



# Doğal Dil İşlemeye Giriş

Dr. Öğr. Üyesi Hayri Volkan Agun

Bilgisayar Mühendisliği

# İçerik

- ☐ Düzenli ifadeler
- ☐ Düzenli ifadelerde
- ☐ Düzenli ifade örnekleri
- ☐ Chomsky hiyerarşisi



# Düzenli İfadeler

- ☐ Cümleler, kelimeler, kelimelere eklenen ekler ardışık olarak belirli karakter dizgelerinden oluşurlar.
- ☐ Bu karakterler içerisinde geçen ardışık anlamlı öbeklerden oluşur.
- ☐ Örneğin:
  - ☐ 'Takım kaptanı Gökhan 15 Haziran 2013 saat 08:05 de otobüse bindi.' cümlesinde geçen farklı öbekler:
    - ☐ 08:05 aslında bir öbeğdir. Benzer saat öbekleri 12:25, 24:00
    - ☐ 15 Haziran 2013 aslında bir öbeğdir. Benzer tarih öbekleri 12 Temmuz 2020, 23 Nisan 1919, 16 Mayıs 1233, ...
    - ☐ Takım kaptanı aslında bir öbeğdir. Benzer isim öbekleri Gemi kaptanı, Uzakyol kaptanı, Yat kaptanı, Seyrüsefer Kaptanı.
    - ☐ Gökhan aslında bir öbeğdir. Benzer isim öbekleri Hunhan, Barkhan, Tunhan, ...

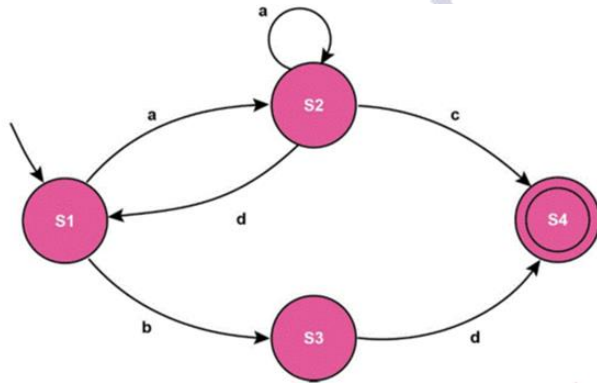
# Düzenli İfadeler

- ❑ Düzenli ifadeleri kullanarak benzer olan öbekleri metin içerisinde arayabiliriz.
- ❑ Metin içerisinde arama için en basit yöntem karakterlere bakarak yapılan aramadır.
- ❑ Belirli bir karakterden başlayarak metin içerisinde aranan öbeğin geçip geçmediğine bakılır.
- ❑ Ancak bu yöntem her bir karakter ile aranan öbekteki her bir karakterin kıyaslamasını içerdiğinden dolayı  $O(n * m)$  zaman karmaşıklığına sahiptir.
- ❑  $O(n * m)$  zaman karmaşıklığında  $n$  metin uzunluğu ve  $m$  ise aranan öbek uzunluğudur.

2010

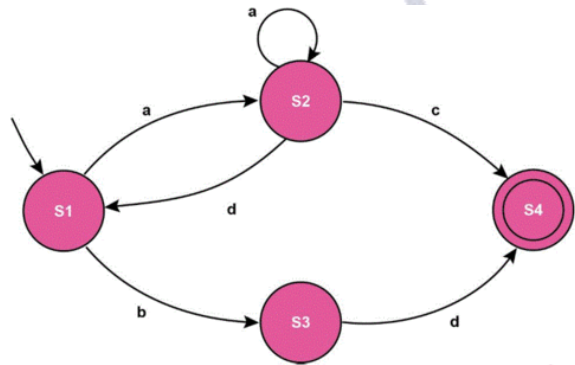
# Düzenli İfadeler

- ❑ Düzenli ifadeler aslında deterministik sonlu durum otomatlarıdır (makinelidir).
- ❑ Sonraki sunumda gösterilen ifadede sonlu durum otomatları ile bir veri yapısı gösterilmektedir.
- ❑ Bu veri yapısında kullanılan düğümlerden çift çizgili olanlar son durumu göstermektedir.



# Düzenli İfadeler

- ❑ Regex:  $(a^+)(d(a^+c|bd)^+)$
- ❑ Yukarıdaki düzenli ifade ile ifade edilen S1, S2, S3, ve S4 düğümleri ile ilgili olarak düğümler arasında geçiş için kullanılan a, b, c, ve d karakterlerini kullanarak tüm düzenli ifadenin tüketilmesi gerekir.
- ❑ Örneğin: aaaadaac, adac, adaabd, ve adbd ifadeleri yukarıdaki düzenli ifade tarafından üretilebilir yada tüketilebilir.





# Düzenli İfadeler

- ❑  $[A-Z]$  : A ve Z arasındaki bütün büyük harfleri yakalamak için kullanılır. Bu harfler sadece İngilizce alfabeden gelmektedir.
- ❑  $[A-ZÜŞİÇÖĞ]$  : A ve Z arasındaki İngilizce alfabeye ek olarak Türkçe karakterleri yakalamak için kullanılır.
- ❑  $[1-9]$  : 1 ve 9 dahil tüm sayıları temsil eder.
- ❑  $[1-9]^+$  : 1 ve 9 dahil tüm sayıların bir veya daha çok geçtiği durumlar örneğin: 2, 123, 111, 122339 gibi sayı dizilerinin yakalanması için kullanılır.
- ❑  $\backslash p\{Lu\}\{2, 5\}$  : 2 veya en fazla 5 karakterden oluşan büyük harflerin yakalanmasında kullanılır. Örneğin: ASV, AB, ABDFFR gibi harf dizilerini yakalar.
- ❑  $^{\wedge}[A-Z]$  : A ve Z karakterleri dışındaki harfler.
- ❑  $[.]$  : Herhangi bir sembol, karakter, yada sayı: &, A, 5, ...

# Düzenli İfade örnekleri

İfade	Örnek
woodchucks?	woodchuck
colou?r	color or colour
^	Satır başı
\\$	Satır sonu
\b	Kelime sınırı
\B	Kelime sınırı olmayan
\d	[0-9] : Rakam
\D	[^0-9] : Rakam olmayan
\w	[a-zA-Z0-9] : Harf yada rakam
\W	[^\w] : Harf yada rakam olmayan
\s	[\r\t\n\f] : Boşluk
\S	[^\s] : Boşluk olmayan



# Düzenli İfadeler Örnek

## Foundations of statistical natural language processing

Yazar	<a href="#">Manning, Christopher D.</a>
Diğer Yazarlar, Sorumlular vb...	<a href="#">Christopher D. Manning</a> , <a href="#">Hinrich Schütze</a> .
Basım Yılı:	1998
Yer Numarası	P98.5 M366 1999
LC Yer Numarası	P98.5
Dil Kodu	İngilizce
ISBN	9780262133609
Fiziksel Tanımlama	680 sayfa : çizim ; 24 cm.
Genel Not	Dizin: 657-680 sayfa.
Bibliyografi, vb. Notu	Kaynakça: 611-655 sayfa.
Konu - Konusal Terim	<a href="#">Bilgisimsel dilbilim</a> , <a href="#">Computational linguistics</a> , <a href="#">İstatistiksel yöntemler</a> , <a href="#">Statistical methods</a> .
Diğer Yazarlar	<a href="#">Schütze, Hinrich</a> , yazar.

[1-9]+

\d{4}

- Yandaki doküman örneğinde bir kütüphaneye ait kitap borkodu verilmektedir. Bu kitap barkodunun içerisinde ayrıca kitabın basım yılı yer almaktadır. Buna göre kitabın basım yılını bulan düzenli ifade nedir?

# Düzenli İfade - Java Örneği

```
package test;
```

```
public class RegexTestStrings {  
    public static final String EXAMPLE_TEST = "Bu kitabın basım yılı 1998 ve bu da barkod numarası 1999-  
20Z1.12 "  
    public static void main(String[] args) {  
        System.out.println(EXAMPLE_TEST.matches("\\d{4}"));  
        String[] splitString = (EXAMPLE_TEST.split("\\s+"));  
        System.out.println(splitString.length);  
        for (String string : splitString) {  
            System.out.println(string);  
        }  
    }  
}
```

2010

# Düzenli İfadeler - Problemleri

- Aşağıdaki metin örnekleri içinde «büyük heykel» geçen örnekleri çıkaran düzenli ifadeyi yazınız.
- Büyük yeşil boğa heykeli uzaktan çok minik göründü.
- büyük ve küçük heykeller sergilendi.
- Büyük güzel ağaç heykeli kapatıyor.

Büyük (.\* ) heykel

büyük (.\* ) heykel

???

2010

# Öbek karmaşıklıkları

- Bir dili başka bir dile çeviren diller bağımsız ya da yinelemeli sıralı (recursively enumerable) dillerdir.
- Karmaşıklık yukarıdaki örnek düşünüldüğünde anlaşılması zor bir hal almaktadır.
- Bir dilin parçalarından oluşturularak dilde geçen tüm ifadeleri üretebilen kuralların tümüne gramer denir.
- Düzenli ifadeler en basit ifade ile düzenli gramerler olarak adlandırılabilir.

2010

# Düzenli ifade çeşitleri

- İki sevyeli morfolojik analiz : Sonlu durum çeviricileri
- Düzenli ifadeler belirli bir öbeği yakalarken sonlu durum otomatlarını kullanırlar.
- Ancak dildeki tüm yapılar sonlu durum otomatları ile modellenemez. Örneğin
- Satılacaklar : kitap +laş +tır +ıl +an +lar
- Yukarıdaki kelimenin eklerinin bulunması için ekler arasındaki bağıntılar dikkate alınır. Örneğin +laş isimden fiil türeten ekten sonra +lar çoğul eki gelmez.
- Bazı durumlarda örneğin pçtk sessiz benzeşmesi (kitapım => kitabım), yada hece düşmesi (alın => alnımda) durumlarında eklerin ayrıştırılması için sonlu durum çeviriciler kullanılır.

2010

# Düzenli ifade çeşitleri

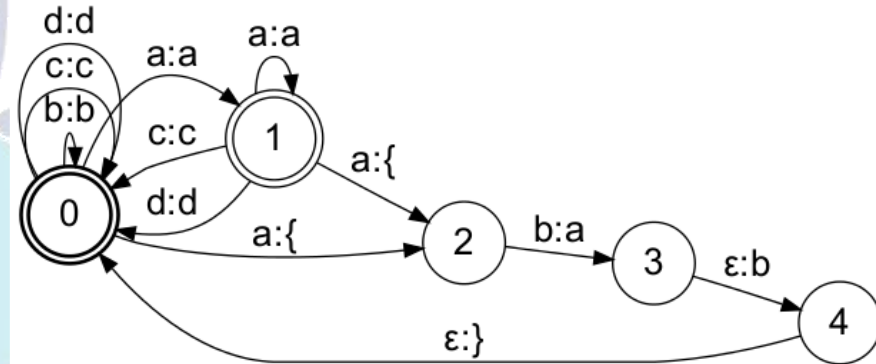
Yandaki düzenli ifade epsilon ( $\epsilon$ ) karakteri boş karakteri ifade eder.

A:  $\epsilon$  şeklinde yazıldığında tüketim için A karakteri ve üretim için boş karakter kullanılır.

Buna göre kitabın kelimesini kitap ve in şeklinde kitap+ın şeklinde yazmak için ne kullanılır.

Boş karakter tüketim için kullanıldığında sonlu durumlar deterministik olmayabilir.

Deterministik olma durumunda birden çok dallanma ortaya çıkacaktır.





# Düzenli çeviriciler

Sonlu durum otomatları bir metin içerisinde geçen düzenli öbekleri yakalamak için kullanılırlar.

Sonlu durum otomatları ile metin içerisinde geçen örneğin tarih, saat, isimler, ekler yakalanabilir.

Sonlu durum otomatları yakalanan ifade üzerinde bir değişiklik yapmazlar.

Türkçe gibi dillerde otomatlar eklerin yakalanmasında kullanılabilir. Ancak bir kelimenin kökünün sondan tüketim ile elde edilmesinde yeterli değildir.

Ekler:

kitaplar  $\Rightarrow$  kitap + lar

kitabı  $\Rightarrow$  kitap + ı

alnından  $\Rightarrow$  alın + ın + dan

Yukarıdaki ek çözümlemelerinde bir otomat için bitiş durumları ek sonlarını ifade etmektedir.

Yukarıdaki otomatın  $b \Rightarrow p$  ve  $\epsilon \Rightarrow ı$  dönüşümlerini yapabilmesi için sonlu durum otomatının, sonlu durum çeviricisi olması gerekmektedir.

Burada  $\epsilon$  epsilon yani boş karakteri temsil etmektedir.

Bu çevirici sadece yukarıdaki üç örnek için ne olmalıdır.



# Düzenli ifadeler

AAAAA BBBB – içerisinde 5 kez A ve sonrasında da 4 kez B geçsin.

Bu ifade de A sayısı  $n$ , B sayısı da  $m$  olsun.

$A\{n\}B\{m\}$  düzenli ifadesi

sadece  $n$  ve  $m$  sabit ise doğru çıkarım yapılabilir.

$N$  ve  $m$  sayısı önceden bilinmiyorsa örneğin  $n > m$  koşulu için çıkarım yapamaz.

Yandaki ifadeleri çıkarım yapabilecek bir düzenli ifade yok.

Yakalanacak öbekler:

AAAAA BBB

AAA B

AAB

Yakalanmayacak öbekler

ABB

AB

AABB

AAABB

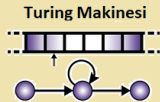

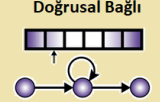

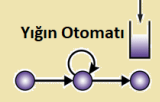
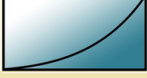
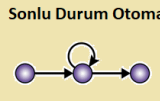

2010

# Öbek karmaşıklıkları – Chomsky Hiyerarşisi

Dilleri ve yakalamak istediğimiz metin karmaşıklıklarını ifade ederken daha önce tanımlanmış olan Chomsky hiyerarşisini kullanıyoruz. Bu hiyerarşide öbek yada dil karmaşıklığı açılım yöntemi ile gösteriliyor.

$A \rightarrow cA$   
 $S \rightarrow gSc$

Yukarıdaki örneklerde A yerine c A ve S yerine g S c yazılarak açılım daha da türetilebilir.

Language	Automaton	Grammar	Recognition
Recursively Enumerable Languages		Bağımsız $Baa \rightarrow A$	Undecidable 
Context-Sensitive Languages		Bağlam Duyarlı $A t \rightarrow aA$	Exponential? 
Context-Free Languages		Bağlamdan Bağımsız $S \rightarrow gSc$	Polynomial 
Regular Languages		Düzenli $A \rightarrow cA$	Linear 

# Referanslar

<https://bilgisayarkavramlari.com/2009/06/27/chomsky-hiyerarsisi-chomsky-hierarchy/>  
<https://bilgisayarkavramlari.com/2007/04/14/regular-expression-regex-duzenli-deyimler-ifadeler/>  
<https://web.cs.hacettepe.edu.tr/~ilyas/Courses/BBM401/lec03-RegularExpressionsRegularLanguages.pdf>  
[https://tr.wikipedia.org/wiki/D%C3%BCzenli\\_ifade](https://tr.wikipedia.org/wiki/D%C3%BCzenli_ifade)

2010