

Doğal Dil İşleme

Temel Problemler ve Yaklaşımlar

Kemal Oflazer
Carnegie Mellon Üniversitesi – Katar
`ko@cs.cmu.edu`

`http://www.andrew.cmu.edu/user/ko/
@oflazer`

- Doğal Dil İşleme
- Sözcük sınıfı işaretleme
- Biçimbirimsel Çözümleme
- Bağlılık Çözümlemesi
- Kaynaklar

Doğal Dil İşleme

Doğal Dil İşleme

- İnsan – Bilgisayar etkileşimini doğal dil ile yapabilen
- İnsan – İnsan etkileşimini zenginleştiren / kolaylaştıran

bilgisayar sistemleri geliştiren bilgisayar mühendisliği alt-alanıdır.



Doğal Dil İşleme ile Bağlantılı Alanlar

- Yapay zeka
- Makina öğrenmesi
- Biçimsel diller ve otomatlar kuramı
- İstatistiksel modelleme
- Bilgi Kuramı (Information Theory)
- Bilişsel Bilimler (Cognitive Science)
- Mantık
- İnsan – Bilgisayar etkileşimi
- Dilbilim
- Ahlakbilim (Ethics)
- Sosyal bilimler, siyasal bilimler, psikoloji, ekonomi, eğitim, ...
- Yazılım mühendisliği

2020'de Bazı Doğal Dil İşleme Uygulamaları

- Soru cevaplama (Siri, Google Assistant, Alexa, vb.)
- Yazı ve konuşma çevirme (Google Translate)
- Bir dizi yazıyı özetleme
- Bir toplantıda konuşulanları takip edip özetleme
- E-postalara otomatik yanıtlar önerme/yazma (Gmail)
- Yanlış / palavra haberleri farketme
- Sosyal medya duygu çözümlemesi

- Görüleni anlatma:
 - konuşma/duyma/görme kısıtlı kişilere yardımcı olma,
 - maç anlatma (:-)
- Yazım veya sözdizim hatalarını düzeltme
- Kompozisyon notlama (SAT)
- Okuduğunu anlama sınavları için soru üretme
- Etkileşimli olarak dil öğretme

tisimptant too spll chck ths dcment.



It's important to spell check this document.

¹<http://www.realworldnlpbook.com/blog/unreasonable-effectiveness-of-transformer-spell-checker.html>

The book Tom and Jerry put on the yellow desk yesterday war about NLP.



The **book** Tom and Jerry put on the yellow desk yesterday **was** about NLP.

The books Tom and Jerry put on the yellow desk yesterday war about NLP.



The **books** Tom and Jerry put on the yellow desk yesterday **were** about NLP.

Örnek Uygulama – Adlandırılmış Varlık Tanıma

The decision by the independent MP Andrew Wilkie to withdraw his support for the minority Labor government sounded dramatic but it should not further threaten its stability. When, after the 2010 election, Wilkie, Rob Oakeshott, Tony Windsor and the Greens agreed to support Labor, they gave just two guarantees: confidence and supply.



The decision by the independent MP Andrew Wilkie to withdraw his support for the minority Labor government sounded dramatic but it should not further threaten its stability. When, after the 2010 election, Wilkie, Rob Oakeshott, Tony Windsor and the Greens agreed to support Labor, they gave just two guarantees: confidence and supply.

Kişi
Tarih
Yer
Organizasyon

yazı veya konuşma



Gerekli işlemeyi destekleyen bir “gösterim” $\mathcal{R}$

“Gösterim” $\mathcal{R}$



yazı veya konuşma

Büyük boyutta yazı veya konuşma örnekleri + bilgi



“Gösterim” $\mathcal{R}$ için çözümleme ve/ya üretme modeli/algoritması

- Gösterim $\mathcal{R}$?
- Nerede, ne amaçla kullanılacağına bağlı!
- $\mathcal{R}$ temelde bilgisayarın kullanacağı bir gösterim.
 - İnsanlar tarafından anlaşılabilir olabilir/olmayabilir!

- Makina öğrenmesi, örnek veriden model/program öğrenmeyi amaçlar.
- Makina öğrenmesi doğal dil işlemede her yerde kullanılan çok önemli bir yöntemdir.
- Doğal dil işleme sembolik, istatistiksel veya yapay sinir ağı temelli makina öğrenmesi için her zaman bir uygulama alanı olmuştur.
- İnsanlar çocukluk dönemlerinde öğreticisiz öğrenme ile dillerini öğreniyorlar.
 - Tam da öğreticisiz değil: görme ve konuşma ilişkilendirme
- Doğal dil işlemede öğreticisiz öğrenme halen yeterli derecede başarılı değil.

- Dilbilim dillerin nasıl çalıştığını, birbirleri ile nasıl ilişkili olduğunu, nasıl değiştiklerini inceler.
- Dilbilim insanların ilk (ve ikinci) dillerini nasıl öğrendiğini anlamaya çalışır.
- Dilbilim (ideal) tümcelerın sözdizim yapılarını modelleyen genel prensipler bulmaya çalışır.
- Doğal dil işleme “ideal” tümceler ile değil de gerçek dünyada görülen dille ilgilenir.

Doğal Dil İşleme [?] = Yapay Zeka

- Yapay zeka insanların akli melekelerini bilgisayarla taklit etmeyi amaçlar.
- Dil ve iletişim insan melekelerin en önemlilerinden birisidir.
- Dil insanlardaki görme, sağduyu, deneyim belleği, vb. gibi melekeler ile bağlantılıdır.
 - Paket **kutu**ya **küçük** olduğu için sığmadı.
 - **Paket** kutuya **büyük** olduğu için sığmadı.

Doğal Dil İşleme = Gösterim Dönüştürümü

Doğal dil işleme temelde bir dizi **gösterim dönüştürümü** problemiyle uğraşır.

- **Konuşma tanıma**: Sayısallaştırılmış ses dalgası dizisi → sözcük dizisi
- **Çeviri**: İngilizce tümce → Türkçe tümce
- **Düzeltilme**: Hatalı tümce → Düzeltilmiş tümce
- **Tümce Çözümlemesi**: Tümce → Sözdizim ağacı
- **Anlamsal Çözümleme**: Sözdizim ağacı → “kim, kime, ne zaman, nerede, nasıl, ne yaptı?”

Doğal Dil İşleme = Gösterim Dönüştürümü

Doğal dil işleme temelde bir dizi gösterim dönüştürümü problemiyle uğraşır.

- **Özetleme**: Uzun yazı(lar) $\rightarrow$ kısa yazı
- **Adlandırılmış varlık işaretlenmesi**: Yazı $\rightarrow$ Kişi, organizasyon, coğrafi öge, ürün, ... adları
- **Soru yanıtlama**: (Yazı, soru) $\rightarrow$ yanıt.
- **Duygu çözümlemesi**: Kısa yazı $\rightarrow$ {iyi, kötü, nötr}
- ...

Doğal Dil İşleme = Gösterim Dönüştürümü

- Genelde gösterim dönüştürümü tek adımda yapılamayacak kadar karmaşıktır.
- Bu nedenle bir dizi birbirine bağlı daha basit ara dönüştürüm adımlarına bölünür.
- Bu adımların hepsi farklı (ara) gösterimler kullanabilirler
 - Bunlar bazen elle tasarlanabilir, veya
 - Makina öğrenmesi sistemleri kendi (ara) gösterimlerini bulabilirler.
- Doğal dil işleme üzerinde çalışanlar bu ara gösterimleri anlamak/yorumlamak durumundadırlar.

Örnek: İngilizce Nasıl Türkçeye Dönüşür?

if we will be able to make ... become strong

if we will be able to make ... become strong

... strong become to make be able will if we

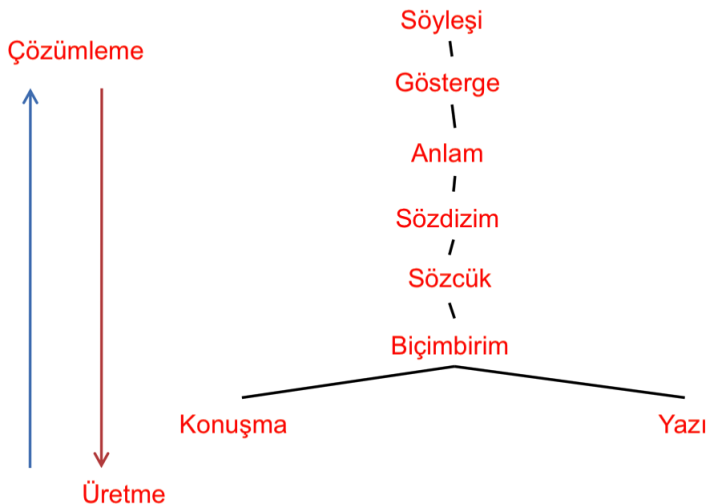
... sağlam +laş +tır +abil +ecek +se +k



... **sağlamlaştırabileceksek**

Gösterim Düzeyleri

Klasik doğal dil işleme şu gösterim düzeylerini kullanır:



Doğal Dil İşleme Neden Zor?

Çünkü (-)



Doğal Dil İşleme Neden Zor?

Çünkü (-)



Biz Türküz. Fransızcadan “Au Gratin” tanımlamasını alır, önce “Qgraten”, sonra “Öğreten” yapar, bununla yetinmez “Teaches” diye İngilizceye çevirir, elin gavurunu dumur ederiz. Yaşasın Chicken Translate!

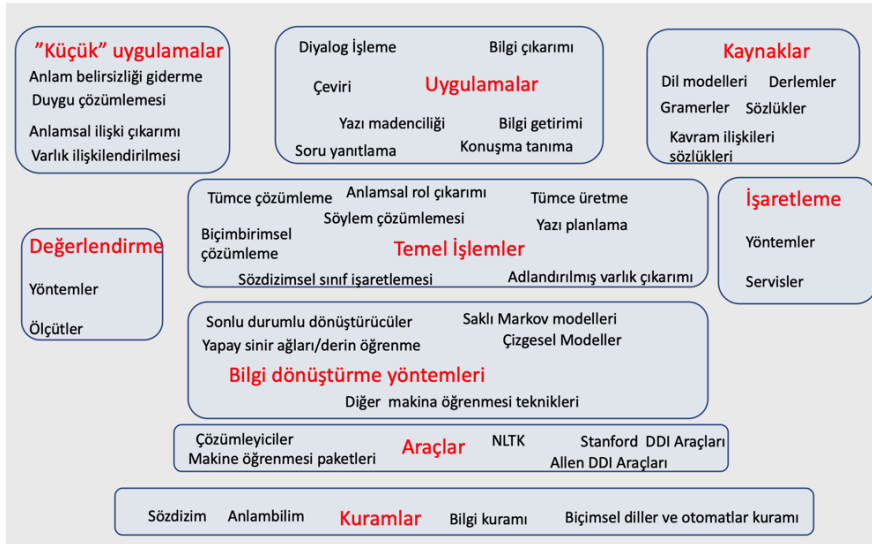
Doğal Dil İşleme Neden Zor?

- Düzeyler arasındaki ilişki ve aktarım çok karmaşık.
- Gösterimlerin detayları ve yeterliliği uygulamanın ne olduğuna bağlı.
- Gösterimler ideal ve kuramsal
 - İnsan beyinde **hiç bir zaman gösterimleri doğrudan gözleyemiyoruz!**
- Girdi genelde “gürültülü”.
 - Yazım hataları, fazla ve eksik sözcükler (“hmm”, “yani”), vb

Çok Yapılilik/Çok Anlamlılık

- Bir konuşma ses dalgası çok sayıda olası tümceye dönüştürülebilir.
 - It is hard to recognize speech / It is hard to wreck a nice beach.
 - Evimi neyle kaplattın. / Evi mineyle kaplat tın.
- Bir tümcedeki her sözcüğün çok sayıda biçimbirimsel yorumu olabilir.
 - elma+sı/elmas+ı
- Her kök sözcüğün birden fazla anlamı olabilir.
 - Hal+den elma aldım / Hal+den anlamıyorsun.
- Bir tümcenin çok sayıda yapısal çözümlemesi olabilir.
 - Time flies like an arrow!
 - Ali Ayşe'yi eve gelirken görmüş.
- Bir yapısal çözümlemenin birden fazla mantıksal anlamı olabilir.
 - Everybody on this island speaks two languages!
- **Esas zorluk olası yorumların hangisinin doğru olduğunu çevrimden kestirmek.**
 - İnsanlar bunu hiç gayret harcamadan bir şekilde yapıyorlar.
 - Bilgisayarlar henüz insanlar kadar iyi yapamıyorlar.

Doğal Dil İşleme – Büyük Resim



(Eduard Hovy'nin "The Past and $3\frac{1}{2}$ Futures of NLP" adlı konuşmasından adapte edildi.)

Doğal dil işlemede kullanılan dönüştürme yöntemleri genelde birkaç sınıfa ayrılır:

- Dizin $\rightarrow$ Dizin dönüştürümü
- Dizin $\leftrightarrow$ Çok boyutlu yapı dönüştürümü
- Dizin $\rightarrow$ Sınıf

Dizin → Dizin Dönüştürümü: (İngilizce) Sözcük Sınıfı İşaretlemesi

- “My cat, which **lives** dangerously, no longer has nine **lives**.”
 - İlk **lives** şimdiki zamanda bir eylem (VBP).
 - ikinci **lives** ise çoğul ad (NNS).
 - Bu sözcükler farklı söylenir.
 - Bazen sınıfa bağlı olarak sözcük söyleme vurguları farklı olabilir **SUSpect**, ad (NN) veya **susPECT**, eylem (VBP)
- “He **can can** the **can**.”
 - İlk **can** yardımcı eylemdir (MD).
 - İkinci **can** ise (zamansız) eylemdir (VB).
 - Üçüncü **can** tekil addır. (NN)
 - **can** için dördüncü bir olasılık şimdiki zamanda eylemdir – “We can tomatotes every summer.”
- He can can the can .→ He/**PRP** can/**MD** can/**VB** the/**DT** can/**NN** ./.

Dizin ← Dizin Dönüştürümü: Biçimbirimsel Çözümleme

- *okuma* →
 - *ok+um+a* ok+Noun+A3sg+P1sg+Dat “to my arrow”
 - *oku+ma* oku+Verb+Neg+Imp+A2sg “do not read!”
 - *oku+ma* oku+Verb+Pos^DB+Noun+Inf2+A3sg+Pnon+Nom “reading”
- *elmasında* →
 - *elma+sı+nda* elma+Noun+A3sg+P3sg+Loc “on his apple”
 - *elmas+ı+nda* elmas+Noun+A3sg+P3sg+Loc “on his diamond”
 - *elmas+ın+da* elmas+Noun+A3sg+P2sg+Loc “on your diamond”
- *koyunu* →
 - *koy+u+nu* koy+Noun+A3sg+P3sg+Acc “his bay (Accusative)”
 - *koy+un+u* koy+Noun+A3sg+P2sg+Acc “your bay (Accusative)”
 - *koyu+n+u* koyu+Adj^DB+Noun+Zero+A3sg+P2sg+Acc “your dark (thing) (Accusative)”
 - *koyun+u* koyun+Noun+A3sg+P3sg+Nom “his sheep”
 - *koyun+u* koyun+Noun+A3sg+Pnon+Acc “sheep (Accusative)”

- *ruhsatlandırılmamasındaki* →

- *ruhsat+lan+dır+ıl+ama+ma+sı+nda+ki*

ruhsat+Noun+A3sg+Pnon+Nom

^DB+Verb+Acquire^DB+Verb+Caus

^DB+Verb+Pass^DB+Verb+Able+Neg

^DB+Noun+Inf2+A3sg+P3sg+Loc

^DB+Adj+Rel

“related to (something) not being able to acquire certification”

- Ön bilgi
- Problem Modellemesi
- (Saklı) Markov Modelleri
- Viterbi Algoritması
- Diğer Yaklaşımlar

İngilizce Sözcük Sınıfı Kestirimi

Rolls-Royce Motor Cars Inc. said it expects its U.S. sales to remain steady at about 1,200 cars in 1990 . The luxury auto maker last year sold 1,214 cars in the U.S.



Rolls-Royce/NNP Motor/NNP Cars/NNPS Inc./NNP said/VBD it/PRP expects/VBZ its/PRP\$ U.S./NNP sales/NNS to/TO remain/VB steady/JJ at/IN about/IN 1,200/CD cars/NNS in/IN 1990/CD ./.
The/DT luxury/NN auto/NN maker/NN last/JJ year/NN sold/VBD 1,214/CD cars/NNS in/IN the/DT U.S./NNP

İngilizce İçin Penn Treebank İşaret Kümesi

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	<i>and, but, or</i>	SYM	symbol	<i>+, %, &</i>
CD	cardinal number	<i>one, two</i>	TO	“to”	<i>to</i>
DT	determiner	<i>a, the</i>	UH	interjection	<i>ah, oops</i>
EX	existential ‘there’	<i>there</i>	VB	verb base form	<i>eat</i>
FW	foreign word	<i>mea culpa</i>	VBD	verb past tense	<i>ate</i>
IN	preposition/sub-conj	<i>of, in, by</i>	VBG	verb gerund	<i>eating</i>
JJ	adjective	<i>yellow</i>	VCN	verb past participle	<i>eaten</i>
JJR	adj., comparative	<i>bigger</i>	VBP	verb non-3sg pres	<i>eat</i>
JJS	adj., superlative	<i>wildest</i>	VBZ	verb 3sg pres	<i>eats</i>
LS	list item marker	<i>1, 2, One</i>	WDT	wh-determiner	<i>which, that</i>
MD	modal	<i>can, should</i>	WP	wh-pronoun	<i>what, who</i>
NN	noun, sing. or mass	<i>llama</i>	WP\$	possessive wh-	<i>whose</i>
NNS	noun, plural	<i>llamas</i>	WRB	wh-adverb	<i>how, where</i>
NNP	proper noun, sing.	<i>IBM</i>	\$	dollar sign	<i>\$</i>
NNPS	proper noun, plural	<i>Carolinas</i>	#	pound sign	<i>#</i>
PDT	predeterminer	<i>all, both</i>	“	left quote	<i>‘ or “</i>
POS	possessive ending	<i>’s</i>	”	right quote	<i>’ or ”</i>
PRP	personal pronoun	<i>I, you, he</i>	(	left parenthesis	<i>[, (, {, <</i>
PRP\$	possessive pronoun	<i>your, one’s</i>	)	right parenthesis	<i>],), }, ></i>
RB	adverb	<i>quickly, never</i>	,	comma	<i>,</i>
RBR	adverb, comparative	<i>faster</i>	.	sentence-final punc	<i>. ! ?</i>
RBS	adverb, superlative	<i>fastest</i>	:	mid-sentence punc	<i>: ; ... --</i>
RP	particle	<i>up, off</i>			

Çok Sınıflılık Ne Kadar Kötü?

Sözcük	Sınıf Sayısı	Sözcük	Sınıf Sayısı	Sözcük/Sınıf	Örnek Sayısı
down	7	run	5	down/RB	317
that	6	repurchase	5	down/RP	200
set	6	read	5	down/IN	138
put	6	present	5	down/JJ	10
open	6	out	5	down/VBP	1
hurt	6	many	5	down/RBR	1
cut	6	less	5	down/NN	1
bet	6	left	5		
back	6				
the	5				
spread	5				
split	5				
say	5				
's	5				

“down” İçin Bazı Örnekler

One/CD hundred/CD and/CC ninety/CD two/CD former/JJ greats/NNS ,/, near/JJ greats/NNS ,/, hardly/RB knowns/NNS and/CC unknowns/NNS begin/VBP a/DT 72-game/JJ ,/, three-month/JJ season/NN in/IN spring-training/NN stadiums/NNS up/RB and/CC down/RB Florida/NNP ...

He/PRP will/MD keep/VB the/DT ball/NN down/RP ,/, move/VB it/PRP around/RB ... As/IN the/DT judge/NN marched/VBD down/IN the/DT center/JJ aisle/NN in/IN his/PRP\$ flowing/VBG black/JJ robe/NN ,/, he/PRP was/VBD heralded/VBN by/IN a/DT trumpet/NN fanfare/NN ...

Other/JJ Senators/NNP want/VBP to/TO lower/VB the/DT down/JJ payments/NNS required/VBN on/IN FHA-insured/JJ loans/NNS ...

Texas/NNP Instruments/NNP ,/, which/WDT had/VBD reported/VBN Friday/NNP that/IN third-quarter/JJ earnings/NNS fell/VBD more/RBR than/IN 30/CD %/NN from/IN the/DT year-ago/JJ level/NN ,/, went/VBD down/RBR 2/CD 1/8/CD to/TO 33/CD on/IN 1.1/CD million/CD shares/NNS

Jaguar/NNP 's/POS American/JJ depository/NN receipts/NNS were/VBD up/IN 3/8/CS yesterday/NN in/IN a/DT down/NN market/NN ,/, closing/VBG at/IN 10/CD ...

Problemi Nasıl Çözebiliriz?

- Elimizde büyük boyutlu işaretlenmiş bir derlem olduğunu kabul edersek, her sözcük için derlemde en sık görülen işareti seçebiliriz.
 - Bu standard bir test kümesi üzerinde sınıfların yaklaşık % 92'sini doğru kestirir.
 - 1 milyon sözcük işaretliyorsunuz, ≈ 80 bin sözcük yanlış işaretleniyor ve de bunların hangilerinin olduğunu bilmiyorsunuz!
 - Ortalama her tümcede en az bir yanlış işaret var.
- Sözcüğün çevrimine bakıp kurallar yazabiliriz.
 - Hemen önce gelen sözcükler ve onların işaretleri
 - Hemen sonra gelen sözcükler ve işaretleri
 - the/DT ...
 - to/TO ...
 - John/NNP 's/POS blue/JJ ...
- Ancak elle kurallar yazmak ömür törpüsü (:-)

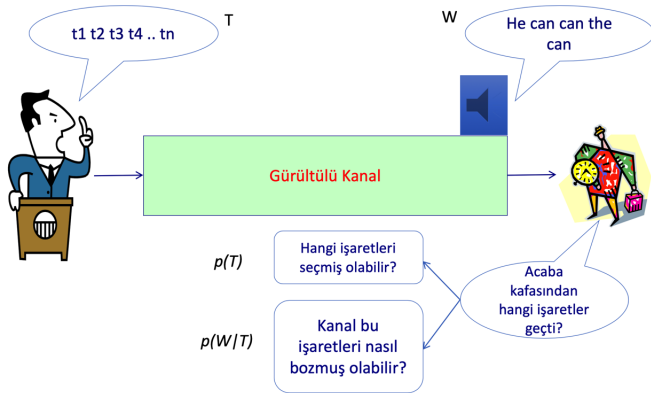
İşaretleme için Markov Modelleri

- **Önceden işaretlenmiş** bir eğitim derlemi olduğunu kabul edelim.
 - Yukardaki örnekler gibi.
- Problemi bir gürültülü kanal problemi olarak modelleyebiliriz:
 - “Kafamda İngilizce bir tümcenin sözcüklerinin işaret dizisi, $\mathbf{T} = \langle t_1, t_2, \dots, t_n \rangle$, var”
 - “Bunlar ağzımdan çıkarken $\mathbf{W} = \langle w_1, w_2, \dots, w_n \rangle$, sözcüklerine dönüşüyor (yani bunları gözlüyoruz!)”
 - “ $\mathbf{W}$ gözlemini üretebilecek en olası işaret dizisi $\hat{\mathbf{T}}$ ne olabilir?”
- Bunu matematiksel olarak şu şekilde ifade edebiliriz:

$$\hat{\mathbf{T}} = \arg \max_{\mathbf{T}} p(\mathbf{T} \mid \mathbf{W})$$

- Burada $\hat{\mathbf{T}}$, $\arg \max$ içindeki değerin en büyük olanına denk gelen işaret dizisidir.

“Gürültülü Kanal” Modeli



$$\hat{T} = \arg \max_T p(\mathbf{T} | \mathbf{W}) = \underbrace{\arg \max_T \frac{p(\mathbf{W} | \mathbf{T})p(\mathbf{T})}{p(\mathbf{W})}}_{\text{Bayes Açılımı}} = \underbrace{\arg \max_T \overbrace{p(\mathbf{W} | \mathbf{T})}^{\text{Kanal Modeli}} \overbrace{p(\mathbf{T})}^{\text{Kaynak Modeli}}}_{\text{Payda önemli değil}}$$

Temel Denklem ve Basitleştirici Önkabuller

$$\hat{T} = \arg \max_T p(\mathbf{T} | \mathbf{W}) = \underbrace{\arg \max_T \frac{p(\mathbf{W} | \mathbf{T})p(\mathbf{T})}{p(\mathbf{W})}}_{\text{Bayes Açılımı}} = \underbrace{\arg \max_T \overbrace{p(\mathbf{W} | \mathbf{T})}^{\text{Kanal Modeli}} \overbrace{p(\mathbf{T})}^{\text{Kaynak Modeli}}}_{\text{Payda önemli değil}}$$

- **Bağımsızlık** önkabulu: Bir sözcüğün gözlenme olasılığı sadece işareti tarafından belirlenir; civardaki sözcükler ve onların işaretlerinden bağımsızdır. Kanal modelini şu şekilde yazabiliriz

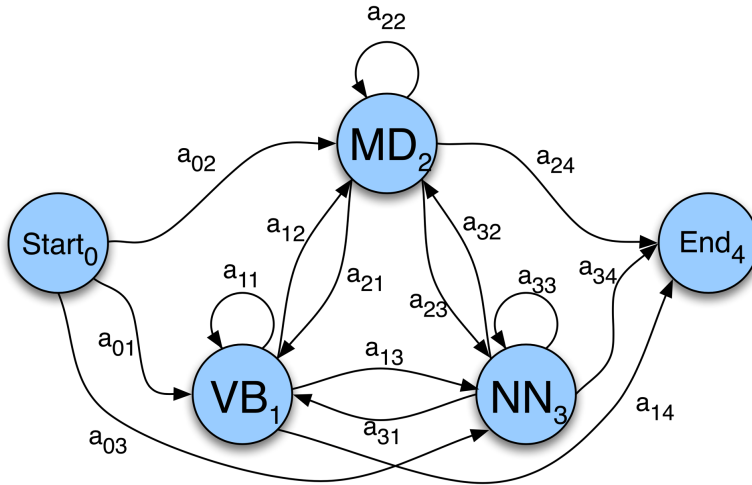
$$p(\mathbf{W} | \mathbf{T}) = p(w_{1:n} | t_{1:n}) \approx \prod_{i=1}^n p(w_i | t_i)$$

- **2-gram** önkabulu: Markov kısıtı – bir işaretin olasılığı sadece kendinden bir önceki işarete bağlıdır. Zincir kuralını uygulayıp kaynal modelini şu şekilde yazabiliriz:

$$p(\mathbf{T}) = p(t_{1:n}) \approx \prod_{i=1}^n p(t_i | t_{i-1})$$

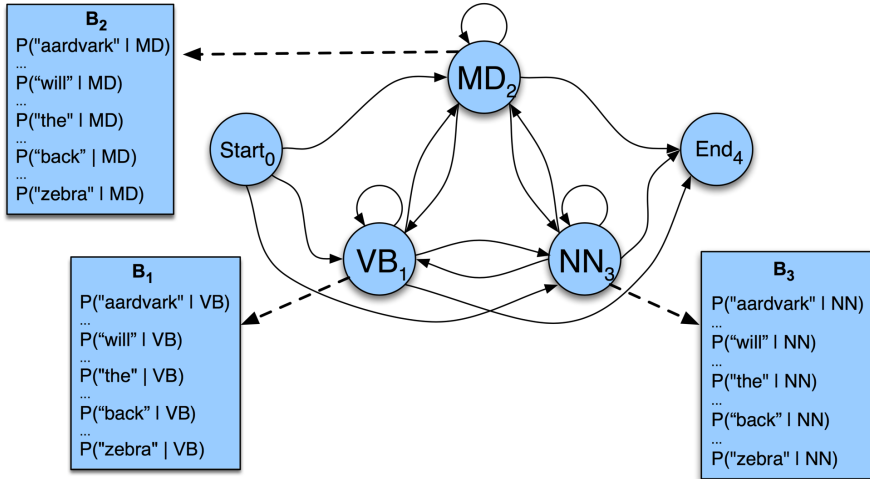
$$\hat{\mathbf{T}} = \hat{t}_{1:n} = \arg \max_{t_{1:n}} p(t_{1:n} \mid w_{1:n}) \approx \arg \max_{t_{1:n}} \prod_{i=1}^n \underbrace{p(w_i \mid t_i)}_{\text{üretim olasılığı}} \underbrace{p(t_i \mid t_{i-1})}_{\text{geçiş olasılığı}}$$

$p(t_i | t_{i-1})$ 'nin Kuşbakışı Görünümü



a_{ij} = işaretler arası geçiş olasılıkları.

$p(w_i | t_i)$ 'nin Kuşbakışı Görünümü



b_{ij} = bir işaretin sözcükleri üretme olasılığı

Olasılıkları Nasıl Kestiriyoruz:

- Olasılıkları işaretlenmiş eğitim derleminden kestirebiliriz.
- Geçiş olasılıkları: $p(t_i | t_{i-1}) = \frac{c(t_{i-1}, t_i)}{c(t_{i-1})}$
- Sözcük üretme olasılıkları: $p(w_i | t_i) = \frac{c(t_i, w_i)}{c(t_i)}$
 - Olasılıkların 0 olduğu durumlarda standart düzgünleştirme yöntemleri ile bunları 0'dan büyük yapmamız gerekir.
- Geçiş modelini 3-gram kullanarak biraz daha iyileştirebiliriz..

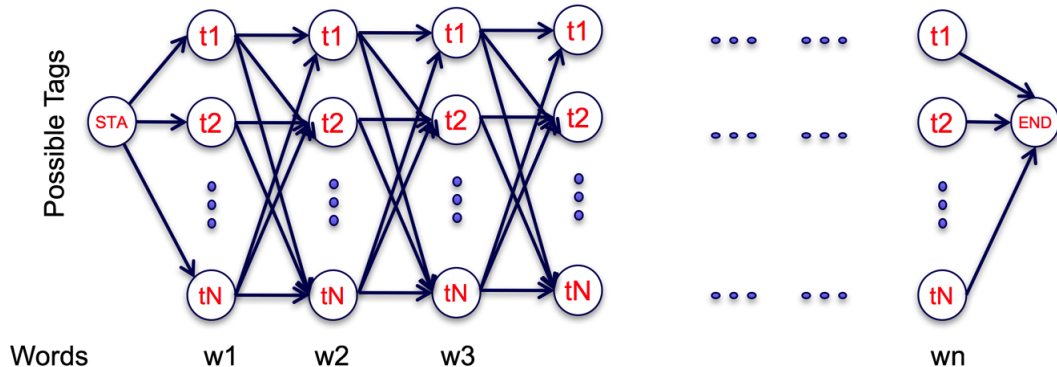
$$p(t_{1:n}) \approx \prod_{i=1}^n \underbrace{p(t_i | t_{i-2}t_{i-1})}_{t_i \text{ son iki işarete bağımlı}}$$

- Bu daha fazla sayıda geçiş olasılığı kestirmeyi gerektirir.

- $W = \langle w_1, w_2, \dots, w_n \rangle$ şeklinde n adet sözcükten oluşan bir dizinimiz var.
- Markov Model durumlarını oluşturan N boyutlu bir işaret kümemiz var (ve *STArt* ve *END* durumları).
- q_i , i 'inci sözcük gözlendikten sonraki durum/işaret.
- $\lambda = (A, B)$ modelimizin parametreleri: geçiş (A) ve üretme (B) olasılıkları
- Sonraki bir dizi şekilde
 - i sözcük pozisyonlarını
 - j ve k durumları/işaretlerigösterecek.

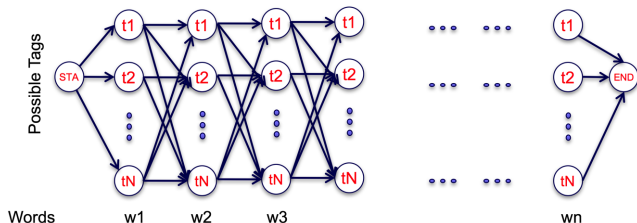
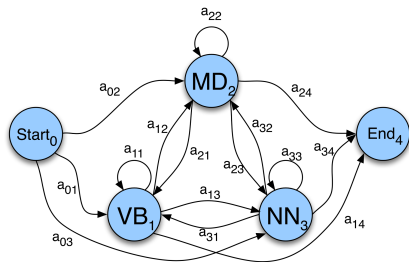
Modelin Zamanda Açılımı

Bir tümce verildiğinde sonlu durumlu geçiş modelini **kafes** şeklinde zamanda açabiliriz.

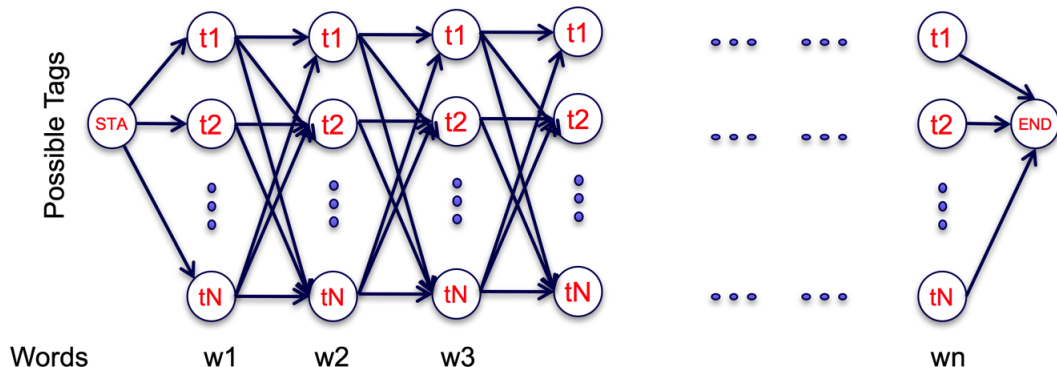


Modelin Zamanda Açılımı

Bir tümce verildiğinde sonlu durumlu geçiş modelini **kafes** şeklinde zamanda açabiliriz.



Modelin Zamanda Açılımı



- Amacımız gözlenen sözcük dizisi verildiğinde, START (0) ile END ($F = n + 1$) durumları arasındaki **en olası yolu** (etkin bir yöntemle) bulmak.
- Yani, model, sözcükleri üretmek için hangi **gizli** durumlardan geçiyor?
- Ancak üssel sayıda olası yol var!

Viterbi Çözümü

- Viterbi algoritması $v_i(j) = \max_{q_0, q_1, \dots, q_{i-1}} p(q_0, q_1, \dots, q_{i-1}, w_1, w_2, \dots, w_i, q_i = j \mid \lambda)$ 'i hesaplar.
- $v_i(j)$
 - $q_i = j$ durumuna bir $q_0, q_1, \dots, q_{i-1}$ durum dizisinden geçerek geline,
 - gözlenen ilk i sözcüğü ($w_1, w_2, \dots, w_i$) üretebilmenin en yüksek olasılığını tanımlar.
- Özyineli olarak şu şekilde tanımlayabiliriz

$$v_i(j) = \max_{k=1 \dots N} v_{i-1}(k) \cdot a_{kj} \cdot b_j(w_i)$$

- Geri takip bağlantıları ise (yani “Hangi önceki işaret en yüksek olasılığı verdi?”) şu şekilde hesaplanır:

$$bt_i(j) = \arg \max_{k=1 \dots N} v_{i-1}(k) \cdot a_{kj} \cdot b_j(w_i)$$

- Bunları geriye takip edip bizi q_n 'e getiren en olası durum dizisini bulabiliriz.
 $q_0 = \text{STArt}, q_1, q_2, \dots, q_n$

Viterbi Çözümü

- **Başlangıç:**

$$\begin{aligned}v_1(j) &= a_{0j} \cdot b_j(w_1) \quad 1 \leq j \leq N \\bt_1(j) &= 0\end{aligned}$$

- **Özyineleme:**

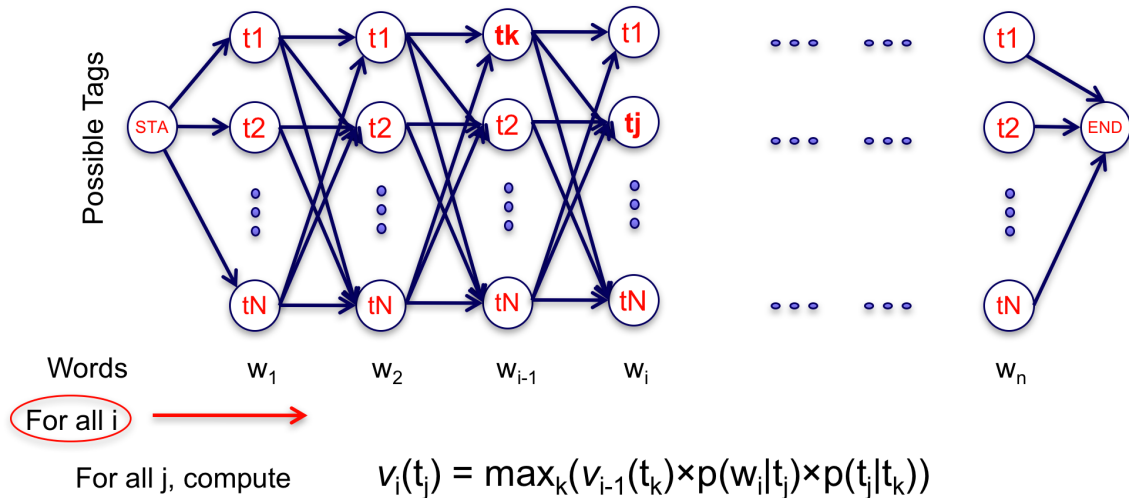
$$v_i(j) = \max_{k=1 \dots N} v_{i-1}(k) \cdot a_{kj} \cdot b_j(w_i) \quad 1 \leq j \leq N, 1 < i \leq n$$

$$bt_i(j) = \arg \max_{k=1 \dots N} v_{i-1}(k) \cdot a_{kj} \cdot b_j(w_i) \quad 1 \leq j \leq N, 1 < i \leq n$$

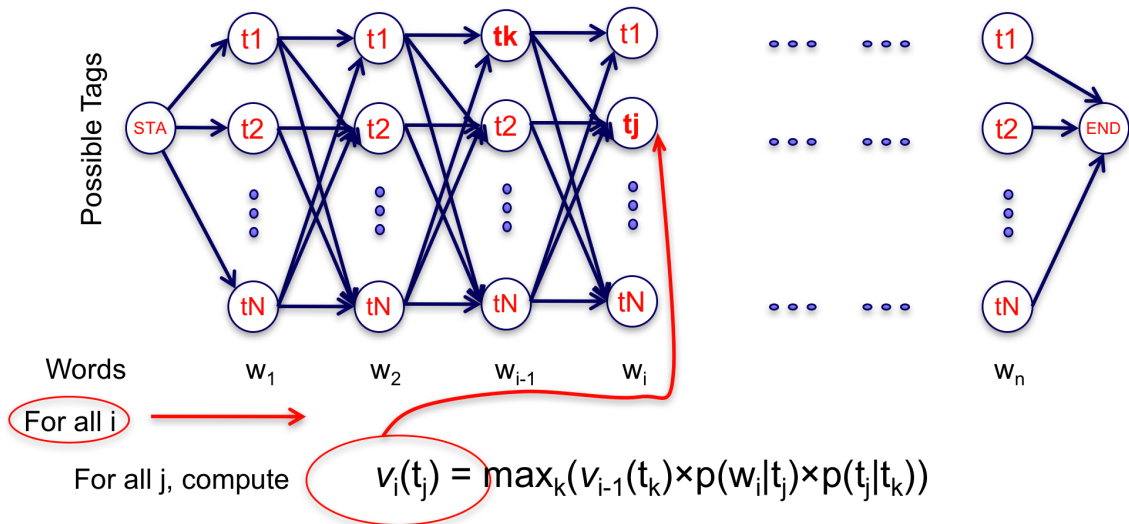
- **Bitiş:**

$$\begin{aligned}p^* = v_n(q_F) &= \max_{k=1 \dots N} v_n(k) \cdot a_{jF} && \text{en büyük olasılık} \\q_n^* = bt_n(q_F) &= \arg \max_{k=1 \dots N} v_n(k) \cdot a_{jF} && \text{geri takip başlangıç durumu}\end{aligned}$$

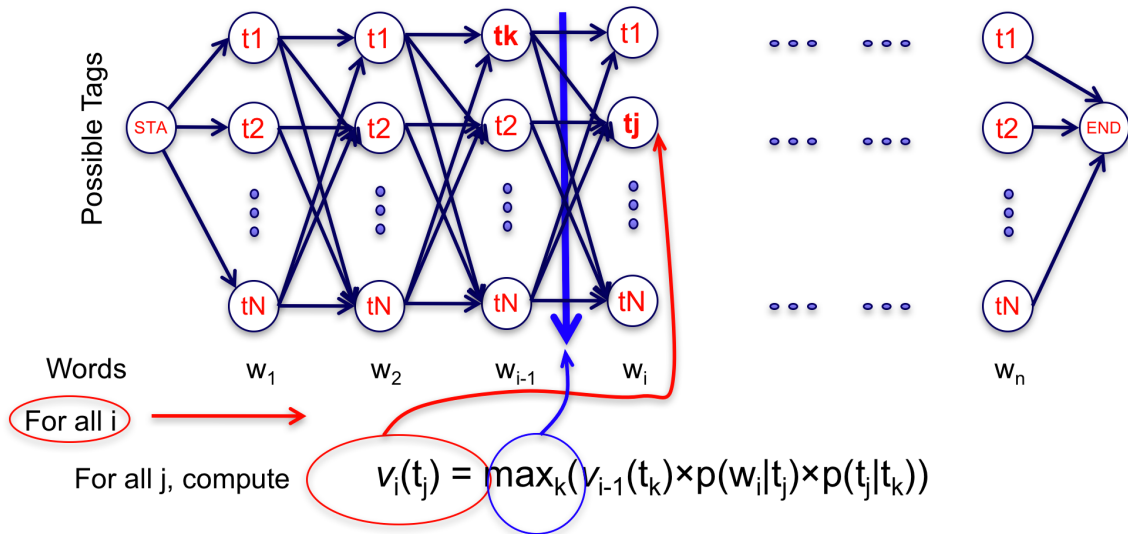
Viterbi Çözümü



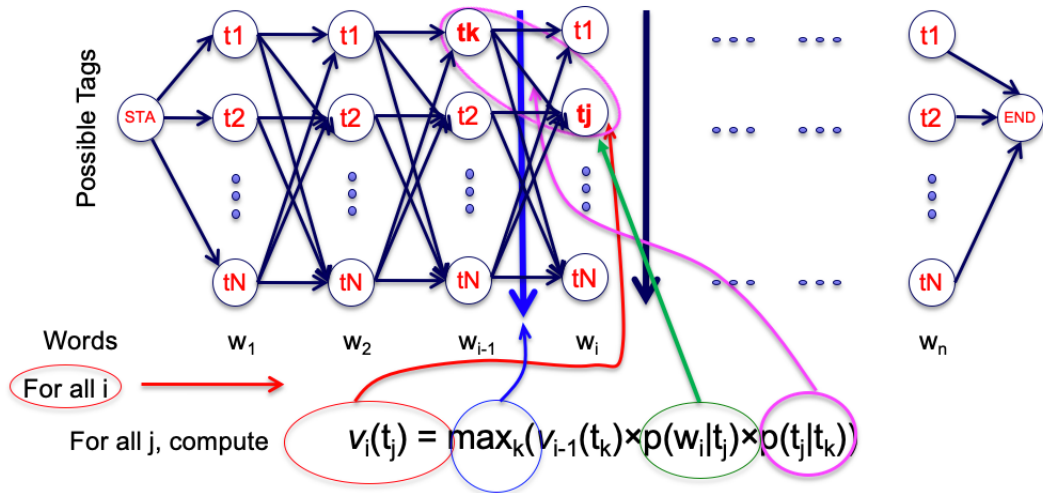
Viterbi Çözümü



Viterbi Çözümü



Viterbi Çözümü



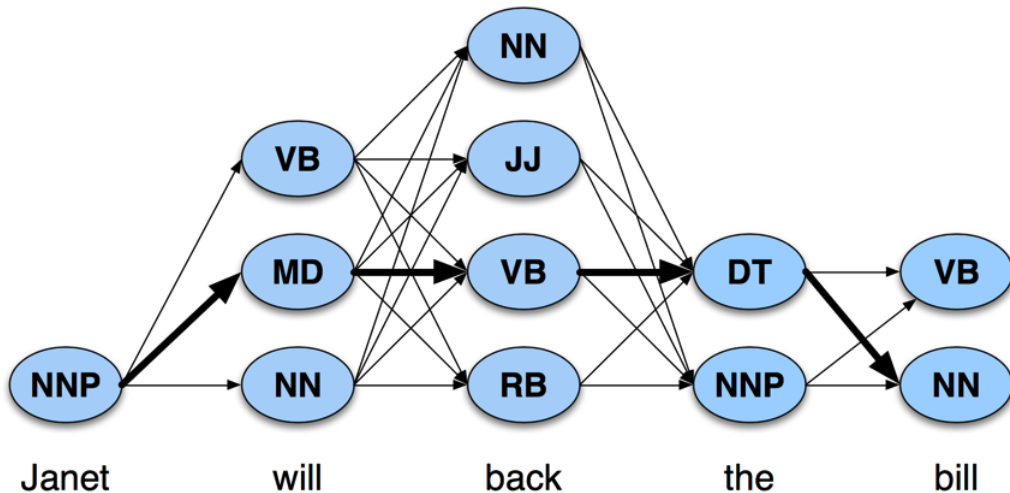
	NNP	MD	VB	JJ	NN	RB	DT
< <i>s</i> >	0.2767	0.0006	0.0031	0.0453	0.0449	0.0510	0.2026
NNP	0.3777	0.0110	0.0009	0.0084	0.0584	0.0090	0.0025
MD	0.0008	0.0002	0.7968	0.0005	0.0008	0.1698	0.0041
VB	0.0322	0.0005	0.0050	0.0837	0.0615	0.0514	0.2231
JJ	0.0366	0.0004	0.0001	0.0733	0.4509	0.0036	0.0036
NN	0.0096	0.0176	0.0014	0.0086	0.1216	0.0177	0.0068
RB	0.0068	0.0102	0.1011	0.1012	0.0120	0.0728	0.0479
DT	0.1147	0.0021	0.0002	0.2157	0.4744	0.0102	0.0017

Figure 10.5 The A transition probabilities $P(t_i|t_{i-1})$ computed from the WSJ corpus without smoothing. Rows are labeled with the conditioning event; thus $P(VB|MD)$ is 0.7968.

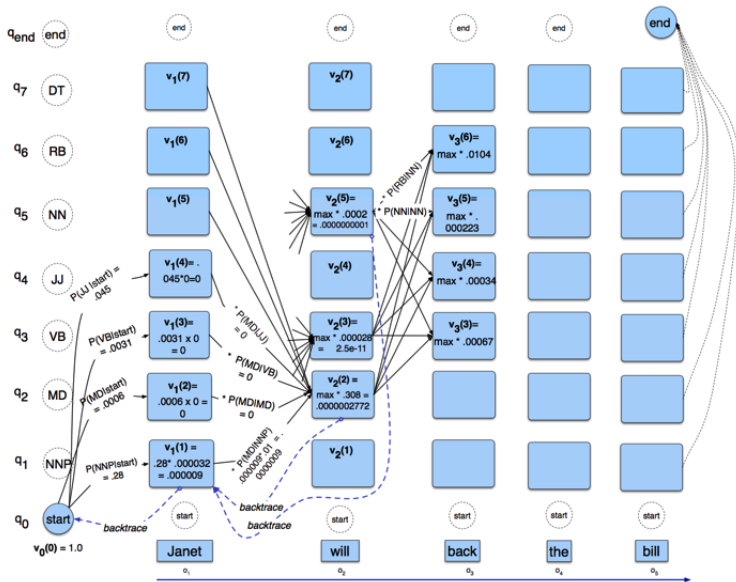
	Janet	will	back	the	bill
NNP	0.000032	0	0	0.000048	0
MD	0	0.308431	0	0	0
VB	0	0.000028	0.000672	0	0.000028
JJ	0	0	0.000340	0.000097	0
NN	0	0.000200	0.000223	0.000006	0.002337
RB	0	0	0.010446	0	0
DT	0	0	0	0.506099	0

Figure 10.6 Observation likelihoods B computed from the WSJ corpus without smoothing.

Viterbi Çözümü



Viterbi Çözümü



Elimizde Eğitim Derlemi Yoksa?

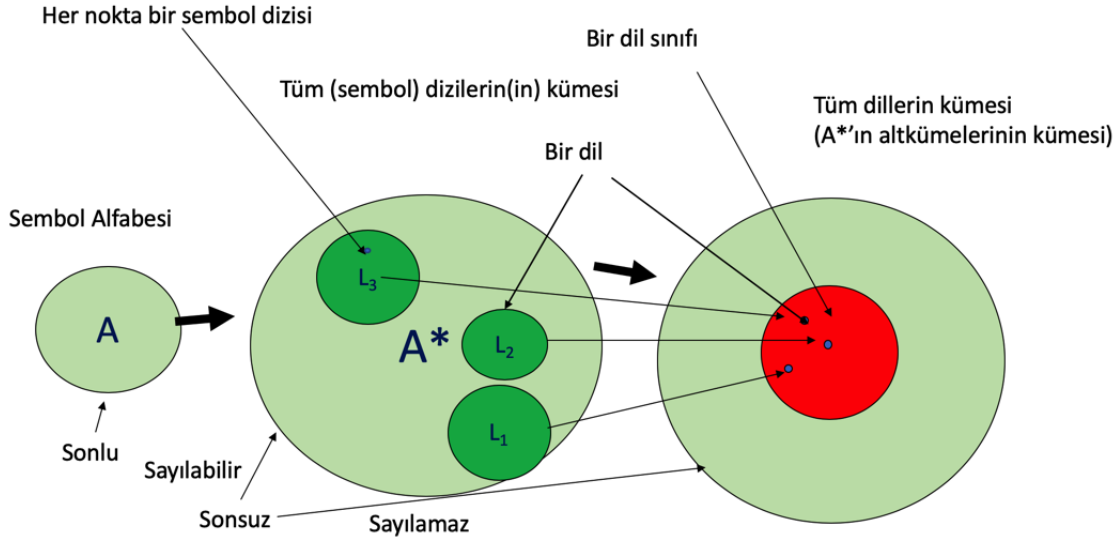
- Elimizde sadece gözlenen sözcük dizisi varsa ve işaret kümesi belli ise
- Baum-Welch / İleri-Geri (Forward-Backward)/Beklenti-Eniyilemesi (Expectation Maximization) algoritması, birörnek dağılımlarla başlayarak A ve B parametrelerini hesaplayabilir.
- Ancak bir miktar elle işaretlenmiş tümce ile başlangıç olasılıklarını kestirerek başlamak parametrelerin daha iyi öğrenilmesini sağlıyor.
 - Ne kadar fazla elle işaretli derlem, o kadar iyi öğrenme!

İşaretleme İçin Başka Yaklaşımlar

- Saklı Markov Modelleri bir üretken modeller sınıfına girer.
 - İşaret dizisinden sözcükleri üretir.
- Ayrıştırıcı modeller $\arg \max_T p(T | W)$ ile T 'yi direk olarak hesaplar.
 - En fazla entropi Markov modelleri (Maximum Entropy Markov Models)
 - Şartlı Rassal Alanlar (Conditional Random Fields)
- Sonuçta her zaman diliminde sonlu bir kümeden işareti seçen herhangi bir makina öğrenmesi yöntemi de kullanılabilir (örn. SVM'ler, özyinelemeli yapay sinir ağları, vb.)

- 3 dakikada biçimsel diller (:-).
- Sonlu durumlu dönüştürücüler
- Biçimbirimsel çözümlemenin alt işlevleri
- Biçimbirimsel çözümleme

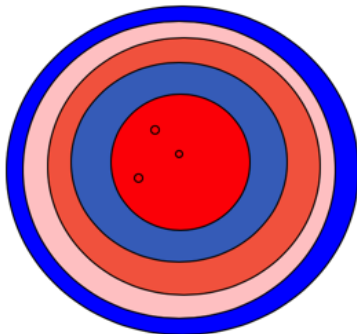
3 Dakikada Biçimsel Diller – Kuşbakışı Görünüş



3 Dakikada Biçimsel Diller – Kuşbakışı Görünüş

- Chomksy Dil Hiyerarşisi

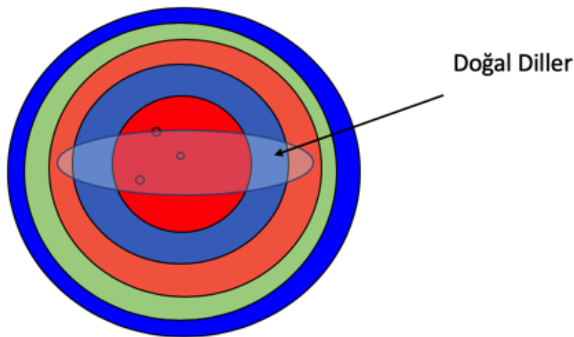
- Düzgün diller (Regular Languages)
- Çevrimden Bağımsız Diller (Context Free Languages)
- Çevrime Bağımlı Diller (Context Sensitive Languages)
- Tanınabilen Diller (Recognizable Languages/Recursively Enumerable Languages)



3 Dakikada Biçimsel Diller – Kuşbakışı Görünüş

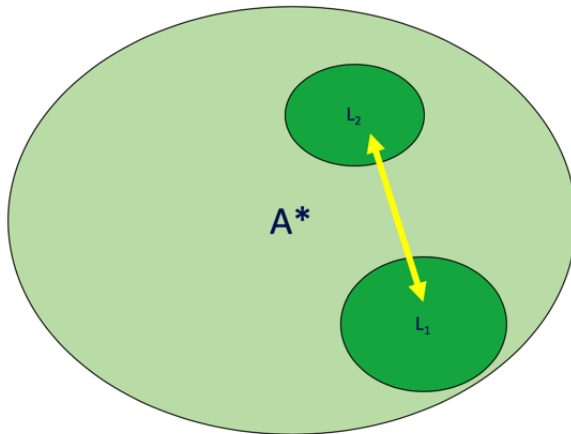
- Chomksy Dil Hiyerarşisi

- Düzgün diller (Regular Languages)
- Çevrimden Bağımsız Diller (Context Free Languages)
- Çevrime Bağımlı Diller (Context Sensitive Languages)
- Tanınabilen Diller (Recognizable Languages/Recursively Enumerable Languages)



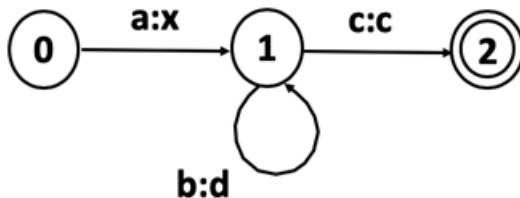
Dilden Dile Dizin Dönüştürümü

Eğer L_1 ve L_2 birer düzgün dil ile bunların arasındaki aktarım bir **Sonlu Durumlu Dönüştürücü** ile gerçekleştirilebilir.



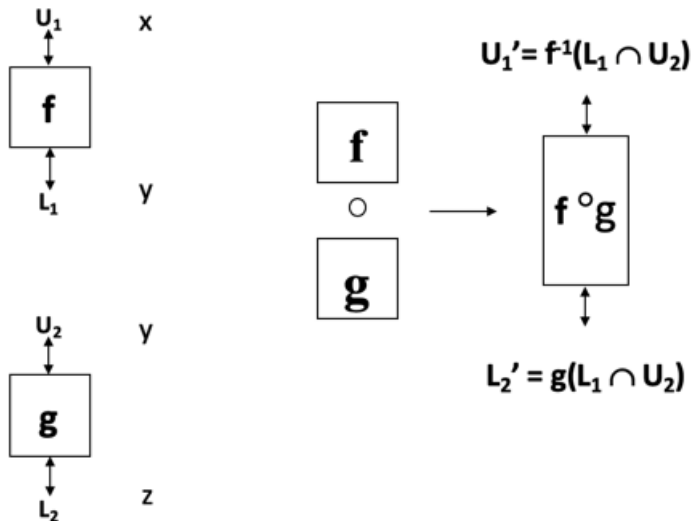
Dilden Dile Dizin Dönüştürümü

- Örneğin L_1 ab^*c düzgün deyimi (regular expression) L_2 de xd^*c düzgün deyimi tarafından tanımlanmışsa



- Bu dönüştürücü örneğin L_1 'deki $abbbbc$ dizinini L_2 'deki $xdddc$ dizinine aktarır.
- Dönüştürücü iki yönlü çalışabilir.

Dönüştürücülerin Birleştirilmesi



Biçimbirimsel çözümleme iki önemli kısımdan oluşur

- Sözcüklerin doğru yazıldığını denetleyip, parçalarına (soyut biçimbirimlere) ayırmak.
- Biçimbirimlerin doğru sırada olduğunu denetleyip her birini çözümleme sembollerine aktarmak.

Biçimbirimsel Çözümleme – 1

- Ünlü uyumları

Soyut Biçimbirimler	masa+l A r	okul+l A r	ev+l A r	gül+l A r
Ayrıştırma	masa 0 l a r	okul 0 l a r	ev 0 l e r	gül 0 l e r
Yazılış	masalar	okullar	evler	güller
Soyut Biçimbirimler	masa+y H	okul+y H	ev+y H	sürü+y H
Ayrıştırma	masa 0 y ı	okul 00 u	ev 00 i	sürü 0 y ü
Yazılış	masayı	okulu	evi	sürüyü
Soyut Biçimbirimler	hilal+l A r		alkol+y H	
Ayrıştırma	hilal 0 l e r	(hilal 0 l a r ×)	alkol 00 ü	(alkol 00 u ×)
Yazılış	hilaller		alkolü	

- **0** düşmeyi, **A** {a, e} kümesini, **H** {ı, i, u, ü} kümesini gösterir.

Biçimbirimsel Çözümleme – 2

- Ünsüz uyumları/düşmeleri/çiftlemeleri

Soyut Biçimbirimler	masa+ H m	ağla+ H yor
Ayrıştırma	masa 00 m	ağl 00 1 yor
Yazılış	masam	ağlıyor

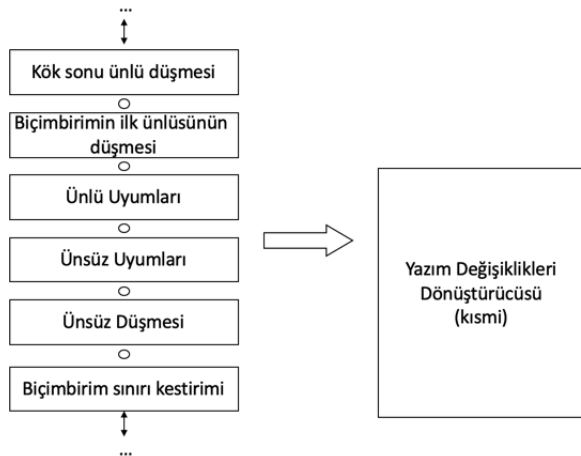
Soyut Biçimbirimler	kitab+ DA	tad+ DH k	saç+ DA	kitab b
Ayrıştırma	kitap 0 ta	tat 0 t 1 k	saç+ ta	kitap p
Yazılış	kitapta	tattık	saçta	kitap

Soyut Biçimbirimler	tı b0 + yH	üs 0 + sH	şık 0 + yH	hak 0 + nH n
Ayrıştırma	tı b 00 1	üs s00 ü	şık k00 1	hak k00 1 n
Yazılış	tıbbı	üssü	şıkkı	hakkın

- **D** {*d*, *t*} kümesini gösterir.

Yazım Değişiklikleri Dönüştürücüsü

- Tüm biçimbirim sınırı kestirimleri ve yazım değişimleri çeşitli formalizmalar kullanılarak sonlu durumlu dönüştürücülerle modellenenir.²



²Bu başlı başına bir 4 saatlik sunum olur. Sondaki kaynaklarda bulabilirsiniz.

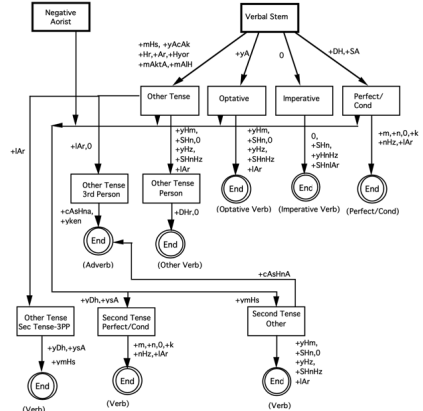
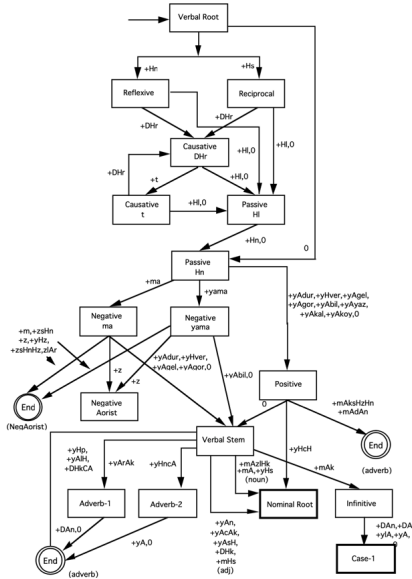
Biçimbirim Sırası Denetleme ve Sembol Aktarma

- Türkçede (ve hemen hemen tüm dillerde) biçimbirim sırası da sonlu durumlu dönüştürücüler ile modellenenebilir ve denetlenebilir.

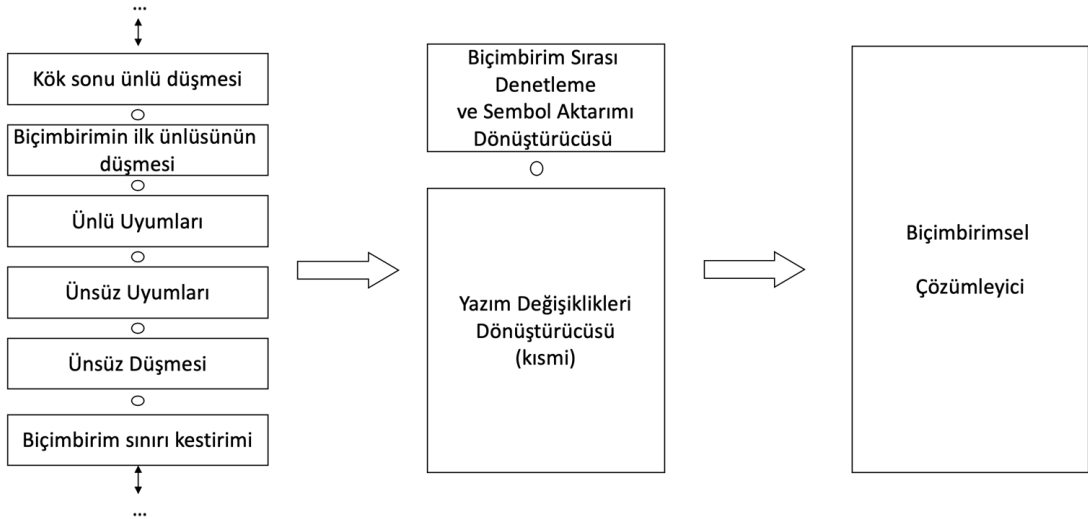
- `evimdekiler` → `ev+Hm+DA+ki+lAr` ✓
→ `ev+Noun+A3sg+P1sg+Loc^DB+Adj+Rel^DB+Noun+Zero+A3pl+Pnon+Nom`
- `evdekilerim` → `ev+DA+ki+lAr+Hm` ✓
→ `ev+Noun+A3sg+Pnon+Loc^DB+Adj+Rel^DB+Noun+Zero+A3pl+P1sg+Nom`
- `evkidelerim` → `ev+ki+DA+lAr+Hm` ✗

- `ruhsatlandırılabilmesindeki` → `ruhsat+lAn+dHr+Hl+yAmA+mA+sH+ndA+ki` ✓
→ `ruhsat+Noun+A3sg+Pnon+Nom^DB+Verb+Acquire^DB+Verb+Caus^DB+Verb+Pass^DB+Verb+Able+Neg^DB+Noun+Inf2+A3sg+P3sg+Loc^DB+Adj+Rel`
- `ruhsatlanılamamasındaki` → `ruhsat+lAn+Hl+dHr+yAmA+mA+sH+ndA+ki` ✗

Türkçe Eylem Biçimbirim Sırası Modeli



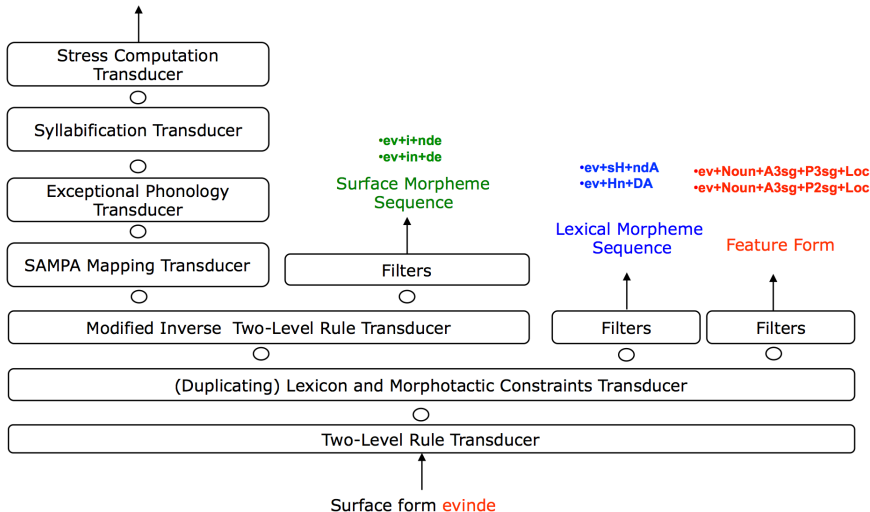
Kuşbakışı Görünüş



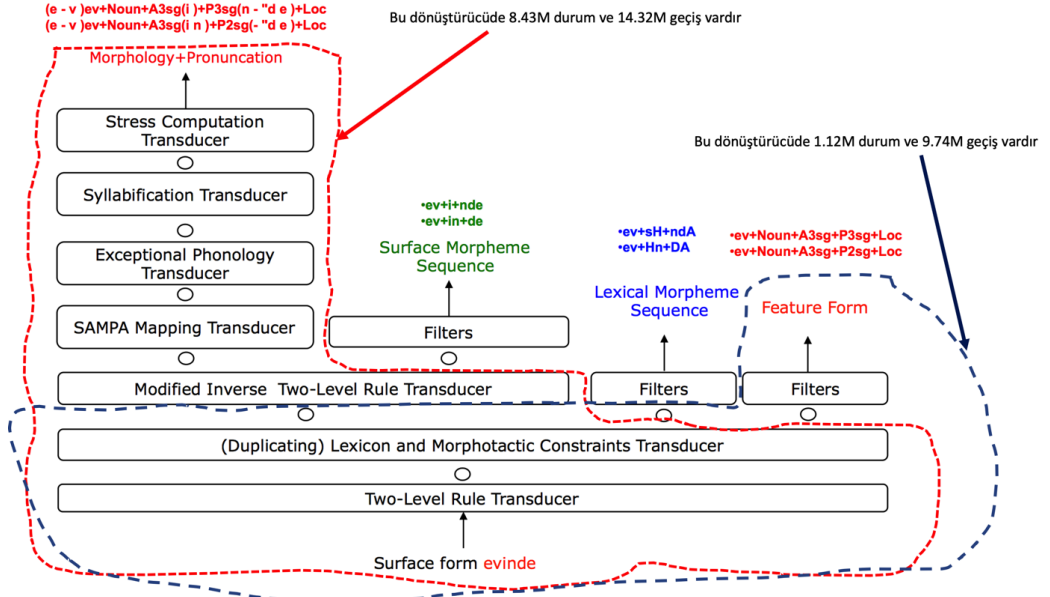
Gerçek Kuşbakışı Görünüş

(e - v)ev+Noun+A3sg(i)+P3sg(n - "d e)+Loc
(e - v)ev+Noun+A3sg(i n)+P2sg(- "d e)+Loc

Morphology+Pronunciation



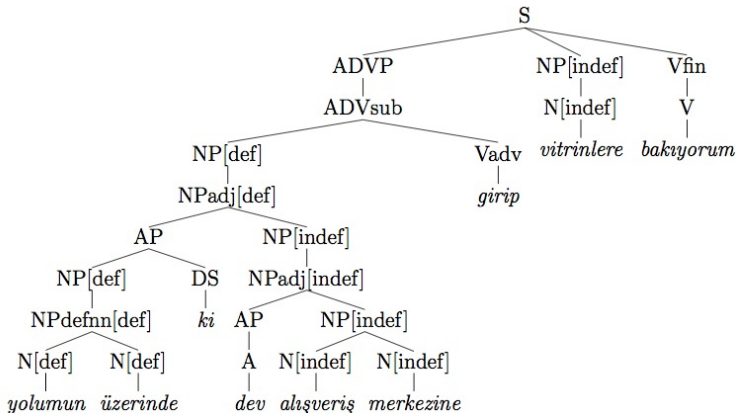
Gerçek Kuşbakışı Görünüş



- Sözdizim çözümlemesi bir tümceyi sözdizimsel öğelerine ayırır.
 - Özne, Nesne, Eylem, Niteleyici,
- Çıktıdaki gösterim çizgisel değildir; hangi sözdizim kuramını kullandığınıza göre değişir.
- Bir tümcenin genelde çok sayıda sözdizimsel çözümü vardır; istatistiksel modeller kullanılarak en olası çözüm elde edilir.
 - I saw the woman with a telescope.
 - I saw the woman with a brown bag.
 - Ali Ayşe'yi eve gelirken görmüş.

Yolumun üzerinde dev
alışveriş merkezine
girip vitrinlere
bakıyorum.

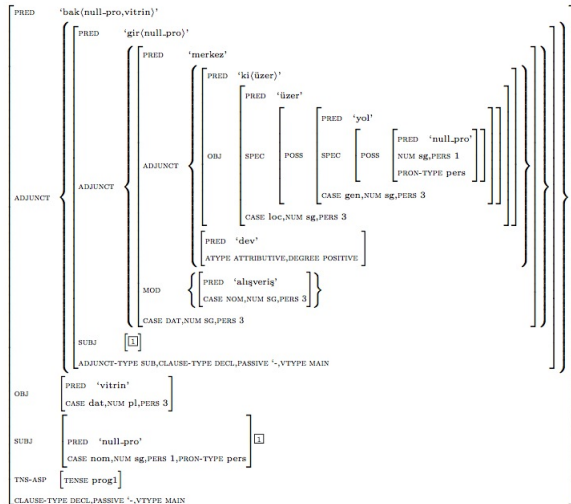
→



- Çevrimden bağımsız gramer ile modelleme

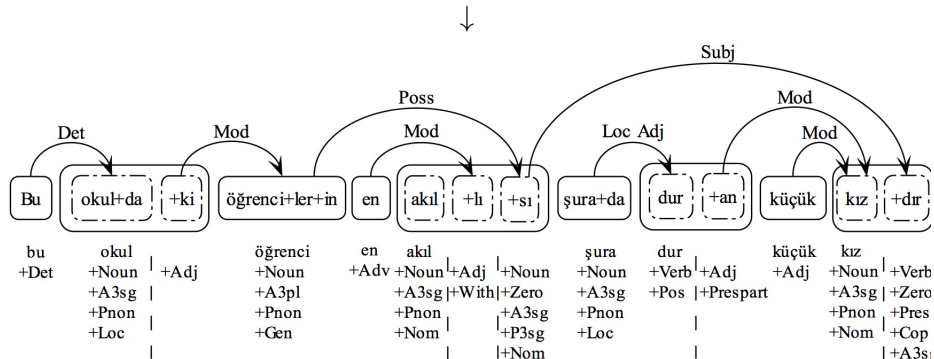
Sözdizim Çözümlemesi

Yolumun üzerinde dev
alışveriş merkezine
girip vitrinlere
bakıyorum.



- Çevrimden bağımsız gramer ve sözcük özellikleri ile modelleme

Bu okuldaki öğrencilerin en akıllısı şurada duran küçük kızdır.



This school-at+that-is student-s-' most intelligence+with+of there stand+ing little girl+is
The most intelligent of the students in this school is the little girl standing there.

- Bağıllık grameri ile modelleme.

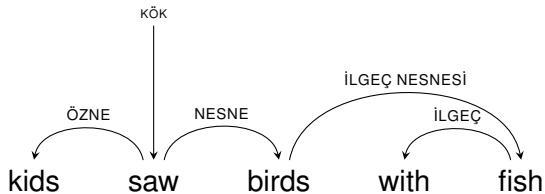
Çözümleme Teknikleri

- Çevrimden bağımsız gramer kullanacaksanız ve de istatistik elde etmek için elinizde öbekler kullanan bir ağaç yapılı derlem var ise
 - Cocke–Younger–Kasami (CYK) algoritması ile $O(n^3)$ adımla, en olası çözüm ağacını bulabilirsiniz (ancak gramerinizi Chomsky Normal Formuna dönüştürmeniz gerekiyor!)
 - Earley Tablo Çözümleme algoritması ile $O(n^3)$ işlemle en olası çözüm ağacını bulabilirsiniz.
- Bağlılık gramer kullanacaksanız ve de istatistik elde etmek için elinizde bağlılık gösterimi kullanan bir ağaç yapılı derlem var ise
 - Geçiş tabanlı çözümleme algoritması ile $O(n)$ adımla en olası çözüm ağacını bulabilirsiniz.
 - Chu-Liu-Edmonds'un en büyük kapsayan ağacı (maximum spanning tree) bulan algoritması ile $O(n^2)$ adımla en olası çözüm ağacını bulabilirsiniz.
- Hiç bir çözümleme algoritması her zaman (%100 doğrulukla) en olası çözümü bulmaz.
- Üssel sayıda çözüm ağacı olabilir: hepsini üretemezsiniz!

Bağılılık Ağacı: Tanım

- $x = [x_1, \dots, x_n]$ bir tümce olsun. Buna KÖK olarak bir “ x_0 ” sembolü ekleriz
- Bağılılık ağacı $[p, c, \ell]$ şeklinde üçlülerden oluşur. Burda
 - $p \in \{0, \dots, n\}$ bir *iye* (*parent*) sözcük indeksidir
 - $c \in \{1, \dots, n\}$ bir *uydu* (*child*) sözcük indeksidir.
 - $\ell \in \mathcal{L}$ bağılılık etiketleridir (özne, nesne, vb.)
- Değişik işaretleme yaklaşımları değişik etiket kümeleri, $\mathcal{L}$ kullanır.³
 - Bir üçlü, x_p 'den x_c 'ye ℓ etiketli bağılılık ilişkisini gösteren bir oku temsil eder.
 - Tüm yönlü oklar KÖKÜ x_0 olan bir yönlü ağaç oluşturur.

³universaldependencies.org



Bazı önemli etiketler:

- Özne
- Nesne
- Dolaylı nesne
- İlgeç nesnesi
- Sıfat
- Zarf

Geçiş-temelli Çözümleme

- Çözümleyici x üzerinden soldan sağa bir kere geçerek bir dizi (geriye döndürülemeyen) çözümleme kararı verir.
- Çözümleyici temelde bir **durum makinasıdır** (bir *sonlu durumlu makina* değil). Makinanın durumu bir *yığın (stack)*, S , ve sonlu bir B girdi belleği tarafından belirlenir.
- Başlangıçta x B 'ye aktarılır; S ise en üstte KÖK ile başlar.

Girdi B

Yığın S

KÖK

we
vigorously
wash
our
cats
who
stink

Girdi B

we
vigorously
wash
our
cats
who
stink

Yığın S

KÖK

- Her hangi bir adımda şu üç adımdan birisine karar veririz:
 - KAYDIRMA: B 'deki bir sonraki sözcüğü S 'nin üstüne ekleriz.
 - SAĞA-OK: $u = \text{pop}(S); v = \text{pop}(S); \text{push}(S, v \rightarrow u)$.
 - En üst iki öğeyi alır, sağa okla birleştirir, geriye koyarız.
 - SOLA-OK: $u = \text{pop}(S); v = \text{pop}(S); \text{push}(S, v \leftarrow u)$.
 - En üst iki öğeyi alır, sola okla birleştirir, geriye koyarız.

ve (gerekirse) kestirilen etiketi de SOLA-OK ve SAĞA-OK geçişlerine ekleriz.
- Çözümleme sırasında (makina öğrenme temelli) bir *sınıflayıcı*, B , ve S 'nin içeriklerine bakarak hangi adımı seçeceğine karar verir.
- Kararın dönüşü yoktur; yani bu karar ilerde gözden geçirilmez!

Yığın S

KÖK

Girdi B

we
vigorously
wash
our
cats
who
stink

Kararlar:

Yığın S

we
KÖK

Girdi B

vigorously
wash
our
cats
who
stink

Kararlar: KAYDIRMA

Yığın S

vigorously
we
KÖK

Girdi B

wash
our
cats
who
stink

Kararlar: KAYDIRMA KAYDIRMA

Yığın S

wash
vigorously
we
KÖK

Girdi B

our
cats
who
stink

Kararlar: KAYDIRMA KAYDIRMA **KAYDIRMA**

Bir Örnek

Yığın S

vigorously wash

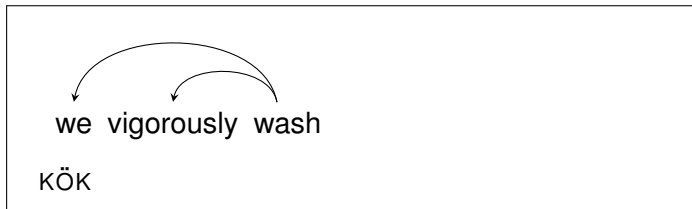
we
KÖK

Girdi B

our
cats
who
stink

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK

Yığın S



Girdi B

our
cats
who
stink

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK

Bir Örnek

Yığın S

our

we vigorously wash

KÖK

Girdi B

cats

who

stink

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA

Bir Örnek

Yığın S

cats
our



we vigorously wash

KÖK

Girdi B

who
stink

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA **KAYDIRMA**

Bir Örnek

Yığın S

our cats

we vigorously wash

KÖK

Girdi B

who
stink

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK

Bir Örnek

Yığın S

who

our cats

we vigorously wash

KÖK

Girdi B

stink

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK KAYDIRMA

Bir Örnek

Yığın S

stink
who

our cats

we vigorously wash

KÖK

Girdi B

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK KAYDIRMA **KAYDIRMA**

Bir Örnek

Yığın S

who stink

our cats

we vigorously wash

KÖK

Girdi B

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK KAYDIRMA KAYDIRMA SAĞA-OK

Bir Örnek

Yığın S

our cats who stink

we vigorously wash

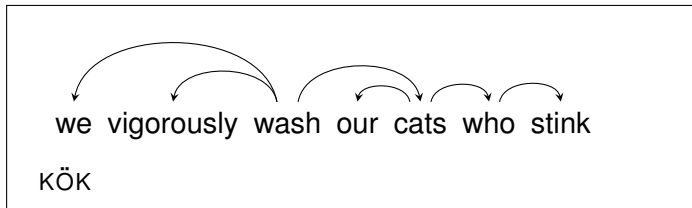
KÖK

Girdi B

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK KAYDIRMA KAYDIRMA SAĞA-OK **SAĞA-OK**

Bir Örnek

Yığın S

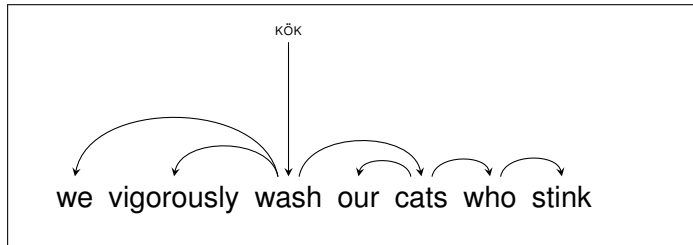


Girdi B

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK KAYDIRMA KAYDIRMA SAĞA-OK **SAĞA-OK**

Bir Örnek

Yığın S



Girdi B

Kararlar: KAYDIRMA KAYDIRMA KAYDIRMA SOLA-OK SOLA-OK KAYDIRMA KAYDIRMA
SOLA-OK KAYDIRMA KAYDIRMA SAĞA-OK SAĞA-OK **SAĞA-OK**

Geçiş-temelli Bağlılık Çözümlemesinde Önemli Noktalar

- Her adımda $\{\text{KAYDIRMA}, \text{SAĞA-OK}, \text{SOLA-OK}\}$ (veya bu ikisinin $\mathcal{L}$ kümesinden etiket eklenmiş halleri (örneğin SAĞA-OK-NESNE , vb.)), bir sınıflayıcı tarafından kestirilir.
- Sınıflayıcının, karar vermek için kullandığı özellikler S , B , ve önceki verilen kararlar dizisinden gelebilir.
- Sınıflayıcı bağlılık-ağacı yapıları derlemdeki ağaçlar “doğru” karar dizisine dönüştürülerek eğitilir.
 - Bu “doğru” karar dizisi doğru ağacı bulmak için yeterlidir.
 - Bu karar dizisi $2 \cdot n \langle \text{durum}, \text{doğru-karar} \rangle$ çiftinden oluşur.
 - Her sözcük bir kere KAYDIRMA bir kere de bağlama olmak üzere iki kararda yer alır.

Bağlılık Çözümü Başarım Değerlendirmesi

- Etiketsiz doğru bağlama skoru: İye-Uydu bağlarının ne kadarı doğru olarak kestirildi?
- Etiketli doğru bağlama skoru: Bağ ve etiketlerin ne kadarı doğru olarak kestirildi?

Team	LAS
1. Stanford (Dozat et al.)	76.30
2. C2L2 (Shi et al.)	75.00
3. IMS (Björkelund et al.)	74.42
4. HIT-SCIR (Che et al.)	72.11
5. LATTICE (Lim and Poibeau)	70.93
6. NAIST SATO (Sato et al.)	70.14
7. Koç University (Kırnap et al.)	69.76
8. ÚFAL (Straka and Straková)	69.52
9. UParse (Vania et al.)	68.87
10. Orange (Heinecke and Asadullah)	68.61
11. TurkuNLP (Kanerva et al.)	68.59
12. darc (Yu et al.)	68.41
13. BASELINE UDPipe 1.1	68.35
14. MQuni (Nguyen et al.)	68.05
15. fbaml (Qian and Liu)	67.87
16. LyS (Vilares and Gómez-Rodríguez)	67.81
17. LIMSI (Aufrant and Wisniewski)	67.72
18. RACAI (Dumitrescu et al.)	67.71
19. IIT Kharagpur (Das et al.)	67.61
20. naistCL (no paper)	67.59
21. Wanghao-ftd-SJTU (Wang et al.)	66.53
22. UALING (Hornby et al.)	65.24
23. Uppsala (de Lhoneux et al.)	65.11
24. METU (Akkuş et al.)	61.98
25. CLCL (Moor et al.)	61.82
26. Mengest (Ji et al.)	61.33
27. ParisNLP (De La Clergerie et al.)	60.02
28. OpenU (More and Tsarfaty)	56.56
29. TRL (Kanayama et al.)	43.07
30. MetaRomance (Garcia and Gamallo)	34.05
31. UT (no paper)	21.10
32. ECNU (no paper)	3.18
33. Wenba-NLU (no paper)	0.58

Table 2: Ranking of the participating systems by the main evaluation metric, the labeled attachment F_1 -score, macro-averaged over 81 test sets.

Teşekkürler

- Türkçe Doğal Dil İşleme

- <https://www.youtube.com/watch?v=CzIUICJXajY>

- İstatistiksel Çeviri (Slaytlar İngilizce)

- <https://www.youtube.com/watch?v=LLSRpOaarNY>

- 11-411 Natural Language Processing
- 15-453 Formal Languages, Automata and Computation

- Daniel Jurafsky ve James H. Martin, **Speech and Language Processing**, Prentice Hall. Upper Saddle River, NJ, 2008
 - <https://web.stanford.edu/~jurafsky/slp3/>
- Chris Manning ve Hinrich Schütze, **Foundations of Statistical Natural Language Processing**, MIT Press. Cambridge, MA: Mayıs 1999.
- Jacob Eisenstein, **Introduction to Natural Language Processing**, MIT Press. Cambridge, MA: Ekim 2019.
- Yoav Goldberg, **Neural Network Models for Natural Language Processing**, Morgan Claypool Publishers, Mayıs 2017.
- Lauri Karttunen and Kenneth Beesley, **Finite State Morphology**, CSLI Press. Stanford, CA: Nisan 2003.



- Kemal Oflazer ve Murat Saraçlar (Editörler), **Turkish Natural Language Processing**, Springer. Ağustos 2018