Chapter 5 Natural Language Processing

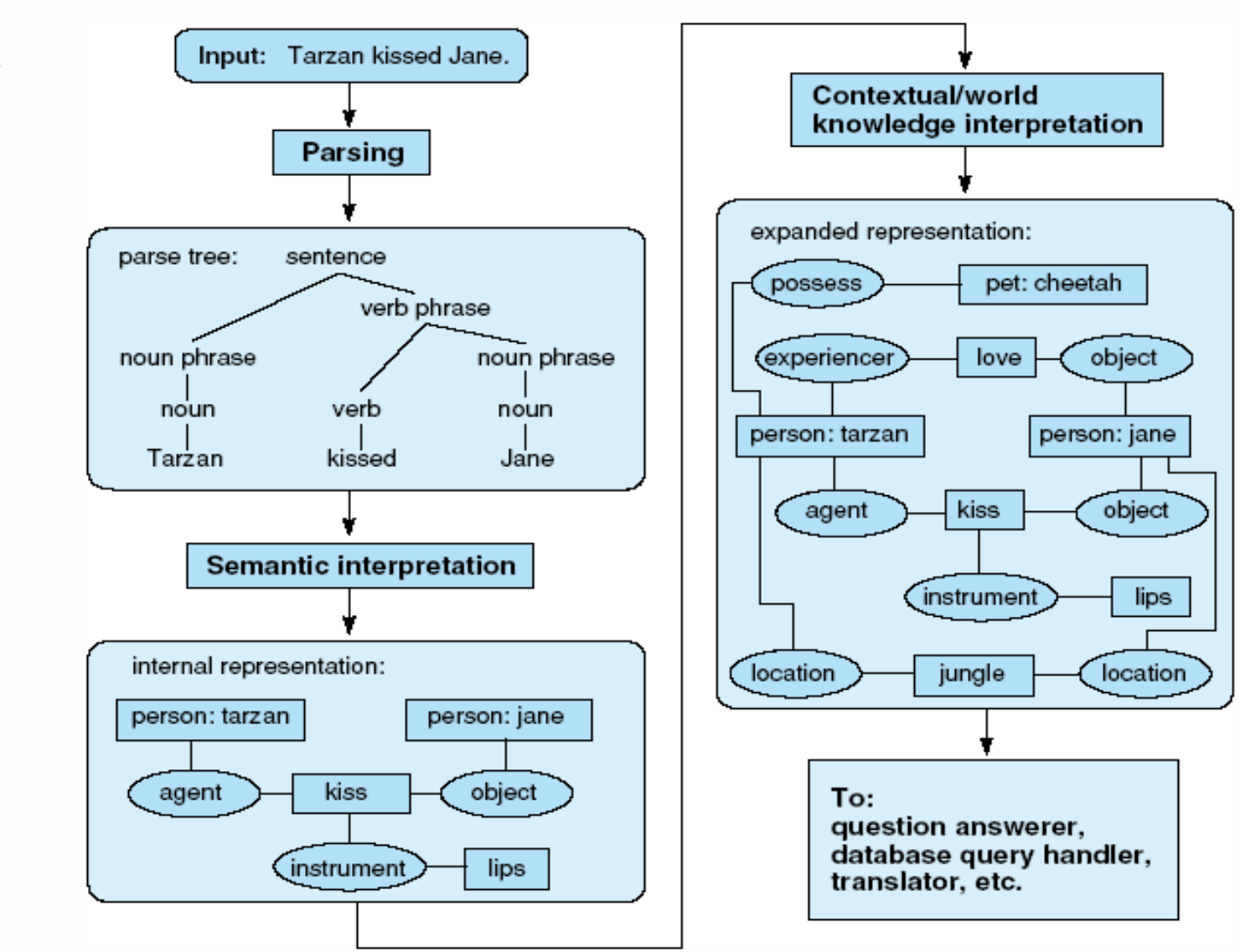
COMP 6721 Introduction of AI

Russell & Norvig – Chapter 23.1 + 23.2 + 23.3

Topics

- Stages of NL Understanding
- Contextualized Word Embeddings

Stages of NL Understanding



source: Luger (2005)

Stages of NL Understanding

Parsing (Syntax):

- What words are available in a language? ***gfiioudd / table***
- How to arrange words together?

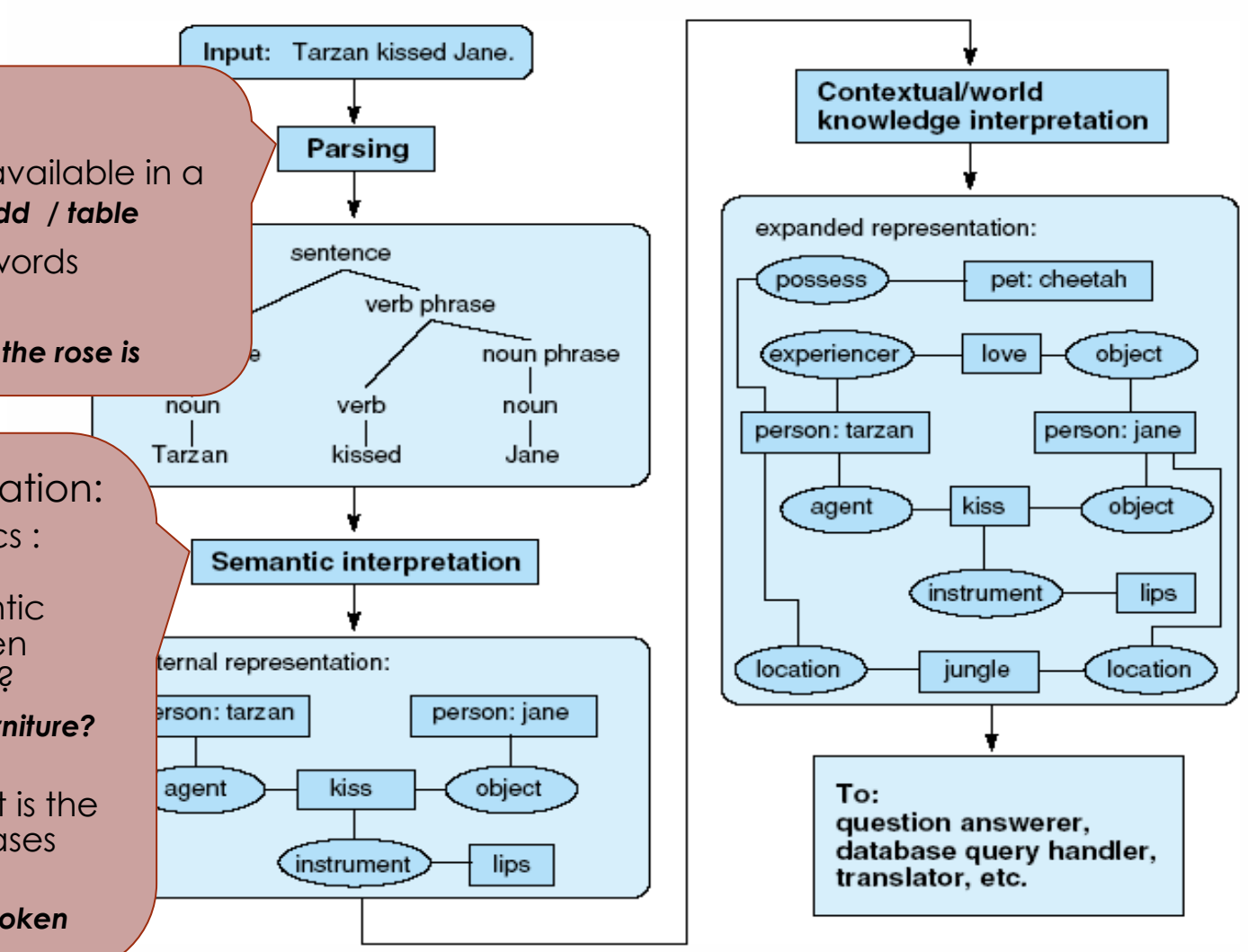
  the rose is red / red the rose is

Semantic interpretation:

- Lexical Semantics : What is the meaning/semantic relations between individual words?

  Chair: person? Furniture?
- Compositional Semantics: What is the meaning of phrases and sentences?

  The chair's leg is broken



source: Luger (2005)

Stages of NL Understanding

- Discourse Analysis

How to relate the meaning of sentences to surrounding sentences?

I have to go to the store. I need butter.

I have to go to the university. I need butter.

- Pragmatics

How people use language in a social environment?

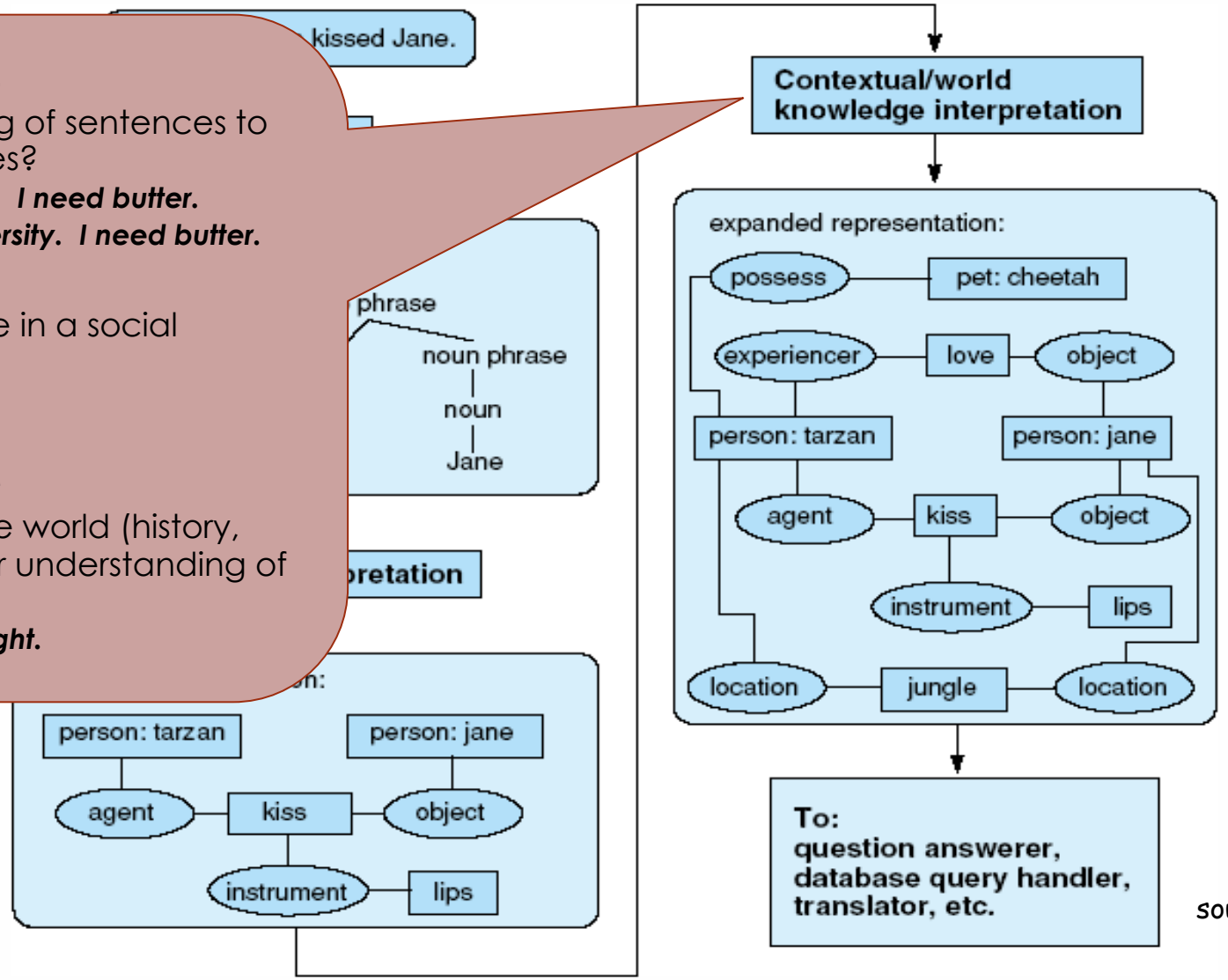
Do you have a child?

Do you have a quarter?

- World Knowledge

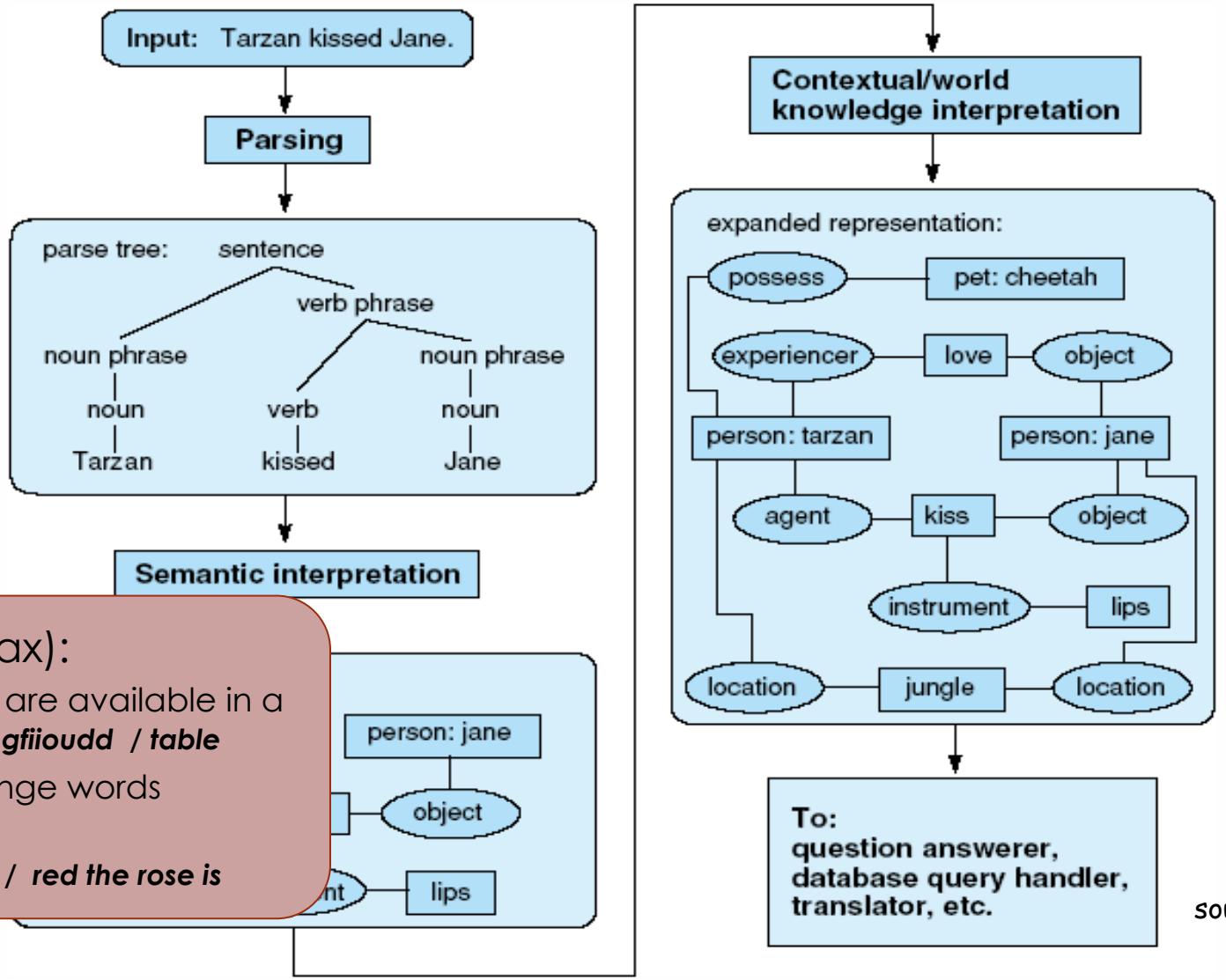
How knowledge about the world (history, facts, …) modifies our understanding of text?

Bill passed away last night.



source: Luger (2005)

Stages of NL Understanding



Parsing (Syntax):

- What words are available in a language? ***gfiioudd / table***
- How to arrange words together?

  the rose is red / red the rose is

source: Luger (2005)

Syntactic Categories

- Syntactic categories (common denotations) in NLP
- np - noun phrase
- vp - verb phrase
- s - sentence
- det - determiner (article)
- n - noun
- tv - transitive verb (takes an object)
- iv - intransitive verb
- prep - preposition
- pp - prepositional phrase
- adj - adjective

Syntactic Parsing

1. Assign the right part of speech (NOUN, VERB, …) to individual words in a text
2. Determine how words are put together to form correct sentences
    - The/DET rose/NOUN is/VERB red/ADJ.
    - Is/VERB red/ADJ the/DET rose/NOUN.

English Parts-of-Speech

- Open (lexical) class words
  - new words can be added easily
  - nouns, main verbs, adjectives, adverbs
  - some languages do not have all these categories
- Closed (functional) class words
  - generally function/grammatical words
  - *stop words*
  - ex. *the, in, and, over, beyond…*
  - relatively fixed membership
  - prepositions, determiners, pronouns, conjunctions, …

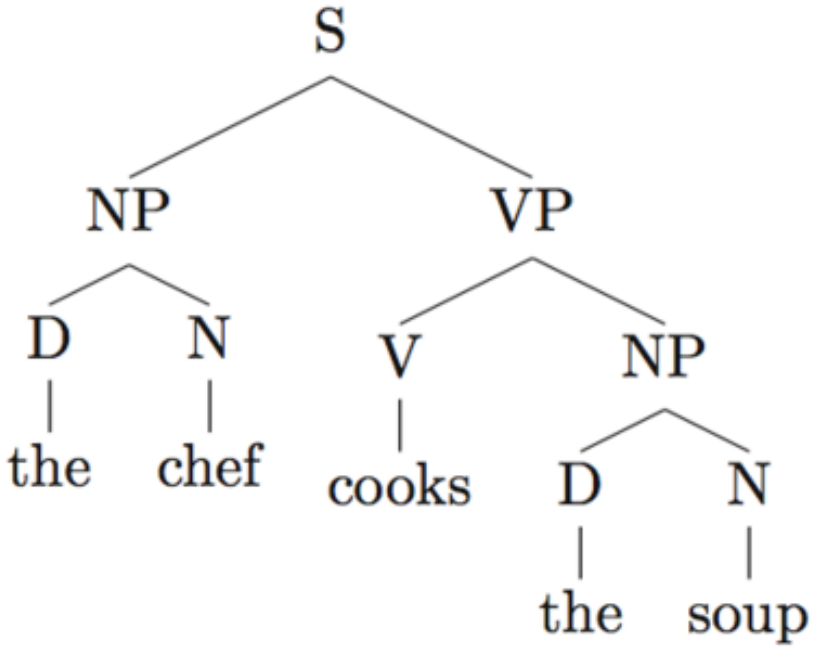
Smurf talk on youtube: https://www.youtube.com/watch?v=7BPx-vl8G00

Syntax

- How parts-of-speech are organised into larger syntactic constituents
- Main Constituents:
  - S: sentence — *The boy is happy.*
  - NP: noun phrase — *the little boy from Paris, Sam Smith, I,*
  - VP: verb phrase — *eat an apple, sing, leave Paris in the night*
  - PP: prepositional phrase — *in the morning, about my ticket*
  - AdjP: adjective phrase — *really funny, rather clear*
  - AdvP: adverb phrase — *slowly, really slowly*

A Parse Tree

- a tree representation of the application of the grammar to a specific sentence.



Context Free Grammar

- set of non-terminal symbols
  - constituents & parts-of-speech
  - S, NP, VP, PP, D, N, V, ...
- set of terminal symbols
  - words & punctuation
  - *cat, mouse, nurses, eat, ...*
- a non-terminal designated as the starting symbol
  - sentence S
- a set of re-write rules
  - having a single non-terminal on the LHS and one or more terminal or non-terminal in the RHS
  - S --> NP VP
  - NP --> Pronoun
  - NP --> PN
  - NP --> D N

Example

- Lexicon:

```
N --> flights | trip | breeze | morning      // noun
V --> is | prefer | like                      // verb
Adj --> direct | cheapest | first             // adjective
Pro --> me | I | you | it                     // pronoun
PN --> Chicago | United | Los Angeles         // proper noun
D --> the | a | this                          // determiner (article)
Prep --> from | to | in                       // preposition
Conj --> and | or | but                       // conjunction
```

- Grammar:

```
S --> NP VP                                   // I + prefer United
NP --> Pro | PN | D N                         // I, Chicago, the morning
VP --> V | V NP | V NP PP                     // is, prefer + United,
PP --> Prep NP                                // to Chicago, to I ??
```

Parsing

- parsing:
  - goal: assign syntactic structures to a sentence
  - result: (set of) parse trees
- we need:
  - a grammar: description of the language constructions
  - a parsing strategy:
    - how the syntactic analysis are to be computed

Parsing Strategies

- Parsing is seen as a search problem through the space of all possible parse trees
  - bottom-up (data-directed): words --> grammar
  - top-down (goal-directed): grammar --> words
  - breadth-first: compute all paths in parallel
  - depth-first: exhaust 1 path before considering another
  - Heuristic search

Example: John ate the cat

Grammar:

```
S --> NP VP
NP --> Pro | PN | D N
VP --> V | V NP | V NP PP
PP --> Prep NP
```

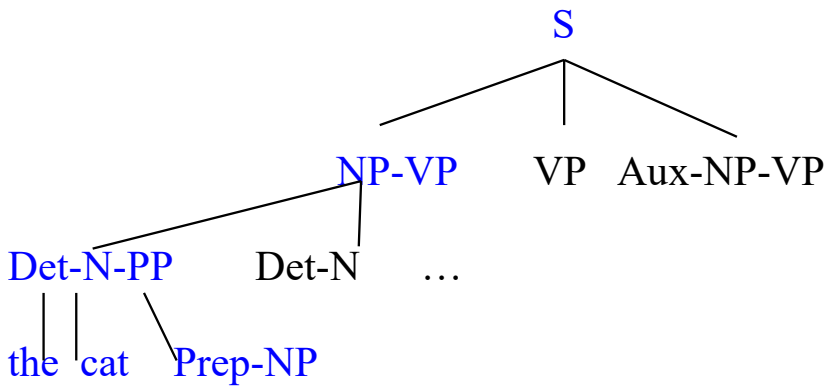
- Bottom-up parsing / breadth first
  1. John ate the cat
  2. PN ate the cat
  3. PN V the cat
  4. PN V ART cat
  5. PN V ART N
  6. NP V ART N
  7. NP V NP
  8. NP VP
  9. S

- Top-down parsing / depth first
  1. S
  2. NP VP
  3. PN VP
  4. John VP
  5. John V NP
  6. John ate NP
  7. John ate ART N
  8. John ate the N
  9. John ate the cat

Depth-first vs Breadth-first

the cat eats the mouse.

- depth-first: exhaust 1 path before considering another



- breadth-first:
  - compute 1 level at a time
- Heuristic search:
  - e.g. preference to shorter rules

Grammar:

(1) S --> NP VP
(2) S --> VP
(3) S --> Aux NP VP
(4) NP --> Det N PP
(5) NP -- > Det N
(6) PP -- > Prep N
…

Lexicon:

(10) Det --> the
(11) N --> cat
(12) VB --> eats
…

Summary of Parsing Strategies

	Depth First	Breath First	Heuristic Search
Top down	✓	✓	✓
Bottom up	✓	✓	✓

Problem: Multiple Parses

- Many possible parses for a single sentence happens very often…
  - Prepositional phrase attachment (PP-attachment)
    - *We painted the wall with cracks.*
    - *The man saw the boy with the telescope.*
    - *I shot an elephant in my pyjamas.*
  - Conjunctions and appositives
    - *Maddy, my dog, and Samy*

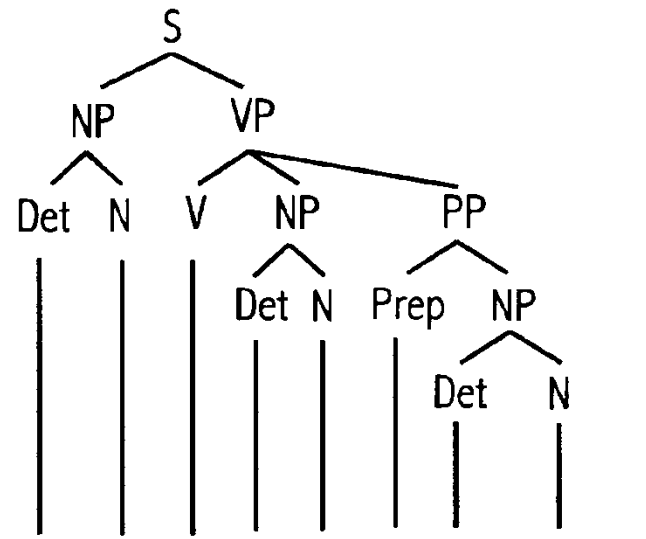
    -- > (Maddy, my dog), and (Samy)

    -- > (Maddy), (my dog), and (Samy)

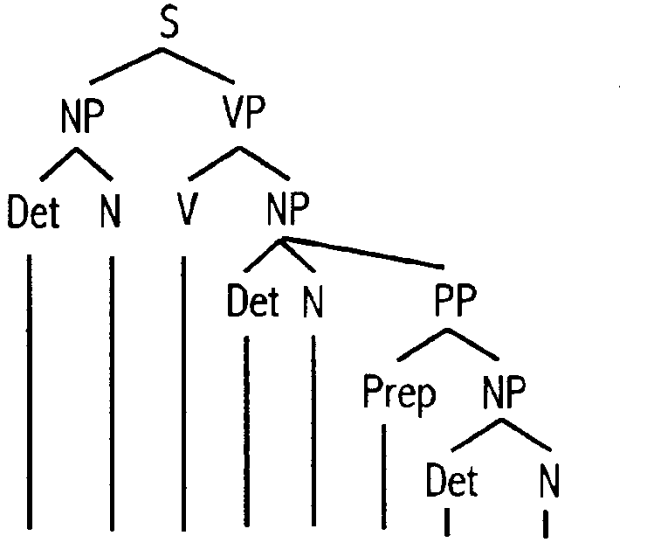
- These phenomena can quickly increase the number of possible parse trees!

Prepositional Phrase Attachment

The man saw the boy with the telescope.



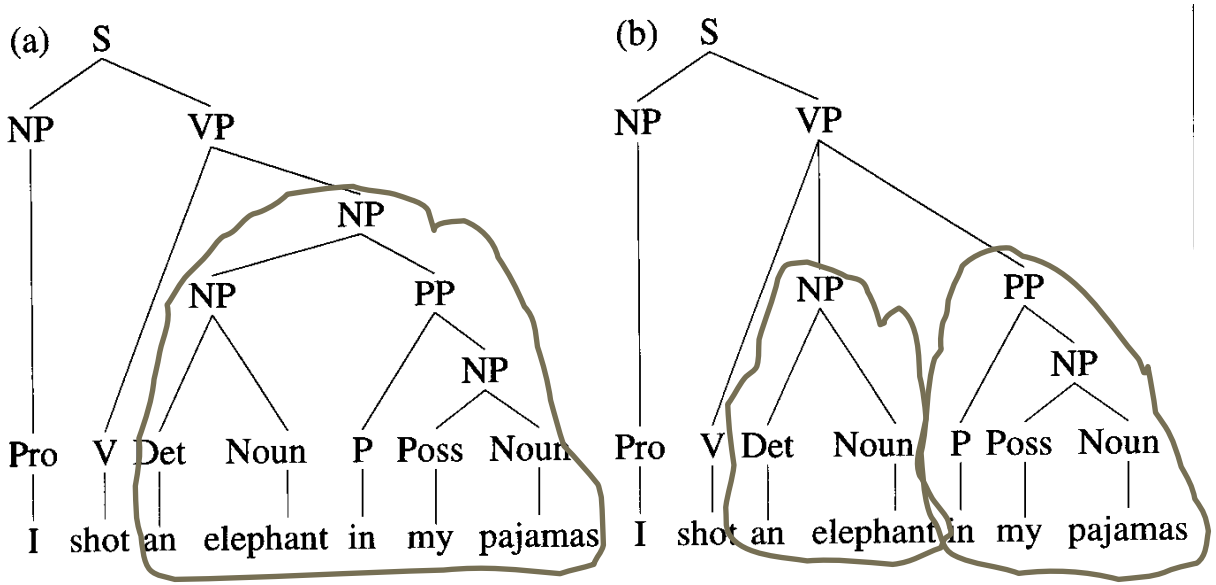
Correct parse 1



Correct parse 2

Probabilistic Parsing

"One morning I shot an elephant in my pyjamas. How he got into my pyjamas, I don't know." G. Marx, *Animal Crackers*, 1930.



- Sentences can be very ambiguous…
  - A non-probabilistic parser may find a large set of possible parses
  - --> need to pick the most probable parse one from the set

Probabilistic Context-Free Grammar

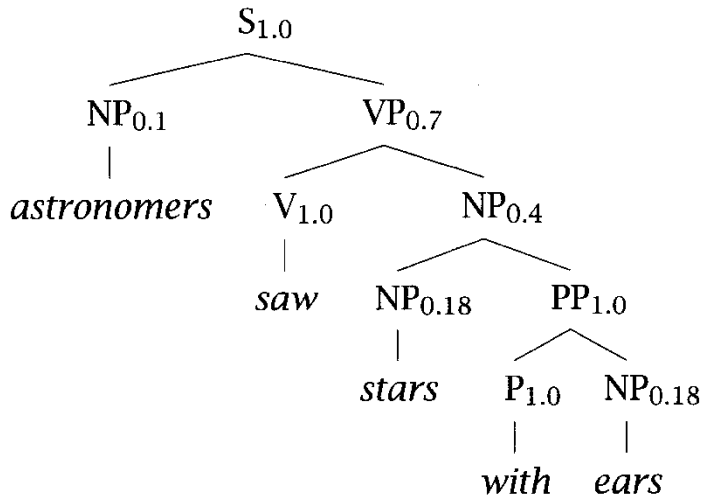
S → NP VP	1.0	NP → NP PP	0.4
PP → P NP	1.0	NP → *astronomers*	0.1
VP → V NP	0.7	NP → *ears*	0.18
VP → VP PP	0.3	NP → *saw*	0.04
P → *with*	1.0	NP → *stars*	0.18
V → *saw*	1.0	NP → *telescopes*	0.1

- Intuitively, P(VP → V NP) is:
  - the probability of expanding VP by a V NP, as opposed to any other rules for VP
- So for:
  - VP: $\forall i \; \Sigma_i$ P(VP) = 0.7 + 0.3 = 1
  - NP: $\forall i \; \Sigma_i$ P(NP) = 0.4 + 0.1 + 0.18 + 0.04 + 0.18 + 0.1 = 1

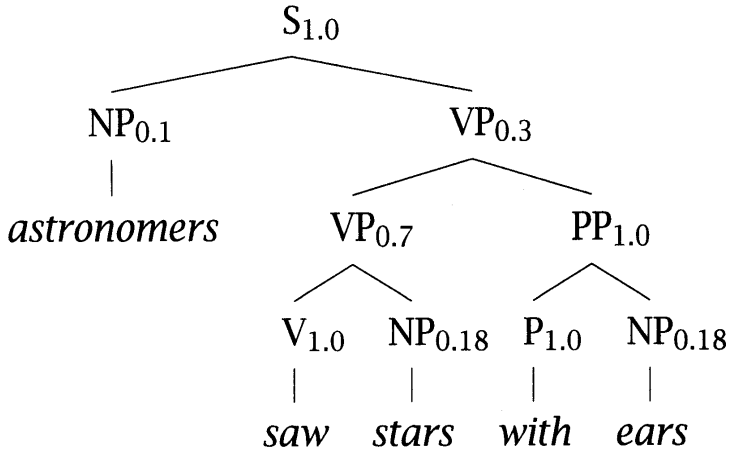
Probability of A Parse Tree

- Product of the probabilities of the rules used in subtrees
- Ex: "*Astronomers saw stars with ears.*"

t_1:



t_2:



$P(t_1)$ = 1x0.1x0.7x1x0.4x0.18x1x1x0.18
= .0009072

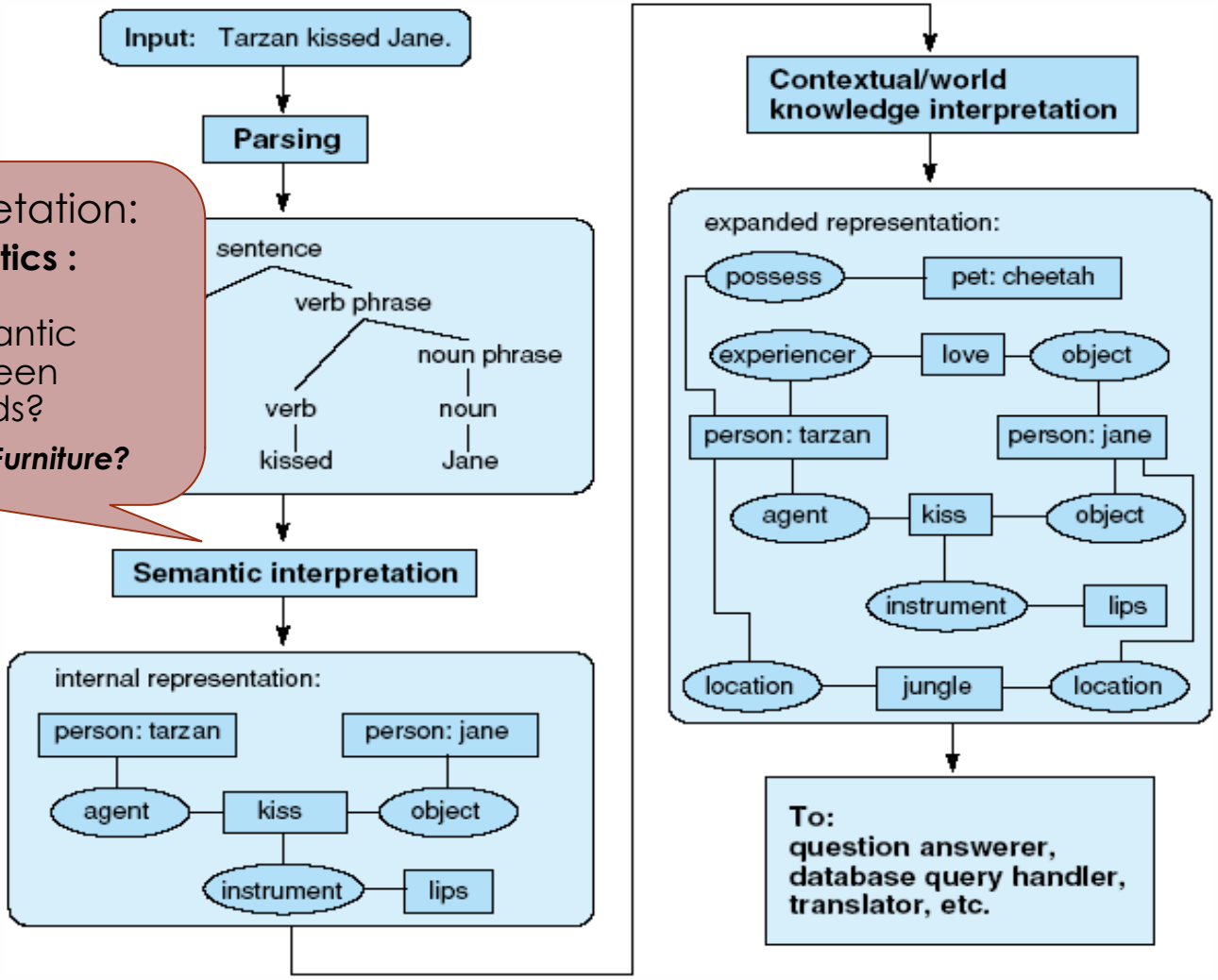
$P(t_2)$ = 1x0.1x0.3x0.7x1x1x0.18x1x0.18
= .0006804

Stages of NL Understanding

Semantic interpretation:

- **Lexical Semantics :** What is the meaning/semantic relations between individual words?

  Chair: person? Furniture?



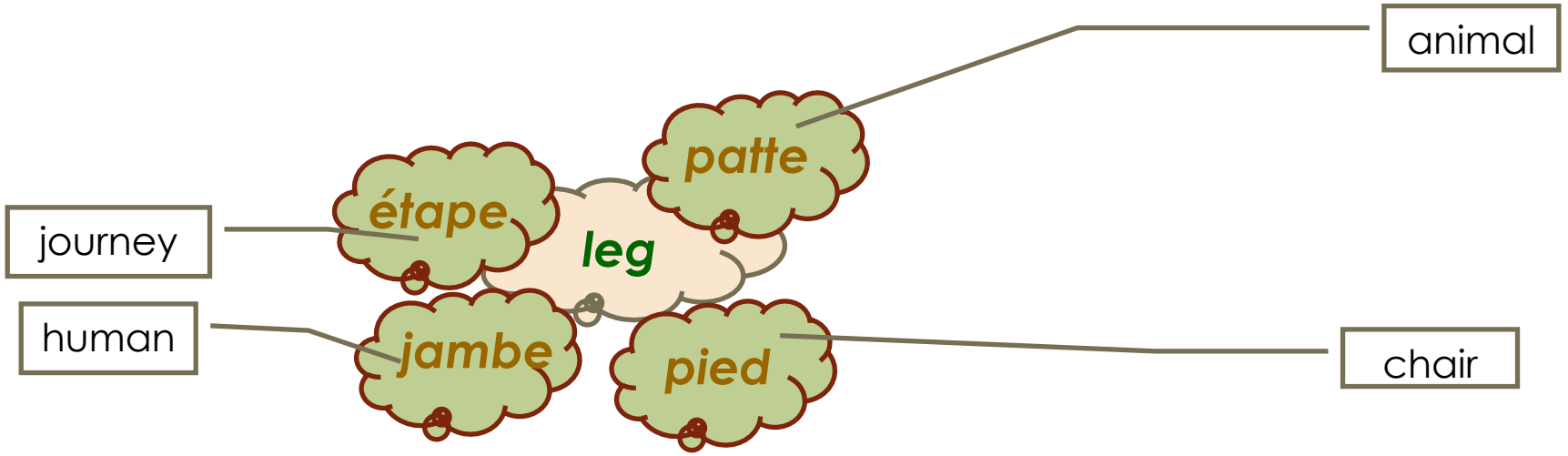
Semantic Interpretation

- Map sentences to some representation of its meaning
  - Representation: logics, semantic network, frames,…

1. Lexical Semantics
   - i.e. Meaning of individual words
2. Compositional Semantics
   - i.e. Meaning of combination of words

Lexical Semantics

- The meaning of individual words
  - A word may denote different things (ex. chair)
  - The meaning/sense of words is not clear-cut
  - E.g. Overlapping of word senses across languages



Word Sense Disambiguation (WSD)

- Determining which sense of a word is used in a specific sentence
  - *I went to the bank of Montreal and deposited 50$.*
  - *I went to the bank of the river and dangled my feet.*

WSD as A Classification Problem

- WSD can be viewed as typical classification problem
  - use machine learning techniques (ex. Naïve Bayes classifier, decision tree) to train a system
  - that learns a classifier (a function *f*) to assign to unseen examples one of a fixed number of senses (categories)
- Input:
  - Target word: The word to be disambiguated
  - Features?
- Output:
  - Most likely sense of the word

Features of WSD

- intuition:
  - sense of a word depends on the sense of surrounding words
- ex: *bass* = fish, musical instrument, ...

Surrounding words	Most probable sense
...river...	fish
...violin...	instrument
...salmon...	fish
...play...	instrument
...player...	instrument
...striped...	fish

- So use a window of words around the target word as features

Features of WSD

- Take a window of *n* words around the target word
- Encode information about the words around the target word
  - *An electric guitar and **bass** player stand off to one side, not really part of the scene, just as a sort of nod to people expectations perhaps.*

Naïve Bayes WSD

- Goal: choose the most probable sense s* for a word given a vector V of surrounding words
- Feature vector V contains:
  - Features: words [*fishing, big, sound, player, fly, rod,* …]
  - Value: frequency of these words in a window before & after the target word [0, 0, 0, 2, 1, 0, …]
- Bayes decision rule:
  - $s^* = \text{argmax}_{s_k} P(s_k \mid V)$
  - where:
    - S is the set of possible senses for the target word
    - s_k is a sense in S
    - V is the feature vector

Naïve Bayes WSD

$$s^* = \underset{s_k}{\operatorname{argmax}} \left(\log P(s_k) + \sum_{j=1}^{n} \log P(v_j \mid s_k) \right)$$

- Training a Naïve Bayes classifier
  - = estimating $P(v_j \mid s_k)$ and $P(s_k)$ from a sense-tagged training corpus
  - = finding the most likely sense k

$$P(v_j \mid s_k) = \frac{\text{count}(v_j, s_k)}{\sum_t \text{count}(v_t, s_k)}$$

Number of occurrences of feature j over the total number of features appearing in windows of S_k

$$P(s_k) = \frac{\text{count}(s_k)}{\text{count}(\text{word})}$$

Number of occurrences of sense k over number of all occurrences of ambiguous word

Example

- Training corpus (context window = ±3 words):

…Today the World **Bank/**BANK1 and partners are calling for greater relief…

…Welcome to the **Bank/**BANK1 of America the nation's leading financial institution…

…Welcome to America's Job **Bank/**BANK1 Visit our site and…

…Web site of the European Central **Bank/**BANK1 located in Frankfurt…

…The Asian Development **Bank/**BANK1 ADB a multilateral development finance…

…lounging against verdant **banks/**BANK2 carving out the…

…for swimming, had warned her off the **banks/**BANK2 of the Potomac. Nobody…

Example

- Training:

corpus中bank1前后所取的单词总数

  - P(*the* | BANK1) = 5/30 P(*the* | BANK2) = 3/12
  - P(*world* | BANK1) = 1/30 P(*world* | BANK2) = 0/12
  - P(*and* | BANK1) = 1/30 P(*and* | BANK2) = 0/12
  - …

corpus中bank2前后所取的单词总数

  - P(*off* | BANK1) = 0/30 P(*off* | BANK2) = 1/12
  - P(*Potomac* | BANK1) = 0/30 P(*Potomac* | BANK2) = 1/12

  - P(BANK1) = 5/7 P(BANK2) = 2/7

Example

此时也仅只关注window内的单词(+-3)

- Disambiguation: "*I lost my left shoe on the **banks** of the river Nile.*"
  - Score(BANK1)=log(5/7) + log(P(*shoe* | BANK1))+log(P(*on* | BANK1))+log(P(*the* | BANK1)) …
  - Score(BANK2)=log(2/7) + log(P(*shoe* | BANK2)+log(P(*on* | BANK2))+log(P(*the* | BANK2)) …

Example (with add 0.5 smoothing)

- Training corpus (context window = ±3 words):

…Today the World **Bank/**BANK1 and partners are calling for greater relief…

…Welcome to the **Bank/**BANK1 of America the nation's leading financial institution…

…Welcome to America's Job **Bank/**BANK1 Visit our site and…

…Web site of the European Central **Bank/**BANK1 located in Frankfurt…

…The Asian Development **Bank/**BANK1 ADB a multilateral development finance…

…lounging against verdant **banks/**BANK2 carving out the…

…for swimming, had warned her off the **banks/**BANK2 of the Potomac. Nobody…

Example (with add 0.5 smoothing)

- Assume V = 50
- Training:
  - P(*the* | BANK1) = (5+0.5) / (30+0.5V) P(*the* | BANK2) = (3+.05) / (12 +0.5V)
  - P(*world* | BANK1) = (1+0.5) / 55 P(*world* | BANK2) = (0+0.5) / 37
  - P(*and* | BANK1) = (1+0.5) / 55 P(*and* | BANK2) = (0+0.5) / 37
  - …
  - P(*off* | BANK1) = (0+0.5) / 55 P(*off* | BANK2) = (1+0.5) / 37
  - P(*Potomac* | BANK1) = (0+0.5) / 55
    P(*Potomac* | BANK2) = (1+0.5) / 37
  - P(BANK1) = 5/7 P(BANK2) = 2/7

V应该为bank1、2中不同单词的个数

Example (with add 0.5 smoothing)

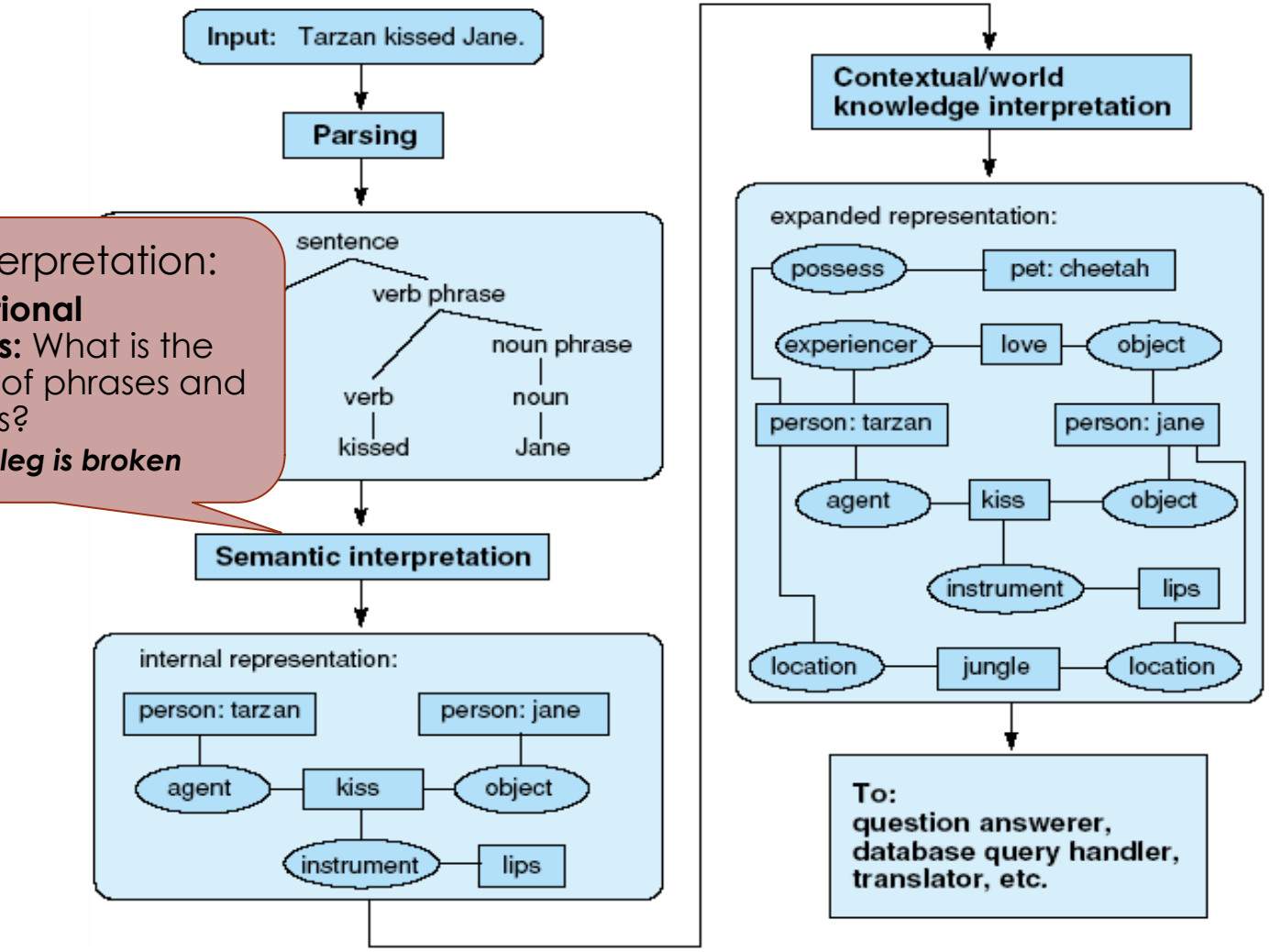
- Disambiguation: "*I lost my left shoe on the* ***banks*** *of the river Nile.*"
  - Score(BANK1)=log(5/7) +
    log(P(*shoe* | BANK1))+log(P(*on* | BANK1))+log(P(*the* | BANK1)) …
  - Score(BANK2)=log(2/7) +
    log(P(*shoe* | BANK2))+log(P(*on* | BANK2))+log(P(*the* | BANK2)) …

Stages of NL Understanding

Semantic interpretation:

- **Compositional Semantics:** What is the meaning of phrases and sentences?

  The chair's leg is broken



Compositional Semantics

- *The cat eats the mouse = The mouse is eaten by the cat.*
- Goal:
  - map an expression into a **knowledge representation**
    - a representation of context-independent, literal meaning
    - e.g. first-order predicate logic, conceptual graph, ...
  - to assign semantic roles (different from grammatical roles):
    - Semantic roles: Agent, Patient, Instrument, Time, Location, …
    - Grammatical roles: subject, direct object, ...
- E.g.
  - *The child hid the candy under the bed.*
  - **agent=child, patient=candy, location=under_the_bed, time=past**

Some Difficulties

- Syntax is not enough
  - *I ate spaghetti with a fork.* <instrument>
  - *I ate spaghetti with my sister.* <accompanying person>
  - *I ate spaghetti with meat balls.* *<attribute of food>*
  - *I ate spaghetti with lots of appetite.* *<manner>*

  - *Gun* = instrument that can kill
  - *Metal gun*… a gun made out of metal
  - *Water gun*… a gun made out of water?
  - *Fake gun*… it is a gun anyways? Can it kill?

  - *General Kane*… person but *General Motors* … corporation

Some Difficulties

- Parallel problems to syntactic ambiguity
  - *Happy [cats and dogs] live on the farm*
  - *[Happy cats] and dogs live on the farm*
- Quantifier Scoping
  - *Every man loves a woman.*
    - $\forall$ **m(**$\exists$ **f man(m)** $\wedge$ **woman(f)** $\wedge$ **loves(m, f))**
    - $\exists$ **f(**$\forall$ **m man(m)** $\wedge$ **woman(f)** $\wedge$ **loves(m, f))**

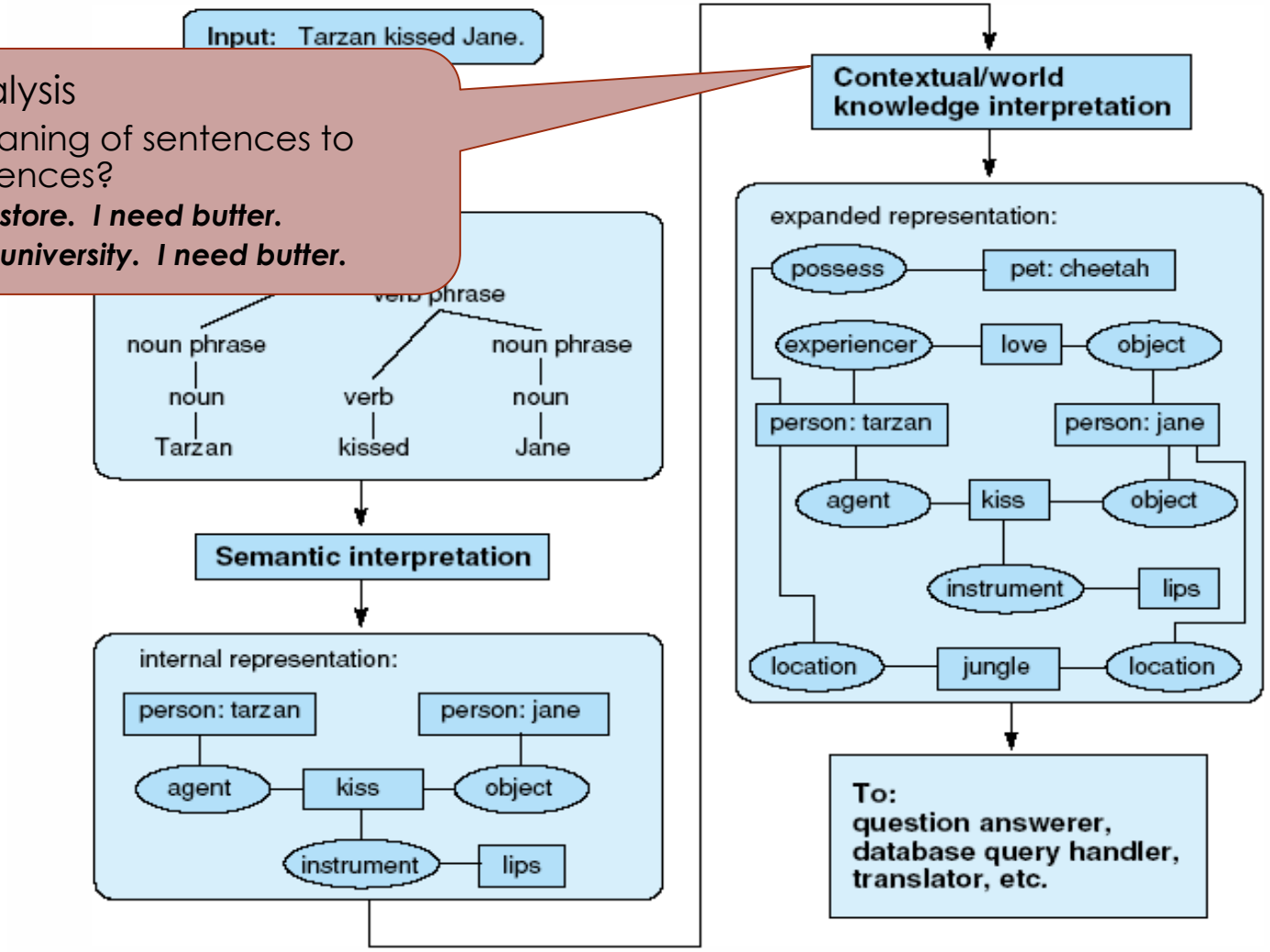
Stages of NL Understanding

- Discourse Analysis

How to relate the meaning of sentences to surrounding sentences?

I have to go to the store. I need butter.

I have to go to the university. I need butter.



Disclosure Analysis

- In logics: $A \wedge B \wedge C \Leftrightarrow C \wedge B \wedge A$
- Not in NL:
  - *John visited Paris. He bought Mary some expensive perfume. Then he flew home. He went to Walmart. He bought some underwear.*
  - *John visited Paris. Then he flew home. He went to Walmart. He bought Mary some expensive perfume. He bought some underwear.*
- Humans infer relations between sentences that may not be explicitly stated in order to make a text coherent.
  - *(?) I am going to Concordia. I need butter.*

Examples of Disclosure Relations

CONDITION	*If it rains, I will go out.*
SEQUENCE	*Do this, then do that.*
CONTRAST	*This is good, but this is better.*
CAUSE	*Because I was sick, I could not do my assignment.*
RESULT	*Click on the button, the red light will blink.*
PURPOSE	*To use the computer, get an access code.*
ELABORATION	*The solution was developed by Alan Turing. Turing was a great mathematician living in Great Britain. He was a computer scientist as well as a logician.*

Another Classification Problem

- Discourse tagging can be viewed as typical classification problem
  - use machine learning techniques (ex. Naïve Bayes classifier, decision tree) to train a system
  - that learns a classifier to assign to unseen sentences one of a fixed number of discourse relations (categories)
- Input:
  - Sentence Ex. If it rains, I will go out.
  - Features?
    - Connectives such as “if”, “however”, “in conclusion”
    - Tense of verb (future, past)
    - …
- Output:
  - Most likely relation in the sentence (none, condition, contrast, purpose, …)

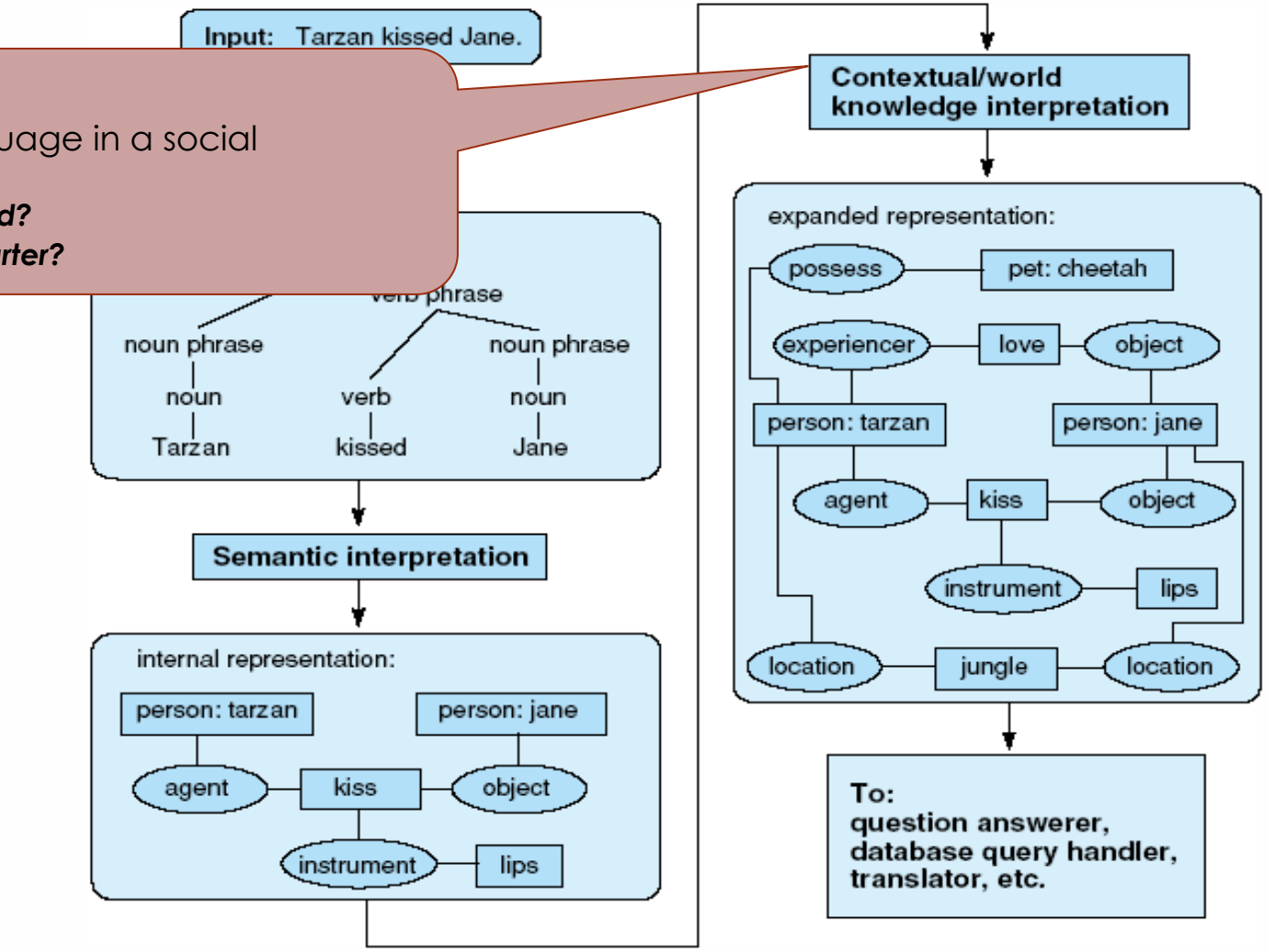
Stages of NL Understanding

- Pragmatics

How people use language in a social environment?

Do you have a child?

Do you have a quarter?



Pragmatics

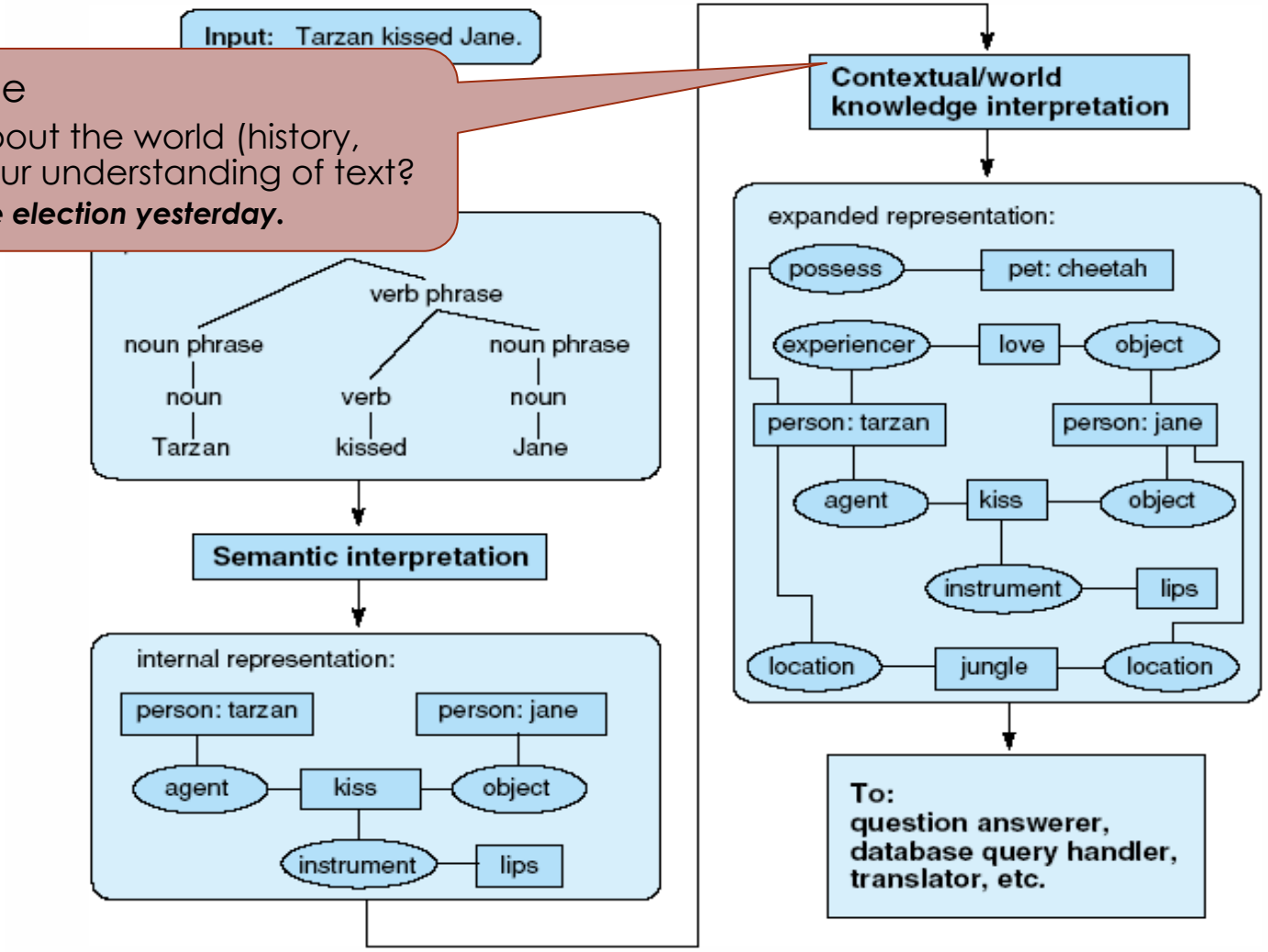
- Go beyond the literal meaning of a sentence
- Try to explain what the speaker is really expressing
- Understand how people use language socially
  - Eg: figures of speech, …
  - Eg: *Could you spare some change?*

Stages of NL Understanding

- World Knowledge

  How knowledge about the world (history, facts, …) modifies our understanding of text?

  Justin Trudeau won the election yesterday.



Using World Knowledge

- Using our general knowledge of the world to interpret a sentence/discourse
- Example:

  The trophy would not fit in the brown suitcase because …

  … it was too big.

  … it was too small.

  The professor sent the student to see the principal because…

  …he wanted to see him.

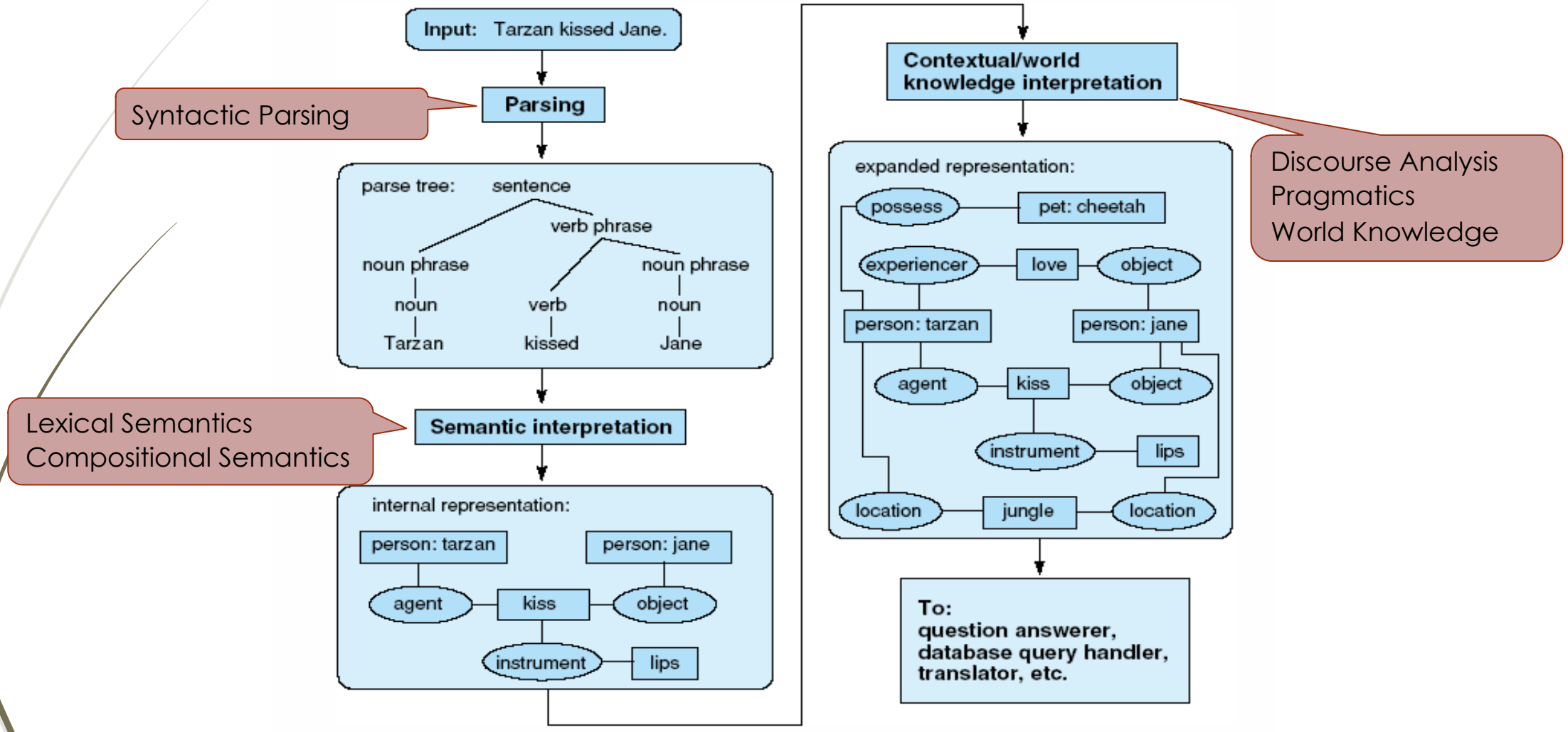
  …he was throwing paper balls in class.

  …he could not take it anymore.

  - *Ex: Silence of the lambs…*

Current Research area: see Winograd Schema Challenge

Summary of NL Understanding

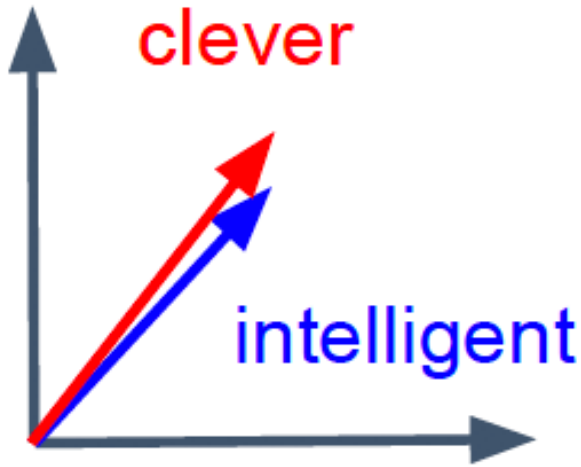


Topics

- ***Stages of NL Understanding***
- Contextualized Word Embeddings

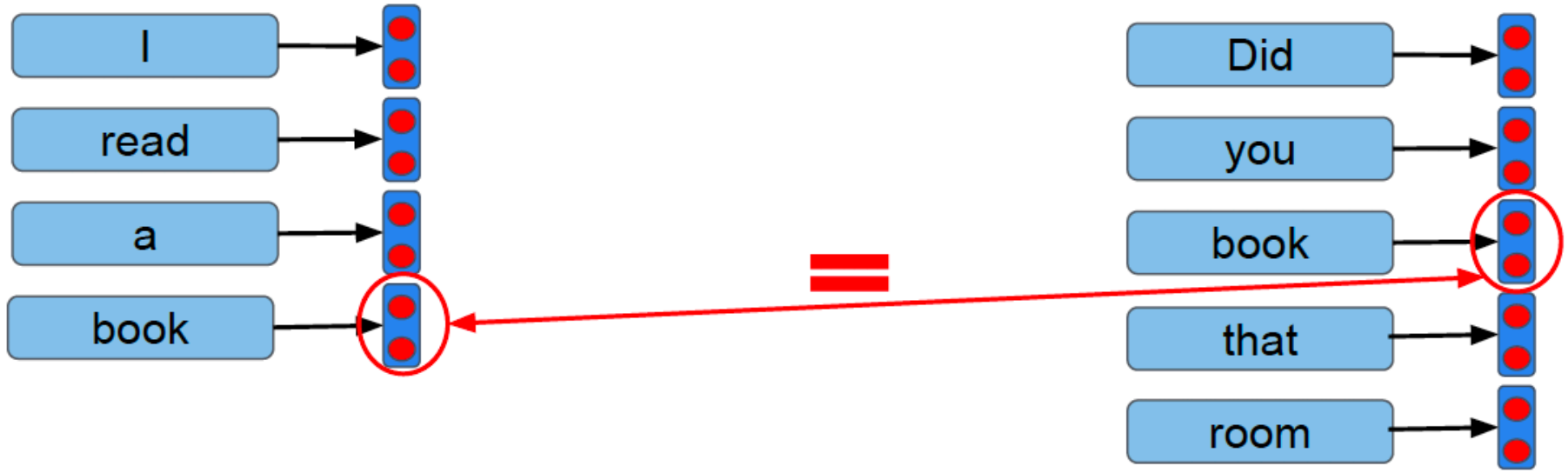
Word Embeddings and Synonymy

- Word embeddings are very useful representations that often lead to improved results.
- This is in part due to the fact that they can capture synonymy properties.
  - Synonyms in general have very close embedding vectors.



Word Embeddings and Polysemy

- Word embeddings do **not** help with polysemy though.
- A word such as "book" has the **same** embedding **regardless of the context**.



Word Embeddings and Polysemy

- Word embeddings reply to the question:

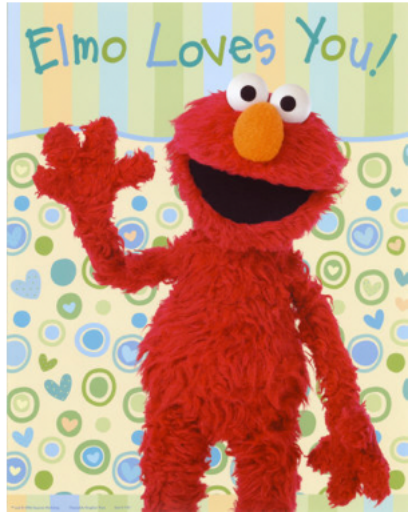
  what is the embedding for “book”?

- In order to consider the context, the question should become:

  what is the embedding for “book” in the sentence “I read a book”?

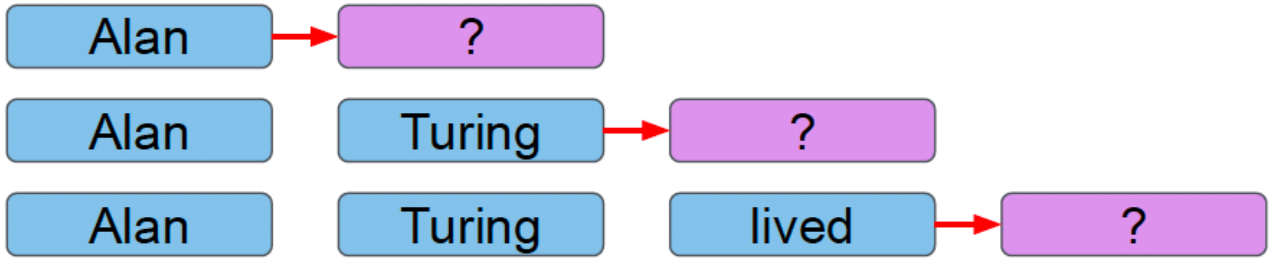
ELMo

- ELMo (Embeddings from Language Models) proposes a solution to two problems:
  - Generate contextualized word embeddings...
  - … in an unsupervised pre-training phase.
- To pre-train, they select the Language Modeling (LM) task.

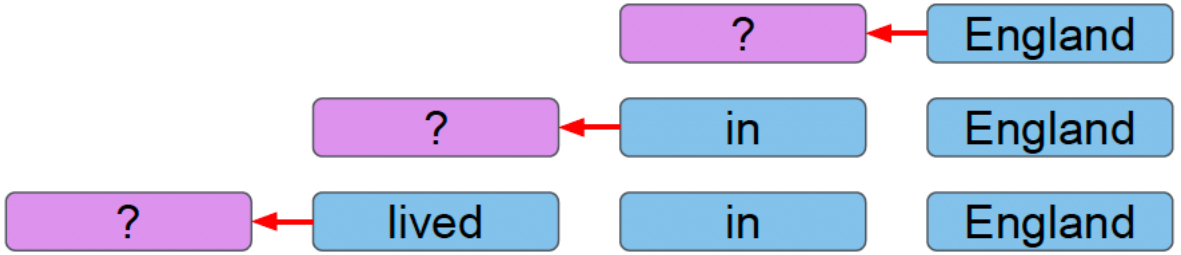


Language Modeling

- Given n words (from a sentence), predict the next word (n+1).



- Language Modeling is an unsupervised task.
- Note it can also work 'right – to – left'.

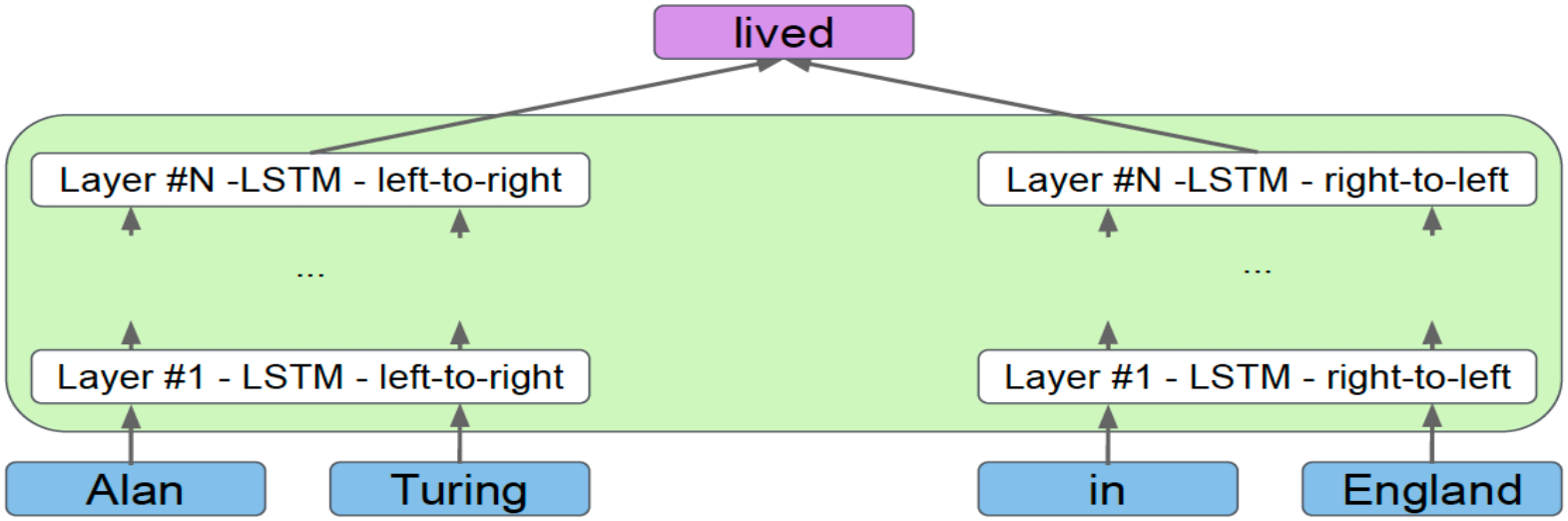


ELMo

- ELMo model is a N-layer bidirectional LSTM.

(Note: **LSTM** (Long Short Term Memory) Networks are called fancy recurrent neural networks with some additional features.)

- The contextualized embedding at a time step corresponds to a combination of the hidden states of all the various layers.



Peters et al. "Deep contextualized word representations."

ELMo – Pre-Training

- In pre-training, the goal is to maximize the following function:

$$\sum_{k=1}^{N} (\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s))$$

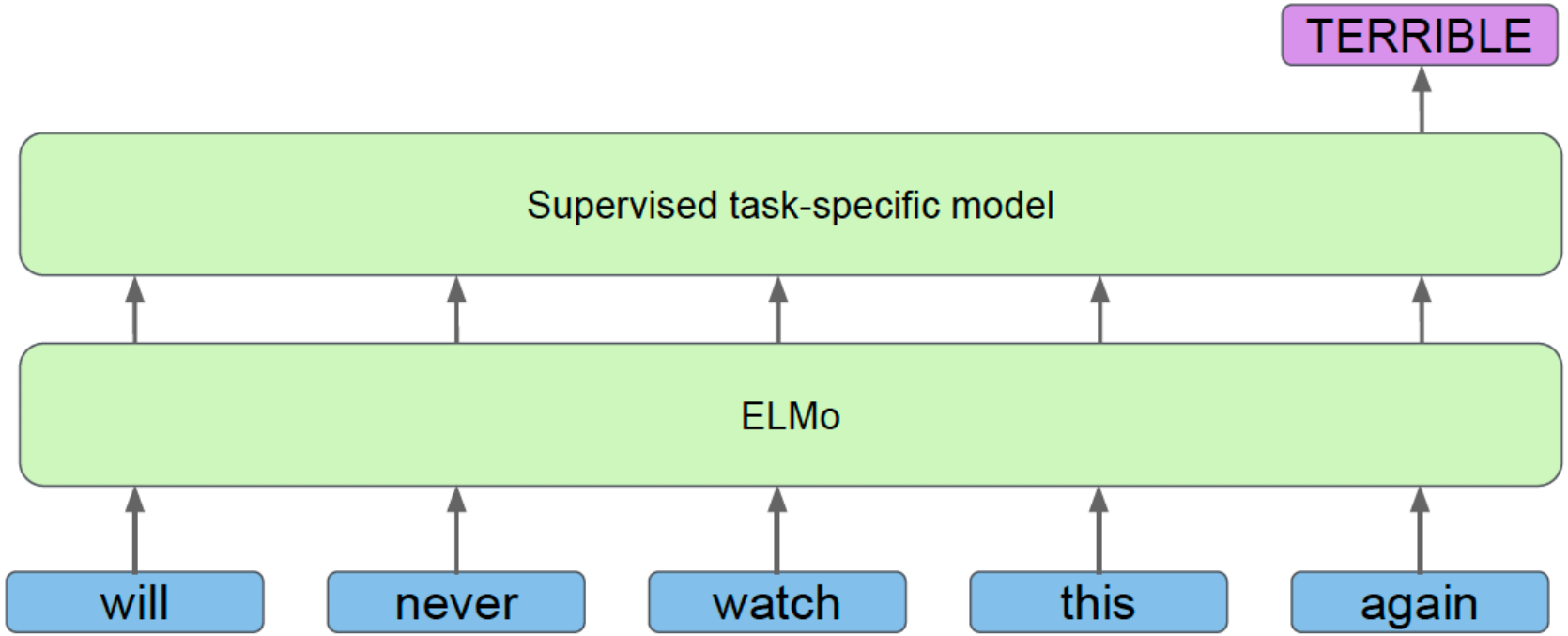
where Θ_x and Θ_s are respectively the token representation and softmax layer parameters. Those parameters are shared by both LMs.

- Note that the hidden states of all layers are used to generated the contextualized embedding.

Peters et al. "Deep contextualized word representations."

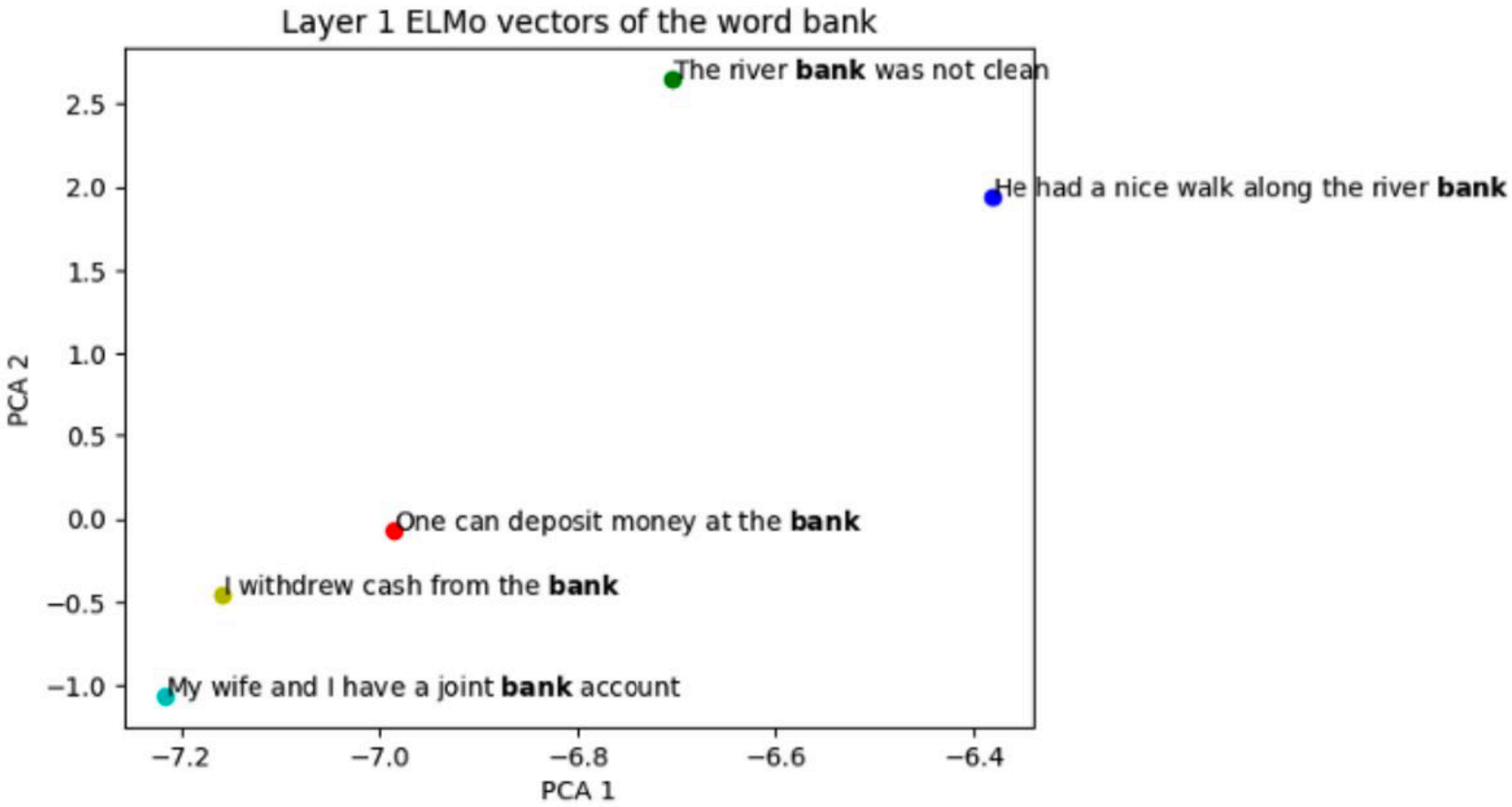
ELMo

- After pre-training, ELMo can be used with any other model (trained on supervised task).



Peters et al. "Deep contextualized word representations."

ELMo - Visualization



Recap

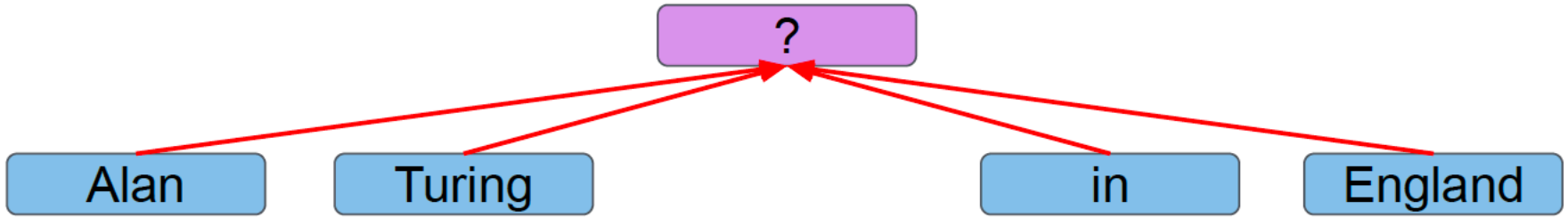
- Word embeddings provide efficient distributed representations for words.
  - They are based on an unsupervised pre-training phase that helps with scarce-data (supervised) tasks.
- ELMo and other similar approaches generate contextualized word embeddings that provide better results than uncontextualized embeddings.
  - Some of those approaches are based on an unsupervised pre-training phase. Others are based on a supervised pre-training phase.
- Do we have the best possible representation for words now?

BERT

- Not yet!
- BERT (Bidirectional Encoder Representations from Transformers) improved results over many NLP tasks by:
  - using a Transformer-based architecture;
  - pre-training on two unsupervised tasks:
    - Masked Language Modeling (MLM) instead of Language Modeling;
    - Next sentence prediction.

Masked Language Model

- Task: given a sentence where some tokens have been randomly masked, reconstruct the masked tokens.



- Note there is no left-to-right or right-to-left order:
  - When predicting the missing word, the model has access to both the past and the future.

Next Sentence Prediction

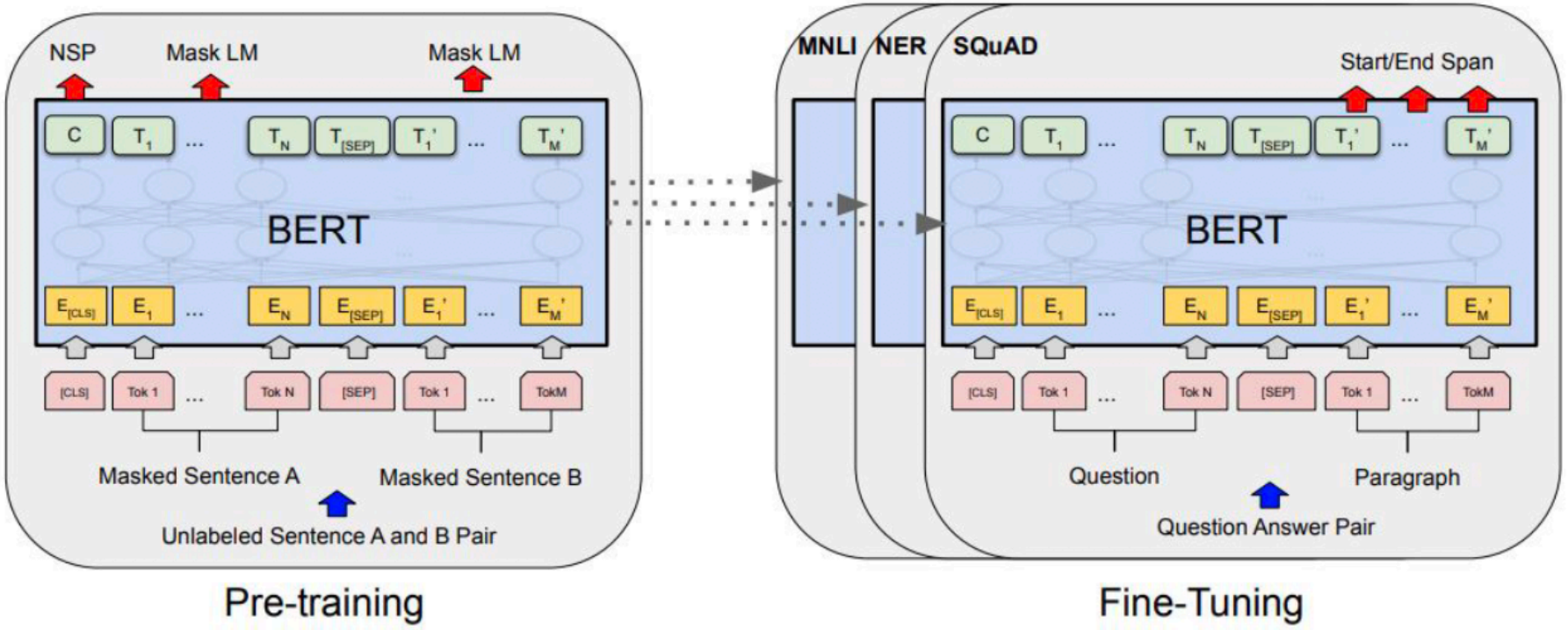
- Task: given two sentences, predict if they are contiguous.

- Each contiguous example is created by extracting 2 successive sentences from a corpus.
- Each non contiguous example is created by selecting 2 random sentences from a corpus.
- This task helps to learn to capture relationships between sentences.
  - E.g., useful for Question Answering.

Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding."

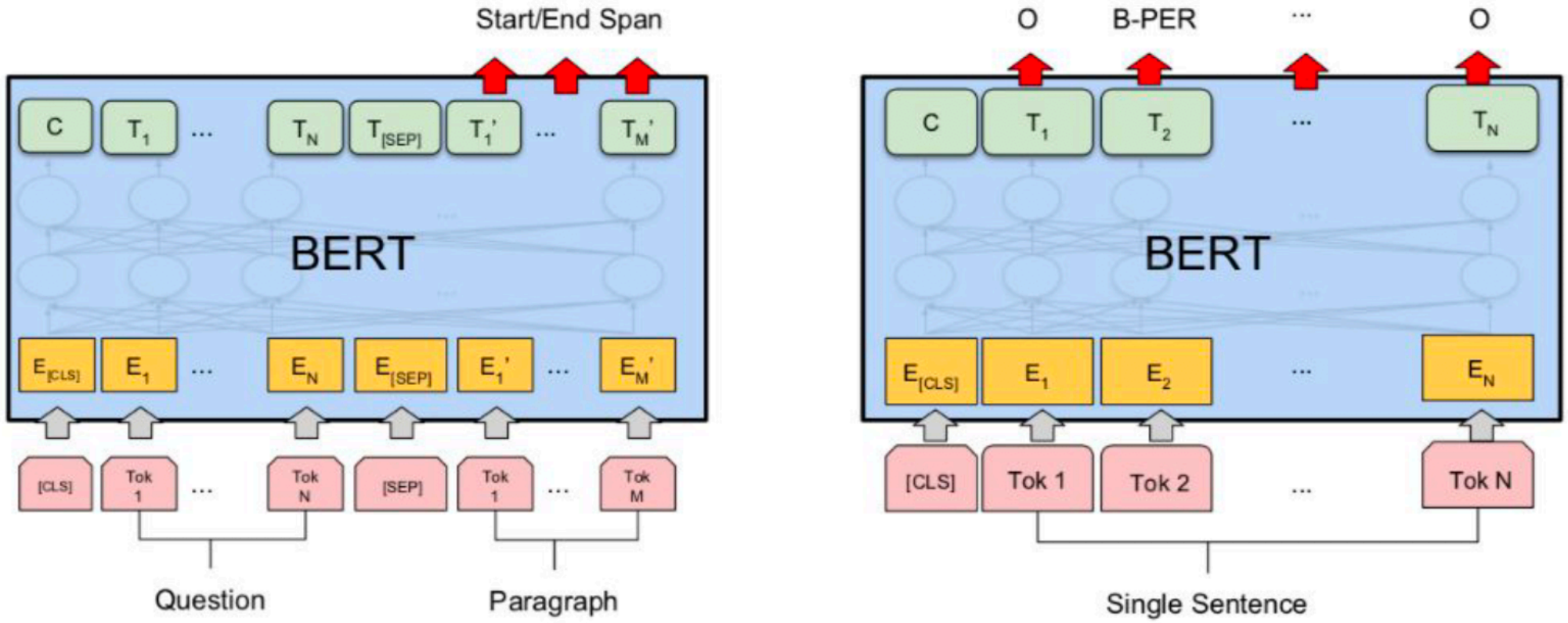
BERT – Training Phase

- After pre-training, the BERT model can be trained on the task of interest.
- Different kinds of tasks will require different input/output formalization.
  - Similar to what we saw in the NLP task section.



BERT – Training Phase

- E.g., Question Answering (left) and Named Entity Recognition (right):

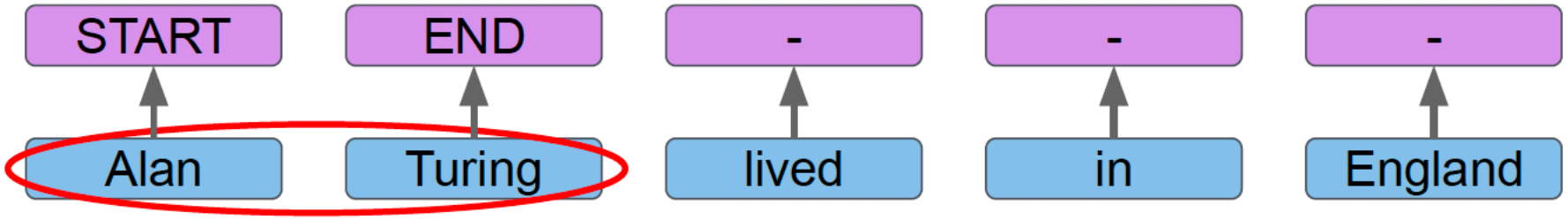


NLP – Extractive Question Answering

- Task: given a question, find the answer in some given text. E.g.,
  - (Input) Question: "who was living in England?"
  - (Input) Context: "Alan Turing lived in England"
  - Target: "Alan Turing" (i.e., first and second words in the context)

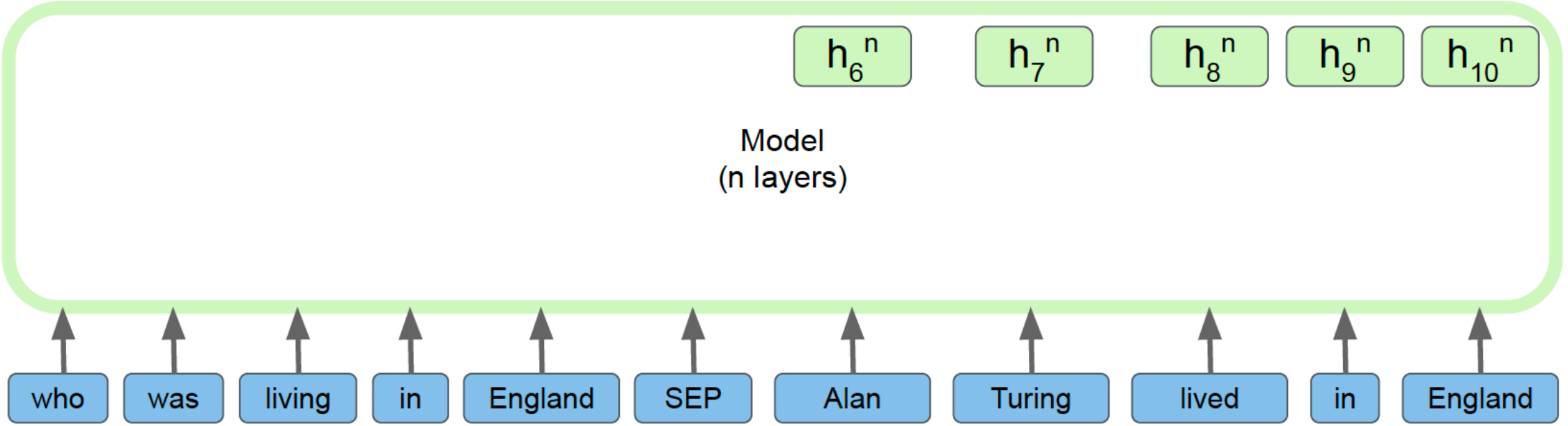
NLP – Extractive Question Answering

- Task: given a question, find the answer in some given text. E.g.,
  - (Input) Question: "who was living in England?"
  - (Input) Context: "Alan Turing lived in England"
  - Target: "Alan Turing" (i.e., first and second words in the context)
- This task can be modeled as a word-level classification task.
  - Each word can be the start of the answer, the end of the answer, or neither of them.

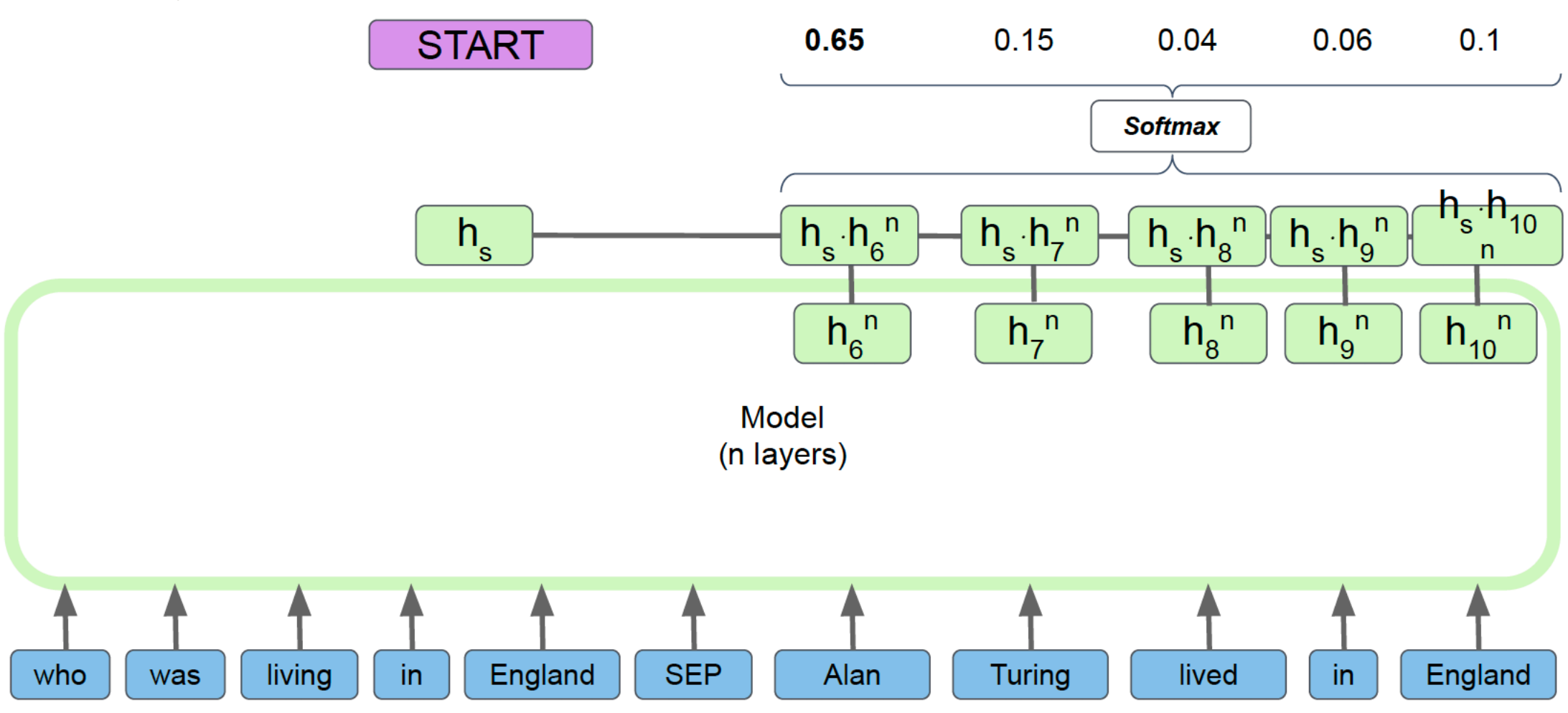


NLP – Extractive Question Answering

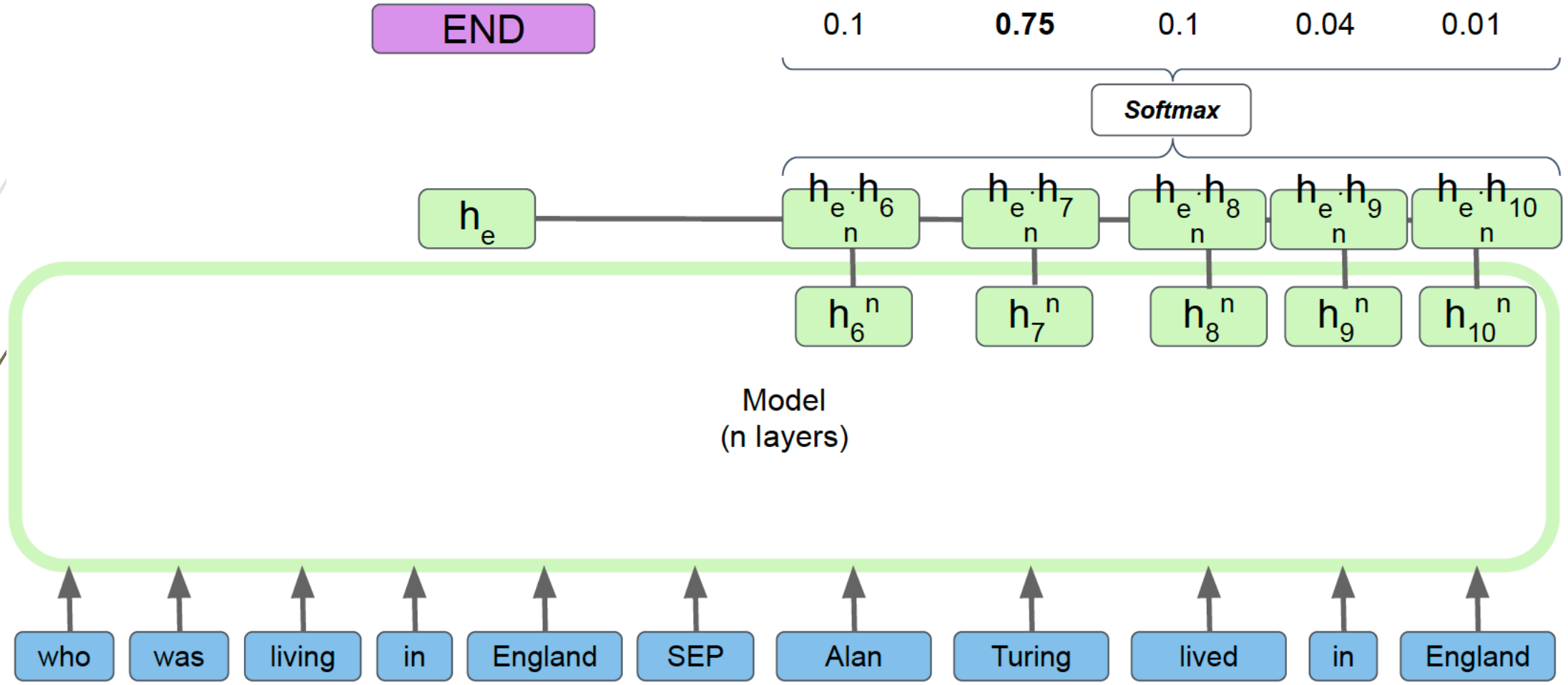
SEP = separator indicating the end of the first sentence.



NLP – Extractive Question Answering



NLP – Extractive Question Answering



Summary

- Natural Language Processing includes many types of tasks.
- These tasks share some common problems such as the need to link words to semantics.
- Several algorithms have been developed to address these problems, mainly word embeddings (such as Word2Vec / FastText) and contextualized word embeddings (such as ELMo / BERT).

Summary

- BERT is able to address many NLP tasks with a common core architecture (based on the Transformer).
- All these algorithms (Word2Vec / FastText / ELMo / BERT) use the idea of (unsupervised) pre-training to help address scarcity of data for supervised tasks.
- New algorithms are created regularly!
  - E.g., XLNet, ERNIE, RoBERTa...

Yang et al. "XLNet: Generalized Autoregressive Pretraining for Language Understanding."
Zhang et al. “ERNIE: Enhanced Language Representation with Informative Entities”
Liu et al. “RoBERTa: A Robustly Optimized BERT Pretraining Approach”

The End