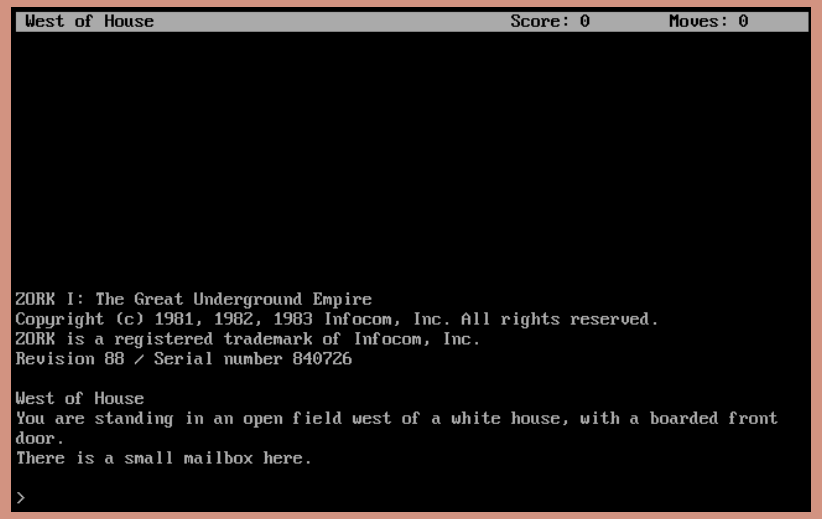
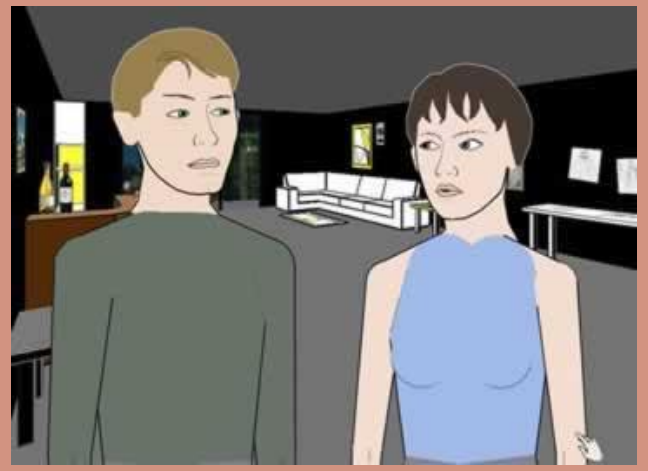
The Story So Far... (Recap)

CIS-700 Interactive Fiction and Text Generation
Module 4 - 3/15/2022
Dr. Lara J. Martin

Interactive Fiction & Storytelling



Zork I



Façade, https://www.playablstudios.com/facade
https://cdn.download-free-games.com/cf/images/nfe/screens/facade_2_m.jpg

Sentient Beings
https://grizel.itch.io/sentient-beings

What makes a good story?

What makes a story "good"?

Coherent

- coherence
- clear logic
- coherent plot lines
- consistency/continuity
- Fun (diverse) but logical.

Interesting

- surprises
- interesting, have a surprising ending
- compelling conflict
- engaging narrative
- convoluted
- coherent, has an element of surprise, complex characters, beautiful worldbuilding

Relatable Characters

- Compelling/relatable characters
- character growth
- Relatability
- Compelling plot, interesting and relatable characters, humor, unexpected but properly explained plot points
- compelling action and characters
- decent storyline, compelling characters and good writing

Something innate in us?

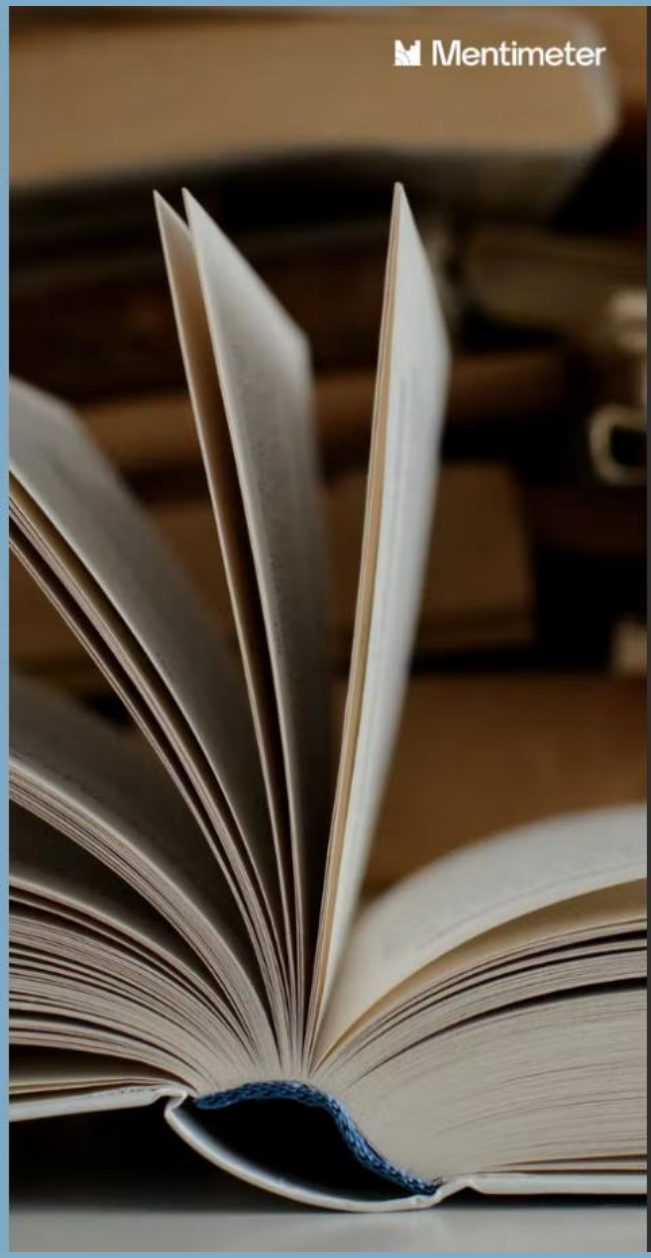
- I know it when I see it

Not everything written explicitly

- A good story make me want to come back and leaves room for the reader to think and come to their own conclusions

Complexity/Theme

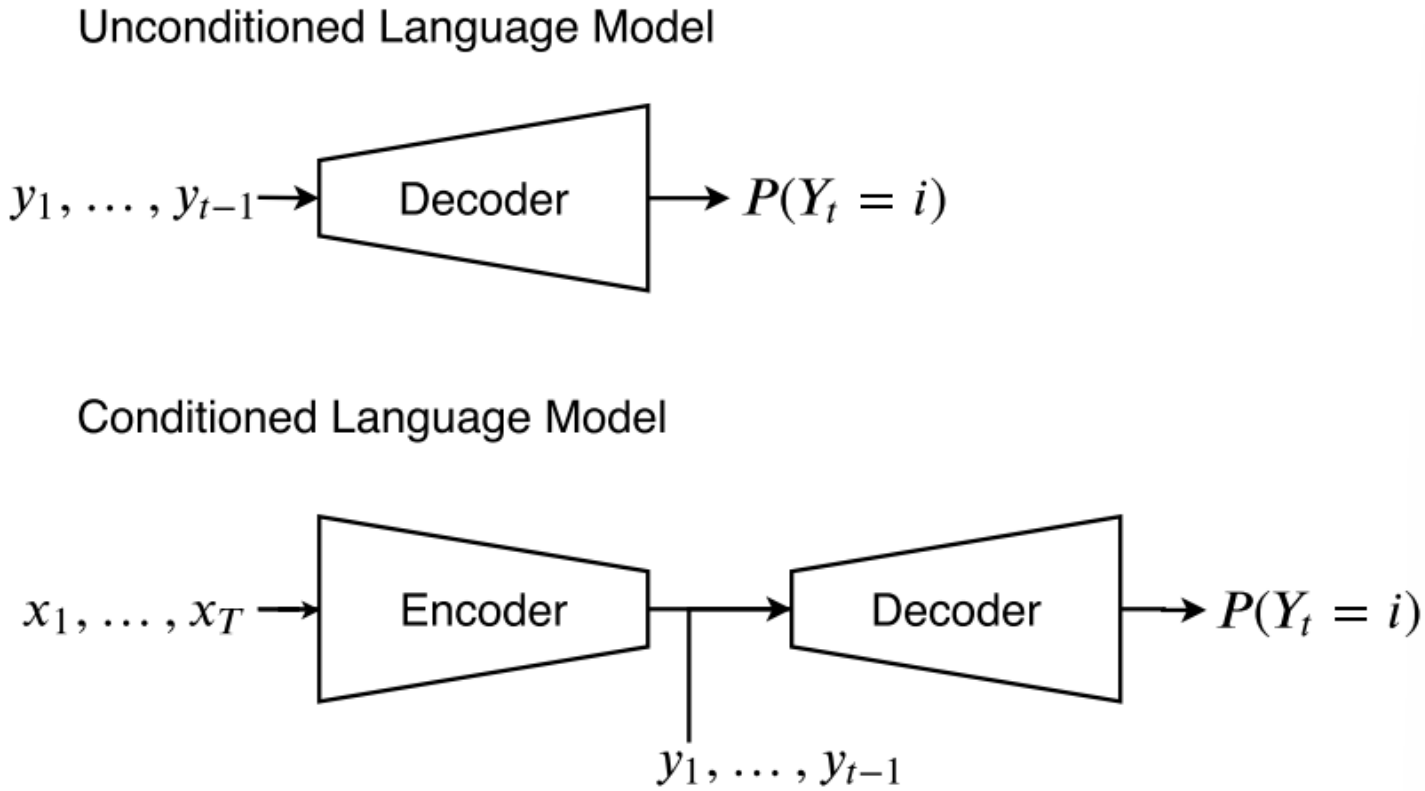
- Underlying ideas/themes
- Multiple plot elements
- underlying deep / philosophical themes
- Satisfying to read, gives interesting insights



Neural Systems

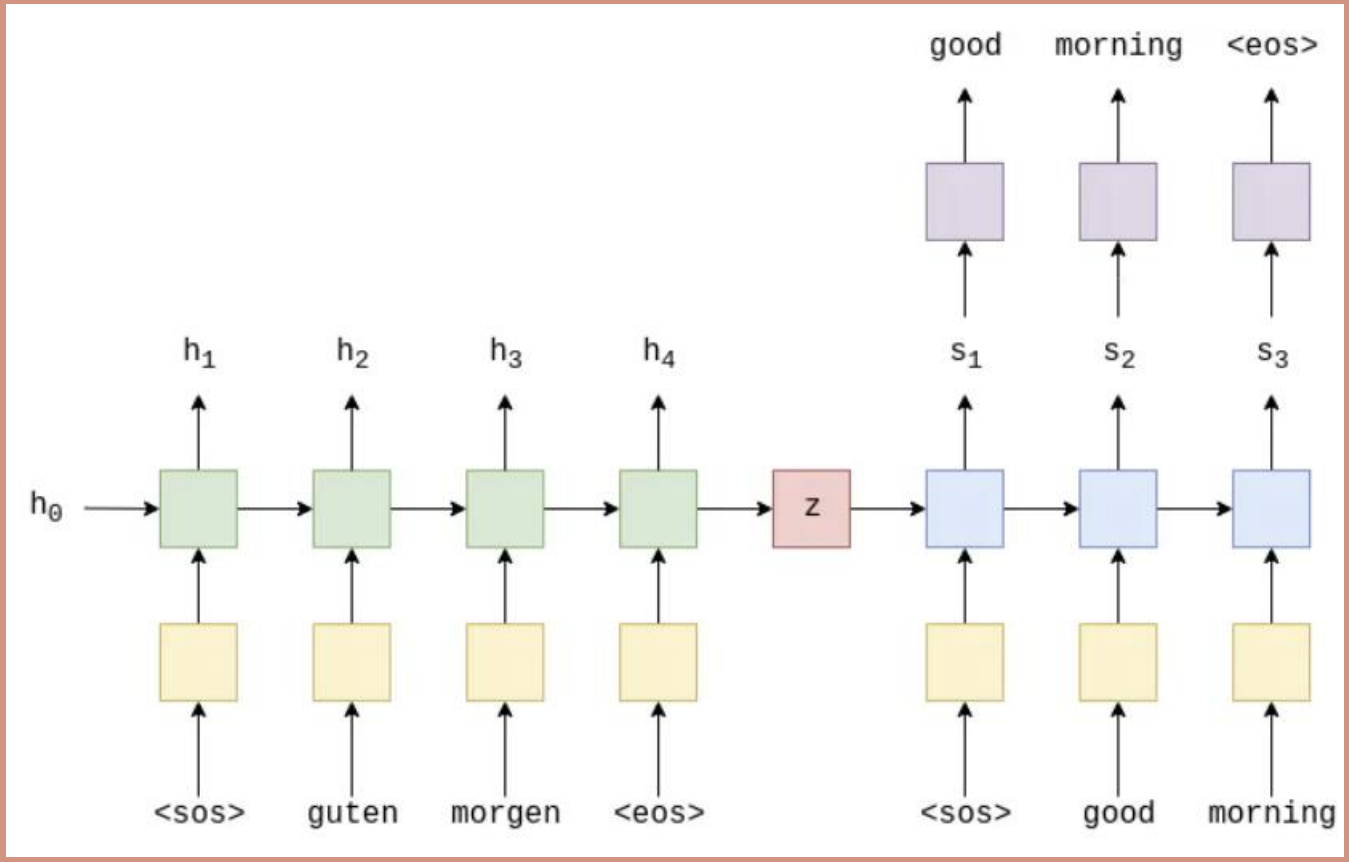
Neural Generation

- Probabilistic
  - Unconditioned
    $P(Y)$
  - Conditioned
    $P(Y \mid X)$



From Daphne Ippolito's lecture

RNNs (Sequence-to-Sequence)



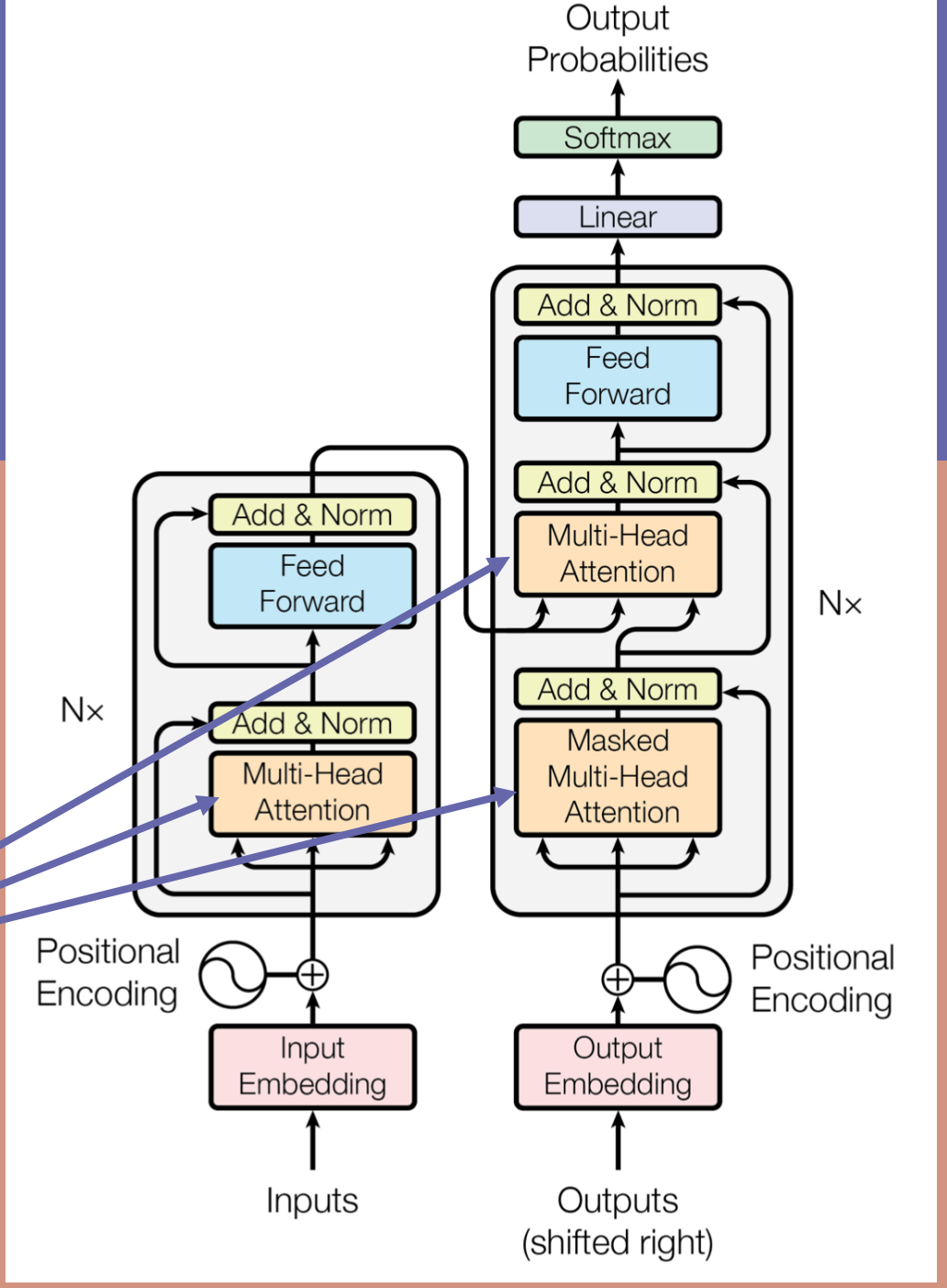
https://www.jianshu.com/p/5b06a91d4f3d

Transformers

Query Q – what you're "searching" for
Key K – what you compare the query against
Value V – the results that is paired to the key

Attention is All You Need!

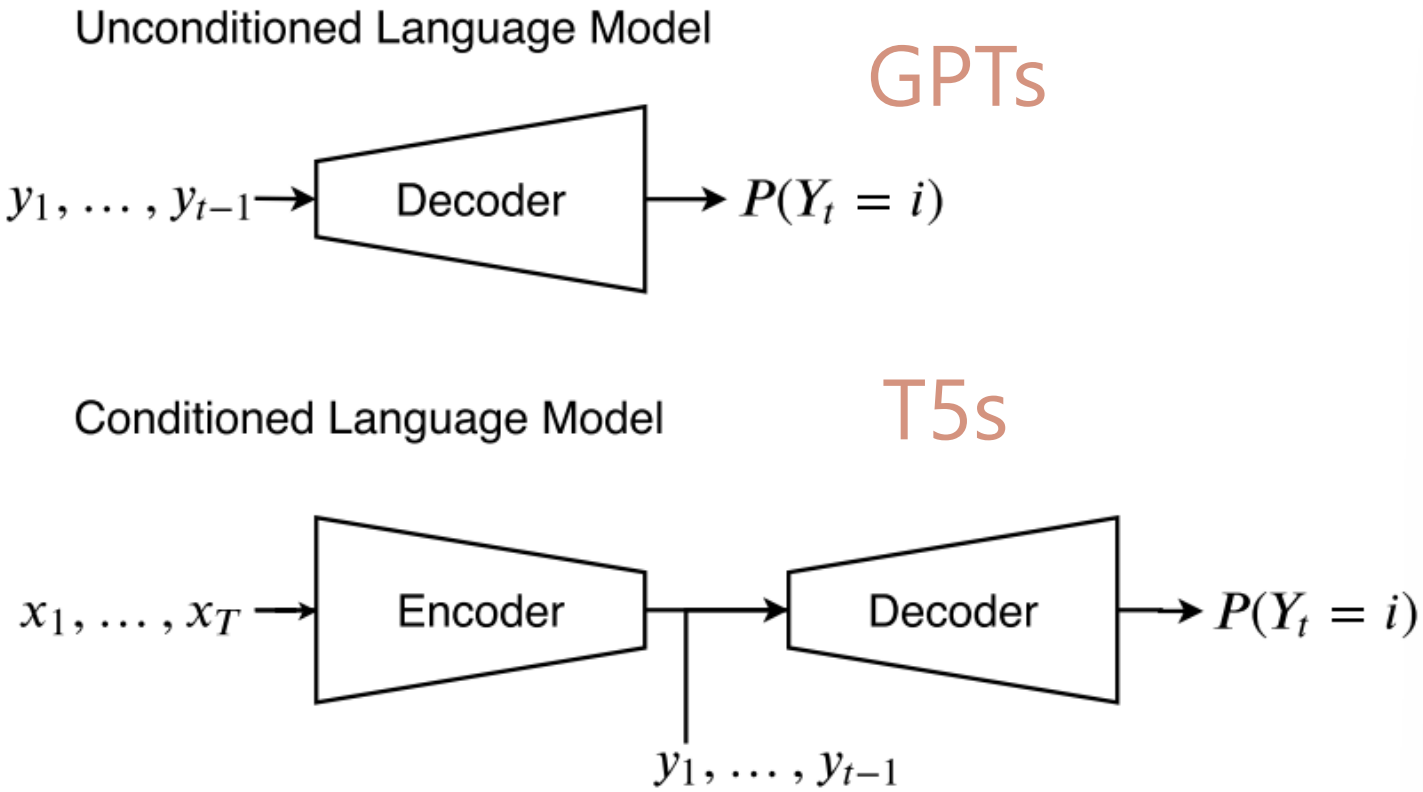
$$softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)v$$



From "Attention is All You Need"

Transformer Types

Encoder-Only:
BERTs



From Daphne Ippolito's lecture

Neural Story Generation

The hungry dog licked her lips as she watched her owner eat.
"You've been a good girl," he told her. "I think you deserve a reward."
Once she was done, she jumped back on the couch and waited patiently.
Her owner took a piece of steak out of the fridge and gave it to her.
"Thank you," he said. "I'm glad you're my dog."
She wagged her tail and ate the steak.
"If you're good, you can have a treat later," he said. "But for now, you have to sleep. I have a long day tomorrow."
She nodded and lay down on the floor.
Her owner got up, turned off the lights, and lay down on the bed.

On Theme ✓

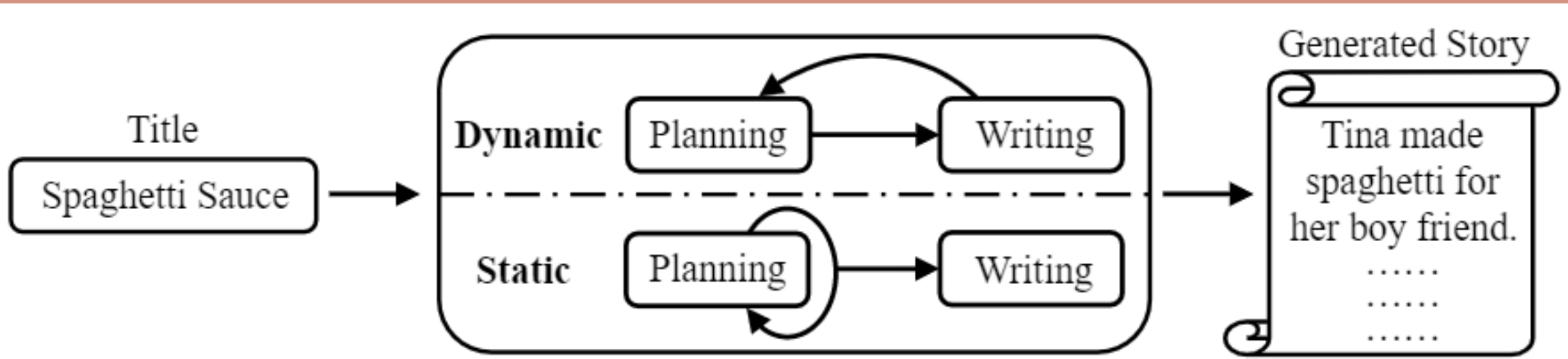
Grammar ✓

Remembering Story State ✗

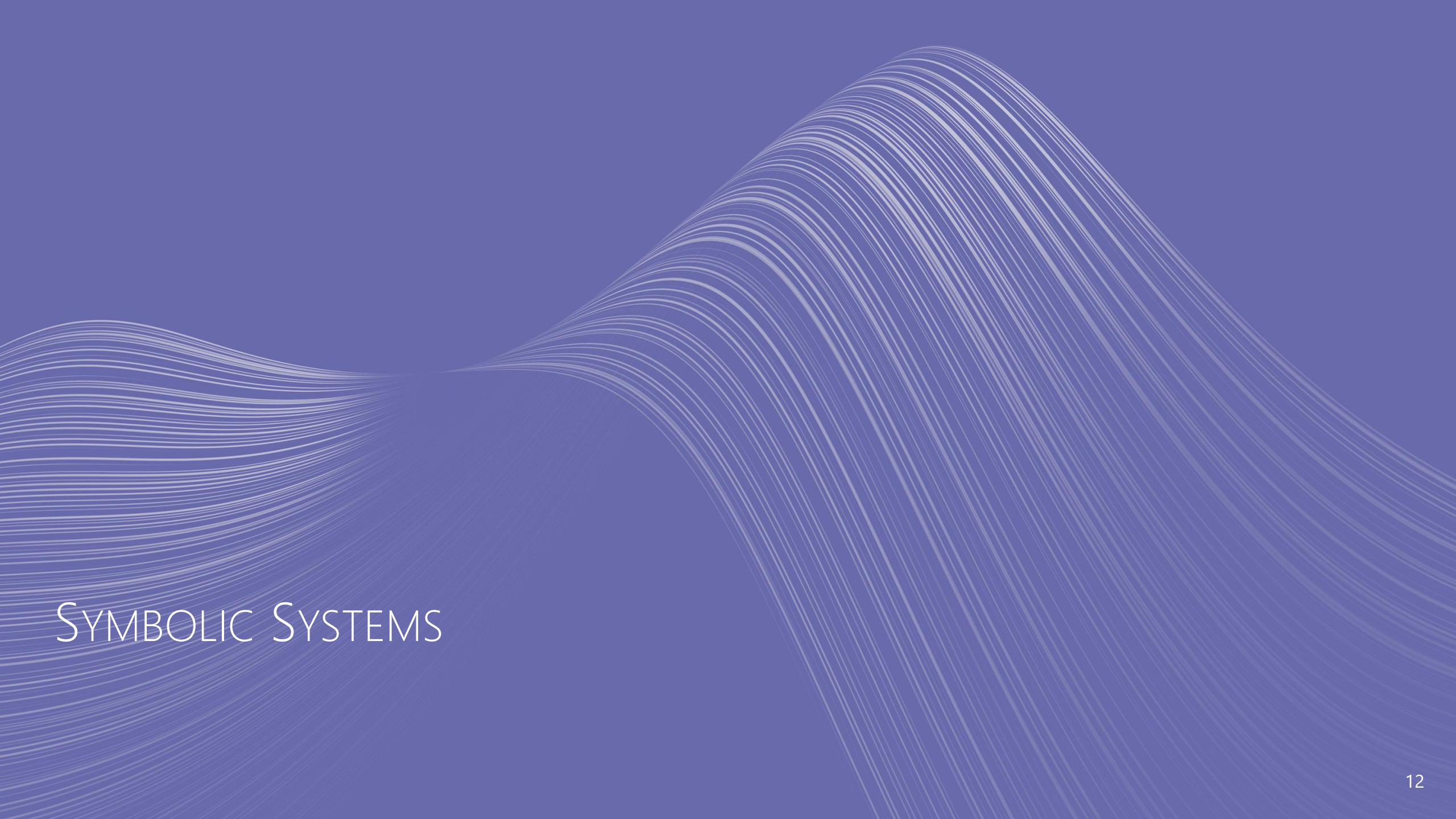
Commonsense Reasoning ✗

Guided Neural Story Generation

Integrating ways of including structure



Yao, L., Peng, N., Weischedel, R., Knight, K., Zhao, D., & Yan, R. (2019). Plan-And-Write: Towards Better Automatic Storytelling. *AAAI Conference on Artificial Intelligence (AAAI)*, *33*(1), 7378–7385. https://aaai.org/ojs/index.php/AAAI/article/view/4726

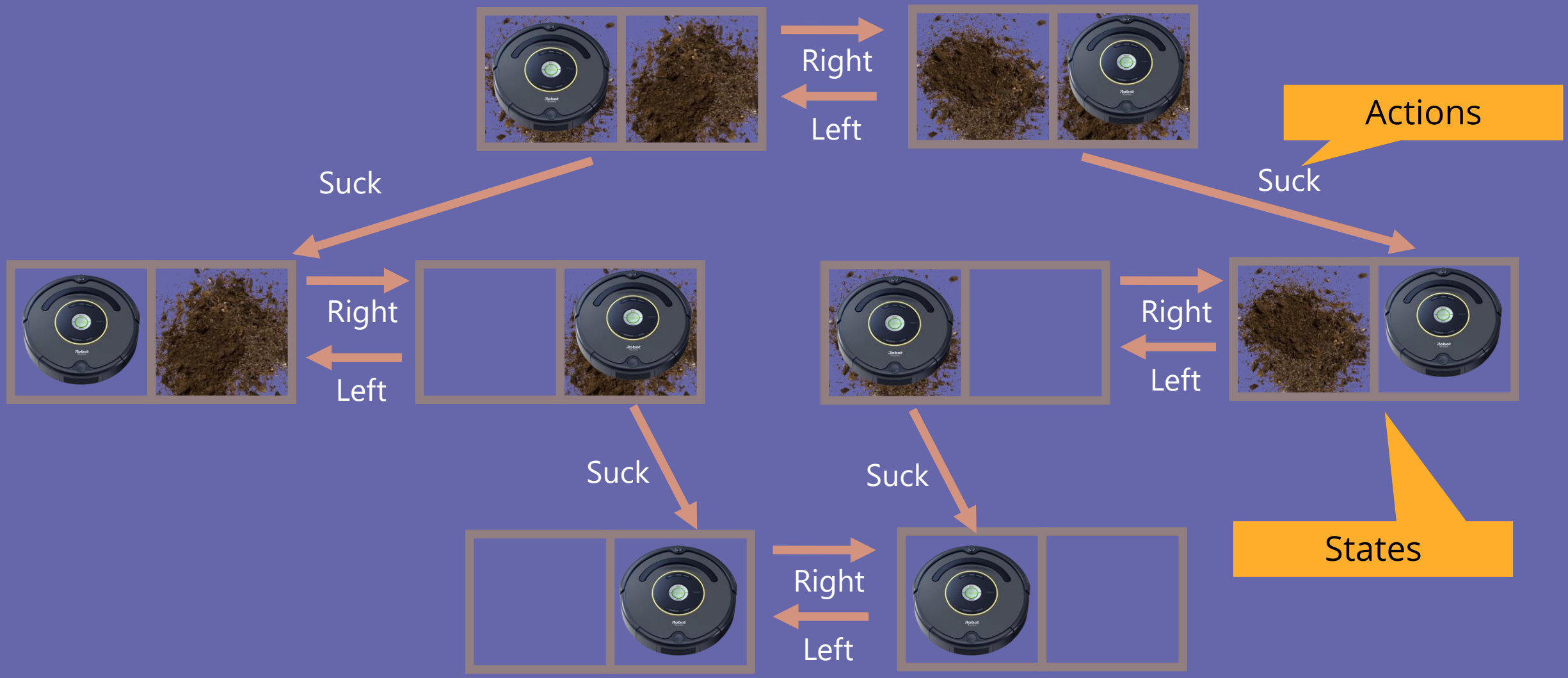


Symbolic Systems

Symbolic Systems: Planning

- Planning = *search* for a *plan*
- In story generation, this means we're looking for a plan where the goal is reached
- What's the goal? Depends on the story you're telling
  - E.g. Ending a conflict between characters, Robber steals from player character

Search


Right
Left
Actions
Suck
Suck
Right
Left
Right
Left
Suck
Suck
Right
Left
States

What are we planning over?

Structure (Schemas)

- KB Schemas
- Scripts
- Procedures

} Organization of Commonsense Knowledge

States

VerbNet Schema

Jen sent the book to Remy from Atlanta.

Atlanta : location
book : concrete
Jen : animate or organization
!has_location(book, Atlanta)
has_location(book, Remy)

COMET-ATOMIC Schema

HW 5: Schemas

In this homework, you will create your own schema to represent the state of a story world as it goes through the story line by line. A **schema** is a structured reprensentation made to hold facts or a plan, which in this case, can be used to track change over time.

The purpose of this homework is to test your understanding of schemas and get hands-on experience with a state-of-the-art tool in commonsense reasoning.

Your Task

You will be creating a schema using ATOMIC to track the state of a fictional world. For each sentence of the story, you will parse it (provided), call COMET (provided, but what you input is up to you), create preconditions to determine if a sentence can be added (TODO), and create effects to use to update your schema (TODO).

Let's teach your agent some basic information about the world!

Formally, the task is:

Given an input sentence at time *t* (*In_t*), produce a schema *S_t*. Do this for each sentence in the story.

For example, using VerbNet:

What are we planning over?

Structure (Schemas)

KB Schemas

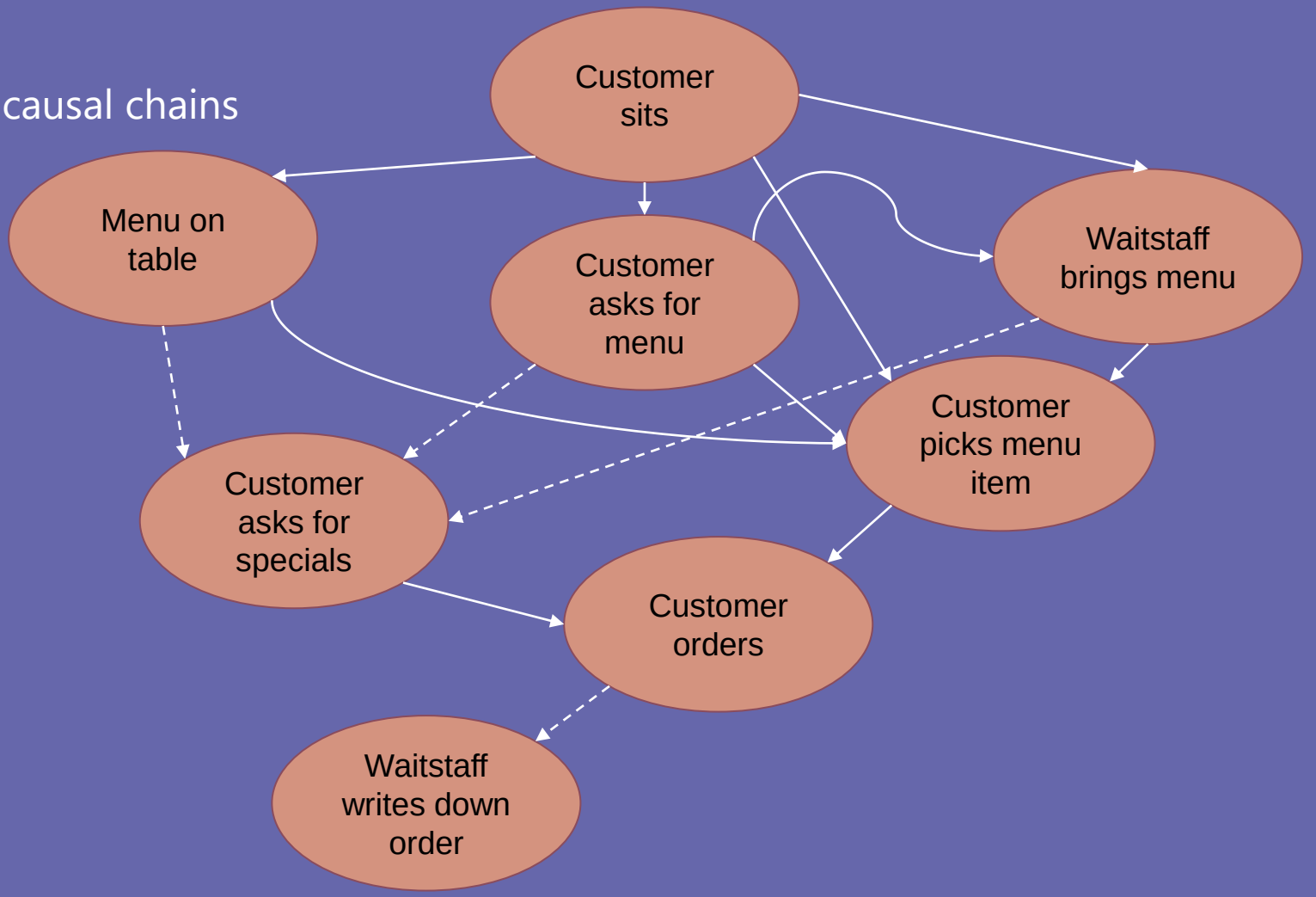
Scripts

Procedures

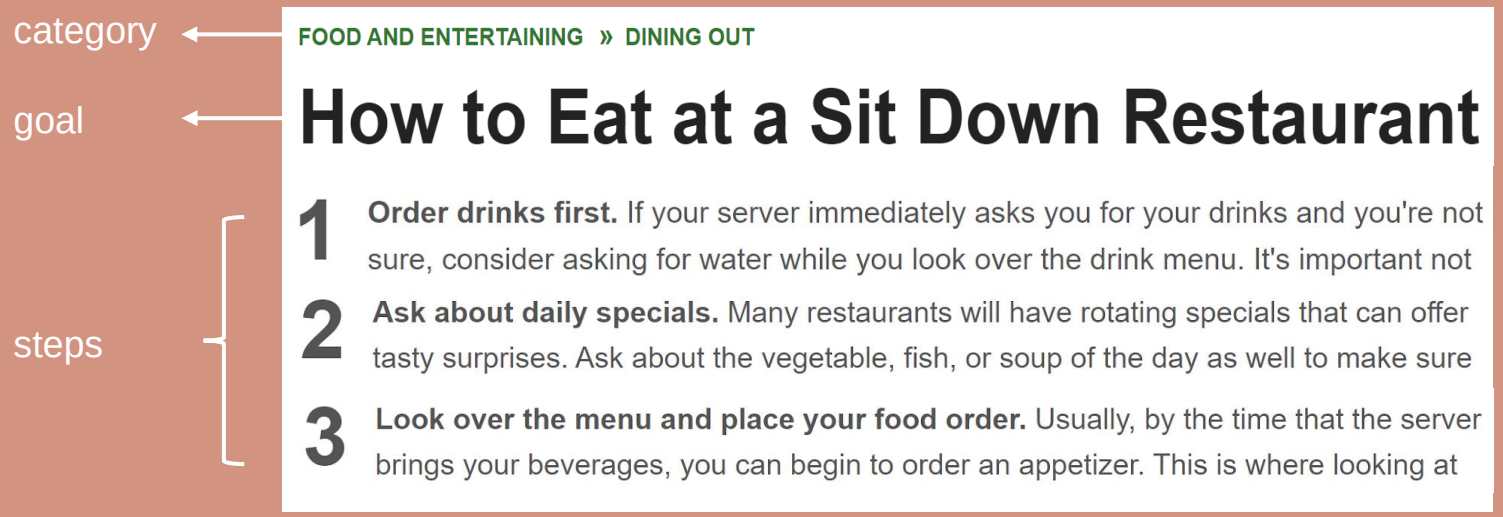
} Organization of Commonsense Knowledge

Scripts

Common sequence of causal chains

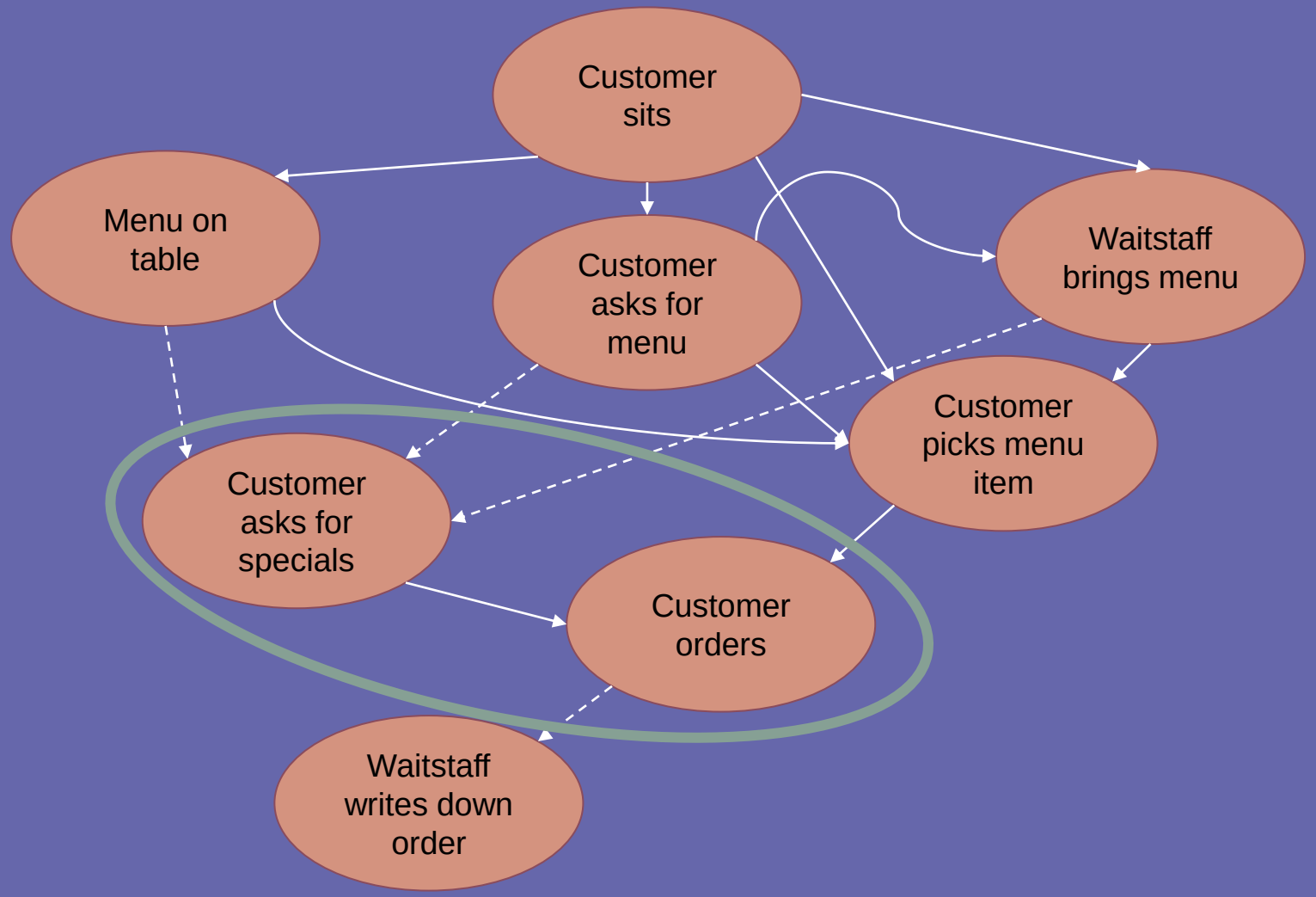


Procedures: Script with a goal

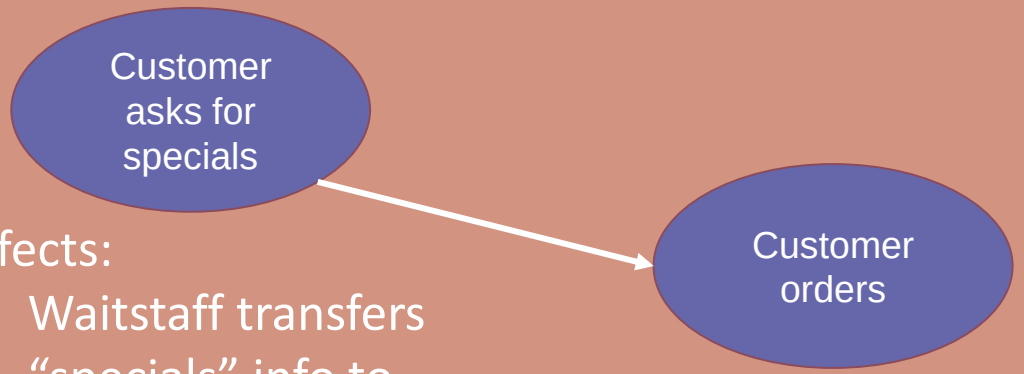


From Veronica Qing Lyu & Harry Li Zhang's lecture

Scripts


Customer sits
Menu on table
Customer asks for menu
Waitstaff brings menu
Customer picks menu item
Customer asks for specials
Customer orders
Waitstaff writes down order

Causal Links



Effects:

- Waitstaff transfers “specials” info to customer
- Customer knows what they want to order

Precondition:

- Customer knows what they want to order

Causal Links→ Actions for Planning

$$
\begin{aligned}
\boldsymbol{a}\text{:}\ & \boldsymbol{buy(Tom, Potion, Merchant, Market)} \\
\text{PRE}(a)\text{:}\ & at(Tom) = Market \wedge at(Merchant) = Market \wedge \\
& at(Potion) = Merchant \wedge wealth(Tom) \geq 1 \\
\text{EFF}(a)\text{:}\ & at(Potion) = Tom \wedge wealth(Merchant) \mathrel{+}= 1 \wedge \\
& wealth(Tom) \mathrel{-}= 1
\end{aligned}
$$

From Stephen Ware's lecture

Representation Language

Planning Domain Definition Language (PDDL) express **actions** as a **schema**

Action name

Variables

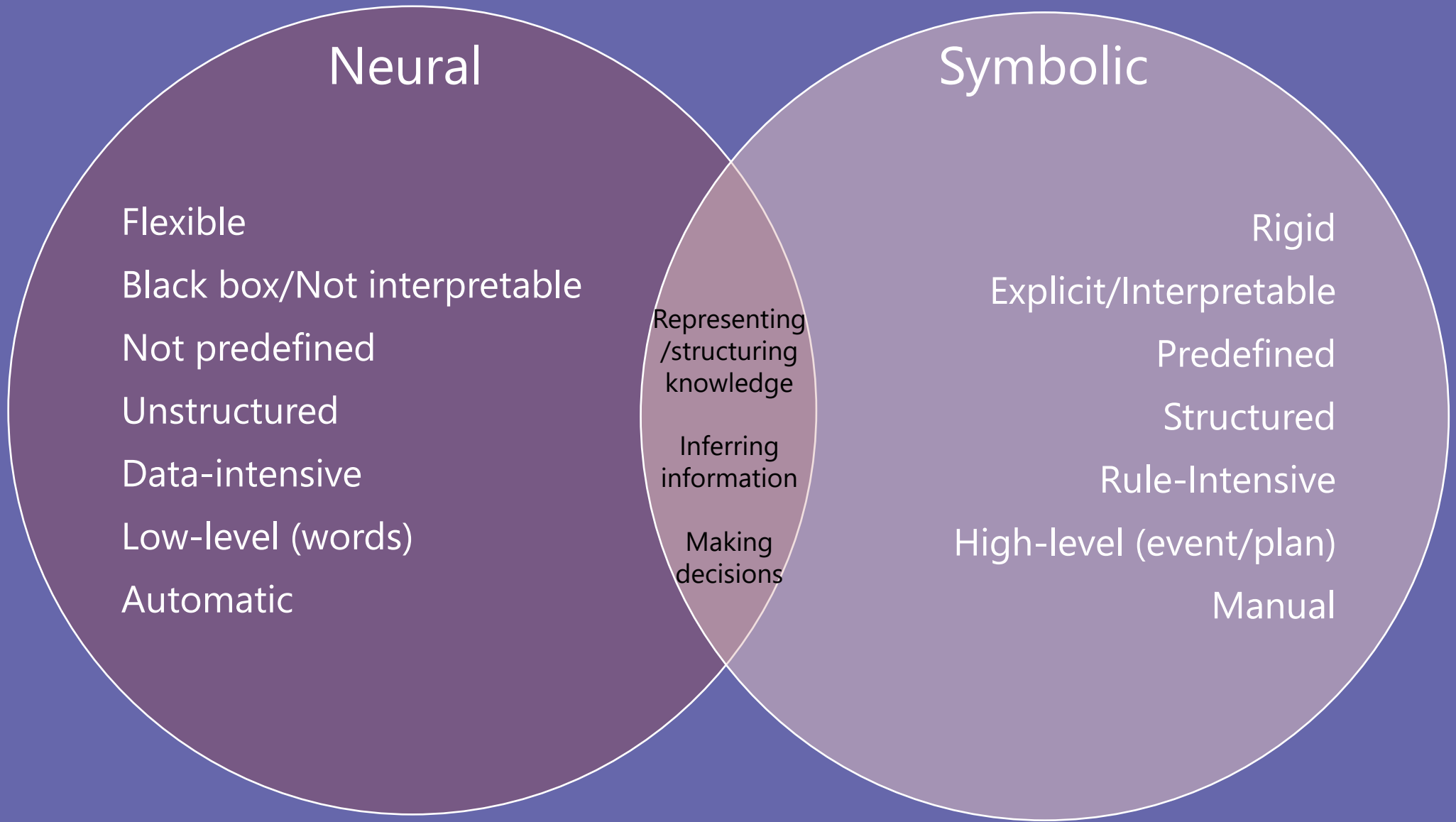
```
(:action go
   :parameters (?dir - direction
                ?p - player
                ?l1 - location ?l2 - location)
   :precondition (and
                  (at ?p ?l1)
                  (connected ?l1 ?dir ?l2)
                  (not (blocked ?l1 ?dir ?l2)))
   :effect (and
                  (at ?p ?l2)
                  (not (at ?p ?l1)))
)
```

Preconditions

Preconditions and effects are **conjunctions** of logical sentences

Effects

These logical sentences are **literals** – positive or negated atomic sentences


Neural
Flexible
Black box/Not interpretable
Not predefined
Unstructured
Data-intensive
Low-level (words)
Automatic
Representing /structuring knowledge
Inferring information
Making decisions
Symbolic
Rigid
Explicit/Interpretable
Predefined
Structured
Rule-Intensive
High-level (event/plan)
Manual