Natural Language Processing
SoSe 2016

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Part-of-Speech Tagging

Dr. Mariana Neves

May 9th, 2016

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

Part-of-Speech (POS) Tags

- Also known as:
  - Part-of-speech tags, lexical categories, word classes, morphological classes, lexical tags

Plays$_{\text{[VERB]}}$ well$_{\text{[ADVERB]}}$ with$_{\text{[PREPOSITION]}}$ others$_{\text{[NOUN]}}$

Plays$_{\text{[VBZ]}}$ well$_{\text{[RB]}}$ with$_{\text{[IN]}}$ others$_{\text{[NNS]}}$

Examples of POS tags

- **Noun**: book/books, nature, Germany, Sony
- **Verb**: eat, wrote
- **Auxiliary**: can, should, have
- **Adjective**: new, newer, newest
- **Adverb**: well, urgently
- **Number**: 872, two, first
- **Article/Determiner**: the, some
- **Conjuction**: and, or
- **Pronoun**: he, my
- **Preposition**: to, in
- **Particle**: off, up
- **Interjection**: Ow, Eh

Motivation: Speech Synthesis

- Word „content“
  - „Eggs have a high protein content.“
  - „She was content to step down after four years as chief executive.“

(http://www.thefreedictionary.com/content)

Motivation: Machine Translation

- e.g., translation from English to German:
  - „I like ...“
    - „Ich mag ….“ (verb)
    - „Ich wie ...“ (preposition)

Motivation: Syntactic parsing

Your query

I saw the man on the roof

Tagging

```
I/PRP  saw/VBD  the/DT  man/NN  on/IN  the/DT  roof/NN
```

Parse

```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBD saw)
      (NP (DT the) (NN man))
      (PP (IN on)
        (NP (DT the) (NN roof))))))
```

(http://nlp.stanford.edu:8080/parser/index.jsp)

Motivation: Information extraction

- Named-entity recognition (usually nouns)

```
> echo "Inhibition of NF-kappaB activation reversed the anti-apoptotic effect of isochamaejasmin." | ./geniatagger

Inhibition      Inhibition      NN      B-NP    O
of              of              IN      B-PP    O
NF-kappaB       NF-kappaB       NN      B-NP    B-protein
activation      activation      NN      I-NP    O
reversed        reverse         VBD     B-VP    O
the             the             DT      B-NP    O
anti-apoptotic  anti-apoptotic  JJ      I-NP    O
effect          effect          NN      I-NP    O
of              of              IN      B-PP    O
isochamaejasmin isochamaejasmin NN      B-NP    O
.               .               .       O       O
```


(http://www.nactem.ac.uk/tsujii/GENIA/tagger/)

Motivation: Information extraction

- Relation extraction (usually verbs)

```
> echo "Inhibition of NF-kappaB activation reversed the anti-apoptotic effect of isochamaejasmin." | ./geniatagger

Inhibition     Inhibition     NN    B-NP   O
of             of             IN    B-PP   O
NF-kappaB      NF-kappaB      NN    B-NP   B-protein
activation     activation     NN    I-NP   O
reversed       reverse        VBD   B-VP   O
the            the            DT    B-NP   O
anti-apoptotic anti-apoptotic JJ    I-NP   O
effect         effect         NN    I-NP   O
of             of             IN    B-PP   O
isochamaejasmin isochamaejasmin NN  B-NP   O
.              .              .     O      O
```

(http://www.nactem.ac.uk/tsujii/GENIA/tagger/)

Open vs. Closed Classes

- Closed
  - limited number of words, do not grow usually
  - e.g., Auxiliary, Article, Determiner, Conjuction, Pronoun, Preposition, Particle, Interjection

- Open
  - unlimited number of words
  - e.g., Noun, Verb, Adverb, Adjective

POS Tagsets

- There are many parts of speech tagsets
- Tag types
  - Coarse-grained
    - Noun, verb, adjective, ...
  - Fine-grained
    - noun-proper-singular, noun-proper-plural, noun-common-mass, ..
    - verb-past, verb-present-3rd, verb-base, ...
    - adjective-simple, adjective-comparative, ...

POS Tagsets

- Brown tagset (87 tags)
  - Brown corpus

- C5 tagset (61 tags)
- C7 tagset (146 tags!)

- Penn TreeBank (45 tags)
  - A large annotated corpus of English tagset

Penn Treebank Tagset

Number	Tag	Description
1.	CC	Coordinating conjunction
2.	CD	Cardinal number
3.	DT	Determiner
4.	EX	Existential *there*
5.	FW	Foreign word
6.	IN	Preposition or subordinating conjunction
7.	JJ	Adjective
8.	JJR	Adjective, comparative
9.	JJS	Adjective, superlative
10.	LS	List item marker
11.	MD	Modal
12.	NN	Noun, singular or mass
13.	NNS	Noun, plural
14.	NNP	Proper noun, singular
15.	NNPS	Proper noun, plural
16.	PDT	Predeterminer
17.	POS	Possessive ending
18.	PRP	Personal pronoun
19.	PRP$	Possessive pronoun
20.	RB	Adverb

21.	RBR	Adverb, comparative
22.	RBS	Adverb, superlative
23.	RP	Particle
24.	SYM	Symbol
25.	TO	*to*
26.	UH	Interjection
27.	VB	Verb, base form
28.	VBD	Verb, past tense
29.	VBG	Verb, gerund or present participle
30.	VBN	Verb, past participle
31.	VBP	Verb, non-3rd person singular present
32.	VBZ	Verb, 3rd person singular present
33.	WDT	Wh-determiner
34.	WP	Wh-pronoun
35.	WP$	Possessive wh-pronoun
36.	WRB	Wh-adverb

(https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html)

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

POS Tagging

- The process of assigning a part of speech to each word in a text

- Challenge: words often have more than one POS
  - On my back$_{[NN]}$ (noun)
  - The back$_{[JJ]}$ door (adjective)
  - Win the voters back$_{[RB]}$ (adverb)
  - Promised to back$_{[VB]}$ the bill (verb)

Ambiguity in POS tags

- 45-tags Brown corpus (word types)
  - Unambiguous (1 tag): 38,857
  - Ambiguous: 8,844
    - 2 tags: 6,731
    - 3 tags: 1,621
    - 4 tags: 357
    - 5 tags: 90
    - 6 tags: 32
    - 7 tags: 6 (well, set, round, open, fit, down)
    - 8 tags: 4 ('s, half, back, a)
    - 9 tags: 3 (that, more, in)

Baseline method

1. Tagging unambiguous words with the correct label
2. Tagging ambiguous words with their most frequent label
3. Tagging unknown words as a noun

- This method performs around 90% precision

POS Tagging

- Plays well with others
  - Plays (NNS/VBZ)
  - well (UH/JJ/NN/RB)
  - with (IN)
  - others (NNS)
  - Plays$_{[VBZ]}$ well$_{[RB]}$ with$_{[IN]}$ others$_{[NNS]}$

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

Rule-Based Tagging

- Standard approach (two steps):
  1. Dictionaries to assign a list of potential tags
     - Plays (NNS/VBZ)
     - well (UH/JJ/NN/RB)
     - with (IN)
     - others (NNS)
  2. Hand-written rules to restrict to a POS tag
     - Plays (VBZ)
     - well (RB)
     - with (IN)
     - others (NNS)

Rule-Based Tagging

- Some approaches rely on morphological parsing
  - e.g., EngCG Tagger

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

Sequential modeling

- Many of the NLP techniques should deal with data represented as sequence of items
  - Characters, Words, Phrases, Lines, …

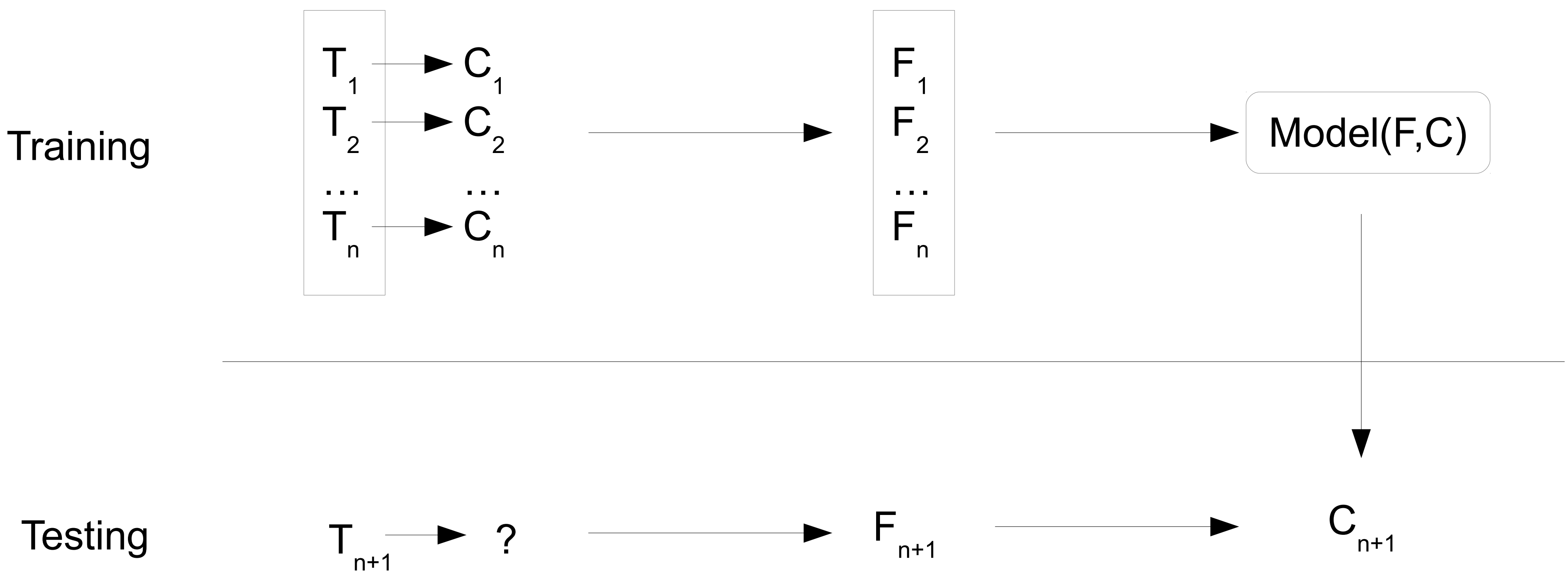
- Part-of-speech tagging
  - $\text{I}_{\text{[PRP]}}$ $\text{saw}_{\text{[VBP]}}$ $\text{the}_{\text{[DT]}}$ $\text{man}_{\text{[NN]}}$ $\text{on}_{\text{[IN]}}$ $\text{the}_{\text{[DT]}}$ $\text{roof}_{\text{[NN]}}$.

- Named-entity recognition
  - $\text{Steven}_{\text{[PER]}}$ $\text{Paul}_{\text{[PER]}}$ $\text{Jobs}_{\text{[PER]}}$ $,_{\text{[O]}}$ $\text{co-founder}_{\text{[O]}}$ $\text{of}_{\text{[O]}}$ $\text{Apple}_{\text{[ORG]}}$ $\text{Inc}_{\text{[ORG]}}$ $,_{\text{[O]}}$ $\text{was}_{\text{[O]}}$ $\text{born}_{\text{[O]}}$ $\text{in}_{\text{[O]}}$ $\text{California}_{\text{[LOC]}}$.

Sequential modeling

- Making a decision based on:
  - Current Observation:
    - Word (W_0): „35-years-old"
    - Prefix, Suffix: „computation" → „comp", „ation"
    - Lowercased word: „New" → „new"
    - Word shape: „35-years-old" → „d-a-a"
  - Surrounding observations
    - Words (W_{+1}, W_{-1})
  - Previous decisions
    - POS tags (T_{-1}, T_{-2})

Learning Model (classification)



Sequential modeling

- Greedy inference
  - Start in the beginning of the sequence
  - Assign a label to each item using the classifier
  - Using previous decisions as well as the observed data

Sequential modeling

- Beam inference
  - Keeping the top k labels in each position
  - Extending each sequence in each local way
  - Finding the best k labels for the next position

Hidden Markov Model (HMM)

- Finding the best sequence of tags ($t_1 \ldots t_n$) that corresponds to the sequence of observations ($w_1 \ldots w_n$)

- Probabilistic View
  - Considering all possible sequences of tags
  - Choosing the tag sequence from this universe of sequences, which is most probable given the observation sequence

$$\hat{t}_1^n = argmax_{t_1^n} P\left(t_1^n | w_1^n\right)$$

Using the Bayes Rule

$$\hat{t}_1^n = argmax_{t_1^n} P(t_1^n | w_1^n)$$

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

$$P(t_1^n | w_1^n) = \frac{P(w_1^n | t_1^n) \cdot P(t_1^n)}{P(w_1^n)}$$

$$\hat{t}_1^n = argmax_{t_1^n} \underbrace{P(w_1^n | t_1^n)}_{\text{likelihood}} \cdot \underbrace{P(t_1^n)}_{\text{prior probability}}$$

Using Markov Assumption

$$\hat{t}_1^n = argmax_{t_1^n} P(w_1^n | t_1^n) \cdot P(t_1^n)$$

$P(w_1^n | t_1^n) \simeq \prod_{i=1}^{n} P(w_i | t_i)$ (it depends only on its POS tag and independent of other words)

$P(t_1^n) \simeq \prod_{i=1}^{n} P(t_i | t_{i-1})$ (it depends only on the previous POS tag, thus, bigram)

$$\hat{t}_1^n = argmax_{t_1^n} \prod_{i=1}^{n} P(w_i | t_i) \cdot P(t_i | t_{i-1})$$

Two Probabilities

- The tag transition probabilities: $P(t_i|t_{i-1})$
  - Finding the likelihood of a tag to proceed by another tag
  - Similar to the normal bigram model

$$P(t_i|t_{i-1})=\frac{C(t_{i-1},t_i)}{C(t_{i-1})}$$

Two Probabilities

- The word likelihood probabilities: $P(w_i|t_i)$
  - Finding the likelihood of a word to appear given a tag

$$P(w_i|t_i)=\frac{C(t_i,w_i)}{C(t_i)}$$

Two Probabilities

I$_{\text{[PRP]}}$ saw$_{\text{[VBP]}}$ the$_{\text{[DT]}}$ man$_{\text{[NN?]}}$ on$_{\text{[]}}$ the$_{\text{[]}}$ roof$_{\text{[]}}$.

$$P([NN]|[DT]) = \frac{C([DT],[NN])}{C([DT])}$$

$$P(man|[NN]) = \frac{C([NN],man)}{C([NN])}$$

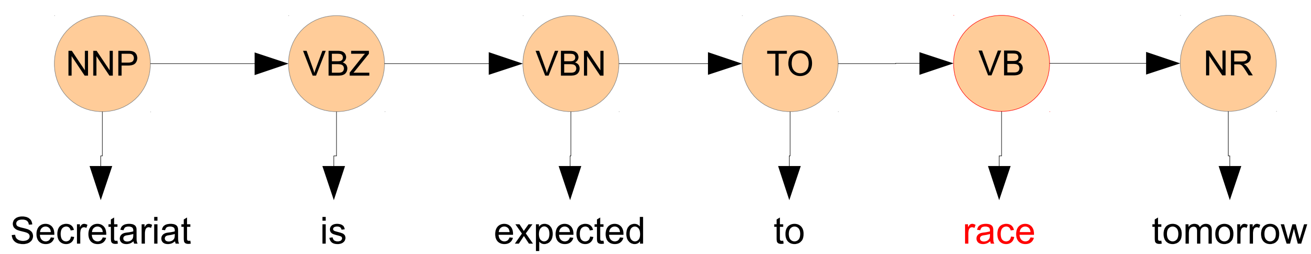
Ambiguity

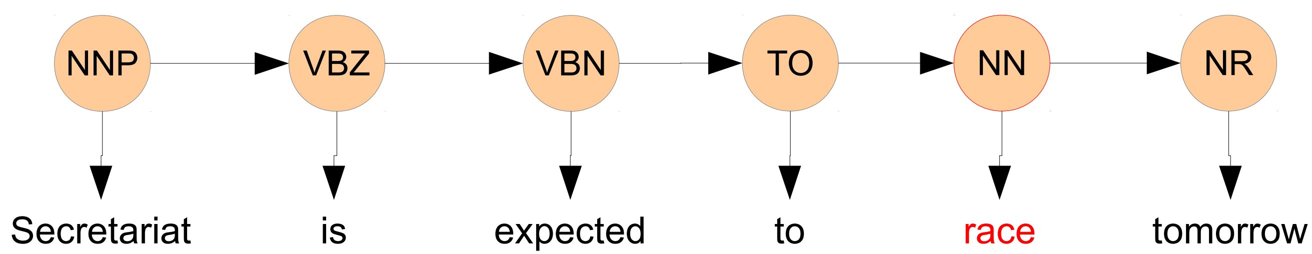
Secretariat$_{[NNP]}$ is$_{[VBZ]}$ expected$_{[VBN]}$ to$_{[TO]}$ race$_{[VB]}$ tomorrow$_{[NR]}$.

People$_{[NNS]}$ inquire$_{[VB]}$ the$_{[DT]}$ reason$_{[NN]}$ for$_{[IN]}$ the$_{[DT]}$ race$_{[NN]}$.

Ambiguity

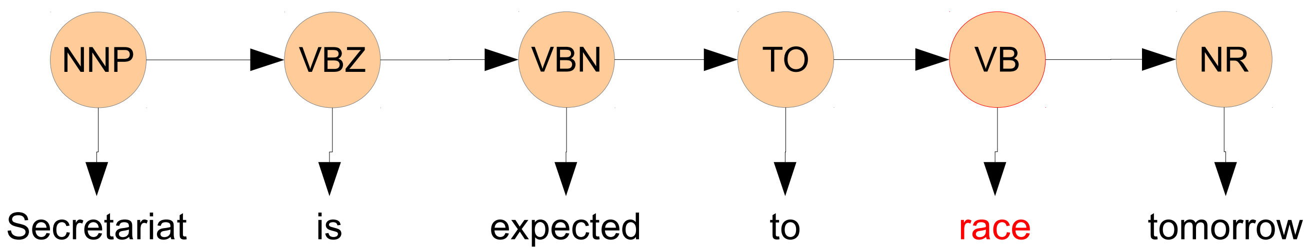
Secretariat$_{\text{[NNP]}}$ is$_{\text{[VBZ]}}$ expected$_{\text{[VBN]}}$ to$_{\text{[TO]}}$ race$_{\text{[VB]}}$ tomorrow$_{\text{[NR]}}$.





Ambiguity

Secretariat$_{[NNP]}$ is$_{[VBZ]}$ expected$_{[VBN]}$ to$_{[TO]}$ race$_{[VB]}$ tomorrow$_{[NR]}$.



$$P(VB|TO) = 0.83$$

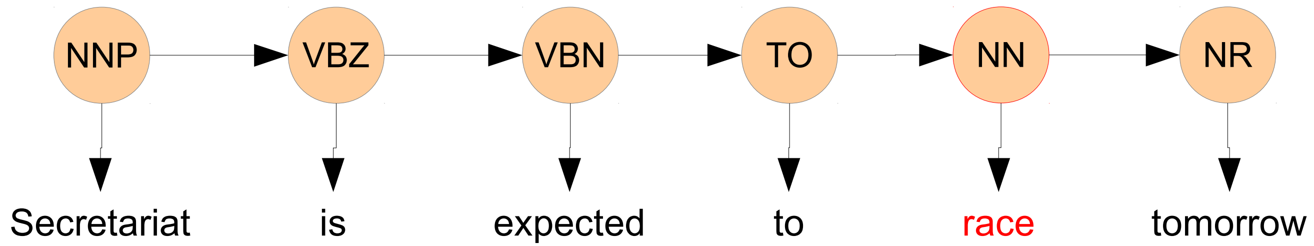
$$P(race|VB) = 0.00012$$

$$P(NR|VB) = 0.0027$$

$$P(VB|TO).P(NR|VB).P(race|VB) = 0.00000027$$

Ambiguity

Secretariat$_{[NNP]}$ is$_{[VBZ]}$ expected$_{[VBN]}$ to$_{[TO]}$ race$_{[VB]}$ tomorrow$_{[NR]}$.



P(NN|TO) = 0.00047

P(race|NN) = 0.00057

P(NR|NN) = 0.0012

P(NN|TO).P(NR|NN).P(race|NN) = 0.00000000032

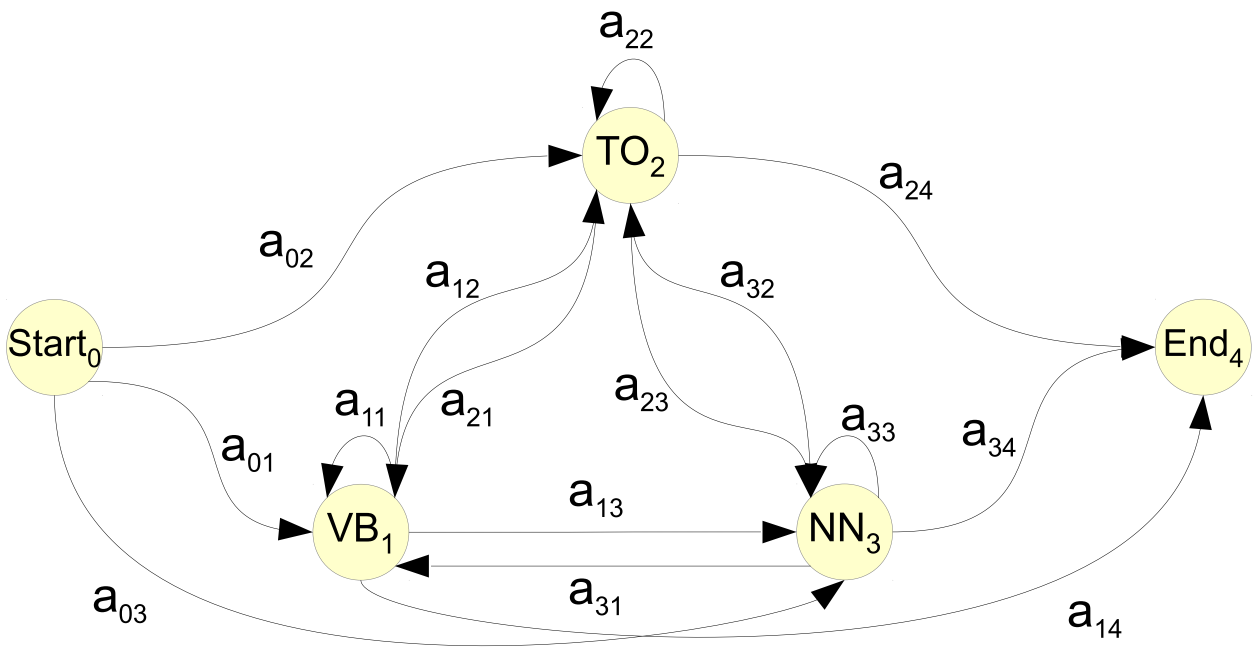
Formalization of Hidden Markov Model (HMM)

- Extension of Finite-State Automata

- Finite-State Automata
  - set of states
  - set of transitions between states

Formalization of Hidden Markov Model (HMM)

- Weighted finite-state automaton
  - Each arc is associated with a probability
  - The probabilities leaving any arc must sum to one



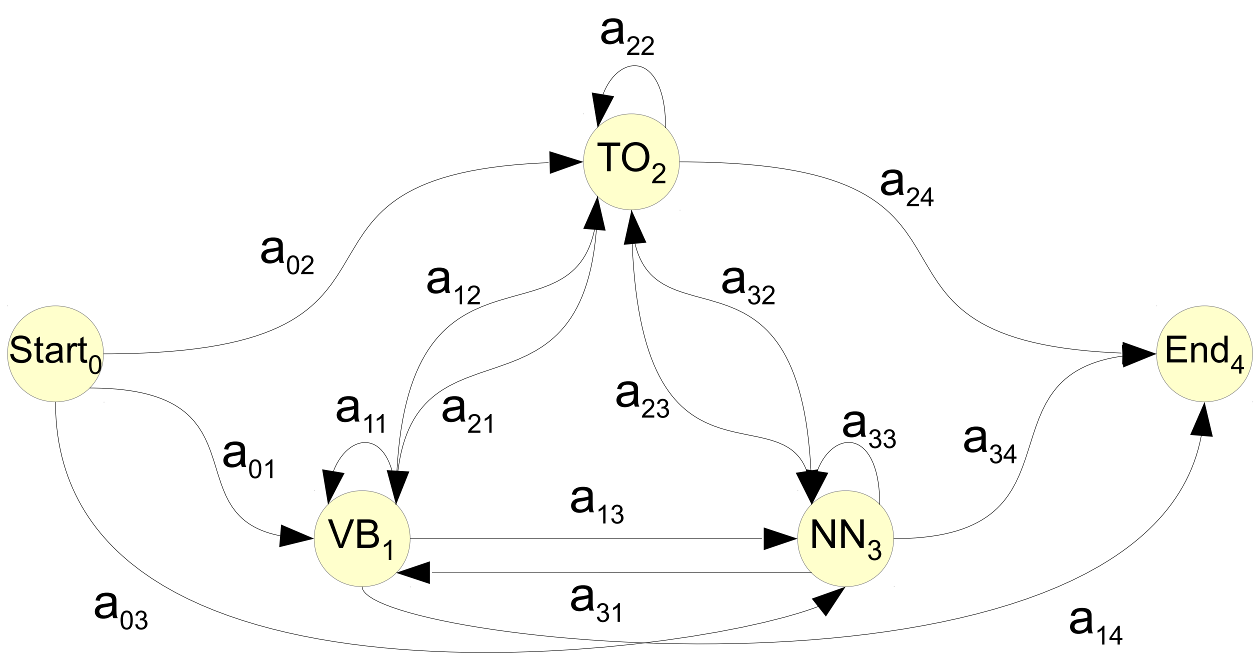
Hidden Markov Model (HMM)

- POS tagging
  - Ambiguous
  - We observe the words, not the POS tags

- HMM
  - Observed events: words
  - Hidden events: POS tags

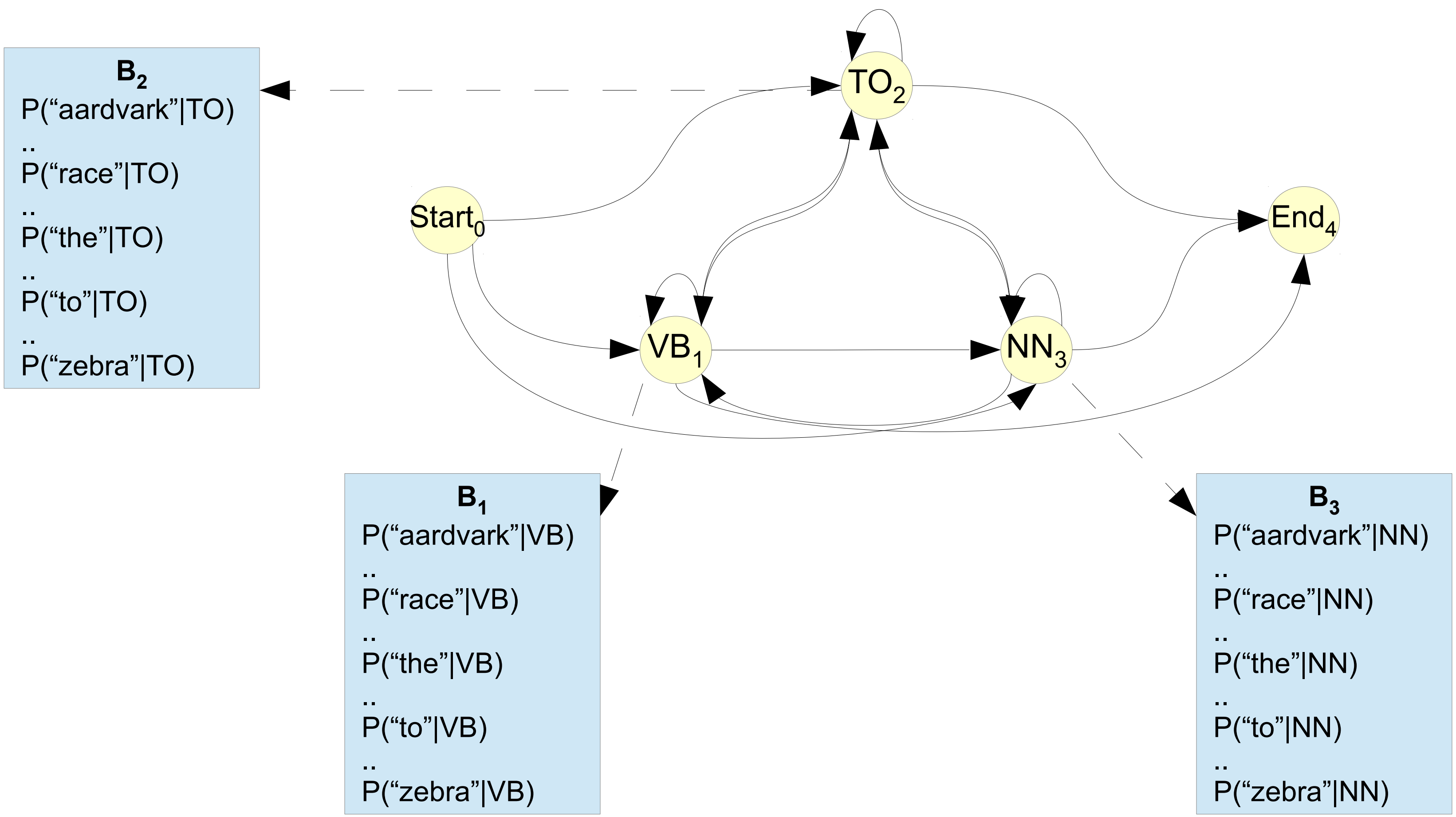
Hidden Markov Model (HMM)

- Transition probabilities: $P(t_i | t_{i-1})$



Hidden Markov Model (HMM)

- Word likelihood probabilities: $P(w_i|t_i)$



Viterbi algorithm

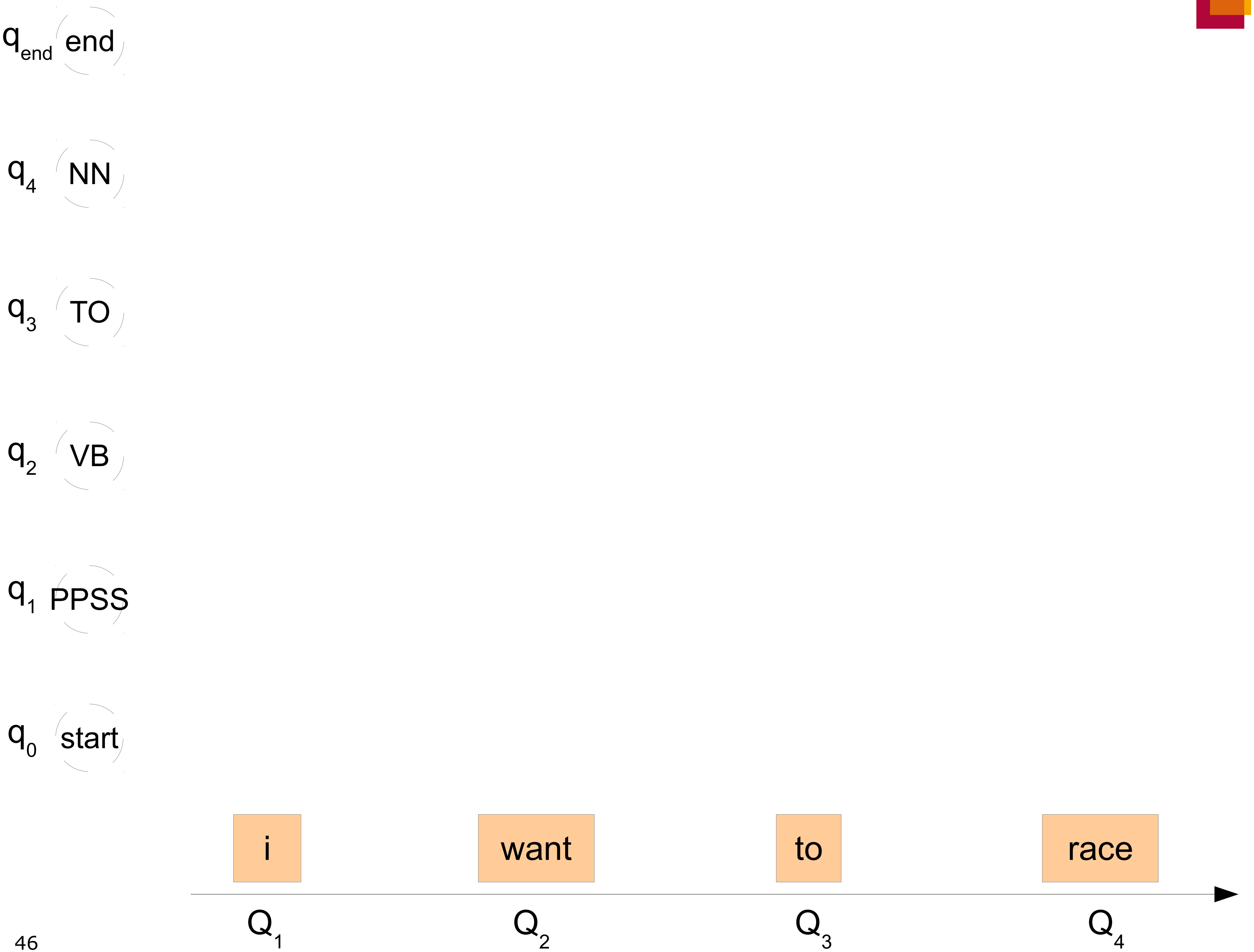
- Decoding algorithm for HMM
  - Determine the best sequence of POS tags

- Probability matrix
  - Columns corresponding to inputs (words)
  - Rows corresponding to possible states (POS tags)

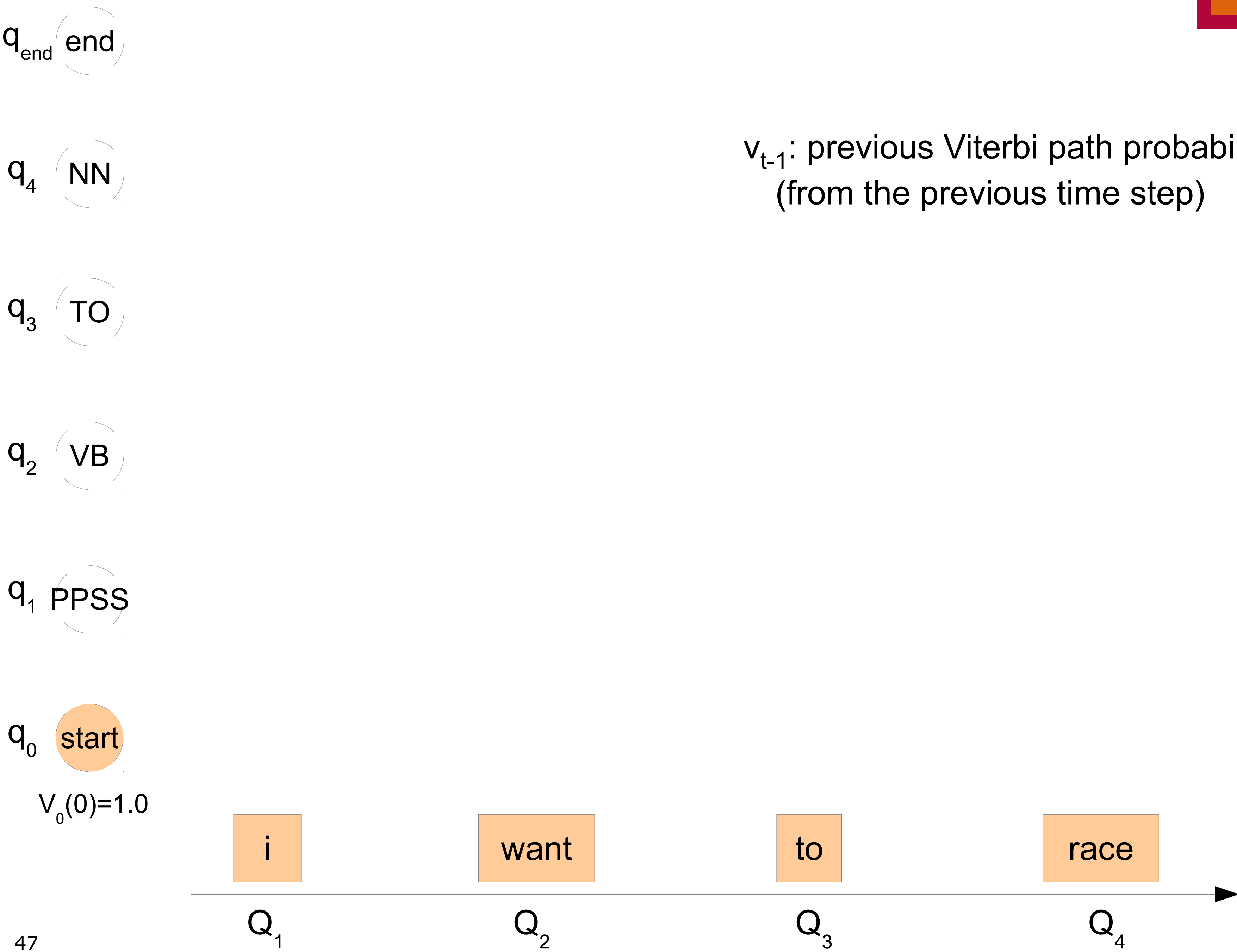
Viterbi algorithm

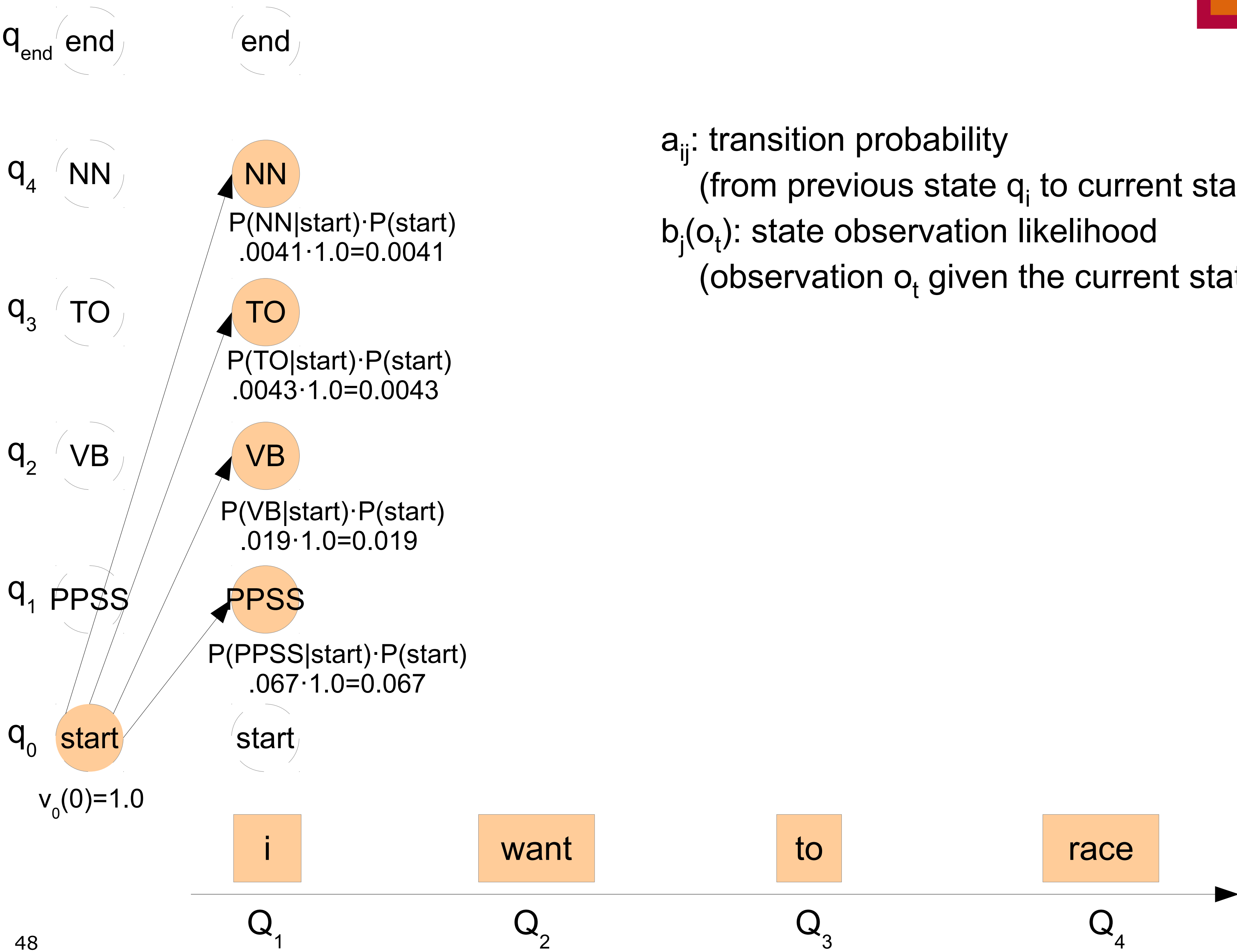
1. Move through the matrix in one pass filling the columns left to right using the transition probabilities and observation probabilities

2. Store the max probability path to each cell (not all paths) using dynamic programming


q_{end} end
q_4 NN
q_3 TO
q_2 VB
q_1 PPSS
q_0 start
i
want
to
race
Q_1
Q_2
Q_3
Q_4

v_{t-1}: previous Viterbi path probability (from the previous time step)

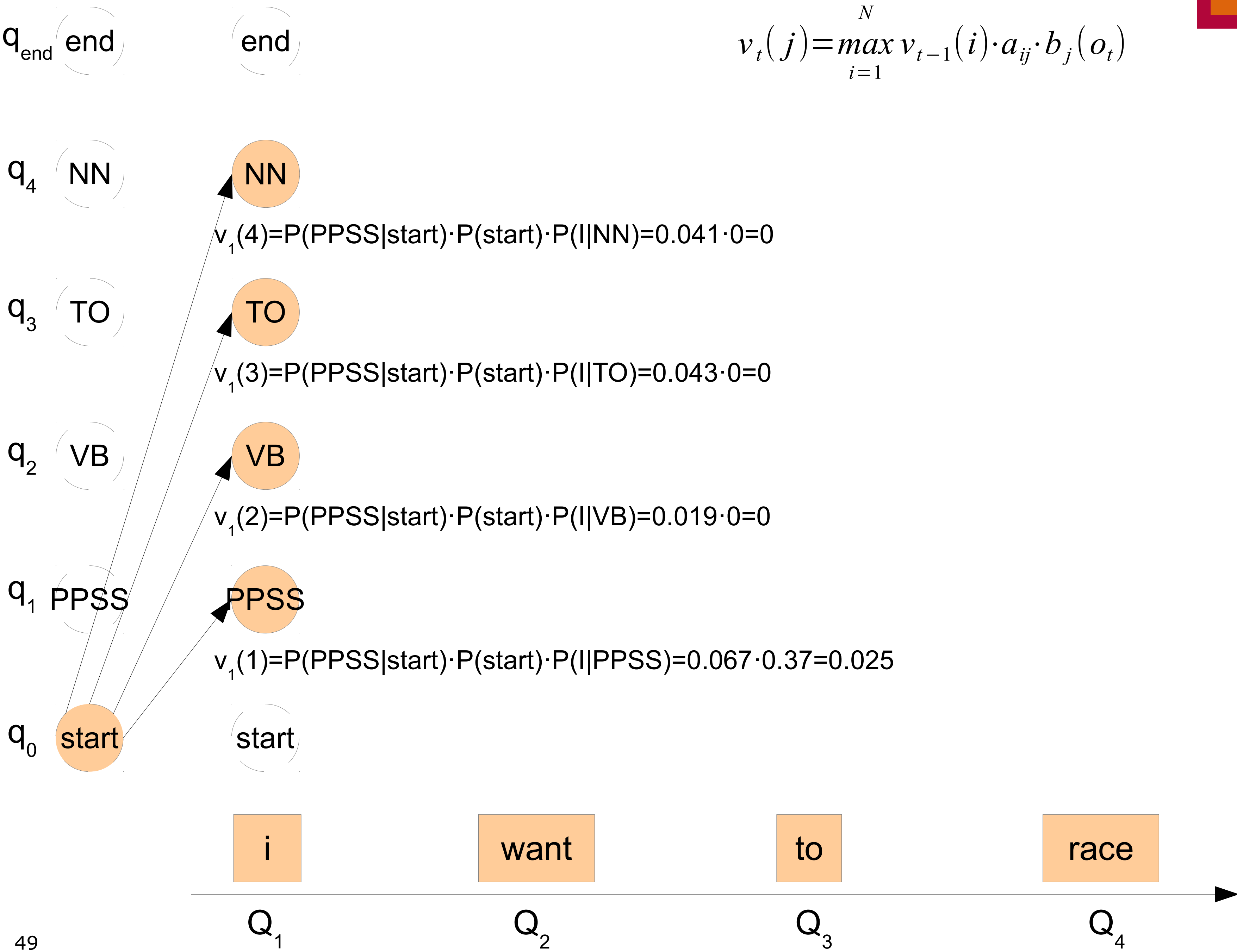


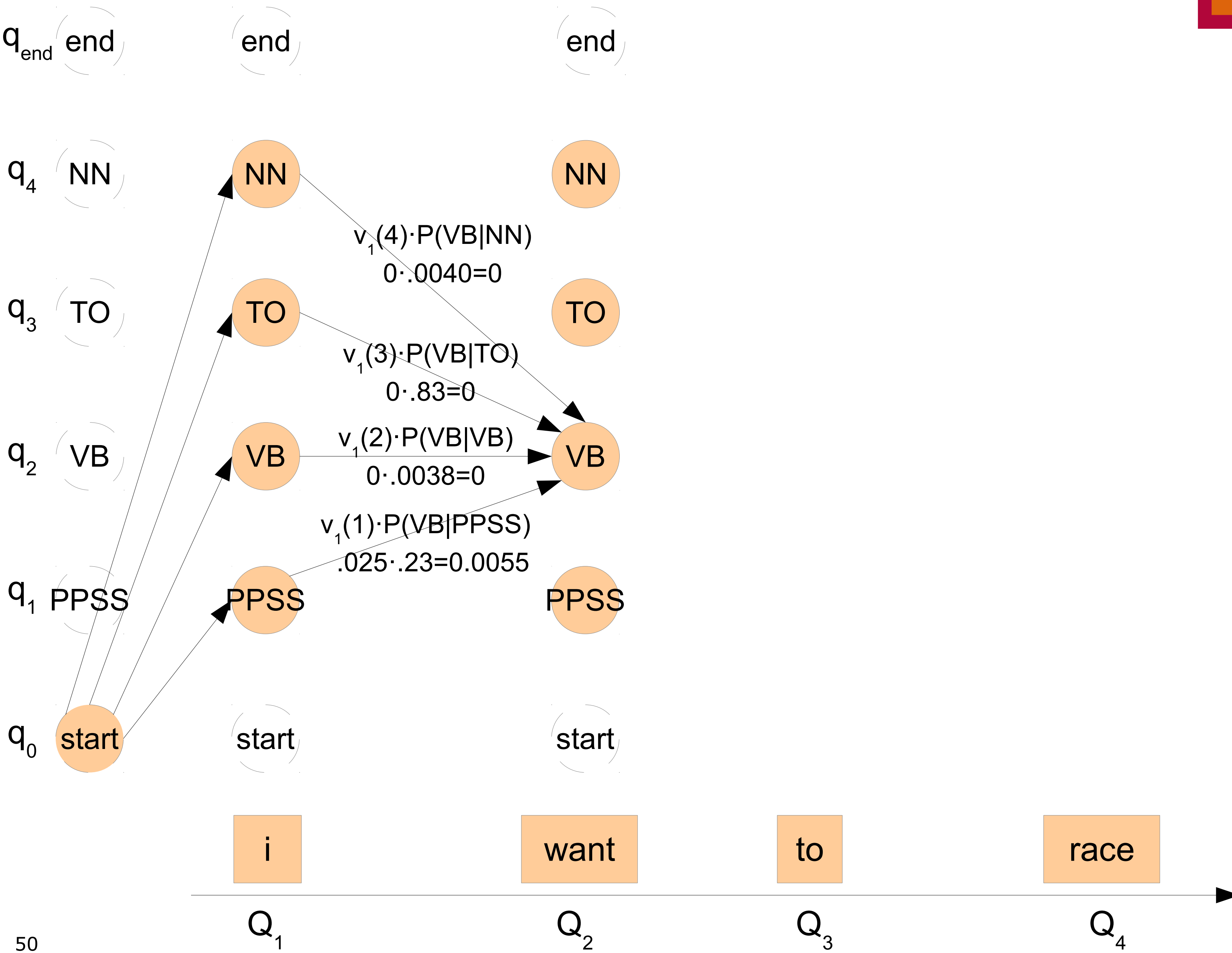


a_{ij}: transition probability
(from previous state q_i to current state q_j)

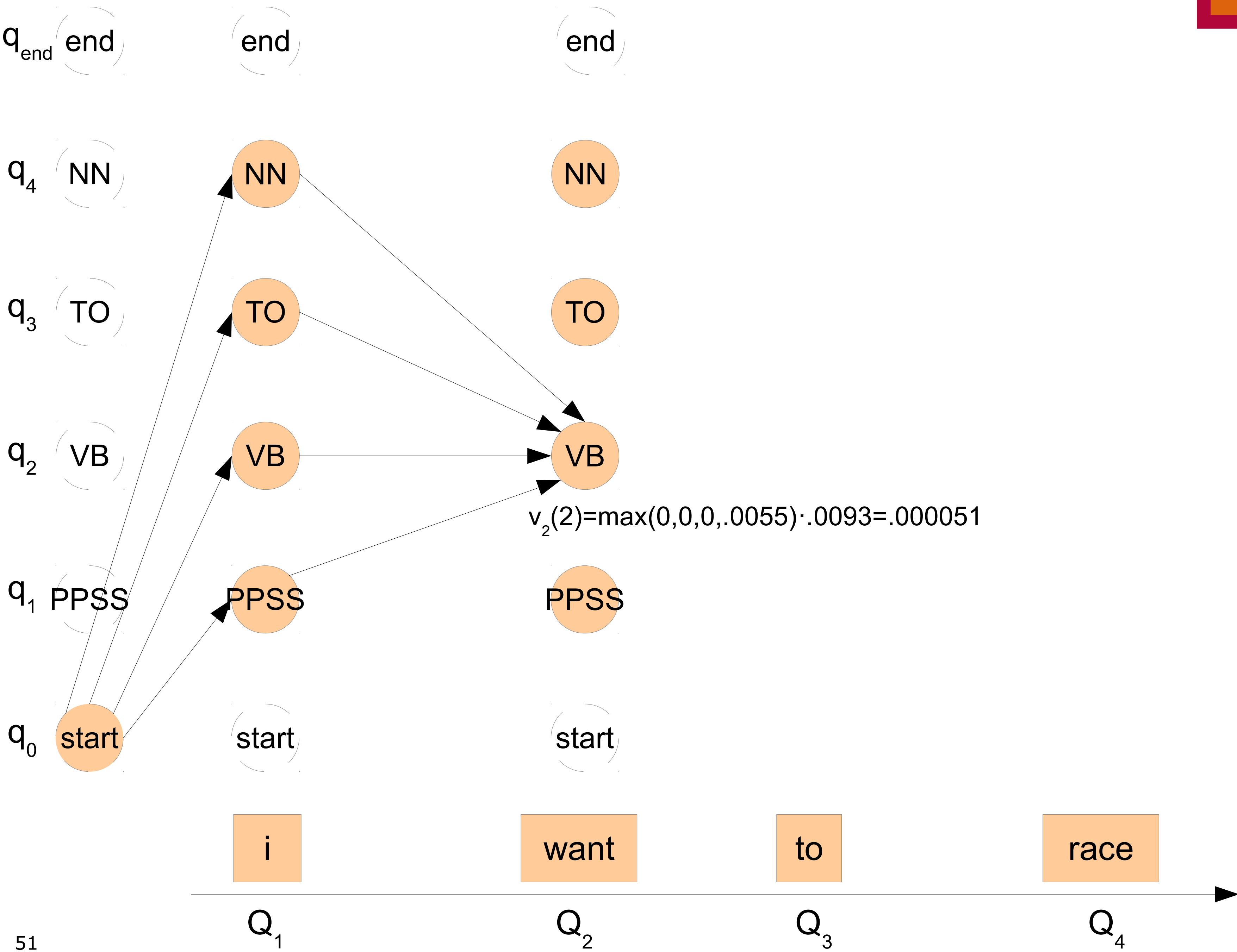
$b_j(o_t)$: state observation likelihood
(observation o_t given the current state j)

$$v_t(j) = \max_{i=1}^{N} v_{t-1}(i) \cdot a_{ij} \cdot b_j(o_t)$$

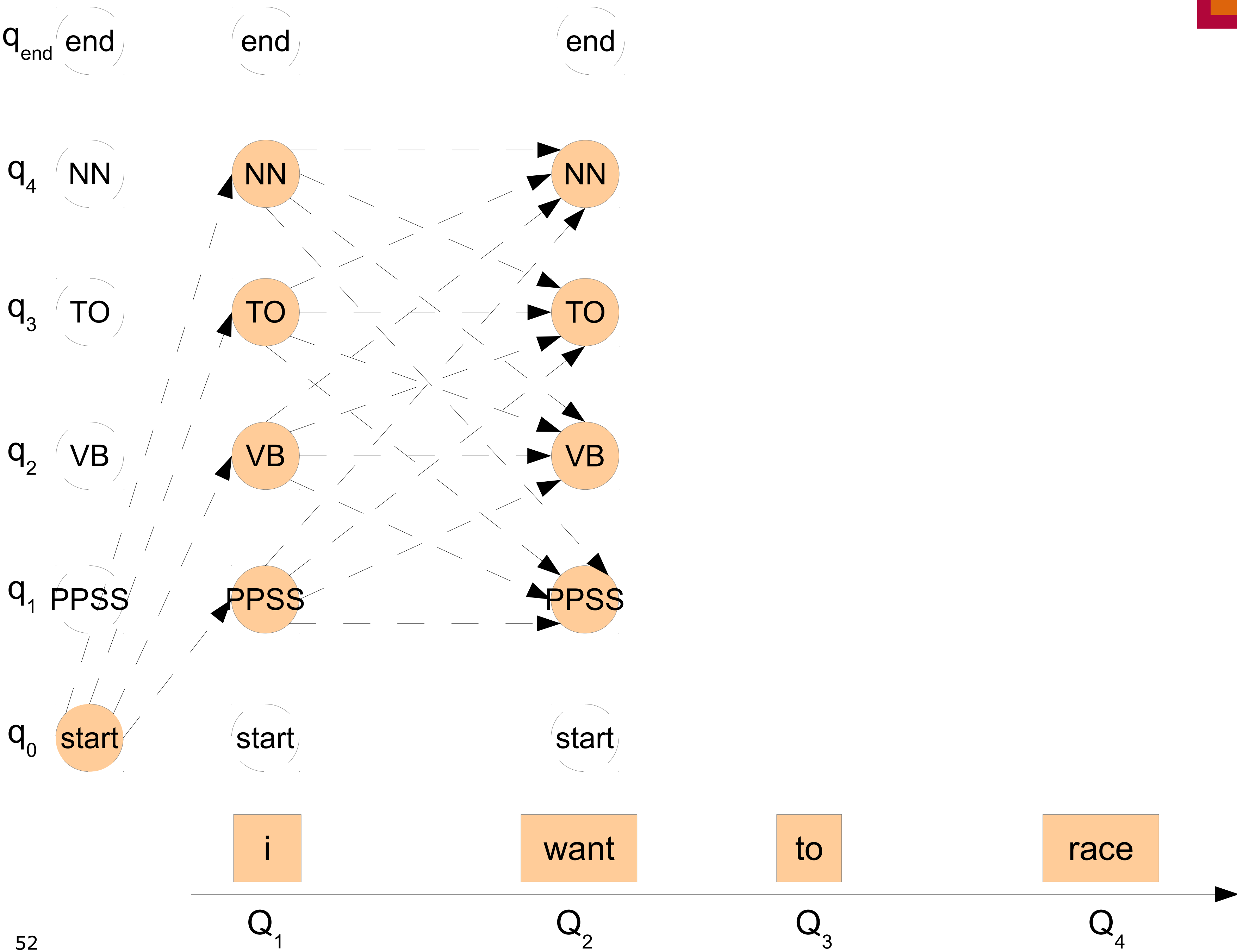




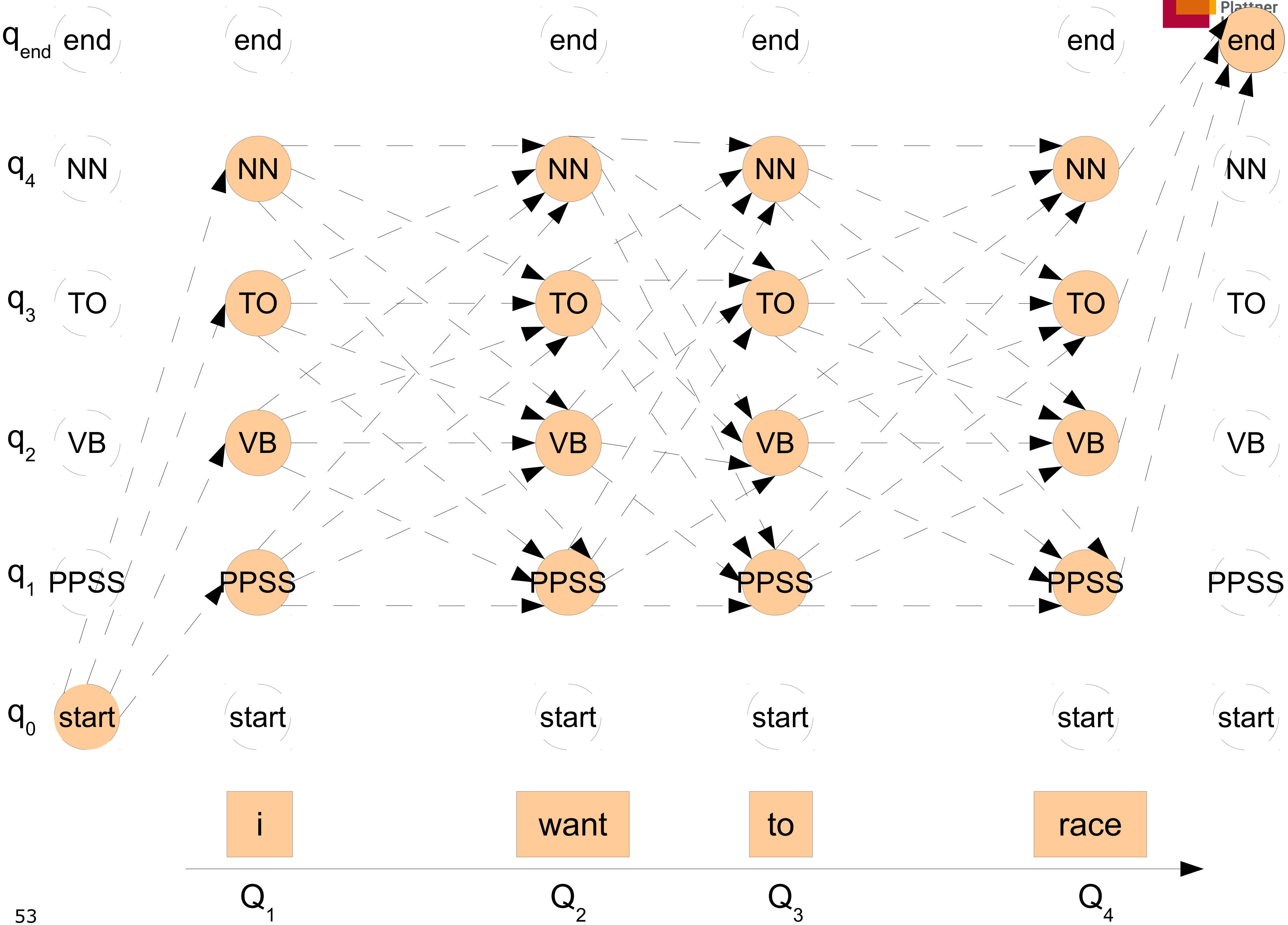
q_end end end end
q_4 NN NN NN
q_3 TO TO TO
q_2 VB VB VB
q_1 PPSS PPSS PPSS
q_0 start start start
v_1(4)·P(VB|NN)
0·.0040=0
v_1(3)·P(VB|TO)
0·.83=0
v_1(2)·P(VB|VB)
0·.0038=0
v_1(1)·P(VB|PPSS)
.025·.23=0.0055
i
want
to
race
Q_1
Q_2
Q_3
Q_4



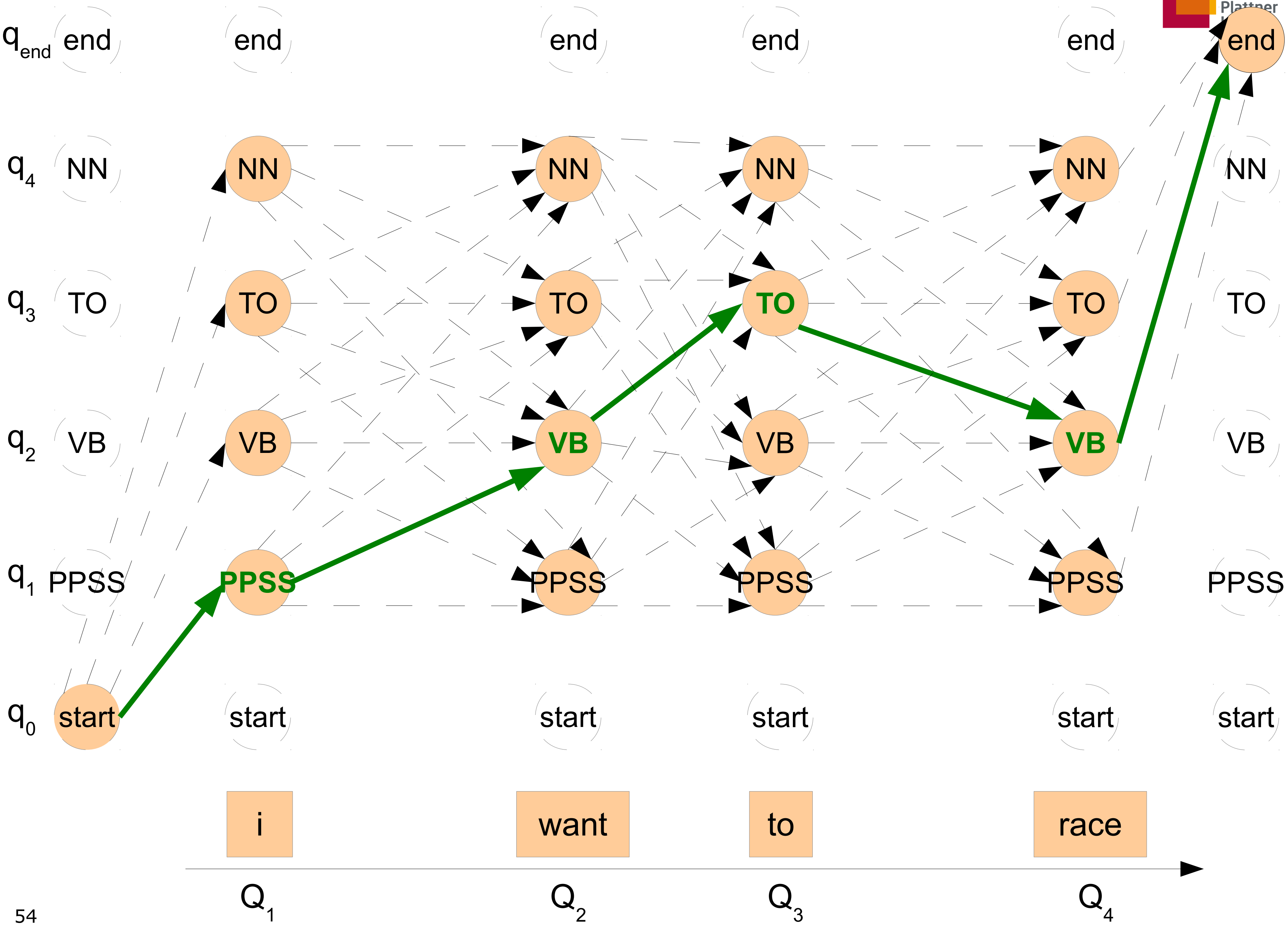
q_{end}
q_4
q_3
q_2
q_1
q_0
end
end
end
NN
NN
NN
TO
TO
TO
VB
VB
VB
PPSS
PPSS
PPSS
start
start
start
$v_2(2)=\max(0,0,0,.0055)\cdot.0093=.000051$
i
want
to
race
Q_1
Q_2
Q_3
Q_4



q_end end end end
q_4 NN NN NN
q_3 TO TO TO
q_2 VB VB VB
q_1 PPSS PPSS PPSS
q_0 start start start
i
want
to
race
Q_1
Q_2
Q_3
Q_4



q_{end}
q_4
q_3
q_2
q_1
q_0
end
NN
TO
VB
PPSS
start
i
want
to
race
Q_1
Q_2
Q_3
Q_4



q_{end}
q_4
q_3
q_2
q_1
q_0
end
NN
TO
VB
PPSS
start
i
want
to
race
Q_1
Q_2
Q_3
Q_4

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

Transformation-Based Tagging

- Also called Brill Tagging
- Uses Transformation-Based Learning (TBL)
  - Rules are automatically induced from training data

Transformation-Based Tagging

- Brill's Tagger
  1. Assign every word with the most likely tag:
     - Secretariat/NNP is/VBZ expected/VBN to/TO race/NN tomorrow/NN
     - The/DT race/NN for/IN outer/JJ space/NN
  2. Apply transformation rules:
     - e.g., change „NN“ to „VB“ when previous tag is „TO“
       - Secretariat/NNP is/VBZ expected/VBN to/TO race/VB tomorrow/NN

Transformation-Based Tagging

- TBL learning process
  - Based on a set of templates (abstracted transformations)
    - „the word two before (after) is tagged **z**"
    - „the preceding word is tagged **z** and the following word is tagged **w**"
    - etc.

Outline

- Part-of-Speech tags
- Part-of-Speech tagging
  - Rule-Based Tagging
  - HMM Tagging
  - Transformation-Based Tagging
- Evaluation

Evaluation

- Corpus
  - Training and test, and optionally also development set
  - Training (cross-validation) and test set

- Evaluation
  - Comparison of gold standard (GS) and predicted tags
  - Evaluation in terms of Precision, Recall and F-Measure

Precision and Recall

- Precision:
  - Amount of labeled items which are correct

$$Precision = \frac{tp}{tp + fp}$$

- Recall:
  - Amount of correct items which have been labeled

$$Recall = \frac{tp}{tp + fn}$$

F-Measure

- There is a strong anti-correlation between precision and recall
- Having a trade off between these two metrics
- Using F-measure to consider both metrics together
- F -measure is a weighted harmonic mean of precision and recall

$$F = \frac{(\beta^2 + 1) P R}{\beta^2 P + R}$$

Error Analysis

- Confusion matrix or contingency table
  - Percentage of overall tagging error

	IN	JJ	NN	NNP	RB	VBD	VBN
IN	-	.2			.7		
JJ	.2	-	3.3	2.1	1.7	.2	2.7
NN		8.7	-				.2
NNP	.2	3.3	4.1	-	.2		
RB	2.2	2.0	.5		-		
VBD		.3	.5			-	4.4
VBN		2.8				2.6	

Further reading and tools

- Book Jurafski & Martin
  - Chapter 5

- Tools
  - Stanford POS parser
  - OpenNLP
  - TreeTagger
  - and many others...

Project update

- Integration of POS tagging
  - Presentation in the next lecture (May 23rd, 2016)