler ile çıkarılması yaklaşımı bazı çalışmalarda (Gelbukh & Sidorov, 2009) (Hirsimäki, Creutz, Siivola, Kurimo, Pylkkönen, & Virpioja, 2006) (Kurimo, Creutz, Varjokallio, Arisoy, & Saraclar, 2005) benimsense de, dilin biçimbilimsel yapısının genel olarak sistematik olmasından ötürü, bu konuda kural tabanlı yaklaşımlar ağırlık kazanmaktadır (Sproat, 1992) (Beesley & Karttunen, 2003). Bu işlem sırasında, bir doğal dilin o dil üzerinde çalışan dilbilimciler tarafından ayrıntılı olarak hazırlanmış yapı bilgisi bir bilgisayarıcı gözüyle irdelenerek, bir bilgisayarın anlayabileceği biçime dönüştürülür. Bu işlem, bir bilgisayarıcının bir dilbilimci titizliği ile dilin kurallarını incelemesini ve işlemesini gerektirir. Bu nedenledir ki doğal dil işleme konusunda çalışan grupların her iki bilim alanında uzmanlaşmaları gerekmektedir.

Biçimbilimsel çözümleme konusunda ilk yapılan kural tabanlı çalışmalar alttan-üste veya üstten-alta ayrıştırma algoritmalarından birine yönelmişlerdir (Jurafsky & Martin, 2009). Burada üst ve alt terimleri, bir sözcüğün görülüş biçimi ve onun altında yatan biçimbilimsel yapı kavramlarını ifade etmektedirler. Üstten-alta analiz yapan çalışmalarda eldeki sözcüğe karşılık düşen analizlerin bulunabilmesi için sözcük üzerindeki ekler tek tek yakalanıp, bunların kurallara uyumları kontrol edilerek analiz yapılmaya çalışılır. Bu tür çalışmalara örnek olarak Brodda ve Fred (1980)'in Fince için, Kaalep (1997)'in Estonyaca için, Cebiroğlu ve Adalı (2004)'nın Türkçe için, Packard (1973)'in eski Yunanca için yaptığı sistemler gösterilebilir. Altan-üste ayrıştırma yapan çalışmalarda ise, bir gövdeye eklenebilecek olası

tüm ekler eklenerek oluşturulan yeni sözcüğün, incelenmesi istenen sözcükle eşleşip eşleşmedikleri kontrol edilir. Eşleşen çözümlemeler doğru kabul edilir. Bu tür çalışmalara örnek olarak da Hankamer'in Türkçe için geliştirdiği KECI (Hankamer, 1986) isimli çözümleyicisi gösterilebilir.

Biçimbilimsel Çözümleme konusunda 1970'li ve 1980'li yıllar arasında en çok araştırma sonlu durumlu dönüştürücüler üzerinde yapılmıştır (Johnson, 1972) (Kaplan & Kay, 1981). Bu yaklaşım, bir dilin biçimbilimsel kurallarının bağlam duyarlı basit yeniden yazım kuralları ile belirtilebileceği düşüncesini savunmaktadır. Dilin biçimbilimsel özelliklerini tanımlayacak bu tür kurallar hazırlanarak bir sonlu durumlu dönüştürücüde toplanacaktır. Bu kurallar, dönüştürücüye verilen girdi üzerine ard arda uygulanarak sonuç üretilecektir. Ancak, kurallar üzerindeki kısıtlamalar sadece dönüştürücüye verilen girişe dayalı kısıtlar olacağından, bir kural kendinden önce uygulanan başka bir kuralın çıktısı üzerinde değişiklik yapamayacaktır.

Sonlu Durumlu Dönüştürücülerin yarattığı sorunlar, 1983 yılında Kimmo Koskenniemi'nin "İki Seviyeli Biçimbilimsel Çözümleyici"si ile çözülmüştür (Koskenniemi, 1983).

Çözümleyici ilk olarak Fince üzerinde denenmiş; ancak bu yöntem daha sonra aralarında Fransızca, İngilizce, İspanyolca, Hollandaca, İtalyanca ve Türkçe'nin de bulunduğu birçok dil için uygulanmıştır. Bu yaklaşımda, sonlu durumlu dönüştürücülerden farklı olarak, oluşturulacak biçimbilim kuralları art arda değil eş zamanlı olarak uygulanacaklardır. Buna ek olarak, kurallardaki kısıtları, oluşturulacak çıktıya bağımlı kılmak da mümkündür. İki seviyeli yaklaşım, bilgisayarıcılar tarafından çok ilgi görmüş, birçok çalışmada kullanılmış ancak zamanın dilbilimcileri tarafından paralel çalışacak dilbilgisi kuralları yazımının zorluğu nedeniyle kabul görmemiştir. Ancak günümüzde sıkça benimsenen optimalite teorisinin de benzer bir görüşü yansıttığı kaynaklarda vurgulanmaktadır (Karttunen & Beesley, 2001).

Koskenniemi'nin iki seviyeli biçimbilim yaklaşımı, literatürde karmaşık biçimbilimsel yapıya sahip dillerin çözümlemesi için genel kapsamlı ilk pratik model olmuştur. Model çift yönlüdür ve hem üretim (analizi verilen bir sözcüğün yeniden oluşturulması) hem de çözümleme (verilen bir sözcük için analiz oluşturulması) için kullanılabilir. Bu yaklaşım için üretilmiş, Xerox'un iki seviyeli derleyicisi birçok büyük ölçekli biçimbilimsel çözümleyici geliştirilmesinde kullanılmıştır.

Bir biçimbilimsel çözümleyicinin 3 temel unsurdan oluştuğu söylenebilir. Bunlar sırasıyla;

1. Sözcükleri ve sözcüklerin bazı temel özelliklerini tutan "leksikon" adı verilen bir sözcük listesi,

2. Dilin ses bilgilerini belirleyen ortografik kurallar ve
3. Eklerin ekleniř sıralarını belirleyen morfotaktik kurallardır.

BÖLÜM 6. TÜRKÇE’NİN YAPISI

Türkçe'nin biçimbilimsel yapısından kısaca ikinci bölümümüzde bahsetmiştik. (Bkz: 2. Biçimbilimsel Çözümleme ve Türkçe'nin Biçimbilimsel Yapısı) Bu konu başlığımız altında Türkçe'nin yapısını daha detaylı bir biçimde inceleyeceğiz.

6.1. Çekim Kümeleri

Türkçede sözcükler sonlarına eklenen eklerle farklı türde sözcüklere de dönüşebilirler; eylemler isimlere, isimler eylemlere vb. Türkçenin bu özelliği ilgili yayınlarda (Oflazer, Say, Tür, & Tür, 2003), (Oflazer, 2003), (Eryiğit, Statistical Dependency Parsing of Turkish, 2006), (Eryiğit, 2006), (Hakkani-Tür, Oflazer, & Tür, 2002) sözcüklerin *çekim kümelerine* (ÇK) ayrılması biçiminde gösterilmektedir. Bu gösterimde, Türkçe bir sözcüğün bir dizi çekim kümesinden oluştuğu ve bu ÇK'lerinin *türetim sınırlarından* (TS) bölündüğü varsayılmaktadır. Bu özellik aşağıdaki gibi gösterilmektedir:

Gövde +ÇK₁+ TS+ ÇK₂+ TS+...+ TS+ ÇK_n

Burada her ÇK_i , ilgili olduğu çekim kümesine ait biçimbilimsel özellikleri ve sözcük sınıflarını belirtmektedir. Aşağıda, bu durum türemiş bir niteleyici olan “sağlamlaştırdığımızdaki” sözcüğünün Oflazer (1994)'in biçimbilimsel çözümleyicisi ile çözümlemesi üzerinde gösterilmektedir. Bu gösterimde kullanılan etiketler şöyledir: +Adj: Sıfat, +Verb: Eylem, +Become: oluşmak, +Caus: Ettirgen, +Pos: olumlu, +Noun: İsim, +PastPart: geçmiş zaman ortacı, +A3sg: 3. tekil kişi kişi/sayı uyum imi, +P3sg: 3. tekil kişi iyelik imi, +Loc: -de hali, +Rel: ilişkilendirici.

Sağlam+ Adj

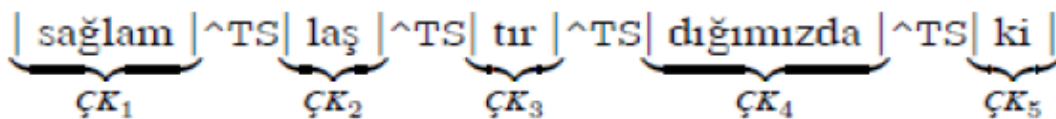
+ TS+ Verb+ Become

+ TS+ Verb+ Caus+ Pos

+TS+ Noun+ PastPart+ A3sg+ Plpl+ Loc

+TS+ Adj+ Rel

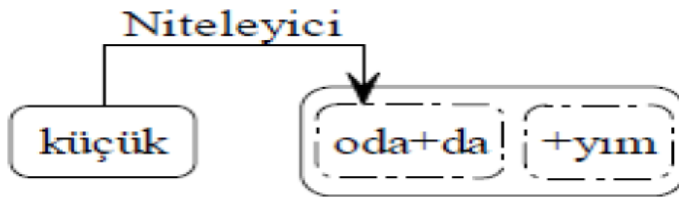
^TS sınırları sözcük üzerinde gösterilmek istenirse şöyle görünecektir. (Bkz: Şekil 6.1.1)



Şekil 6.1.1

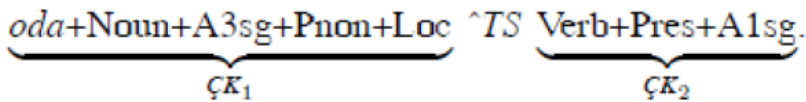
Buradaki beş çekim kümesi, TS türetim sınırı işaretleri ile birbirinden ayrılmış özellik dizileridir. İlk ÇK gövdenin tek özelliği olan sözcük sınıfını göstermektedir. ``sağlam" sözcüğü bir sıfattır. İkinci ÇK, önceki sığata "oluřmak" anlamı katılarak bir eylem türetmeyi göstermektedir. Üçüncü ÇK, önceki eylemden olumlu bir ettirgen eylemin türetildiğini belirtmektedir. Dördüncü ÇK, alt sözcük sınıfı olarak geçmiş zaman ortacı taşıyan, birinci çoğul kiři iyelik ve *-de* hal eki almış bir isimin türetilmesini belirtmektedir. Son olarak da, beşinci ÇK, ilişkilendirici bir sıfat türetilmesini belirtmektedir.

Türkçe'nin biliřimsel dil işlemeđi konusunda çalışanların sözcüklerden daha küçük birimler (ÇK'leri) tanımlama ihtiyaçları, Türkçe bir cümledeki cümle içi ilişkilerin sözcüklerin alt bölümleri arasında gerçekleşebilmesinden doğmaktadır. Türkçe'nin türetim sistemi çok üretkendir ve bir sözcüğün uydu (alt terim) veya iye (üst terim) olarak içerisinde bulunduğú cümle yapısı ilişkileri, sözcüğün içerdigi bir veya daha fazla türemiş yapının biçimbilimsel özellikleriyle belirlenmektedir. Bu ilişkileri, sadece sözcükler arasında göstermek çoğú doğal dil işleme uygulaması için yeterince anlamlı bilgi taşımamaktadır. Bu durum Şekil 6.1.2'de "küçük odadayım" cümleciğı içerisindeki bağılılık ilişkisi ile örneklenmektedir. Şekildeki gösterimde sözcükler dikdörtgenler içerisinde ve bunların içerdikleri çekim kümeleri de noktalı dikdörtgenler içerisinde gösterilmektedirler.



Şekil 6.1.2

Şekil 6.1.3'te ise "odadayım" sözcüğünün ÇK'leri gösterilmiştir.

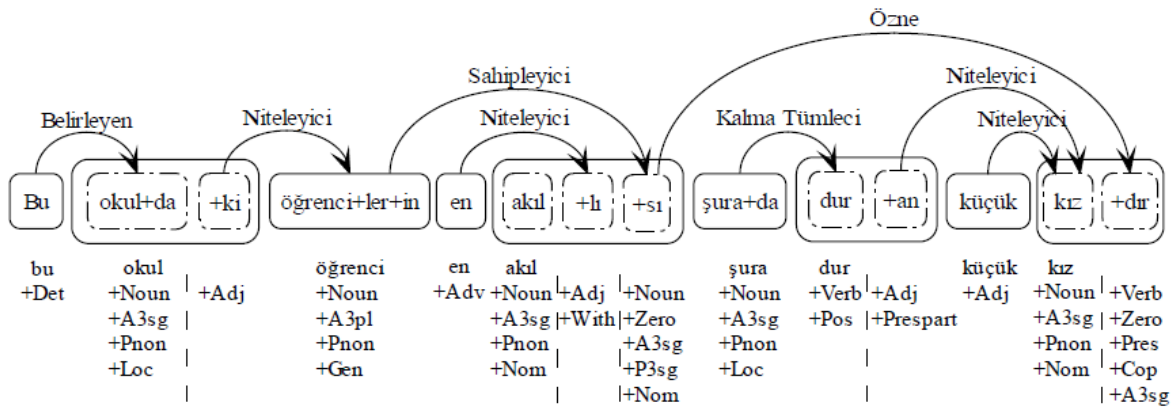


Şekil 6.1.3

Birinci ÇK "oda" isimi ve bu isime ait biçimbilimsel özellikleri içermektedir. Bu özellikler, isimin tekil, iyelik eki almamış ve *-de* halinde olduğunu belirtmektedir. İkinci ÇK ise bu

isimden türemiş “odada olmak” eylemini ve biçimbilimsel özelliklerini içermektedir. Eylem birinci tekil kişi eki almıştır ve şimdiki zamandadır. Örnekte “küçük” olan, “odadayım” sözcüğü değil “oda”dır. “odadayım” isimden eyleme dönüşmüş bir sözcüktür. İki sözcük arasında kurulan bağlantı “odadayım” sözcüğünün eyleme dönüşmeden önceki isim halinden kaynaklanmaktadır. Bu durum sıfatların genel olarak isimlere bağlanması kuralından kaynaklanmaktadır. Bu cümlecikte, “küçük olan nedir?” sorusunun cevabı bir sözcük değil, bir çekim kümesidir.

Cümle içi ilişkilerin sözcüklerden daha küçük birimler arasında kurulabildiğine güzel bir örnek de Şekil 6.1.4’te gösterilmektedir. Şekil “Bu okuldaki öğrencilerin en akıllısı şurada duran küçük kızdır” cümlesi üzerinde çekim kümelerini ve bunlar arasında oluşan ilişkileri göstermektedir. Bağlılıkların yönü uydu (alt terim) ÇK'den iye (üst terim) ÇK'ye doğru gösterilmiştir. Bağlılık türleri bağlılık oklarının üzerinde belirtilmektedir. Her sözcüğe ait biçimbilimsel çözümleme ilgili sözcüğün altında Oflazer (1994)’in gösterimiyle verilmiştir. Bağlılıklar uydu sözcüğün sadece son ÇK'sinden çıkmaktadırlar. Bu nedenle, şekilde bağlılık çıkmayan bazı ÇK'ler bulunmaktadır (Örn: okuldaki sözcüğünün ilk ÇK'si). Bu tip ÇK'ler bağlılıklarda sadece iye olarak bulunurlar. Bu ara ÇK'lerin arkalarından gelen ÇK'ye biçimbilimsel olarak bağlı oldukları varsayılır. Ancak bu bağlılıklar özellikle belirtilmez. İye ÇK ise iye sözcüğün herhangi bir ÇK'si olabilir. Bir başka deyişle, bağlılık herhangi bir sözcüğün herhangi bir ÇK'sinde sonlanabilir.



Şekil 6.1.4

Şekilden de görülebileceği gibi bir sözcükten sadece bir bağlılık oku çıkarken, birden fazla bağlılık oku girebilmektedir. Bir diğer deyişle, her sözcüğün sadece bir iyisi vardır ancak bir iye sözcüğün birden fazla uydusu olabilir. Birden çok ÇK içeren sözcüklere gelen bağlılıklar sözcüğün farklı ÇK'lerinde sonlanabilir. Şekildeki “akıllısı” sözcüğü bu duruma güzel bir

örnek teşkil etmektedir. “en” sözcüğü “akıllısı” sözcüğünün ikinci ÇK'sine (“en akıllı”) bağlıdır. “öğrencilerin” sözcüğü ise aynı sözcüğün üçüncü ÇK'sine bağlıdır (“öğrencilerin akıllısı”).

Türkçe için bu tür bir yaklaşımın makine çevirisi, otomatik cümle analizi gibi üst düzey doğal dil işleme uygulamalarının başarımını önemli ölçüde arttırdığı çeşitli çalışmalarda gösterilmiştir (Eryiğit, Nivre, & Oflazer, 2008), (El-Kahlout & Oflazer, 2009), (Hakkani-Tür, Oflazer, & Tür, 2002).

6.2. Eklerin Diziliş Kuralları

Kullanılacak herhangi bir yöntemden (Bknz: Bölüm 5. Biçimbilimsel Çözümleme Yaklaşımları) bağımsız olarak, Türkçe eklerin birbiri ardına geliş kuralları aşağıda iki farklı biçimde gösterilecektir. Bunlardan birincisi eklerin bir sözcük içerisinde soldan sağa doğru sözcük gövdesine hangi sırada ekleneceklerini sonlu durumlu makineler (SDM) ile gösterilmesidir. İkinci gösterim ise bu makinelerin ters çevrilmiş halleri yani sağdan sola halleridir. Birinci gösterim temel olarak Oflazer (1994)'in çalışmasındaki sonlu durum makineleri ile aynıdır. İkinci gösterim ise Cebiroğlu (2002)'nin çalışmasından alınmıştır. Cebiroğlu (2002)'nin geliştirmiş olduğu üstten-alta ek-atmalı ayrıştırma algoritmasında, bir sözcüğün ekleri sözcük sonundan başlanarak gövdeye doğru sözcükten birer birer çıkarılarak çözümleme yapılır. Bu çıkarma işlemi sonucunda elde edilen sözcük gövde kabul edilebileceği gibi, bu varsayımı doğrulama için bir sözlük de kullanılabilir.

Sadece bu makineleri kullanmak, Türkçe sözcüklerin tümünü doğru bir şekilde çözümleyecek yetkin bir çözümleyici oluşturmaya yeterli değildir. Yetkin bir çözümleyicinin çok fazla istisna durumu ve farklı kuralı ele alması ve Bölüm 4’de anlatılan kısıtlara uyması gerekecektir. Ancak bu SDM’ler ile hızlıca basit bir çözümleyici geliştirmek mümkündür.

Türkçede ekler sözcüklere bazı kurallara sadık kalınarak eklenirler. Benzer şekilde bir sözcük gövdesine birden fazla ek eklenmek istendiğinde bunların da belirli bir sıraya uyarak eklenmeleri gerekmektedir. Örneğin “kitap” sözcüğüne çoğul ve 1. Tekil kişi iyelik eki eklenmek istendiğinde, oluşacak sözcük “kitaplarım”dır. Bu durumda, “kitabımlar” şeklinde bir türeme yanlış kabul edilecektir. Buradan görülebilir ki, Türkçede iyelik ekleri genelde

çoğul ekinden sonra gelmektedirler: arkadaşlarım, ödevleriniz vb. Ancak bazen kural dışı durumlar ortaya çıkabilir. Örn: halamlar, babaannemler vb.

Bir sözcüğün birden fazla olası gövdesi olabilmesi durumunda ortaya çıkan belirsizlikler, sözcüğün üzerine farklı ekler eklendikçe ortadan kalkmaktadırlar. Bu fayda, Türkçede sadece bazı eklerin birbirlerinin ardına gelebilmesi sayesinde ortaya çıkmaktadır. “Kalem” sözcüğünü ele alacak olursak, bu kelimenin kökünün hem “kale” hem “kalem” olabileceği görülmektedir. Ancak “kalemlerim” sözcüğünün gövdesi sadece “kalem” olacaktır. “kale” gövdesi bu sözcük için mümkün değildir; çünkü bir sözcük içerisinde hem iki tane iyelik eki olamaz hem de iyelik ekleri genelde çoğul ekinden sonra gelirler.

Bu bölümde, Türkçe dilbilgisinin ayrıntılarına girilmeyecektir. Bu tür bilgiler, herhangi bir Türkçe dil bilgisi kitabından kolayca elde edilebilirler. Ancak bu yapıların bilişimsel açıdan nasıl temsil edilebileceği ve bilgisayar tarafından nasıl işlenebileceği gibi konulara açıklık getirilmesi gerekmektedir. Bu amaçla, aşağıdaki bölümlerde Türkçe ekler gruplanarak, birbirleri ardına eklenişleri sonlu durum makineleri kullanılarak modelleneyecektir. Bu modelleme sayesinde, oluşturulan makineler bir bilgisayar yazılımı sayesinde kolaylıkla işlenebilirler. Bu makineler tamamen doğru olmayıp, ekleniş kurallarının genel hallerini temsil etmektedirler. Bazı istisnai durumlar için makinelerde güncellemeler yapılması gerekebilir.

Takip eden bölümlerde, ekler aşağıdaki şekilde kümelere ayrılarak incelenmektedir. Her bir bölümde, ilgili kümeye ait çekim eklerinin listesi bir tablo şeklinde verilecek, daha sonra sıra ile eklerin soldan sağa ve sağdan sola sıralanış kurallarını belirten sonlu durumlu makineler gösterilecektir.

- İsim Çekim Ekleri Kümesi
- Ek-Eylem Ekleri Kümesi
- Eylem Zaman Ekleri Kümesi
- Eylem Çekim Ekleri Kümesi

İsim Çekim Ekleri Kümesi

Ek No	Ek	Açıklama	Örnek
1	-lAr	Çoğul	anne-ler
2	-(H)m	1. tekil kişi iyelik	anne-m
3	-(H)mHz	1. çoğul kişi iyelik	anne-miz
4	-(H)n	2. tekil kişi iyelik	anne-n
5	-(H)nHz	2. çoğul kişi iyelik	anne-niz
6	-(s)H	3. tekil kişi iyelik	anne-si
7	-lArI	1. çoğul kişi iyelik	anne-leri
8	-(y)H	-i hali	anne-yi
9	-nH	-i hali (3.t.k. iyelikten sonra)	anne-si-ni
10	-(n)Hn	tamlama	anne-nin
11	-(y)A	-e hali	anne-ye
12	-nA	-e hali (3.t.k. iyelikten sonra)	anne-si-ne
13	-DA	-de hali	anne-de
14	-nDA	-de hali (3.t.k. iyelikten sonra)	anne-si-nde
15	-DAn	-den hali	anne-den
16	-nDAn	-den hali (3.t.k. iyelikten sonra)	anne-sin-den
17	-(y)IA	birliktelik	anne-yle
18	-ki	ilgi	annem-de-ki
19	-(n)cA	görelilik	annem-ce

Şekil 6.2.1

ekleneceği sözcükle birlikte uğrayacağı sesli uyumuna göre “-lar” veya “-ler” şeklinde görülebilir. Benzer şekilde “-(H)nIz” şeklinde gösterilen 2.çoğul kişi iyelik eki de ekleneceği sözcüğe göre “-nız, -niz, -nuz, -nüz, -ınız, -iniz, -unuz, -ünüz” (hırkanız, iğneniz, kokunuz, ütünüz, gözünüz vb.) şekillerinde görülebilir.

Şekil 6.2.2’de görülen makine, bir isim köküne getirilebilecek çekim eklerini ve bunların sıralanışındaki kuralları belirler. Bu SDM’de (SDM: Sonlu Durum Makinesi) dikdörtgenler bir sözcüğün makine içerisinde bulunduğu durumları belirtirler. Daire içerisinde bulunan çıkışlar ise makinenin sonlanma durumlarını belirtirler. Makinenin bir tek başlangıç durumu “İsim Kökü” ve birden fazla sonlanma durumu vardır. Durumlar arası geçişler oklar ile belirtilmiştir. Oklar üzerinde yazılı ekler, makine içerisinde bir durumdan öteki duruma geçiş için sözcükte bulunması gereken ekleri belirtir. Örneğin “İsim Kökü” durumundan “İyelik (3)” durumuna geçilmesi için “-lArI” ekinin kök sonuna getirilmiş olması gerekir.

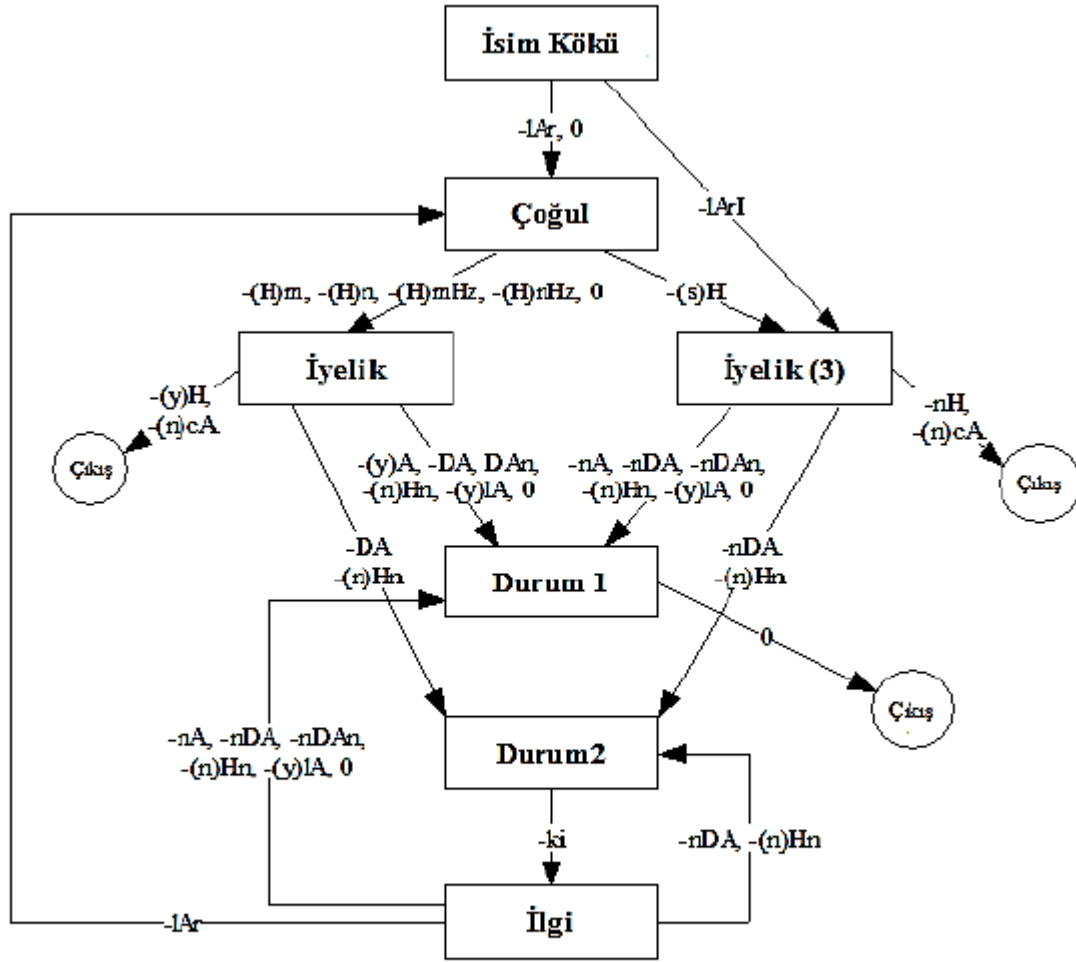
Şekil 6.2.1’de isim soylu bir sözcüğe eklenebilecek çekim ekleri numaralandırılmış ve örneklenmiştir. Tablodaki “Ek” isimli sütunda yer alan ek tanımlamalarında kullanılan büyük harfler ve parantezler şu anlamlara gelmektedirler:

- A: a veya e harfi yerine
- C: c veya ç harfi yerine
- D: d veya t harfi yerine
- H: ı, i, u veya ü harfi yerine
- I: ı veya i harfi yerine
- (): paranteziçerisindeki harf ek içinde yer almayabilir.

Örneğin, bu gösterimde “-lAr” şeklinde temsil edilen çoğul eki

ÖRN: Oyuncaklar \rightarrow Oyuncak-ları

İsim Kökü $^{-lArH} \rightarrow$ İyelik(3)



Şekil 6.2.2

Aynı şekilde “İsim Kökü” durumundan “Çoğul” durumuna geçilmesi için şu ekler bulunmalıdır: “-lAr, 0”. Bu gösterimde, oklar üzerinde yazılı eklerin yanında bulunan ‘0’ bir ek değildir. Durumlar arası boş geçişleri gösterir. Böylece başlangıç durumu olan “İsim Kökü” durumundan “-(H)m, -(H)n, -(H)mHz, -(H)nHz” ekleri ile geçiş yapılabilir. Bu eklerden önce kökün sonuna “-lAr” çoğul eki getirilmesi zorunlu değildir.

ÖRN: simitler \rightarrow simit-ler

İsim Kökü $^{-lAr} \rightarrow$ Çoğul

simitiniz \rightarrow simit – iniz

İsim Kökü $^0 \rightarrow$ Çoğul $^{-(H)nHz} \rightarrow$ İyelik

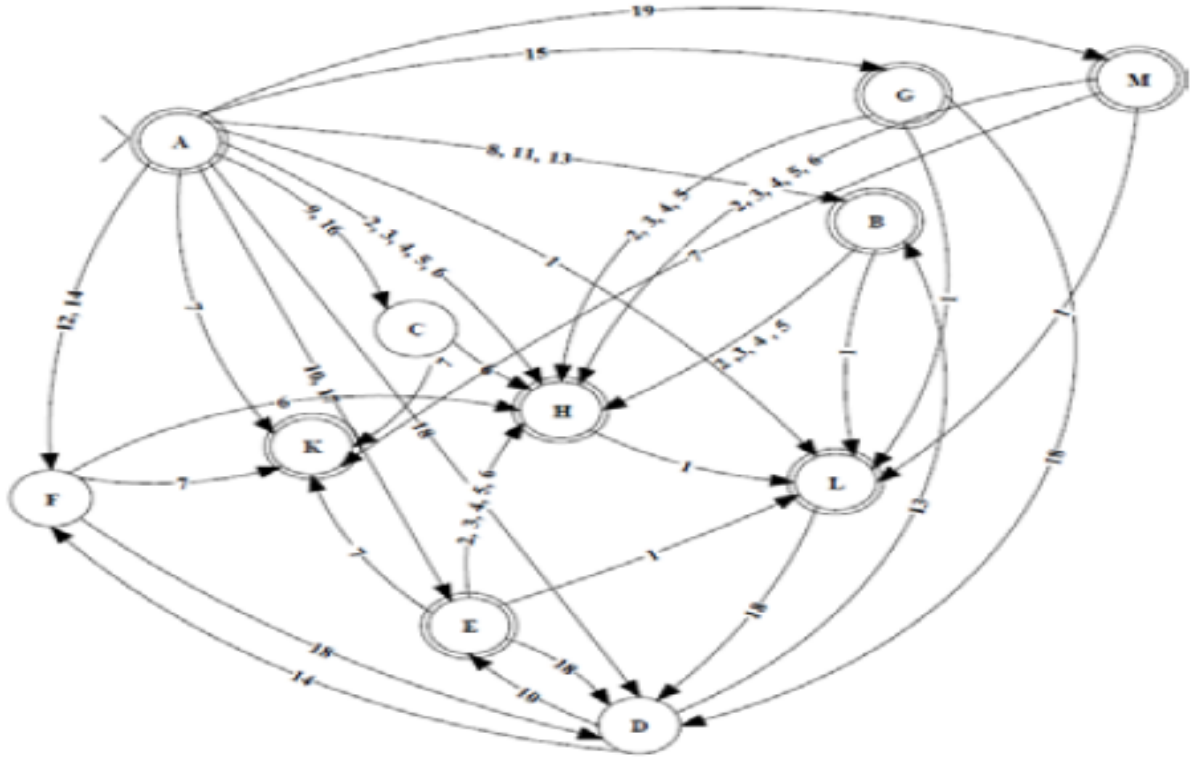
Bu sonlu durum makinesi ile yapılan analizde, isim soylu köke eklenen ekler ekleniş sıralarına göre incelenir ve “çıkış” durumuna ulaşılması ile sonlanır.

“sandıktakilerden” sözcüğünün İsim Çekim Ekleri Soldan Sağa SDM’si ile incelenmesi sırasında üzerinden geçilen durumlar aşağıda gösterilmiştir.

sandık – ta – ki – ler – den

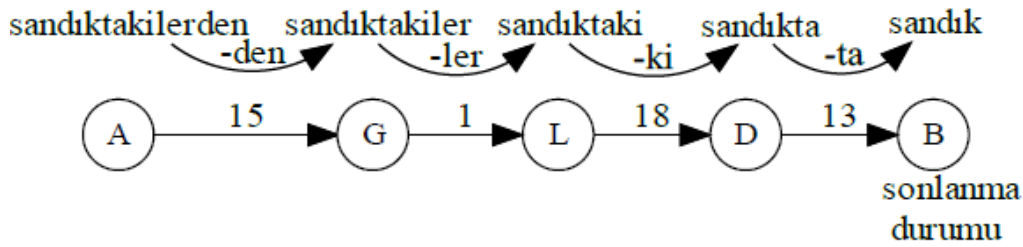
İsim Kökü $^0 \rightarrow$ Çoğul $^0 \rightarrow$ İyelik $^{-DA} \rightarrow$ Durum 2 $^{-ki} \rightarrow$ İlgi $^{-lar} \rightarrow$ Çoğul $^0 \rightarrow$ İyelik $^{-dan} \rightarrow$ Durum 1 $^0 \rightarrow$ Çıkış

Şekil 6.2.3’te Şekil 6.2.2’deki makinenin ters çevirme ve indirgeme işlemleri uygulanarak, gerekirci bir hale dönüştürülmesi ile ortaya çıkan ve eklerin sözcük sonundan başlanarak başa doğru sözcükten ayıklanmasını sağlayan sonlu durumlu makine gösterilmektedir. Bu gösterimde, durumlar daireler içerisinde, sonlanma durumu olabilen durumlar çift daire içerisinde, giriş durumları ise dairenin yanına büyük bir ok işareti konularak belirtilmiştir. Durumlar arası geçişlerin üzerinde yer alan sayılar, Şekil 6.2.1 İsim Çekim Ekleri tablosundaki eklerin numaralarıdır.

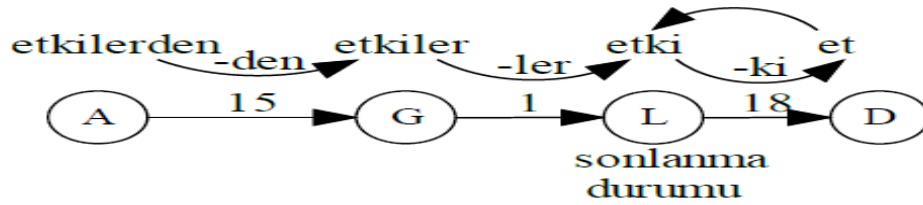


Şekil 6.2.3

“Sandıktakilerden” sözcüğünün bu makine ile incelenmesi sırasında üzerinden geçilecek durumlar aşağıdaki gibidir. İnceleme sonucu *sandık* isim kökü bulunmuştur:

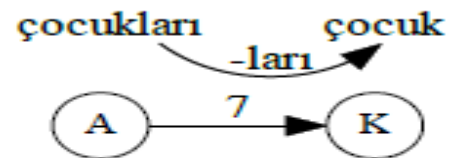
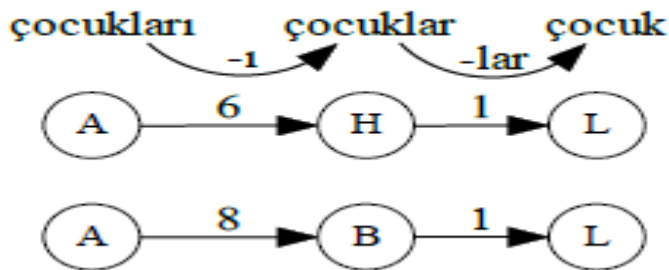


İnceleme sonuçlandığında üzerinde bulunulan son durum bir sonlanma durumu olmalıdır. Aksi takdirde, üzerinden geçilmiş olan en son sonlanma durumu, incelemenin durdurulduğu yer olarak kabul edilir. Buna örnek olarak *etkilerden* sözcüğünün incelenmesi gösterilebilir:



Bu örnekte en son durum olan 'D' durumu bir sonlanma durumu değildir. Bu sebeple, kendisinden bir önceki sonlanma durumu olan 'L', incelemenin sonlandığı nokta olarak belirlenir. Bulunan isim kökü "et" değil, "etki"dir.

Bazı sözcükler için birden çok biçimbilimsel analiz söz konusudur. Bu durum sözcük sonunda bulunan olası ekin birden fazla olduğu zamanlarda ortaya çıkar. Aşağıdaki örnekte *çocukları* sözcüğü için isim çekim ekleri kümesi için yaratılmış sonlu durum makinesi kullanılarak, üç farklı biçimbilimsel analiz yapılmıştır. Örnekte sözcük sonunda bulunan ilk ek *-ı* veya *-ları* eki olabilir. *-ı* eki '6' numaralı üçüncü tekil kişi iyelik eki olabileceği gibi, '8' numaralı ismin *-i* hali eki de olabilir. Bu sebeple aynı sözcük için üç farklı inceleme yapılabilir:



Ek-Eylem Ekleri Kümesi

Bu alt bölümde isim soylu sözcüklere eklenen ek-eylem ekleri için oluşturulan SDM tanıtılacaktır. Ek-eylem ekleri, bir isme getirilen zaman ve kişi eklerini içermektedirler. Şekil

6.2.4'te ek listesi açıklamalar ve örneklerle birlikte verilmektedir. İsim Çekim Ekleri bölümüne benzer şekilde bu bölümde de iki farklı gösterimde SDM verilecektir.

Aşağıdaki örneklerde *akıllıymışsın* ve *okuldaysalar* sözcüklerinin soldan sağa SDM ile incelenmeleri gösterilmektedir:

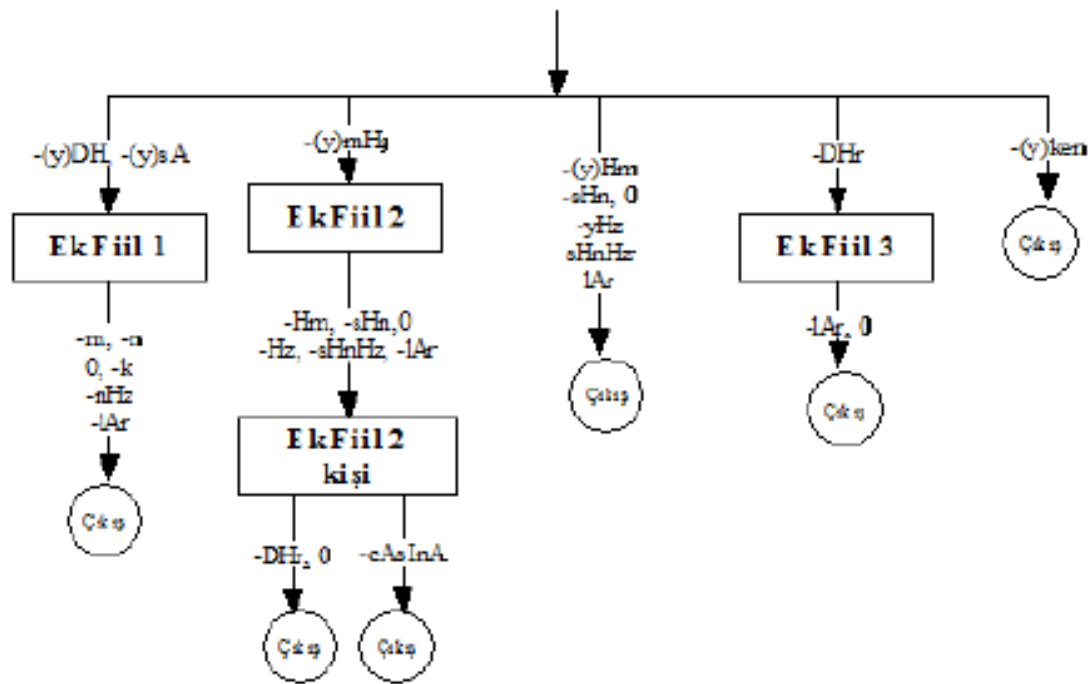
akıllı – ymış – sın

İsim Soylu Sözcük $^{-ymış} \rightarrow$ Ekfiil 2 $^{-sın} \rightarrow$ Ekfiil 2 kişi $^0 \rightarrow$ Çıkış

Okulda – ysa – lar

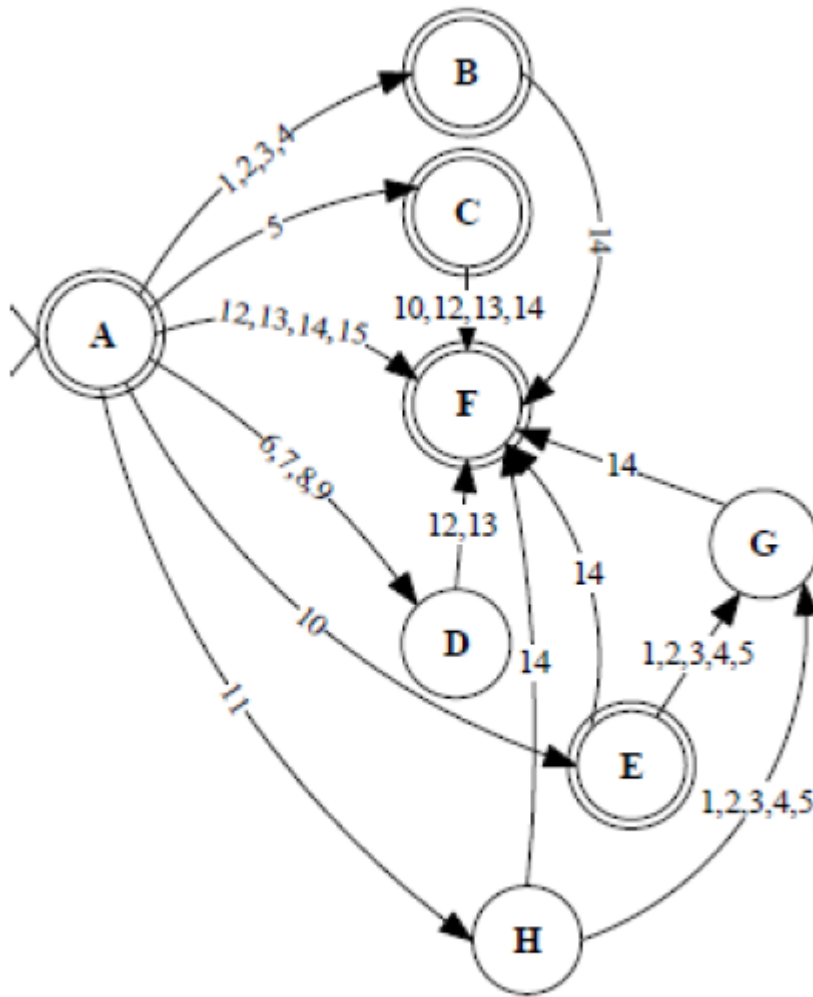
İsim Soylu Sözcük $^{-ysa} \rightarrow$ Ekfiil 1 $^{-lar} \rightarrow$ Çıkış

Ek-Eylem Ekleri Soldan Sağa Sonlu Durum Makinesi şeklindeki gibidir:



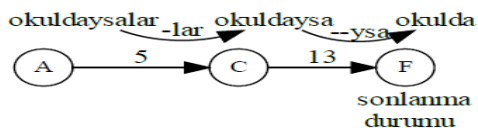
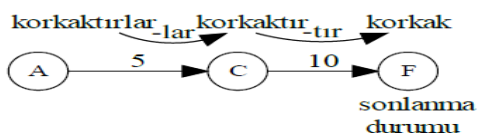
(Tabloda yer alan Ek Numaraları Şekil 6.2.5'te kullanılan makine için belirtilmiştir.)

Ek No	Ek	Açıklama	Örnek
1	-(y)Hm	1. tekil kişi	akıllı-yım
2	-sHn	2. tekil kişi	akıllı-sın
3	-(y)Hz	1. çoğul kişi	akıllı-yız
4	-sHnHz	2. çoğul kişi	akıllı-sınız
5	-IAr	3. çoğul kişi	akıllı-lar
6	-m	1. tekil kişi ((y)DH ve (y)sA eklerinden sonra)	akıllı-ydı-m
7	-n	2. tekil kişi ((y)DH ve (y)sA eklerinden sonra)	akıllı-ysa-n
8	-k	1. çoğul kişi ((y)DH ve (y)sA eklerinden sonra)	akıllı-ysa-k
9	-nHz	2. çoğul kişi ((y)DH ve (y)sA eklerinden sonra)	akıllı-ydı-nız
10	-DHr	çevrik kip	akıllı-dır
11	-cAsInA	tarz zarfı	akıllı-ymış-casına
12	-(y)DH	di'li geçmiş zaman	akıllı-ydı
13	-(y)sA	dilek-şart kipi	akıllı-ysa
14	-(y)mHş	miş'li geçmiş zaman	akıllı-ymış
15	-(y)ken	zaman zarfı	akıllı-yken



Şekil 6.2.5

Şekil 6.2.5'te gösterilen makine ile incelenen sözcüklerin sonunda ek-eylem eklerine rastlanması durumunda, bu ekler sözcükten çıkarılarak isim soylu sözcüğe ulaşılır. Elde edilen bu sözcük isim kökü olabileceği gibi, isim çekim ekleri almış isim soylu bir sözcük de olabilir. Aşağıdaki örneklerde incelenen *korkaktırlar* ve *okuldaysalar* sözcüklerinde elde edilen sonuç sırasıyla *korkak* ve *okulda* sözcükleridir. Bunlardan *korkak* bir isim kökü iken, *okulda* “okul” isim kökünün çekim eki almış halidir.



Eylem Zaman Ekleri Kümesi

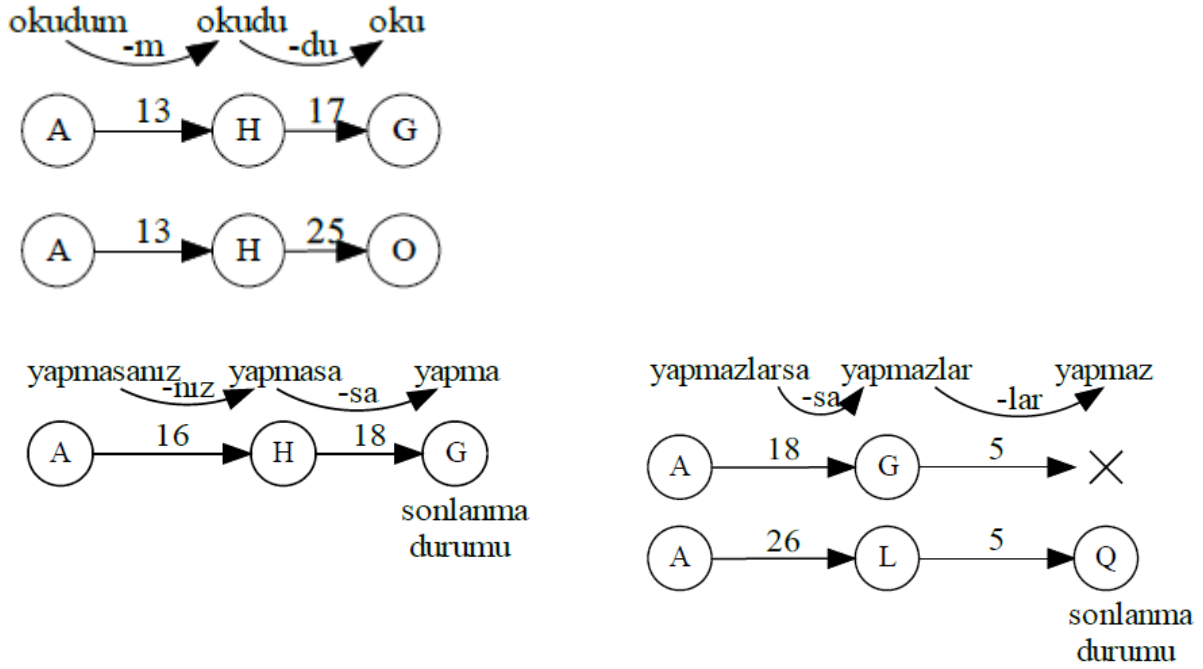
Eylem soylu sözcüklere eklenen çekim ekleri, eylem çekim ekleri ve eylem zaman ekleri olmak üzere iki kümeye ayrılabilir. Bu alt bölümde, eylem zaman ekleri kümesi için geliştirilen SDM tanıtılacaktır.

Ek No	Ek	Açıklama	Örnek
1	-(y)Hm	1. tekil kişi	gör-üyor-um
2	-sHn	2. tekil kişi	gör-üyor-sun
3	-(y)Hz	1. çoğul kişi	gör-üyor-uz
4	-sHnHz	2. çoğul kişi	gör-üyor-sunuz
5	-lAr	3. çoğul kişi	gör-üyor-lar
6	-mHş	miş'li geçmiş zaman	gör-müş
7	-(y)AcAk	gelecek zaman	gör-ecek
8	-(H)r	geniş zaman	gör-ür
9	-Ar	geniş zaman	iste-r
10	-(H)yor	şimdiki zaman	gör-üyor
11	-mAktA	sürerlilik	gör-mekte
12	-mAII	gereklilik	gör-meli
13	-m	1. tekil kişi	gör-dü-m
14	-n	2. tekil kişi	gör-dü-n
15	-k	1. çoğul kişi	gör-dü-k
16	-nHz	2. çoğul kişi	gör-dü-nüz
17	-DH	di'li geçmiş zaman	gör-dü
18	-sA	dilek-şart kipi	gör-se
19	-İIm	1. çoğul kişi	gör-e-lim
20	-(y)A	istek kipi	gör-e
21	-(y)HnHz	2. çoğul kişi	gör-ünüz
22	-(y)Hn	2. tekil kişi	gör-ün
23	-sHnlAr	3. çoğul kişi	gör-sünler
24	-DHr	çevrik kip	gör-müş-sün-dür
25	-(y)DH	hikaye bileşik zaman	gör-müş-tü-m
26	-(y)sA	şart bileşik zaman	gör-müş-se-m
27	-(y)mHş	rivayet bileşik zaman	gör-meli-ymiş
28	-cAsInA	tarz zarfı	gör-müş-cesine
29	-(y)ken	zaman zarfı	gör-müş-ken

Şekil 6.2.6

Üzerinde çalışılan bu kümede ekler, bir eyleme getirilebilecek zaman ve kişi eklerinden oluşmaktadır.

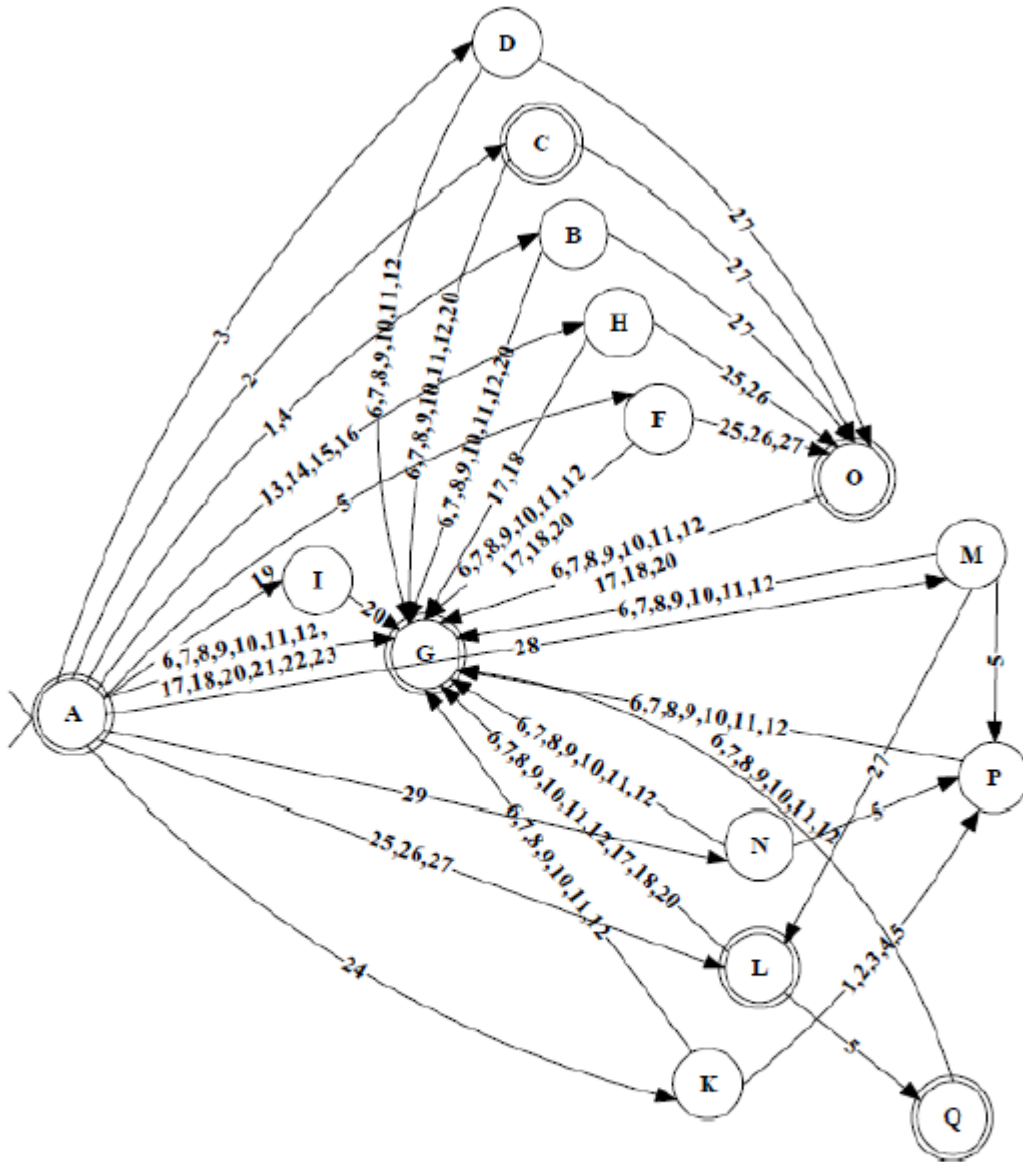
Şekil 6.2.7’de eylem zaman ekleri kümesi için oluşturulmuş soldan sağa SDM görülmektedir. Bu makinenin “olumsuz” ve “fiil kökü” durumları olmak üzere iki adet başlangıç durumu ve “çıkış” olarak adlandırılan birden çok sonlanma durumu vardır.



Bir eyleme bileşik zaman ekinin getirilebilmesi için kendisinden önce gelen bir normal zaman ekinin bulunması şarttır. Yukarıdaki örnekte *-du* eki veri tabanında arandığında iki farklı sonuç elde edilir: *-di*'li geçmiş zaman eki veya hikaye bileşik zaman eki. Bu sonuçların her ikisi de sonlanma durumuna geçiş sağlar. Ancak, 'O' durumu olumsuz eylemler için bir sonlanma durumu olduğundan, ilk inceleme doğru olarak kabul edilmesi gerektir. SDM eylemin olumsuz olup olmadığı kontrolünü yapamaz. Bu görev bir sonraki SDM olan Eylem Çekim Ekleri Kümesi'ne aittir. İki farklı inceleme sonucu üretilir. Diğer SDM'ler tarafından yapılan incelemelerde eylemin olumsuz olduğu belirlenemez ise 'O' çıkışlı inceleme kaydı silinir.

Yukarıdaki *yapmazlarsa* örneğinin birinci incelemesinde analiz 'G' durumuna kadar ilerleyebilir, '5' numaralı ek ile bu durumdan başka herhangi bir duruma geçiş yoktur. Elde edilen sözcük *yapmazlar*'dır. İkinci inceleme de ise 'Q' sonlanma durumuna kadar ulaşılmıştır, elde edilen sözcük *yapmaz*'dır.

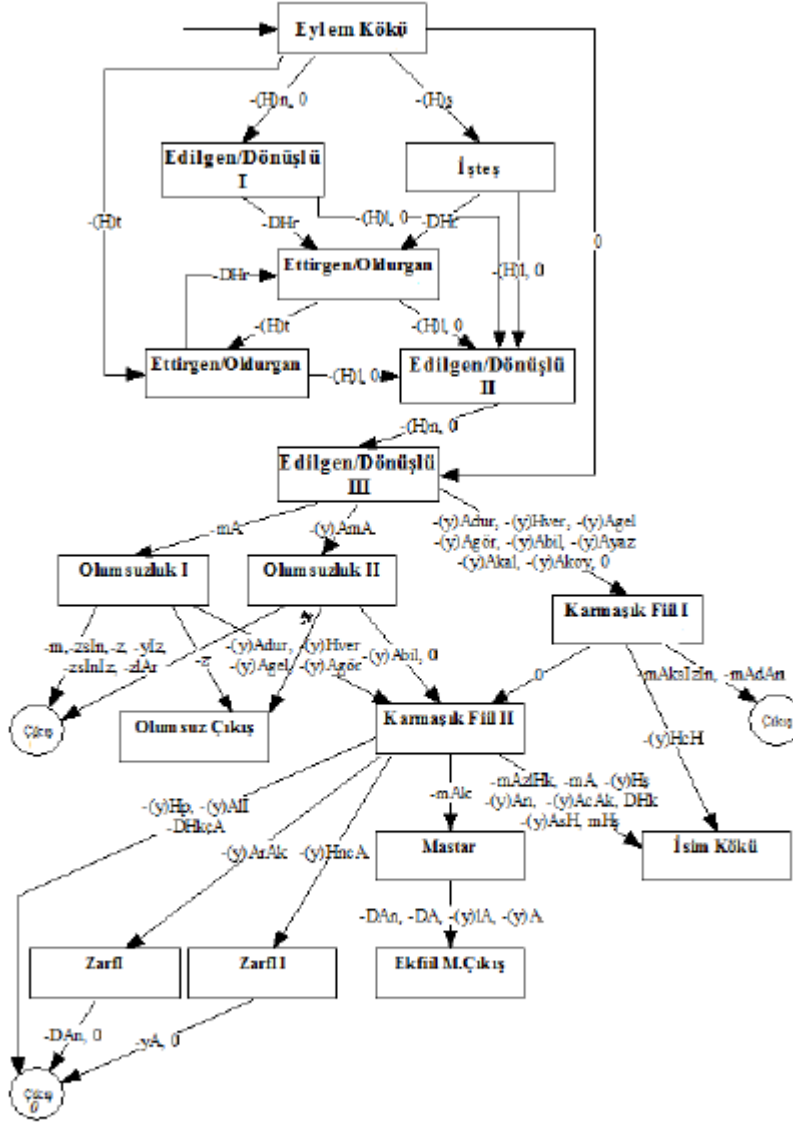
yapmasanız örneğinde ise olumsuz bir eylem için inceleme görülmektedir. Kullanılan *-sa* eki hem dilek-şart eki hem de şart bileşik zaman eki olabilir. Bu yüzden veri tabanında arama yapıldığında birden fazla sonuç geriye döner: '18' ve '26'.



**Şekil 6.2.8 Eylem Zaman Ekleri
Sağdan Sola Sonlu Durum
Makinesi**

Eylem Çekim Ekleri Kümesi

Kişi ve zaman ekleri dışında, eylemlere eklenen çekim ekleri, eylem çekim ekleri kümesi altında toplanmıştır. Bu küme, karmaşık fiil eklerini, tasvir fiil eklerinin ve çatı eklerini içerir. Bu alt bölümde eylem çekim ekleri kümesi için oluşturulmuş SDM tanıtılacaktır. Şekil



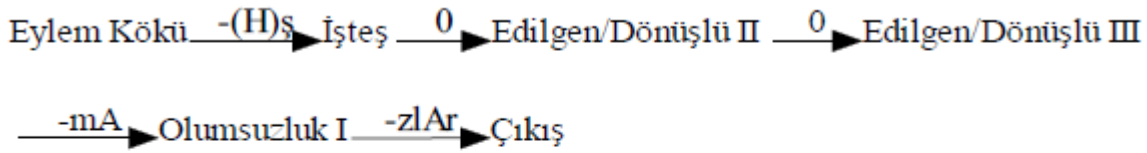
Şekil 6.2.9

okuyamazmışım sözcükleri üzerinde incelemeler yapılmıştır:

6.2.9’da verilen eylem çekim ekleri kümesi için oluşturulmuş soldan sağa SDM’de “Eylem Kökü” durumu makinenin giriş durumudur. Eylem çekim ekleri SDM’sinde, sonlanma durumları sadece “çıkış” durumları değildir. “Çıkış” durumlarının dışında, kendisinden farklı SDM’lere geçişi sağlayan sonlanma durumları vardır. Bunlar “Karmaşık Fiil II”, “İsim Kökü”, “Mastar”, “Ekfiil M.Çıkış” ve “Olumsuz Çıkış” durumlarıdır. Bunların her biri farklı SDM’lere geçişi sağlar.

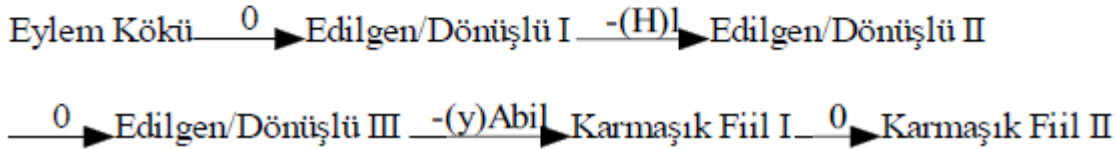
Farklı çıkışları tanıtmak üzere aşağıda görüşmezler, gönderilebilir, okuyucular, okumak, okumaktadır ve

gör - üş - me - zler



” Çıkış” normal sonlanma durumudur, inceleme bu SDM’de başlar ve sona erir.

gönder - il - e bil - ir



“Karmaşık Fiil II” durumu, bu SDM’den eylem zaman ekleri SDM’sine geçişi sağlayan kapıdır. Yukarıdaki örnekte *-ir* eki, eylem çekim eklerinde incelenecek bir ek değildir.

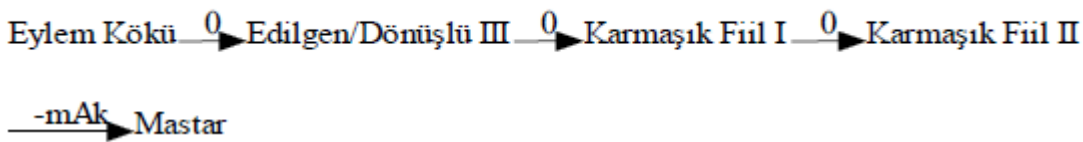
“Karmaşık Fiil II” sonlanma durumuna ulaşan inceleme, buradan eylem zaman ekleri SDM’sine geçer ve devam eder.

oku - yucu - lar



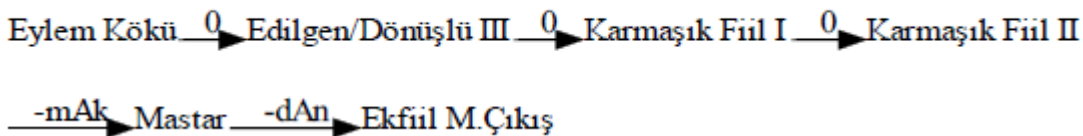
“İsim Kökü” sonlanma durumu bu SDM’sinden isim çekim ekleri SDM’sine geçişi sağlar. İnceleme bu aşamadan sonra isim çekim ekleri SDM’sinden devam eder.

oku - mak



“Mastar” normal sonlanma durumudur, inceleme bu SDM’de başlar ve sona erer.

oku - mak - tan - dır



“Ekfiil M.Çıkış” sonlanma durumu bu SDM’den eylem ekleri SDM’sine geçişi sağlar. Bu duruma gelmiş inceleme eylem ekleri SDM’sinden devam eder. “-dır” eki eylem ekleri SDM’sinde incelenir.

oku - yama - z - mış - ım

Eylem Kökü $\xrightarrow{0}$ Edilgen/Dönüşlü III $\xrightarrow{(y)AmA}$ Olumsuzluk II $\xrightarrow{-Z}$ Olumsuz Çıkış

“Olumsuz Çıkış” durumu bu SDM’den, eylem zaman ekleri SDM’sine geçişi sağlar.

Yukarıda örnekler ile açıklanan, eylem çekim ekleri SDM’sinden diğer SDM’lere geçişler şöyledir:

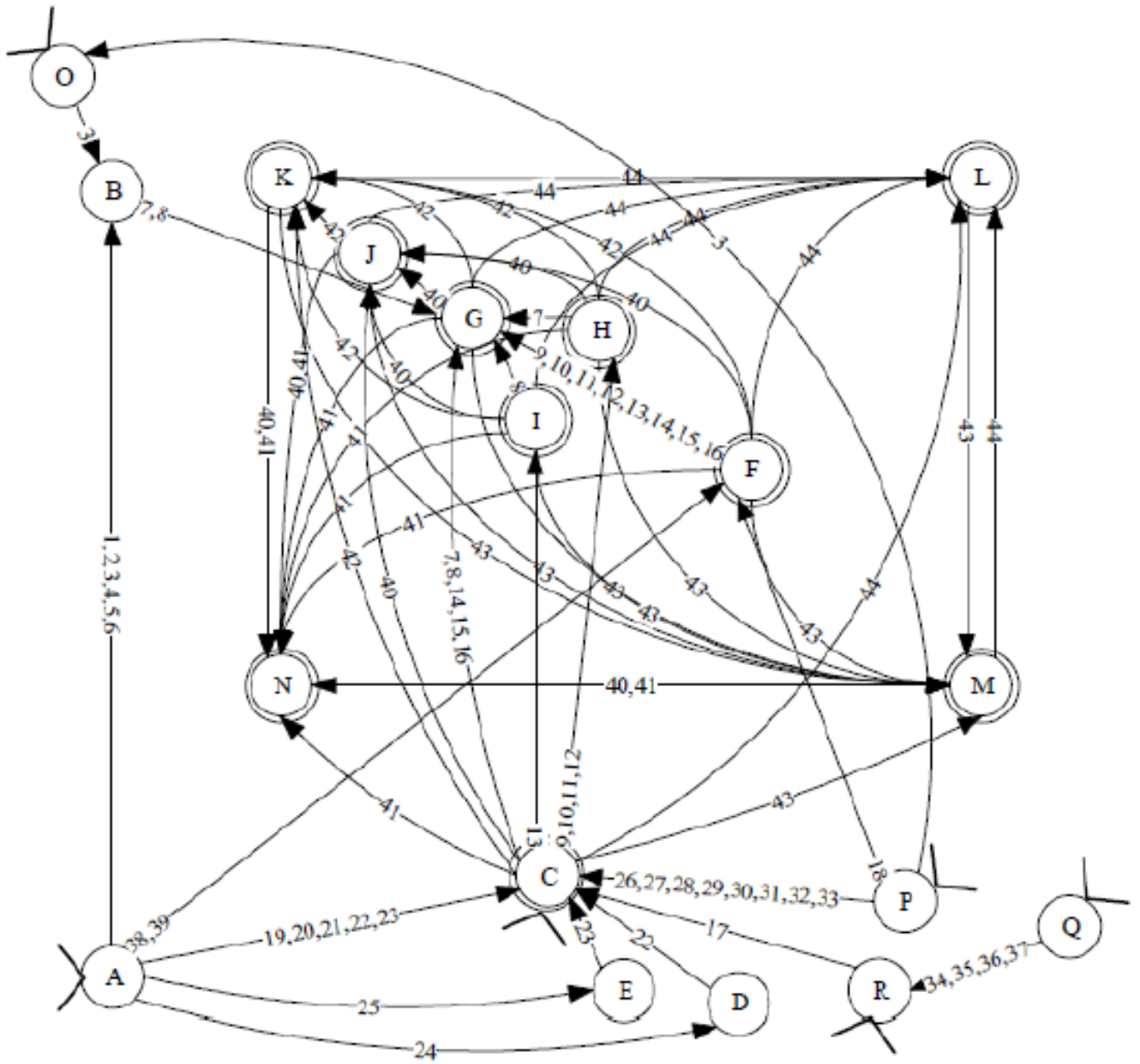
“Olumsuzluk Çıkış” Durumu \rightarrow Eylem Zaman Ekleri SDM’si

“Karmaşık Fiil I” Durumu \rightarrow Eylem Zaman Ekleri SDM’si

“İsim Kökü” Durumu \rightarrow İsim Çekim Ekleri SDM’si

Ekfiil M.Çıkış” Durumu \rightarrow Ekeylem Ekleri SDM’si

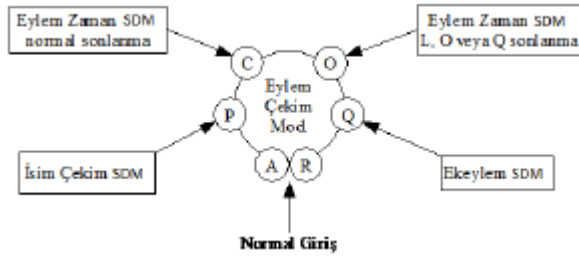
Eylem çekim ekleri için oluşturulmuş sağdan sola SDM Şekil 6.2.10’da gösterilmektedir. Bu makineden önce birçok farklı SDM çalışabileceğinde, makineye girişler öncesinde çalışan SDM’ye göre farklılık göstermektedirler. Bu girişlerin neler olduğu Şekil 6.2.11’de gösterilmektedir.



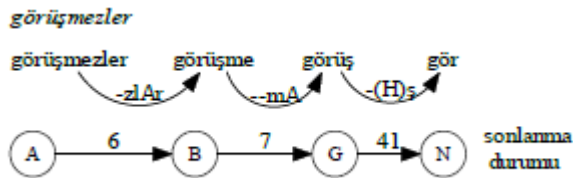
**Şekil 6.2.10 Eylem Çekim Ekleri
Sağdan Sola Sonlu Durum Makinesi**

Ek No	Ek	Açıklama	Örnek
1	-m	1. tekil kişi	yap-ma-m
2	-zsIn	2. tekil kişi	yap-ma-zsIn
3	-z	3. tekil kişi	yap-ma-z
4	-yIz	1. çoğul kişi	yap-ma-yIz
5	-zsInIz	2. çoğul kişi	yap-ma-zsInIz
6	-zIAr	3. çoğul kişi	yap-ma-zIAr
7	-mA	olumsuzluk	yap-ma
8	-(y)AmA	olumsuzluk	yap-ama
9	-(y)Adur	sürekli bileşik eylem kipi	yap-adur
10	-(y)Hver	tezlik bileşik eylem kipi	yap-iver
11	-(y)Agel	sürekli bileşik eylem kipi	ol-agel-di
12	-(y)Agör	sürekli bileşik eylem kipi	iste-yegör-sün
13	-(y)Abil	yeterlik bileşik eylem kipi	yap-abil
14	-(y)Ayaz	yaklaşma bileşik eylem kipi	düş-eyaz-dI-m
15	-(y)AkAl	sürekli bileşik eylem kipi	don-akAl
16	-(y)Akoy	sürekli bileşik eylem kipi	alı-koy-sun
17	-mAk	mastar	yap-mak
18	-(y)HoH	görev eki	yap-ıcı
19	-(y)Hp	zarf eki	yap-ıp
20	-(y)AlI	zarf eki	yap-alı
21	-DHkçA	zarf eki	yap-tıkça
22	-(y)ArAk	zarf eki	yap-arak
23	-(y)HncA	zarf eki	yap-ınca
24	-DAn	zarf eki	yap-arak-tan
25	-yA	zarf eki	yap-ınca-ya
26	-(y)An	sıfat fiil	oku-yan
27	-(y)AcAk	fiilden isim yapma eki	oku-yacak
28	-(y)AsI	sıfat fiil	oku-yası
29	-DHk	sıfat fiil	oku-duk
30	-mHş	sıfat fiil	oku-muş
31	-mAzIlk	fiilden isim yapma eki	oku-ma-mazlık
32	-mA	mastar eki	oku-ma
33	-(y)Hş	fiilden isim yapma eki	oku-yuş
34	-DAn	-den hali	oku-mak-tan
35	-DA	-de hali	oku-mak-ta
36	-(y)IA	birliktelik	oku-ma-yla
37	-(y)A	-e hali	oku-ma-ya
38	-mAksIzIn	zarf eki	oku-maksızın
39	-mAAdAn	zarf eki	oku-madan
40	-(H)n	edilgen dönüşlü	oku-n-mak
41	-(H)ş	işteş	gör-üş-mek
42	-(H)I	edilgen dönüşlü	gör-ül-mek
43	-DHr	oldurgan	oku-t-tur-mak
44	-(H)t	ettirgen	oku-t-mak

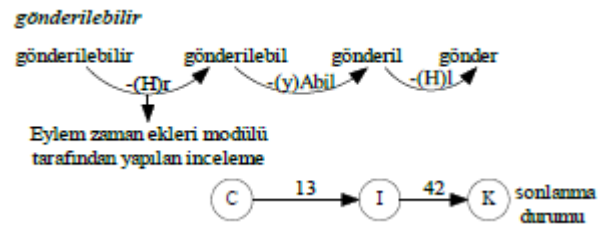
Şekil 6.2.11



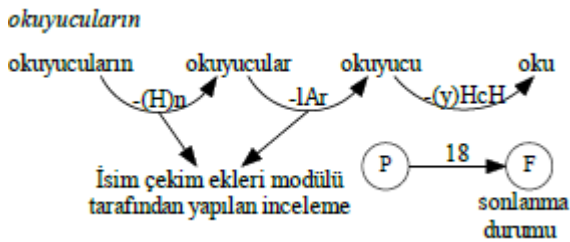
Yandaki şekilde Eylem Çekim Ekleri SDM'si Giriş Kapıları gösterilmektedir.



Yandaki örnekte eylem çekim ekleri SDM'sinden önce inceleme yapan başka bir SDM olmadığından giriş "A" durumundadır.

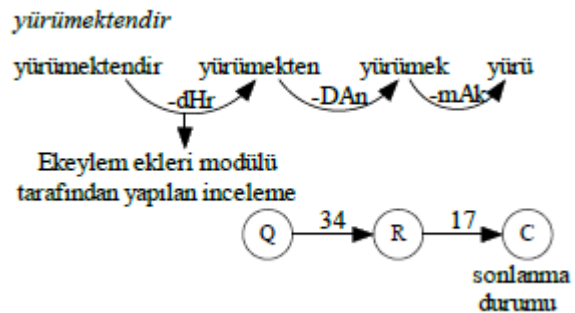


Yandaki örnekte, inceleme yapan bir önceki SDM eylem zaman ekleri SDM'si olduğundan, başlangıç olarak "C" durumu kabul edilir. Bu SDM'deki inceleme *gönderebil* sözcüğünden başlar.



Yandaki örnekte, bir önceki SDM isim çekim ekleri SDM'si olduğundan başlangıç durumu olarak 'P' durumu kabul edilir. Bu SDM'deki inceleme *okuyucu* sözcüğünden başlar.

Yandaki örnekte, eylem çekim ekleri SDM'sinden bir önce inceleme yapan SDM ekeylem ekleri SDM'sidir. Bu nedenle 'Q' durumu başlangıç durumu olarak kabul edilir. İnceleme, *yürümekten* sözcüğünden başlar.



BÖLÜM 7. TÜRKÇE DİLİNİN BİÇİMSEL YAPISINDAN YARARLANILARAK YAPILAN PROGRAMLAR

7.1. Koşullu Rastgele Alanlar ile Türkçe Haber Metinlerinin Etiketlenmesi

Bu çalışmada, Seda Kazkılınç ve Eşref Adalı, Doğal Dil İşleme, Bilgi Çıkarımı, Koşullu Rastgele Alanlar, Varlık İsmi Tanıma kullanılarak haber metinlerinden etiket çıkarmayı hedeflemektedir.

İnternetin yaygınlaşması ile birlikte artan belge sayısı, istenilen belgeye erişmedeki zorluğuda beraberinde getirmiştir. İstenilen belgeye daha kolay ulaşabilmek için belgeyi en iyi temsil eden söz öbeklerinin bulunması doğru bir yaklaşım olmaktadır. Metni okumadan metni en iyi tasvir eden söz öbeklerine ulaşmak kullanıcı ve tarayıcı açısından büyük önem taşımaktadır.

Çalışmanın amacı metni en iyi niteleyen özne, yüklem, yer ve zaman etiketlerinin metinden çıkarılmasıdır. Elde edilen bu etiketler sayesinde metnin konusu çıkarılabilir.

7.1.1. Koşullu Rastgele Alanlar

$$p_{\theta}(y|x) = \frac{1}{Z_{\theta}(x)} \exp \left\{ \sum_{t=1}^T \sum_{k=1}^K \theta_k f_k(y_{t-1}, y_t, x_t) \right\}$$

KRA, Lafferty ve arkadaşları tarafından önerilen istatistiksel dizilim sınıflandırmasına dayanan bir makine öğrenmesi yöntemidir. Dizilim sınıflandırıcıları bir dizilim içerisindeki her birime bir etiket atamaya çalışırlar. Olası etiketler üzerinde bir olasılık dağılımı hesaplar ve en olası etiket dizilimini seçerler.

Buna göre KRA modeli $p(y^*|x^*)$

$$Z_{\theta}(x) = \sum_{y \in Y^T} \exp \left\{ \sum_{t=1}^T \sum_{k=1}^K \theta_k f_k(y_{t-1}, y_t, x_t) \right\}.$$

olasılığını hesaplamak üzere geliştirilmiş

bir olasılık modeli olarak tanımlanabilir. Burada $y^* = y_1, \dots, y_n$ olası çıktı etiketlerini belirtirken, $x^* = x_1, \dots, x_n$ giriş verilerini belirtir. Buna göre KRA modeli bağıntı 1 ile gösterilebilir.

Burada Z_{θ} tüm olası etiket dizileri için normalleştirme faktörüdür ve bağıntı 2'deki gibi tanımlanır:

(2)

Burada, bağıntı 2'de de görüleceği üzere nitelik fonksiyonu parametreleri t. etiket yt ve t-1. etiket yt-1 ve sözcük dizilimi x olan bir fonksiyonudur. Nitelik fonksiyonları makine öğrenmesinde kullanmak istenilen nitelikleri belirleyen fonksiyonlardır.

7.1.2. Metinlerin Önişlenmesi

Bu çalışmada amaç, haber metinlerinden özne, yüklem, yer ve zaman etiketlerinin çıkarılmasıdır. Kapsam olarak Türkçe dilinde yazılmış metinler incelendiği için Türkçe diline özgü bir çözüm geliştirilmektedir. Türkçe dilinin eklemeli bir dil olması nedeniyle sözcük kökleri çok sayıda ek alabilmektedir. Bu nedenle ilk olarak sözcüklerin gövdelerine erişmek gerekir. Biçimbilimsel çözümleyici işlemi sonucu sözcüklerin ek ve kök bilgileri yanında sözcüklerin türlerine ait bilgilere de erişilmektedir. Bu işlem için Oflazer'in biçimbilimsel çözümleyicisi kullanılmıştır. Biçimbilimsel çözümleyiciden elde edilen çıktı Tablo 1'de verilmiştir.

Türkçe'de sözcüklerin ortalama iki adet biçimbilimsel çözümü bulunmaktadır. Bu nedenle en doğru çözümü bulabilmek için belirsizlik giderme işlemi yapılmıştır. Belirsizlik giderici olarak Sak tarafından hazırlanan belirsizlik giderici çıkarılmıştır. Belirsizlik giderici'nin girdi ve çıktılarına ait bir örnek Tablo 2'de verilmiştir.

Son olarak sözdizimsel çözümleme yapılarak sözcüklerin cümle içerisindeki görevlerine erişilebilmiştir. Eryiğit'in çalışmasından faydalanılmıştır. Sözdizimsel çözümleyiciye bir örnek Tablo 3'de verilmiştir.

Çizelge-1: Biçimbilimsel çözümleyiciye bir örnek

Çözümleyici Girdisi	Çözümleyici Çıktısı
Her	Her Her Noun+Prop+A3sg+Pnon+Nom
şey	şey şey+Noun+A3sg+Pnon+Nom
çok	çok çok+Adverb çok çok+Det çok çok+Adj çok çok+Postp+PCAbI
güzel	güzel güzel+Adj
olacak	olacak ol +Verb+Pos+Fut+A3sg olacak ol

Çizelge-2: Belirsizlik gidericiye bir örnek

Belirsizlik Giderici Girdisi	Belirsizlik Giderici Çıktısı
Her	Her Her Noun+Prop+A3sg+Pnon+Nom
şey	şey şey+Noun+A3sg+Pnon+Nom
çok	çok çok+Adverb
güzel	güzel güzel+Adj
olacak	olacak ol +Verb+Pos+Fut+A3sg
.	. . +Punc

7.1.3. Metinlerin İşlenmesi

Türkçe gibi kuralların bol olduğu bir dilde hedefe ulaşmak için bir çok kuralın tanımlanması gerekmektedir. Bu nedenle probleme makine öğrenmesi yöntemi ile yaklaşmak doğru bir yaklaşım olmaktadır.

Bu çalışmada amaç bir dökümana ait özne, yüklem, yer ve zaman etiketlerini bulmaktır. Bu bakış açısı ile problem dizilimlerden oluşan, bir sınıflandırma problemidir. Her bir sözcük ya belgenin öznesidir, ya yüklemidir, ya yeridir, ya zamanıdır yada bunlardan hiçbiridir. Tablo 4'de etiketler ve anlamları gösterilmiştir.

Sorun dizilim sınıflandırma problemi olarak ele aldığından ve sistemi eğitmedeki veriminden dolayı Koşullu Rastgele Alanlar yöntemi ile sistemi modelleme tercih edilmiştir. Bu çalışmada, doğrusal zincir Koşullu Rastgele Alanlar kullanılmıştır.

7.1.4. Niteliklerin Belirlenmesi

Koşullu Rastgele Alanlarda sistemi eğitmek için nitelikleri belirlemek gerekir. Kural tabanlı çalışmalarında elde ettikleri deneyimlerinin sonucu olarak, KRA için gerekli olan nitelikler belirlenmiştir.

7.1.4.1. Kural Tabanlı Nitelikler

Bu kısımda bahsedilen nitelikler, sorunu kural tabanlı olarak çözmek için yapılan araştırmalardan sonra çıkarılmıştır. Sadece kural tabanlı olarak sorunu çözmeye çalışarak,

Çizelge-4: Etiketler ve Anlamları

Etiket	Anlamı	Örnek
SUBJ	Metnin Öznesi	Mustafa Kemal
PRED	Metnin Yüklemi	Çıktı
LOC	Metnin Yeri	Samsun
DATE	Metnin Zamanı	19 Mayıs 1919
O	Yukarıdakilerden hiç biri	-

istenilen başarı oranına erişilememiştir. Fakat bu kurallardan bazılarının başarı oranını olumlu etkilediği gözlemlenmiş ve KRA için nitelik olarak kullanılmıştır.

7.1.4.2. Özne ve Yer Etiketleri için Kural Tabanlı Nitelikler

Bu çalışmada kullanılan metin kümesi, haber nitelikli olduğu için, özne etiketleri ile yer etiketleri genelde özel isimlerden oluşmaktadır. Daha önce değinildiği gibi özne tek bir sözcük olabileceği gibi sözcük öbeği de olabilir.

Aşağıda ayrıntıları anlatılan kurallar ile sözcüğün bir özel isim kümesine dahil olup olmadığı olasılığı bir nitelik olarak alınmıştır. Buna göre eğer bir sözcük özel isim kümesinin bir üyesi ise ve bu sözcük cümlede özne olarak kullanılmış ise 'POSSUB' isimli bir nitelikle tanımlanır. Eğer sözcük özel isim kümesinin bir üyesi ise ve bu sözcük cümlede dolaylı tümleç olarak kullanılmış ise bu sözcük 'POSLOC' olacak şekilde nitelik olarak tanımlanır.

Özel isim öbekleri

Türkçe’de yazım kuralları gereği özel isimler büyük harfle başlar. Büyük harfle başlayan her sözcük özel isim değildir. Buna en iyi örnek cümle başındaki sözcüktür.

Buna göre aşağıdaki kurallar çıkarılabilir.

Kural 1: Eğer bir sözcüğün ilk harfi büyük harf ise ve cümle başında değil ise özel isimdir.

Kural 2: Eğer bir sözcük cümlede ilk sözcüğü ise ve özel isim türü ile etiketlenmiş ise (Prop etiketli) özel isimdir.

Kural 3: Eğer iki özel isim arasında bağlaç var ise bu bağlaç özel isim öbeğine aittir. Örnek olarak, “Çalışma ve Sosyal Güvenlik Bakanlığı” , “Gençlik ve Spor Bakanlığı” verilebilir.

Özel isim öbekleri için sınır kuralları

Özel isim grupları çıkarılırken kullanılan sınır kuralları aşağıda detaylı anlatılmıştır.

Sınır Kuralı 1: Eğer özel bir isim –i halini almış bir sözcük ise bu sözcük özel isim öbeğinin son sözcüğüdür. Sadece son sözcüğü –i hali almış özel isim öbekleri içerisinde bağlaç barındırabilir. Örnek olarak, “Savunma ve Sosyal Güvenlik Bakanlığı”, “Öğrenci Seçme ve Yerleştirme Sınavı” verilebilir.

Sınır Kuralı 2: Herhangi bir noktalama işareti o sözcük öbeğinin bittiğine işaret eder. Bu sayede virgül “,” ve kesme “” işaretleri sadece sözcük öbeğindeki son sözcükte bulunabilir.

7.1.4.3. Biçimbilimsel nitelikler

Biçimbilimsel nitelikler biçimbilimsel çözümleyici ve sonrasında belirsizliği giderilmiş çözümler içerisinde seçilir. Temel olarak sözcük sınıfları ve ek sınıflarından oluşan bu nitelikler her bir sözcüğün biçimbilimsel belirsizlik gidericiden çıkmış hali işlenerek çıkartılır.

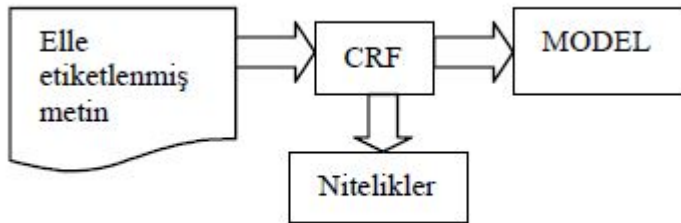
7.1.4.4. Sözdizimsel nitelikler

Sözdizimsel çözümleyiciden çıkan çözümlerden o sözcüğe ait sözdizimsel niteliği çıkarılır.

7.1.4.5. Yapısal nitelikler

Yapısal niteliklere veri seti içerisinde metnin sırası, metin içerisinde cümle sırası, sıklık ve ilk gözlemlendiği yer olarak ve büyük harf ile başlama verilebilir.

Nitelik Seçimi ve Performans İlişkisi



Şekil-1: KRA ile eğitim işlemi

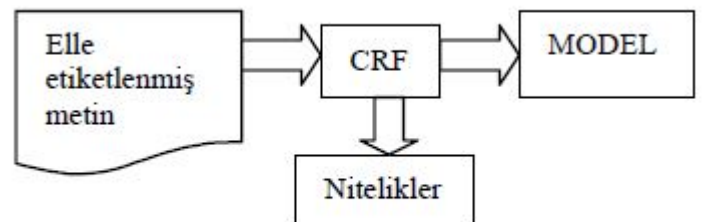
Nitelikleri seçerken Koşullu Rastgele Alanlar yönteminin bağıntısından da anlaşılacağı üzere nitelik fonksiyonlarının ağırlıkları normalize edilmiş şekildedir. Uygulamanın gerçekleşmesi kısmında da

bahsedileceği üzere fazla sayıda nitelik seçmenin sistemin performansına etkisi ihmal edilebilir düzeydedir. Bu nedenle Maksimum İlinti Minimum Fazlalık (MrMr) gibi nitelik seçme işlemine veya en çok etki eden nitelikleri bulma gibi bir yöntem başvurulmamıştır.

7.1.5. Koşullu Rastgele Alanlar'ın

Kullanımı

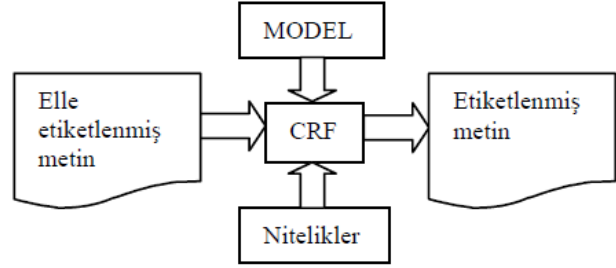
Elle etiketlenmiş sözcükler nitelikleri ile birlikte kaydedilerek, KRA



Şekil-1: KRA ile eğitim işlemi

modellenmesinde eğitim kümesi olarak kullanılmıştır. Böylece eğittikleri KRA yapısına hiç etiketlenmemiş belgeler girdi olarak verilmiş ve etiketlenmeleri sağlanmıştır. Şekil 1 ve Şekil 2'de tasarlanan eğitim ve sınama aşamaları gösterilmiştir.

Son olarak öğrenme kümesi kullanılarak eğittikleri KRA yapısına elle işaretlenmiş belgeler, giriş olarak uygulanmış geliştirdikleri yöntemin bulma ve tuturma olasılıklarıyla birlikte başarıımı ölçülmüştür.



Şekil-2: KRA ile sınama işlemi

7.1.6. Başarımın Ölçülmesi

Başarımı ölçerken bulma(precision) ve tuturma(recall) oranları ile tutturma ve bulmanın harmonik ortalamasından hesaplanan f-ölçümü hesaplanmıştır.

Bulma oranı sınama kümesi içinde var olan etiketlerden programın bulabildikleridir ve Bağıntı 3’de bulunabilir.

$$Bulma = \frac{\text{elle atanan etiketler içerisinde programın bulunduğu etiketler}}{\text{sınama kümesindeki etiketler}}$$

Tutturma oranı ise Bağıntı 4’de görüleceği üzere programın bulabildiği etiketlerin elle etiketlenmiş etiket listesinde olmasının oranıdır.

Çizelge-5: Her bir etiketin başarı oranı

Etiket Tipi	Elle Etiketlenen	Programın Bulduğu	Çakışan
Özne Etiketi	50	62	40
Yüklem Etiketi	50	64	39
Yer Etiketi	12	11	8
Zaman Etiketi	11	7	5
Etiket Tipi	Bulma	Tutturma	F-Ölçümü
Özne Etiketi	0,8	0,645	0,72
Yüklem Etiketi	0,78	0,61	0,684
Yer Etiketi	0,667	0,727	0,695
Zaman Etiketi	0,45	0,71	0,55

Sonuçların karşılaştırılması için kullanılan F-değeri bağıntı 5’de gösterilmiştir.

$$F = \frac{2 * tutturma * bulma}{tutturma + bulma}$$

Bu bilgiler ışığında çıkarılan sonuçlar Tablo 5’de gösterilmiştir.



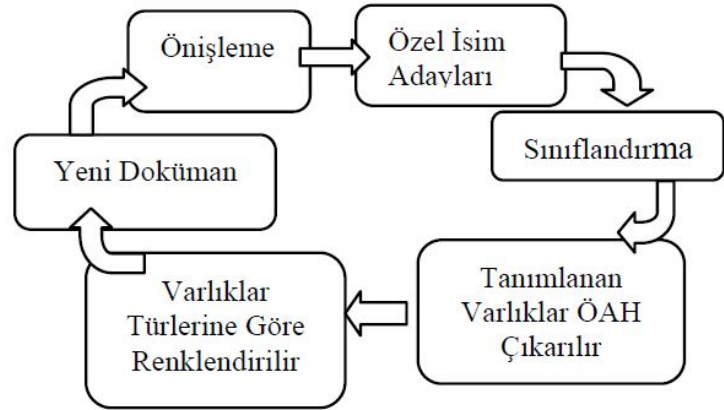
7.2. Türkçe Dokümanlar İçin Kural Tabanlı Varlık İsmi Tanıma

Bu çalışmada, Zeynep Banu

ÖZGER ve Banu DİRİ, hazırladıkları program ile döküman metninden kelime ya da kelime öbeklerinin hangi gruba ait olduğunu bulmayı hedeflemişlerdir.

7.2.1. Geliştirilen Sistemin Yapısı

Kural tabanlı tanıma için oluşturulan sistemin amacı, Türkçe dokümanlarda geçen kişi, yer ve kurum isimleri ile para, tarih ve saat varlıklarının bulunarak etiketlenmesidir. Sistemin gerçekleşmesi sırasında varlık isimlerini içerisinde tutan herhangi bir sözlüksel kaynak kullanılmamıştır. Şekil 1’de gösterildiği gibi oluşturulan sistem, içerikten bağımsız olacak şekilde tasarlanmıştır.



Şekil-1: Sistemin Yapısı

Sistemin kelimeleri birbirinden ayırt edebilmesi için kelimelerden

ve noktalama işaretlerinden sonra gelen boşluklardan yararlanılmıştır. Bu nedenle dokümanı işlemeye başlamadan önce, sistem dokümandaki noktalama işaretlerinden sonra boşluk olup olmadığını kontrol eder, eğer bir sonraki karakteri boşluk olmayan bir noktalama işareti ile karşılaşırsa boşluğu kendisi ekler.

Bu işlem önişleme olarak adlandırılmıştır. Önişleme adımı tamamlandıktan sonra, sistem büyük harf ile başlayan tüm kelimeleri "Özel İsim Adayı Havuzunda-ÖAH" saklar. Ardından tüm varlık isimleri için tanımlanmış olan kurallar tek tek çalıştırılarak etiketleme işlemi

gerçekleştirilir. Her bir varlık türü için tanımlanan kuralların uygulanması tamamlandığında, sistem sınıflandırdığı varlık isimlerini ÖAH'dan silmektedir. Böylece tüm varlık türlerinin sınıflandırılması işlemi bittiğinde, özel isim adayı havuzunda özel isim olmayan veya sistem tarafından etiketlenemeyen varlık isimleri kalmaktadır. Kullanıcı isterse tanımlanamayan varlık isimlerini el ile etiketleyebilir.

7.2.2. Varlık İsimleri İçin Tanımlanan Kurallar

Gerçekleştirilen sistem kural tabanlı bir yapı üzerine kurulmuş olup, her bir varlık için o türe ait belirleyici durumlar araştırılıp, kural haline getirilmiştir. Her bir varlık türü için belirleyici olan anahtar kelimeler tanımlanmıştır. Anahtar kelimelerin tanımlanmasında'daki çalışmadan yararlanılmış olup sistem eğitim verisi oluşturulurken gözlemlenen durumlara göre yeni kurallar ve yeni anahtar kelimeler eklenmiştir.

Son durumda sistem üzerinde tanımlı toplam 33 adet kural 335 adet anahtar kelime kullanılmıştır. Varlık isimlerinin çıkarılmasında her bir varlık ismi bağımsız olarak değerlendirilmiştir.

7.2.2.1. Tanımlama Kurallarına Örnekler

Projede kullanılan tanımlama kuralları ve kısıtları ile ilgili birer örnek verilmiştir:

Kişi İsimleri

Kişi isimlerinden önce gelen kelimeler araştırılarak (bay, başbakan, ağabey, tuğgeneral, diyen, eşi, annesi, diyetisyen, veren, üyesi, inanan, doğan, yaşındaki, babası, sayın, muhtar, temsilcisi, konuşan, vurgulayan, vd.) gibi 116 adet kişiyi tanımlayan kelime bulunmuştur.

Kişi İsimlerindeki Kısıtlar

Sistem, anahtar kelimedenden önce veya sonra gelen dört kelimeyi dikkate aldığından bir kişi ismi dörtten fazla kelimedenden oluşuyorsa, sadece ilk veya son dördü bulunabilmektedir (4'ün üzeri kelime seçildiğinde hata oranının arttığı tespit edilmiştir).

Kurum İsimleri

Kurum isimlerini niteleyen veya kurum isimlerinden sonra gelen kelimeler araştırılmış ve (gazetesi, kulübü, fabrikası, enstitüsü, bakanlığı, vakfı, başkanlığı, üniversitesi, lisesi, kurumu, müdürlüğü, komutanlığı, vd.) 79 tane anahtar kelime tespit edilmiştir.

Kurum İsimlerindeki Kısıtlar

Bazı durumlarda kısaltmalar kurum ismi olmamasına karşın kurum ismi olarak etiketlenmektedir. Ör: "TL".

Yer İsimleri (Location Names)

Yer isimlerinden önce gelen “kuzey, güney, doğu, batı, başkent, vd.” gibi kelimeler anahtar kelime olarak tespit edilmiş, öncesinde gelen ve ardışık olarak büyük harfle başlayan en fazla dört kelime yer ismi olarak etiketlenmiştir. Ör: Kuzey Amerika, başkent Ankara.

Yer İsimlerindeki Kısıtlar

Bir yer ismi dört kelimedenden uzunsa sistem sadece ilk veya son dördünü bulabilmekte ve yer ismi herhangi bir anahtar kelime içermiyorsa bulunamamaktadır.

Tarih Varlıkları

29.07.2011, 29-07-2011, 29/07/2011, 29.07.11, 29-07-11 ve 29/07/11 kullanımları için sonlu durum algoritmaları oluşturulmuştur.

Tarihler İçin Kısıtlar

-de ve -da ekleri hem yer hem de tarih varlıklarına eklenebilmektedir. Rakamlardan oluşan tarihlerin yer ismi gibi algılanmasını önlemek amacıyla hal ekini alan kelimenin rakam veya harflerden oluşmasına göre etiketleme gerçekleştirilir. Ör: "Temmuz' da".

Saat Varlıkları

15:20, 15.20; 5:20, 5.20 ve 15:20:10, 15.20.10 şeklindeki gösterimler için üç farklı sonlu durum algoritması tanımlanmış olup, bu formatlara sahip tüm saat varlıkları etiketlenmektedir. Ayrıca, saat 7 gibi gösterimler için de düzenleme yapılmıştır.

Saat Varlıkları İçin Kısıtlar

Yüzdelik veya reel sayı içeren dokümanlarda yanlış saat etiketlemeleri olabilmektedir. Ör: "...borsadaki 3.10'luk düşüş...".

Para Varlıkları

Bir para birimi dokümanda geçiyorsa, para biriminden önceki karakterler bir sonlu otomat algoritmasıyla incelenmektedir. Virgülden önce ve sonra gelen rakamların adedinde bir kısıtlama yapılmamaktadır. Ör: “2,300 TL”.

Para Varlıkları İçin Kısıtlar

Türkçe dokümanlar içerisinde para birimi içermeyen varlıklar yer alabilir. Ör: 60 milyar.

7.2.3. Sonuç

Tasarlanan sistem üzerinde yapılan testler sonucunda sırasıyla kurum, yer ve kişi isimlerinden %86, %83 ve %84 ortalama başarı elde edilmiştir. Sayısal varlık türlerinden de tarih, saat ve para varlıkları için sırasıyla %92, %94 ve %96 başarı alınmıştır.

BÖLÜM 8. SONUÇ

Türkçe dilinin morfolojik ve biçimbilimsel özellikleri üzerine araştırmalar yaparak Türkçe için biçimsel bir makinenin nasıl tasarlanabileceğini öğrendik. Makine öğrenmesi kullanılmadan bir sonlu durum makinesi ile doğruluk oranı %80 nin altında bir makine tasarlanabilir. Bunun sebebi Türkçe’de bir kökten yapım ve çekim ekleriyle beraber tahminen 1,5 milyon kelime türetilbilir olmasıdır. Türkçe dili için %100 doğrulukla çalışan bir makine tasarlamak günümüz şartlarıyla neredeyse imkansızdır.

KAYNAKLAR

- [1] Aksan, Y., Aksan, M., Koltuksuz, A., Sezer, T., Ümit Mersinli, Demirhan, U.U., Yilmazer, H., Atasoy, G., Öz, S., Ipek Yıldız, Özlem Kurtoglu: Construction of the Turkish National Corpus (TNC). In: N. Calzolari, K. Choukri, T. Declerck, M.U. Doğan, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis (eds.) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), İstanbul, Türkiye (2012)
- [2] Arısoy, E.: Statistical and discriminative language modeling for Turkish large vocabulary continuous speech recognition. Doktora Tezi, Boğaziçi Üniversitesi (2009)
- [3] Bilgin, O., Çetinoğlu, O., Oflazer, K.: Building a Wordnet for Turkish. Romanian Journal of Information Science and Technology 7(1-2), 163–172 (2004)
- [4] Buchholz, S., Marsi, E.: CoNLL-X shared task on multilingual dependency parsing. Proceedings of CoNLL, Sayfa 149–164 (2006)
- [5] Butt, M., Dyvik, H., King, T.H., Masuichi, H., Rohrer, C.: The parallel grammar project. Proceedings of the COLING-2002 Workshop on Grammar Engineering and Evaluation, Sayfa 1–7 (2002)
- [6] Çetinoğlu, O.: A large scale LFG grammar for Turkish. Doktora Tezi, Sabancı Üniversitesi (2009)
- [7] Çetinoğlu, O., Oflazer, K.: Integrating derivational morphology into syntax. In: N. Nicolov, G. Angelova, R. Mitkov (eds.) Recent Advances in Natural Language Processing. John Benjamins (2009)
- [8] Durgar-El-Kahlout, I.: A prototype English-Turkish statistical machine translation system. Doktora Tezi, Sabancı Üniversitesi (2009)
- [9] Durgar-El-Kahlout, I., Oflazer, K.: Exploiting morphology and local word reordering in English to Turkish phrase-based statistical machine translation. IEEE Transactions on Audio, Speech, and Language Processing 18(6), 1313–1322 (2010)
- [10] Eryiğit, G., Oflazer, K.: Statistical dependency parsing of Turkish. Proceedings of the 11th EACL, Sayfa 89–96. Trento, İtalya (2006)
- [11] Eryiğit, G., Nivre, J., Oflazer, K.: Dependency parsing of Turkish. Computational Linguistics 34(3), 357–389 (2008)
- [12] Hakkani-Tür, D., Oflazer, K., Tür, G.: Statistical morphological disambiguation for agglutinative languages. Computers and the Humanities 36(4) (2002)
- [13] Kerslake, C., Göksel, A.: Turkish: A Comprehensive Grammar. Comprehensive Grammars. Routledge (Taylor and Francis), New York, ABD (2005)
- [14] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual

Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, Sayfa 177–180. Association for Computational Linguistics, Prag, Çek Cumhuriyeti (2007)

[15] Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryiğit, G., Kübler, S., Marinov, S., Marsi, E.: Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering Journal* 13(2), 99–135 (2007)

[16] Oflazer, K.: Two-level description of Turkish morphology. *Literary and Linguistic Computing* 9(2), 137–148 (1994)

[19] Oflazer, K., Inkelas, S.: The architecture and the implementation of a finite state pronunciation lexicon for Turkish. *Computer Speech and Language* 20(1) (2006)

[20] Oflazer, K., Kuruöz, 'I.: Tagging and morphological disambiguation of Turkish text. *Proceedings of the Fourth Conference on Applied Natural Language Processing*, Sayfa 144–149. Association for Computational Linguistics, Stuttgart, Almanya (1994)

[21] Oflazer, K., Say, B., Hakkani-Tür, D.Z., Tür, G.: Building a Turkish treebank. A. Abeillé (ed.) *Treebanks: Building and Using Parsed Corpora*, Sayfa 261–277. Kluwer, Londra (2003)

[22] Oflazer, K., Tür, G.: Combining hand-crafted rules and unsupervised learning in constraintbased morphological disambiguation. E. Brill, K. Church (eds.) *Proceedings of the ACLSIGDAT Conference on Empirical Methods in Natural Language Processing* (1996)

[23] Sak, H., Güngör, T., Saraçlar, M.: Morphological disambiguation of Turkish text with perceptron algorithm. *CICLing 2007*, vol. LNCS 4394, Sayfa 107–118 (2007)

[24] Sak, H., Güngör, T., Saraçlar, M.: Resources for Turkish morphological processing. *Language Resources and Evaluation* 45(2), 249–261 (2011)

[25] Stamou, S., Oflazer, K., Pala, K., Christodoulakis, D., Cristea, D., Tufis, D., Koeva, S., Totkov, G., Dutoit, D., Grigoriadou, M.: Balkanet: A multilingual semantic network for: Balkan languages. *Proceedings of the 1st Global Wordnet Conference*. Mysore, Hindistan (2002)

[26] Wickwire, D.E.: The "sevmek thesis", a grammatical analysis of the Turkish verb system illustrated by the verb "sevmek"-to love. Master Tezi, Pacific Western Üniversitesi (1987)

[27] Yeniterzi, R., Oflazer, K.: Syntax-tomorphology mapping in factored phrase-based statistical machine translation from English to Turkish. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Sayfa 454–464. Association for Computational Linguistics, Uppsala, İsveç (2010)

[28] Yuret, D., Türe, F.: Learning morphological disambiguation rules for Turkish. *Proceedings of HLT/NAACL-2006*, Sayfa 328–334. New York, ABD (2006)

[29] Akın, M. D., & Akın, A. A. (2009). Zemberek - Açık Kaynak Kodlu Türkçe DDİ kütüphanesi. Aralık 2009 tarihinde <http://code.google.com/p/zemberek/>. adresinden alındı

[30] Beesley, K. R., & Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications.

[31] Brodda, B., & Fred, K. (1980). *An experiment with Automatic Morphological Analysis of Finnish*. Helsinki: University of Helsinki.

- [32] Cebiroğlu, G. (2002). Sözlüksüz Köke Ulaşma Yöntemi, Y.L.Tezi. İstanbul Teknik Üniversitesi.
- [33] Cebiroğlu, G., & Adalı, E. (2004). An Affix Stripping Morphological Analyzer For Turkish. International Conference on Artificial Intelligence and Applications, (s. 209-304). Innsbruck.
- [34] Cilden, E. K. (2009). Snowball. Aralık 2009 tarihinde <http://snowball.tartarus.org/algorithms/turkish/stemmer.html>. adresinden alındı
- [35] Eryiğit, G. (2006). Statistical Dependency Parsing of Turkish. EACL. Trento.
- [36] Eryiğit, G. (2006). Türkçe'nin bağıllık Ayırıştırması, Doktora Tezi. İstanbul: İstanbul Teknik Üniversitesi.
- [37] Gelbukh, A., & Sidorov, G. (2009). Approach to Construction of Automatic Morphological Analysis Systems for Inflective Languages with Little Effort. Computational Linguistics and Intelligent Text Processing (s. 157-162). içinde Heidelberg: Springer Berlin.
- [38] Hakkani-Tür, D., Oflazer, K., & Tür, G. (2002). Statistical Morphological Disambiguation for Agglutinative Languages. Journal of Computers and Humanities , 36 (4).
- [39] Hankamer, J. (1986). Finite state morphology and left to right. Fifth West Coast Conference on Formal Linguistics, (s. 29-34). Stanford.
- [40] Hirsimäki, T., Creutz, M., Siivola, V., Kurimo, M., Pytkönen, J., & Virpioja, S. (2006). Unlimited vocabulary speech recognition with morph language models applied to Finnish. Computer Speech and Language , 20 (4), 515-541.
- [41] Johnson, C. D. (1972). Formal aspects of phonological description. The Hague: Mouton.
- [42] Jurafsky, D., & Martin, J. H. (2009). Speech and Language Processing. New Jersey: Pearson Education, Inc.
- [43] Kaalep, H.-J. (1997). An Estonian Morphological Analyzer. Computers and Humanities , 31 (2), 115-133.
- [44] Kaplan, R., & Kay, M. (1981). Phonological rules and finite-state transducers. 56th Annual Meeting of the Linguistic Society of America. New York.
- [45] Karttunen, L., & Beesley, K. R. (2001). A Short History of Two-Level Morphology. ESSLLI-2001 Special Event "Twenty Years of Finite-State Morphology". Helsinki.
- [46] Koskenniemi, K. (1983). Two-level morphology: a general computational model for word-form recognition and production. Department of General Linguistics. Helsinki: University of Helsinki .
- [47] Köksal, A. (1975). A First Approach to a Computerized Model for the Automatic Morphological Analysis of Turkish, Doktora Tezi. Ankara: Hacettepe Üniversitesi.

- [48] Kurimo, M., Creutz, M., Varjokallio, Arisoy, E., & Saraclar, M. (2005). Unsupervised Segmentation of Words into Morphemes Challenge 2005, An Introduction and Evaluation Report. In Proceedings of PASCAL Challenge Workshop on unsupervised segmentation of words into morphemes. Venice.
- [49] Oflazer, K. (2003). Dependency parsing with an extended finite-state approach. *Computational Linguistic* , 29 (4), 515-544.
- [50] Oflazer, K. (2009). Türkçe Sözcük Veritabanı projesi. Aralık 2009 tarihinde <http://www.hlst.sabanciuniv.edu/TL/>. adresinden alındı
- [51] Oflazer, K., Say, B., Tür, D. Z., & Tür, G. (2003). Building a Turkish treebank. A. Abeille içinde, *Treebanks: Building and Using Parsed Corpora*. Kluwer.
- [52] Packard, D. W. (1973). Computer-assisted morphological analysis of ancient Greek. *International Conference on Computational Linguistics*. Pisa.
- [53] Sak, H., Güngör, T., & Saraçlar, M. (2008). Turkish Language Resources: Morphological Parser, Morphological Disambiguator and Web Corpus. *GoTAL*. Springer.
- [54] Sproat, R. (1992). *Morphology and computation*. Cambridge: MIT Press.
- [55] Yılmaz, S. (2009). Türkçe için iyileştirilmiş biçimbilimsel çözümleyici, Y.L. Tezi. İstanbul: İstanbul Teknik Üniversitesi.
- [56] Silviu Cucerzan and David Yarowsky, Language Independent Named Entity Recognition, Combining Morphological and Contextual Evidence. s. 90-99, 1999
- [57] Soderland, S., Fisher, D., Aseltine, J. ve Lehnert, W., 1995, CRYSTAL: Inducing a Conceptual Dictionary,
- [58] Bikel, D.M., Miller, S., Schwartz, R. ve Weischedel R., 1997. Nymble: a high-performance learning name-finder, Proceedings of the fifth conference on Applied natural language processing, ANLC '97, Association for Computational Linguistics, Stroudsburg, PA, USA, s.194–201
- [59] NetOwl Server, Proceedings of the fifth conference on Applied natural language processing, ANLC '97, Association for Computational Linguistics, Stroudsburg, PA, USA, s.15–16
- [60] Kucuk, D. ve Yazici, A., 2009. Named Entity Recognition Experiments on Turkish Texts, Proceedings of the 8th International Conference on Flexible Query Answering Systems, FQAS '09, Springer-Verlag, Berlin, Heidelberg,
- [61] Özkan Bayraktar, 1991. Local Grammar, Person Name Recognition in Turkish Financial Texts by Using Local Grammar Approach, METU, s.19–27.
- [62] Tür, G., Hakkani-tür, D. ve Oflazer, K., 2003. A statistical information extraction system for Turkish, *Nat. Lang. Eng.*, 9(2), 181–210,

- [63] Nallapati, R., Allan, J. ve Mahadevan, S., Extraction of Key Words from News Stories.
- [64] Cohen, J.D., 1995. Highlights: Language- and Domain-Independent Automatic Indexing Terms for Abstracting, *Journal of the American Society for Information Science*, 46(3), 162–174.
- [65] Matsuo, Y. ve Ishizuka, M., 2004. Keyword extraction from a single document using word co-occurrence statistical information, *International Journal on Artificial Intelligence Tools*, 13(1), 157–169
- [66] van der Plas, L., Pallotta, V., Rajman, M. ve Ghorbel, H., 2004. Automatic Keyword Extraction from Spoken Text. A Comparison of two Lexical Resources: the EDR and WordNet, CoRR, cs.CL/0410062
- [67] Hulth, A., 2003. Improved automatic keyword extraction given more linguistic knowledge, *Proceedings of the 2003 conference on Empirical methods in natural language processing, EMNLP '03*, Association for Computational Linguistics, Stroudsburg, PA, USA, s.216–223
- [68] Pala, N. ve Çiçekli, I., 2007. Turkish Keyphrase Extraction Using KEA, in, *Proceedings of the 22nd International Symposium on Computer and Information Sciences (ISCIS 2007)*.
- [69] Wang, J., Peng, H. ve Hu, J.s., 2006. Automatic keyphrases extraction from document using neural network, *Proceedings of the 4th international conference on Advances in Machine Learning and Cybernetics, ICMLC'05*, Springer-Verlag, Berlin, Heidelberg, s.633–641
- [70] Quinlan, J.R., 1993. C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA.
- [71] Cicekli, I. ve Kalaycilar, F., 2008. TurKeyX: Turkish Keyphrase Extractor, *Proceedings of the 23rd International Symposium on Computer and Information Sciences, TeX Users Group*, s.84–89.
- [72] Lafferty, J., McCallum, A. ve Pereira, F., 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, *Proc. 18th International Conf. on Machine Learning*, Morgan Kaufmann, San Francisco, CA, s.282–289.
- [73] Oflazer, K., 1993. Two-level description of Turkish morphology, *Proceedings of the sixth conference on European chapter of the Association for Computational Linguistics, EACL '93*, Association for Computational Linguistics, Stroudsburg, PA, USA, s.472–472,
- [74] Eryiğit, G., 2007. ITU Treebank Annotation Tool, *Proceedings of the ACL workshop on Linguistic Annotation (LAW 2007)*, Prague.
- [75] Ozkaya, S.; Diri, B., "Named Entity Recognition by Conditional Random Fields from Turkish informal texts," *Signal Processing and Communications Applications (SIU)*, 2011 IEEE 19th Conference on , vol., no., pp.662-665, 20-22 April 2011

- [76] Satoshi, S., 1998. Nyu: Description of the Japanese NE System Used For Met-2, The 7th Message Understanding Conference, Virginia, USA.
- [77] Borthwick, A., Sterling, J., Agichtein, E. ve Grishman, R., 1998. NYU: Description of the MENE Named Entity System as used in MUC-7, Message Understanding Conference, Virginia, USA.
- [78] Masayuki, A. ve Matsumoto, Y., 2003. Japanese Named Entity Extraction with Redundant Morphological Analysis, The North American Chapter of the Association for Computational Linguistics on Human Language Technology, NAACL.
- [79] McCallum, A. ve Li, W., 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Features Induction and Web-Enhanced Lexicons, Computational Natural Language Learning- CoNLL, Edmonton, Canada.
- [80] Brin, S., 1999. Extracting Patterns and Relations from the World Wide Web, The World Wide Web and Databases - LNCS, Vol. 1590, 172-183.
- [81] Cucchiarelli, A. ve Velardi, P., 2001. Unsupervised Named Entity Recognition Using Syntactic and Semantic Contextual Evidence, Computational Linguistics, Vol. 27:1, 123-131, Cambridge: MIT Press.
- [82] Etzioni, O., Cafarella, M. J., Downey, D., Popescu, A. M., Shaked, T., Soderland, S., Weld, D. S. ve Yates, A., 2006. Unsupervised Named-Entity Extraction from the Web: An Experimental Study, Artificial Intelligence, Vol. 165, 91-134, Essex: Elsevier.
- [83] Rau, L., 1991. Extracting Company Names from Text, The 7th IEEE Conference on Artificial Intelligence Applications, 29-32.
- [84] Lee, S. ve Lee, G. G., 2005. Heuristic Methods for Reducing Errors of Geographic Named Entities Learned by Bootstrapping, Natural Language Processing, Korea, 658-669.
- [85] Fleischman, M. ve Hovy, E., 2002. Fine Grained Classification of Named Entities, The 19th International Conference on Computational Linguistics, Taiwan.
- [86] Narayanaswamy, M., Ravikumar, K. E. ve Vijay-Shanker, K., 2003. A Biological Named Entity Recognizer, The 8th Pacific Symposium on Biocomputing, Hawaii, USA, 427-438.
- [87] Rindfleisch, T. C., Tanabe, L. ve Weinstein, J. N., 2000. EDGAR: Extraction of Drugs, Genes and Relations from the Biomedical Literature, The 6th Pacific Symposium on Biocomputing, Hawaii, USA, 517-528.
- [88] Sekine, S. ve Nobata, C., 2004. Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy, The 4th International Conference on Language Resources and Evaluation, Lisbon, Portugal.
- [89] Maynard, D., Tablan, V., Ursu, C., Cunningham, H. ve Wilks, Y., 2001. Named Entity Recognition from Diverse Text Types, Recent Advances in Natural Language Processing, Bulgaria.

- [90] Minkov, E., Wang, R. ve Cohen, W., 2005. Extracting Personal Names from Email: Applying Named Entity Recognition to Informal Text, Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, Canada.
- [91] Yu, S., Bai S. ve Wu, P., 1998. Description of the Kent Ridge Digital Labs System Used for MUC-7, The 7th Message Understanding Conference, Virginia, USA.
- [92] Poibeau, T., 2003. The Multilingual Named Entity Recognition Framework, The 10th Conference on European Chapter of the Association for Computational Linguistics-EACL, Budapest, Hungary.
- [93] Silva, D., Ferreira, J., Kozareva, Z., Gabriel, J. ve Lopes, P., 2004. Cluster Analysis and Classification of Named Entities, The 4th International Conference on Language Resources and Evaluation, Lisbon, Portugal.
- [94] Whitelaw, C. ve Patrick, J., 2003. Evaluating Corpora for Named Entity Recognition Using Character-Level Features, The 16th Australian Conference on Artificial Intelligence-AI, 910-921, Perth, Australia.
- [95] Piskorski, J., 2004. Extraction of Polish Named-Entities, The 4th International Conference on Language Resources and Evaluation, Lisbon, Portugal.
- [96] Popov, B., Kirilov, A., Maynard, D. ve Manov, D., 2004. Creation of Reusable Components and Language Resources for Named Entity Recognition in Russian, The 4th International The Conference on Language Resources and Evaluation, Lisbon, Portugal.
- [97] Cucerzan, S. ve Yarowsky, D., 1999. Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence, The Joint Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, USA 90-99.
- [98] Bayraktar, Ö. ve Taşkaya-Temizel, T., 2008. Person Name Extraction From Turkish Financial News Text Using Local Grammar-Based Approach, The 23rd International Symposium on Computer and Information Sciences-ISCIS, Istanbul, Turkey.
- [99] Küçük, D. ve Yazıcı, A., 2008. Identification of Coreferential Chains in Video Texts for Semantic Annotation of News Videos, The 23rd International Symposium on Computer and Information Sciences-ISCIS, Istanbul, Turkey.
- [100] Küçük, D. ve Yazıcı, A., 2009. Exploiting Information Extraction Techniques for Semantic Annotation of Videos in Turkish, The 14th International Conference on Applications of Natural Language to Information Systems-NLDB, Saarland, Germany.
- [101] <http://www.ce.yildiz.edu.tr/personal/zbanu/file>

ÖZGEÇMİŞ

Taşkın Temiz, 10.05.1992 de İstanbul’da doğdu. İlk ve orta eğitimini Küçükçekmece’de tamamladı. Lise eğitimini öncelikle Hadımköy Örfi Çetinkaya Anadolu Lisesi’nde sonrasında Bakırköy Yahya Kemal Beyatlı Lisesi’nde alarak 2010 yılında tamamladı. 2010 yılında başladığı Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü’de eğitimini sürdürmekte.

Egemen Zeytinci, 09.11.1993 de Aydın’da doğdu. İlk ve orta eğitimini Aydın’da tamamladı. Lise eğitimini öncelikle Aydın Emel Mustafa Uşaklı Anadolu Lisesi’nde sonrasında Aydın Atatürk Anadolu Lisesi’nde alarak 2011 yılında tamamladı. 2012 yılında başladığı Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü’de eğitimini sürdürmekte.