# Improving Coherence and Consistency in Neural Sequence Models with Dual-System, Neuro-Symbolic Reasoning

**Maxwell Nye[1*], Michael Henry Tessler[1], Joshua B. Tenenbaum[1], Brenden M. Lake[2]**
[1]MIT [2]NYU and Facebook AI Research

MITON Times
2021-08-03
Radek Bartyzal

# Human reasoning

- the intuitive and associative (“System 1”)
    - fast, cheap
- the deliberative and logical (“System 2”)
    - slow, expensive

- use system 1 for a quick guess, check it with system 2

# Problem

*A ball and a bat cost **$1.10**.*
*The bat costs one dollar more than the ball.*
*How much does the ball cost?*

| Total cost in prompt | GPT-3 response |
|---|---|
| $1.10 | 10 cents |
| $1.20 | 20 cents |
| $1.30 | $0.30 |
| $1.70 | $0.70 |

# Proposed solution

- System 1: Generation
  - use a pretrained model to generate suggestion
- System 2: Extract facts:
  - parse the suggestion into objects and relations
- System 2: World Model:
  - insert the relations into a hand-made world-model
- if it violates the world model => reject the suggestion and generate a new one

# Example task = generate coherent story

- generate a story, sentence by sentence:

  *Daniel went to the garden. Mary traveled to the office. Daniel grabbed the apple.*

- what’s a better next sentence?

  *(a) Daniel went to the patio. (b) Mary dropped the apple there.*

- Mary does not have the apple = not consistent with the story,

# System 1: Generation of suggestions

- use a pretrained GPT-3 model without any finetuning
  - or a different LM finetuned on desired domain

- simply seed with previous sentences
- and extract next predicted sentence

# System 2: Extract facts

- use a clean GPT-3 without any changes
- **few-shot prompting** to parse the sentence = 8 **handmade** examples:

Please parse the following statements into commands. The available commands are pickup, drop, and go.

Sentence: Max journeyed to the bathroom. Semantic parse: go(Max, bathroom)

Sentence: Mary grabbed the football there. Semantic parse: pickup(Mary, football)

Sentence: <suggested sentence> Semantic parse:

# System 2: Extract facts

- few-shot prompting works surprisingly well
- 100% accuracy on unchanged GPT-3 when parsing the simple sentences
  - checked by humans

# System 2: Mini World Model

- **handmade** world model = set of hard coded rules:

1. Tracks the people, objects and locations which have been mentioned so far.
2. Modifies the world state changes as a result of parsed actions.
3. Checks if the candidate action violates the current world state, as defined by (1) and (2).

# Results

- In a set of 50 generated stories, all stories required at least one sentence to be resampled to maintain coherence
- in QA task = *where is the apple at the end of the story?*
  - orig GPT-3 has 29% accuracy
  - GPT-3 + world model has 100% accuracy
    - (because the parsing had 100% accuracy)

# Another task: coherent family relations: CLUTRR dataset

- it's a QA dataset but can be used a coherent sentences as well:

**Kristin** and her son **Justin** went to visit her mother **Carol** on a nice Sunday afternoon. They went out for a movie together and had a good time.

Q: How is **Carol** related to **Justin** ?

A: Carol is the **grandmother** of Justin

- Generator: BART finetuned on story generation
- Fact Extraction: same = few-shot prompting on GPT-3
- World Model: constraint solver with family relations rules

# Fact extraction

The following sentences contain people and their family relationships. Please parse each sentence into family relationships. The available relationships are sibling, parent, child, grandchild, uncle, spouse. If a sentence has no relationship, say "None".

Sentence: Michael's sister, Mary, was crying, so he told her a joke.

Semantic parse: Mary is Michael's sister.

Sentence: Joshua's son, Clarence, loves trains.

Semantic parse: Clarence is Joshua's child.

# Results

- 36% stories generated by model were coherent
- 93% stories generated by model+world model were coherent
  - with 10 suggestions max

# Conclusion

- GPT-3 can be a good extractor of basic facts
- so why not just train a classifier on top of its embeddings?
- => instead of handmade world model
- because we want something more general than single task classifier
- but not as general as end-to-end GPT-3 because that does not work that well right now

- authors suggest that:
    - world-model could be learned
    - rejected sample used for training

# My ideas

- why not use the world model state as an input to the generator?
- world model would be used as a long term memory + consistency arbiter
- using a simple object-place-actor world model as in 1st task would improve long term coherence