Biçimsel Diller ve Otomata Teorisi

Sunu I

İZZET FATİH ŞENTÜRK

Diller

• İngilizce: Harfler, kelimeler, cümleler • Harfler

grubu –> Sözcükler • Kelime grubu –> Cümleler •

Cümle grubu –> Paragraflar –> Hikayeler ..

• İnsanlar (çoğunlukla) hangi dizilerin geçerli olduğu konusunda hemfikirdir ve Bunlar değil. Nasıl?

Bilgisayar dili

- Belirli karakter dizileri tanınabilir sözcüklerdir (DO, IF, END, ..)

- Belirli dizeler -> komutlar
- Belirli komut setleri -> program (verilerle/verilerle ve derlenebilen)

Dil Yapısı

- Tüm bu örnekleri birleştiren genel bir teori oluşturmak
  - Bir dil yapısının tanımını benimseyin
  - Belirli bir birim dizisinin geçerli bir daha büyük sayı oluşturup oluşturmadığına karar birim
  - Tahmin meselesi değil
  - Açıkça belirtilen kurallara dayanarak

Amaç

- Bir girişin geçerli bir iletişim olup olmadığını tanımak için kurallar belirleyin
- İletişimin ne anlama geldiğiyle ilgilenmiyoruz

- Programın derlenmesi önemlidir.
- Programcının amaçladığı şeyi yapıp yapmadığıyla ilgilenmiyoruz
- Derlenirse, dilde geçerli bir örnektir.

Resmi Kurallar

• Konuşulan dilin tüm kurallarını belirtmek çok zor • Argo, deyim, lehçe, şiirsel metafor vb.

• Resmi dillerin genel bir teorisini tanımlamak • Kesin kurallar üzerinde ısrar etmek • Bilgisayarlar kusurlu komutları affetmezler

Resmi Kurallar

- Resmi: Dilin tüm kuralları açıkça belirtilmiştir (hangi sembol dizileri olabilir)
  - Hiçbir özgürlüğe müsamaha gösterilmez
  - Herhangi bir derin anlayışa atıfta bulunulmasına gerek yoktur
- Dil
  - Fikirlerin ifadesi değil kağıt üzerindeki semboller
  - Akıllar arasındaki iletişim değil, bir sembol oyunudur. resmi kurallar

Alfabe

- Oluşturmak için sınırlı bir temel birimler kümesiyle başlıyoruz
  yapılar
- Alfabeden -> dilden belirli bir dizi karakter dizisi
- Dilde izin verilen dizeler -> kelimeler
- Alfabedeki sembollerin Latin harfleri olması gerekmez
- Olası bir dize için yalnızca evrensel gereksinim: yalnızca sonlu sayıda simge içerir

Boş/Boş Dize

• Bir dizede harf olmamasına izin vermek istiyoruz: boş/boş dize

• Şununla ifade edin: Λ (Yunan başkenti lambda)

• Boş dize her zaman Λ'dir (hangi alfabe kullanılırsa kullanılsın) • Boş sözcük her zaman Λ'dir (eğer dilde bir kelimeyse)

Kelimeleri Karşılaştırma

- Aşağıdaki durumlarda iki kelime aynı kabul edilir.
  - tüm harfleri aynı • Tüm harfleri aynı

  sırada

- Harfsiz tek bir olası sözcük vardır: Λ • Açıklık açısından, genellikle

Λ sembolünün herhangi bir dil için alfabenin parçası olmasına izin vermeyiz.

Sözsüz Dil

- Arasındaki önemli fark.. • Harfi olmayan kelime • Λ

  - Kelime içermeyen dil (φ – (küçük Yunan harfi phi))

- Λ'nin φ'nin bir parçası olduğu doğru değil, φ'de

kelime yok • L bir dil Λ içermiyorsa, onu ekleyebiliriz: L+ {Λ}

  - L+ {Λ}

  L • L + φ = L

Sözsüz Dil

- φ'nin kelime içermeyen bir dil olması önemli bir ayrımdır.

- Bir dil üretmek için bir “yöntemimiz” olduğunda • Yöntem başarısız olabilir ve hiçbir şey üretemez veya.. • Yöntem dili başarıyla üretir φ

Alfabeyi Tanımlamak

- $\Sigma = \{abcde \ldots z\}$

- Öğeleri ayırmak için boşluk veya virgül kullanabiliriz

Geçerli Kelimeleri Belirtme

- Tüm geçerli kelimeleri bir sözlükte olduğu gibi listeleyebiliriz
  - Uzun liste ama sınırlı!

- WORDS = {standart bir sözlükteki tüm kelimeler}

- Bir dili sonsuz bir sözlükle tanımlama olanağına izin vermiyoruz

Geçerli Bir Cümle Oluşturun

- Sonlu bir dildeki (İngilizce vb.) tüm kelimeleri bilmek, geçerli bir cümle oluşturma yeteneği anlamına gelmez.
- Yeni bir alfabe tanımlayın Γ (büyük gama)
- Γ = {standart bir sözlükteki girişler, artı bir boşluk artı noktalama işaretleri}
- Tüm geçerli İngilizcelerin tam bir listesini asla üretemeyiz cümleler
    - Γ'de sonsuz sayıda kelime (Bir elma, iki elma, üç... yedim)
    - Sonsuz bir dilin sonlu tanımı!

Dilbilgisi Kuralları ve Anlamı

• Sadece Γ gramer kurallarına uyarak. • Üç

Salı yedim • Γ'de geçerli bir kelime • Bu
dizgeye izin vermeliyiz • Dilbilgisi
açısından doğru • Anlam saçma

• Yalnızca sözdizimiyle ilgileniyoruz, anlambilim yok!

Geçerli Kelimeleri Belirtmek Zor Olabilir

- MY-PET dili
- Alfabe {acdgot}

- Bu dilde sadece bir kelime var
  - Dünya ve Ay çarpışırsa, MY-PET = {cat}
  - Dünya ve Ay asla çarpışmazsa, MY-PET = {dog}
- Kesin değil. Dilin yeterli bir özelliği değil. Kurallar, bir kelimenin dilin bir parçası olup olmadığına sonlu bir süre içinde karar vermemizi sağlamalıdır.

Dilleri Tanımlama

- Dil tanımlayan kurallar seti iki tür olabilir
  - Bir dizgenin geçerli bir sözcük olup olmadığını nasıl test edeceğimizi bize söyleyebilirler.
  - Dildeki tüm kelimeleri nasıl oluşturacağımızı bize söyleyebilirler.

- Σ = {x} Tek harfli bir alfabe: x
- L1'i Tanımla: Boş olmayan herhangi bir alfabe karakter dizisi, bir kelime
  - L1 = {x xx xxx xxxx ...} alternatif olarak L1 = {xn for n = 1 2 3 ...}

Dilleri Tanımlama - Birleştirme

- Birleştirme işlemini tanımlıyoruz
  - Yeni bir dize oluşturmak için yan yana yazılan iki dize •

xxx ve xx'i birleştirdiğimizde kelimeyi elde ederiz
 xxxxx

- Eklemeye benzer • xm
  ile birleştirilmiş xn : xn+m

- Alfabe dışındaki yeni sembolleri kullanmak daha uygun • xxx a, xx
  b ve xxxxx ab'dir

tanımlama Diller - birleştirme

- İki kelime birleştirildiğinde, dilde başka bir kelime ürettikleri her zaman doğru değildir.
- a = xxx ve b = xxxxx L2'deki kelimelerdir ancak ab değil
- Örneklerde ab=ba ama bu her zaman böyle değildir!

$$
\begin{aligned}
L_2 &= \{x \quad xxx \quad xxxxx \quad xxxxxxx \ldots\} \\
&= \{x^{\text{odd}}\} \\
&= \{x^{2n+1} \quad \text{for} \quad n = 0 \quad 1 \quad 2 \quad 3 \ldots\}
\end{aligned}
$$

Dilleri Tanımlama

- $\Sigma = \{0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\}$
- L3 = {sıfır harfiyle başlamayan herhangi bir sonlu alfabe harfi dizisi}

- L3 , 10 tabanındaki tüm pozitif tam sayıların kümesine benziyor.
- L3 = {1 2 3 4 5 6 7 8 9 10 11 12 ...}
- 0 kelimesini içeren L3'ü tanımlamak istersek
- L3 = {0 ile başlıyorsa, ilk harften sonra başka harf içermeyen herhangi bir sonlu alfabe harfi dizisi}

Uzunluk Fonksiyonu

• Bir dizgenin fonksiyon uzunluğunu, dizgideki harf sayısı olarak tanımlarız • Eğer a = xxxx ise, uzunluk(a) = 4 • Eğer c = 428 ise,

uzunluk(c) = 3 • uzunluk(xxxxx) = 5 • uzunluk (Λ) = 0 • Herhangi bir

dilde herhangi bir w kelimesi için, uzunluk(w) = 0 ise, w = Λ

Aynı Dil için Çoklu Tanımlar

- L3 = {0 ile başlıyorsa, ilk harften sonra başka harf içermeyen herhangi bir sonlu alfabe harfi dizisi}
- L3'ün bir tanımı daha
- L3 = {uzunluğu 1'den büyükse 0 ile başlamayan herhangi bir sonlu alfabe harfi dizisi}
- L3'ün daha iyi bir tanımı olması gerekmez, ancak aynı dili belirtmenin genellikle farklı yolları olduğunu gösterir.

Λ hakkında daha fazlası

- “Herhangi bir sonlu dizide” belirsizlik •
  Λ'nin L3'ün parçası olup olmadığı net değil

- L3 Λ içermez • L3'ün tamsayılar
  gibi görünmesini istiyoruz • Rakamsız
  tamsayı yoktur

- L4 = {Λ x xx xxx xxxx ...}, L4 = {xn for n = 0 1 2 3 ...} tanımlarız • X0 = Λ, albegra'daki

gibi X0 = 1 değil (xn , n x'lerdir)

Ters Fonksiyon

- Ters fonksiyonu tanımlıyoruz. a, L dilinde bir kelimeyse, ters(a) geriye doğru yazılan aynı harf dizisidir.
    - Geriye doğru dize L'de bir kelime olmayabilir
- ters(xxx) = xxx
- ters(145) = 541
- ters(140) = 041 -> 140, L3'te bir kelimedir, ancak 041 değildir!

palindrom

- Alfabesi üzerinde PALINDROME adında yeni bir dil tanımlıyoruz $\Sigma = \{ab\}$
- PALINDROME = {Λ ve tüm x dizileri ters(x) = x} olacak şekilde
- PALINDROME = {Λ ab aa bb aaa baba bbb aaaa baba ...}

Kleene Kapatma

- Σ alfabesi verildiğinde, Σ'den gelen herhangi bir harf dizisinin bir kelime olduğu, hatta boş karakter dizisi bile olduğu bir dil tanımlamak istiyoruz. Bu dile alfabenin kapanışı denir.

S*

- Σ = {x} ise Σ* = L4 = {Σ x xx xxx ...}
- Σ = {0 1} ise Σ* = {Λ 0 1 00 01 10 11 000 001...}
- Σ = {abc} ise Σ* = {Λ abc aa ab ac ba bb bc ca cb cc aaa ...}

Kleene Kapatma

- Kleene yıldızı, bir alfabeden sonsuz bir harf dili oluşturan bir işlemdir.
- Sonsuz dil -> her biri sonlu uzunlukta sonsuz sayıda kelime

sözlük düzeni

- Σ* = {Λ abc aa ab ac ba bb bc ca cb cc aaa ...}
- Dilde ilk birkaç kelimeyi yazdığımızda, onları büyüklük sırasına (uzunluk) koyarız ve sonra aynı uzunluktaki tüm kelimeleri alfabetik olarak sıralarız.
- Bu sıralamaya sözlük sıralaması denir
- Sözlükte aardvark kelimesi kediden önce gelir. Sözlükbilimsel sıralamada ise durum tam tersidir.
- Alfabetik olarak sıralansaydı, liste {Λ a aa aaa aaaa ...} ile başlardı ve dilin gerçek doğası hakkında bilgi vermezdi.

Kelimelerde Yıldız Operasyonu

- Yıldız operatörünün kullanımını sadece alfabedeki harf kümelerine değil, kelime kümelerine de genelleyebiliriz.
- S bir sözcük kümesiyse, o zaman S*, S'den sözcüklerin birleştirilmesiyle oluşturulan tüm sonlu dizelerin kümesidir, burada herhangi bir sözcük istediğimiz sıklıkta oluşturulabilir, burada boş dize de dahildir

Kelimelerde Yıldız Operasyonu

- S = {aa b} ise
- S* = {Λ artı aa ve b faktörlerinden oluşan herhangi bir kelime}
- S* = {Λ artı a'ların çift kümeler halinde gerçekleştiği tüm a ve b dizileri}
- S* = {Λ b aa bb aab bbb aaaa aab bbbb bbbbb ...}

- a'nın uzunluğu 3 olan bir kümeye sahip olduğu için aabaaab S* değildir

Kelimelerde Yıldız Operasyonu

- O zaman S = {a ab} olsun
- S* = {Λ artı a ve ab faktörlerinden oluşan herhangi bir kelime}
- S* = {Λ artı b ile başlayanlar ve çift b içerenler hariç tüm a ve b dizileri}
- S* = {Λ a aa ab aaa aab aba aaaa aaab aaba abaa abab aaaaa aaaab aaaba aabaa aabab abaaaa abaab ababa...}
- Çift b, bb anlamına gelir. S*'deki her kelime için her b'nin hemen solunda bir a olmalıdır. bb ab ile başladığı için imkansız

Kelimelerde Yıldız Operasyonu

- Belirli bir kelimenin S*'de olduğunu kanıtlamak için, temel kümeden kelimelerin bir dizilimi olarak nasıl yazılabileceğini göstermeliyiz.

  S

- Son örnekte, abaab S*'dedir, şu şekilde çarpanlarına ayırabiliriz: (ab)

  (a)(ab)

- Bu üç faktörün tümü S'dedir, bu nedenle sıralanmaları

  S'dedir*

- Bu örnek için faktoring benzersizdir. bazen öyle

  olumsuzluk

Benzersiz olmayan Faktoring

• S = {xx xxx} •

S* = {Λ ve birden fazla x'in tüm dizeleri} • S* = {xn for

n = 0 2 3 4 5 ...} • S* = {Λ xx xxx xxxx xxxxx xxxxxx ... }

• x'in S*'de olmadığına dikkat edin

• xxxxxx, (xx)(xx)(xxx) veya (xx)(xxx)(xx) veya (xxx)

(xx)(xx)'den herhangi biri nedeniyle S*'dedir

Ayrıca x6 , x2x2x2 veya x3x3'tür

Son Açıklamalar

- İki kümenin Kleene kapanışı, başladığımız iki küme birbirinden farklı olsa bile aynı dil olabilir.

- S = {ab ab} ve T = {ab bb}
- Hem S* hem de T*, a'nın ve b'nin tüm dizelerinin dilleridir ve herhangi bir a ve b dizesi (a) veya (b) hecelerine ayrılabilir, her ikisi de S ve T'dedir

Son Açıklamalar

- Kapatma kavramını sadece bir S kümesindeki sıfır olmayan dizelerin birleştirilmesine atıfta bulunacak şekilde değiştirmek istersek, * yerine + notasyonunu kullanırız.
- Σ = {x} ise, o zaman Σ+ = {x xx xxx ...}
- S, Λ içermeyen bir dizi diziyse, o zaman S+, Λ sözcüğü olmayan S* dilidir.
- S, Λ içermeyen bir dilse, S+ = S*
- Artı işlemine bazen pozitif kapatma denir