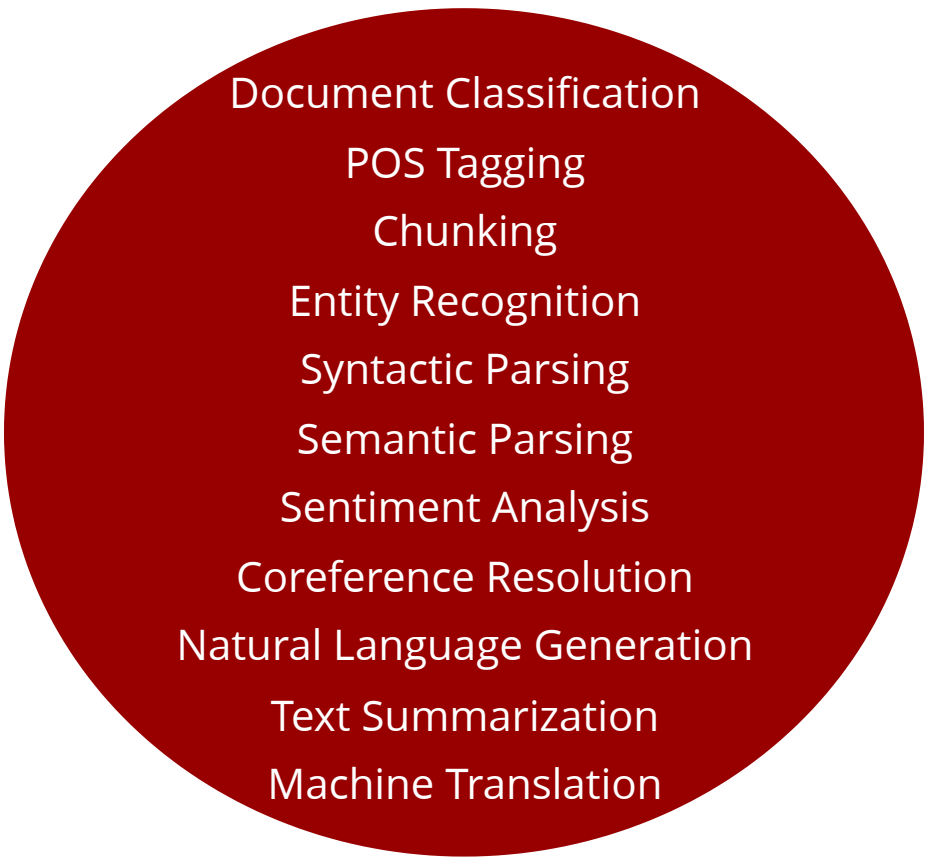
# 50.040
# Natural Language Processing

Lu, Wei

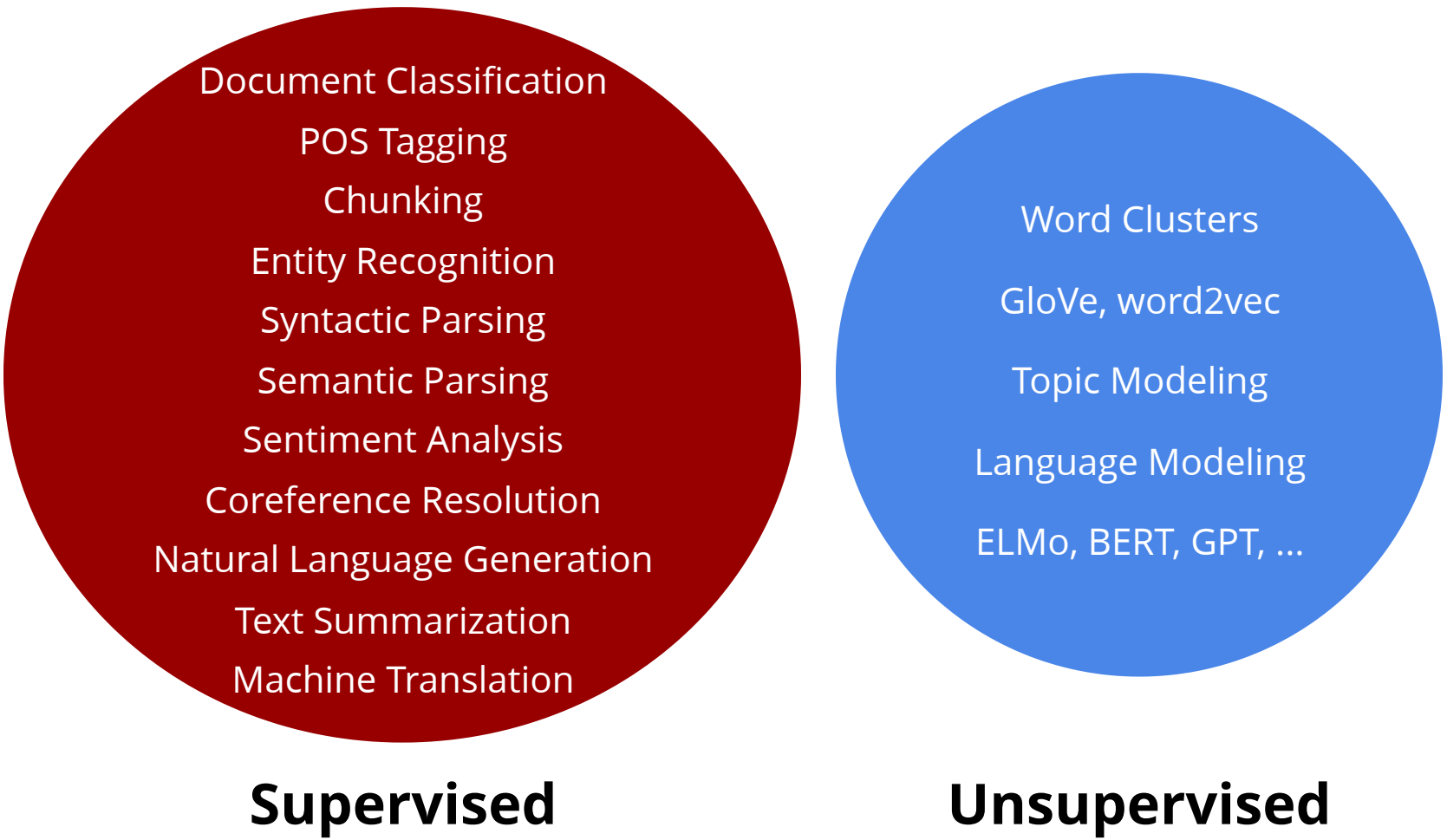
SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

# Tasks in NLP

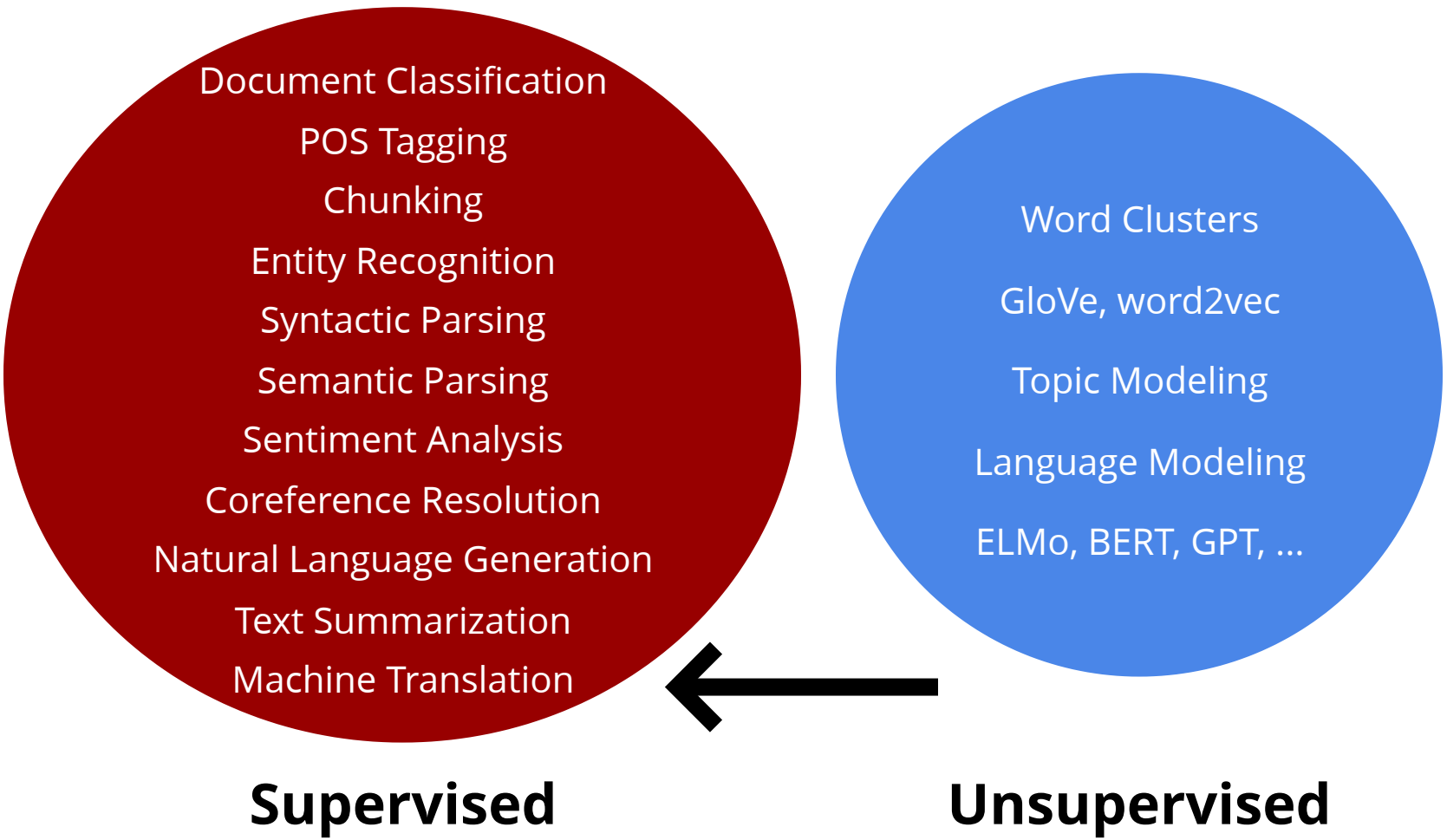
# Tasks in NLP



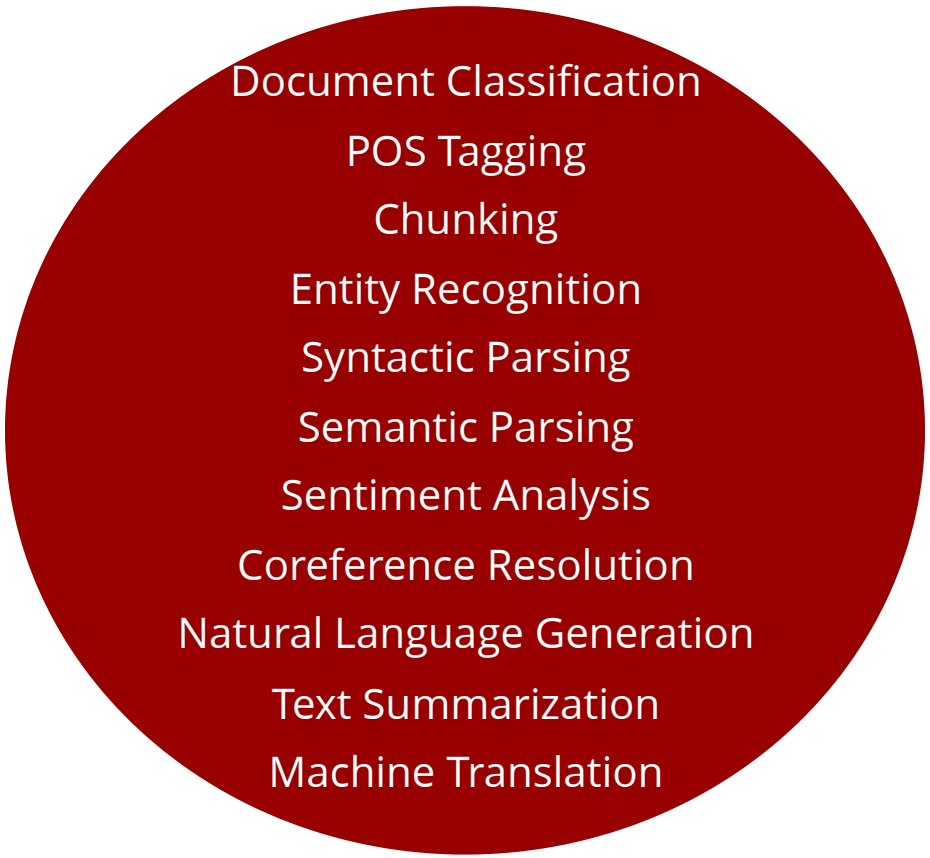

# Tasks in NLP



# Tasks in NLP

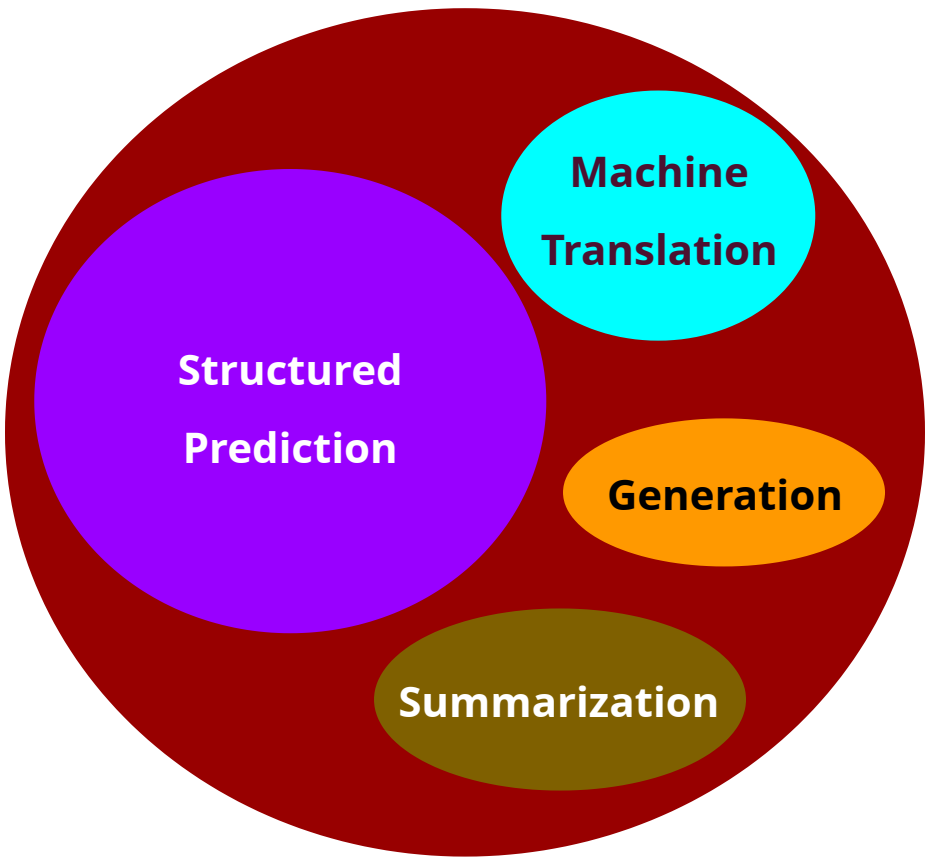


# Tasks in NLP




**Supervised**

# Tasks in NLP



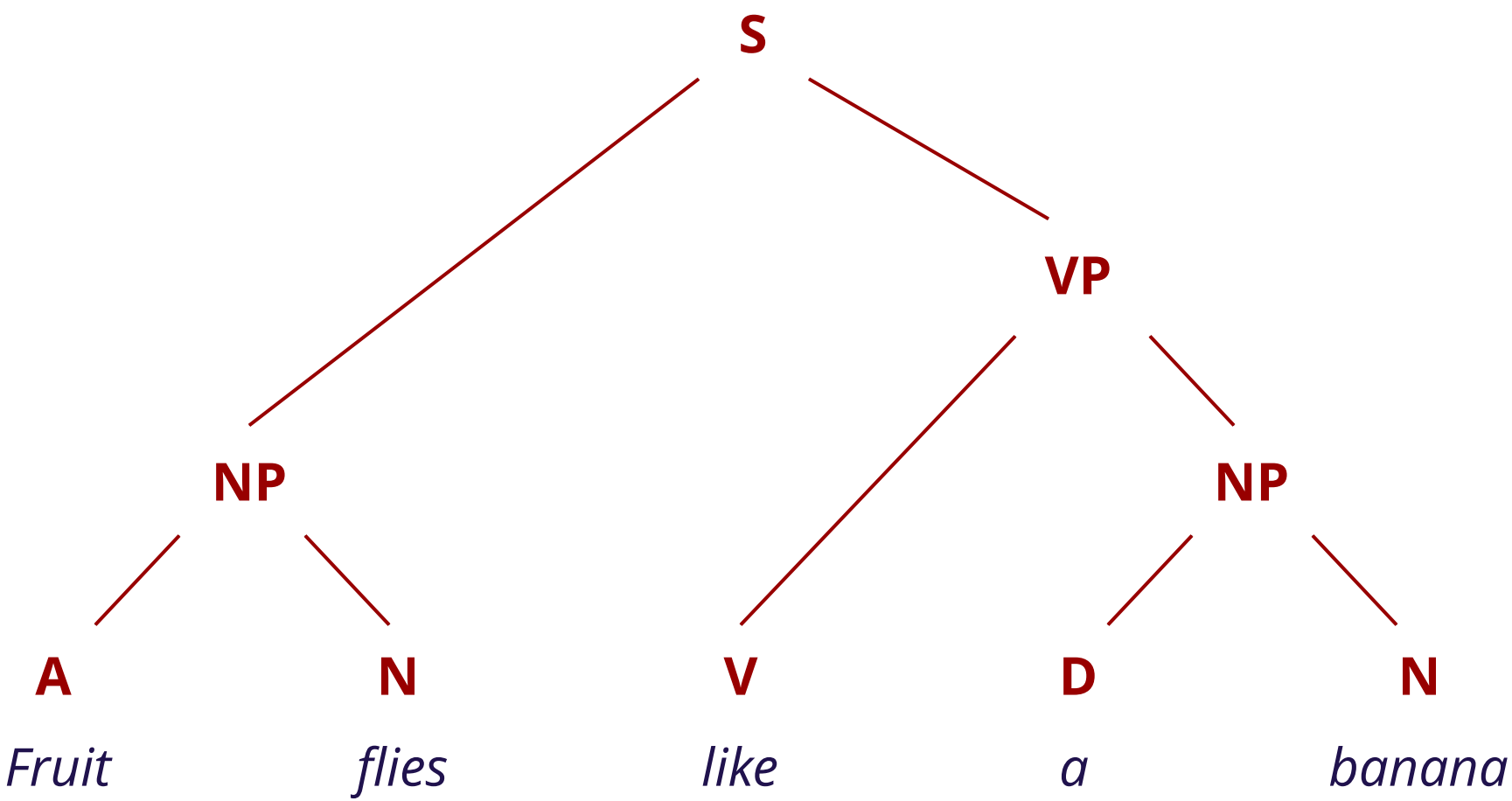

**Supervised**

# Part-of-Speech Tagging

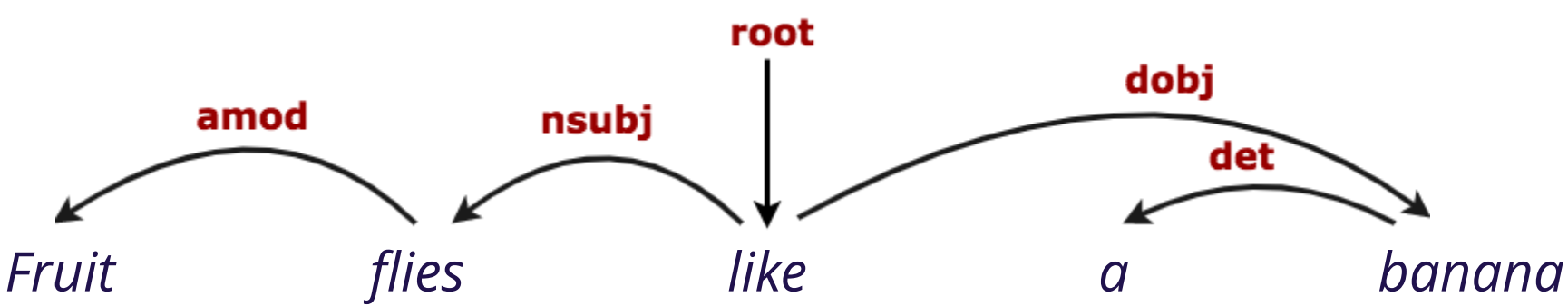
| A | N | V | D | N |
|---|---|---|---|---|
| *Fruit* | *flies* | *like* | *a* | *banana* |

# Noun-Phrase Chunking

# Constituency Parsing



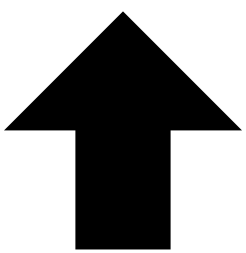
# Dependency Parsing



# Semantic Parsing

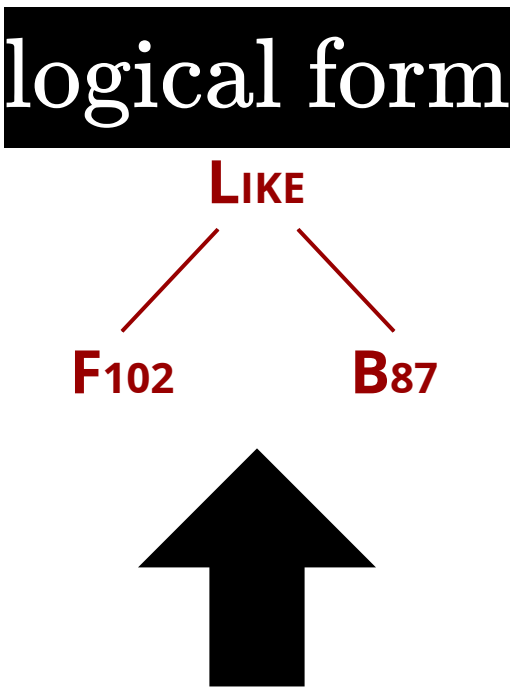
logical form

LIKE(F102, B87)



*Fruit* *flies* *like* *a* *banana*

# Semantic Parsing



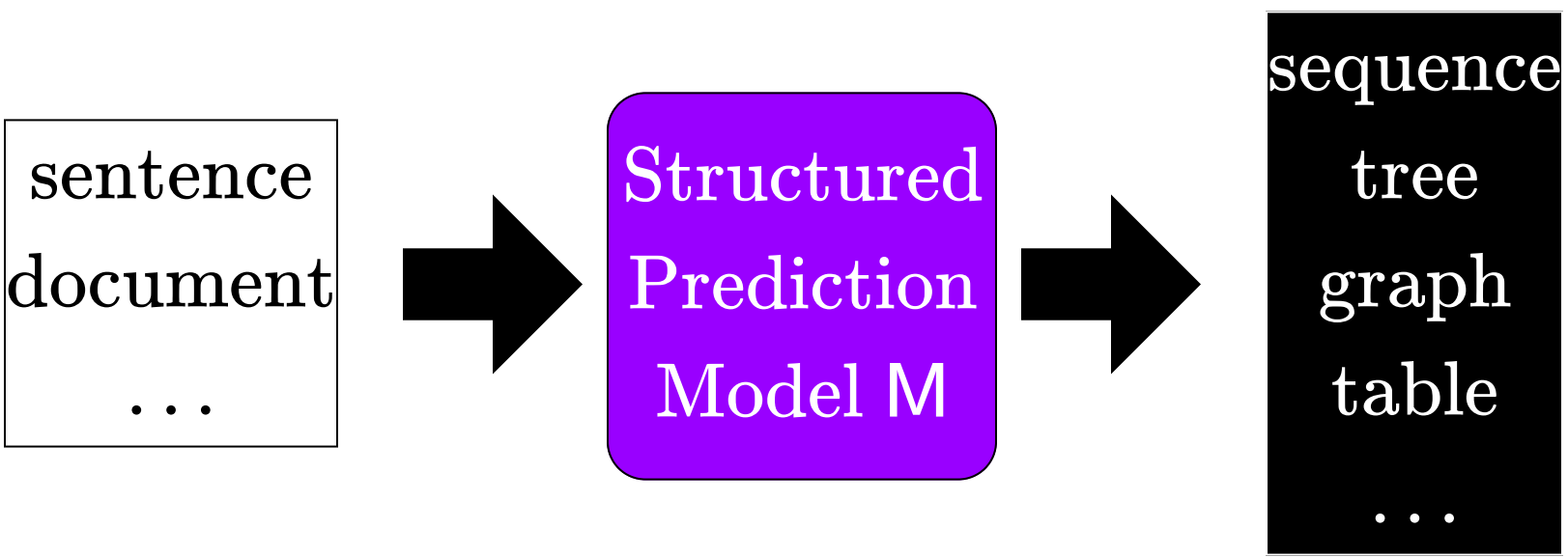

*Fruit* *flies* *like* *a* *banana*

# Sentiment Analysis

| | | | | |
|---|---|---|---|---|
| **( neutral )** | | | **( positive )** | |
| *Fruit* | *flies* | *like* | *a* | *banana* |

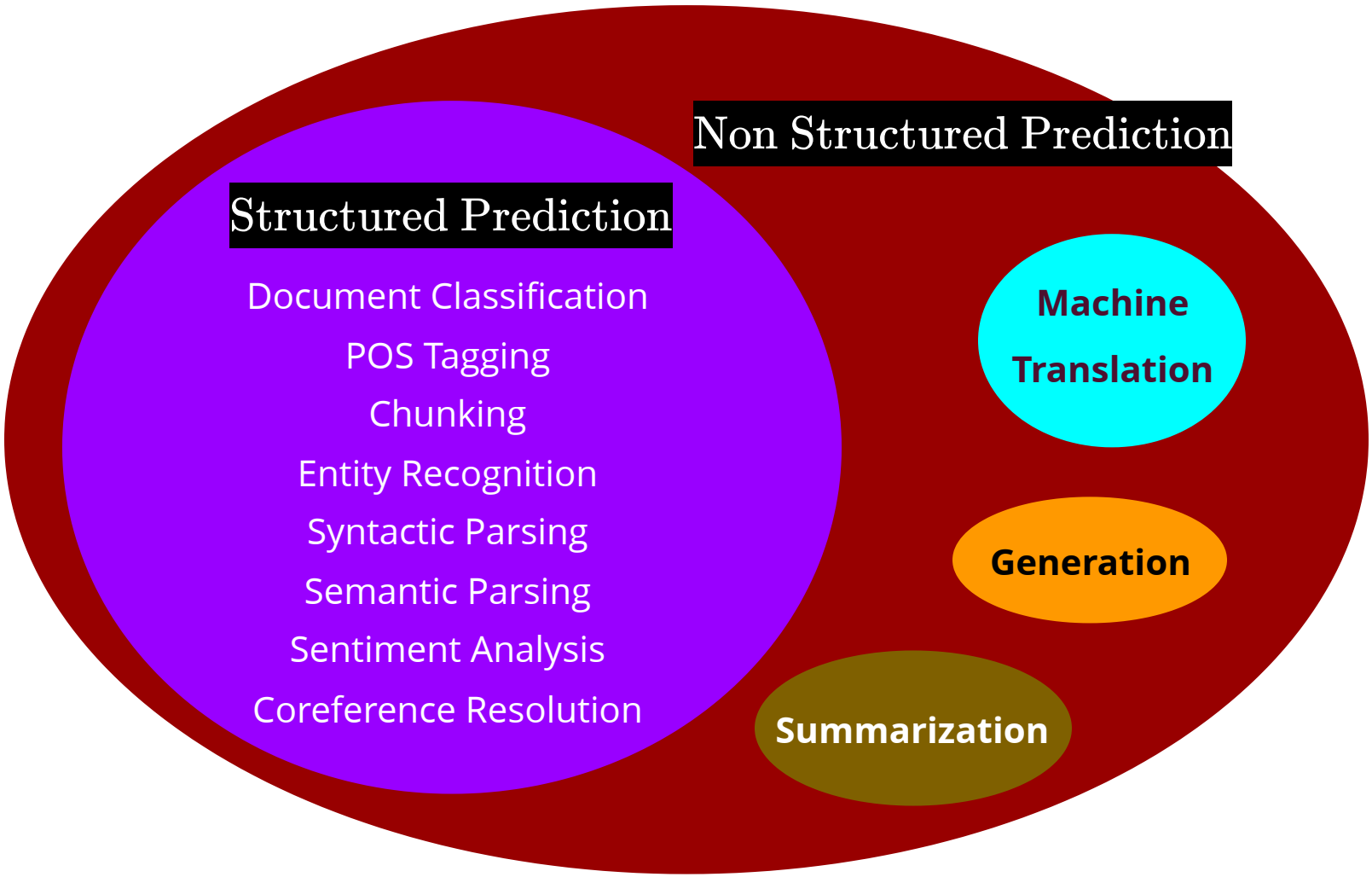
# Structured Prediction

These supervised problems are "structured prediction" problems




Structured Space $\mathcal{S}_i$ | M : $\mathcal{S}_i \rightarrow \mathcal{S}_o$ | Structured Space $\mathcal{S}_o$

# Tasks in NLP

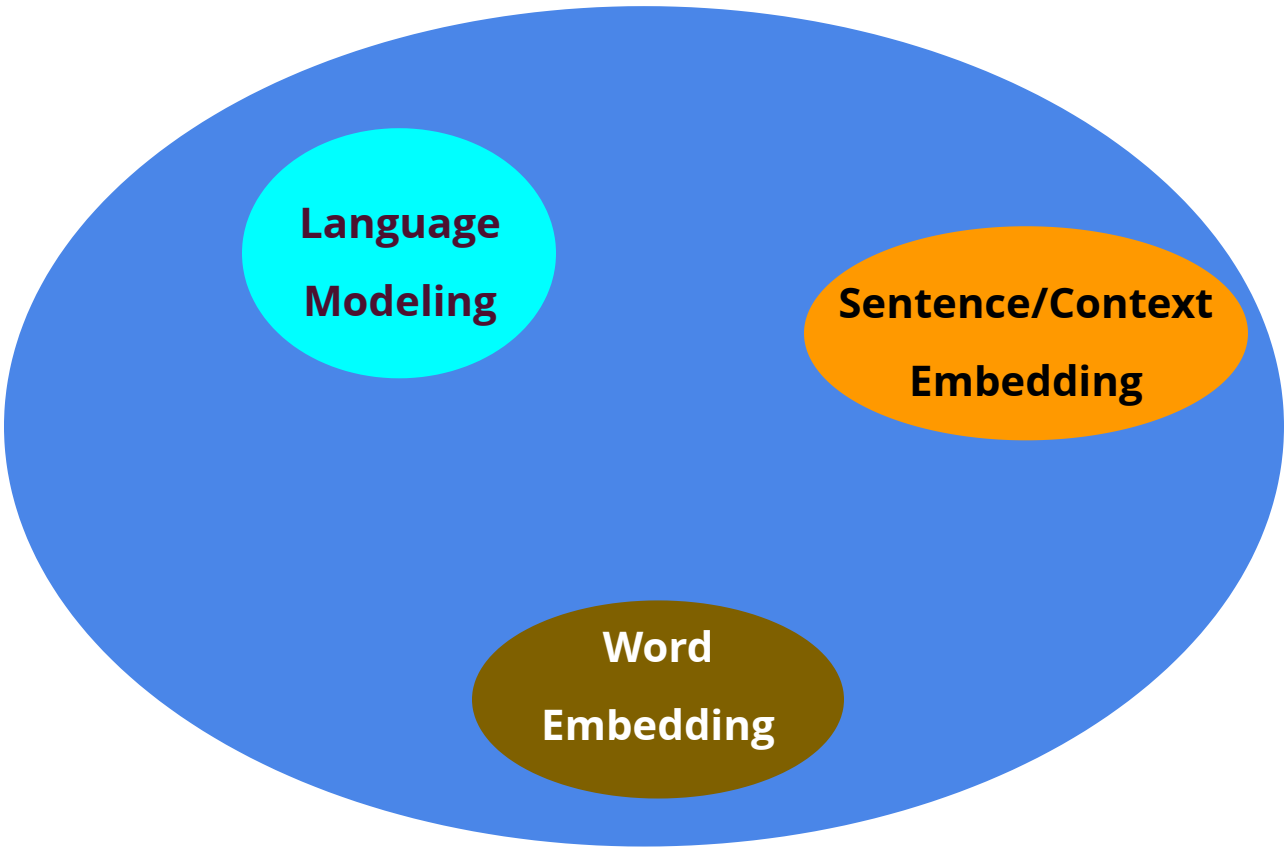



**Supervised**

# Tasks in NLP

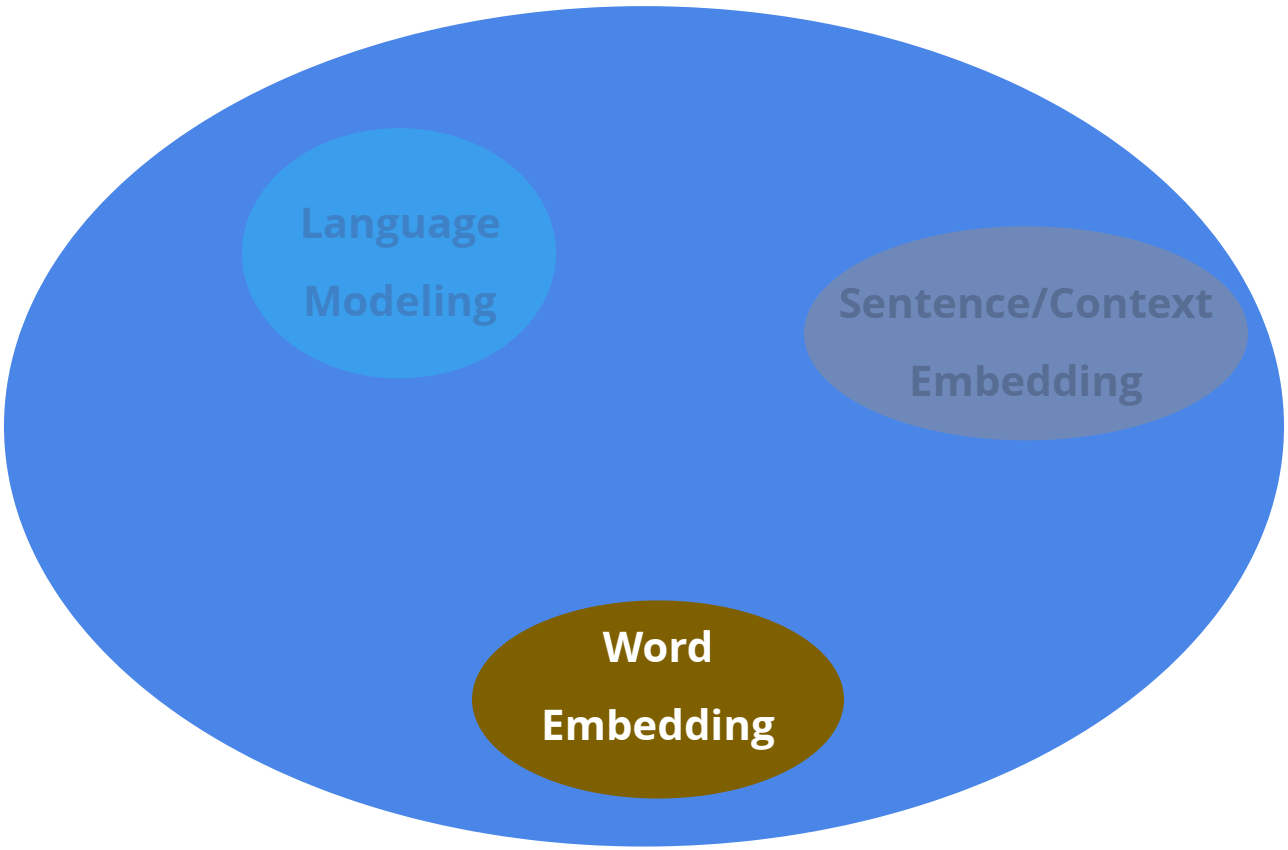

**Unsupervised**

# Tasks in NLP




**Unsupervised**

# Tasks in NLP



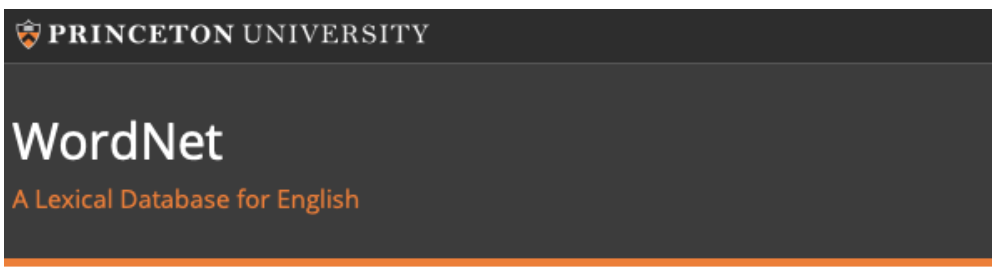

**Unsupervised**

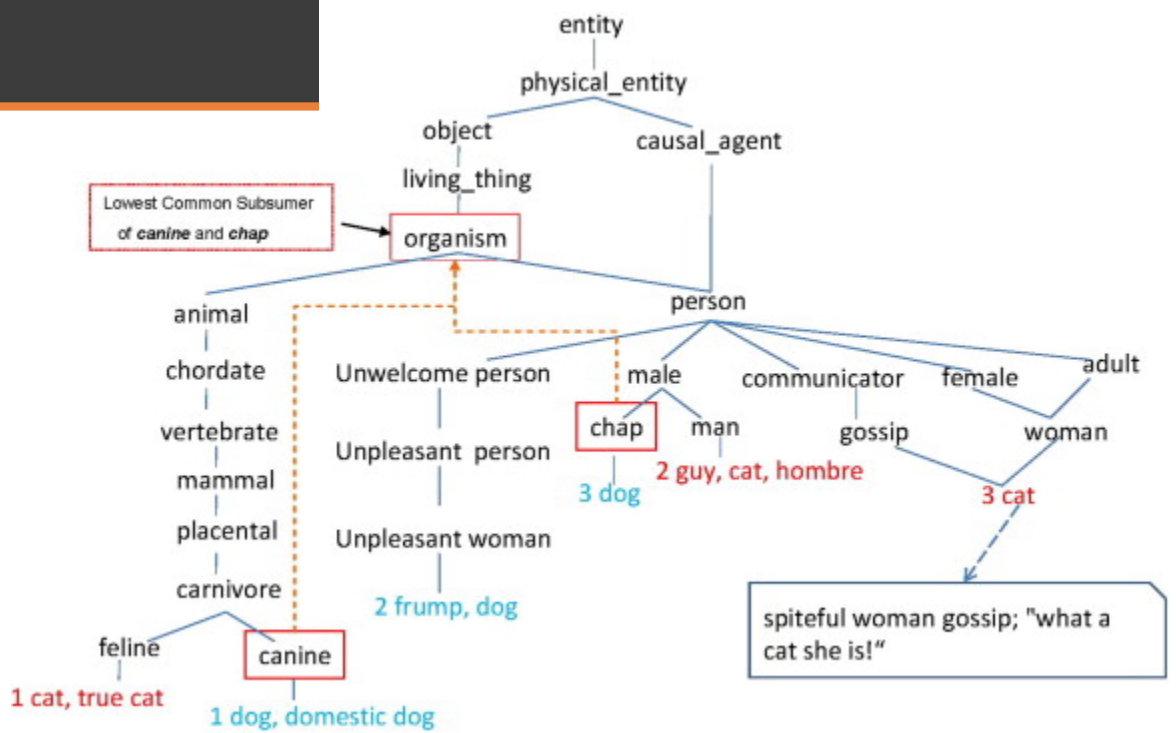
# Fundamental Question

How do we capture the "meaning" of a word?

# WordNet

https://wordnet.princeton.edu




Wei, Tingting, et al. "A semantic approach for text clustering using WordNet and lexical chains." *Expert Systems with Applications* 42.4 (2015): 2264-2275.

# WordNet

https://wordnet.princeton.edu

PRINCETON UNIVERSITY

WordNet

A Lexical Database for English

Limitations:
Only for English
Incomplete
Subjective
Expensive to Maintain


Wei, Tingting, et al. "A semantic approach for text clustering using WordNet and lexical chains." *Expert Systems with Applications* 42.4 (2015): 2264-2275.

# WordNet

https://wordnet.princeton.edu




Limitations:
Only for English
Incomplete
Subjective
Expensive to Maintain

Possible to capture word semantics *automatically*?

Wei, Tingting, et al. "A semantic approach for text clustering using WordNet and lexical chains." *Expert Systems with Applications* 42.4 (2015): 2264-2275.

# How to Represent Words

## Traditionally ...

$$\texttt{cat} \quad \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{pmatrix} \qquad \texttt{eat} \quad \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{pmatrix} \qquad \texttt{dog} \quad \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 0 \end{pmatrix}$$

# How to Represent Words

## Ideally ...

cat

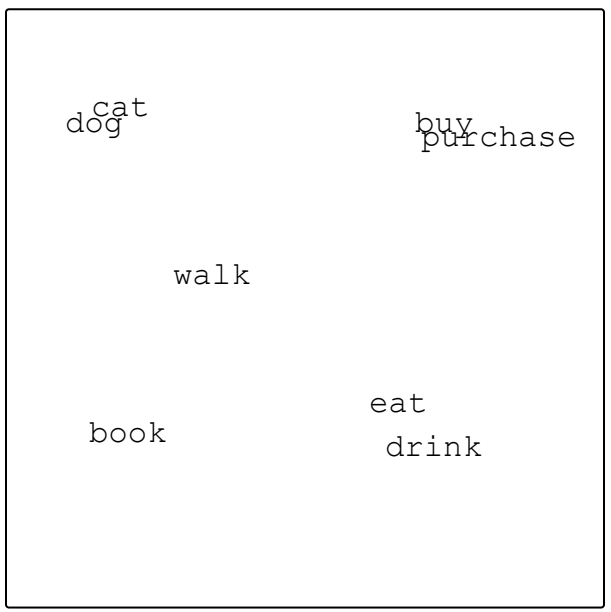
$$\begin{pmatrix} 0 \\ 1 \\ 0.232 \\ -0.903 \\ 0.213 \\ 0.239 \\ -0.679 \\ 0.923 \\ 0 \\ 0 \end{pmatrix}$$

eat

$$\begin{pmatrix} 0 \\ 0 \\ -0.932 \\ -0.903 \\ 0.213 \\ 0.590 \\ 0.609 \\ 0.388 \\ 0 \\ 0 \end{pmatrix}$$

dog

$$\begin{pmatrix} 0 \\ 0 \\ 0.190 \\ -0.887 \\ 0.193 \\ 0.257 \\ -0.708 \\ 0.906 \\ 0 \\ 0 \end{pmatrix}$$



# Distributional Semantics

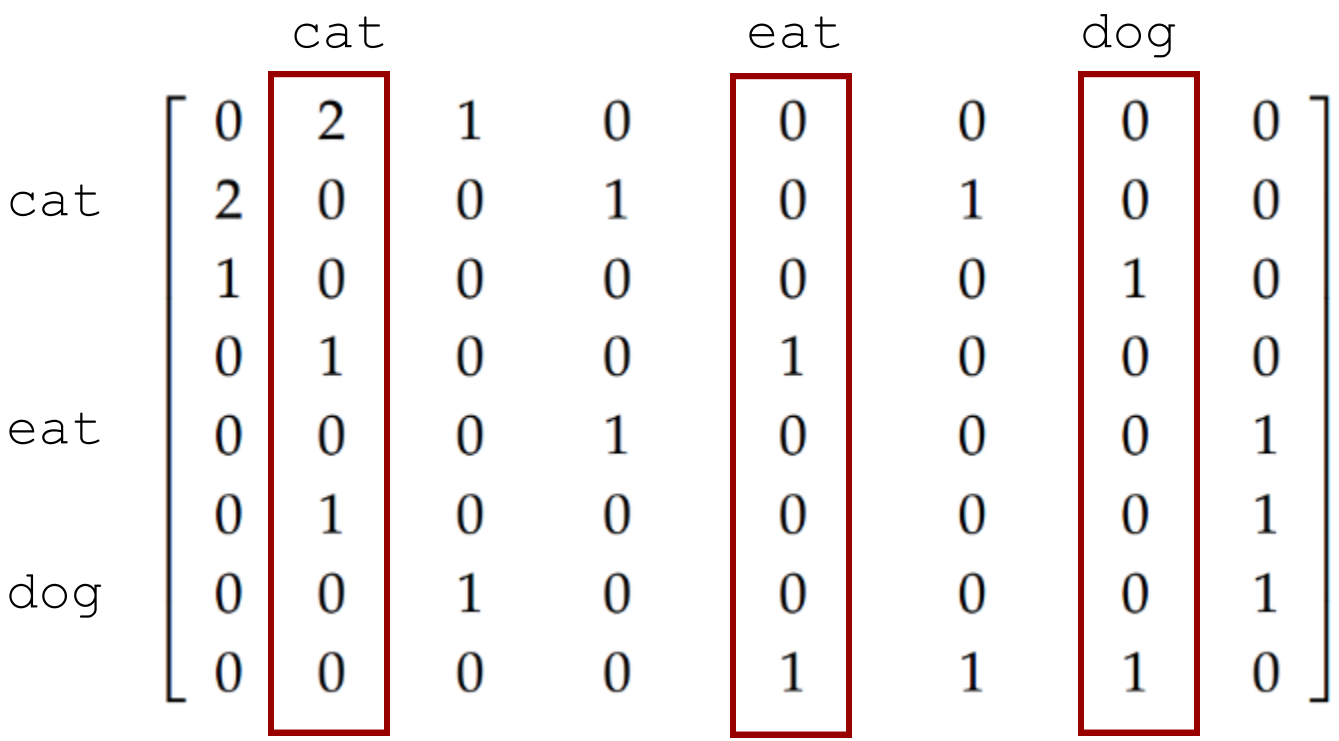
*You shall know a word by the company it keeps*

- Firth, J. R. 1957

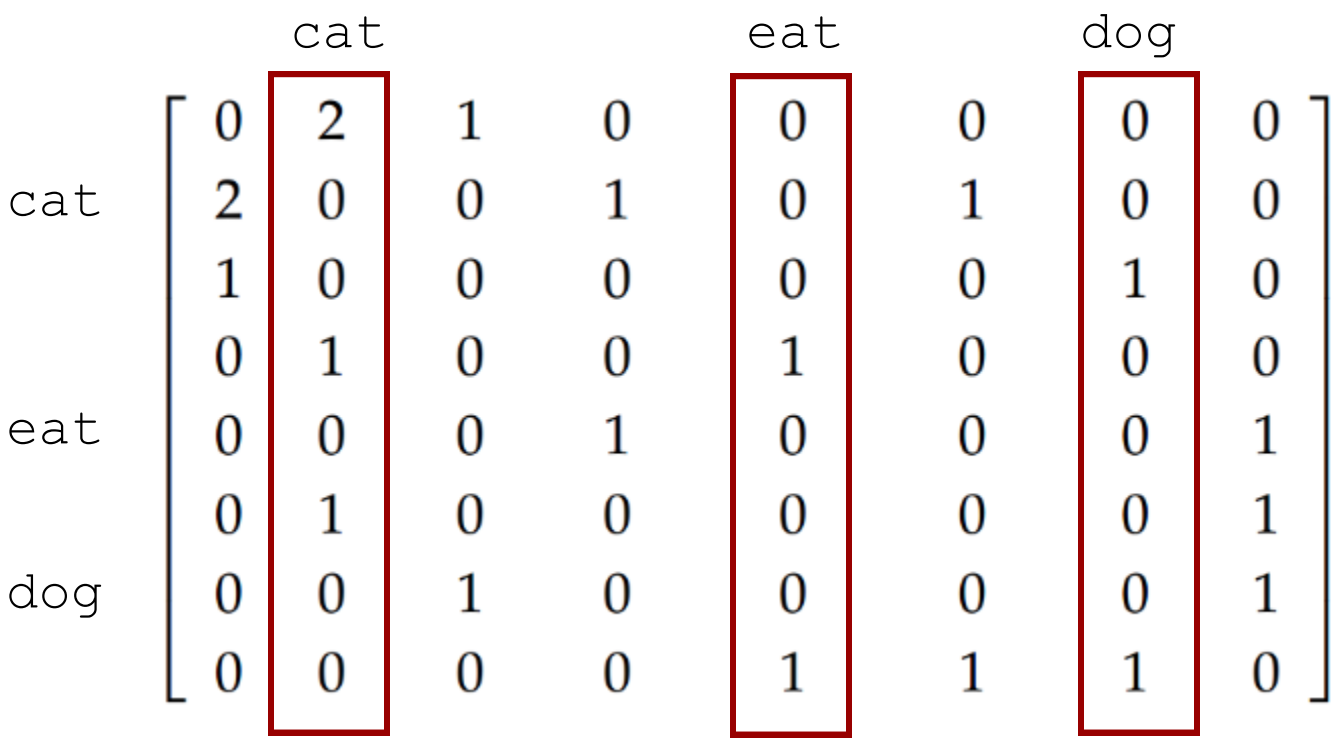
# Co-Occurence Matrix

| | cat | | | eat | | | dog | |
|---|---|---|---|---|---|---|---|---|
| | 0 | 2 | 1 | 0 | 0 | 0 | 0 | 0 |
| cat | 2 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| eat | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| dog | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |

# Co-Occurence Matrix

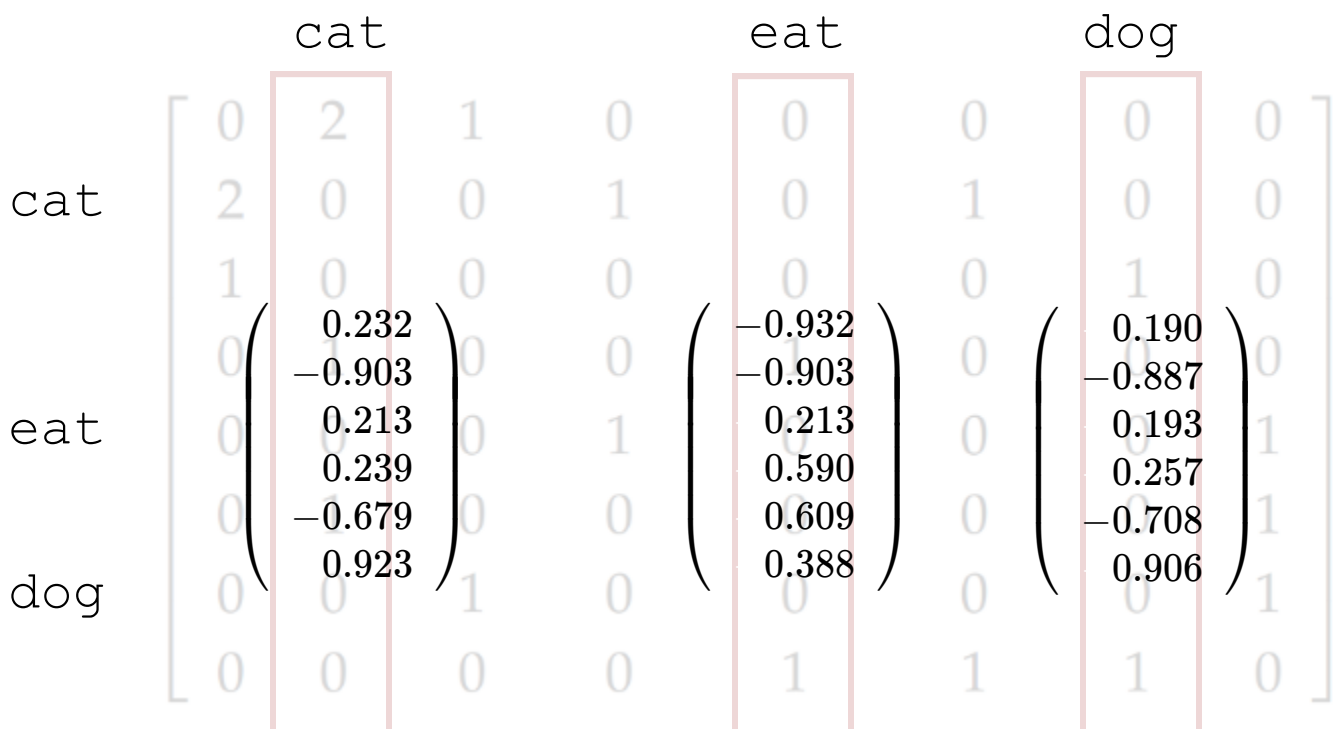


# Co-Occurence Matrix



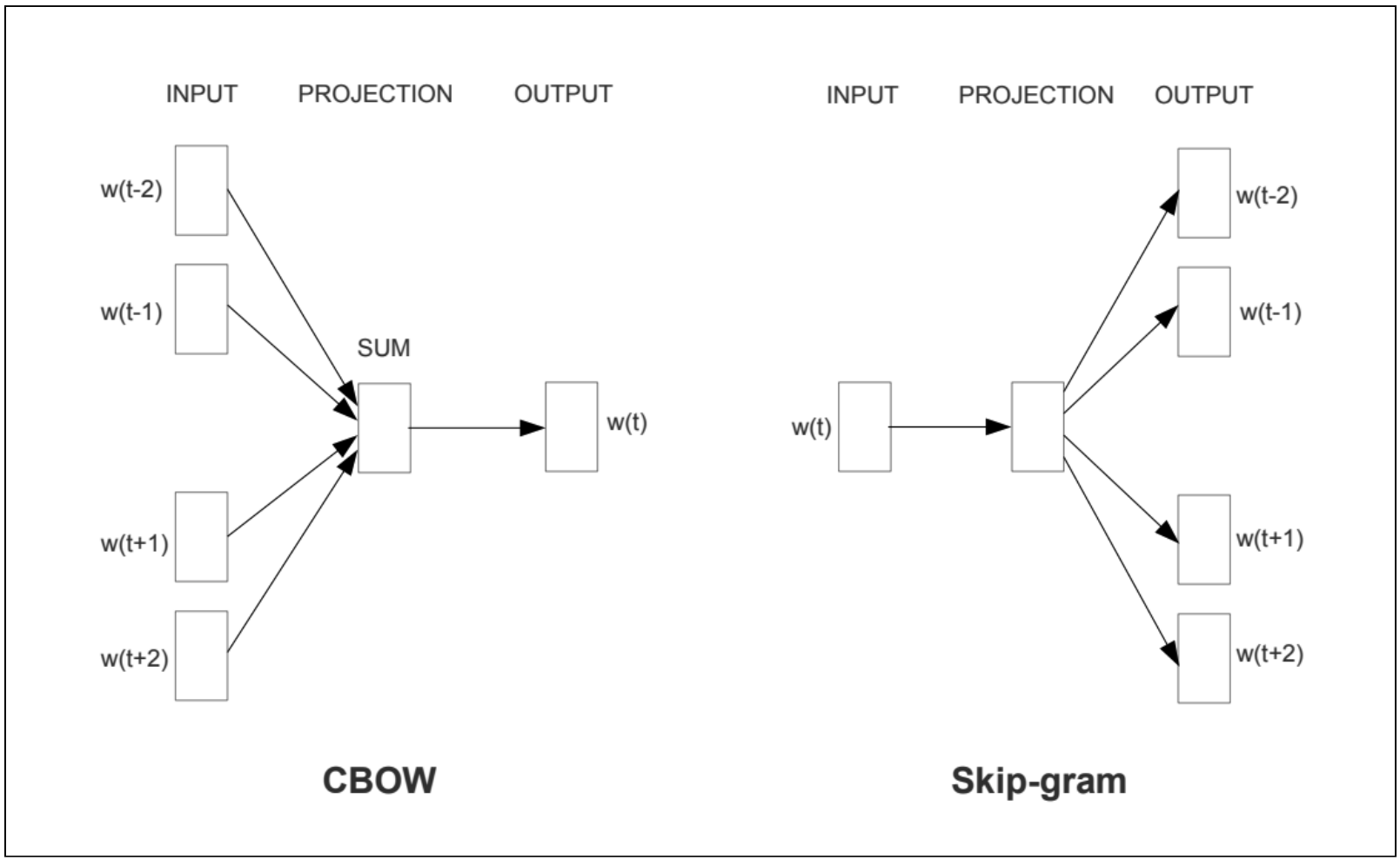

Apply SVD to compress the vectors!

# Co-Occurence Matrix

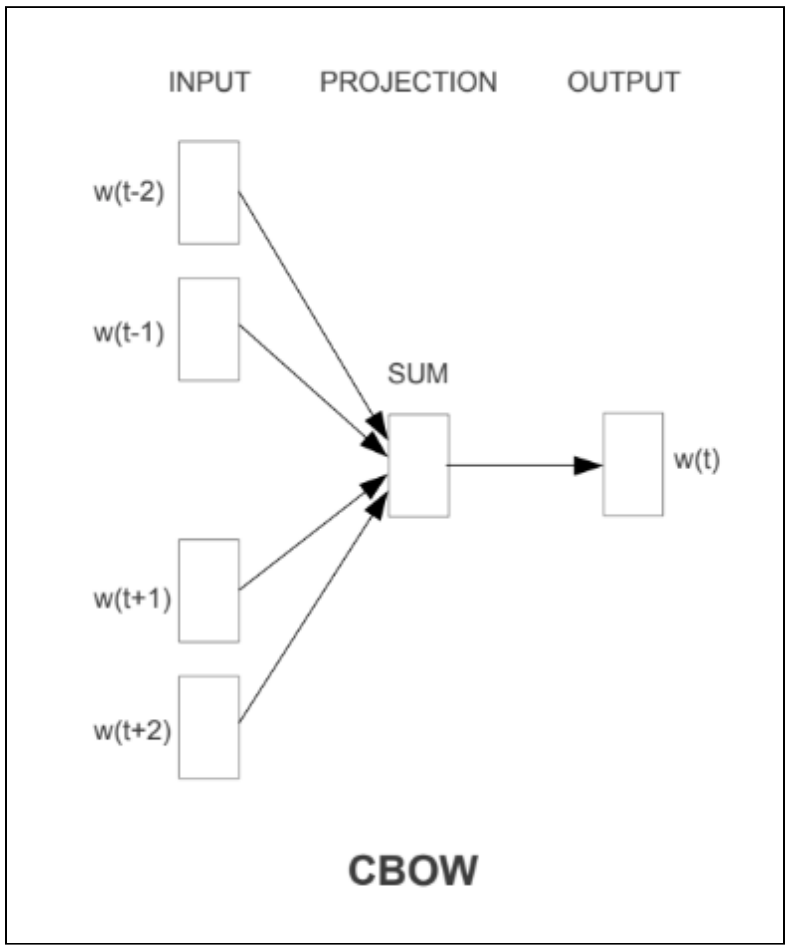



We call the resulting vectors **word embeddings.**

# Word2Vec
## (Mikolov et al. 2013)



# Word2Vec

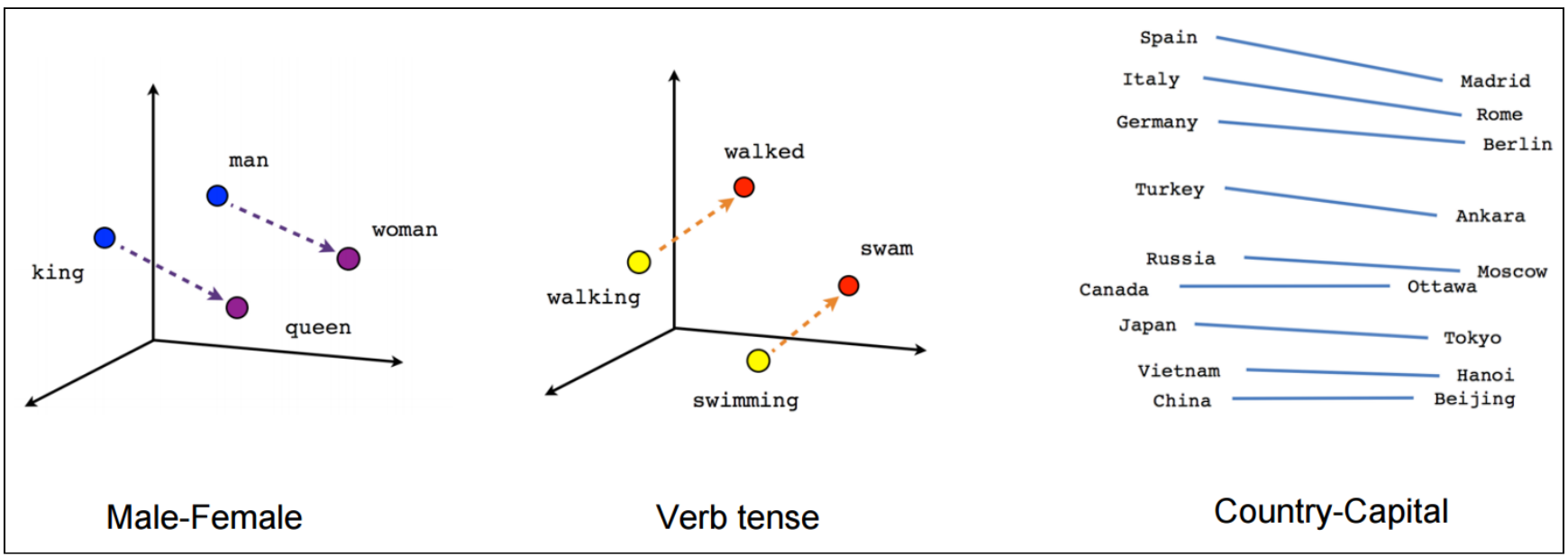



1. Project each context word from |V| dimensional space to its (current) embedding space

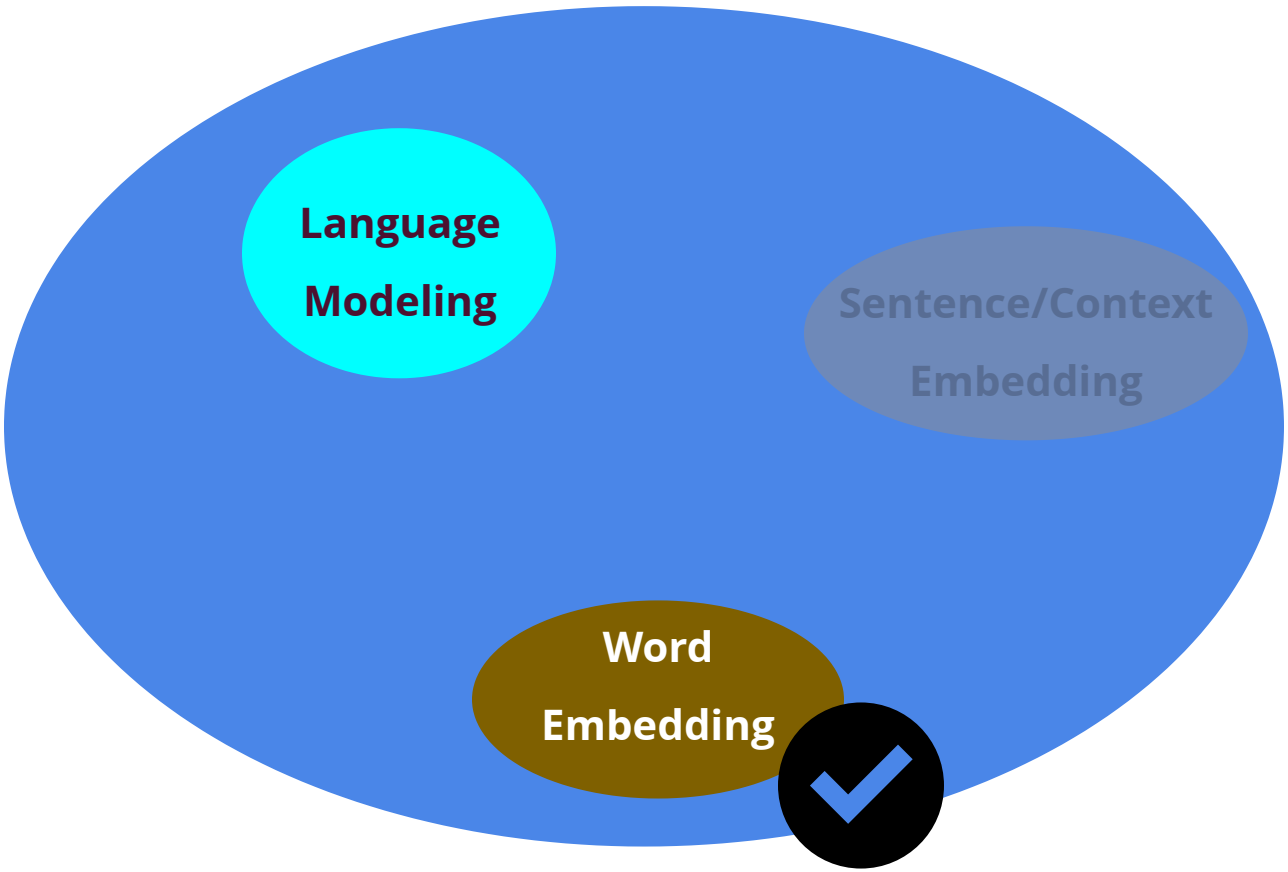
2. Sum these vectors

3. Project it to another space of dimension |V| (we expect this vector to be similar to the one-hot vector of the target word)

# Word2Vec



# Tasks in NLP




**Unsupervised**

# Fundamental Question

How likely can we ever see each English sentence?

# Language Modeling

How likely can we see these sentences in English?

*Fruit flies like a banana* → 0.00078

*I love NLP* → 0.00428

*flies Fruit* → 0.000000001

# Language Model

Each possible sentence should be assigned a probability score (ideally, a "good" sentence should have a higher probability).

The sum of all such probabilities should be 1.

# *n*-Gram Language Model

$$p(x_1, x_2, \ldots, x_m) = \prod_{i=1,\ldots,m} p(x_i | x_{i-n+1}, \ldots, x_{i-1})$$

## Bigram LM

$$n = 2$$

$$p(x_1, x_2, \ldots, x_m) = \prod_{i=1,\ldots,m} p(x_i | x_{i-1})$$

# *n*-Gram Language Model

$$p(x_1, x_2, \ldots, x_m) = \prod_{i=1,\ldots,m} p(x_i | x_{i-n+1}, \ldots, x_{i-1})$$

How many model parameters do we have to store for a n-gram model?

# Curse of Dimensionality

$$\text{in the order of } |V|^n$$

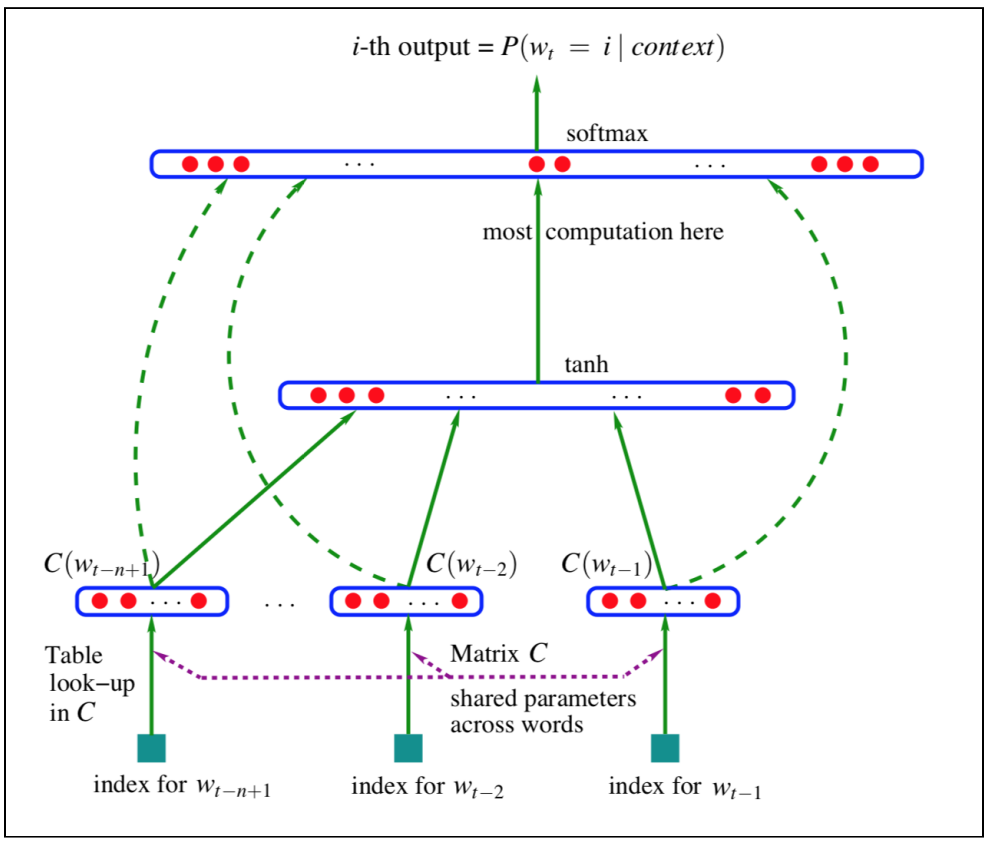
**Too many model parameters!**

One Solution:

Learn language model with word embeddings!

# Neural Language Model

## (Bengio et al. 2003)

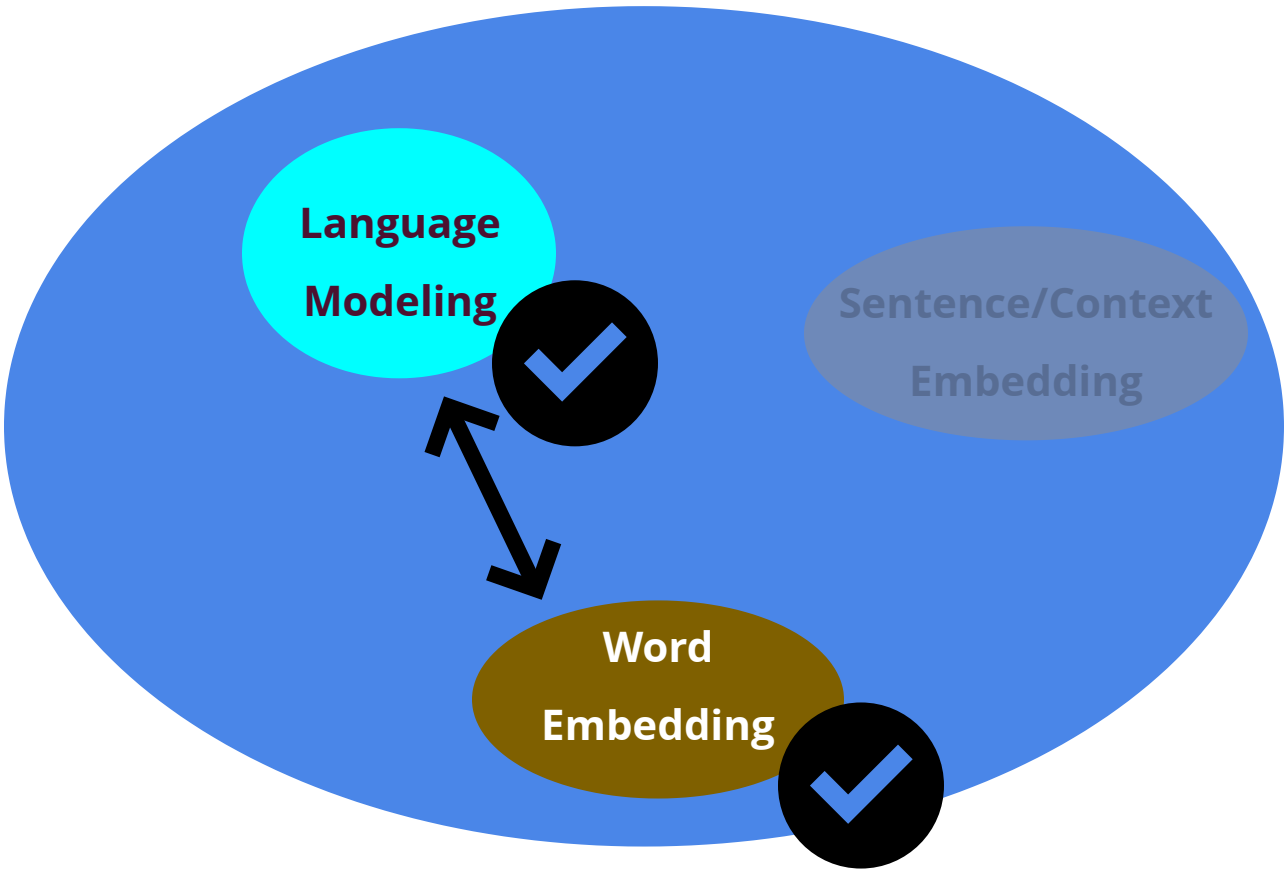



Learn simultaneously the *word embeddings* and the parameters of the language model *probability function*.

# Neural Language Model

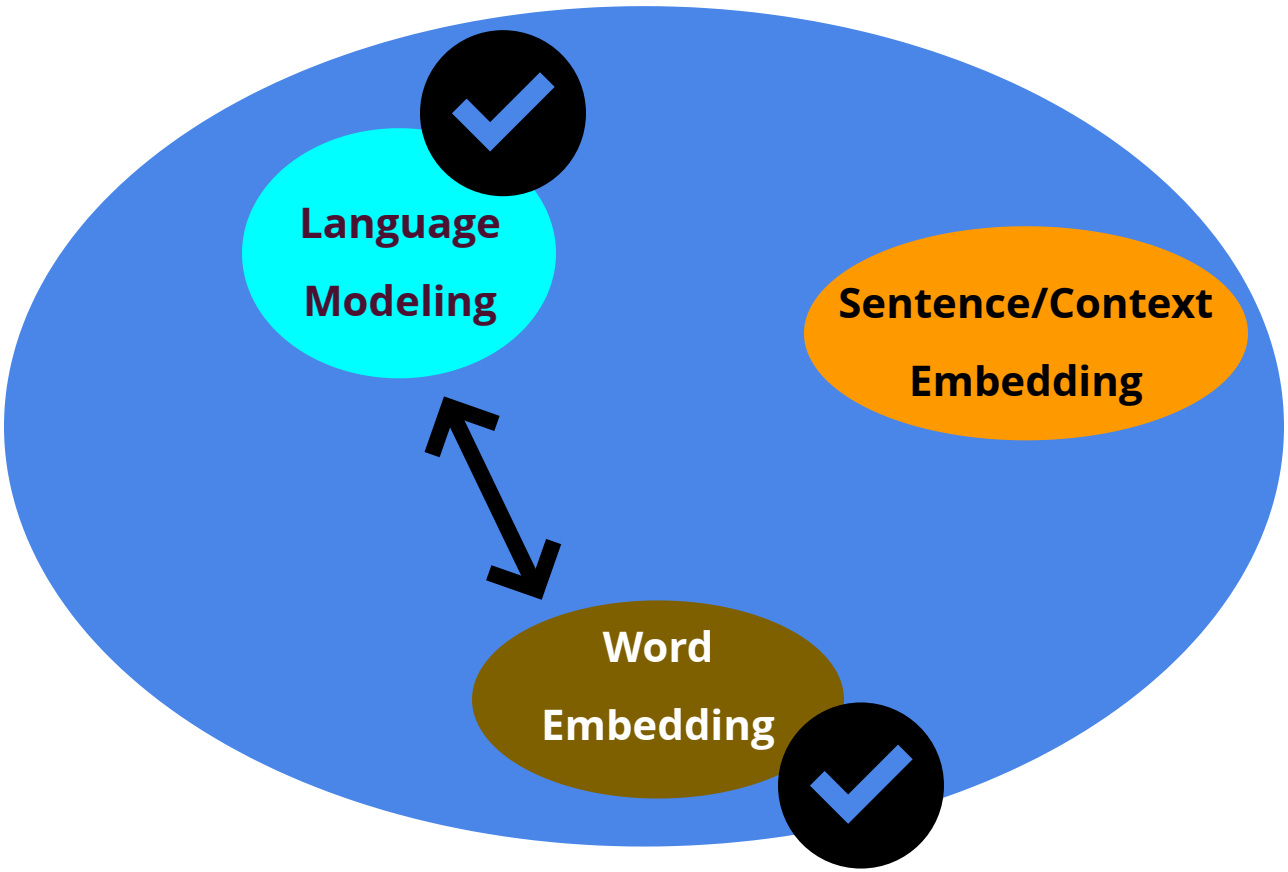
1. associate with each word in the vocabulary a distributed *word embedding*,
2. express the joint probability function of word sequences in terms of the embeddings of these words in the sequence, and
3. learn simultaneously the *word embeddings* and the parameters of that *probability function*.

# Tasks in NLP




**Unsupervised**

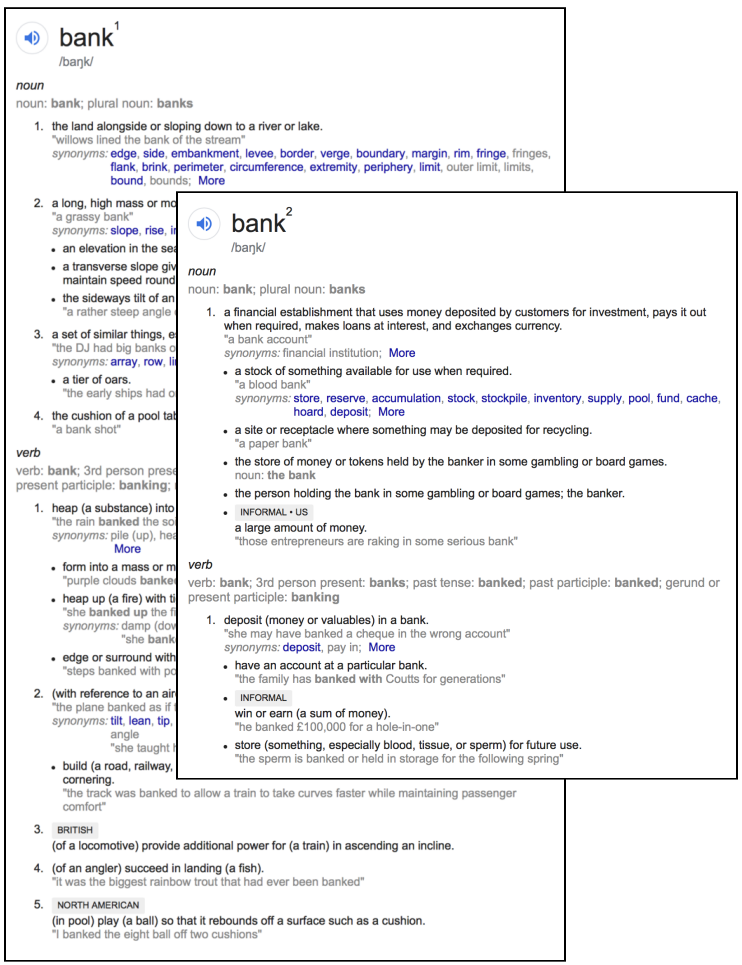
# Tasks in NLP




**Unsupervised**

# Word Senses

A word may have multiple senses.



**bank**[1]
/baŋk/

*noun*

noun: **bank**; plural noun: **banks**

1. the land alongside or sloping down to a river or lake.
   "willows lined the bank of the stream"
   *synonyms:* edge, side, embankment, levee, border, verge, boundary, margin, rim, fringe, fringes, flank, brink, perimeter, circumference, extremity, periphery, limit, outer limit, limits, bound, bounds; More
2. a long, high mass or mo
   "a grassy bank"
   *synonyms:* slope, rise, i
   - an elevation in the sea
   - a transverse slope giv
     maintain speed round
   - the sideways tilt of an
     "a rather steep angle
3. a set of similar things, e
   "the DJ had big banks o
   *synonyms:* array, row, li
   - a tier of oars.
     "the early ships had o
4. the cushion of a pool tab
   "a bank shot"

*verb*

verb: **bank**; 3rd person prese
present participle: **banking**;

1. heap (a substance) into
   "the rain **banked** the so
   *synonyms:* pile (up), hea
   More
   - form into a mass or m
     "purple clouds **banke**
   - heap up (a fire) with ti
     "she **banked up** the f
     *synonyms:* damp (dow
     "she **bank**
   - edge or surround with
     "steps banked with po
2. (with reference to an air
   "the plane banked as if
   *synonyms:* tilt, lean, tip,
   angle
   "she taught h
   - build (a road, railway,
     cornering.
     "the track was banked to allow a train to take curves faster while maintaining passenger comfort"
3. BRITISH
   (of a locomotive) provide additional power for (a train) in ascending an incline.
4. (of an angler) succeed in landing (a fish).
   "it was the biggest rainbow trout that had ever been banked"
5. NORTH AMERICAN
   (in pool) play (a ball) so that it rebounds off a surface such as a cushion.
   "I banked the eight ball off two cushions"

**bank**[2]
/baŋk/

*noun*

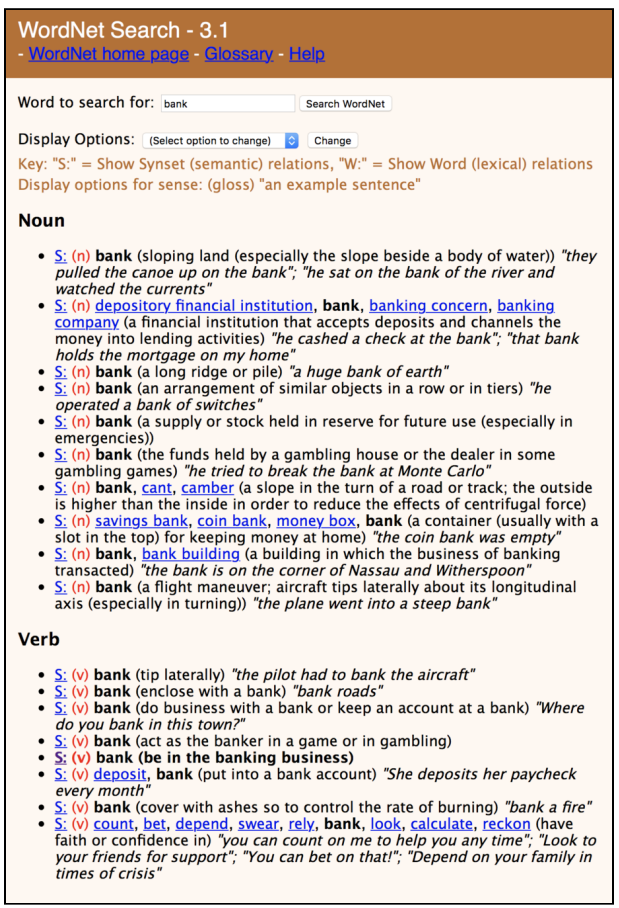
noun: **bank**; plural noun: **banks**

1. a financial establishment that uses money deposited by customers for investment, pays it out when required, makes loans at interest, and exchanges currency.
   "a bank account"
   *synonyms:* financial institution; More
   - a stock of something available for use when required.
     "a blood bank"
     *synonyms:* store, reserve, accumulation, stock, stockpile, inventory, supply, pool, fund, cache, hoard, deposit; More
   - a site or receptacle where something may be deposited for recycling.
     "a paper bank"
   - the store of money or tokens held by the banker in some gambling or board games.
     noun: **the bank**
   - the person holding the bank in some gambling or board games; the banker.
   - INFORMAL • US
     a large amount of money.
     "those entrepreneurs are raking in some serious bank"

*verb*

verb: **bank**; 3rd person present: **banks**; past tense: **banked**; past participle: **banked**; gerund or present participle: **banking**

1. deposit (money or valuables) in a bank.
   "she may have banked a cheque in the wrong account"
   *synonyms:* deposit, pay in; More
   - have an account at a particular bank.
     "the family has **banked with** Coutts for generations"
   - INFORMAL
     win or earn (a sum of money).
     "he banked £100,000 for a hole-in-one"
   - store (something, especially blood, tissue, or sperm) for future use.
     "the sperm is banked or held in storage for the following spring"



**WordNet Search - 3.1**
- WordNet home page - Glossary - Help

Word to search for: bank Search WordNet

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (gloss) "an example sentence"

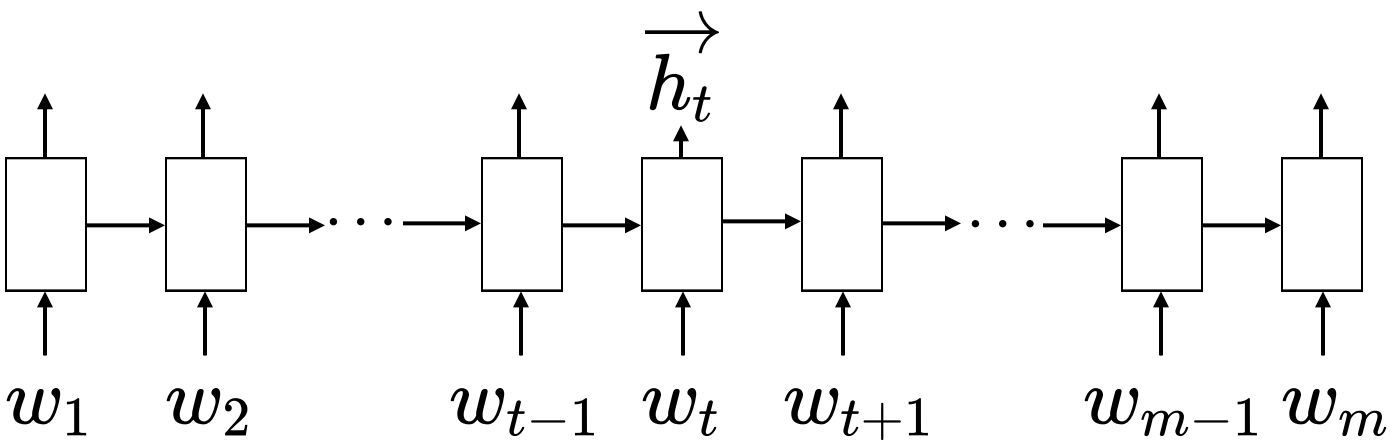
**Noun**

- S: (n) **bank** (sloping land (especially the slope beside a body of water)) *"they pulled the canoe up on the bank"; "he sat on the bank of the river and watched the currents"*
- S: (n) depository financial institution, **bank**, banking concern, banking company (a financial institution that accepts deposits and channels the money into lending activities) *"he cashed a check at the bank"; "that bank holds the mortgage on my home"*
- S: (n) **bank** (a long ridge or pile) *"a huge bank of earth"*
- S: (n) **bank** (an arrangement of similar objects in a row or in tiers) *"he operated a bank of switches"*
- S: (n) **bank** (a supply or stock held in reserve for future use (especially in emergencies))
- S: (n) **bank** (the funds held by a gambling house or the dealer in some gambling games) *"he tried to break the bank at Monte Carlo"*
- S: (n) **bank**, cant, camber (a slope in the turn of a road or track; the outside is higher than the inside in order to reduce the effects of centrifugal force)
- S: (n) savings bank, coin bank, money box, **bank** (a container (usually with a slot in the top) for keeping money at home) *"the coin bank was empty"*
- S: (n) **bank**, bank building (a building in which the business of banking transacted) *"the bank is on the corner of Nassau and Witherspoon"*
- S: (n) **bank** (a flight maneuver; aircraft tips laterally about its longitudinal axis (especially in turning)) *"the plane went into a steep bank"*

**Verb**

- S: (v) **bank** (tip laterally) *"the pilot had to bank the aircraft"*
- S: (v) **bank** (enclose with a bank) *"bank roads"*
- S: (v) **bank** (do business with a bank or keep an account at a bank) *"Where do you bank in this town?"*
- S: (v) **bank** (act as the banker in a game or in gambling)
- **S: (v) bank (be in the banking business)**
- S: (v) deposit, **bank** (put into a bank account) *"She deposits her paycheck every month"*
- S: (v) **bank** (cover with ashes so to control the rate of burning) *"bank a fire"*
- S: (v) count, bet, depend, swear, rely, **bank**, look, calculate, reckon (have faith or confidence in) *"you can count on me to help you any time"; "Look to your friends for support"; "You can bet on that!"; "Depend on your family in times of crisis"*

# Long Short-Term Memory
## (Hochreiter & Schmidhuber 1997)




$$\overrightarrow{h_t} = \overrightarrow{\text{LSTM}}(w_1, w_2, \ldots, w_t)$$

# LSTM

$c_{t-1}$

The long-term memory

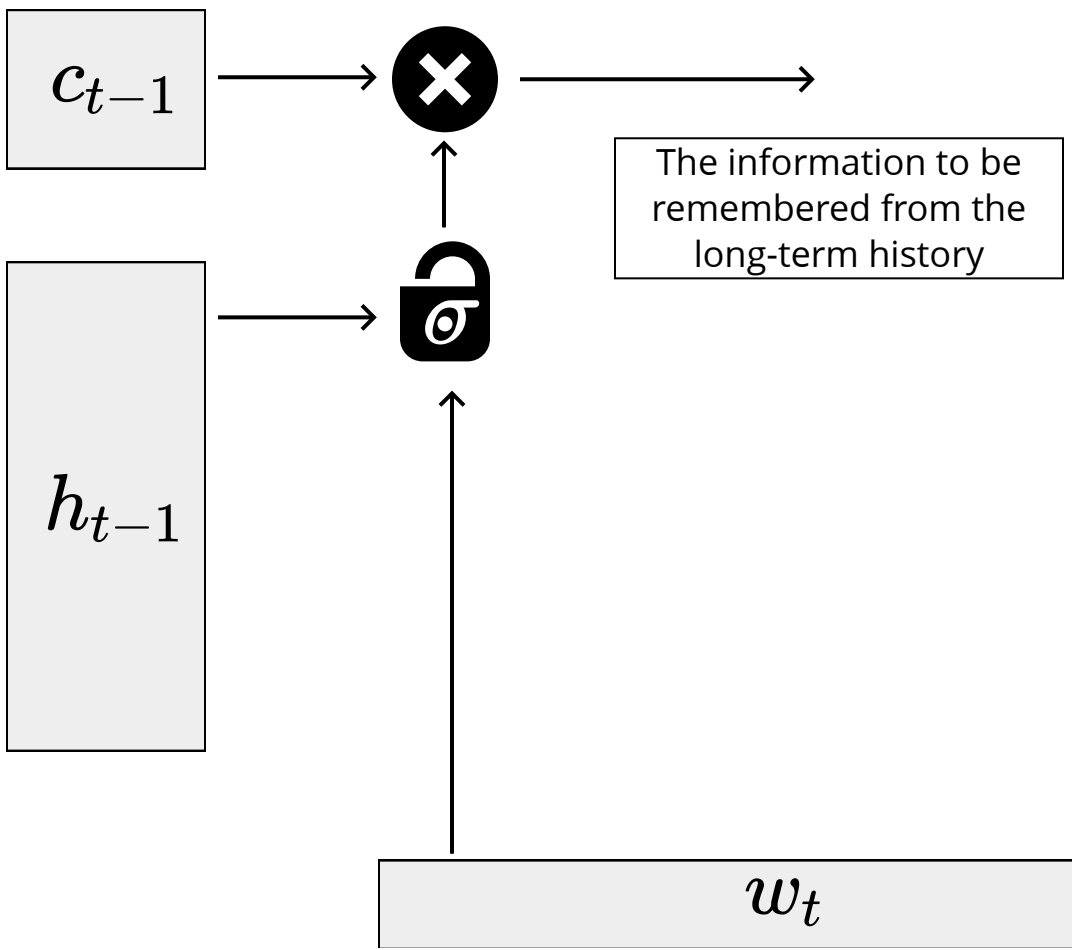
$h_{t-1}$

The working memory

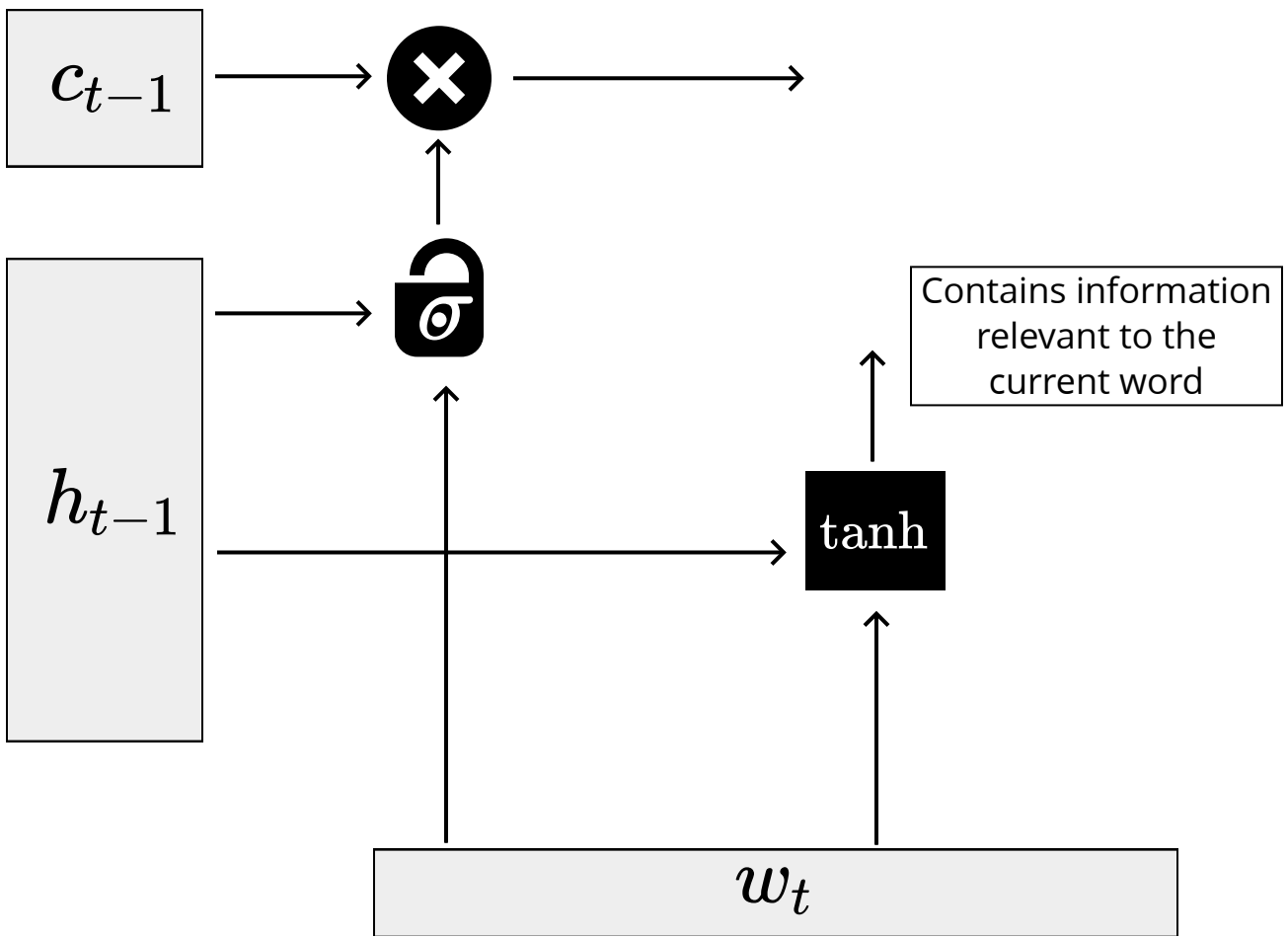
The current word

$w_t$

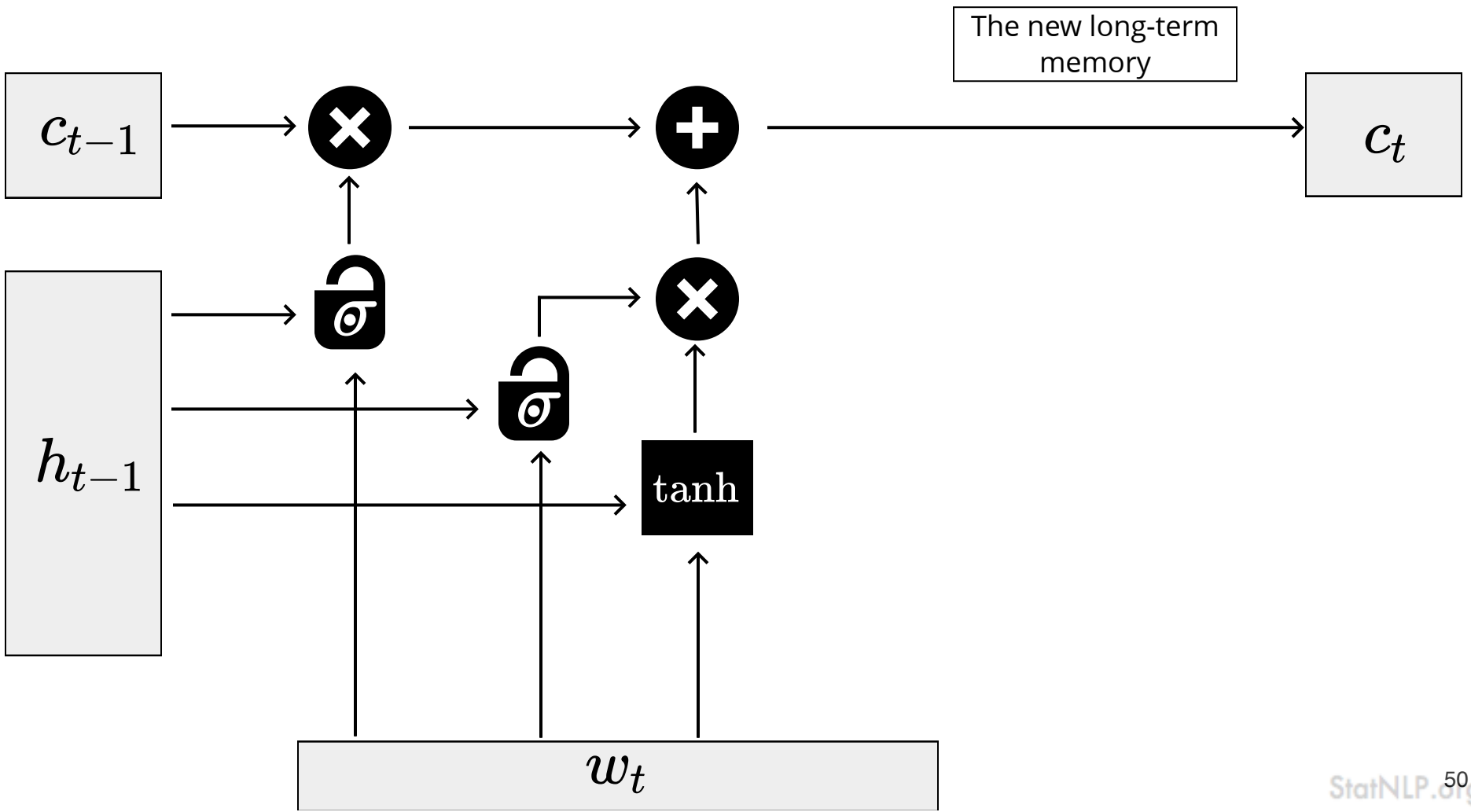
# LSTM



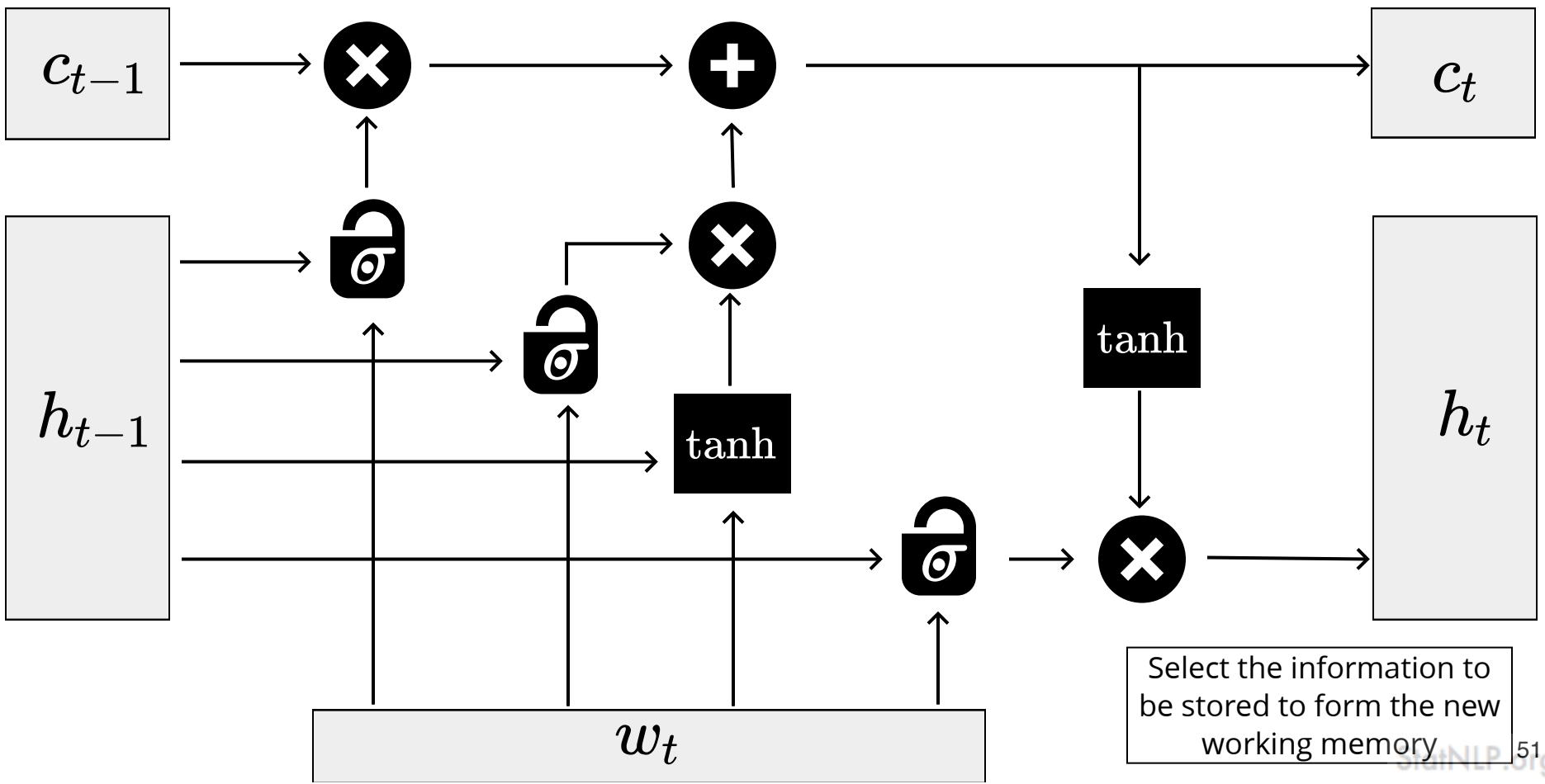
# LSTM



# LSTM

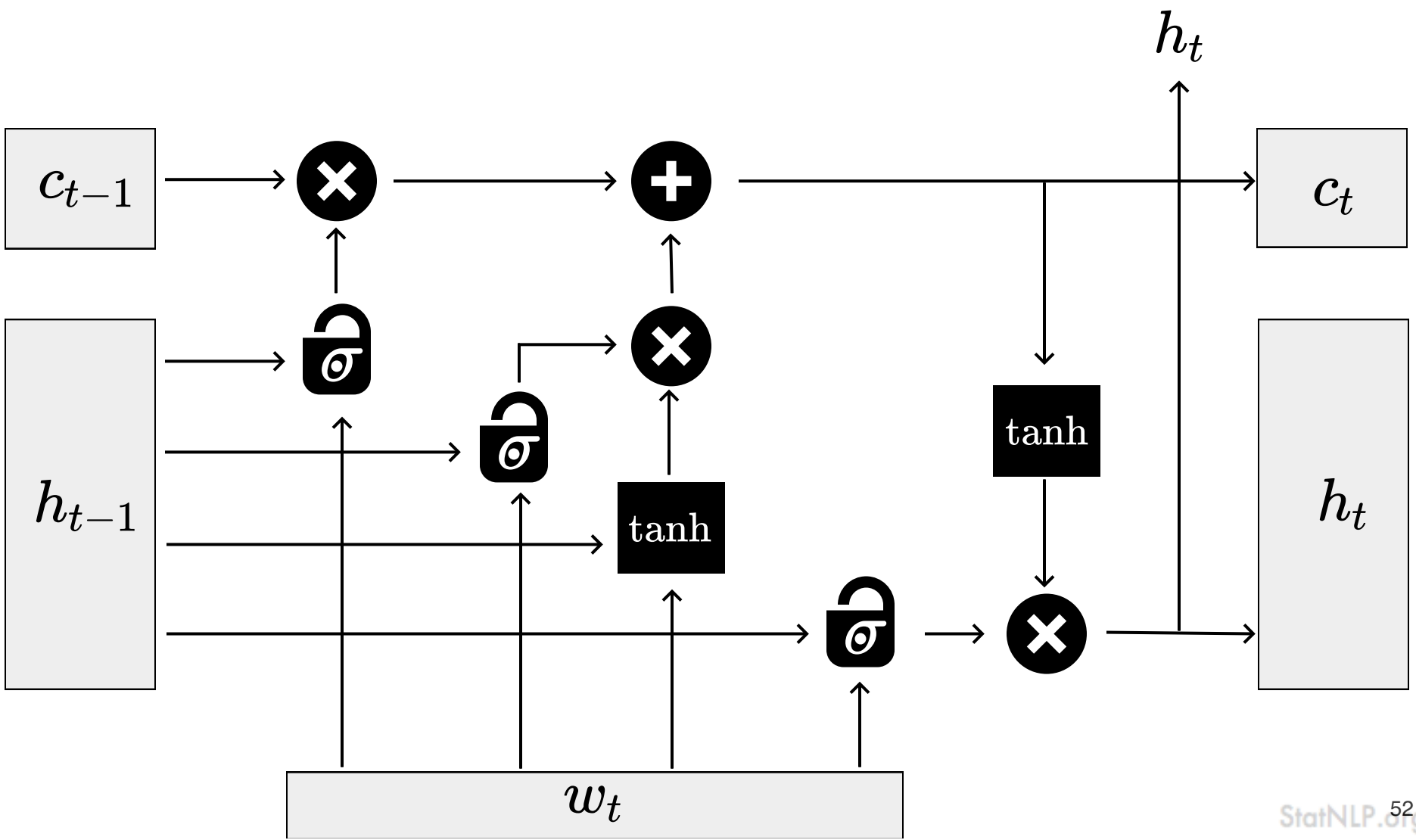


# LSTM

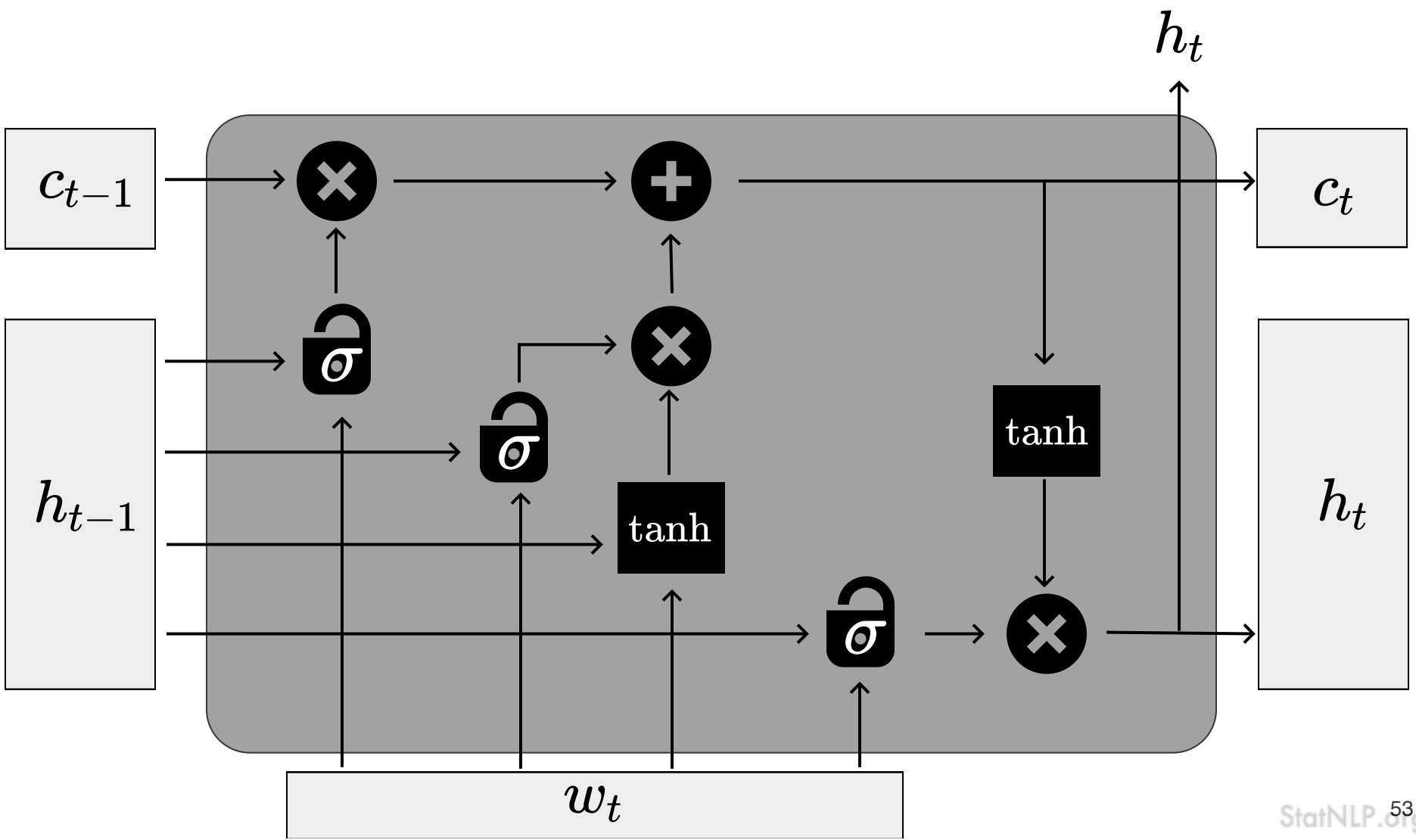



Select the information to be stored to form the new working memory

# LSTM

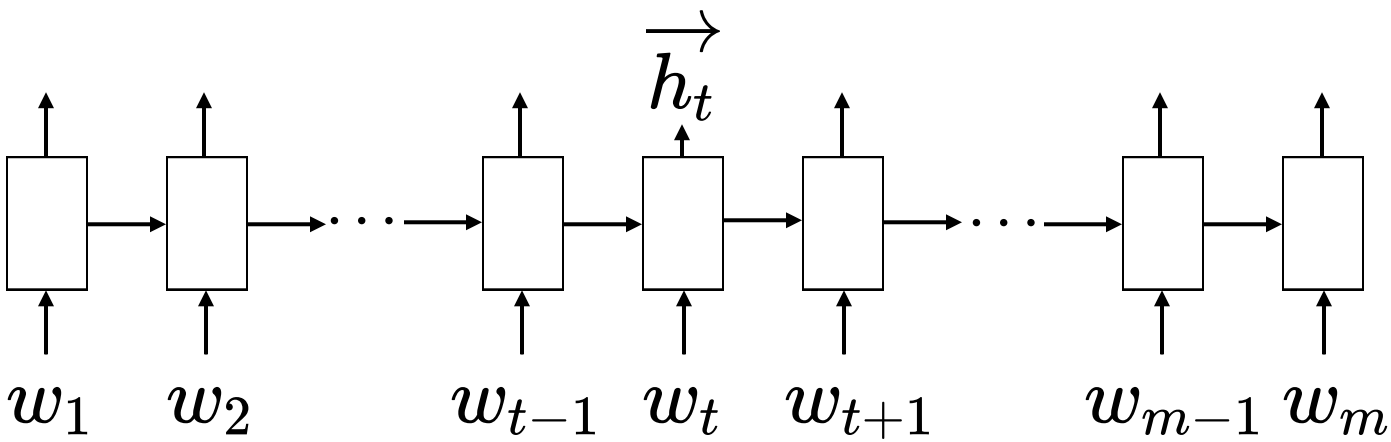


# LSTM



# LSTM Language Model

## (Sundermeyer et al. 2012)



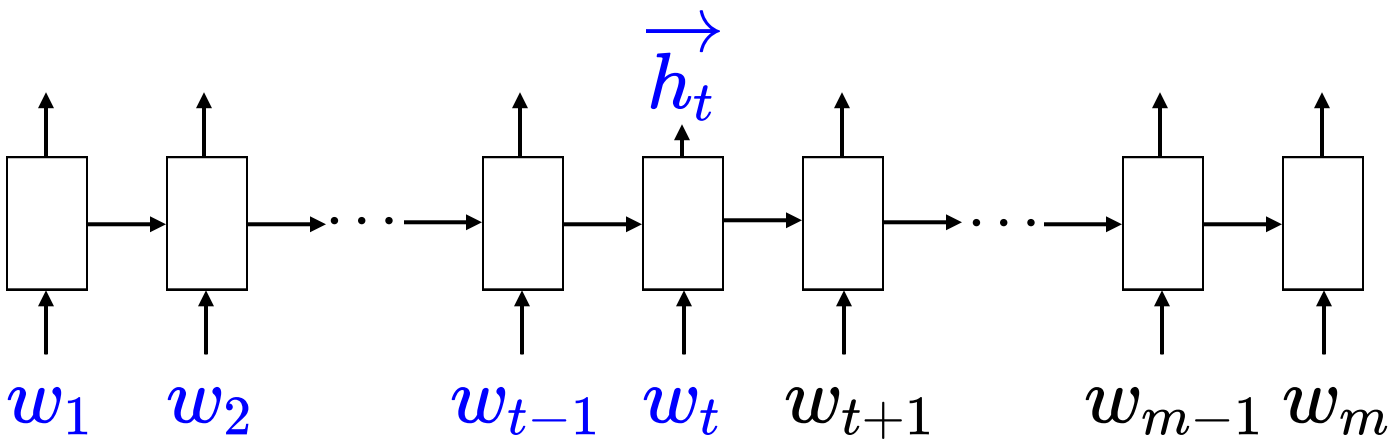

$$\overrightarrow{h_t} = \overrightarrow{\text{LSTM}}(w_1, w_2, \ldots, w_t)$$

Individual probability at each position:

1. project $\overrightarrow{h_t}$ to a $|V|$ dimensional space
2. apply softmax on top of the vector
3. get the probability of generating the desired word $w_{t+1}$

# LSTM Language Model

## (Sundermeyer et al. 2012)



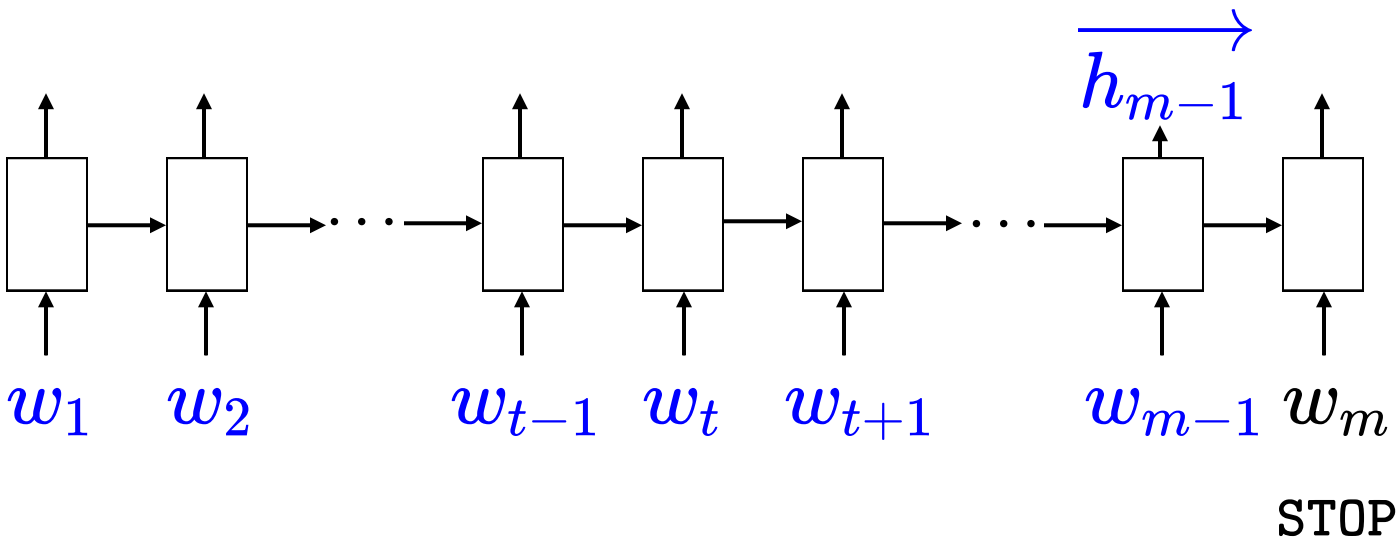

$$\overrightarrow{h_t} = \overrightarrow{\text{LSTM}}(w_1, w_2, \ldots, w_t)$$

Context Embedding!

A function over a sequence of word embeddings.

# LSTM Language Model

## (Sundermeyer et al. 2012)




$$\overrightarrow{h_{m-1}} = \overrightarrow{\text{LSTM}}(w_1, w_2, \ldots, w_{m-1})$$

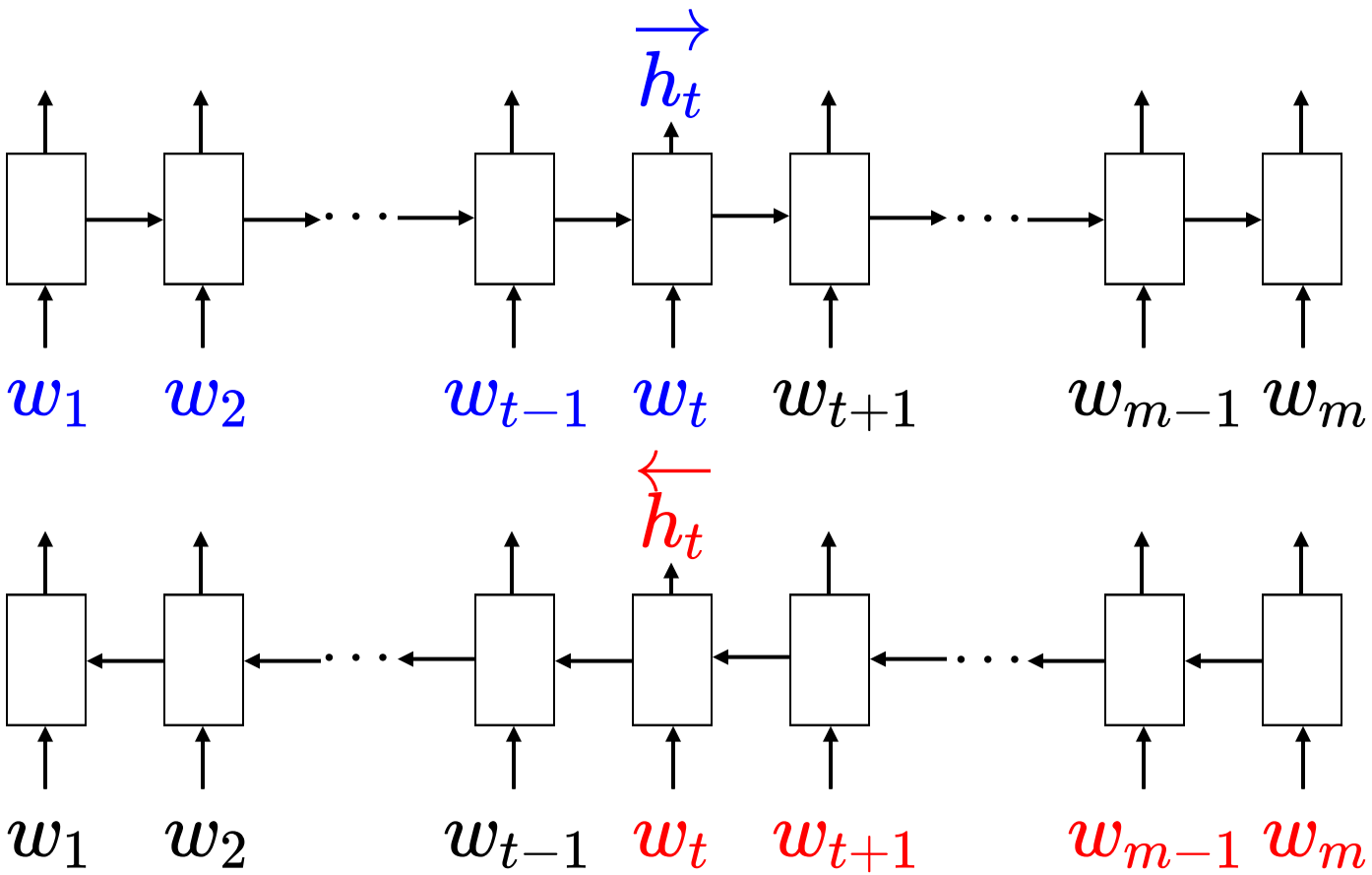
Sentence Embedding!

It is essentially a special context embedding.

# Bidirectional LSTM

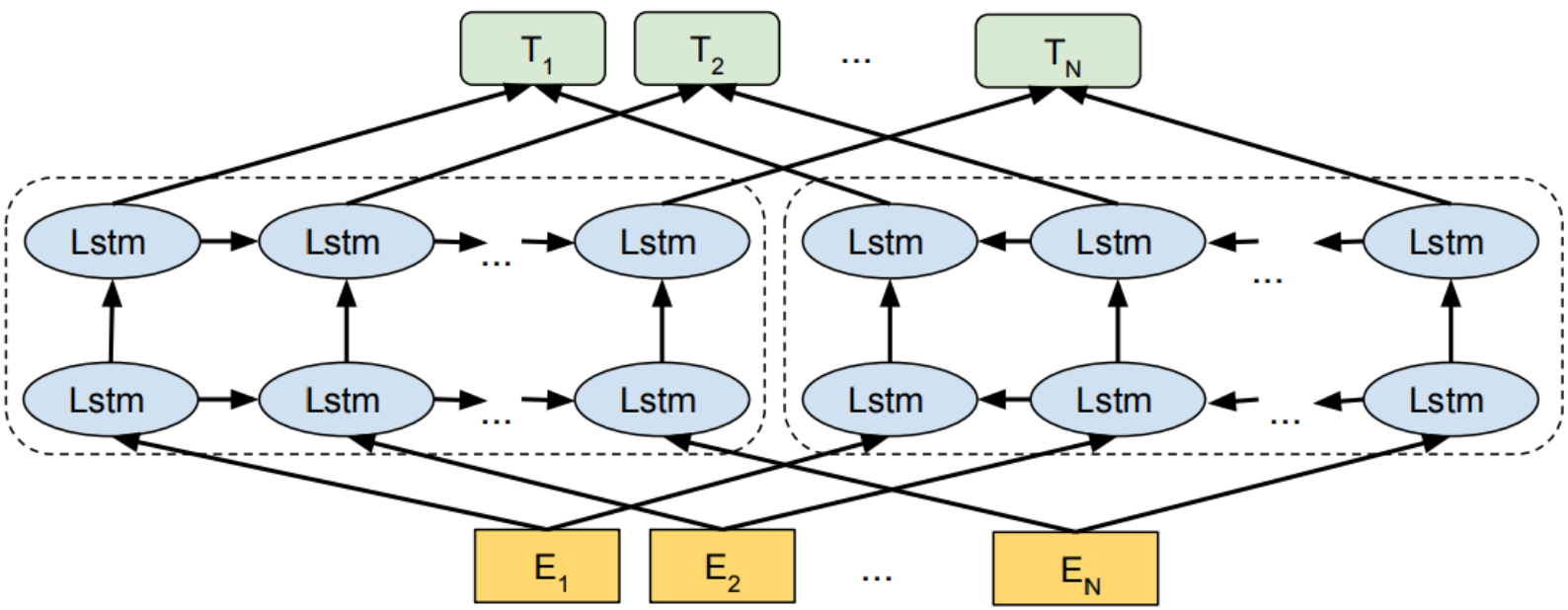
Context Embedding

$$[\overrightarrow{h_t}; \overleftarrow{h_t}]$$



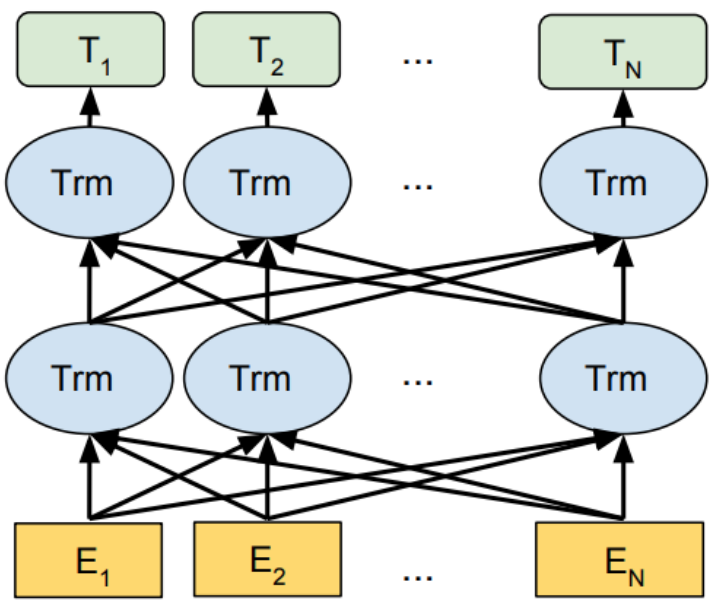
# ELMo

## Embeddings from Language Models Trained with Two LSTM LMs (Peters et al. 2018)



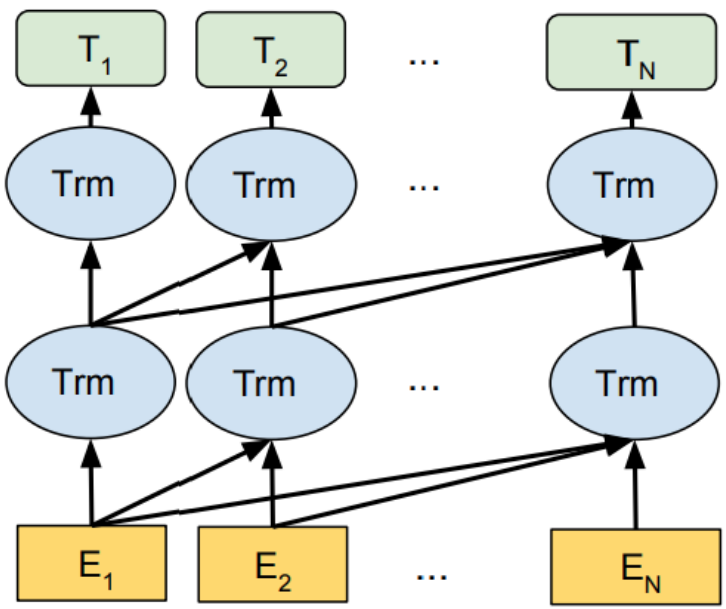
# BERT

## Bidirectional Transformer Encoder Trained with Masked LM (Devlin et al. 2019)

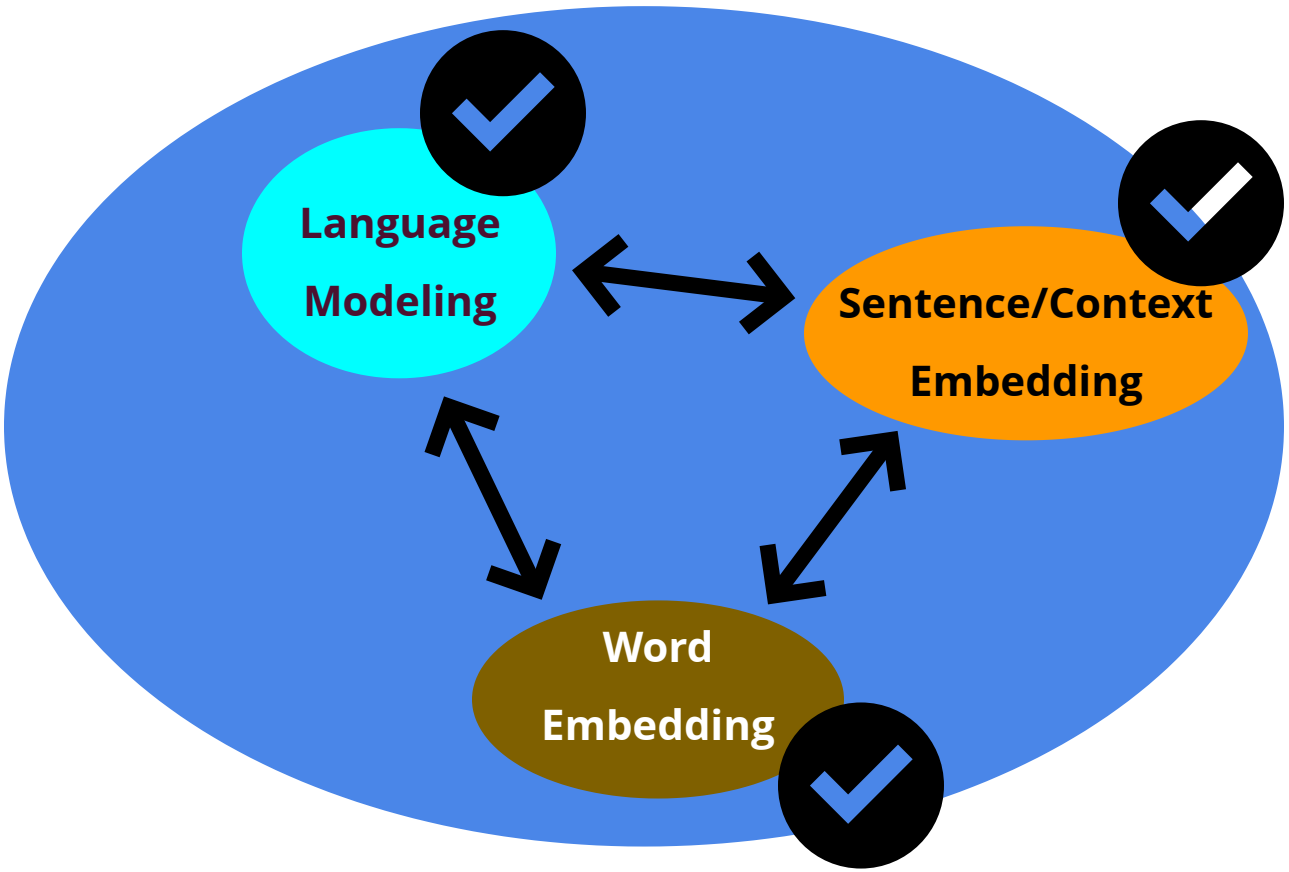


# GPT-3

## (Directional) Transformer Encoder Trained with LM (Brown et al. 2020)

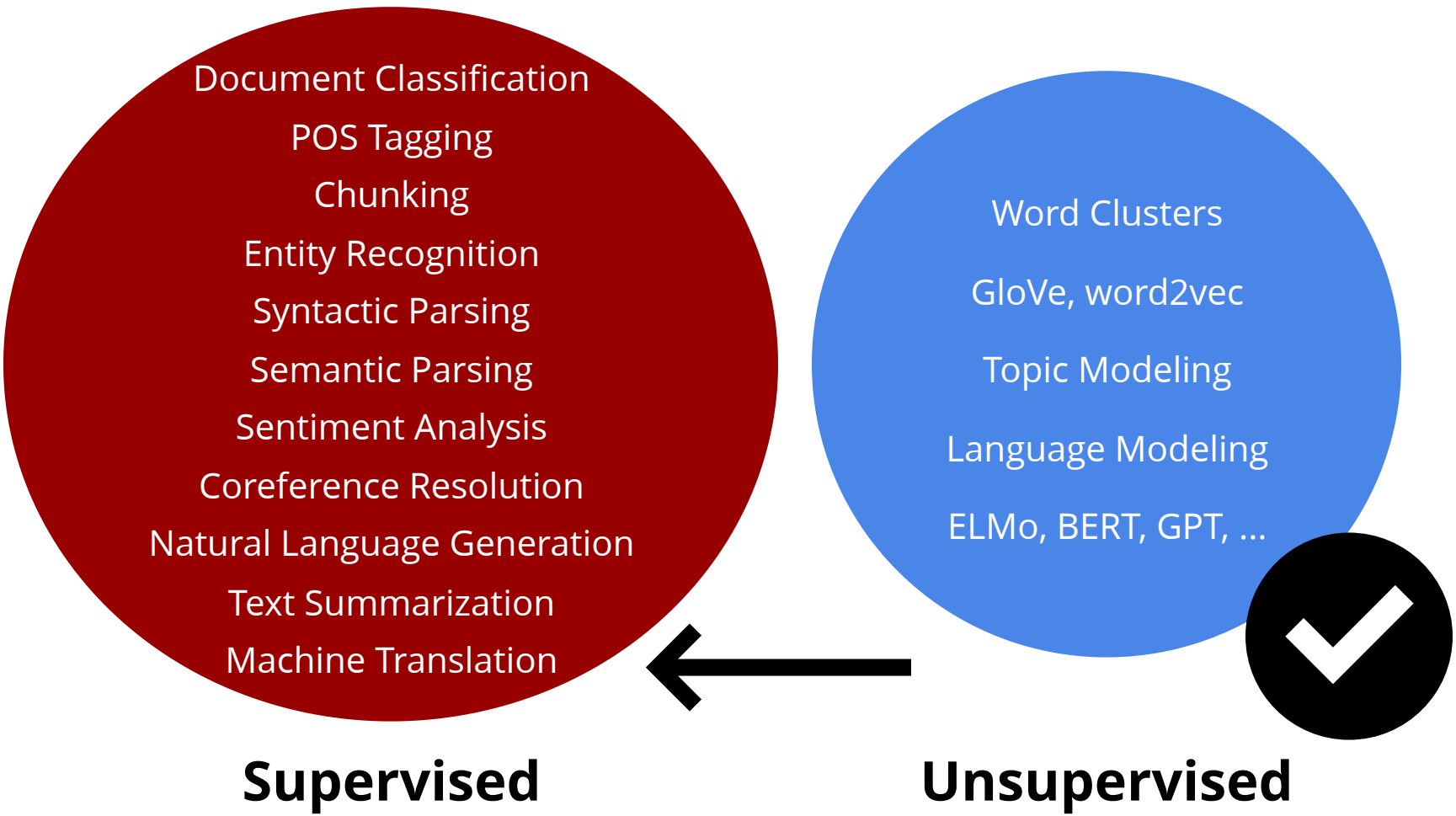


# Tasks in NLP

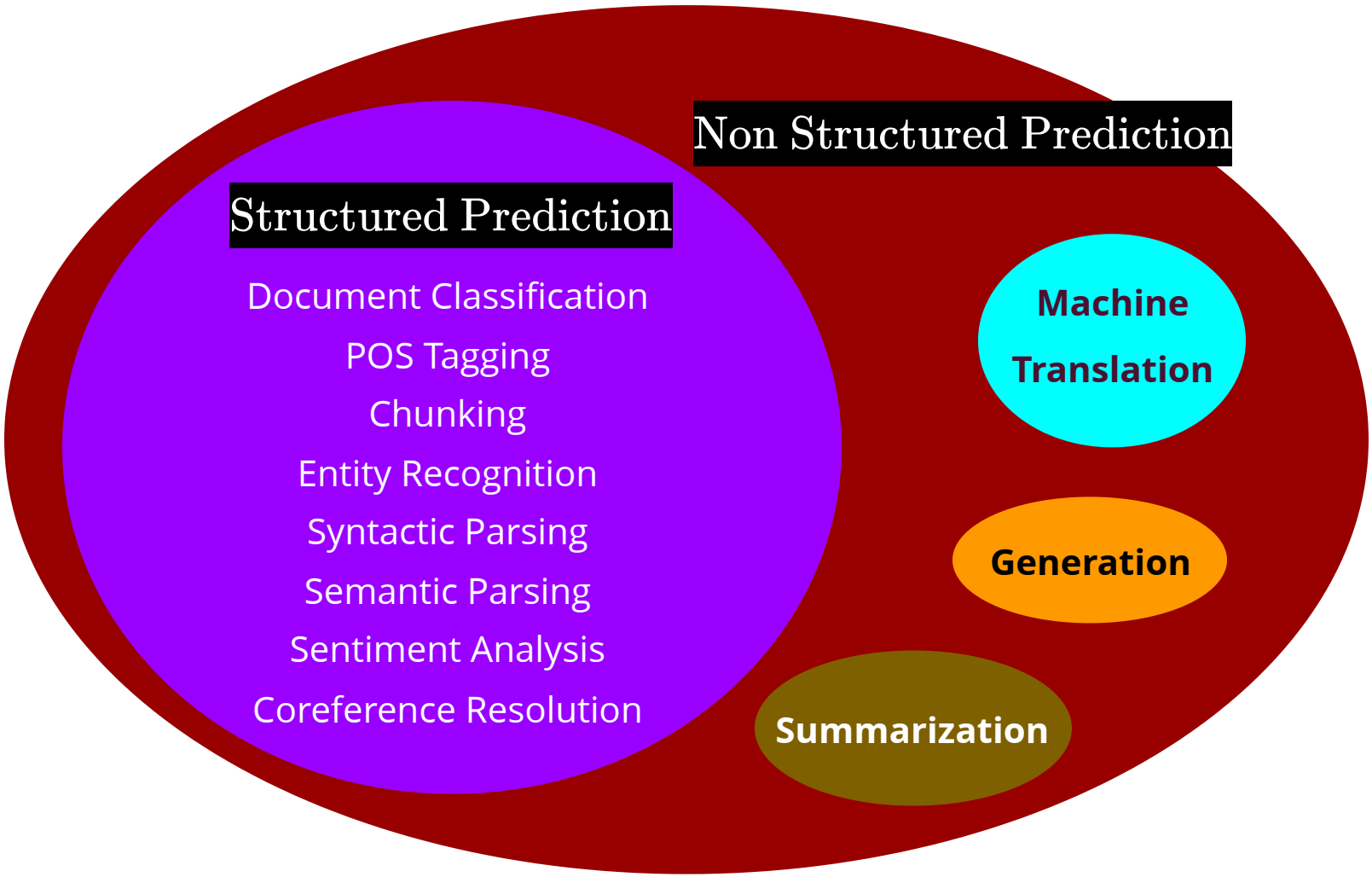



**Unsupervised**

# Tasks in NLP

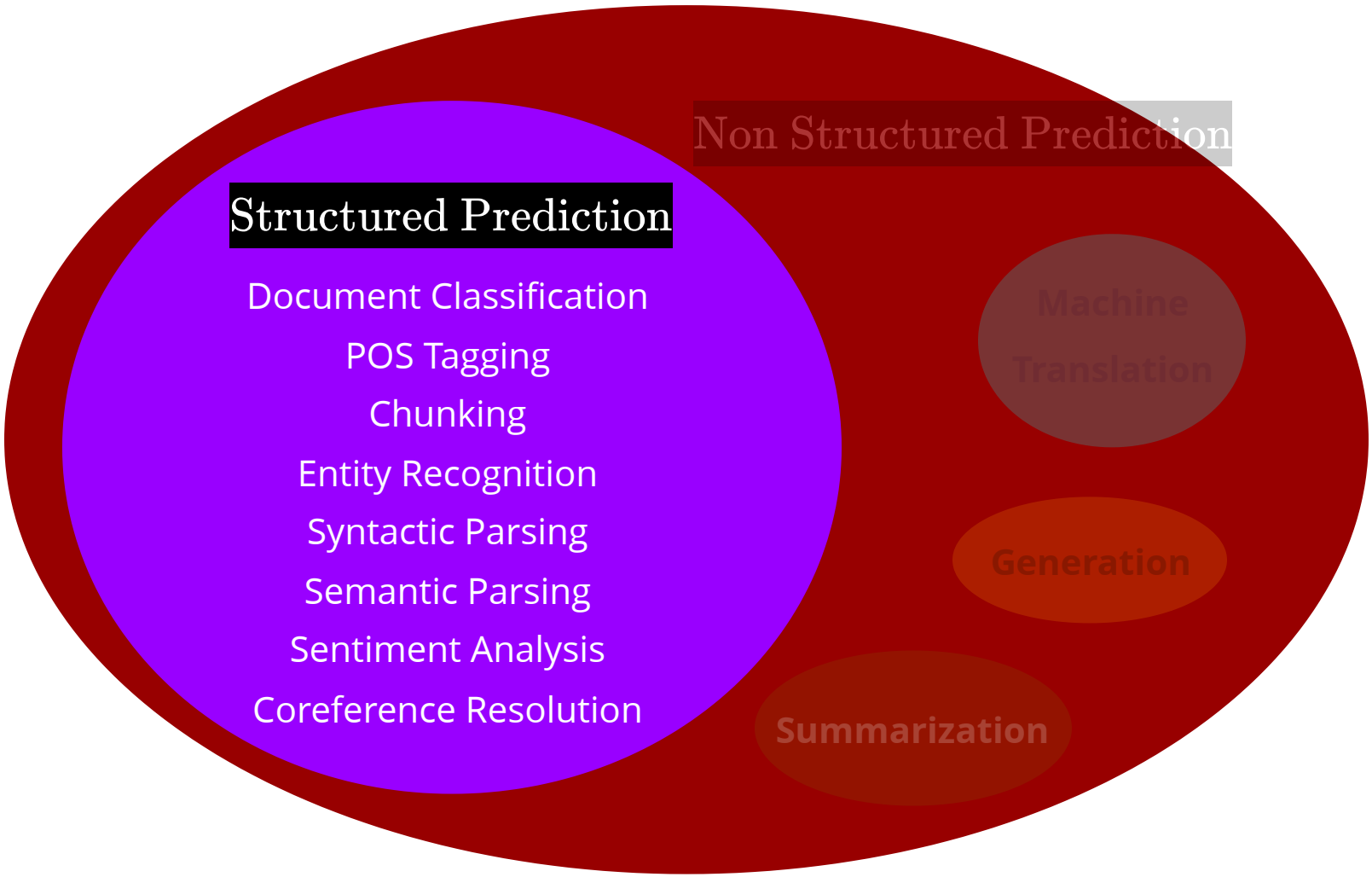


# Tasks in NLP



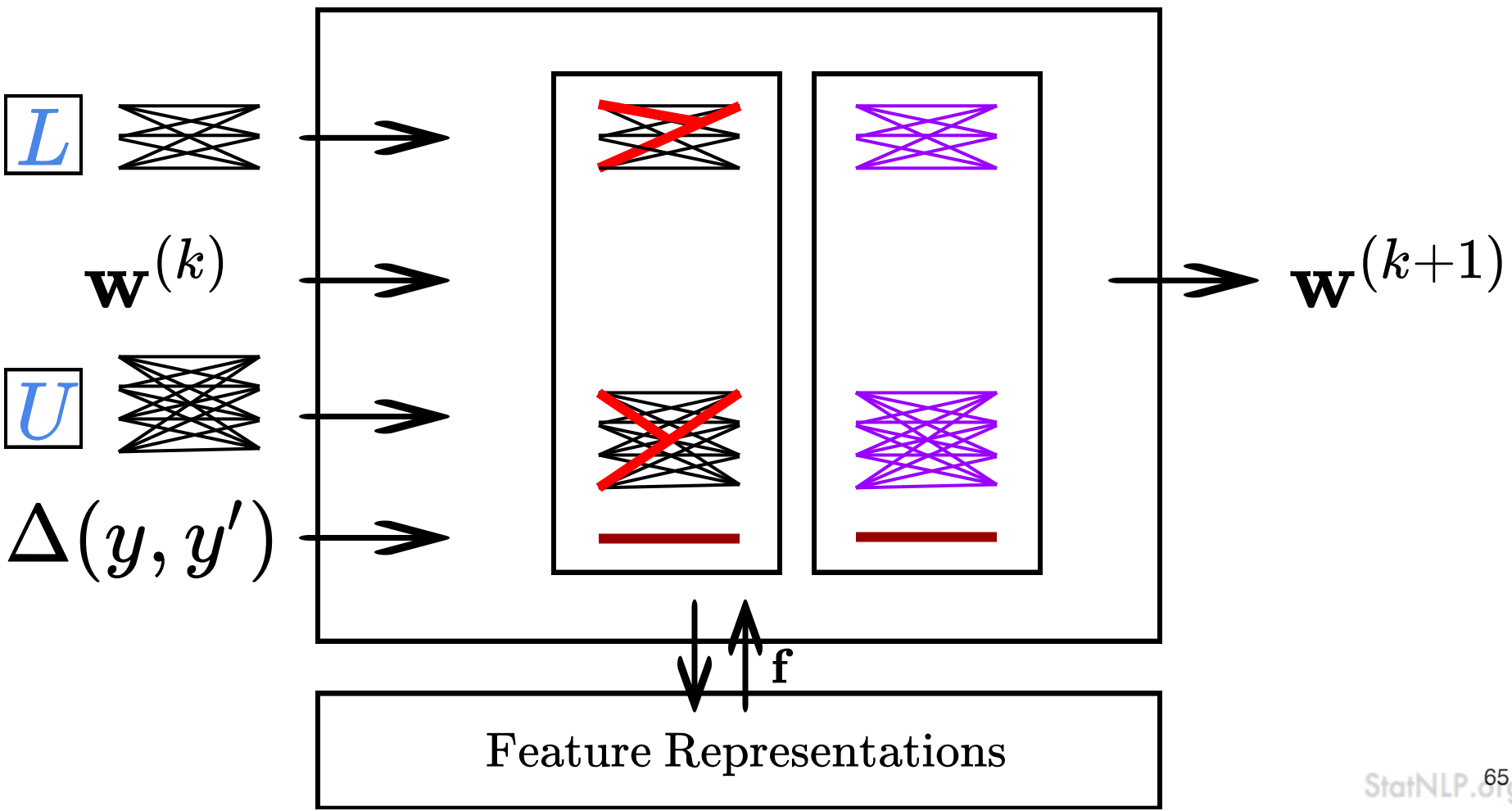

**Supervised**

# Tasks in NLP




**Supervised**

# A Unified Framework for Structured Prediction



# Structured Prediction

**One Assumption**

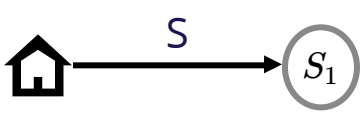
Structures are constructed by following a collection of discrete actions.

# States, Actions

S: shift

L: left-arc

R:right-arc



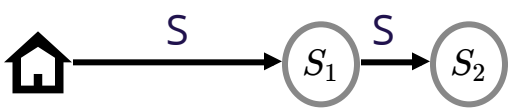

*Fruit* *flies* *like* *a* *banana*

# States, Actions

S: shift

L: left-arc

R:right-arc



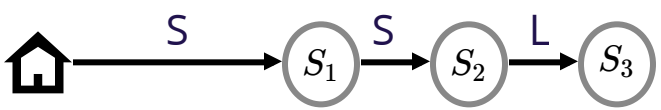

*Fruit* *flies* *like* *a* *banana*

# States, Actions

S: shift

L: left-arc

R:right-arc





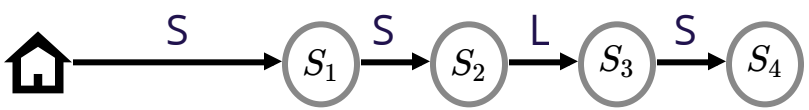
*Fruit* *flies* *like* *a* *banana*

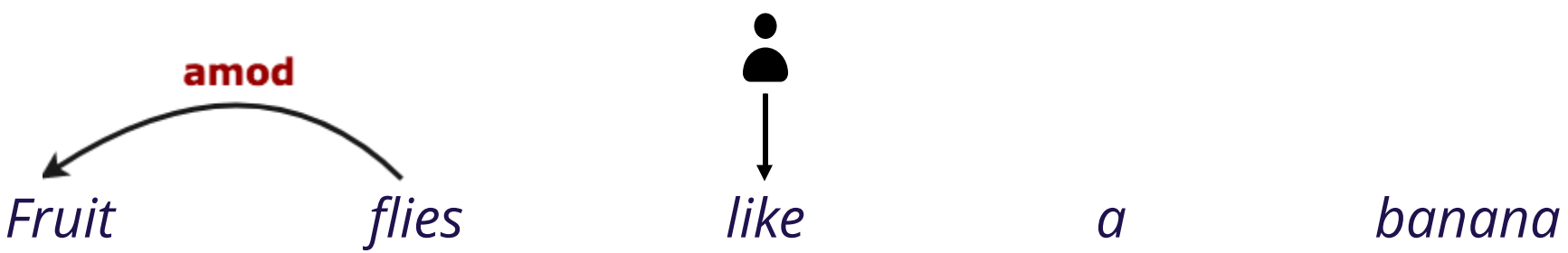
# States, Actions

S: shift

L: left-arc

R:right-arc



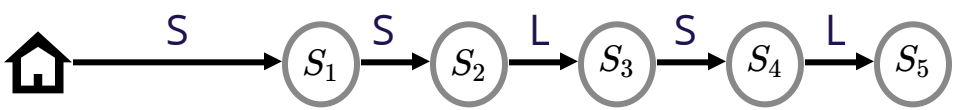



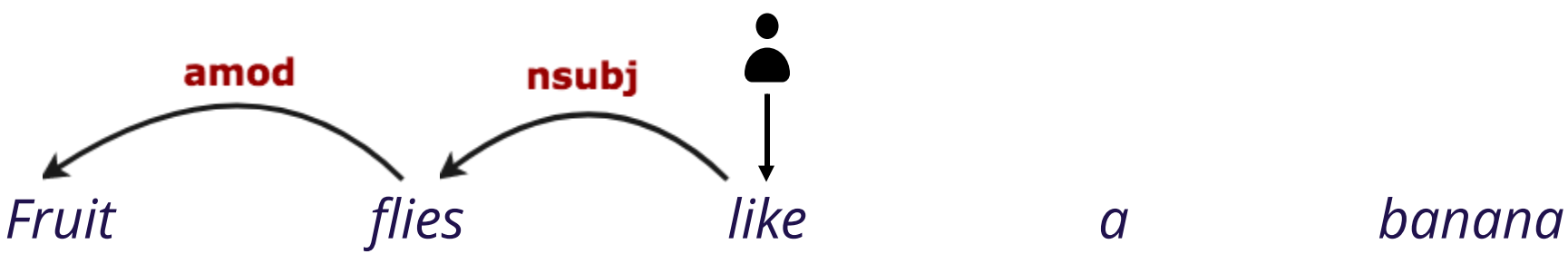
# States, Actions

S: shift

L: left-arc

R:right-arc



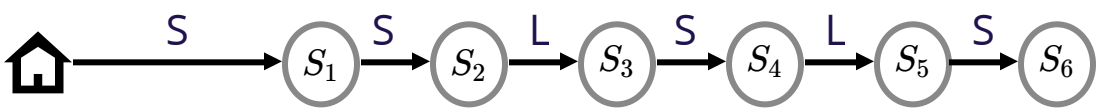



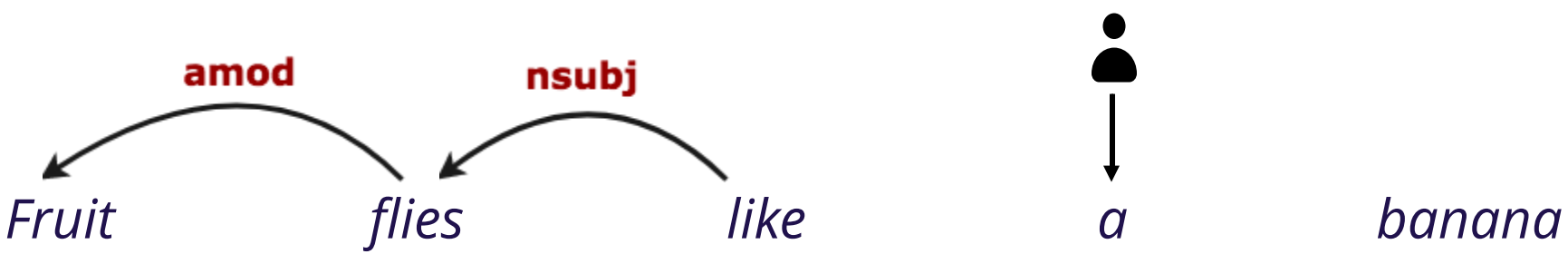
# States, Actions

S: shift

L: left-arc

R:right-arc



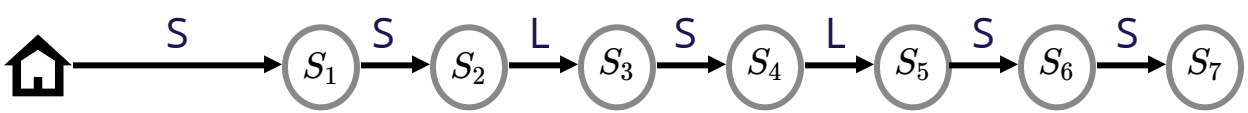



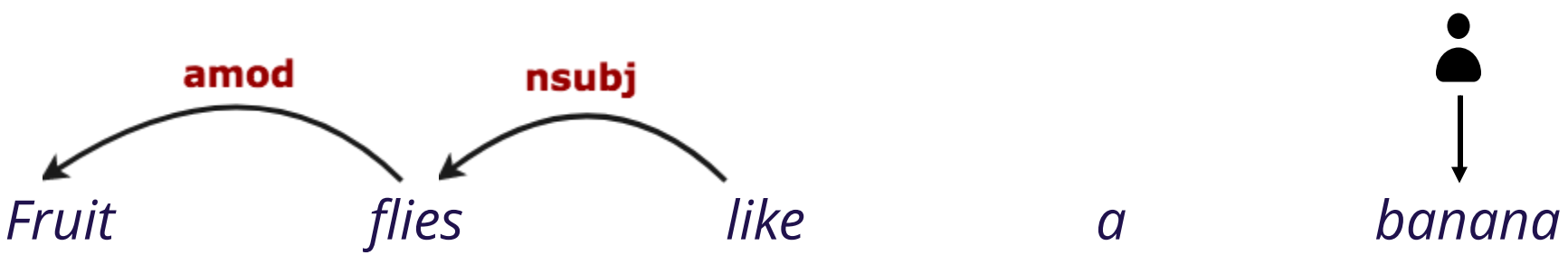
# States, Actions

S: shift

L: left-arc

R:right-arc



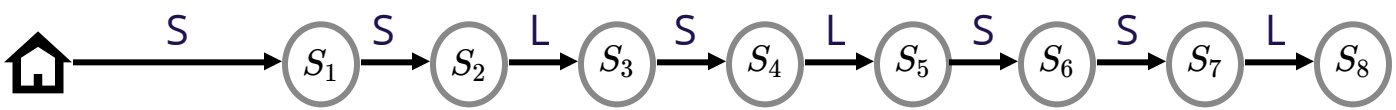



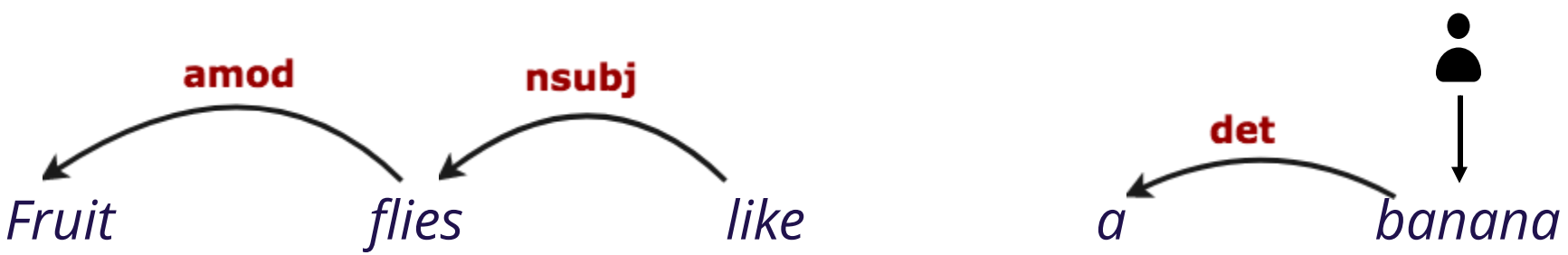
# States, Actions

S: shift

L: left-arc

R:right-arc



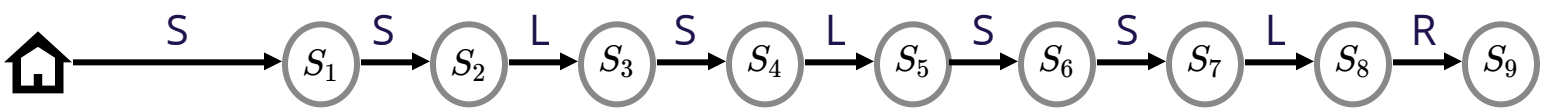



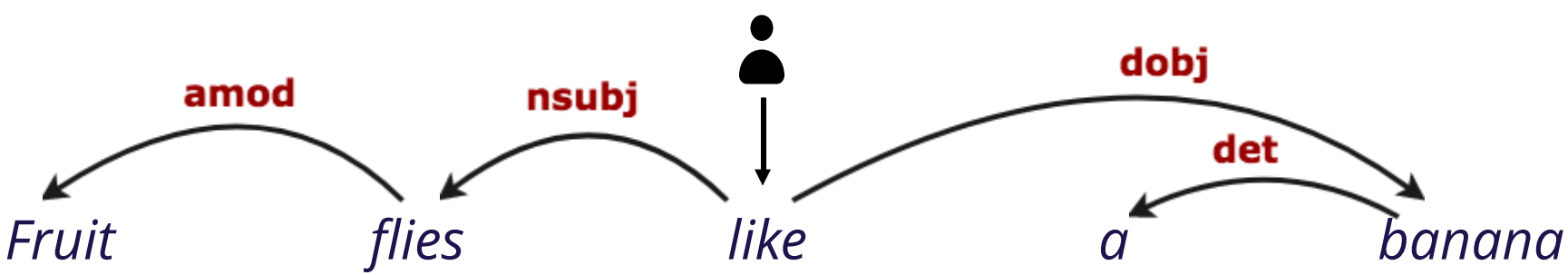
# States, Actions

S: shift

L: left-arc

R:right-arc



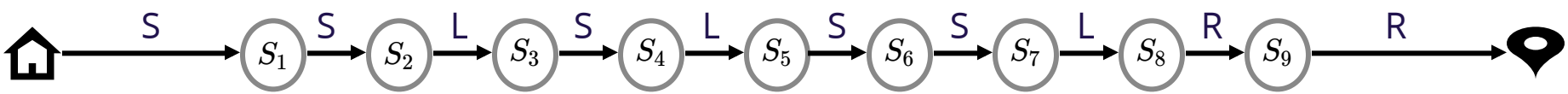



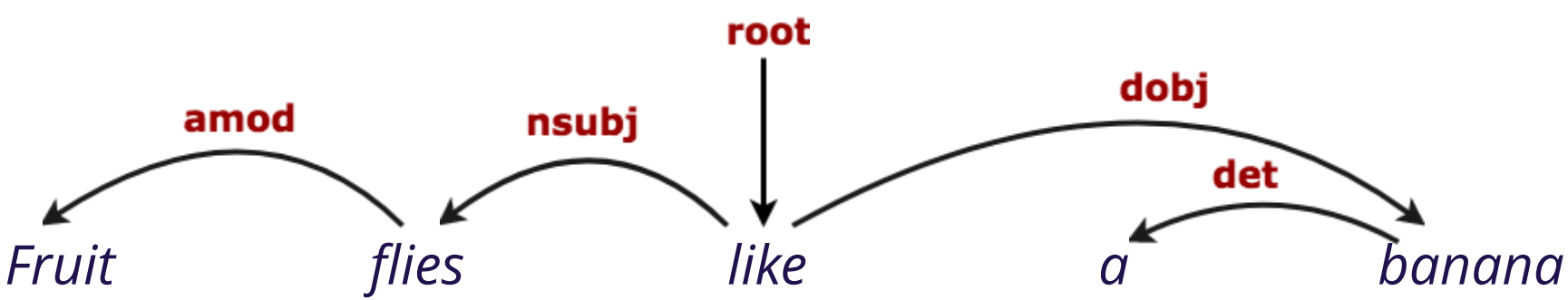
# States, Actions

S: shift

L: left-arc

R:right-arc



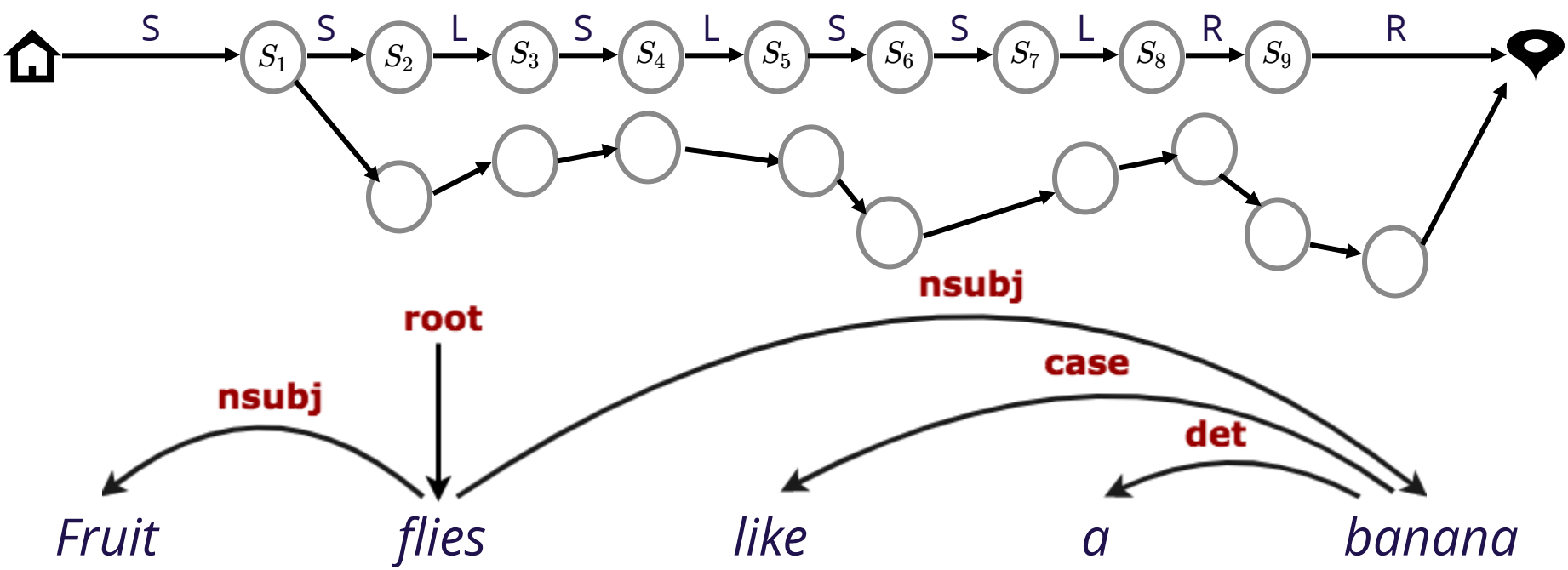



# States, Actions

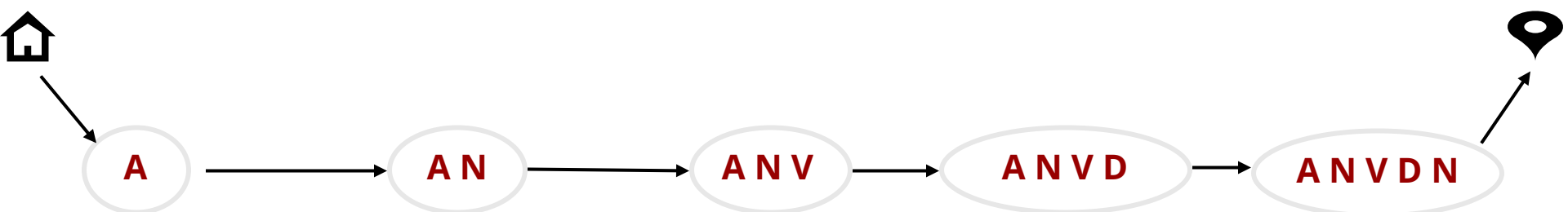
S: shift

L: left-arc

R:right-arc

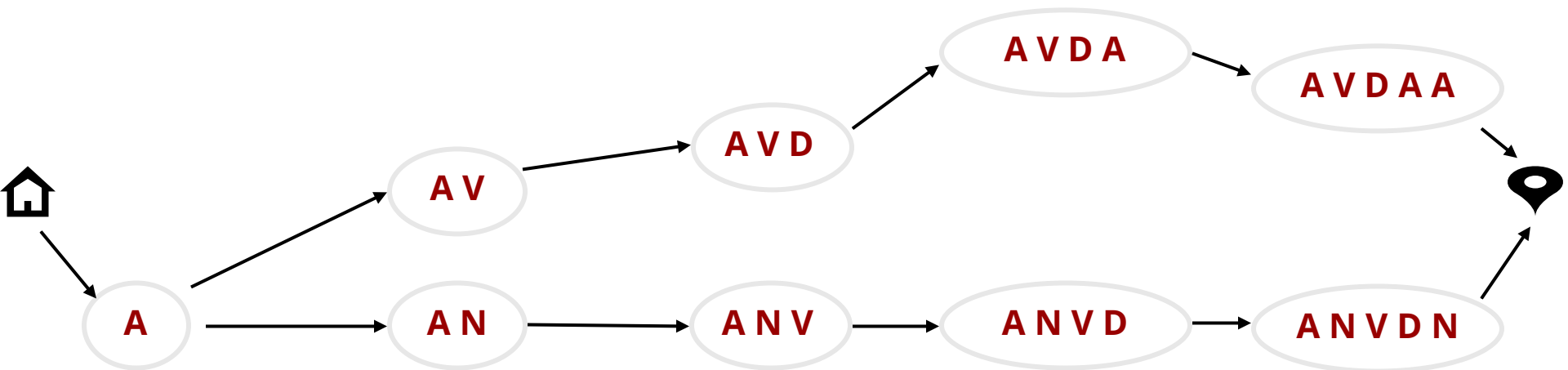


# States, Actions, Paths




*Fruit* *flies* *like* *a* *banana*

# States, Actions, Paths

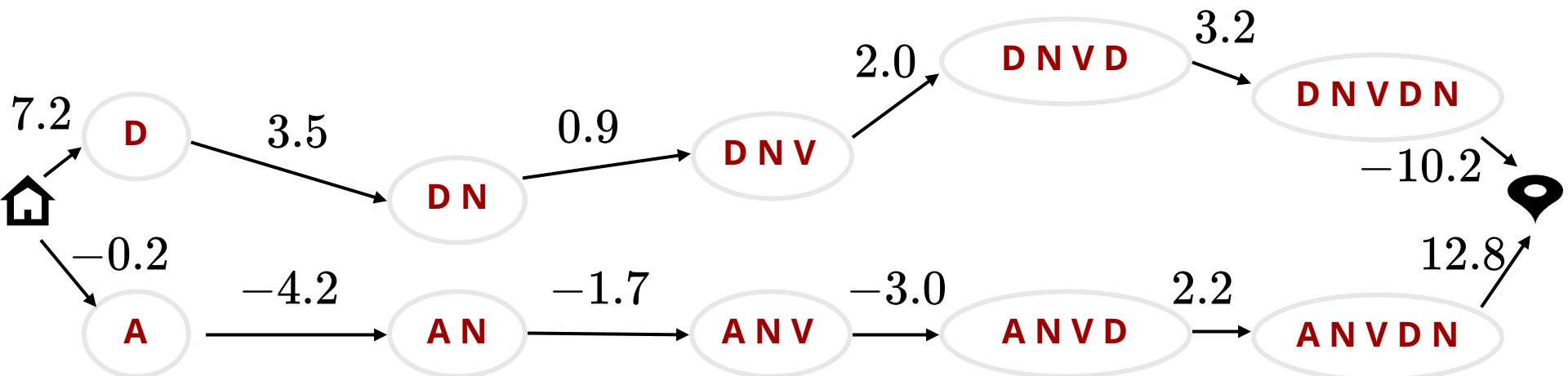



*Fruit* *flies* *like* *a* *banana*

# Score of a Path

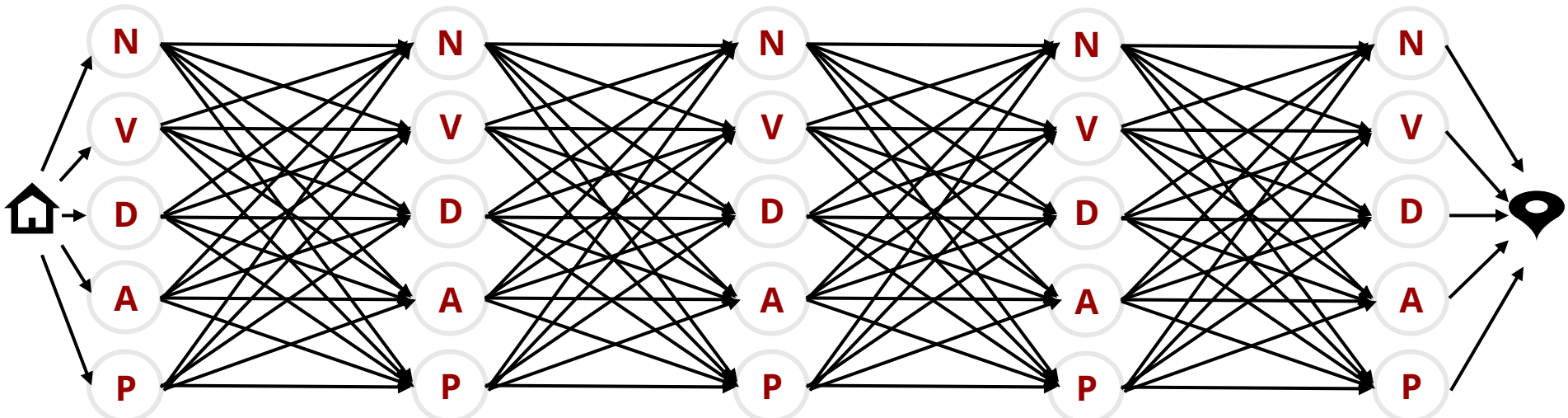
$$S_{\mathbf{w}}(p) = \sum_{e \in p} s_{\mathbf{w}}(e)$$

$$s_{\mathbf{w}}(e) = \mathbf{w} \cdot \mathbf{f}(e)$$




*Fruit* *flies* *like* *a* *banana*

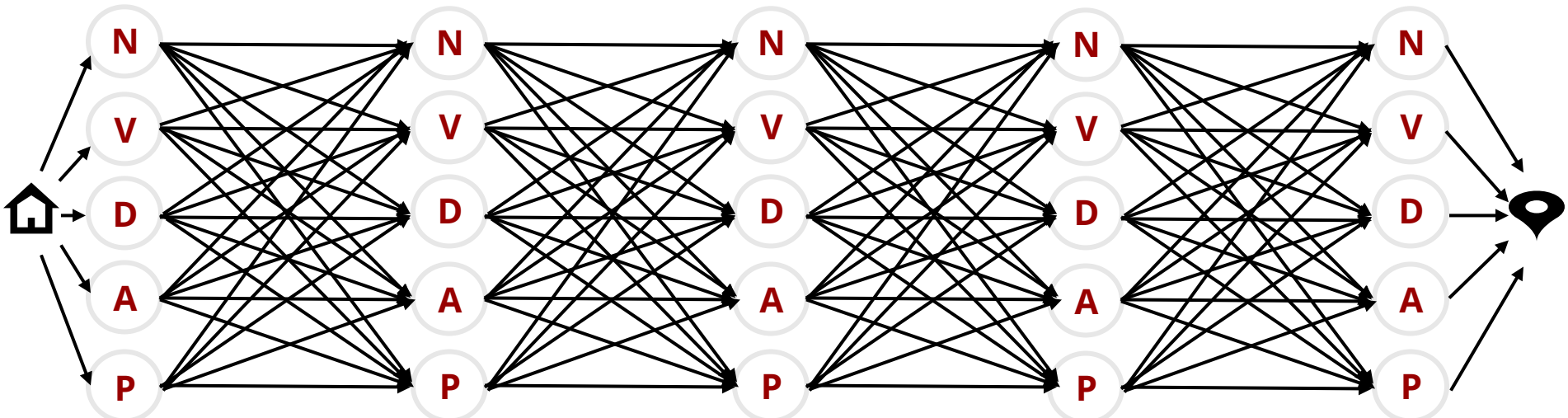
# Compact Search Graph




*Fruit* *flies* *like* *a* *banana*

# Structured Prediction

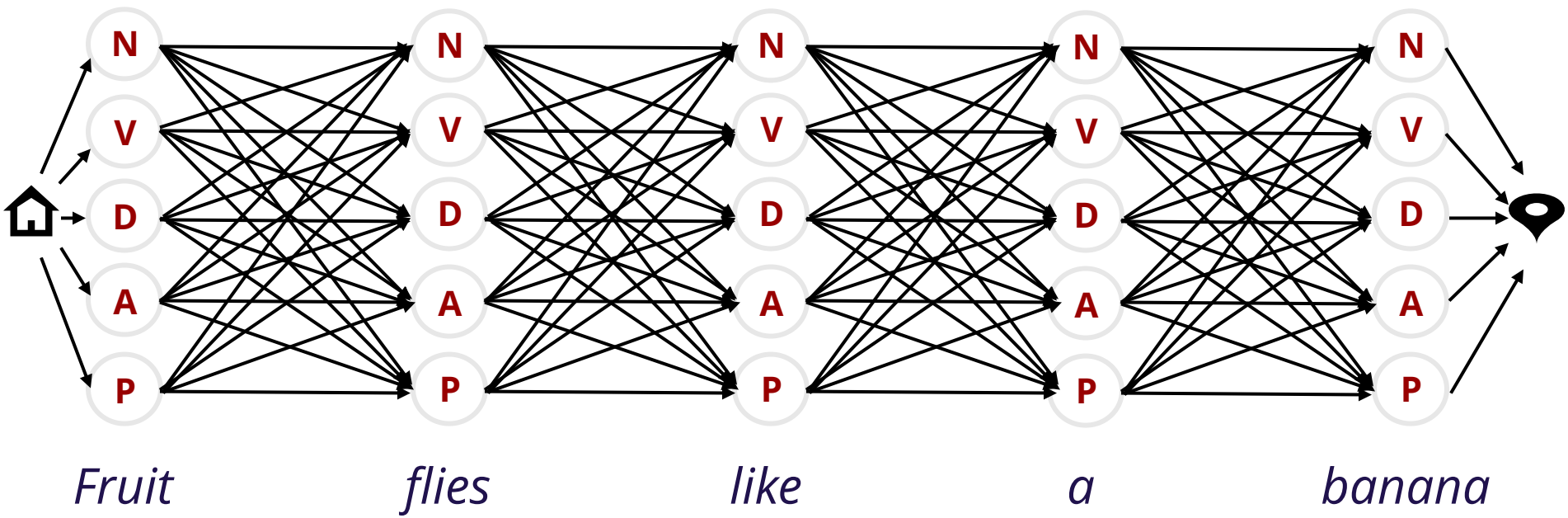
| Given | Find |
|---|---|
| $x, \mathbf{w}$ | $y$ |




*Fruit* *flies* *like* *a* *banana*

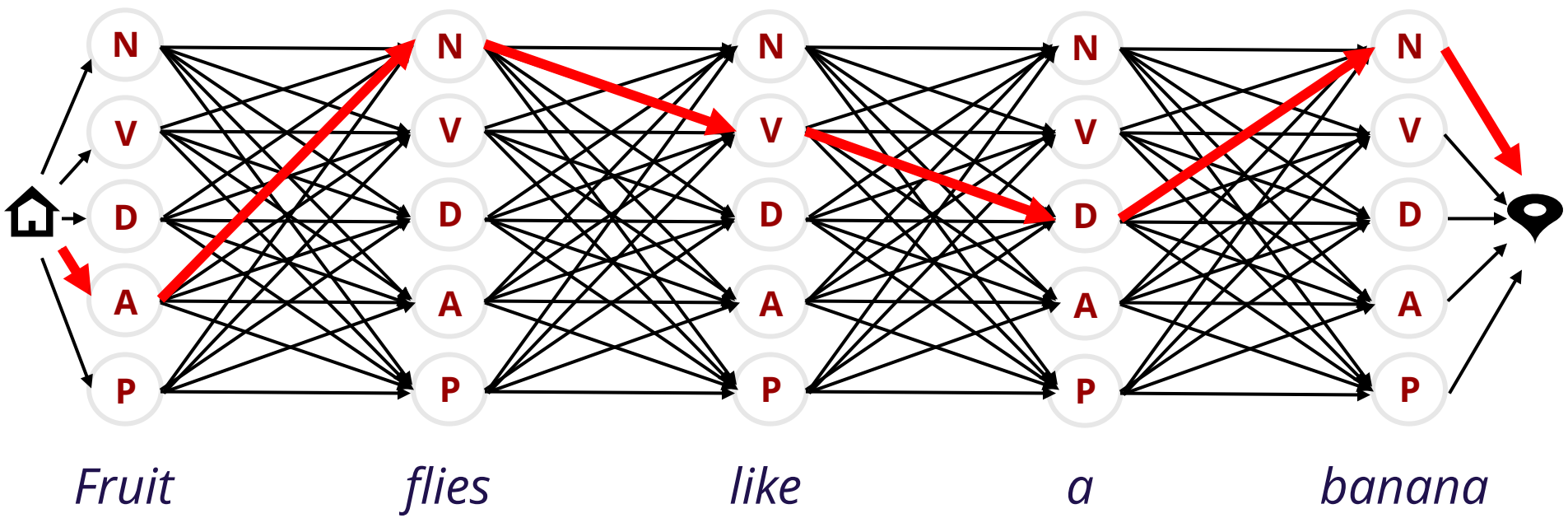
# The Search Problem

$$\arg\max_y \mathbf{w} \cdot \mathbf{f}(x, y)$$

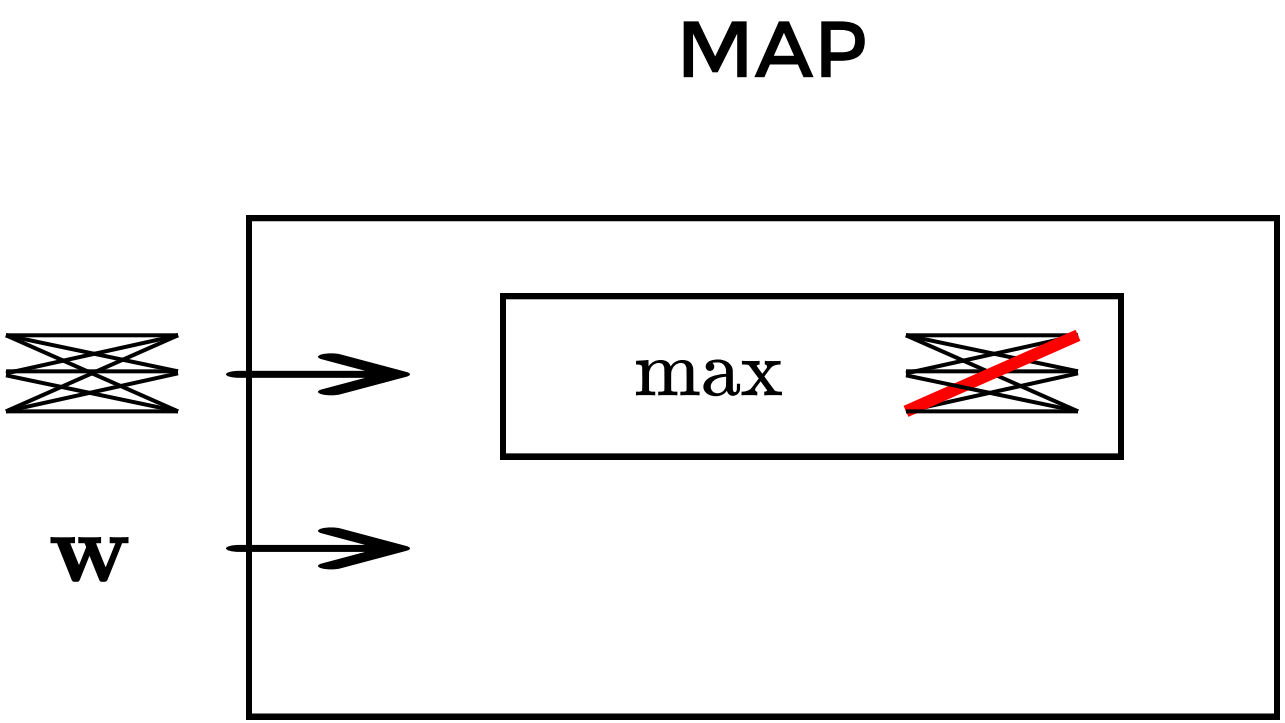


# MAP Inference

$$\arg\max_y \mathbf{w} \cdot \mathbf{f}(x, y)$$



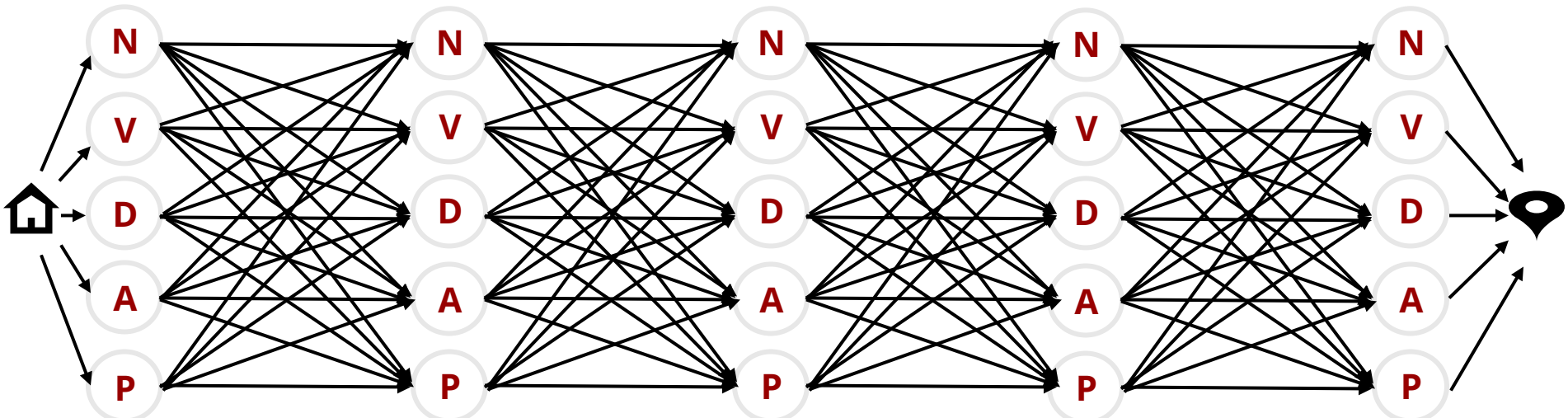
# Inference




To find the optimal structure, just construct the graph, and tell the inference module the current weights, and specify the inference algorithm (in this case MAP)

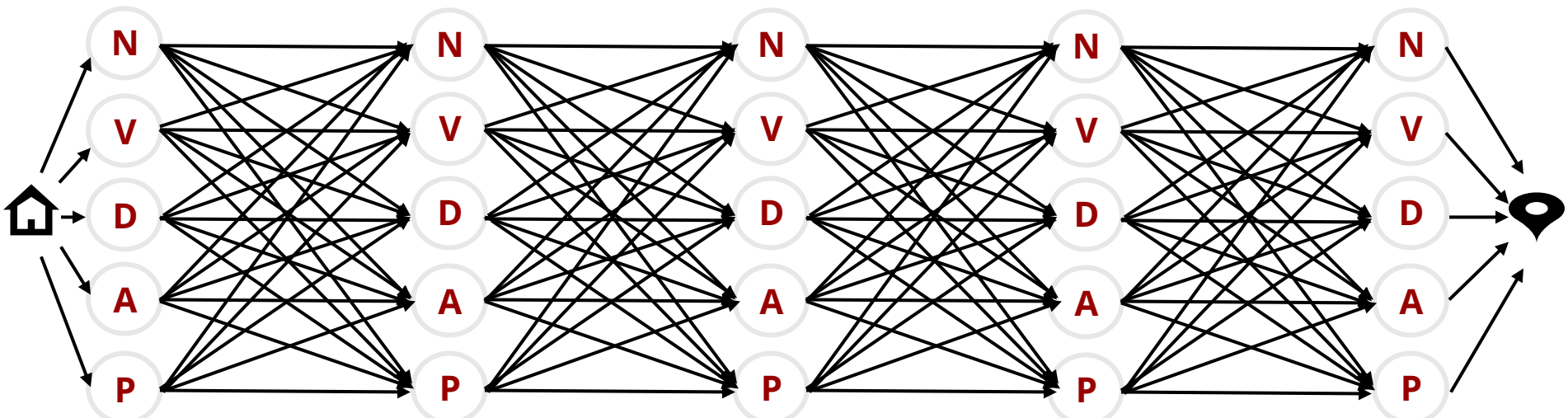
# Learning

| Given | Find |
|---|---|
| $(x_i, y_i)$ | $\mathbf{w}$ |



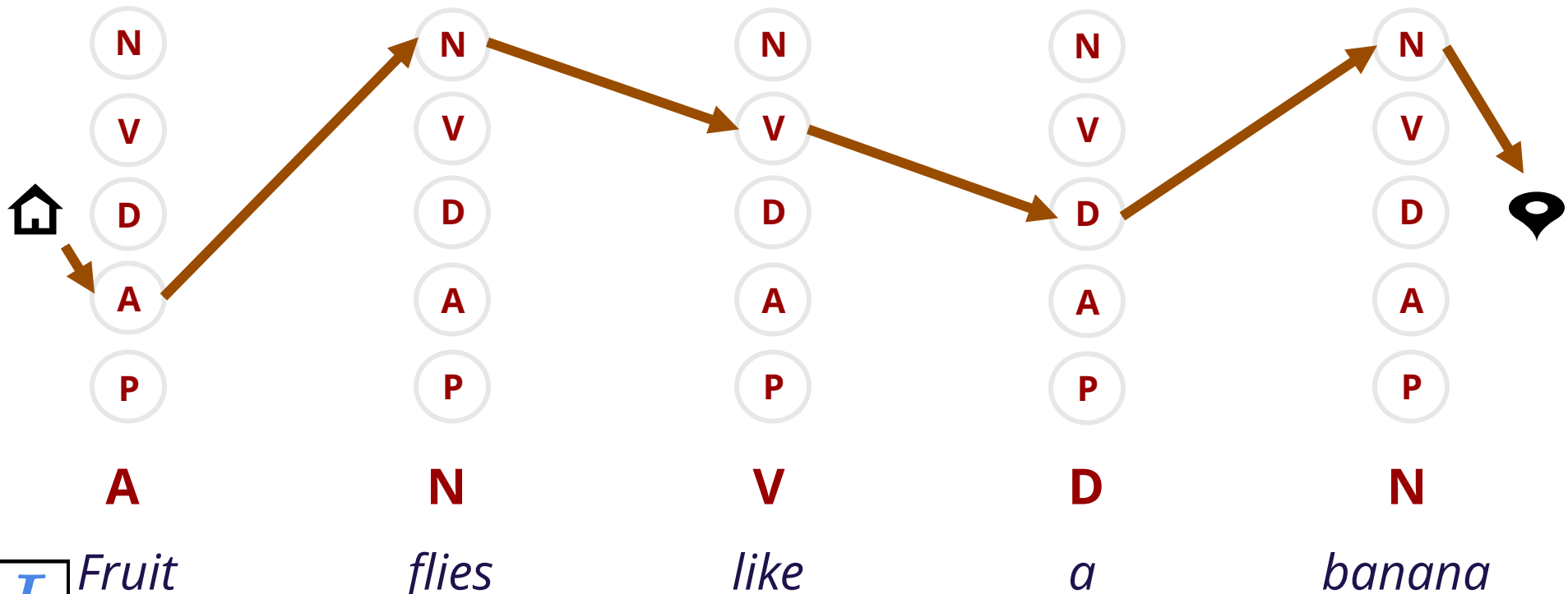
# Structured Perceptron

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \left( \mathbf{w} \cdot \mathbf{f}(x_i, y) \right) \right)$$



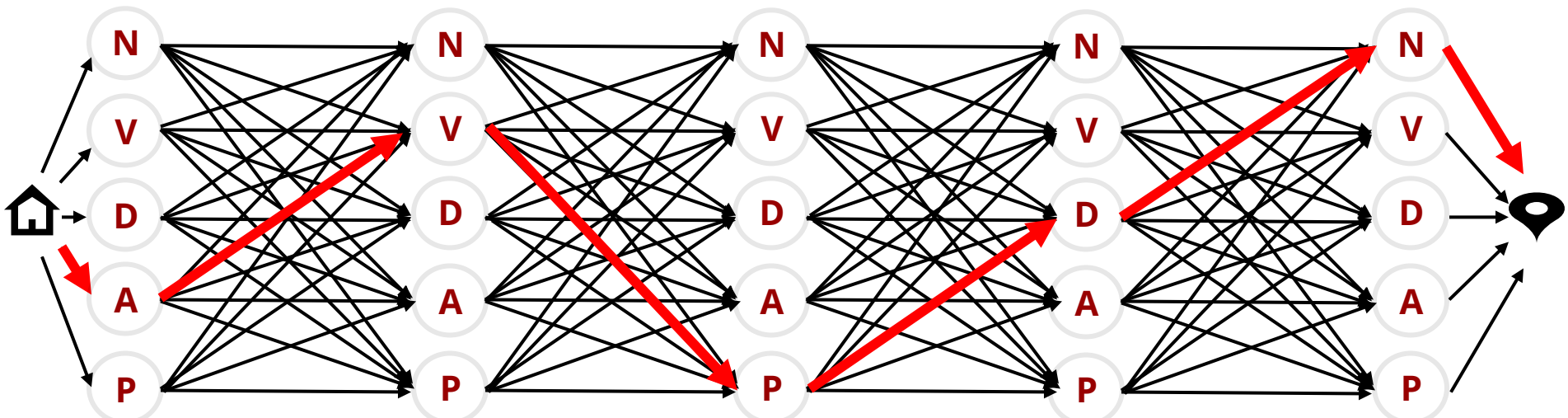
# Structured Perceptron

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \left( \mathbf{w} \cdot \mathbf{f}(x_i, y) \right) \right)$$



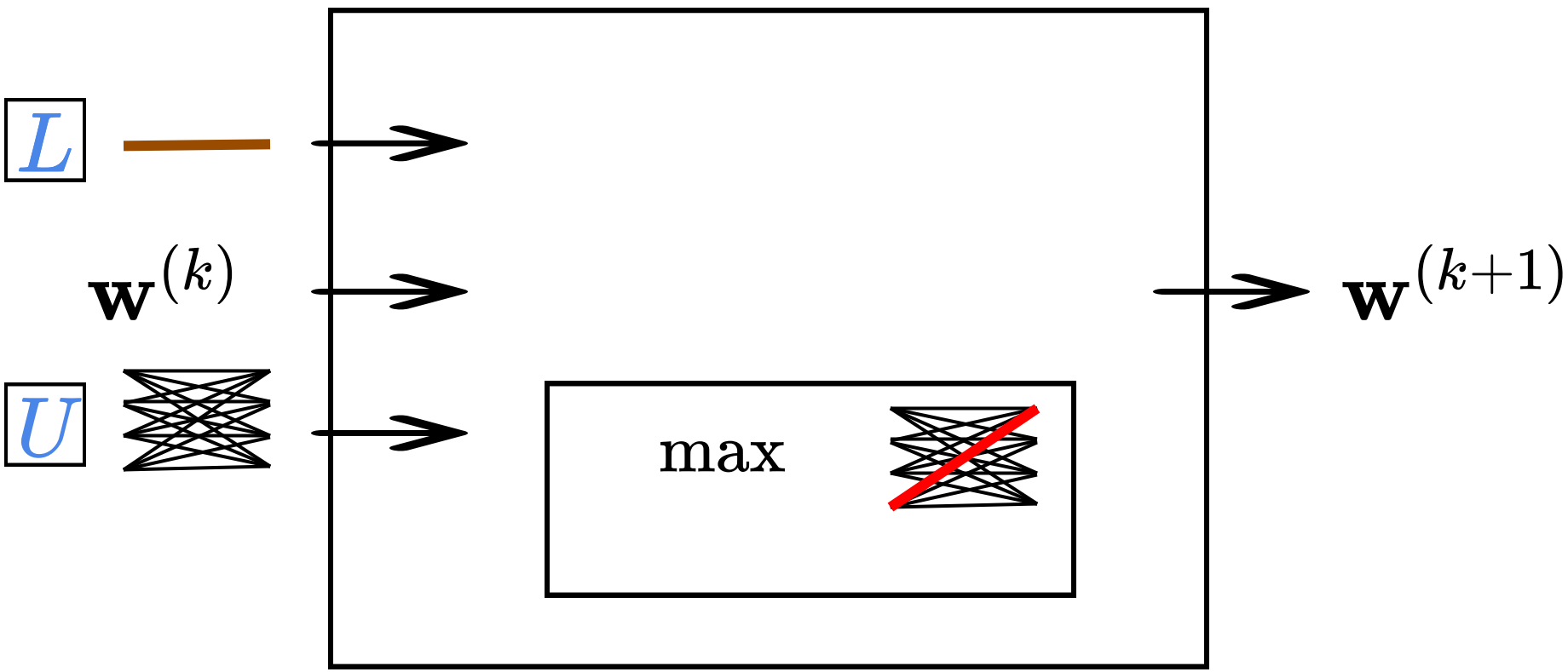
# Structured Perceptron

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \left( \mathbf{w} \cdot \mathbf{f}(x_i, y) \right) \right)$$



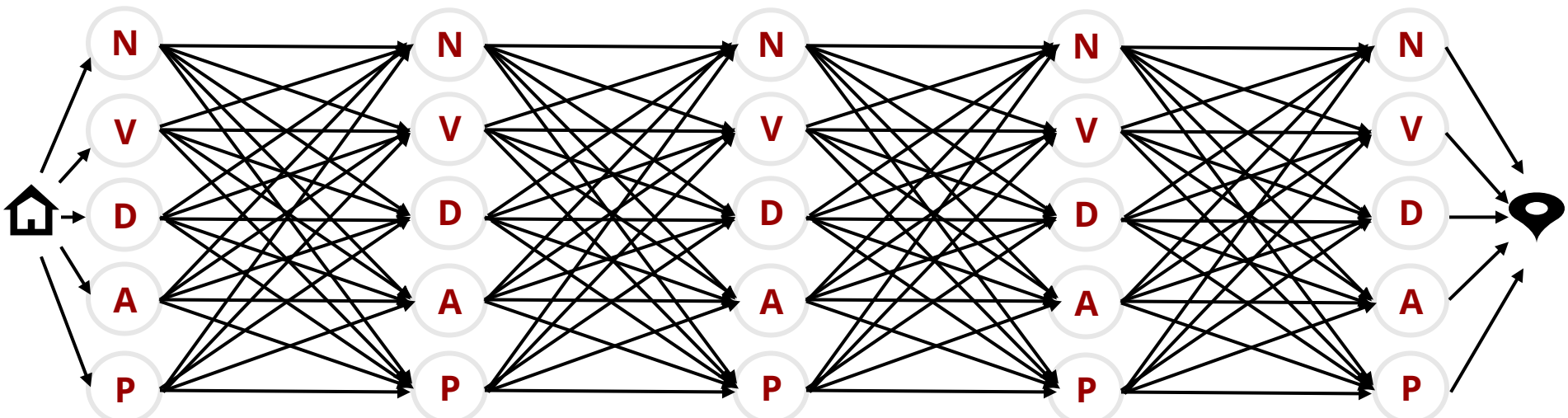
# Learning

## Structured Perceptron



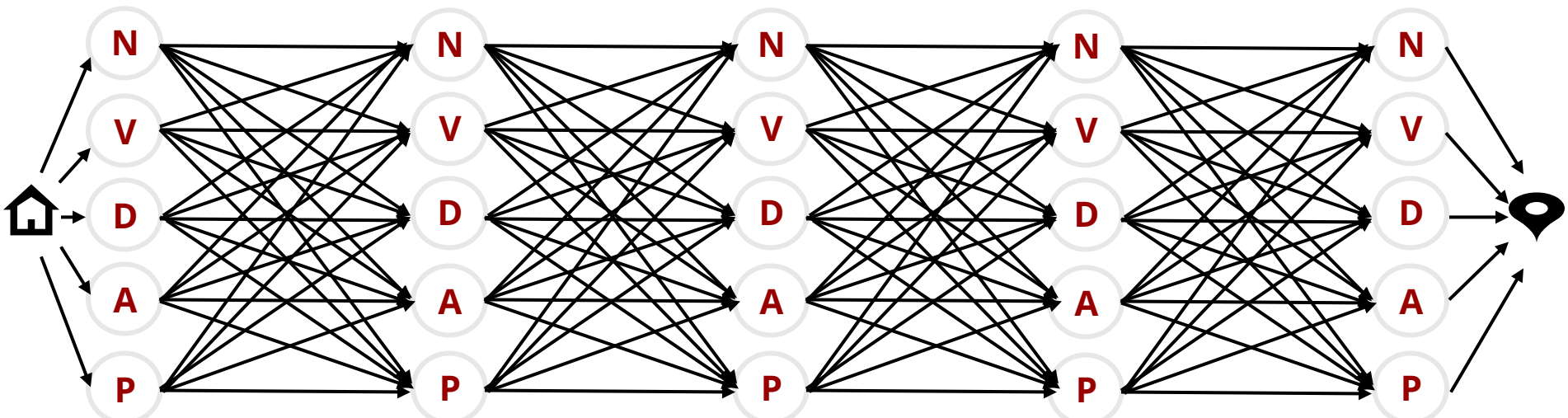
# A Probabilistic View

$$\max_{\mathbf{w}} \sum_i \log p(y_i|x_i)$$



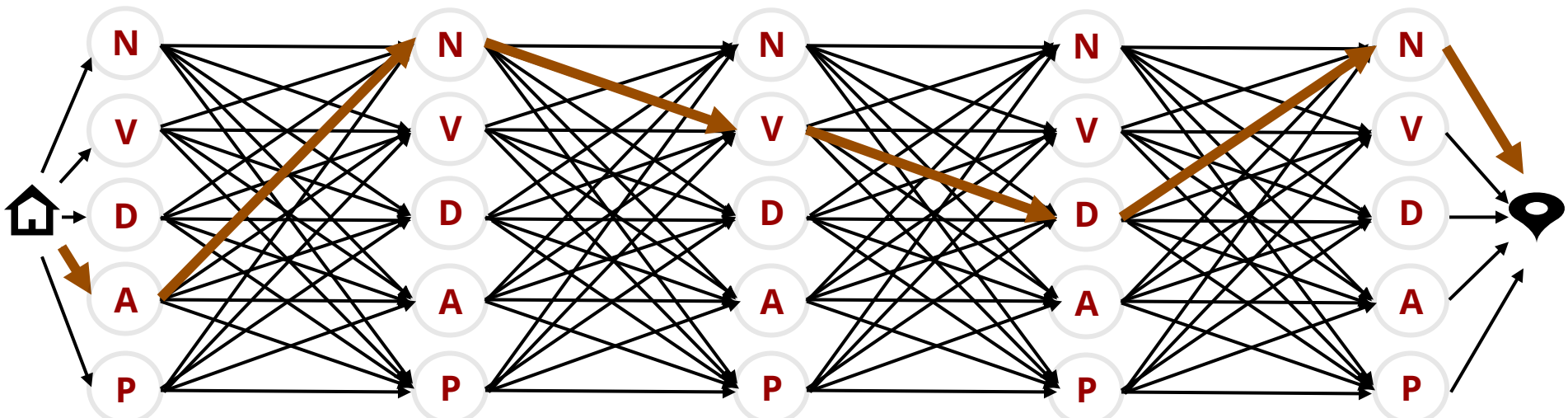
# Conditional Random Fields

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$



# Conditional Random Fields

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

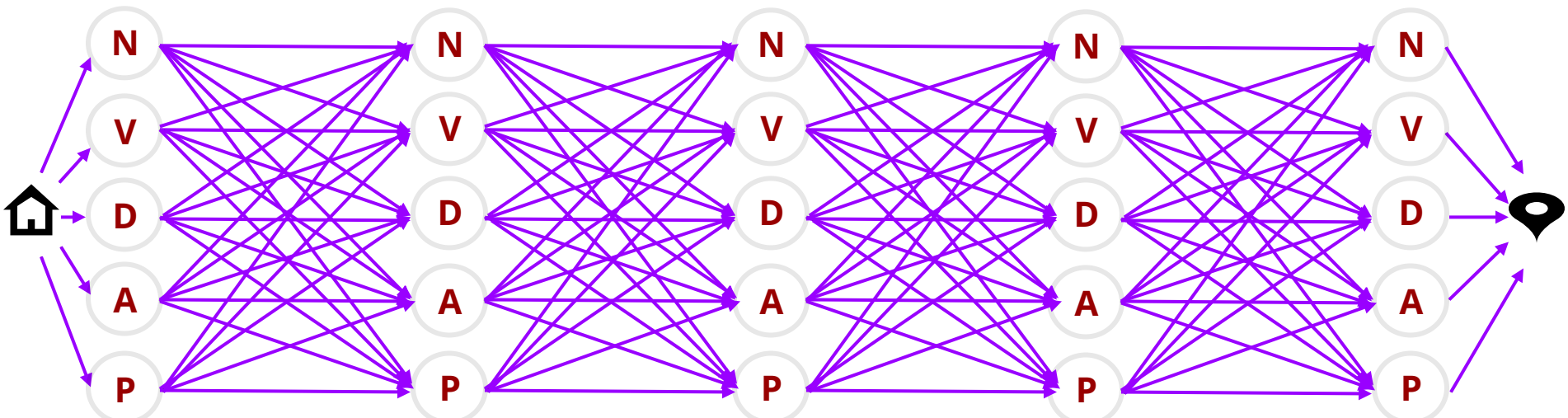



A N V D N

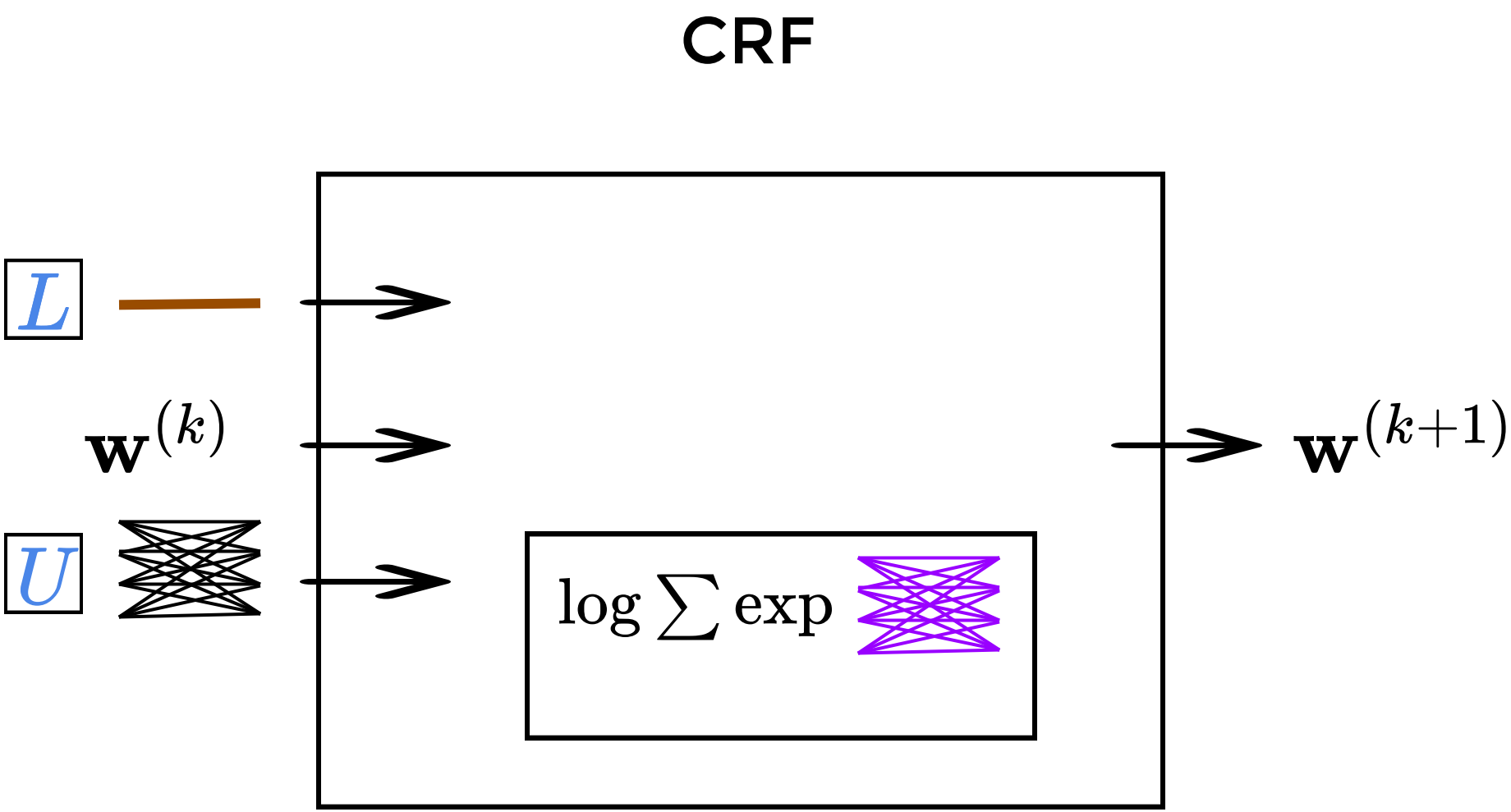
*Fruit flies like a banana*

# Conditional Random Fields

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$



# Learning

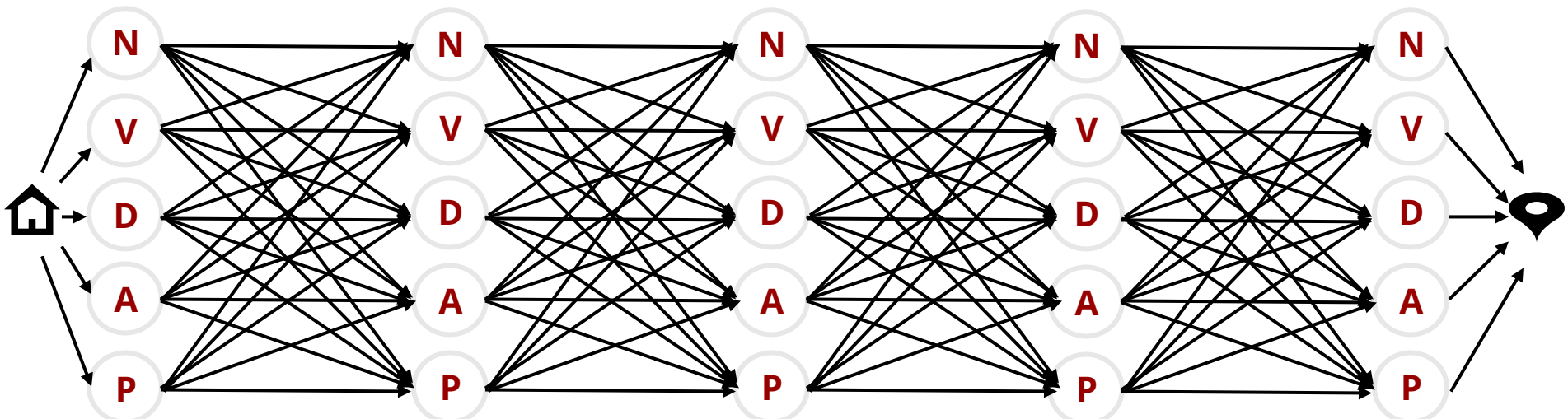


# Max-Margin

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \left( \mathbf{w} \cdot \mathbf{f}(x_i, y) \right) \right)$$

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \left( \Delta(y_i, y) + \mathbf{w} \cdot \mathbf{f}(x_i, y) \right) \right)$$

$$\Delta(\mathbf{D}, \mathbf{P}) = 1, \Delta(\mathbf{N}, \mathbf{N}) = 0, \Delta(\mathbf{N}, \mathbf{V}) = 10$$

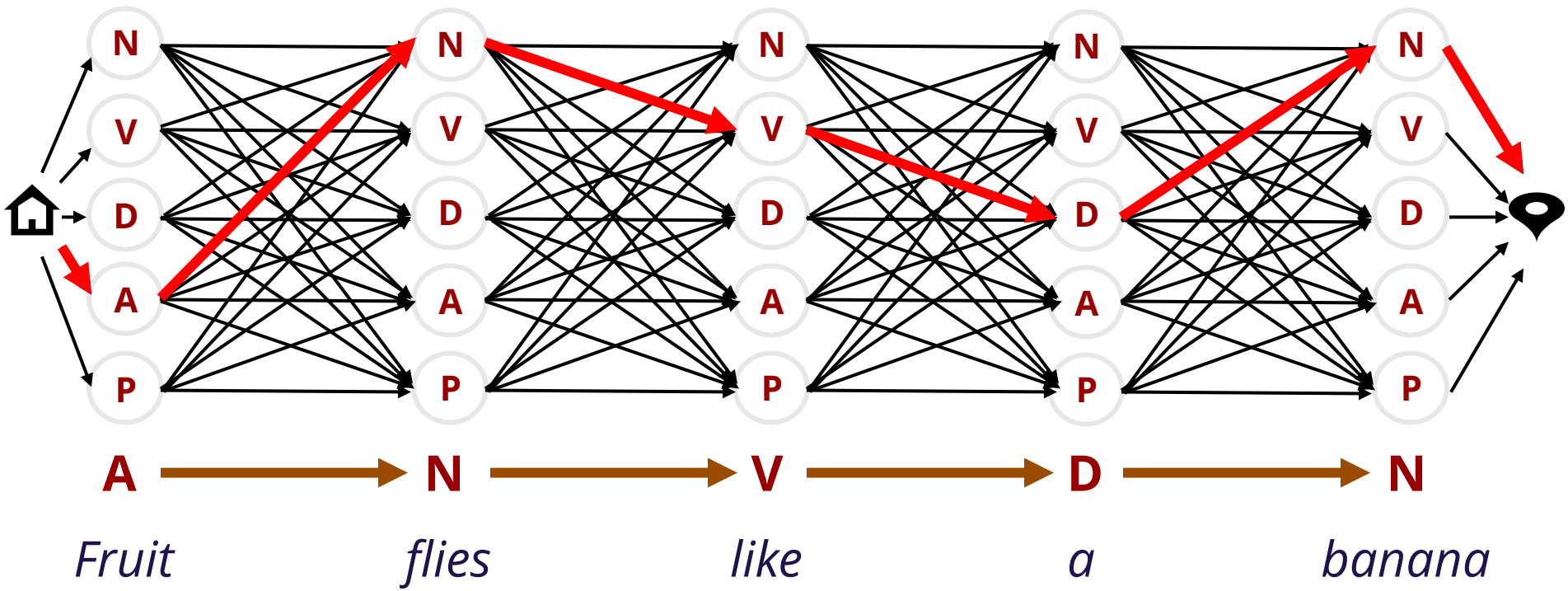


# Decode with Oracle

$$\min_{\mathbf{w}} \sum_i \Big( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \big( \mathbf{w} \cdot \mathbf{f}(x_i, y) \big) \Big)$$

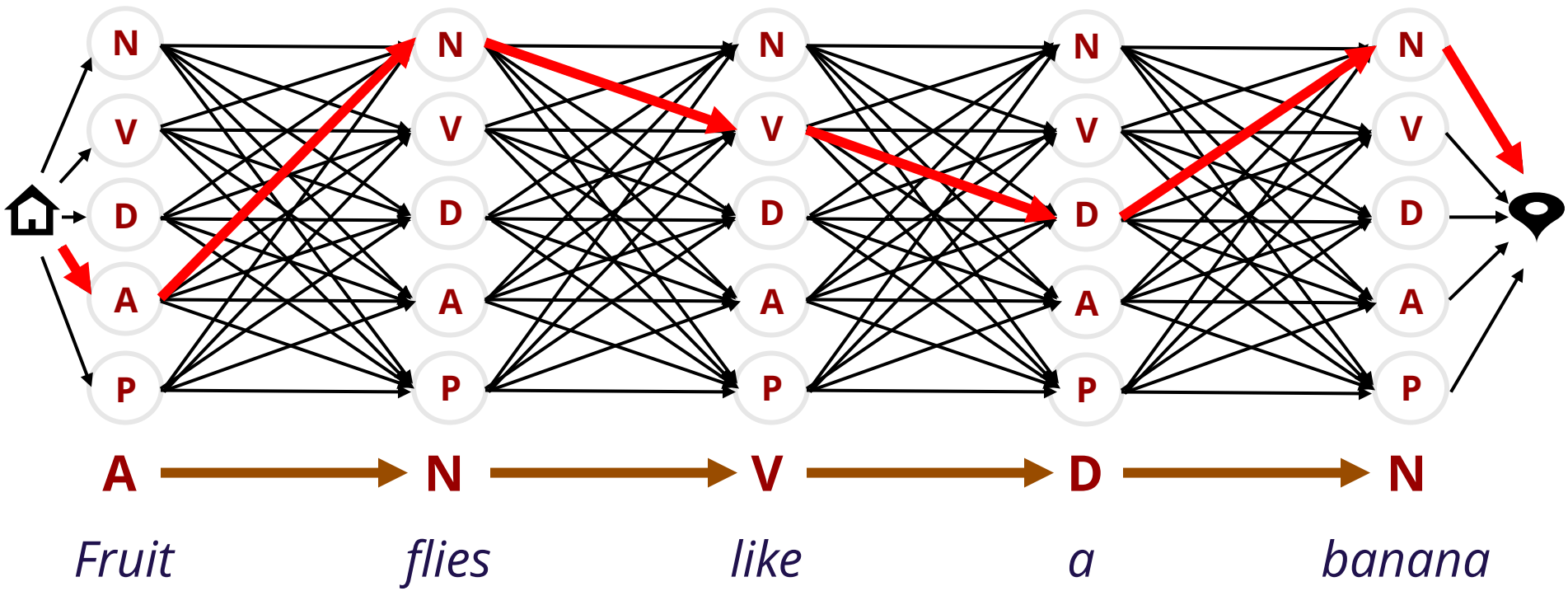
$$\min_{\mathbf{w}} \sum_i \Big( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \big( \Delta(y_i, y) + \mathbf{w} \cdot \mathbf{f}(x_i, y) \big) \Big)$$

$$\Delta(\mathbf{D}, \mathbf{P}) = 1, \Delta(\mathbf{N}, \mathbf{N}) = 0, \Delta(\mathbf{N}, \mathbf{V}) = 10$$



# Structural SVM

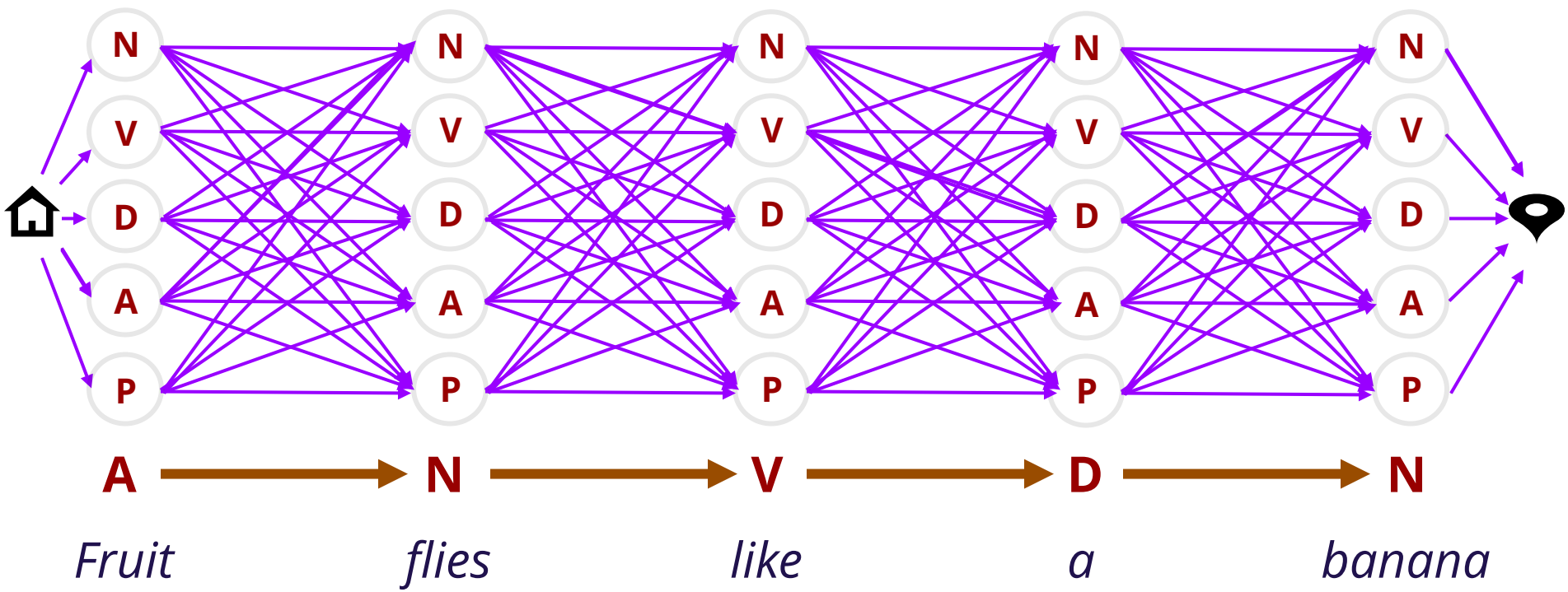
$$\min_{\mathbf{w}} \sum_i \Big( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \max_y \big( \Delta(y_i, y) + \mathbf{w} \cdot \mathbf{f}(x_i, y) \big) \Big)$$



# "Soft"Max-Margin CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

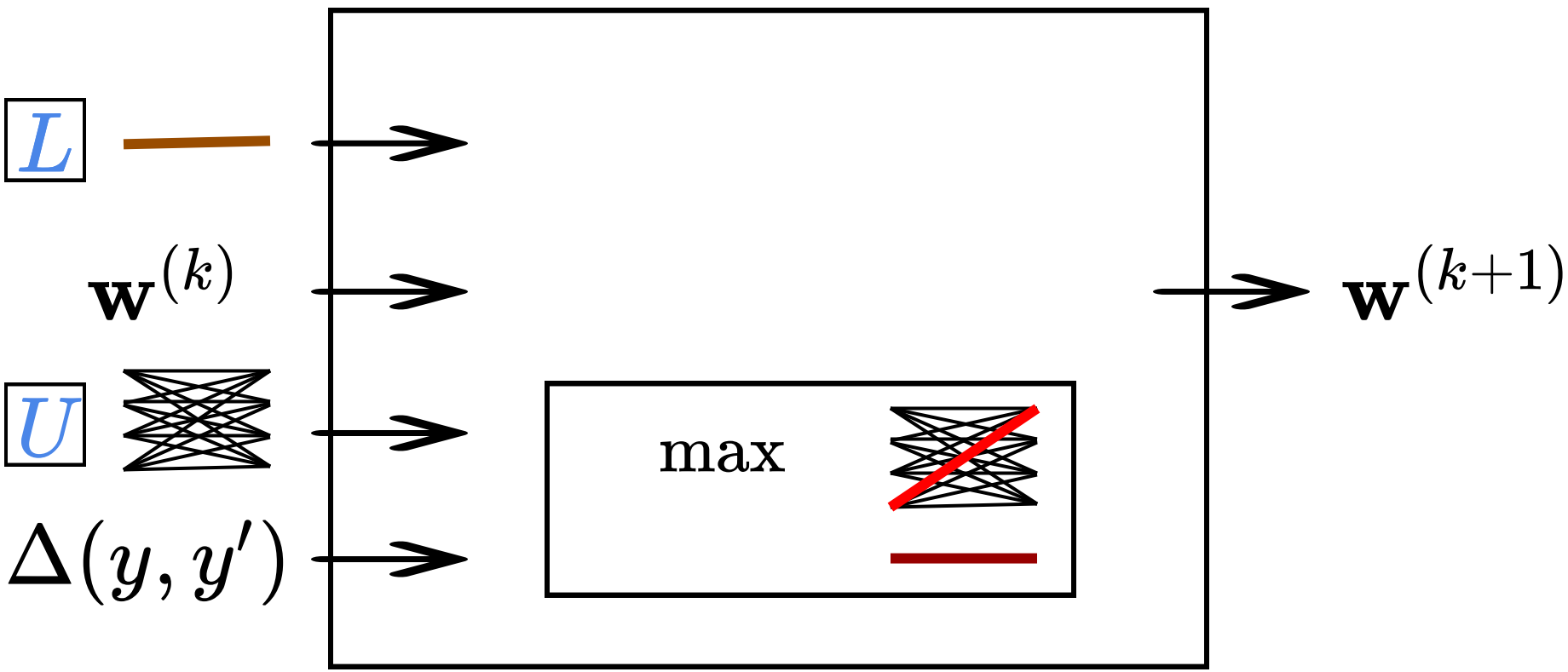
$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\Delta(y_i, y) + \mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$




Gimpel, K., & Smith, N. A. (2010). Softmax-margin CRFs: Training log-linear models with cost functions. In NAACL-HLT.

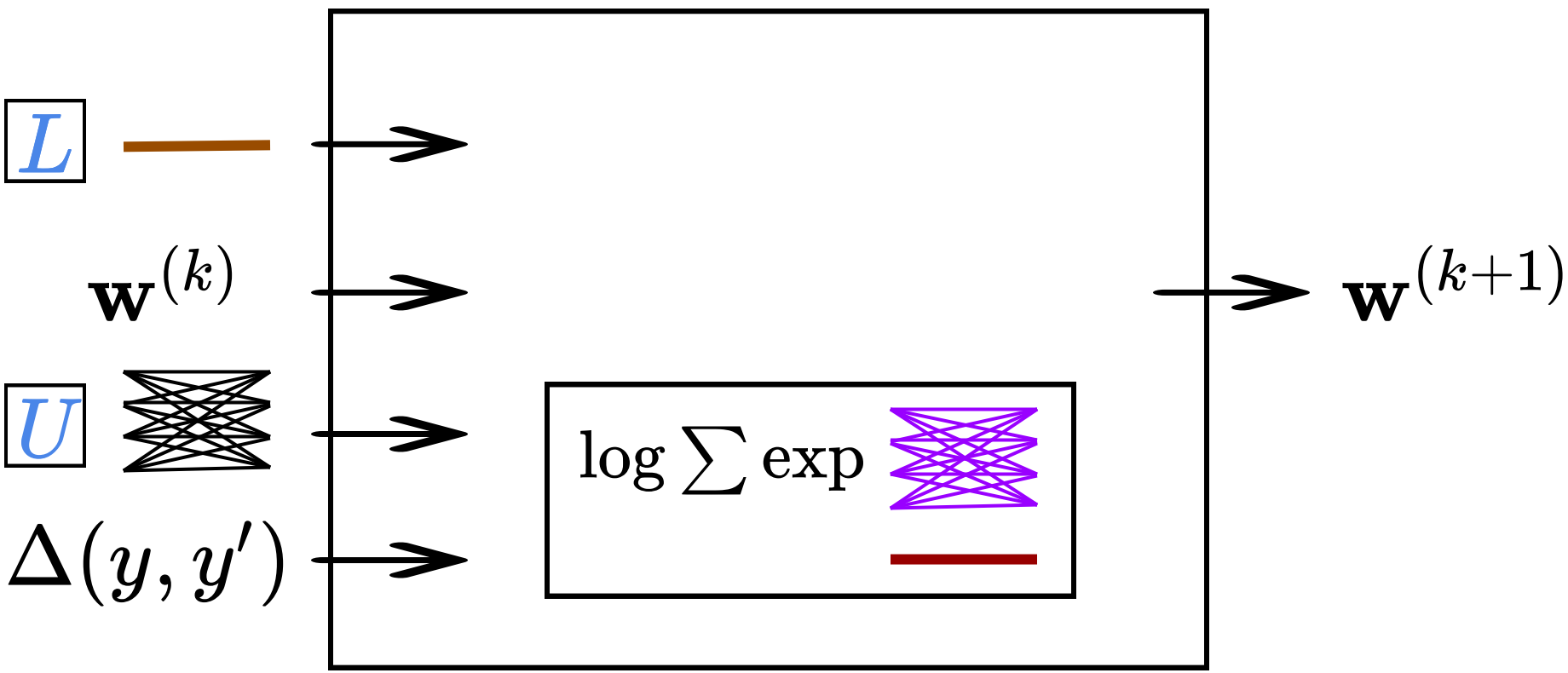
# Learning

## Structured Perceptron, Structural SVM



# Learning

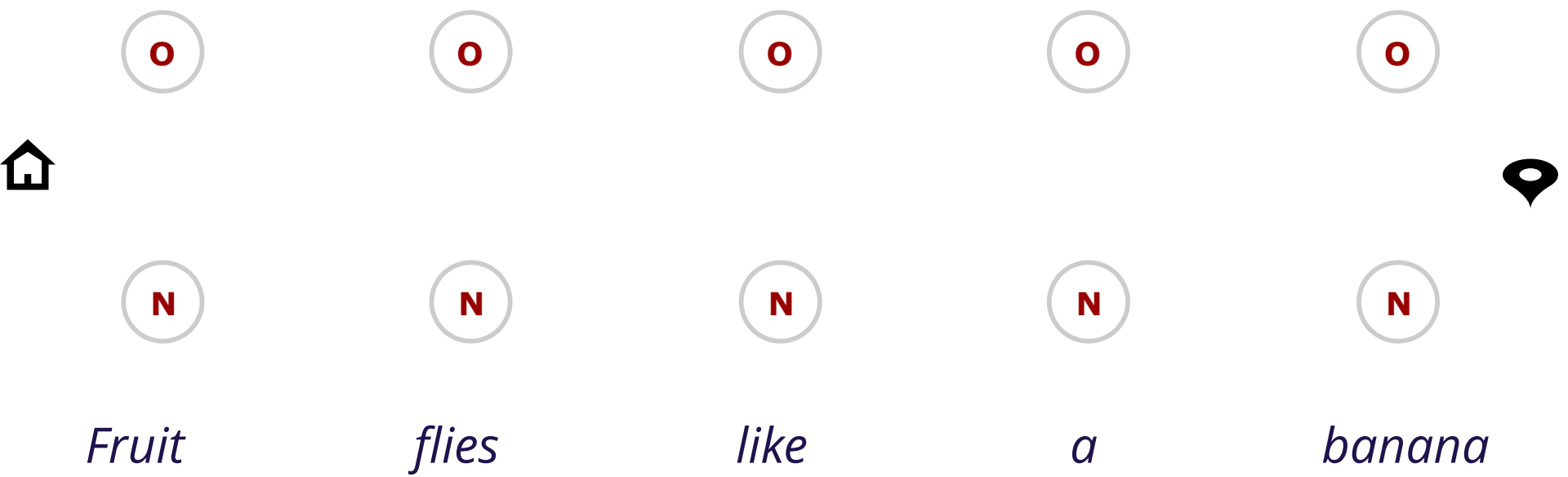
## CRF, "Soft"Max-Margin CRF



# Semi-Markov CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

$$\mathbf{f}_1(x, [y^{0-1} = \mathbf{N}, y^2 = \mathbf{O}])$$

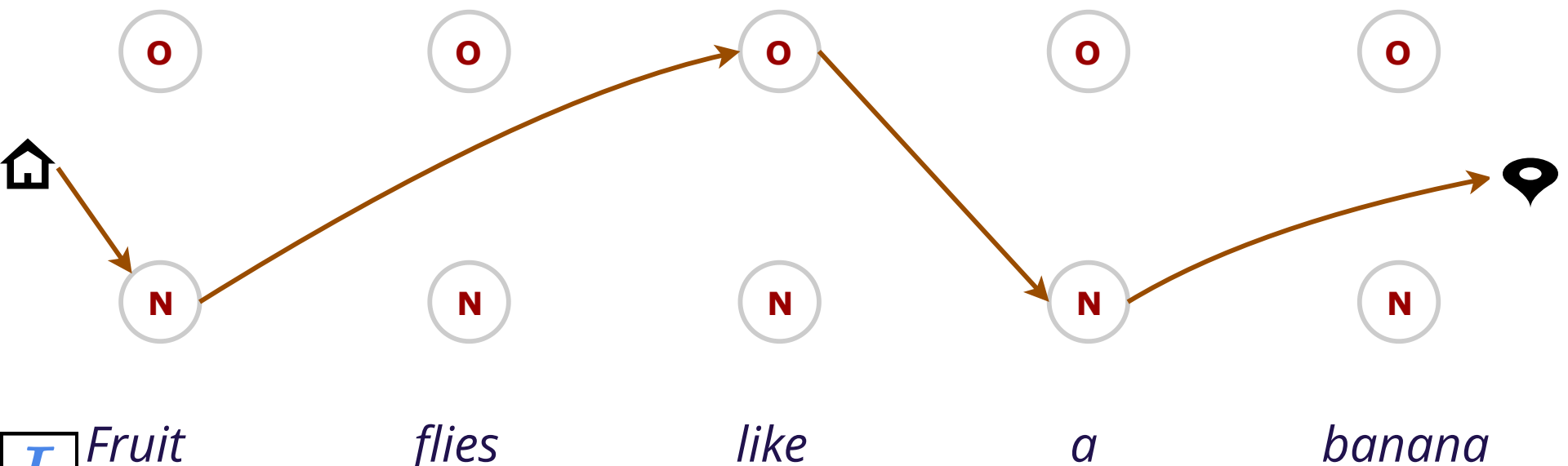



Sarawagi, S., and Cohen, W. W. (2005). Semi-markov conditional random fields for information extraction. In NIPS.

# Semi-Markov CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

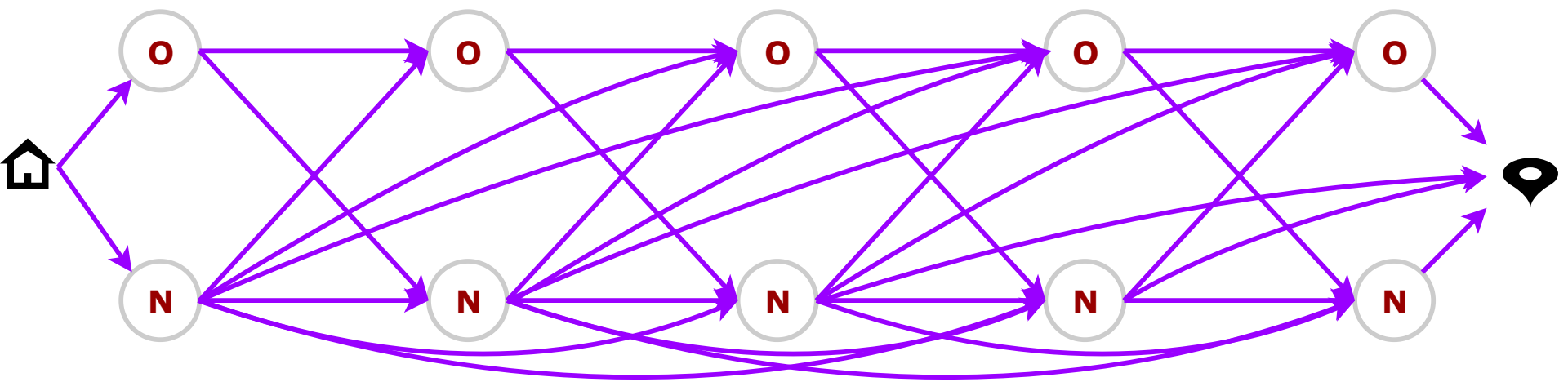
$$\mathbf{f}_1(x, [y^{0-1} = \mathbf{N}, y^2 = \mathbf{O}])$$



# Semi-Markov CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

$$\mathbf{f}_1(x, [y^{0-1} = \mathbf{N}, y^2 = \mathbf{O}])$$

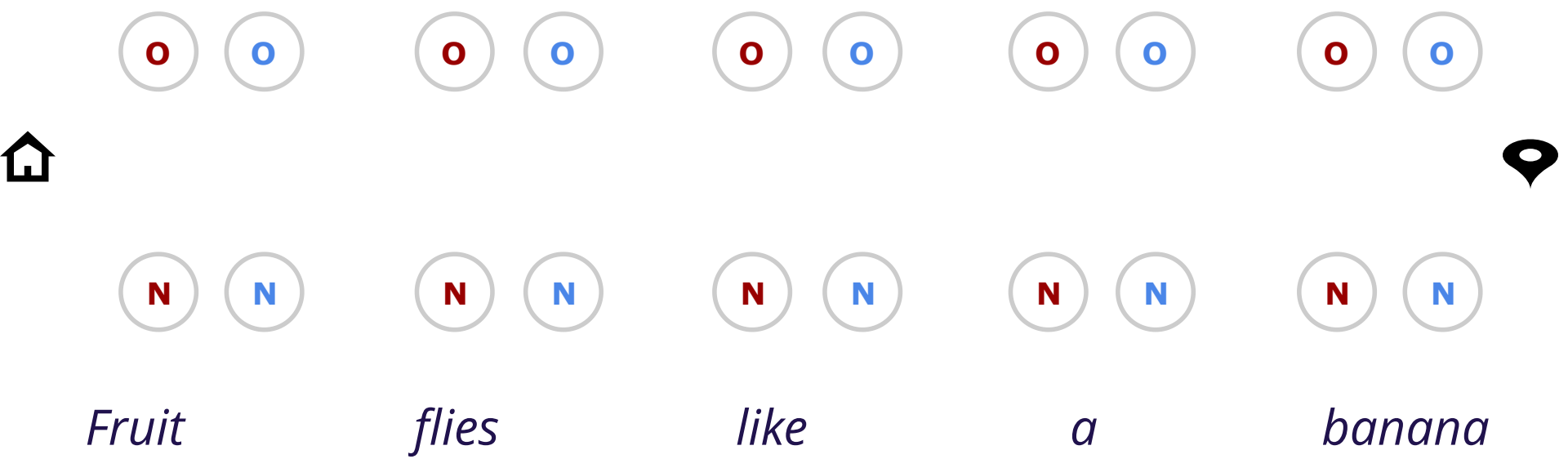



*Fruit* *flies* *like* *a* *banana*

# Weak Semi-Markov CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

$$\mathbf{f}_1(x, [y^{0-1} = \mathbf{N}]) \qquad \mathbf{f}_2(x, [y^1 = \mathbf{N}, y^2 = \mathbf{O}])$$

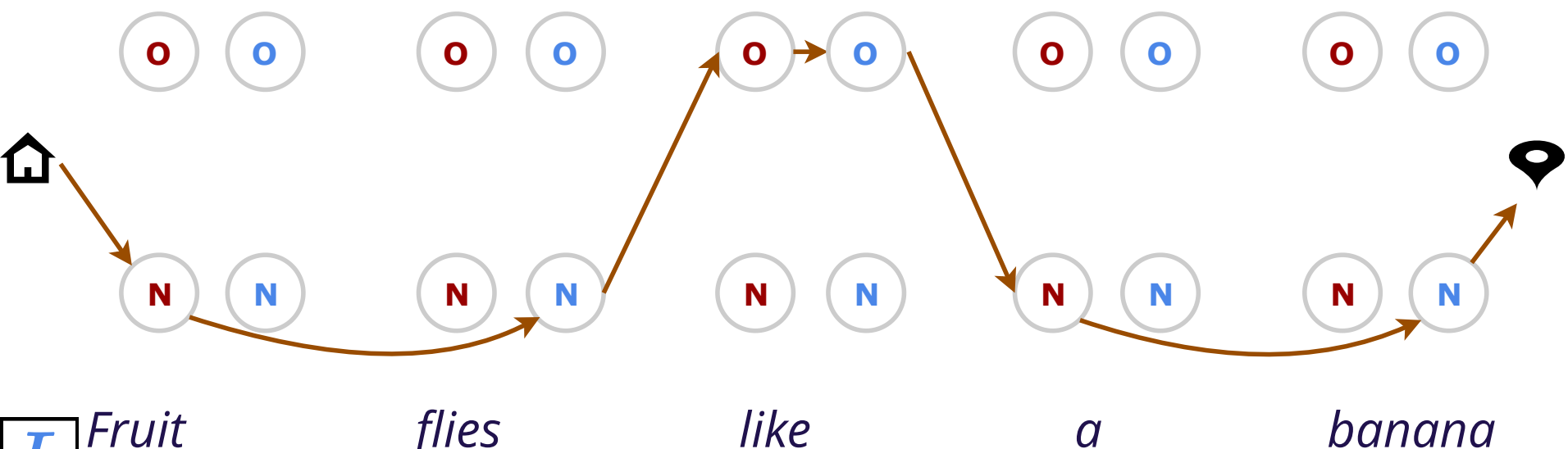



Muis, A. O., & Lu, W. (2016). Weak Semi-Markov CRFs for NP Chunking in Informal Text. In *NAACL-HLT.*

# Weak Semi-Markov CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

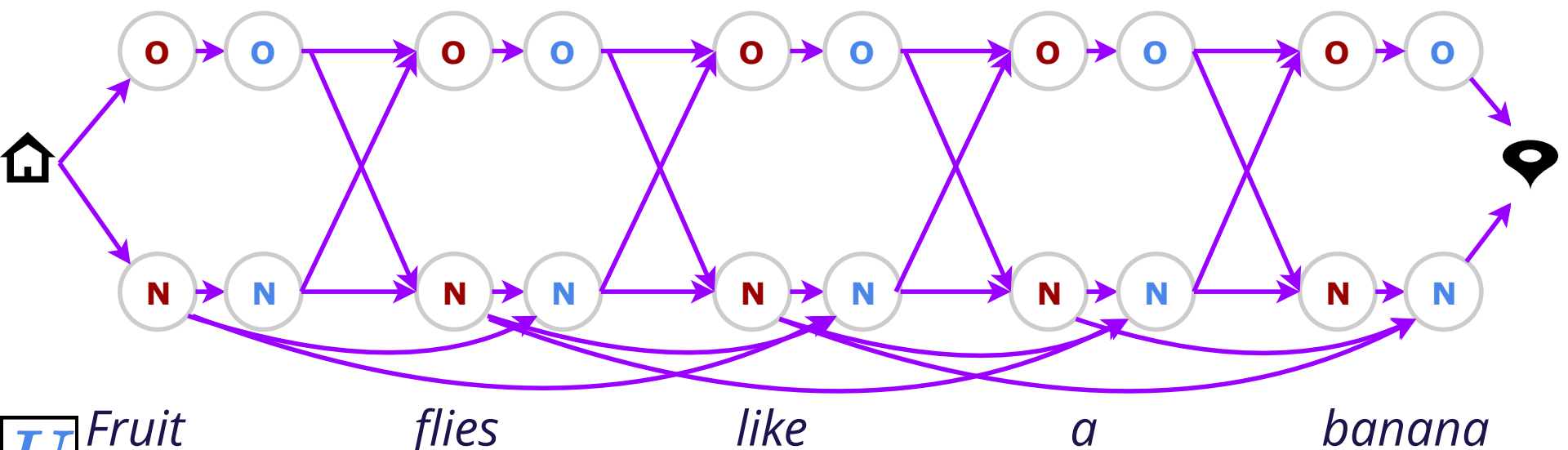
$$\mathbf{f}_1(x, [y^{0-1} = \mathbf{N}]) \qquad \mathbf{f}_2(x, [y^1 = \mathbf{N}, y^2 = \mathbf{O}])$$



# Weak Semi-Markov CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$

$$\mathbf{f}_1(x, [y^{0-1} = \mathbf{N}]) \qquad \mathbf{f}_2(x, [y^1 = \mathbf{N}, y^2 = \mathbf{O}])$$

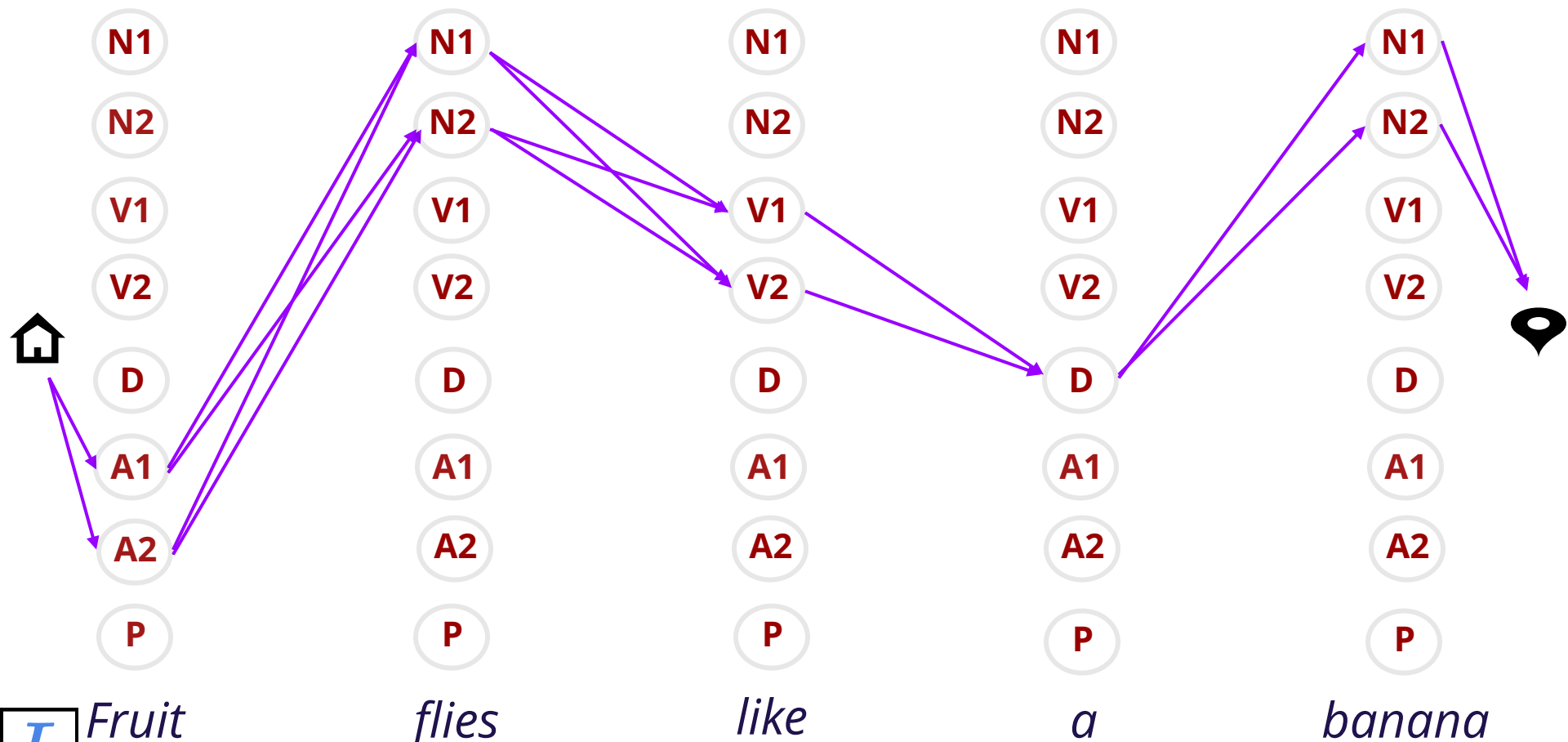


# Latent-Variable CRF

$$\min_{\mathbf{w}} \sum_i \left( -\log \sum_h \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h, y_i)\right) + \log \sum_{h',y} \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h', y)\right)\right)$$

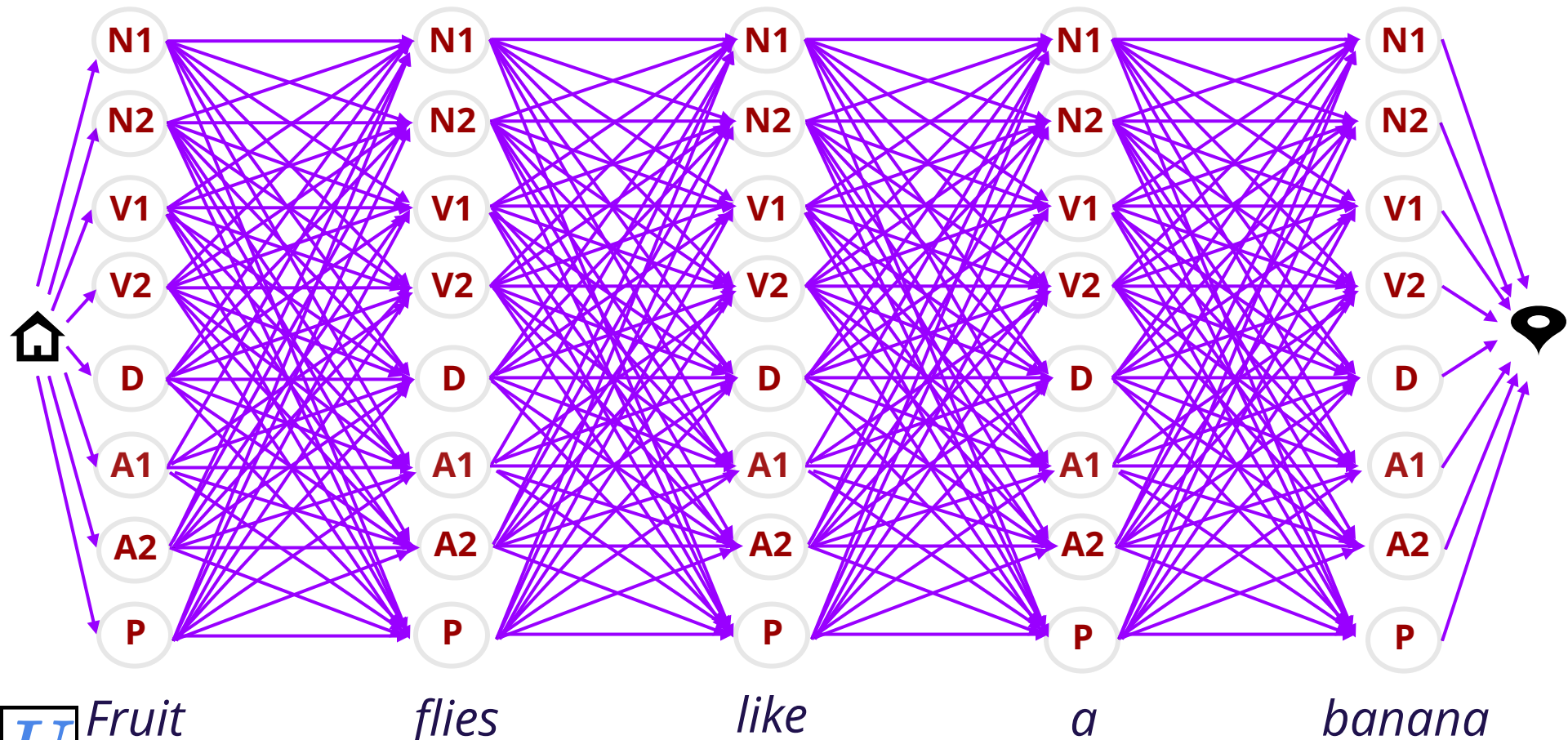
# Latent-Variable CRF

$$\min_{\mathbf{w}} \sum_i \left( -\log \sum_h \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h, y_i)\right) + \log \sum_{h',y} \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h', y)\right)\right)$$



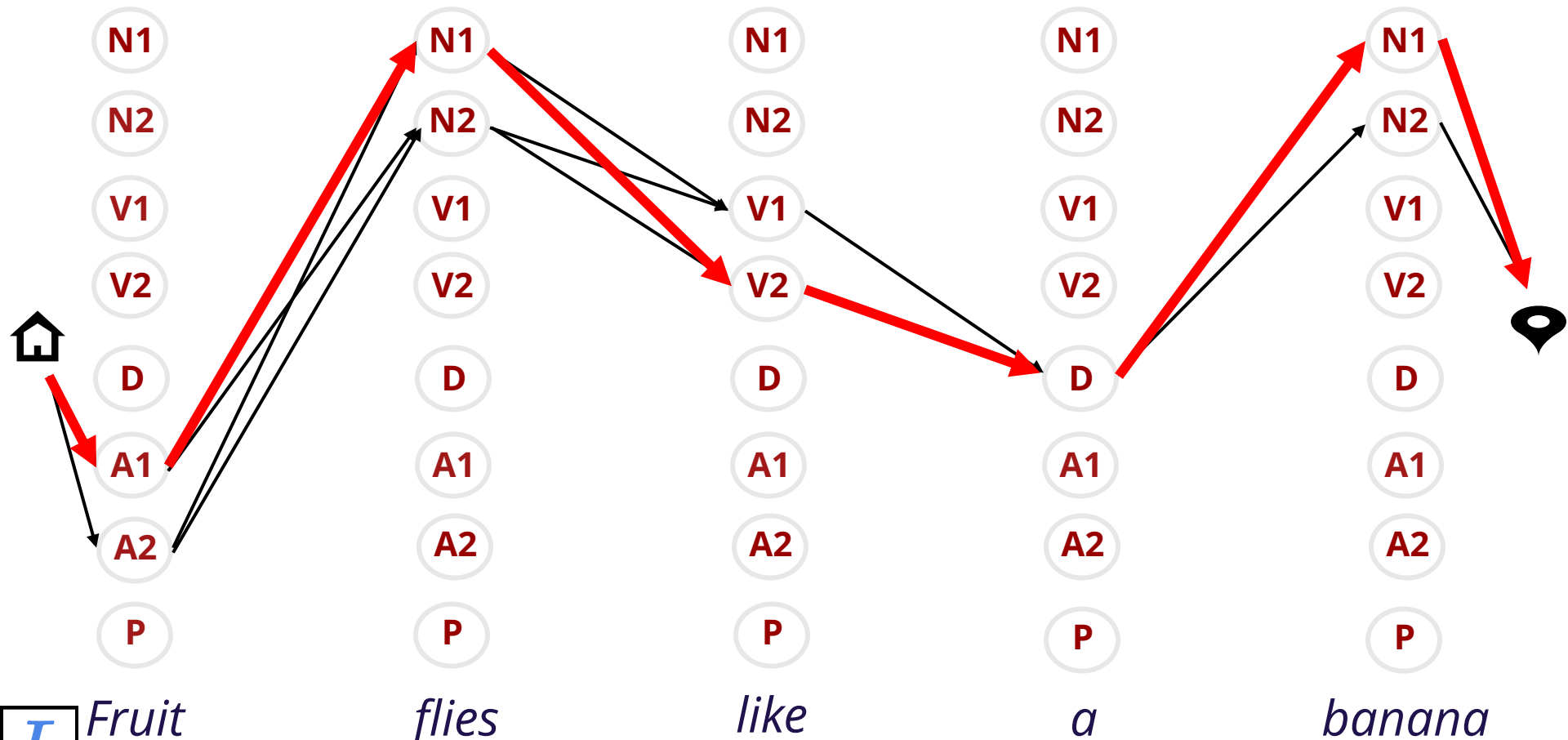
# Latent-Variable CRF

$$\min_{\mathbf{w}} \sum_i \left( -\log \sum_h \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h, y_i)\right) + \log \sum_{h',y} \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h', y)\right) \right)$$



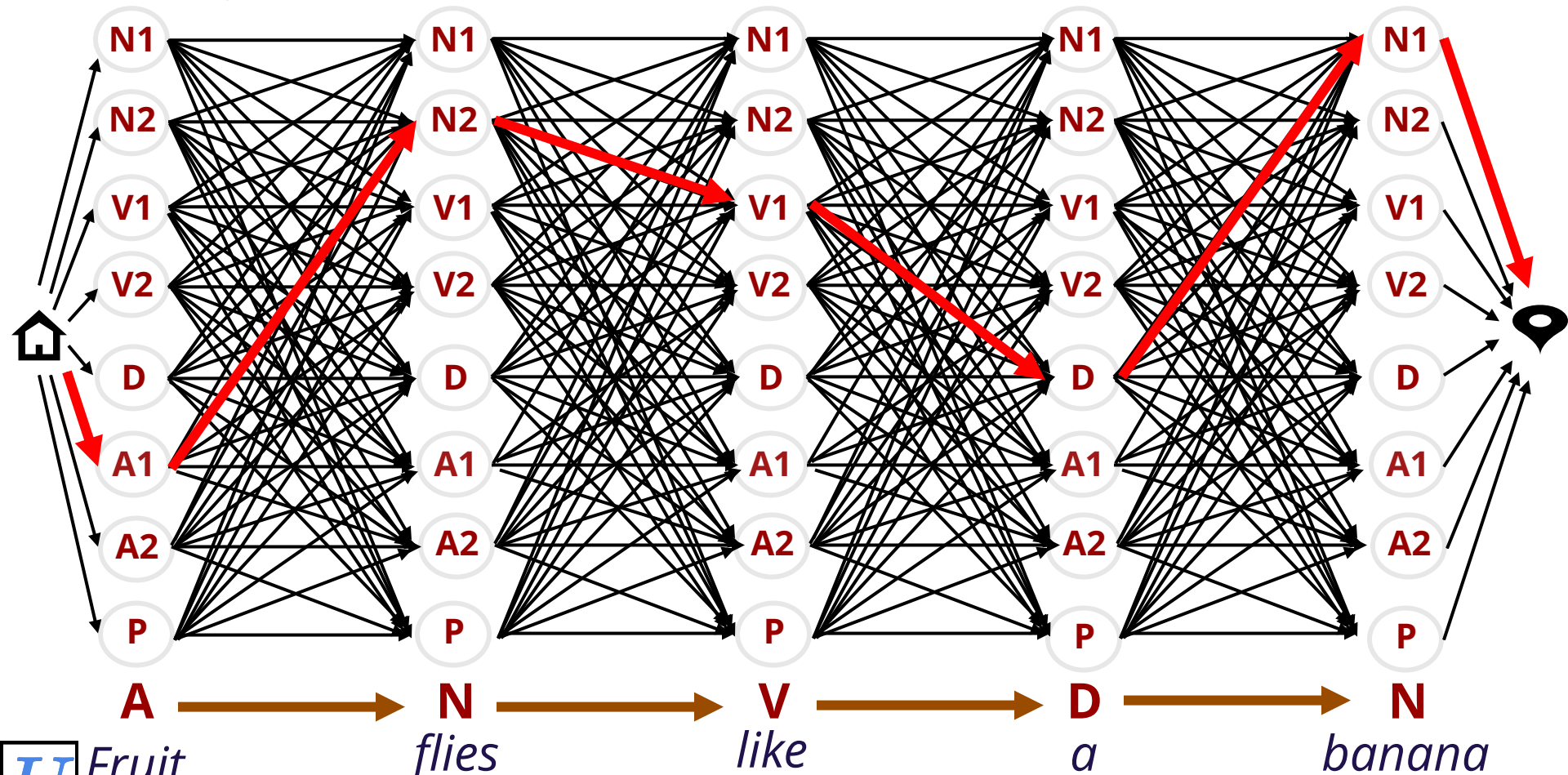
# Latent-Variable SSVM

$$\min_{\mathbf{w}} \sum_i \left( -\max_h \left( \mathbf{w} \cdot \mathbf{f}(x_i, h, y_i) \right) + \max_{h', y} \left( \Delta(y_i, y, h') + \mathbf{w} \cdot \mathbf{f}(x_i, h', y) \right) \right)$$



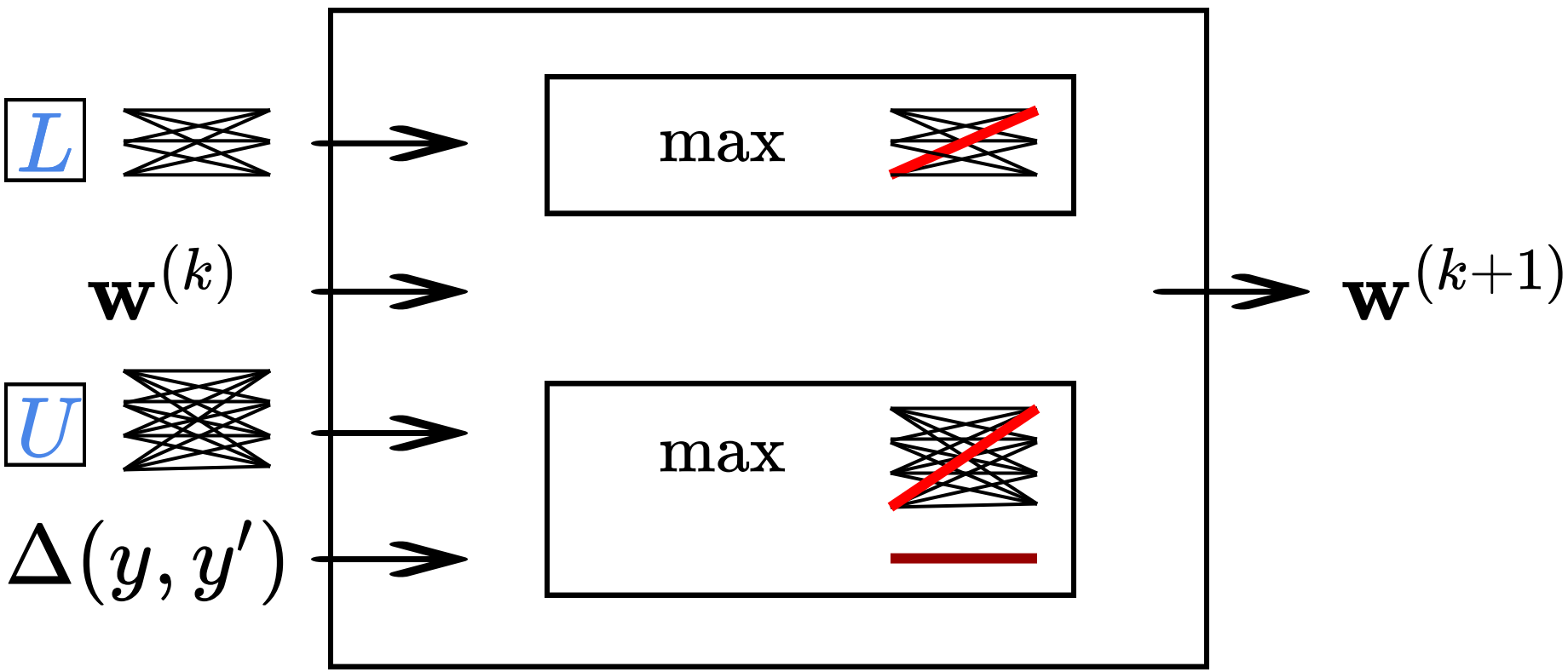
# Latent-Variable SSVM

$$\min_{\mathbf{w}} \sum_i \left( -\max_h \left( \mathbf{w} \cdot \mathbf{f}(x_i, h, y_i) \right) + \max_{h', y} \left( \Delta(y_i, y, h') + \mathbf{w} \cdot \mathbf{f}(x_i, h', y) \right) \right)$$



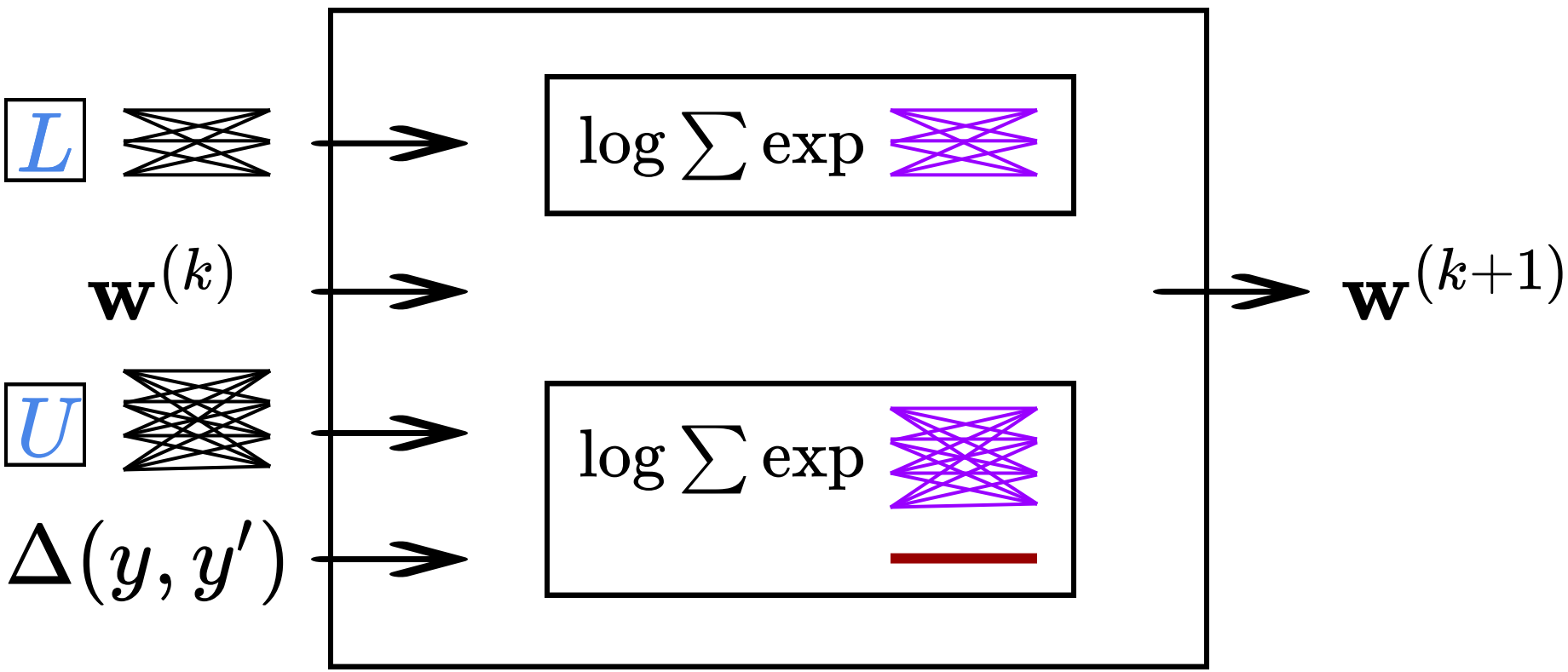
# Learning

Structured Perceptron, SSVM, Latent SSVM, ...

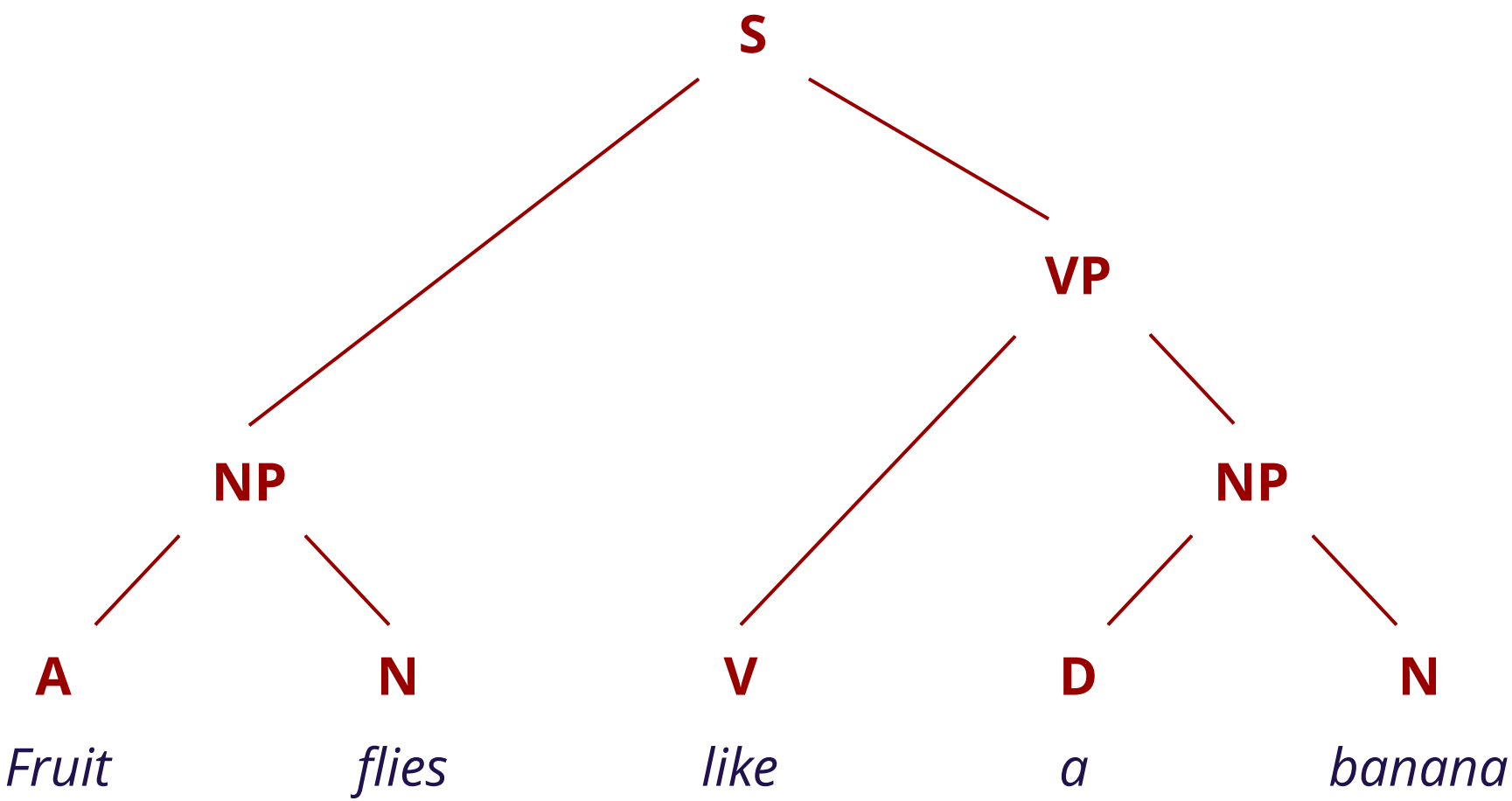


# Learning

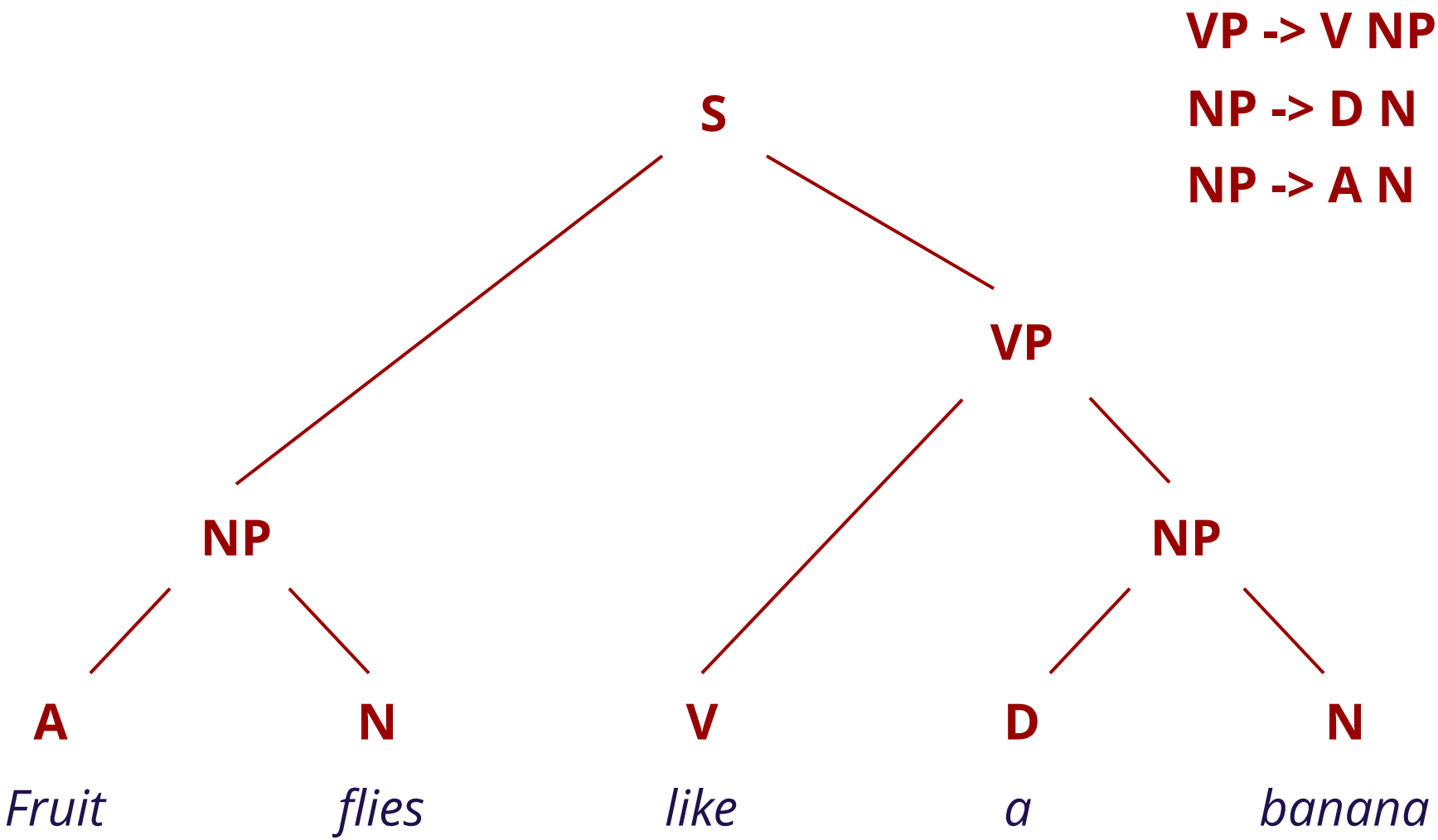
Linear/Semi/Latent/Softmax-margin CRF, ...



# Constituency Parsing

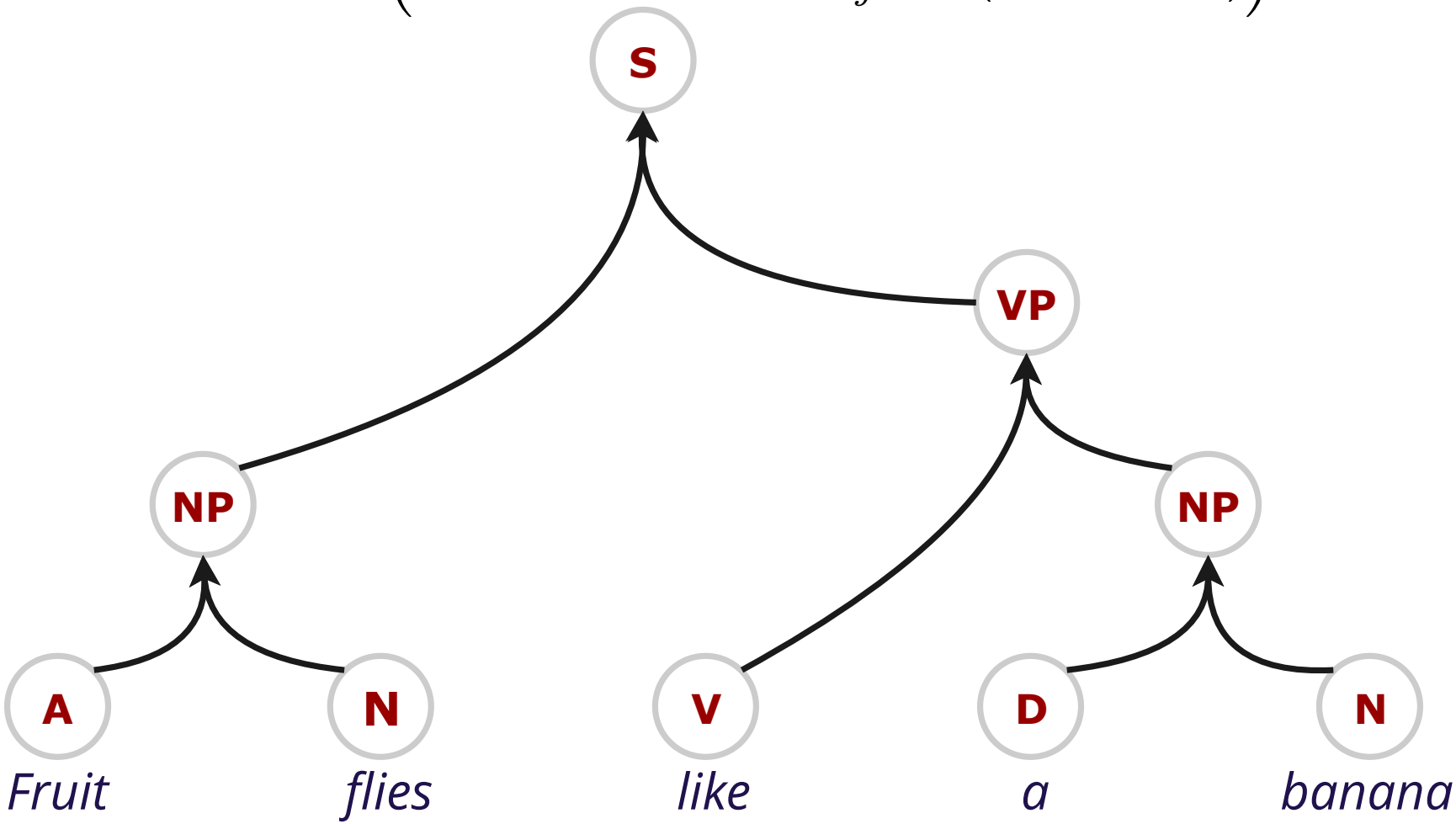


# Constituency Parsing



# Parsing with CRF

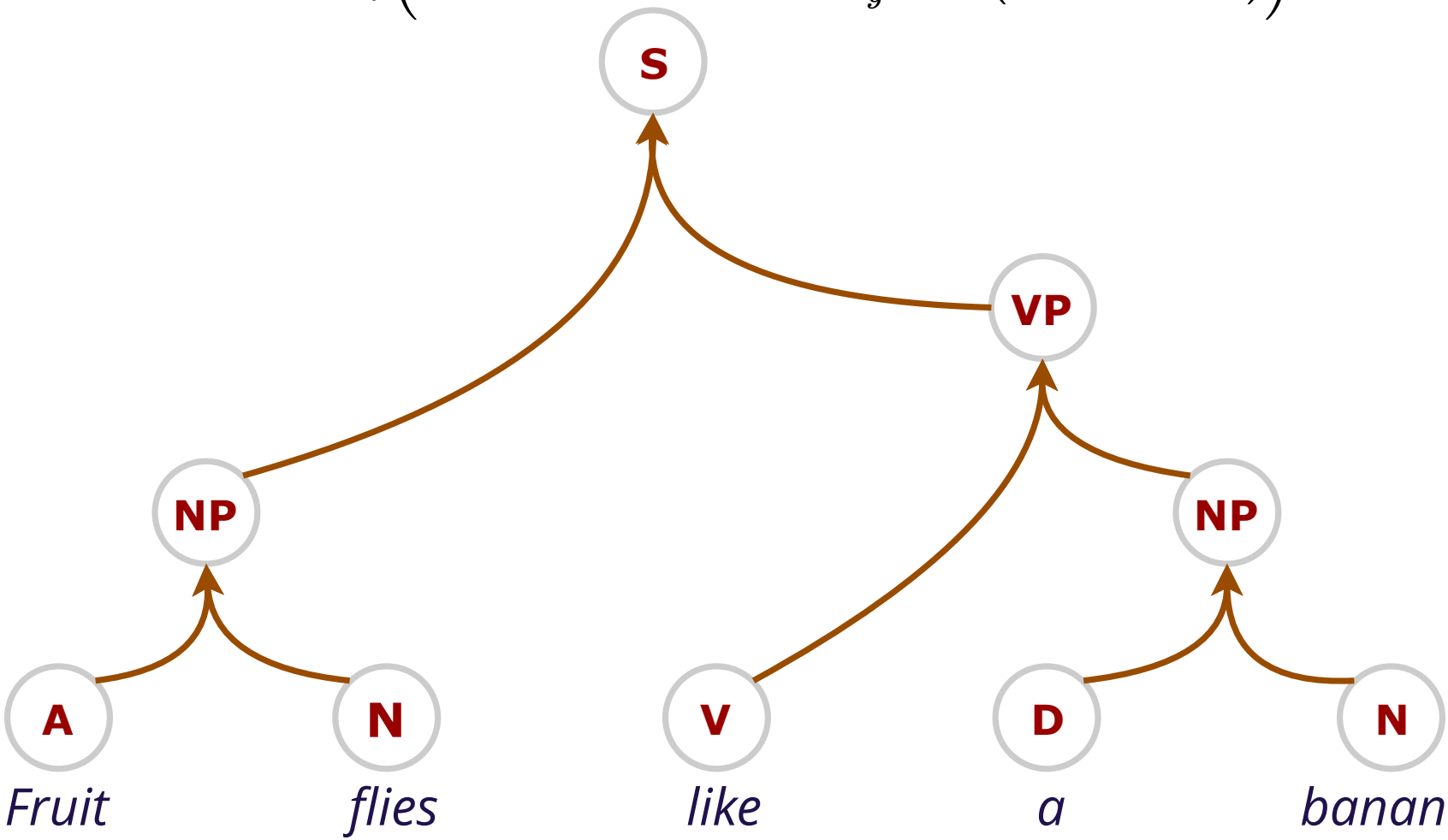
$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$




Finkel, J. R., Kleeman, A., & Manning, C. D. (2008). Efficient, Feature-based, Conditional Random Field Parsing.

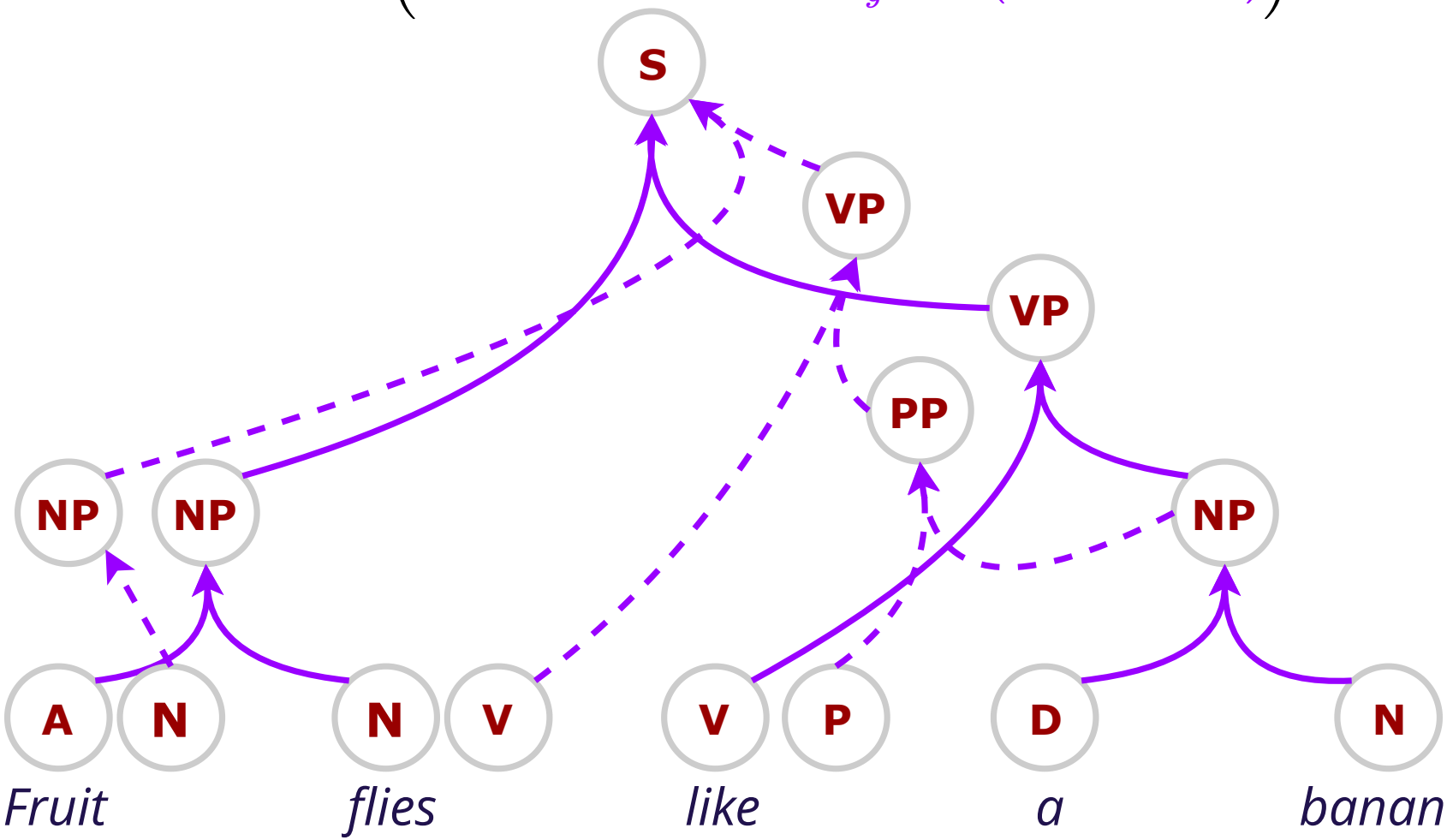
# Parsing with CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$



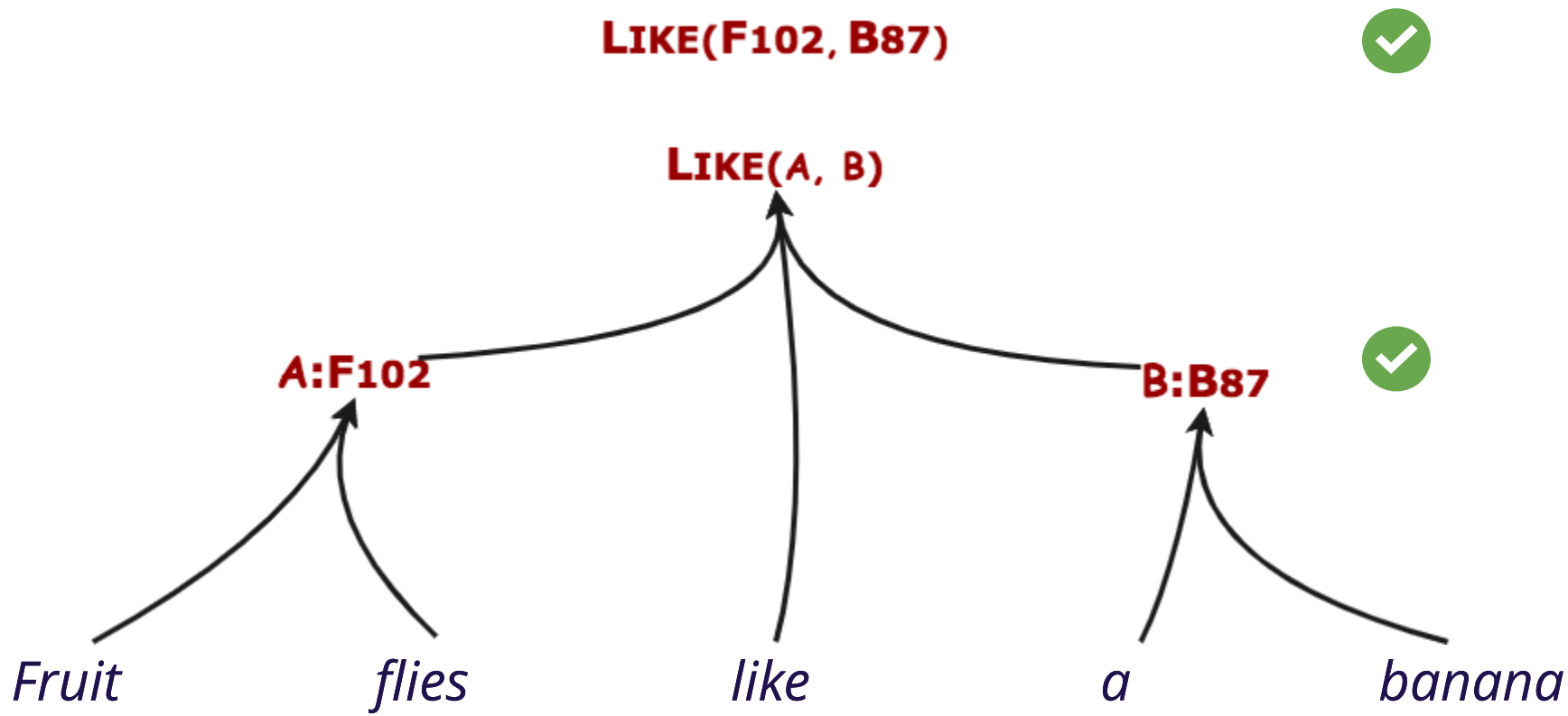
# Parsing with CRF

$$\min_{\mathbf{w}} \sum_i \left( -\mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \log \sum_y \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, y)\right) \right)$$



# Hybrid Tree

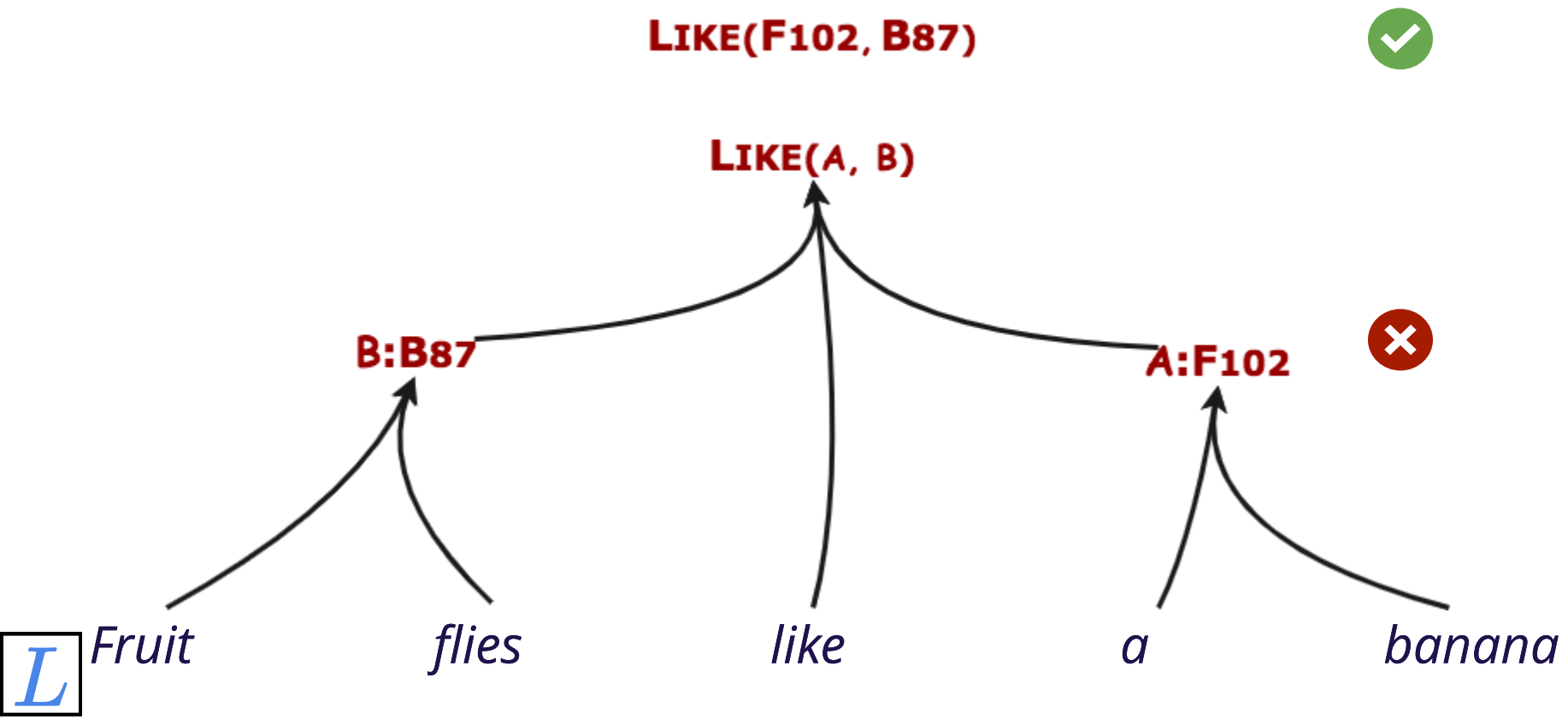
$$\min_{\mathbf{w}} \sum_i \left( -\log \sum_h \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h, y_i)\right) + \log \sum_{h',y} \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h', y)\right)\right)$$




Lu, W. (2014). Semantic Parsing with Relaxed Hybrid Trees.

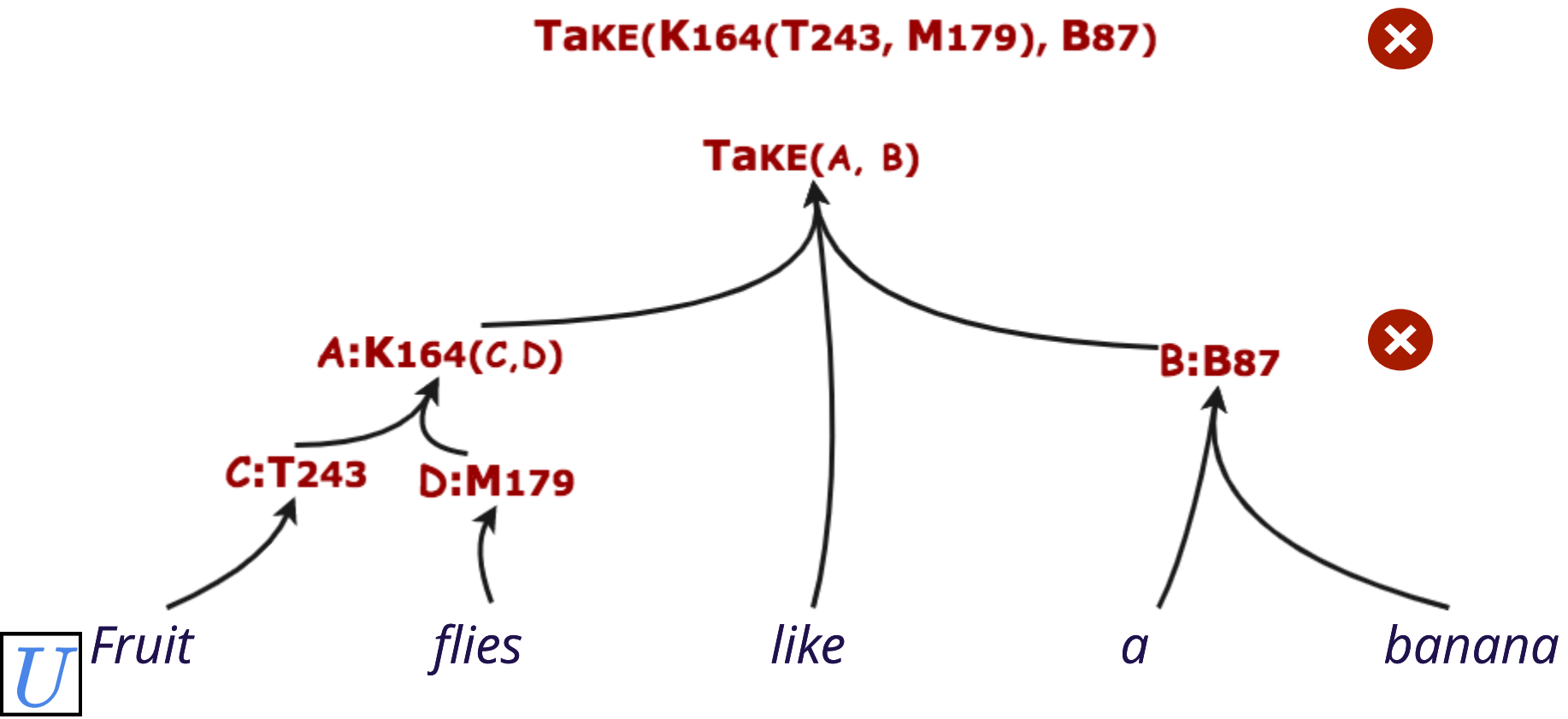
# Hybrid Tree

$$\min_{\mathbf{w}} \sum_i \left( -\log \sum_h \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h, y_i)\right) + \log \sum_{h',y} \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h', y)\right)\right)$$

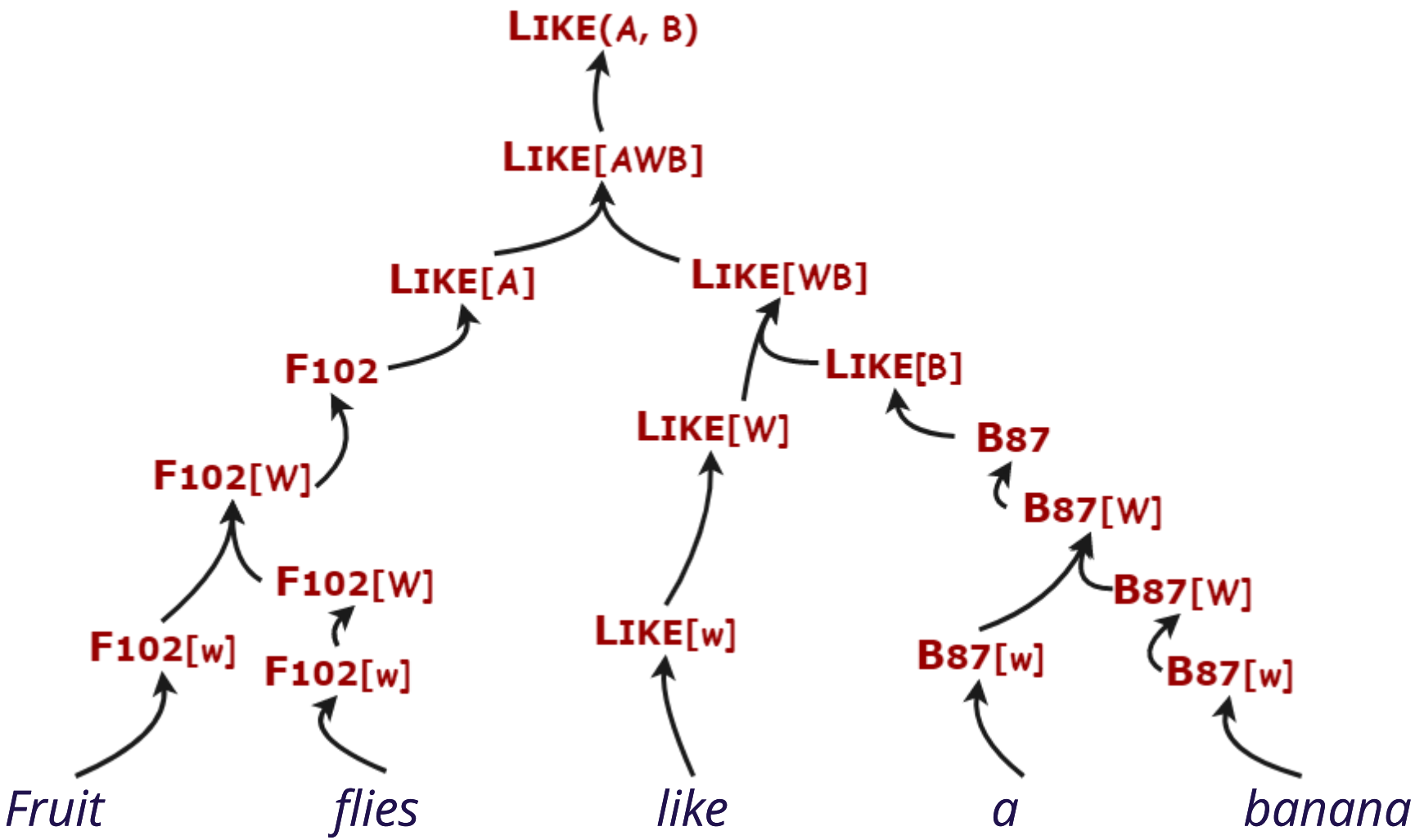


# Hybrid Tree

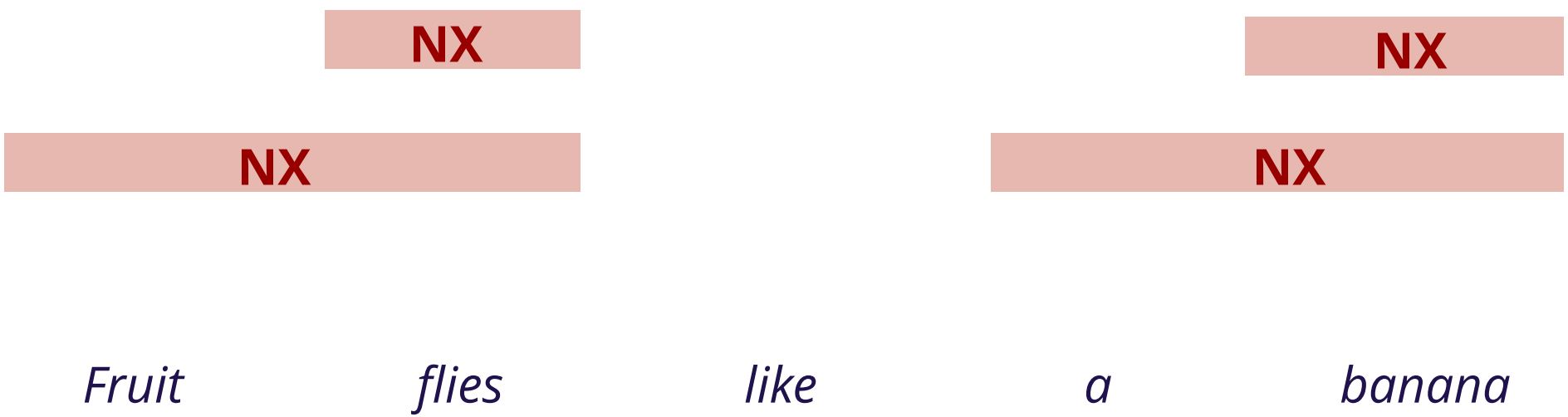
$$\min_{\mathbf{w}} \sum_i \left( -\log \sum_h \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h, y_i)\right) + \log \sum_{h',y} \exp\left(\mathbf{w} \cdot \mathbf{f}(x_i, h', y)\right) \right)$$



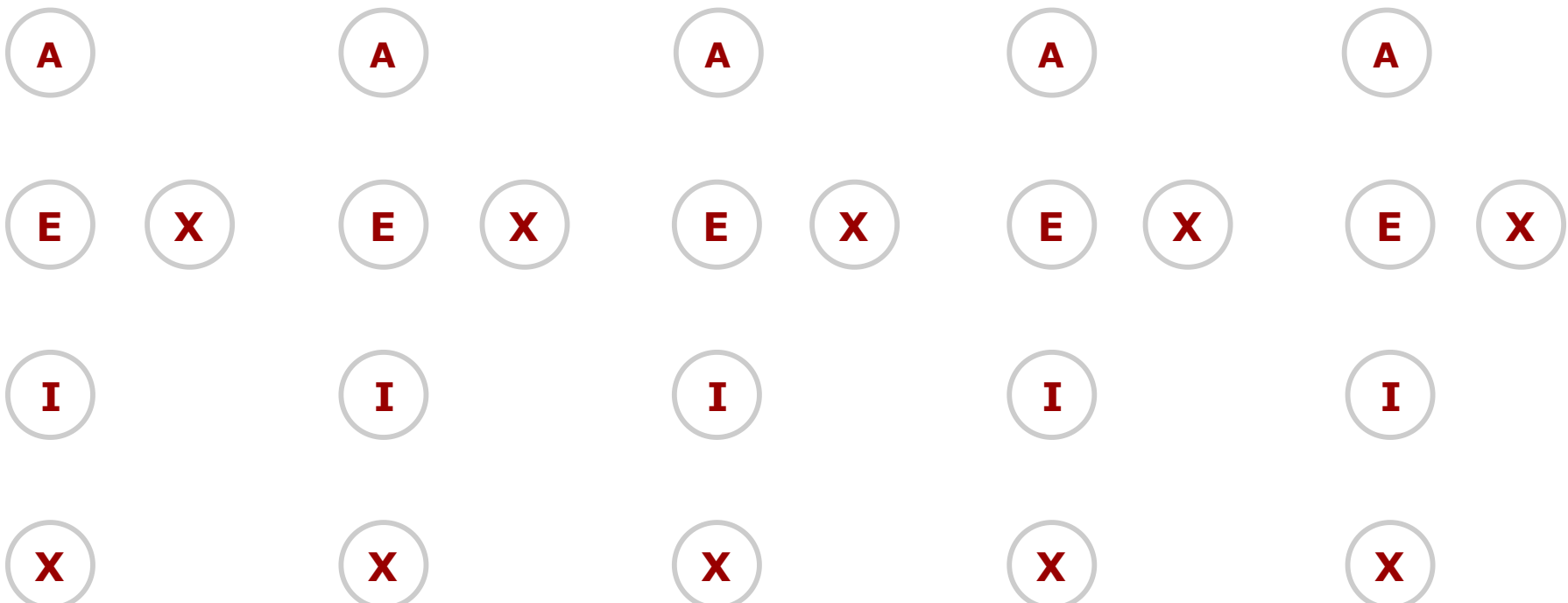
# Hybrid Tree


LIKE(A, B)
LIKE[AWB]
LIKE[A]
LIKE[WB]
F102
LIKE[B]
LIKE[W]
B87
F102[W]
B87[W]
F102[W]
B87[W]
F102[w]
F102[w]
LIKE[w]
B87[w]
B87[w]
Fruit
flies
like
a
banana

# Overlapping Structures

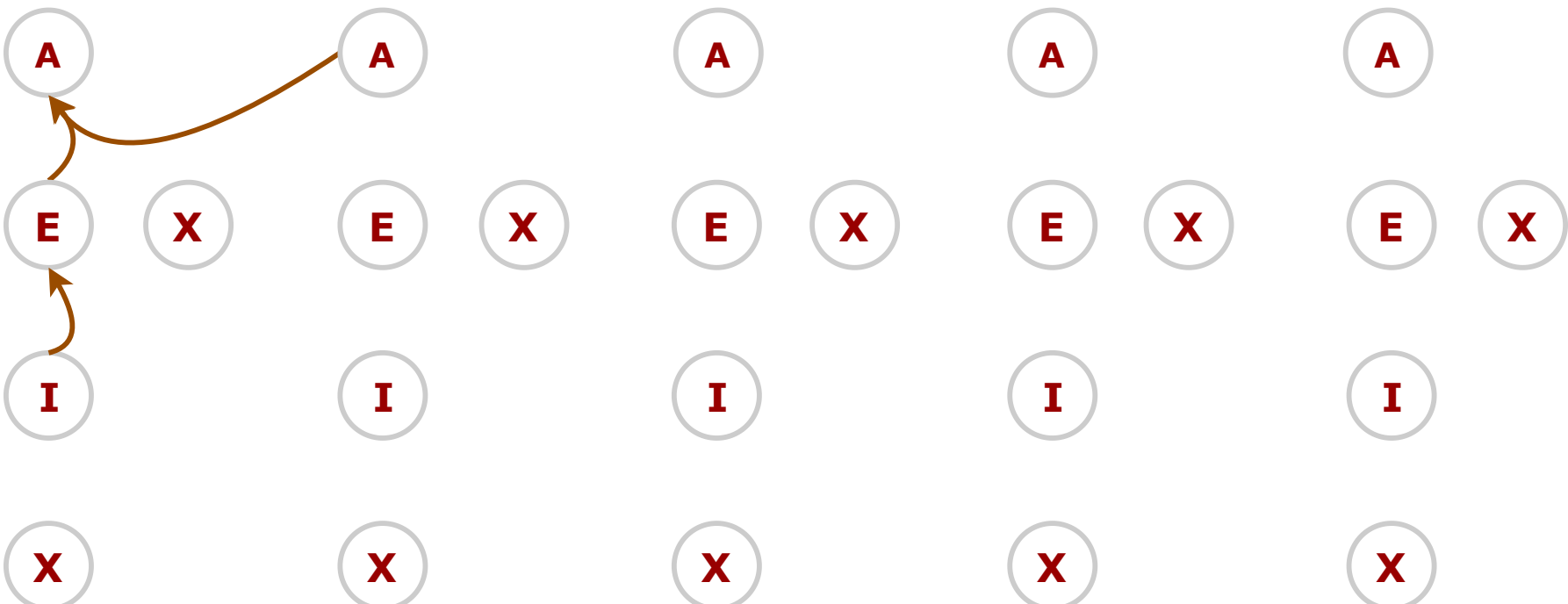


# Mention Hypergraphs

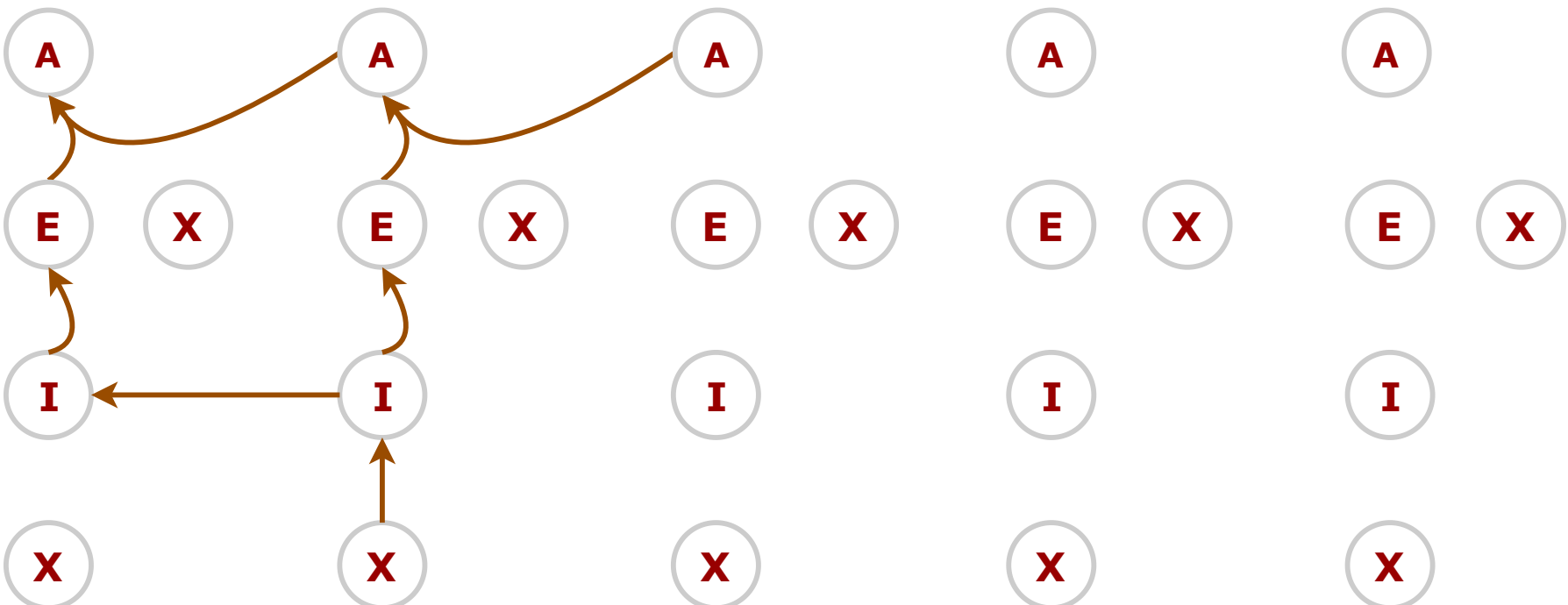



Lu, W., & Roth, D. (2015). Joint Mention Extraction and Classification with Mention Hypergraphs. In EMNLP.

# Mention Hypergraphs

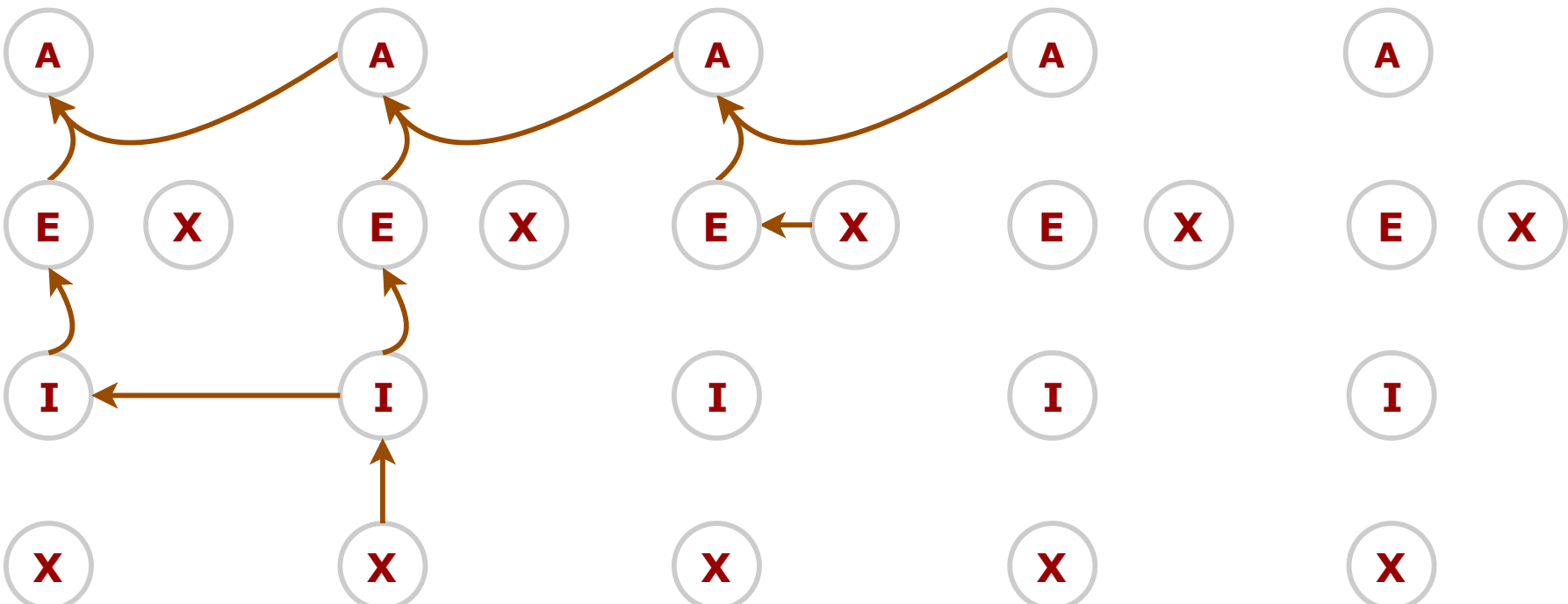


# Mention Hypergraphs



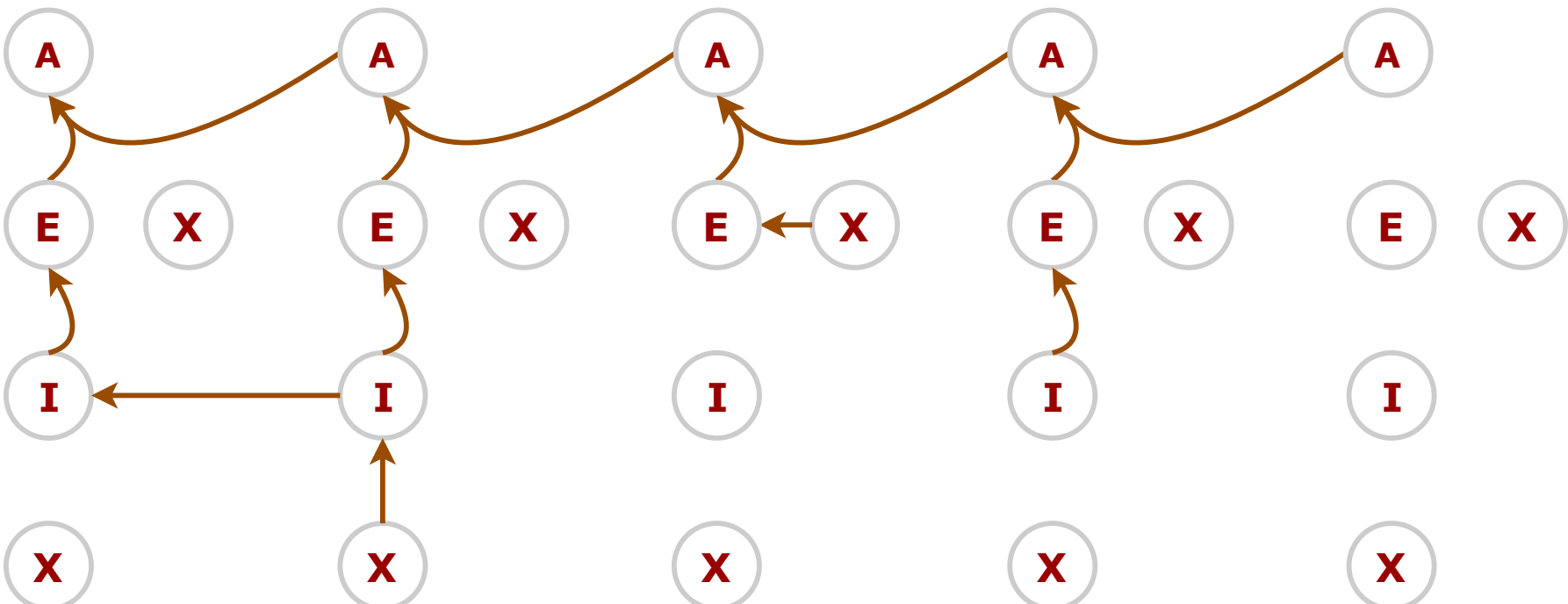

*Fruit* *flies* *like* *a* *banana*

# Mention Hypergraphs



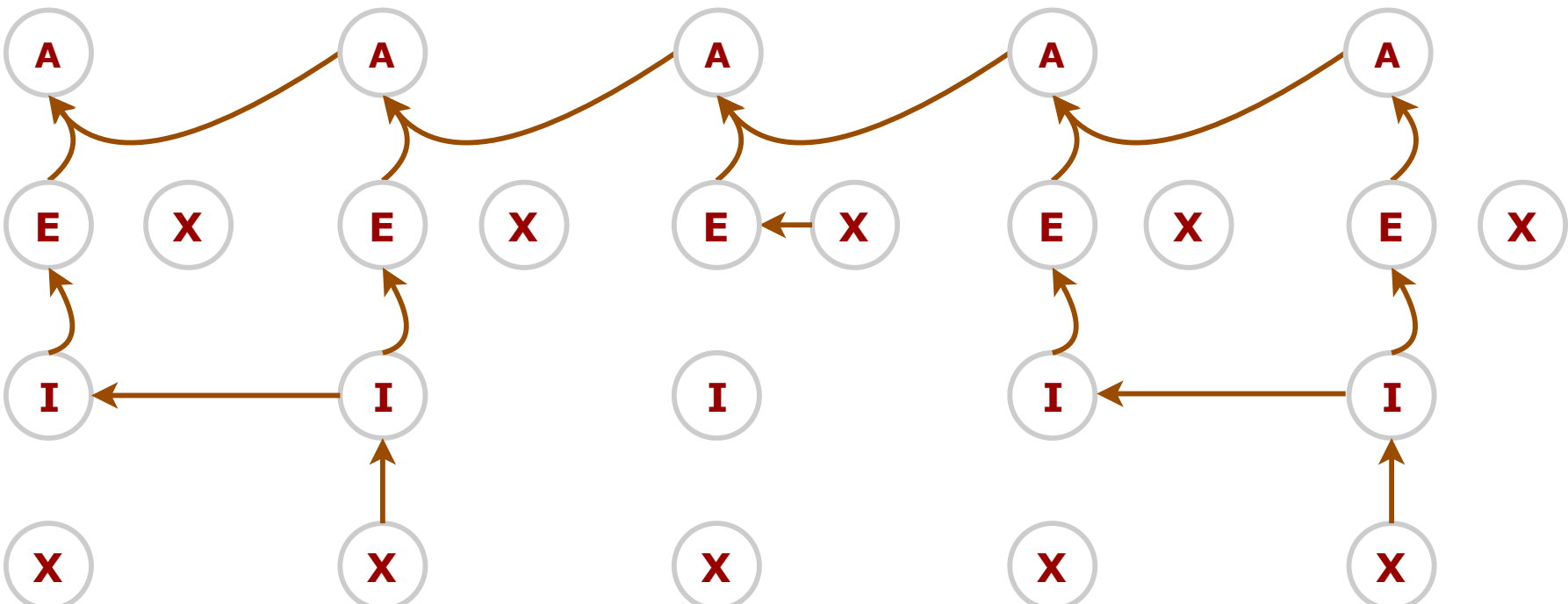

*Fruit* *flies* *like* *a* *banana*

# Mention Hypergraphs



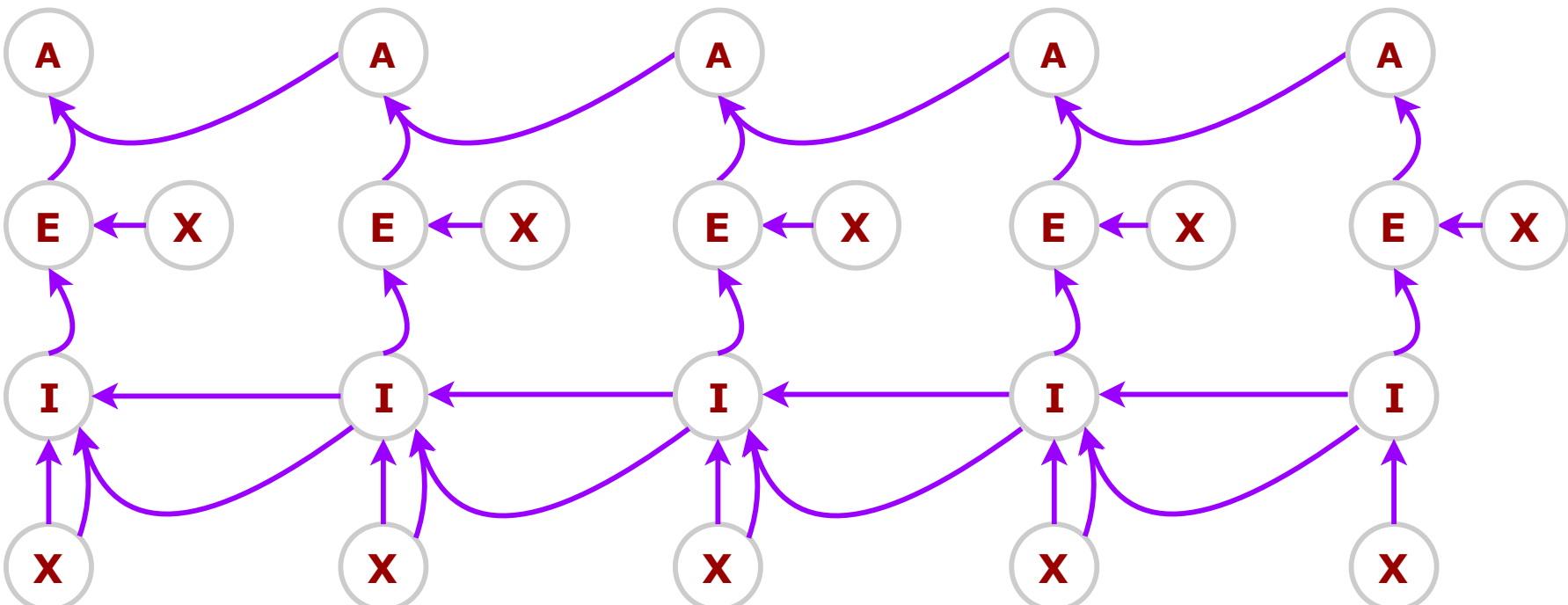

*Fruit* *flies* *like* *a* *banana*

# Mention Hypergraphs



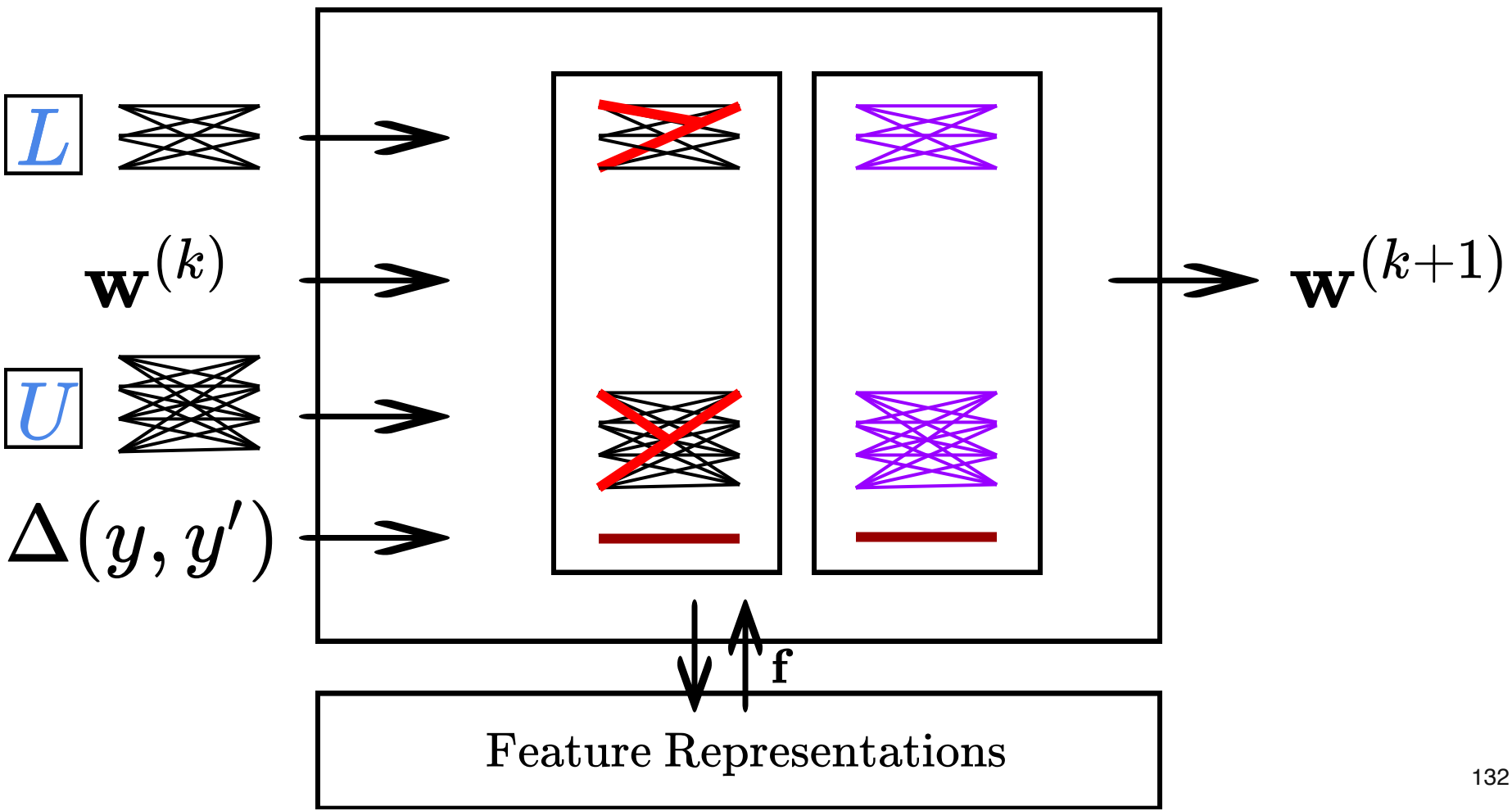

*Fruit* *flies* *like* *a* *banana*

# Mention Hypergraphs

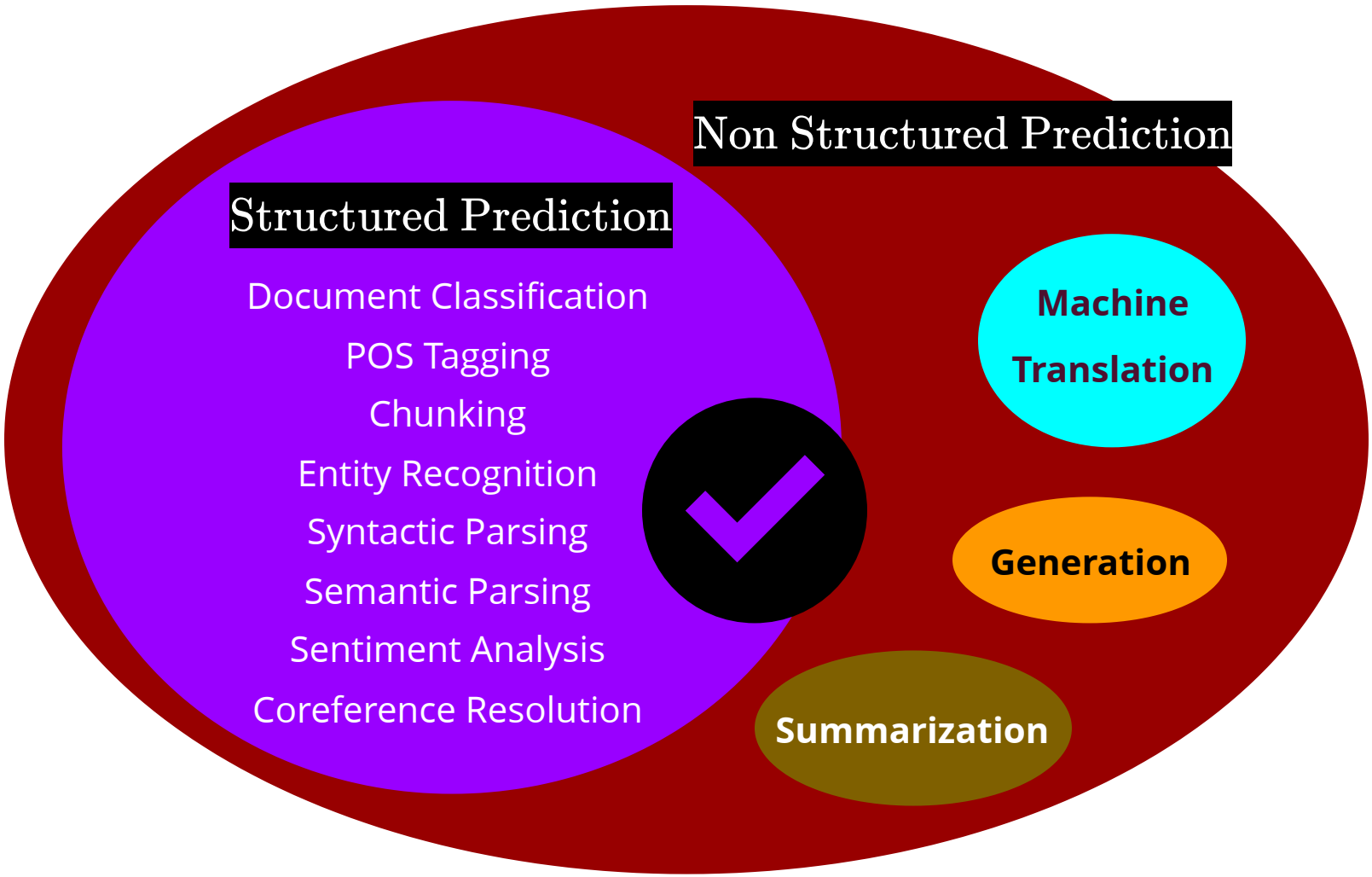



*Fruit* *flies* *like* *a* *banana*

# A Unified Framework for Structured Prediction

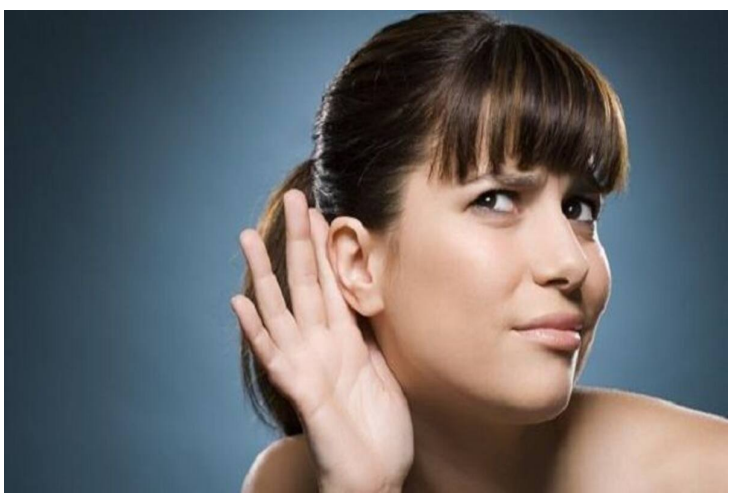


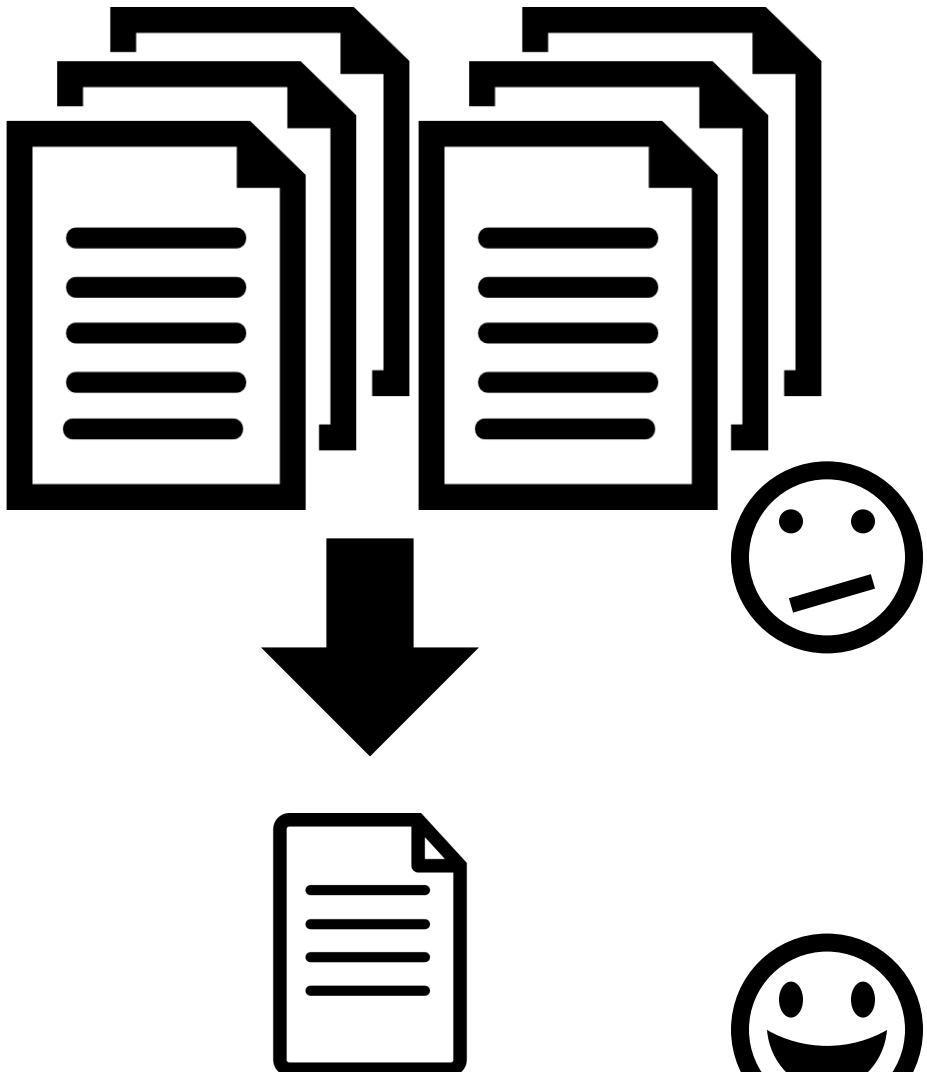
# Tasks in NLP




**Supervised**

# Text Summarization





https://www.thehealthsite.com/news/hearing-difficulty-might-keep-older-patients-away-from-health-care-routine-study-659121/

# Other Generation Tasks

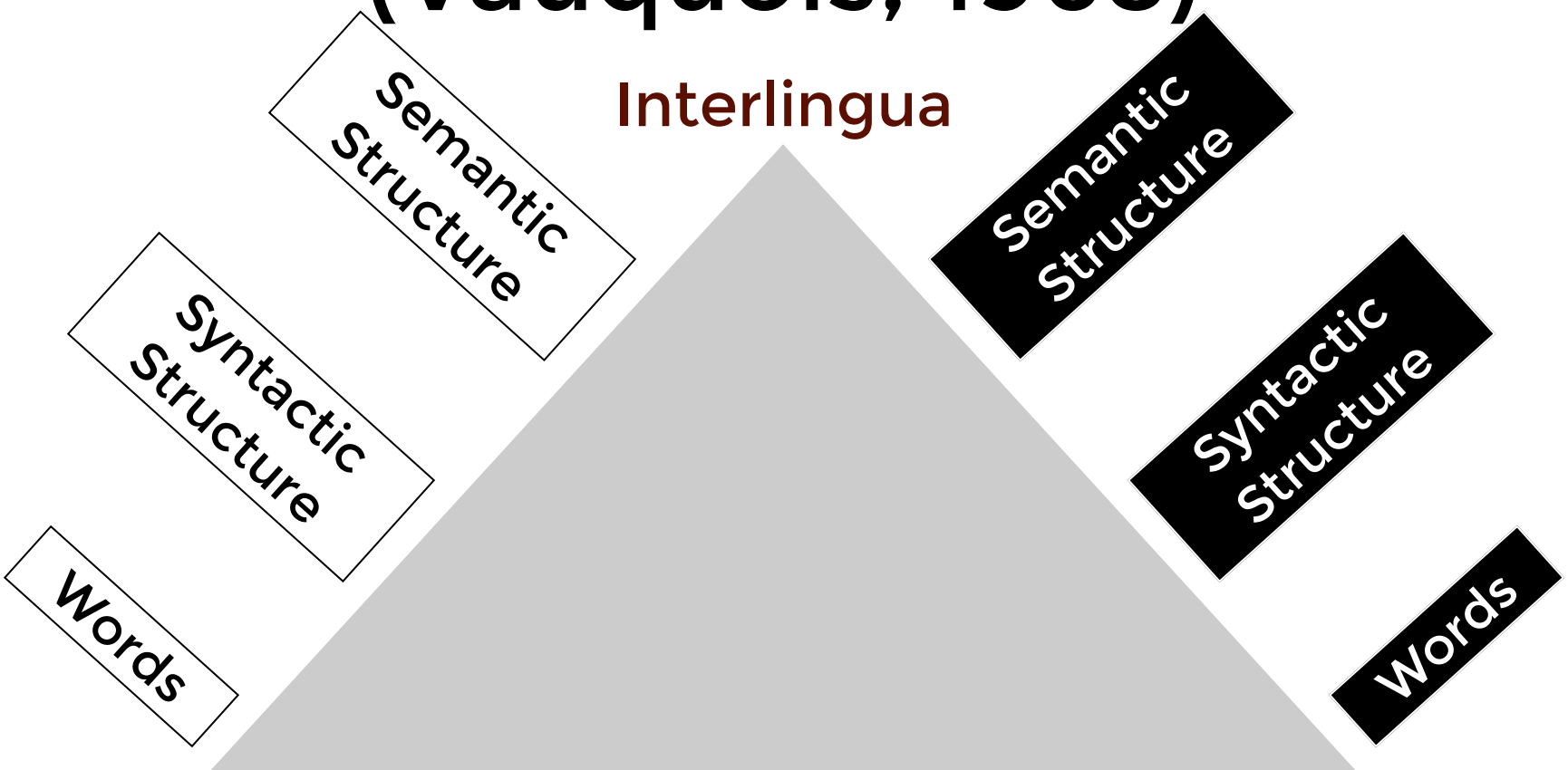
Dialogue Systems

Question Answering

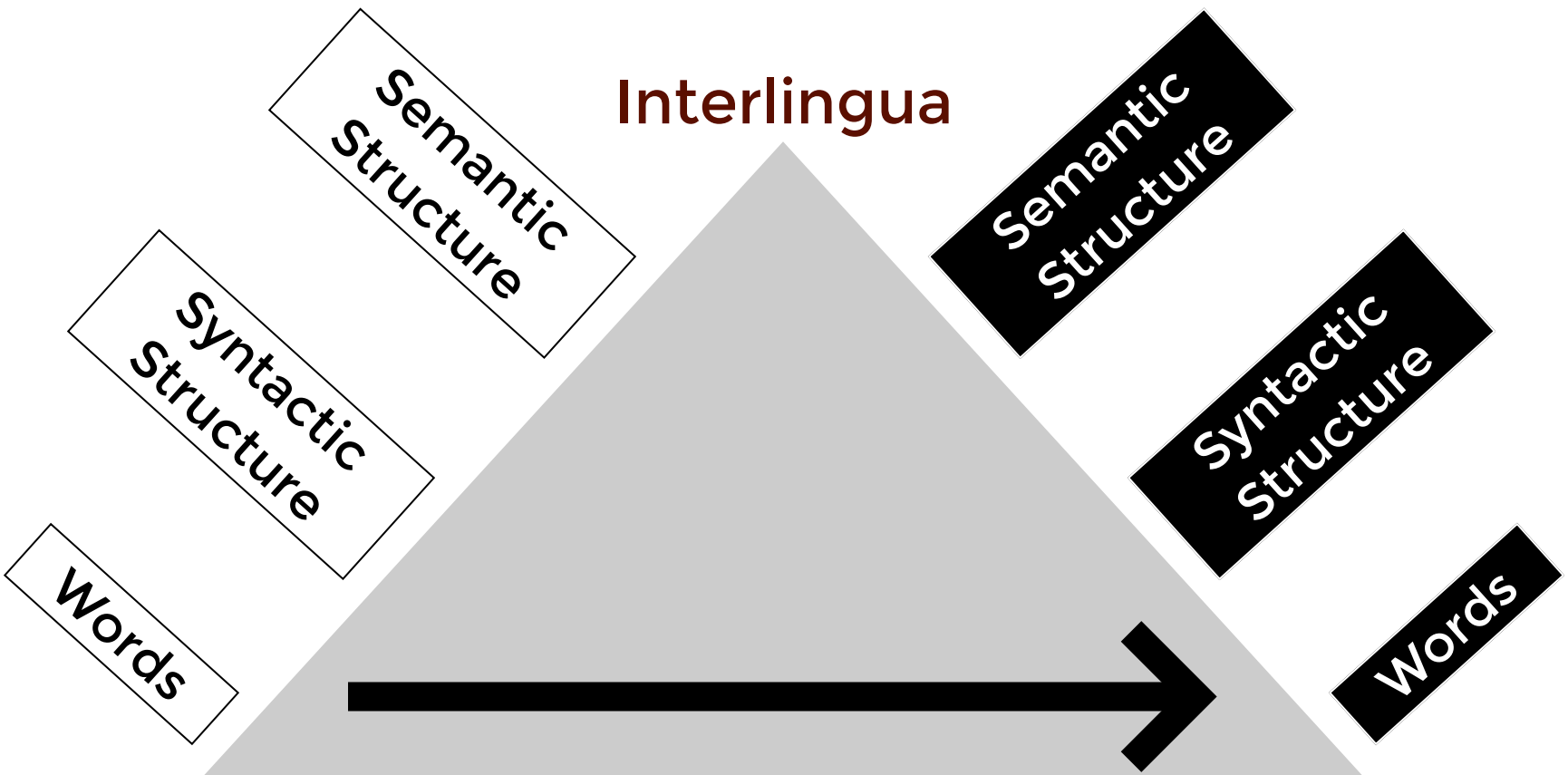
Machine Translation

They are all text-to-text problems!

# Machine Translation (Vauquois, 1968)



# Machine Translation




Text-to-text Problem

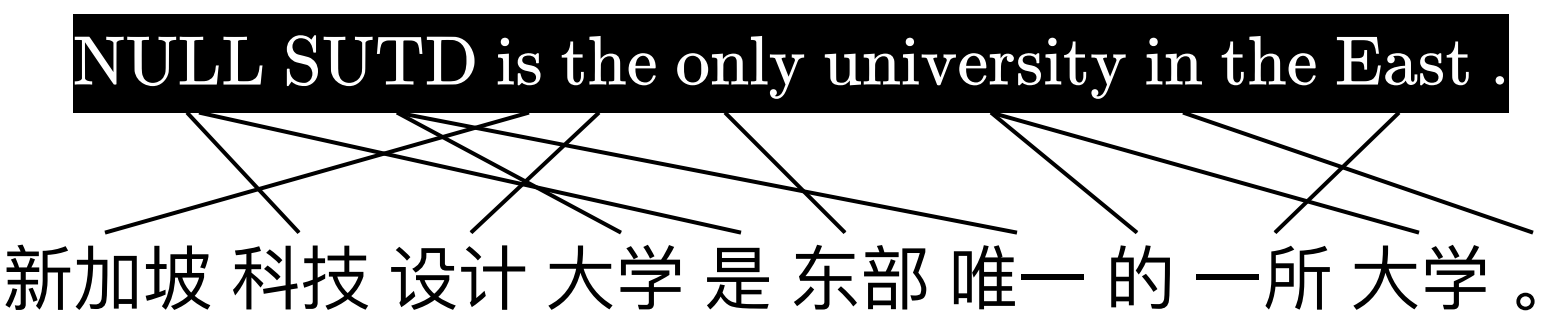
# IBM Model 1

$$p(f_1, f_2, \ldots, f_m | e_0, e_1, e_2, \ldots, e_n, m)$$

We would like to maximize:

$$\sum_{i=1}^{m} \log \left( \sum_{j=0}^{n} t(f_i | e_j) \right)$$

Ignores the word ordering information.
Learn with Expectation Maximization.



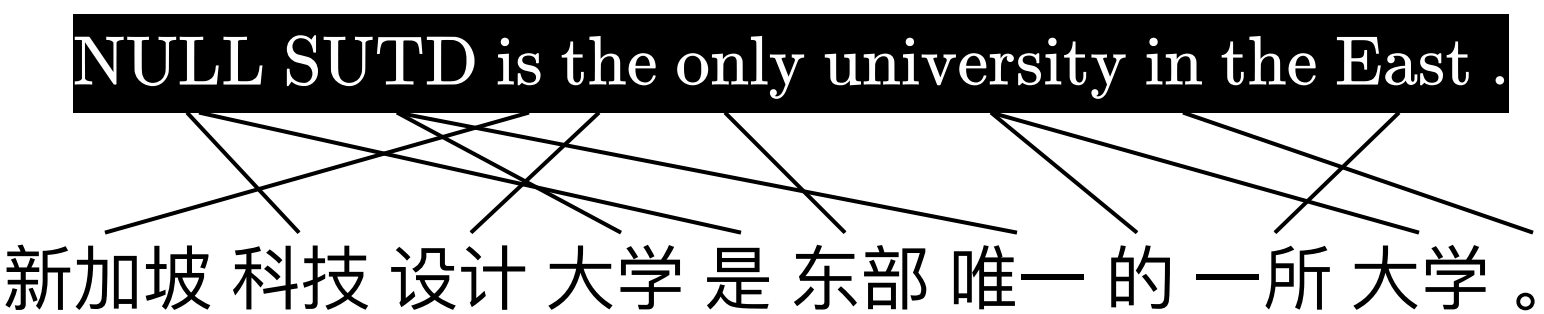
# IBM Model 2

$$p(f_1, f_2, \ldots, f_m | e_0, e_1, e_2, \ldots, e_n, m)$$

$$p(f_1, f_2, \ldots, f_m, a_1, a_2, \ldots, a_m | e_0, e_1, e_2, \ldots, e_n, m)$$

$$\prod_{i=1}^{m} q(a_i | i, n, m) t(f_i | e_{a_i})$$

It additionally learns a distribution that captures (absolute) word reordering



# IBM Models

Pioneering work on word-level translation

Not particularly useful models for translation themselves, but can yield useful alignment information for the training set

A first step towards building other advanced models such as phrase-based and syntax-based models

# Phrase-based Translation

## LM : Language Model

How well the translated sentence reads

## TM : Phrase Translation Model

How faithful the translation is to the original

## DM : Distortion Model

How much efforts on "moving the eyes" in translation is required

# Phrase-based Translation

$p_5 = (6, 6, \text{in the East})$

SUTD is the only university in the East

新加坡 科技 设计 大学 是 东部 唯一 的 一所 大学 。

$$\underbrace{\log q(\text{in}|\text{only}, \text{university}) + \log q(\text{the}|\text{university}, \text{in}) + \log q(\text{East}|\text{in}, \text{the})}_{\text{Language model}}$$

$$+ \quad \underbrace{\log t(\text{东部}|\text{in the East})}_{\text{Phrase translation model}}$$

$$+ \quad \underbrace{\eta \times 5}_{\text{Distortion model}}$$

# Phrase-based Translation

We know how to score a translation derivation, but how do we search for the most optimal derivation?

The position of the last French word in the previous French phrase translated

The score of the partial derivation so far

A state

$$s = (e_1, e_2, b, r, \alpha)$$

The last two English words in the previous translated English phrase

A bit string indicating which words in French are (not yet) translated.

# Phrase-based Translation

$p_2 = (5, 5, \text{is})$

SUTD is

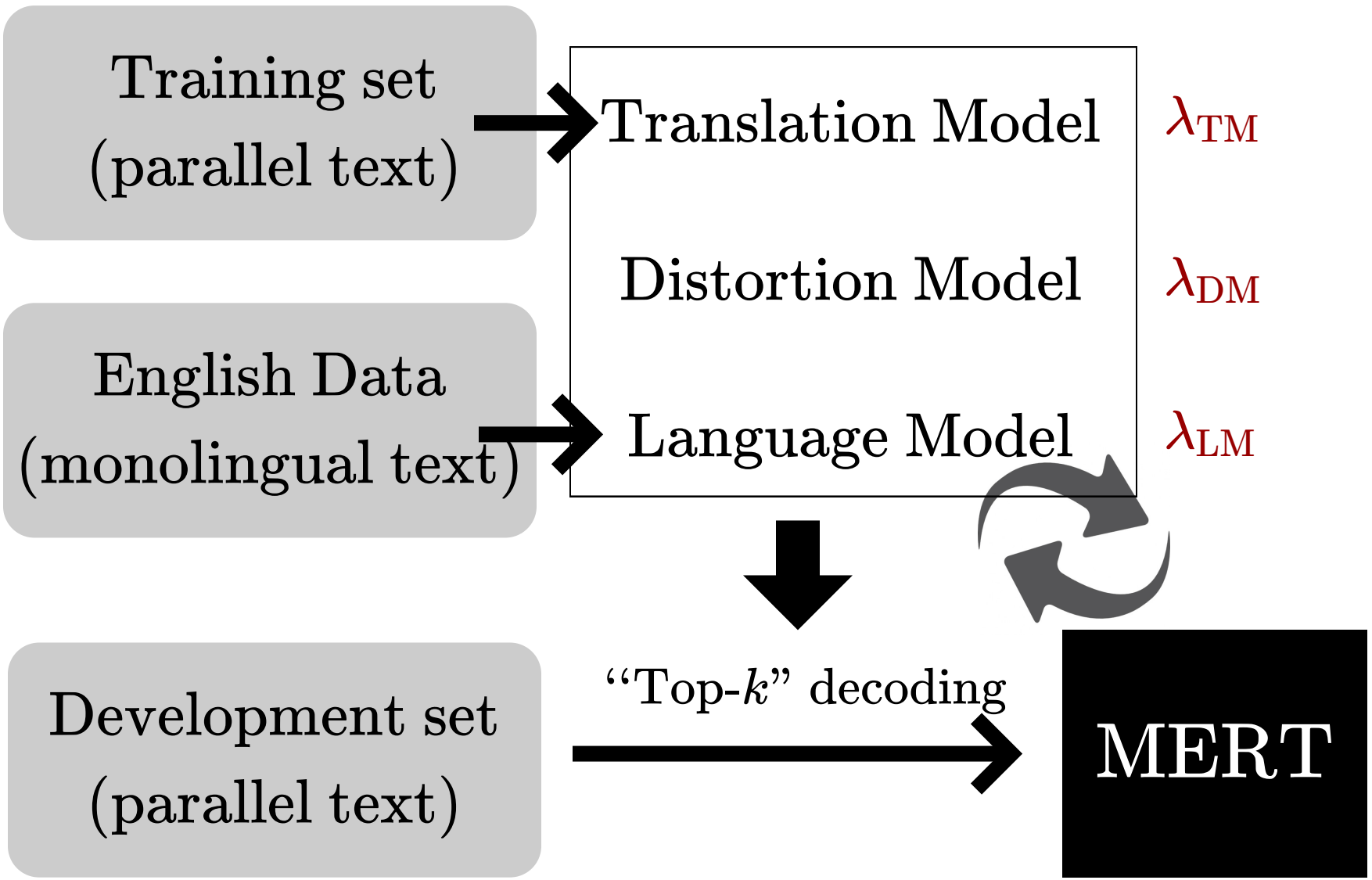
新加坡 科技 设计 大学 是 东部 唯一 的 一所 大学 。

**Each $p_k$ is essentially an action!**

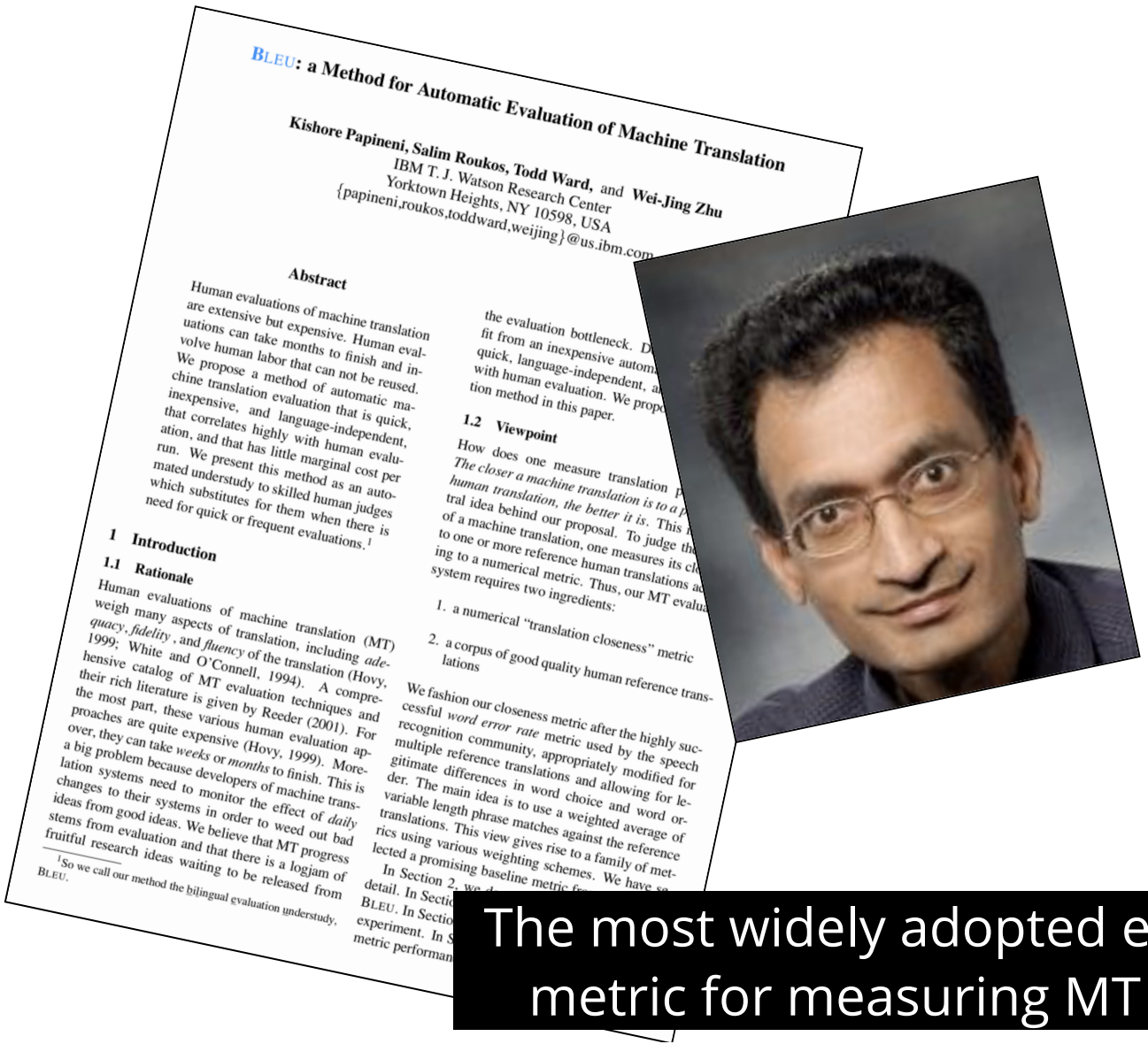
$$(\langle \texttt{START} \rangle, \text{SUTD}, 1111100000, 5, 8.9)$$

$$(\ldots, 1111100010, \ldots) \quad \ldots \quad (\ldots, 1111100111, \ldots)$$

# Phrase-based Translation



# BLEU



BLEU: a Method for Automatic Evaluation of Machine Translation

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598, USA
{papineni,roukos,toddward,weijing}@us.ibm.com

Abstract

Human evaluations of machine translation are extensive but expensive. Human evaluations can take months to finish and involve human labor that can not be reused. We propose a method of automatic machine translation evaluation that is quick, inexpensive, and language-independent, that correlates highly with human evaluation, and that has little marginal cost per run. We present this method as an automated understudy to skilled human judges which substitutes for them when there is need for quick or frequent evaluations.[1]

1 Introduction

1.1 Rationale

Human evaluations of machine translation (MT) weigh many aspects of translation, including *adequacy*, *fidelity* , and *fluency* of the translation (Hovy, 1999; White and O'Connell, 1994). A comprehensive catalog of MT evaluation techniques and their rich literature is given by Reeder (2001). For the most part, these various human evaluation approaches are quite expensive (Hovy, 1999). Moreover, they can take *weeks* or *months* to finish. This is a big problem because developers of machine translation systems need to monitor the effect of *daily* changes to their systems in order to weed out bad ideas from good ideas. We believe that MT progress stems from evaluation and that there is a logjam of fruitful research ideas waiting to be released from the evaluation bottleneck. D
fit from an inexpensive autom
quick, language-independent, a
with human evaluation. We propo
tion method in this paper.

1.2 Viewpoint

How does one measure translation p
*The closer a machine translation is to a p
human translation, the better it is*. This
tral idea behind our proposal. To judge th
of a machine translation, one measures its cl
to one or more reference human translations a
ing to a numerical metric. Thus, our MT evalua
system requires two ingredients:

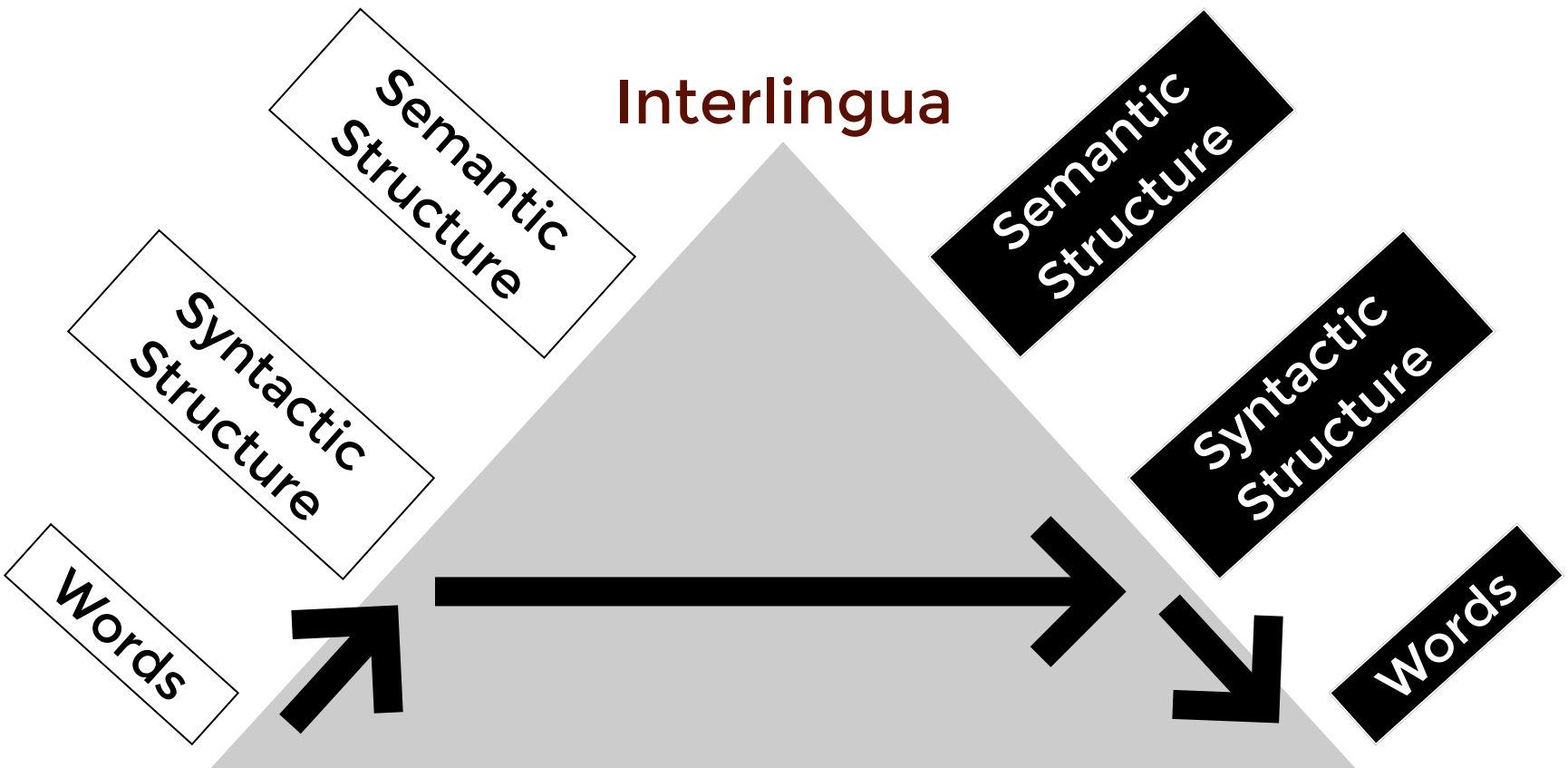
1. a numerical "translation closeness" metric
2. a corpus of good quality human reference translations

We fashion our closeness metric after the highly successful *word error rate* metric used by the speech recognition community, appropriately modified for multiple reference translations and allowing for legitimate differences in word choice and word order. The main idea is to use a weighted average of variable length phrase matches against the reference translations. This view gives rise to a family of metrics using various weighting schemes. We have se
lected a promising baseline metric fr
In Section 2, We d
detail. In Sectio
BLEU. In Sectio
experiment. In S
metric performan

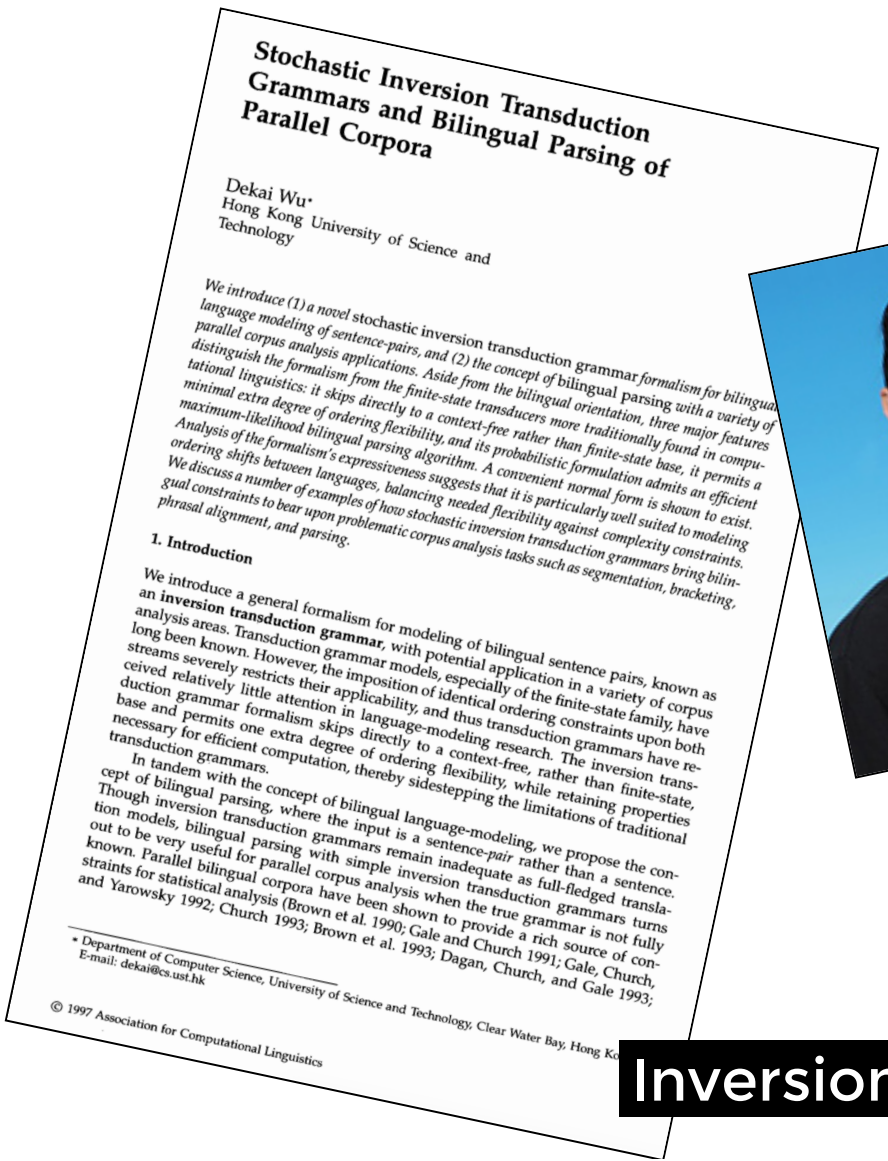
[1] So we call our method the bilingual evaluation understudy, BLEU.

The most widely adopted evaluation metric for measuring MT quality.

# Machine Translation



# Synchronous Grammar



**Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora**

Dekai Wu*
Hong Kong University of Science and Technology

*We introduce (1) a novel* stochastic inversion transduction grammar *formalism for bilingual language modeling of sentence-pairs, and (2) the concept of* bilingual parsing *with a variety of parallel corpus analysis applications. Aside from the bilingual orientation, three major features distinguish the formalism from the finite-state transducers more traditionally found in computational linguistics: it skips directly to a context-free rather than finite-state base, it permits a minimal extra degree of ordering flexibility, and its probabilistic formulation admits an efficient maximum-likelihood bilingual parsing algorithm. A convenient normal form is shown to exist. Analysis of the formalism's expressiveness suggests that it is particularly well suited to modeling ordering shifts between languages, balancing needed flexibility against complexity constraints. We discuss a number of examples of how stochastic inversion transduction grammars bring bilingual constraints to bear upon problematic corpus analysis tasks such as segmentation, bracketing, phrasal alignment, and parsing.*

**1. Introduction**

We introduce a general formalism for modeling of bilingual sentence pairs, known as an **inversion transduction grammar,** with potential application in a variety of corpus analysis areas. Transduction grammar models, especially of the finite-state family, have long been known. However, the imposition of identical ordering constraints upon both streams severely restricts their applicability, and thus transduction grammars have received relatively little attention in language-modeling research. The inversion transduction grammar formalism skips directly to a context-free, rather than finite-state, base and permits one extra degree of ordering flexibility, while retaining properties necessary for efficient computation, thereby sidestepping the limitations of traditional transduction grammars.

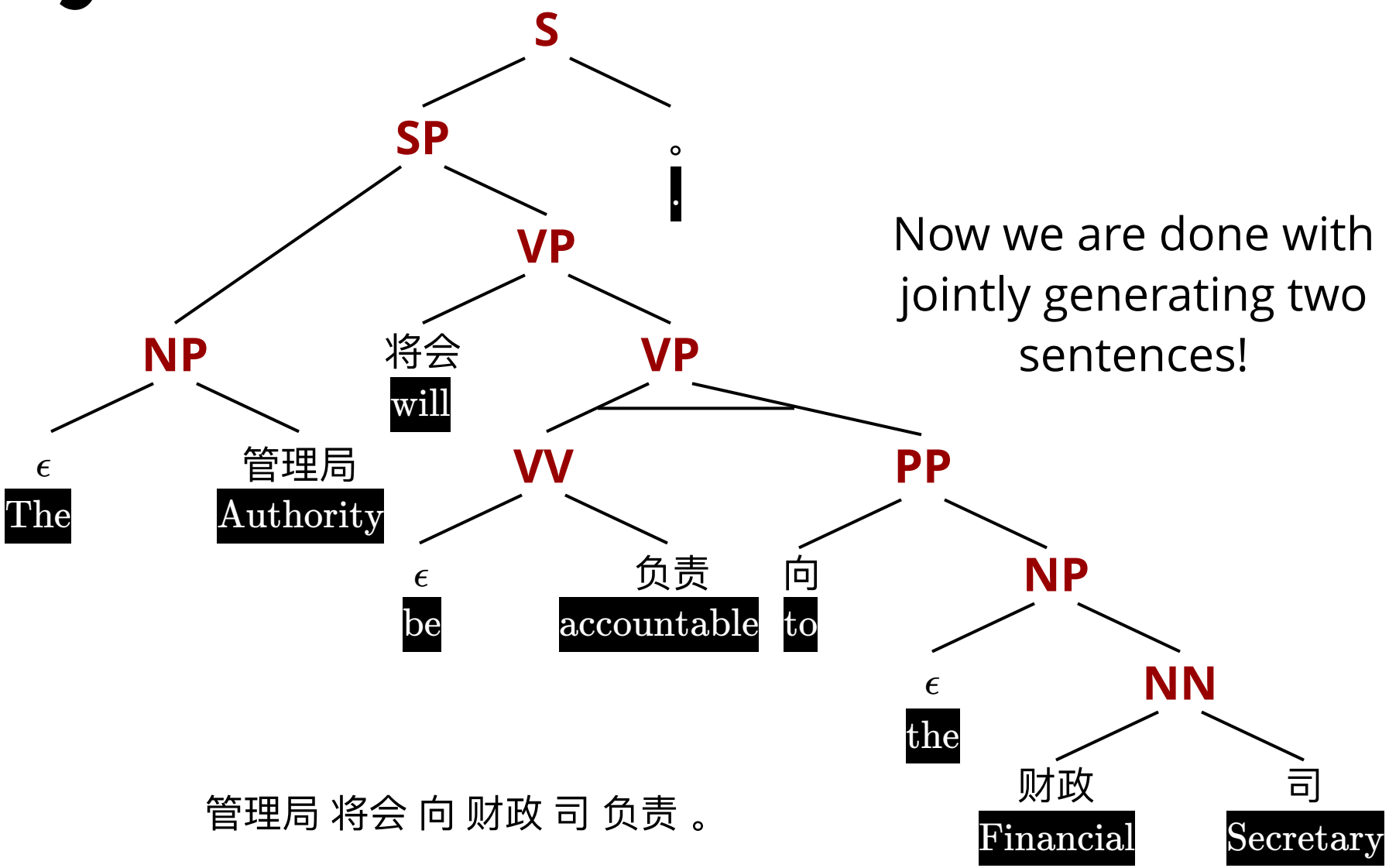
In tandem with the concept of bilingual language-modeling, we propose the concept of bilingual parsing, where the input is a sentence-*pair* rather than a sentence. Though inversion transduction grammars remain inadequate as full-fledged translation models, bilingual parsing with simple inversion transduction grammars turns out to be very useful for parallel corpus analysis when the true grammar is not fully known. Parallel bilingual corpora have been shown to provide a rich source of constraints for statistical analysis (Brown et al. 1990; Gale and Church 1991; Gale, Church, and Yarowsky 1992; Church 1993; Brown et al. 1993; Dagan, Church, and Gale 1993;

* Department of Computer Science, University of Science and Technology, Clear Water Bay, Hong Ko
E-mail: dekai@cs.ust.hk



Inversion Transduction Grammar

# Synchronous Derivation

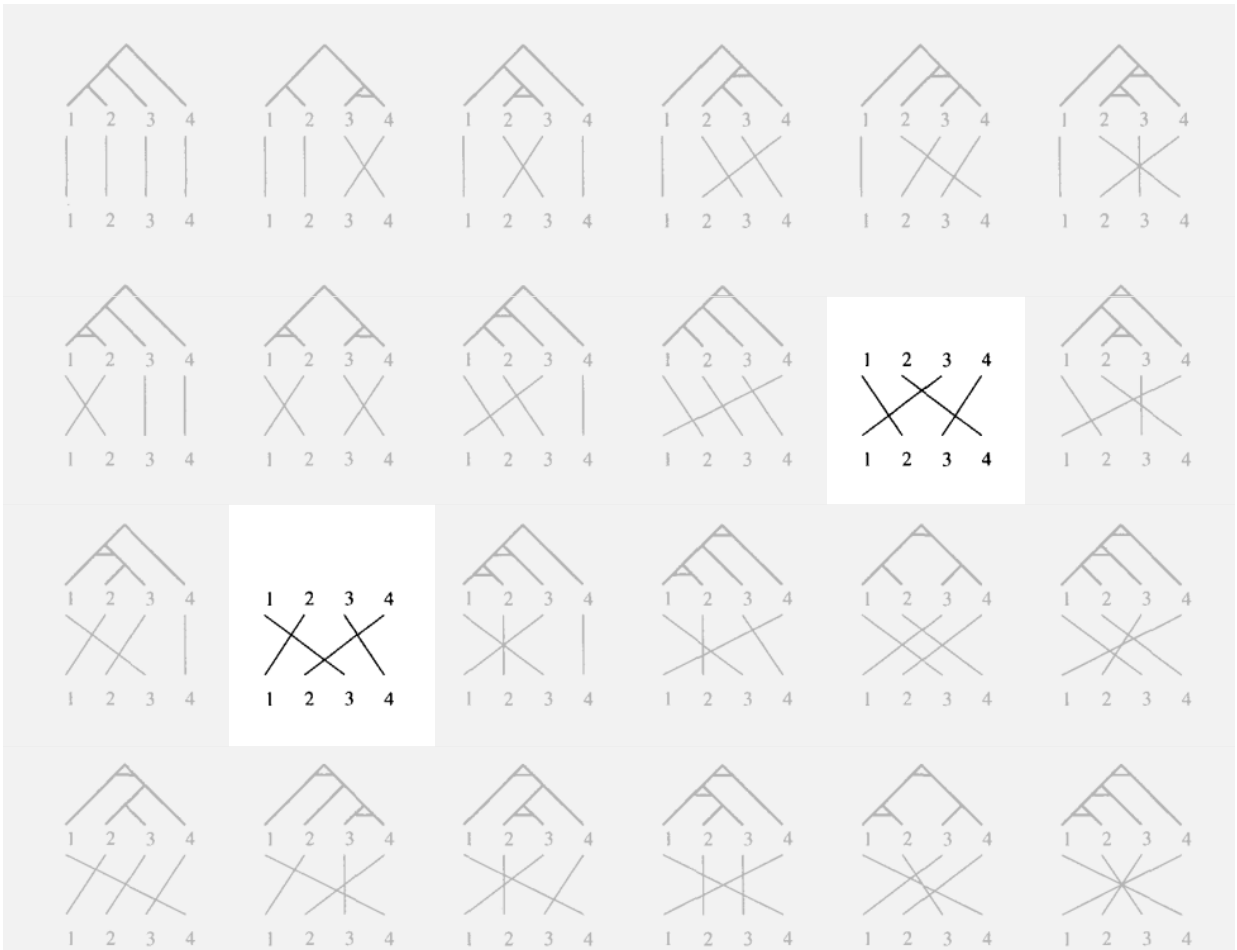



Now we are done with jointly generating two sentences!

管理局 将会 向 财政 司 负责 。

The Authority will be accountable to the Financial Secretary .

# Limitations with ITG



2 out of 24 cases where the ITG is unable to handle.

# Formal Syntax

**Hierarchical Phrase-Based Translation**

David Chiang*
Information Sciences Institute
University of Southern California

*We present a statistical machine translation model that uses hierarchical phrases—phr*
*that contain subphrases. The model is formally a synchronous context-free grammar but*
*learned from a parallel text without any syntactic annotations. Thus it can be seen as combinin*
*fundamental ideas from both syntax-based translation and phrase-based translation. We describe*
*our system's training and decoding methods in detail, and evaluate it for translation speed and*
*translation accuracy. Using BLEU as a metric of translation accuracy, we find that our system*
*performs significantly better than the Alignment Template System, a state-of-the-art phrase-*
*based system.*

**1. Introduction**

The alignment template translation model (Och and Ney 2004) and related phrase-based models advanced the state of the art in machine translation by expanding the basic unit of translation from words to **phrases**, that is, substrings of potentially unlimited size (but not necessarily phrases in any syntactic theory). These phrases allow a model to learn local reorderings, translations of multiword expressions, or insertions and deletions that are sensitive to local context. This makes them a simple and powerful mechanism for translation.

The basic phrase-based model is an instance of the noisy-channel approach (Brown et al. 1993). Following convention, we call the source language "French" and the target language "English"; the translation of a French sentence $f$ into an English sentence $e$ is modeled as:

$$\arg\max_e P(e \mid f) = \arg\max_e P(e, f) \tag{1}$$

$$= \arg\max_e (P(e) \times P(f \mid e)) \tag{2}$$

The phrase-based translation model $P(f \mid e)$ "encodes" $e$ into $f$ by the following steps:

1. segment $e$ into phrases $\bar{e}_1 \cdots \bar{e}_I$, typically with a uniform distribution over segmentations;

* 4676 Admiralty Way, Suite 1001, Marina del Rey, CA 90292, USA. E-mail: chiang@isi.edu. Much of the research presented here was carried out while the author was at the University of Maryland Institute for Advanced Computer Studies.

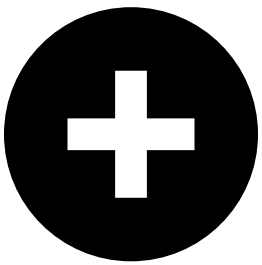
Submission received: 1 May 2006; accepted for publication: 3 October 2006.



Hierarchical Phrase based Translation

# Formal Syntax

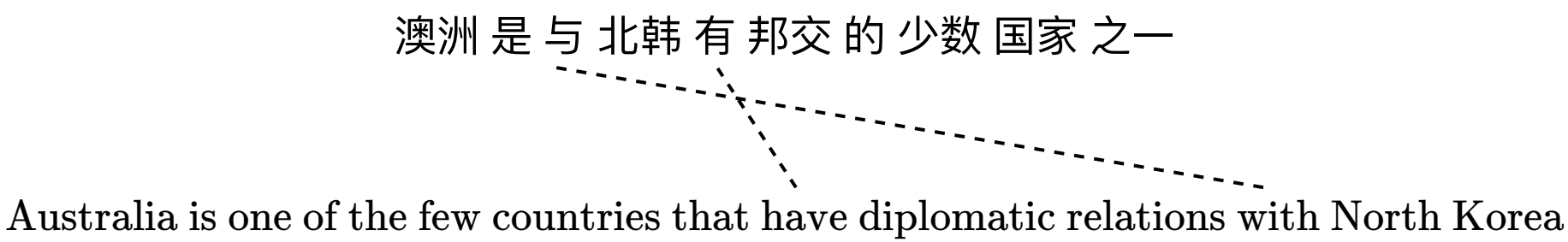
Phrase-based Translation



Synchronous Grammar

It combines the idea of phrase-based translation and synchronous parsing.

# Phrase-based SCFG




$$\mathbf{X} \rightarrow (\text{与 } \mathbf{X}_1 \text{ 有 } \mathbf{X}_2,\ \text{have } \mathbf{X}_2 \text{ with } \mathbf{X}_1)\ +7.2$$

$$\mathbf{X} \rightarrow (\text{北韩},\ \text{North Korea})\ -0.7$$

$$\mathbf{X} \rightarrow (\text{邦交},\ \text{diplomatic relations})\ +3.9$$

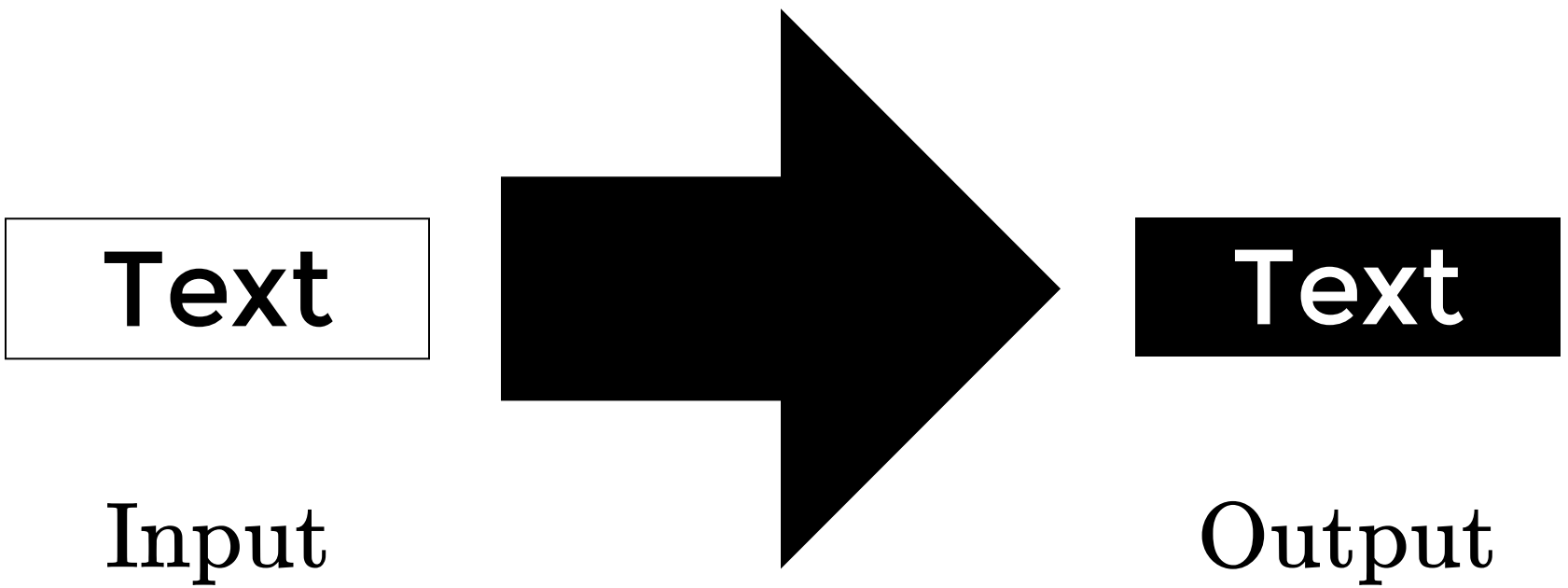
We are interested in a weighted phrase-level synchronous context-free grammar (SCFG)

# Score of Derivation

$$score(D) = \underbrace{\sum_{\mathbf{X} \rightarrow \langle \gamma, \alpha \rangle \in D} score(\mathbf{X} \rightarrow \langle \gamma, \alpha \rangle)}_{\text{Grammar rules}}$$

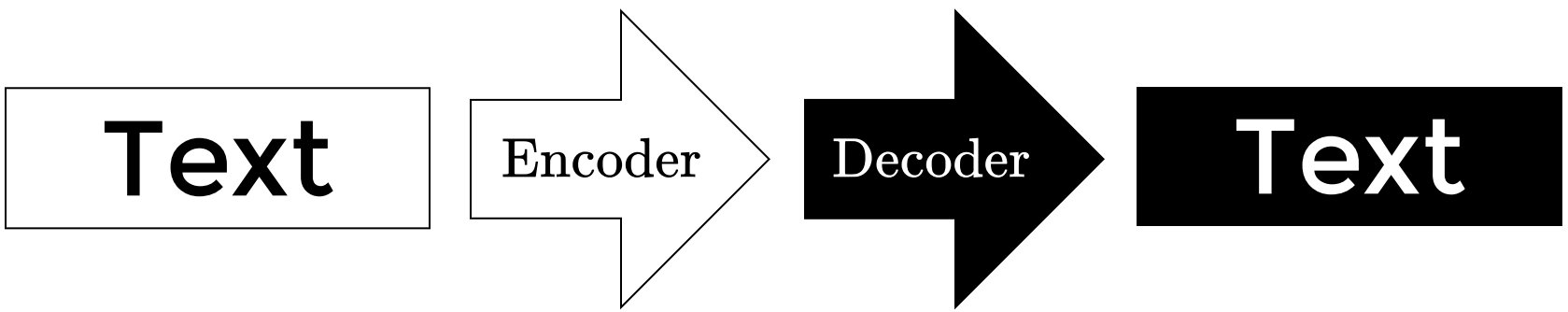
$$+ \quad \underbrace{\log q(e)}_{\text{Language model}}$$

# Text-to-Text Generation




Text-to-text problem with neural networks

# Encoder-Decoder

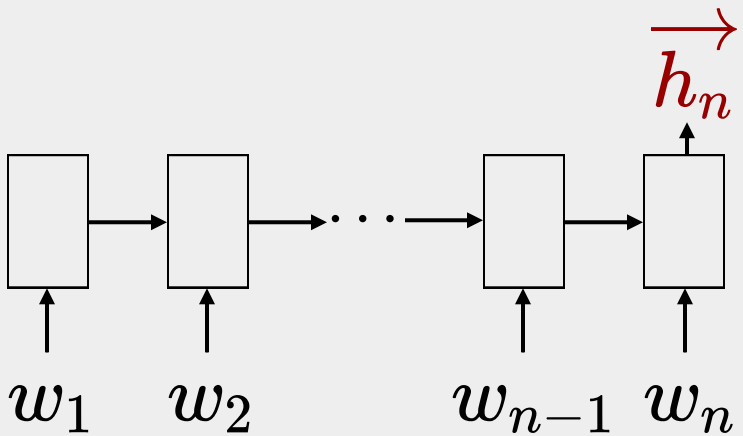



Input

Output

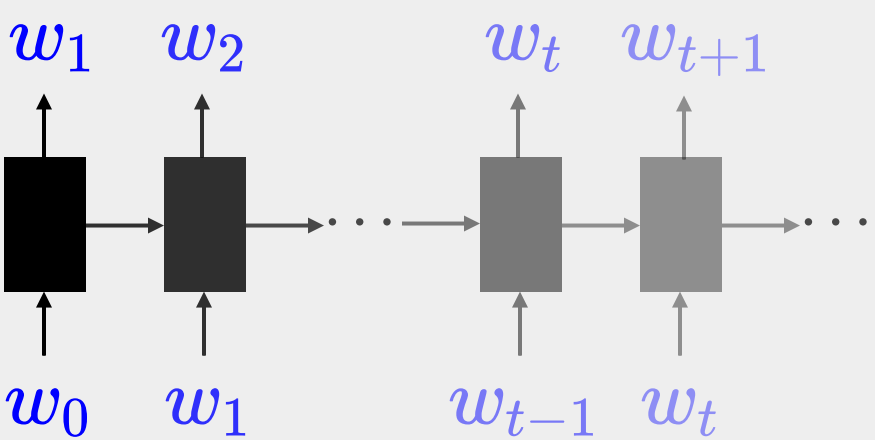
# LSTM

## Two applications of LSTM




Encoder

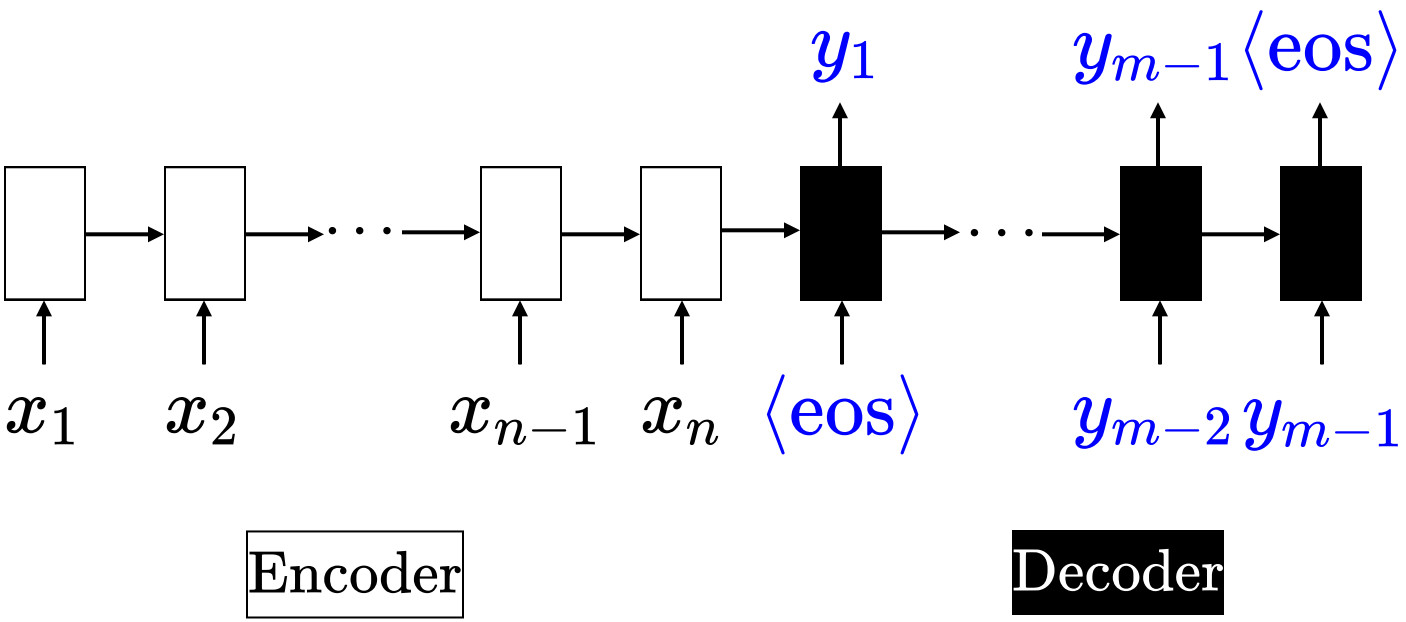
Sentence Embedding




Decoder

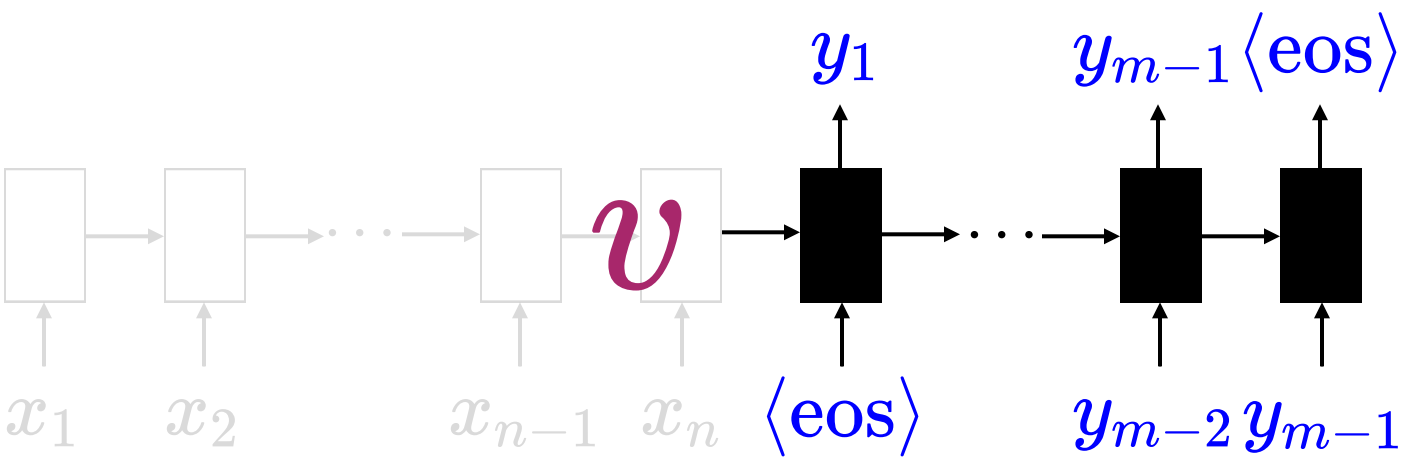
Sentence Generation

# Sequence to Sequence



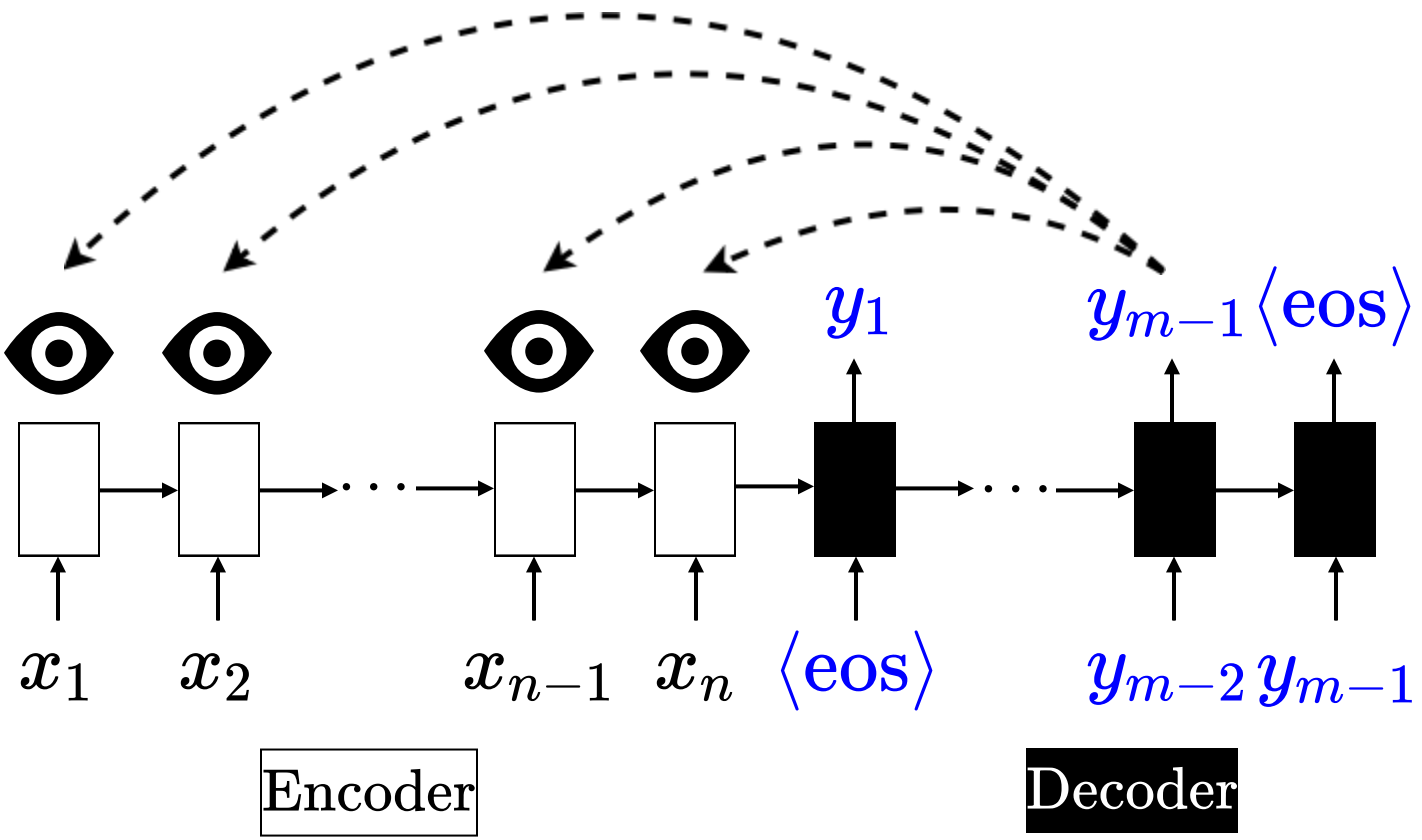

$$p(y_1, \ldots, y_m | x_1, \ldots, x_n)$$

# Sequence to Sequence




$$p(y_1, \ldots, y_m | x_1, \ldots, x_n)$$

# Seq2Seq with Attention (Bahdanau et al. 2015)




We may need to look at the input sequence when generating the outputs...

# Attention

## Cosine

$$\frac{d_i^T e_j}{||d_i|| \cdot ||e_j||}$$

(Graves et al. 2014)

## Concatenation

$$v^T \tanh(W[e_j; d_i])$$

(Bahdanau et al. 2015)

## Scaled Product

$$\frac{d_i^T e_j}{\sqrt{\delta}}$$

(Vaswani et al. 2017)

## Dot-Product

$$d_i^T e_j$$

(Luong et al. 2015)

## General Product

$$d_i^T W e_j$$

(Luong et al. 2015)

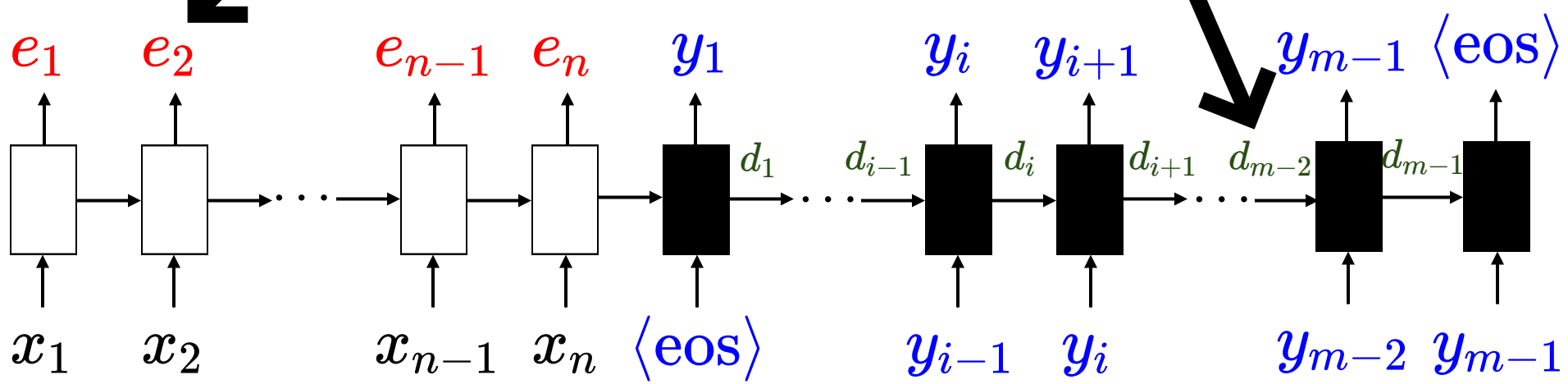
# Seq2Seq with Attention

$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i)$$

Learnable parameters

Hidden state for encoder

Hidden state for decoder




Encoder

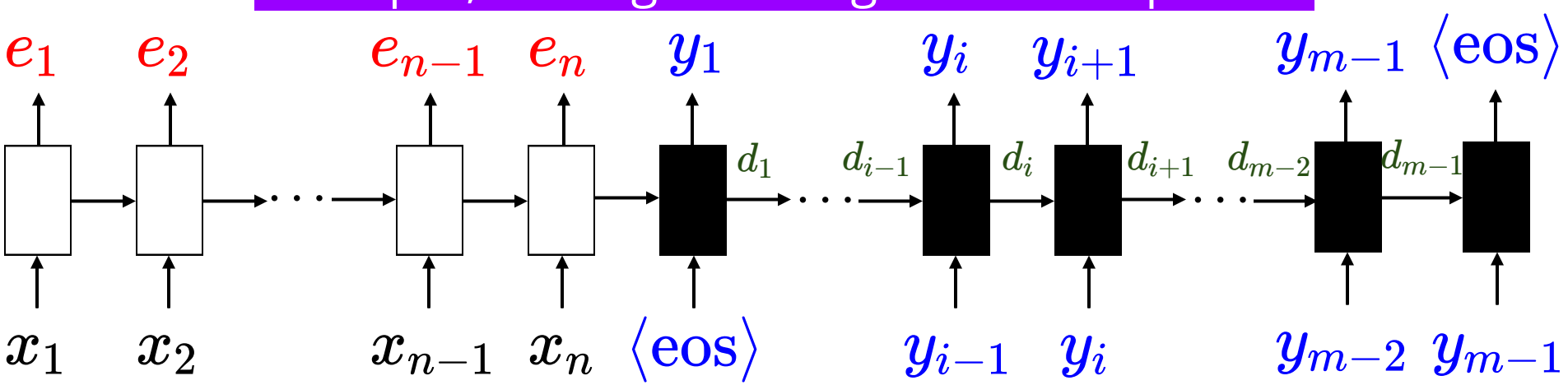
Decoder

# Seq2Seq with Attention

$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i)$$

$$a_j^i = \text{softmax}(u_j^i)$$

How much I would like to focus on the j-th input, when generating the i-th output




Encoder

Decoder

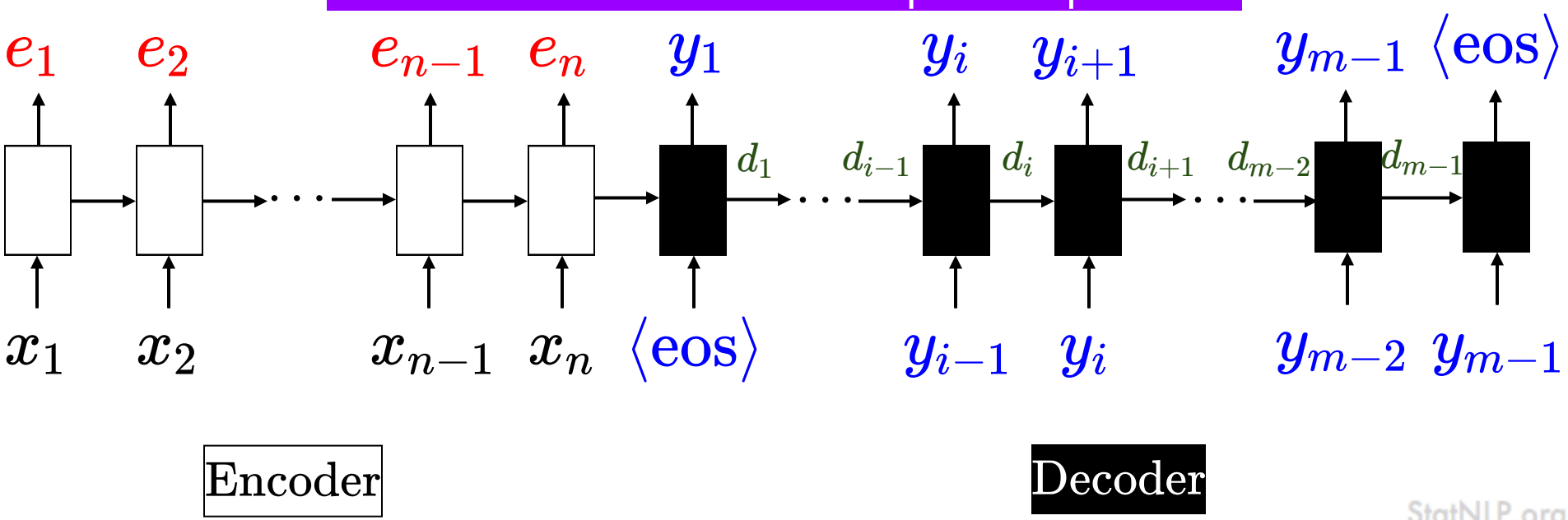
# Seq2Seq with Attention

$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i)$$

$$a_j^i = \text{softmax}(u_j^i)$$

$$d'_i = \sum_{j=1}^{n} a_j^i e_j$$

The weighted combination of hidden states from the entire input sequence



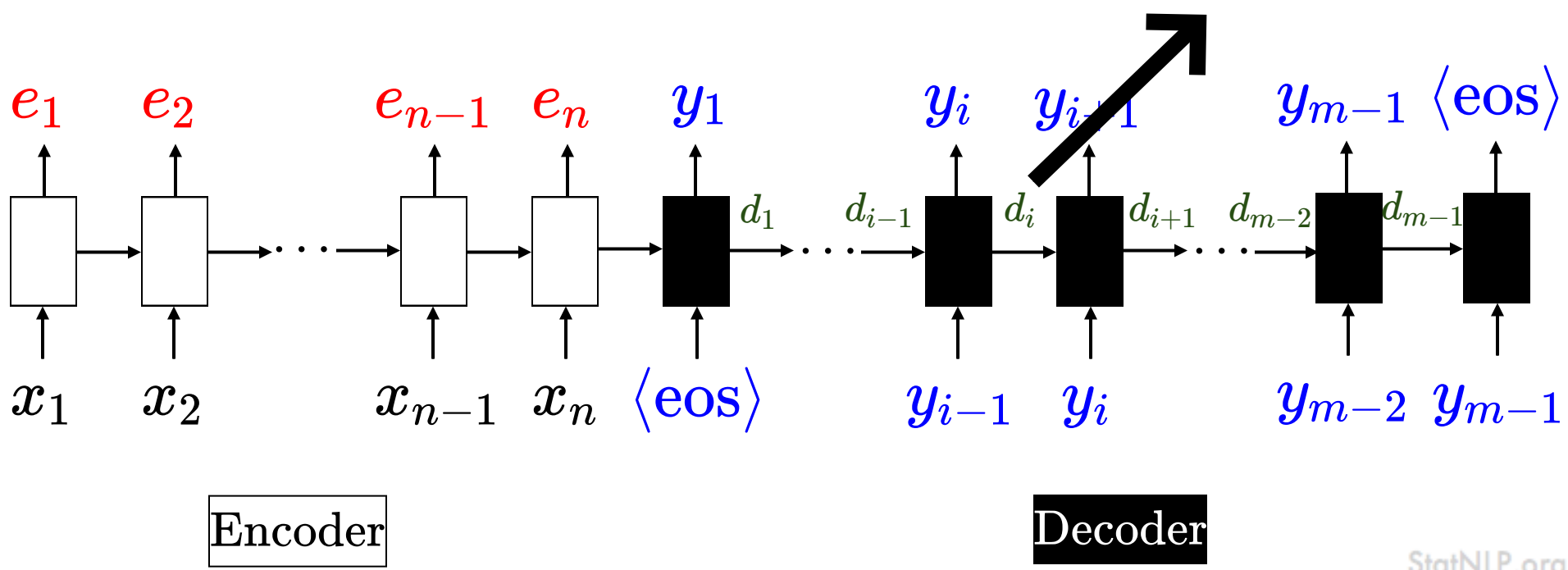
# Seq2Seq with Attention

$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i)$$

$$a_j^i = \text{softmax}(u_j^i)$$

$$d_i' = \sum_{j=1}^{n} a_j^i e_j$$

When predicting $y_i$ use $[d_i, d_i']$ instead



# Neural Models

**seq2seq**

recurrent network with attention

**fairseq**

convolutional network based encoder

**transformer**

self-attention network based encoder

# Neural Models

**seq2seq**

recurrent network with attention

**fairseq**

convolutional network based encoder

**transformer**

self-attention network based encoder

# Attention

Query

$$\begin{bmatrix} - & d_1^T & - \\ - & d_2^T & - \\ \vdots & \vdots & \vdots \\ - & d_m^T & - \end{bmatrix}$$

$Q$

Key

$$\begin{bmatrix} - & e_1^T & - \\ - & e_2^T & - \\ \vdots & \vdots & \vdots \\ - & e_n^T & - \end{bmatrix}$$

$K$

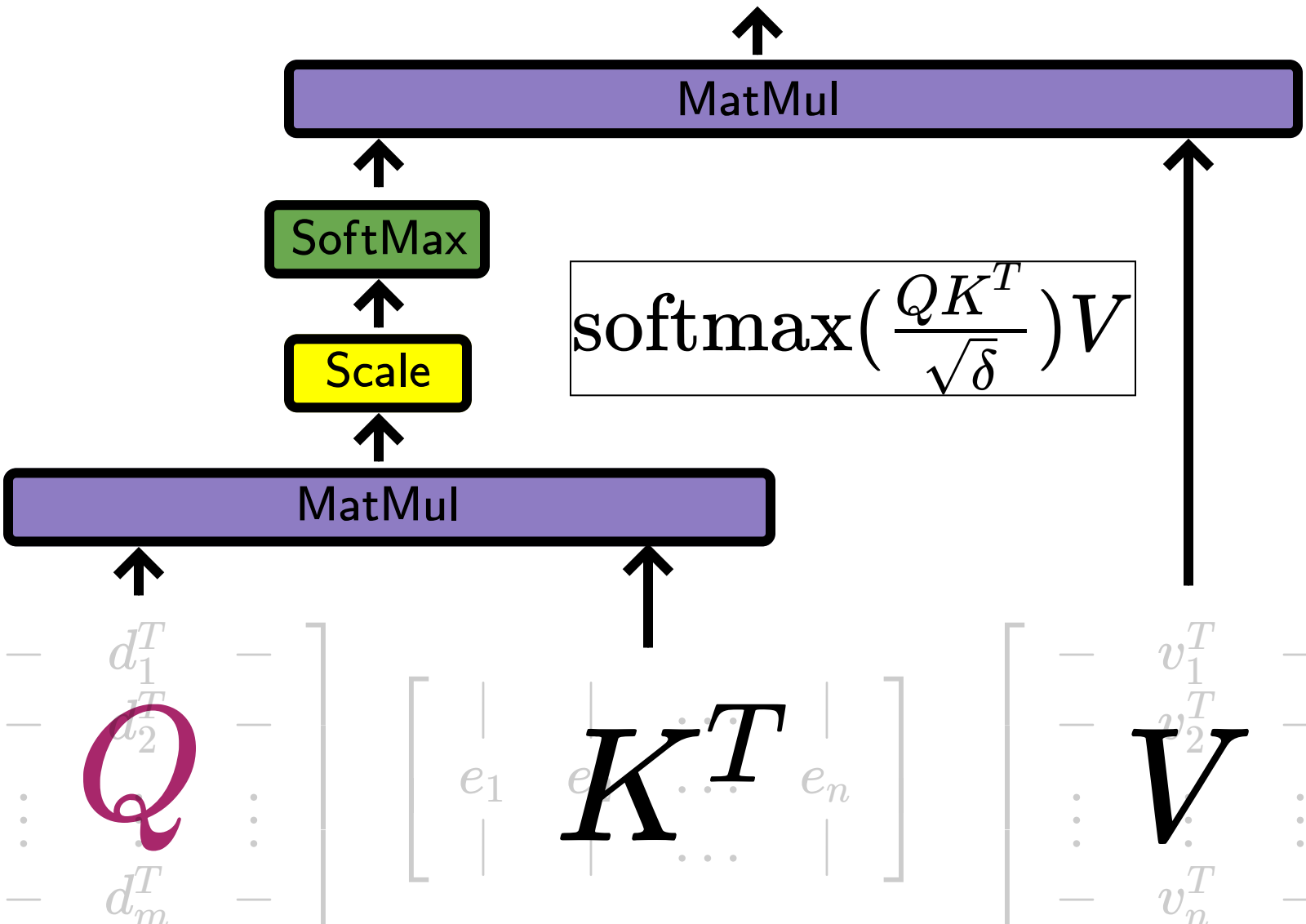
Value

$$\begin{bmatrix} - & v_1^T & - \\ - & v_2^T & - \\ \vdots & \vdots & \vdots \\ - & v_n^T & - \end{bmatrix}$$

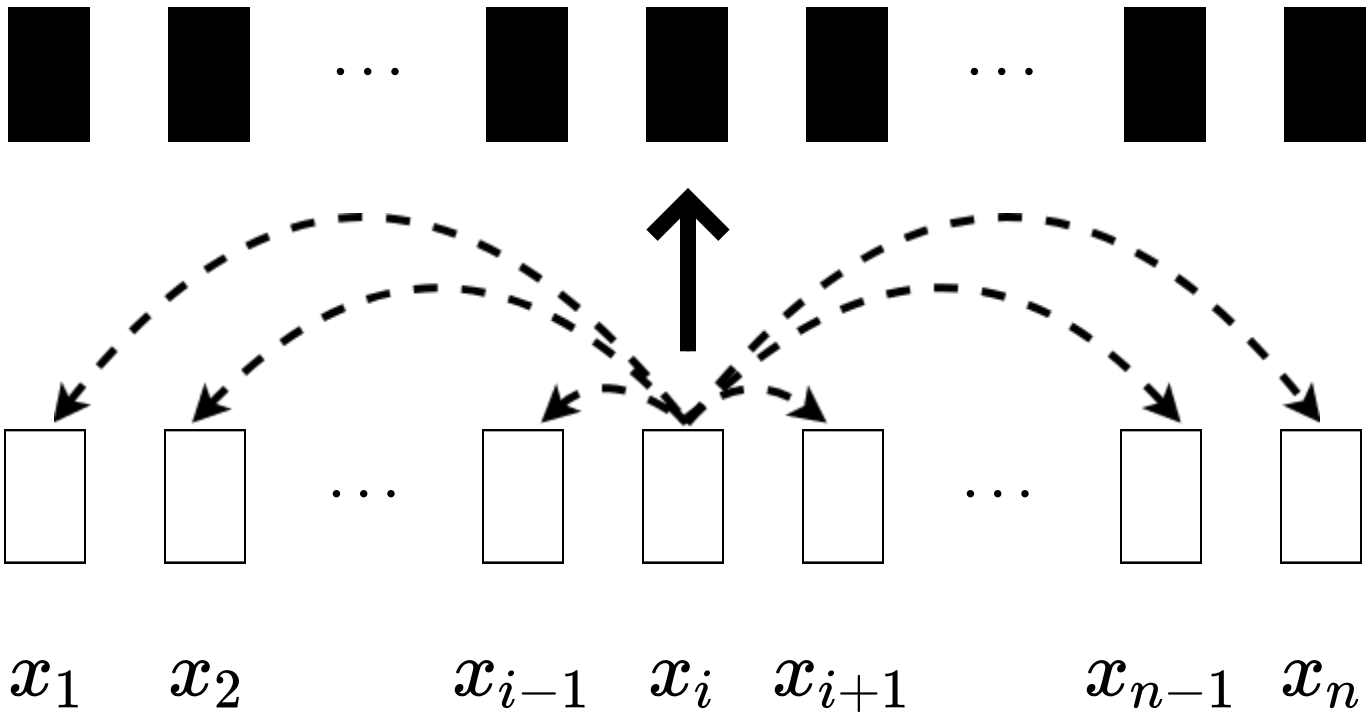
$V$

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{\delta}})V$$

# Scaled Dot Product



# Self Attention

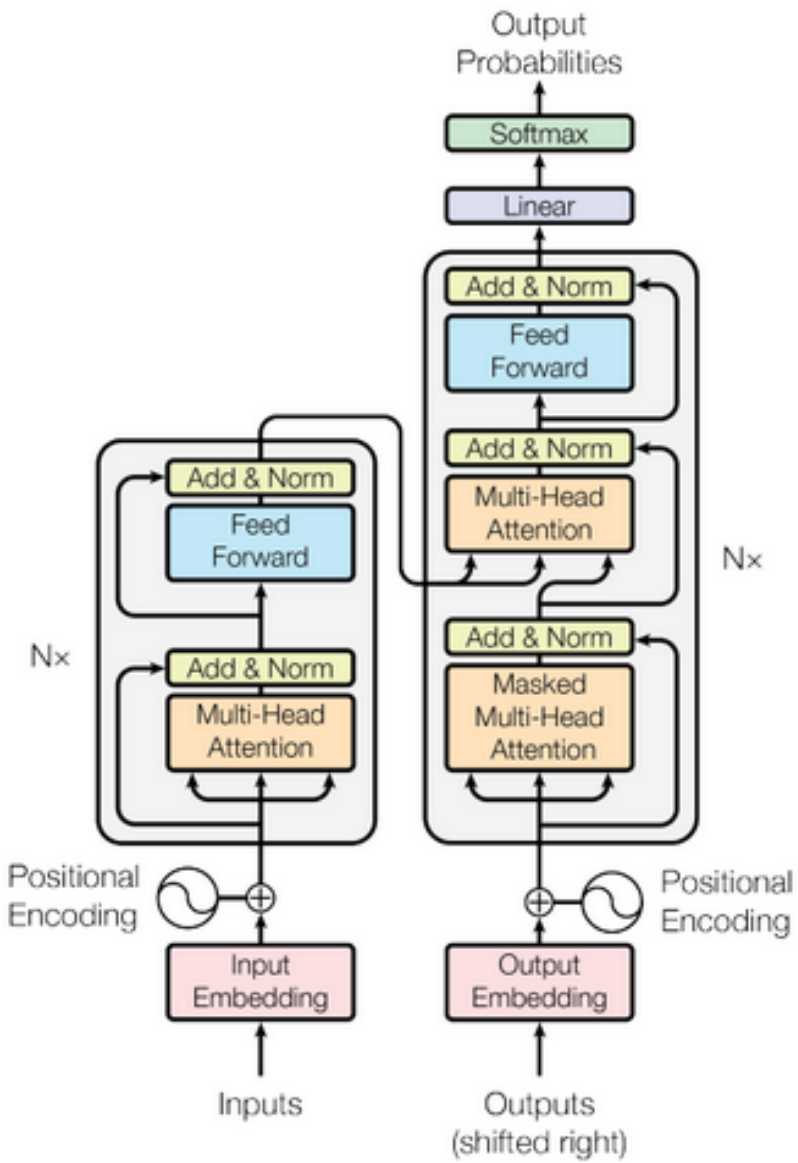



$$Q = K = V$$

Query, key and value matrices are the same!

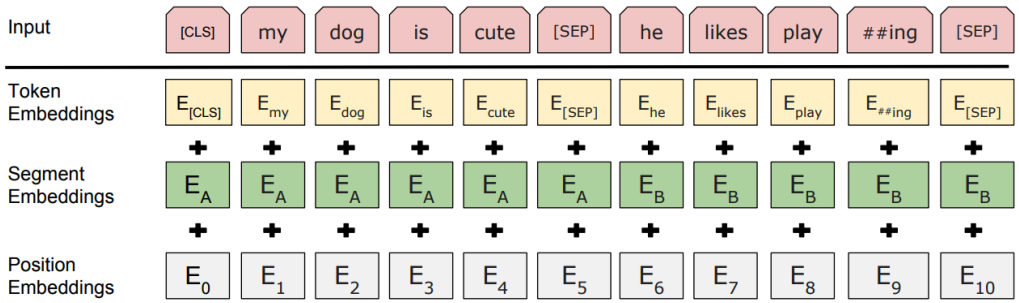
# Transformer

Encoder

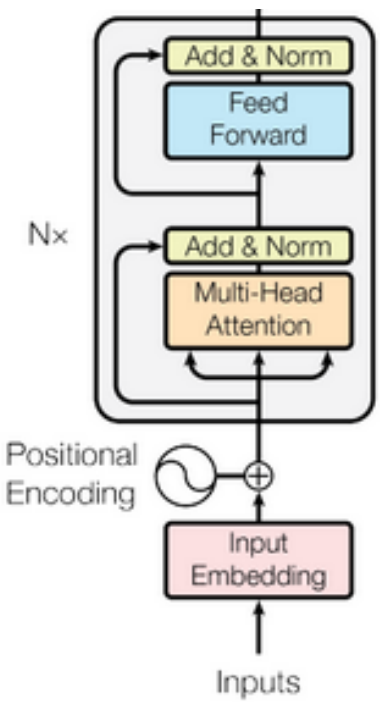



Decoder

# BERT




Encoder



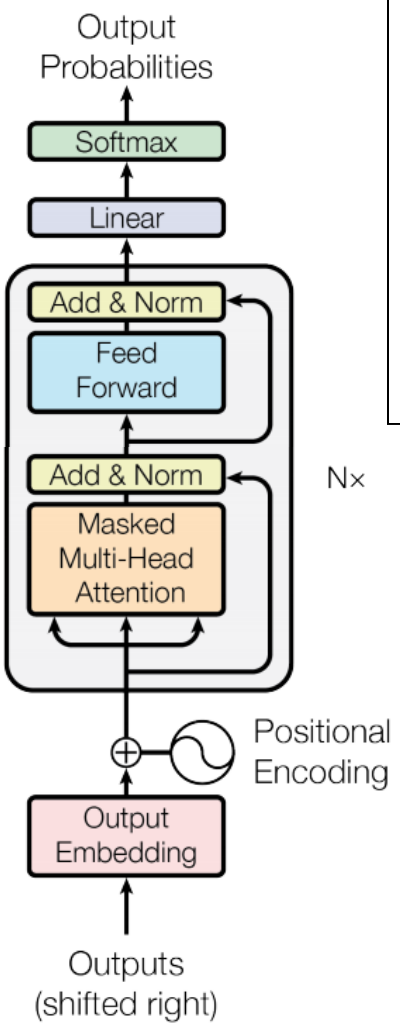

Task #1: Masked LM

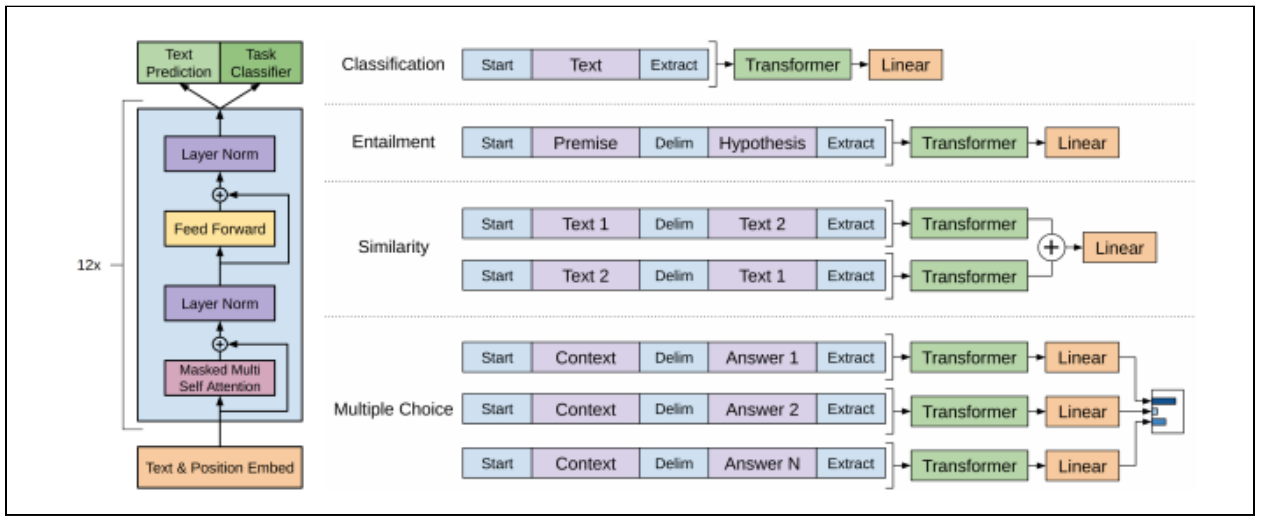
Useful for learning context representations within a sequence

Task #2: Next Sentence Prediction

Useful for tasks that involve identifying relations between multiple sentences

# GPT

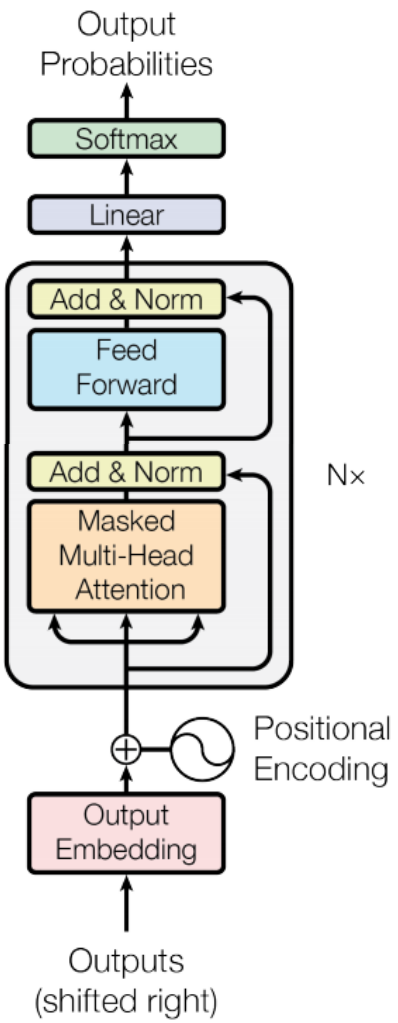



## Decoder

GPT: relies on task-specific fine-tuning, similar to BERT

# GPT-2, GPT-3



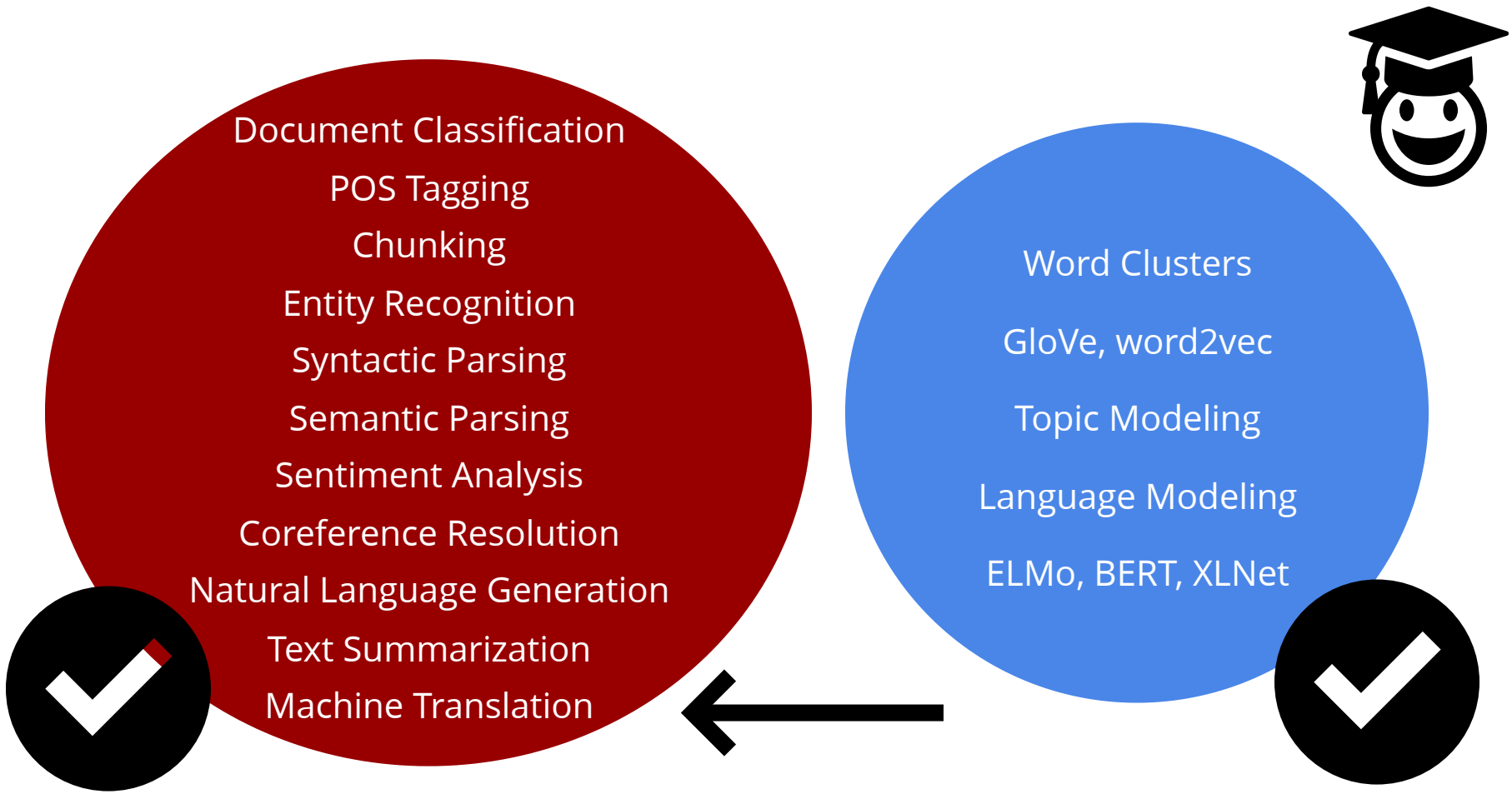

| Model | Size (# Params) |
|---|---|
| GPT-2 (Open AI) | 1.5 Billion |
| Megatron (NVidia) | 8 Billion |
| Turing NLG (Microsoft) | 17 Billion |
| GTP-3 (Open AI) | 175 Billion |

**Decoder**

GPT-2: LM as unsupervised multitask learner

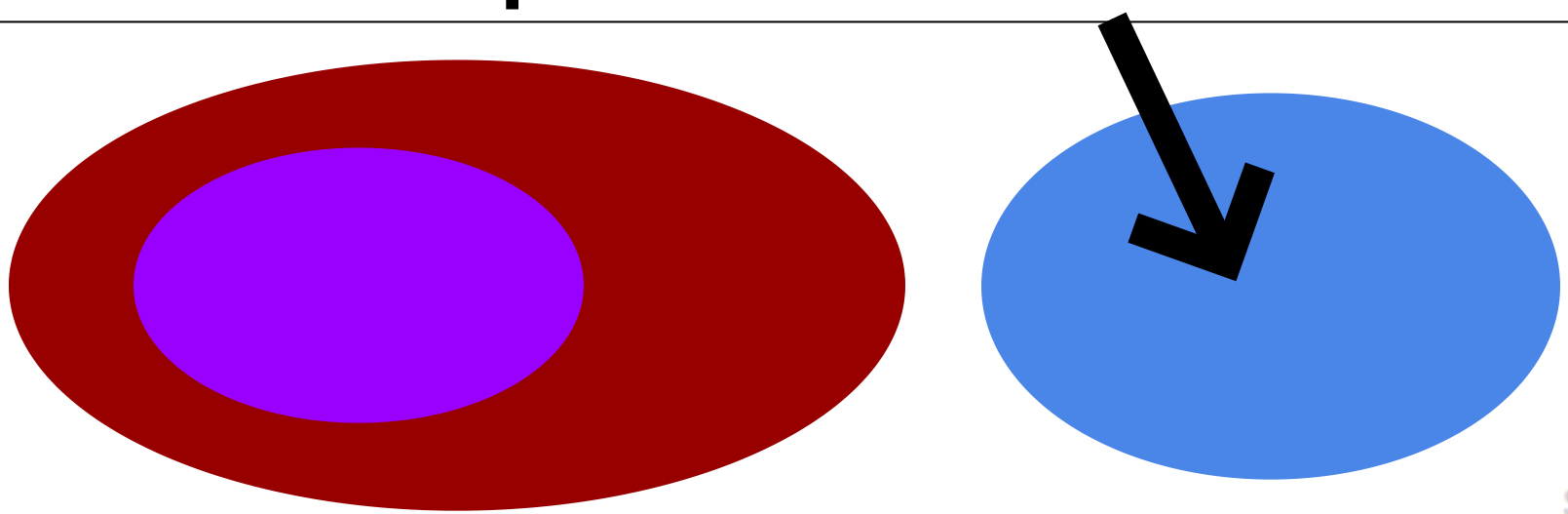
GPT-3: LM as few-short learner

# Tasks in NLP



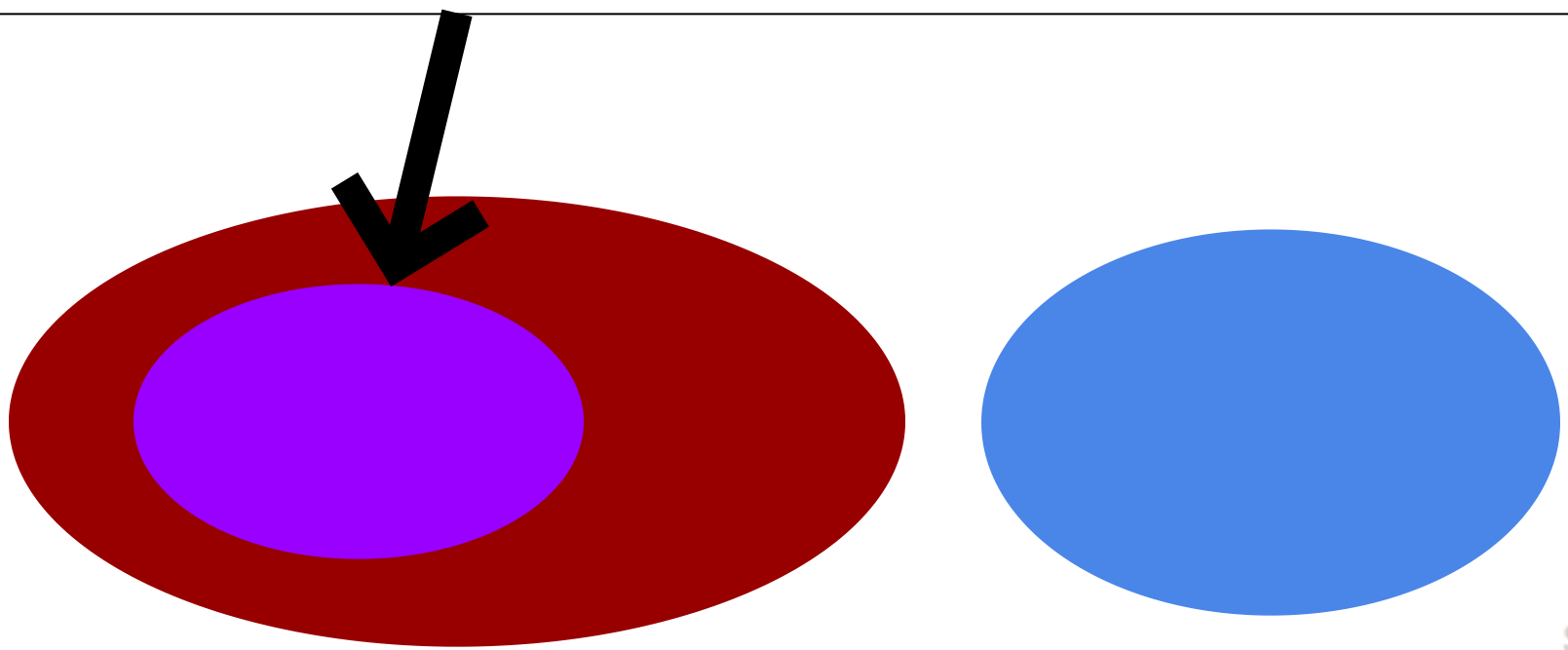
# Some Future Directions

Better methods for unsupervised learning of word/phrase/sentence representations



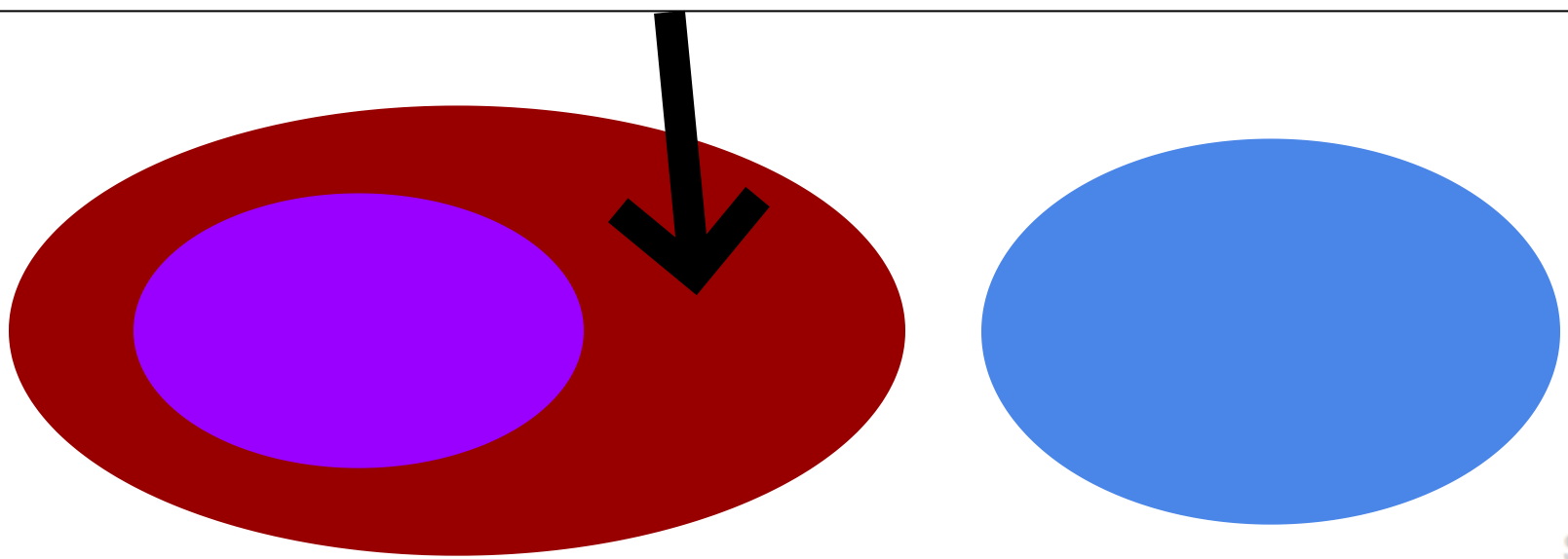
# Some Future Directions

Better understanding of the connection between Different Structured Prediction Models



# Some Future Directions

Better ways of training non structured prediction models



# Thank You

statnlp.org

# We Are Hiring

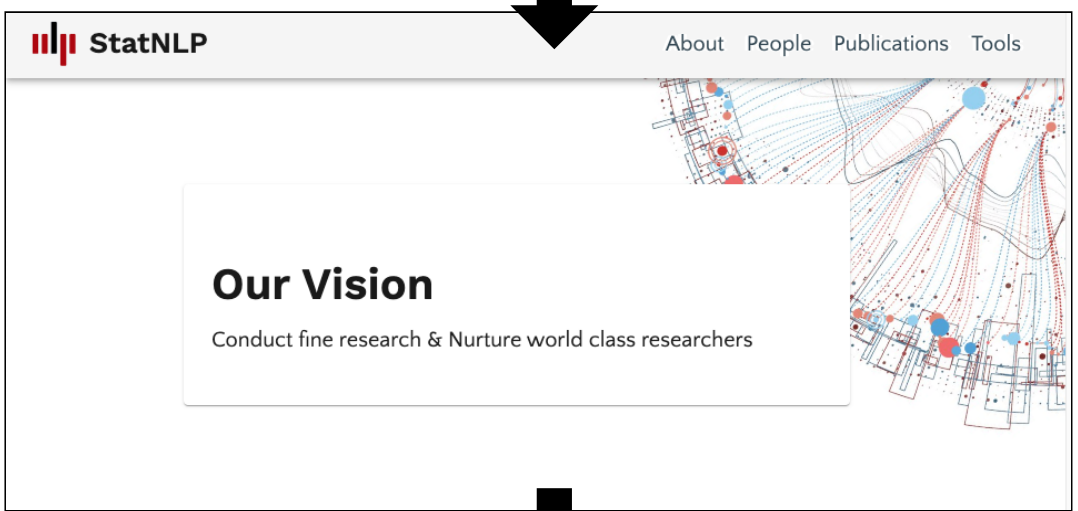


























StatNLP.org