

Biçimsel Diller ve Otomata Teorisi

*Dr. Öğr. Üyesi Hayri Volkan Agun
Bilgisayar Mühendisliği Bölümü
Bursa Teknik Üniversitesi*

Kaynaklar

Ders Kitabı

- An Introduction to Formal Languages and Automata, Peter Linz, 6th Edition, 2017.
- An Introduction to Computer Theory, Daniel Isaac Aryeh Cohen, 2nd Edition, 1996.

İçerik

- %100 Teorik
- Klasik sınav
- Vize %40, Final %60

Giriş

- Biçimsel diller ve otomat teorisi Bilgisayar Bilimleri'nin temel konularından bir tanesidir.
- Programlama dilleri, algoritmalar ve bilgisayarın üretebileceği her türlü çıktı için bir tanımlaması yapılabilir.
- Hatta konuştuğumuz dil, matematik, algoritmalar bir bilgisayar programı ile üretilebilir.
- Otomatalar biçimsel dillerin uygulamasında kullanılan bir hesaplama yöntemidir.
- Yunanca'da kendi kendine karar veren makineler olarak atlandırılan Otomat kelimesinden türetilmiştir.

Giriş

- Biçimsel dillerin temeli Matematikçi ve Bilgisayar Bilimlerinin kurucularından olan **Alan Turing** tarafından atılmıştır.
- Alan Turing 2. Dünya savaşında kod kırma görevinde yer almış ve **Enigma** projesinde çalışan bir mühendis olarak da tanınmaktadır.
- Alan Turing aynı zamanda günümüzde en ünlü yapay zeka testi olan Turing Test'i geliştirmiştir.



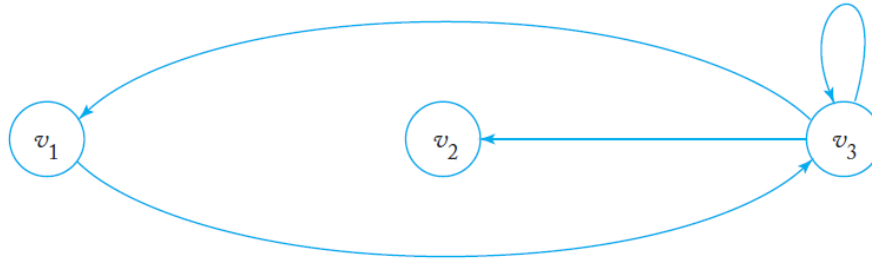
Ders İçeriği

- Ders içeriği 3 ana kısımda özetlenebilir:
 - Klasik otomat teorisi:
 - Chomsky hiyerarşisi, sonlu durum otomatları,
 - Push down otomatı, lineer olarak sonlu durumlar,
 - Sonlu durum otomatları
 - Kararlı (deterministic) ve kararsız durumlar.
 - Myhill-Nerode theorem, Kapalılık özellikleri,
 - Kleene Kuramı ve Minimum Otomatlar
 - Karar verme problemleri (Durma Problemi) ve ifade edilebilirlik

Temel yöntemler

- Küme teorisi
- Fonksiyonlar ve karmaşıklık
- Ağaç ve çizge yapıları
- İspat yöntemleri
 - Tüme varım
 - Çelişki ile kanıt

$$2^S = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}.$$



$$\begin{aligned} f(n) &= 2n^2 + 3n, \\ g(n) &= n^3, \\ h(n) &= 10n^2 + 100. \end{aligned}$$

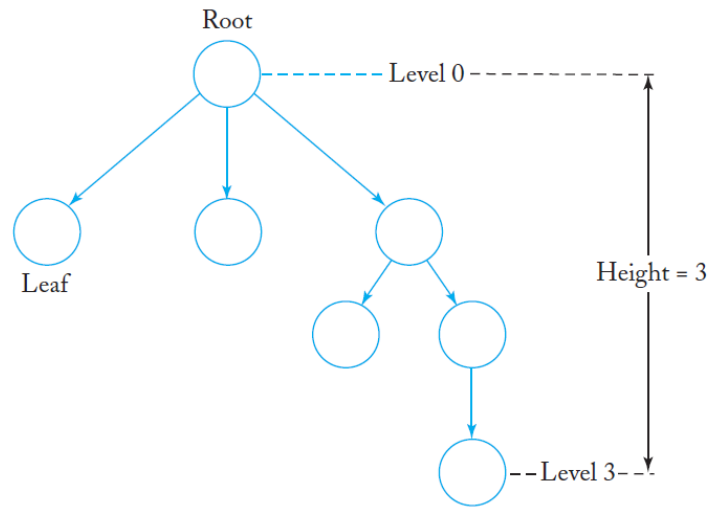
$$\begin{aligned} f(n) &= O(g(n)), \\ g(n) &= \Omega(h(n)), \\ f(n) &= \Theta(h(n)). \end{aligned}$$

$$O(n) + O(n) = 2O(n)$$

Tüme varım ile ispat

- Tüme varım ile ispat yönteminde ispatlamada kullanılanılacak 3 adım vardır. Bunlar;
 - Temel durum (Basis): her zaman doğru kabul edilir.
 - Ara adımlar (Inductive steps): Genele giden ara adımlar ve bu adımların birbirleri arasındaki bağıntılardan oluşur. Her bir ara adımın doğruluğu ispatlanır ve doğru kabul edilir olmalıdır.
 - Son adım (Proof): İspatı tamamlayan temel adım ve ara adımları kullanarak genelleme yapılan son adımdır.
- Örneğin bir ağaç veri yapısı n yükseklikte ise bu ağacın $I(n)$ adet son yaprağı bulunmaktadır. Bu durumda $I(n) < 2^n$ ifadesini ispatlayınız.

Tüme varım ile ispat



- Tüme varım ile ispat yönteminde ilk durum $n=0$ için $l(0) = 1$ dir. Kısaca ağacın yüksekliği 0 ise son yaprak adeti 1 olur. Bu bağıntı için $l(0) = 2^0$ kullanalım.
- Ara adımlar ise n değerinin 1'den büyük olduğu durumlar olsun. $l(i) \leq 2^i, \text{ for } i = 0, 1, \dots, n.$
- Ara adımlar arasındaki bağıntı ise $l(n+1) = 2l(n).$ Kısaca eğer ağaca yeni bir düğüm eklenirse ağaçtaki yaprak sayısı bir önceki yüksekliğin 2 katı olacaktır. Kısaca yeni bir alt düğüm eklenmesi ile bir önceki yükseklik arasında bir bağıntı kurulur.
- Son adımda önceki tüm adımlar doğru olarak ispatlandığına göre $l(n+1) \leq 2 \times 2^n = 2^{n+1}$ ifadesi doğrudur diyebiliriz.

Çelişki ile ispat

- Ortak böleni olmayan n ve m sayılarının oranı şeklinde ifade edilen sayılara rasyonel sayılar denir. Bir gerçek sayı rasyonel değil ise irasyoneldir. Örneğin $\sqrt{2}$ nin irasyonel olduğunu ispatlayınız.
- Bu tür bir ispat için tersinin doğru olmadığını ispatlamak kendisinin doğru olduğunu ispatlamak için yeterlidir. Çünkü rasyonel ve irrasyonel sayılar arasındaki bağıntı sadece bir birinin tersi olmasıdır.
- Çelişki ile ispat yönteminde temel durumu yani tersini doğru kabul ediyoruz. Ve genel durum için bir çelişki varsa o zaman bu çelişkiyi gösteriyoruz.

Çelişki ile ispat

- Farzedelim karekök değerler irrasyonel değil de rasyonel olsun. Bu durumda; $\sqrt{2} = \frac{n}{m}$,
- Her iki tarafın karesini alalım. $2m^2 = n^2$
- Bu durumda karesi çift sayı olan sayılar mutlaka çift sayıdır. O zaman n bir çift sayıdır; $n = 2k$
- Eğer n çift sayı ise yukarıdaki eşitlik yandaki gibi yazılır;
$$2m^2 = 4k^2,$$
$$m^2 = 2k^2.$$
- Bu durumda m'de çift sayıdır.
- Eğer her iki değerde çift sayı ise ortak bölenleri vardır ve 2 dir. O zaman bu sayı rasyonel olamaz çünkü ortak bölenleri olmama tanımına uymaz.

Dil

- Dil dediğimiz zaman aklımıza ilk gelen konuştuğumuz (doğal) diller; örneğin: Türkçe, Almanca, Arapça ve İngilizce gibi.
- Ancak dil çok geniş bir yelpazede incelendiğinde programlama dilleri, algoritmalar, matematik, ve semboller ile üretilebilen hemen hemen herşeye karşılık gelen bir kavramdır.
- Bilgisayar bilimlerinde karakter katarı (string) ifadesi olarak sembollerin belirli kurallar vasıtasıyla üretilmesi 'DİL' olarak tanımlanmaktadır.
- Biçisel olarak tanımlanabilen her dil Formal Dil yada Biçisel Dil olarak isimlendirilir.

Dil

- Biçimsel dilleri tanımlamak için semboller ve semboller arasındaki ilişkiler tanımlanır.
- Semboller aynı zamanda alfabeye olarak isimlendirilir. Aşağıdaki ifadeler dil tanımında kullanılır.
- $\Sigma = \{a, b\}$; a ve b sembolleri barındıran bir alfabe. Dil tanımlanırken alfabe boş olamaz ve en az bir boş olmayan sembol barındırmalıdır.
- Dil örneği: $w = abaaa$
- Boş sembol (empty symbol): λ ; boş karakter katarını temsil eder.
- Uzunluk operatörü $|w| = 5$; $|\lambda| = 0$,
 $\lambda w = w\lambda = w$

Dil

- Ön ek ve son ek; İngilizce prefix ve suffix terimleri ile adlandırılır. Dilde ön ve son ek için aşağıdaki örnek verilebilir.
- Bu dil için $w = abbab$, $u = ab$, $v = bab$ $w=uv$
- u bir ön ek, ve v bir son ektir. Tüm ön ekler $\{\lambda, a, ab, abb, abba, abbab\}$
- Tekrar eden karakter katarı için üst kullanılır. Örneğin üst sıfır ise uzunluk 0 olur bu boş sembole eşittir. $w^0 = \lambda$
- Verilen bir dil tanımına göre $L = \{a^n b^n : n \geq 0\}$; aab bu dile ait değildir. Bu dil ile üretilen örnek sayısı sonsuz olarak ifade edilir, çünkü n için bir üst değer tanımlı değildir.

Dil bilgisi (Grammar)

- Dili tanımlandığı alfabe ve geçişleri barındıran üretim kurallarıdır. Gramer kuralları açılım yada üretim şeklinde belirtmek için \rightarrow işaretini kullanır.
- Örneğin; cümle \rightarrow isim_öbeği fiil_öbeği. Burada cümle isim öbeğinin yanına fiil öbeği olarak oluşturulur.
- İngilizce için aşağıdaki tanımlama yapılabilir.

$$\langle sentence \rangle \rightarrow \langle noun_phrase \rangle \langle predicate \rangle$$
$$\langle noun_phrase \rangle \rightarrow \langle article \rangle \langle noun \rangle ,$$
$$\langle predicate \rangle \rightarrow \langle verb \rangle ,$$

Dil bilgisi (Grammar)

- Dilin biçimsel tanımlaması gramer ile yapılır. Aşağıda baş harf kısaltmaları kullanılarak gramer bileşenleri verilmiştir.

$$G = (V, T, S, P),$$

- V değişkenleri velirtir. Bu değişkenler sonlu sembol tanımlamalarını tutar. Önceki örnekte V <noun_phrase>, <predicate>, <sentence> değişkenlerdir.
- T sonlu sembol tanımlamalarıdır. Bunlar metin içinde gözlemlenen kelimeler, harfler yada noktalama işaretleri olabilir.
- S, başlangıç açılımını barındıran değişkendir. Her dilin bir veya birden fazla başlangıç açılımı vardır. Bu yukarıdaki örnek için <sentence> olarak belirtilmiştir.
- P ise \rightarrow sembolünü barındıran sonlu açılım kurallarıdır.

Dil bilgisi (Grammar)

- Aşağıda örnek bir dil bilgisi verilmiştir.

$$G = (\{S\}, \{a, b\}, S, P),$$

$$S \rightarrow aSb,$$

$$S \rightarrow \lambda.$$

- Bu kurallar ile üretilen bir örnek ise aşağıdaki şekilde türetilebilir.

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb,$$

- Bunu daha kısa yoldan göstermek için aslında aşağıdaki şekilde yazılabilir. Bu yazım şekli dilin karmaşıklığını göstermek için daha uygundur.

$$L(G) = \{a^n b^n : n \geq 0\},$$

Dil bilgisi (Grammar)

■ Örnek soru: $L = \{a^n b^{n+1} : n \geq 0\}$ ait olan dil bilgisi kurallarını yazınız.

■ Cevap: önce gramer tanımlanır sonra kurallar belirtilir.

■ 1. $G = (\{S, A\}, \{a, b\}, S, P)$

■ 2. $S \rightarrow Ab,$
 $A \rightarrow aAb,$
 $A \rightarrow \lambda.$

Dil bilgisi (Grammar)

- Örnek soru;

$$G_1 = (\{A, S\}, \{a, b\}, S, P_1)$$

$P_1 \Rightarrow$

$$\begin{aligned} S &\rightarrow aAb|\lambda, \\ A &\rightarrow aAb|\lambda. \end{aligned}$$

- Yukarıdaki gramerin G gramerine eşlenik olduğunu gösteriniz. Not: | sembolü alternatif açılım için kullanılır.

$$G = (\{S\}, \{a, b\}, S, P),$$

$P \Rightarrow$

$$\begin{aligned} S &\rightarrow aSb, \\ S &\rightarrow \lambda. \end{aligned}$$

$$L(G_1) = L(G) ?$$

Dil Bilgisi (Grammar)

- Herhangi bir karakter katarının (string) tersini bulmak için aşağıdaki kurallar kullanılabilir.

$$\begin{aligned}a^R &= a, \\ (wa)^R &= aw^R\end{aligned}$$

- Yukarıdaki kuralları doğruluğu tüme varım ile ispatlanabilir. Temel durumda bir prefix ve suffix için ters hesaplama yerleri değiştirilerek bulunur. Yani $(wa)^R \Rightarrow a^R w^R$ şeklindedir. Burada a ve w bir karakter katarına karşılık gelmektedir. Eğer a bir karakter ise bu durumda $a^R = a$ olurken w nin tersi karakter katarı olarak w^R olacağından ifade bulunmuş olur.
- Benzer durum iki dil ile üretilen tüm ifadeler içinde geçerlidir. $(L_1 L_2)^R = L_2^R L_1^R$

Dil Bilgisi (Grammar)

- Aşağıdaki her bir dil $\Sigma = \{a, b\}$ alfabetesine sahip olsun. Bu dilleri üretmekte kullanılan gramer kurallarını yazınız?

(a) $L_1 = \{a^n b^m : n \geq 0, m < n\}.$

(b) $L_2 = \{a^{3n} b^{2n} : n \geq 2\}.$

(c) $L_3 = \{a^{n+3} b^n : n \geq 2\}.$

(d) $L_4 = \{a^n b^{n-2} : n \geq 3\}.$

(e) $L_1 L_2.$

(f) $L_1 \cup L_2.$

(g) $L_1^3.$

(h) $L_1^*.$

(i) $L_1 - \overline{L_4}.$

Dil bilgisi (Grammar)

- L1 için dilde geçen a karakterinin sayısı b karakterinin sayısında her zaman fazla olmalıdır. Bu durumda;
 - Önce a'nın tekrarı sonra b'nin a ile eşit sayıda tekrarını içeren kuralı yazalım.
 - $B \rightarrow a B b$
 - Sonlu durumlar
 - $B \rightarrow \lambda$
 - a ve B kuralını birleştirip a'nın sayısı fazla olacak şekilde yazalım.
 - $S \rightarrow a B$
 - $S \rightarrow \lambda$

Örnek sorular

Sorular

Aşağıdaki dil tanımlamaları için gramerleri yazınız.

(a) $L = \{w : |w| \bmod 3 > 0\}.$

(b) $L = \{w : |w| \bmod 3 = 2\}.$

(c) $w = \{|w| \bmod 5 = 0\}.$

Aşağıdaki dil bilgisi kuralı ile belirtilen dile ait 3 örnek karakter katarı veriniz.

$$S \rightarrow aSb|bSa|a.$$

Aşağıda S tanımlamalarını barındıran iki farklı dilin aynı olmadığını gösteriniz.

Dil 1 $\Rightarrow S \rightarrow aSb|bSa|SS|a$

Dil 2 $\Rightarrow S \rightarrow aSb|bSa|a$