

Part I: Enhancing Text with Graph Structure

Bowen Jin, Yu Zhang, Sha Li, Jiawei Han
Department of Computer Science
University of Illinois at Urbana-Champaign
Mar 4, 2024

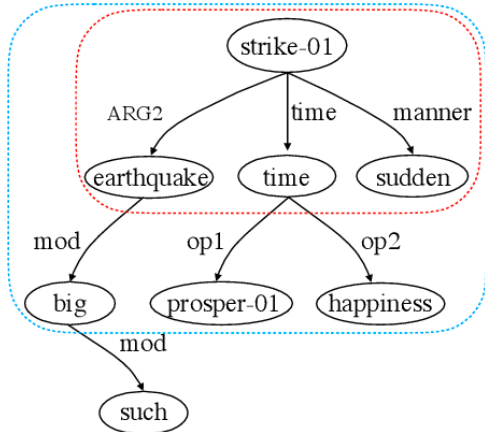
Tutorial Website:



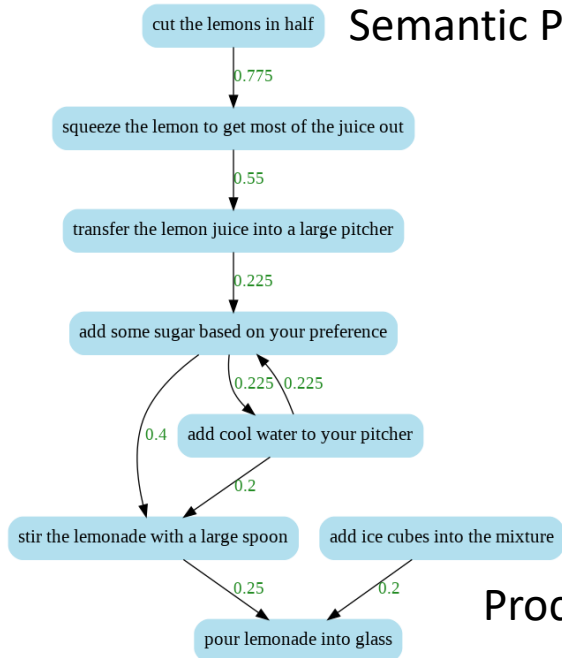
Graph Structures of Language

- ❑ Natural language can be viewed as a **sequence** of words
 - ❑ In the language modeling task, we care about the conditional probability of the next token
- ❑ Natural language also exhibits **structure** (relations and hierarchy)
 - ❑ The relationship between two words in a sentence are not always proportional to their distance
 - ❑ The attention mechanism creates a fully connected weighted graph between tokens
 - ❑ Beyond words, there's also hierarchical relations between higher-level semantic concepts (events, beliefs etc.)

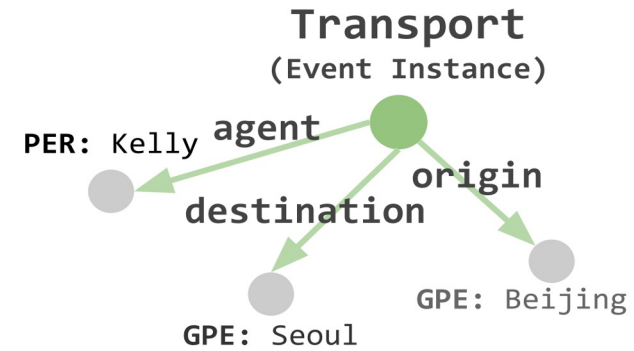
Types of Graphs Structures in Text



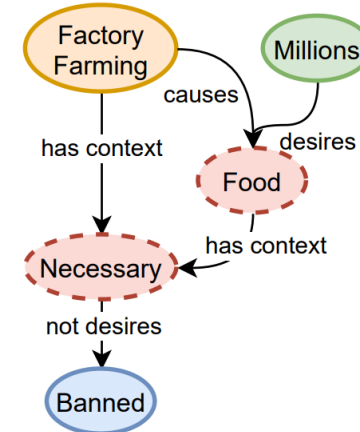
Semantic Parse Graphs



Procedure Graphs




Information Extraction Graphs



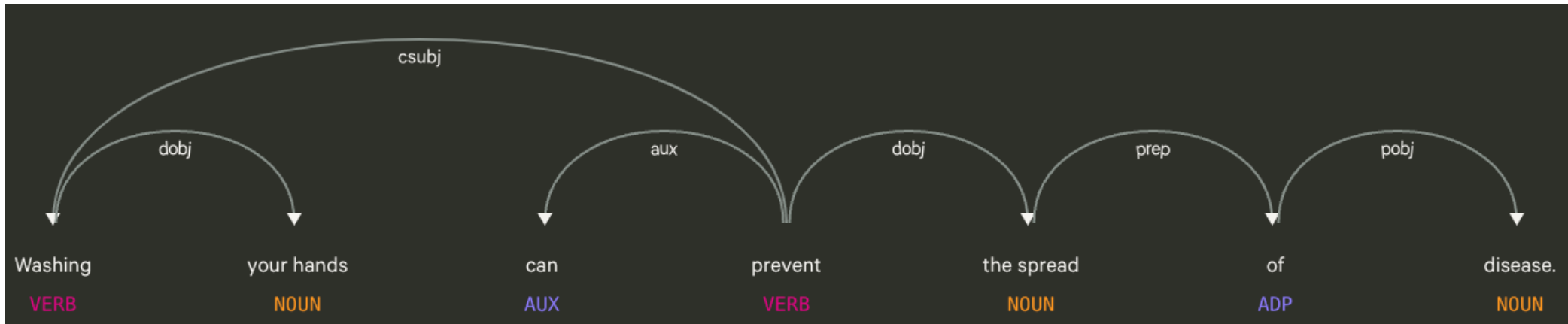
Reasoning Graphs

Outline

- Syntactic and Semantic Parse Graphs 
 - Types of Parse Graphs
 - Parse Graph Applications
- Information Extraction Graphs
- Procedure and Schema Graphs
- Belief and Reasoning Graphs

Dependency Parse Graph

- Dependency graphs
 - Each sentence is transformed into a tree structure
 - Nodes are words in the sentence, edges are dependency tags
 - Available in commonly used NLP packages such as Spacy and Stanza

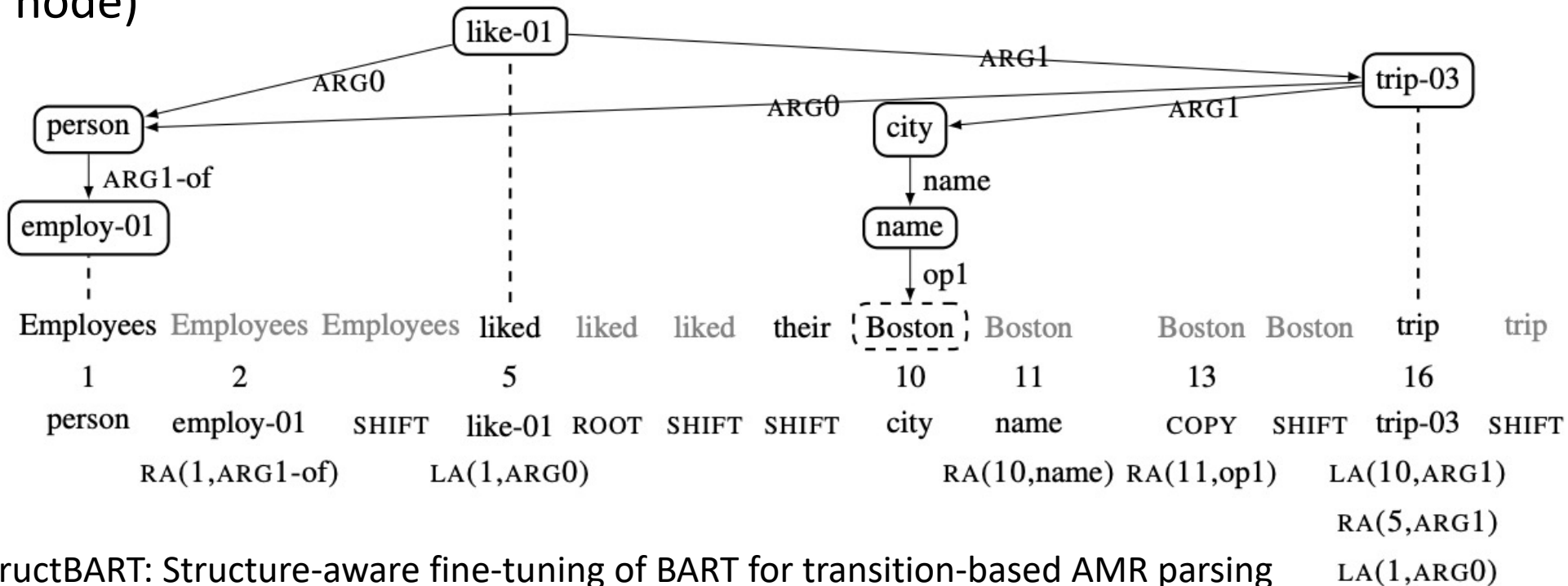


[1] <https://spacy.io>

[2] <https://stanfordnlp.github.io/stanza/>

AMR Graph

- Abstract Meaning Representation (AMR)
 - Each sentence is represented as a tree
 - Nodes are concepts (might be linked to Wikipedia), edges are semantic role labels
 - Intra-sentence co-reference is resolved (“employees” and “their” map to the same node)



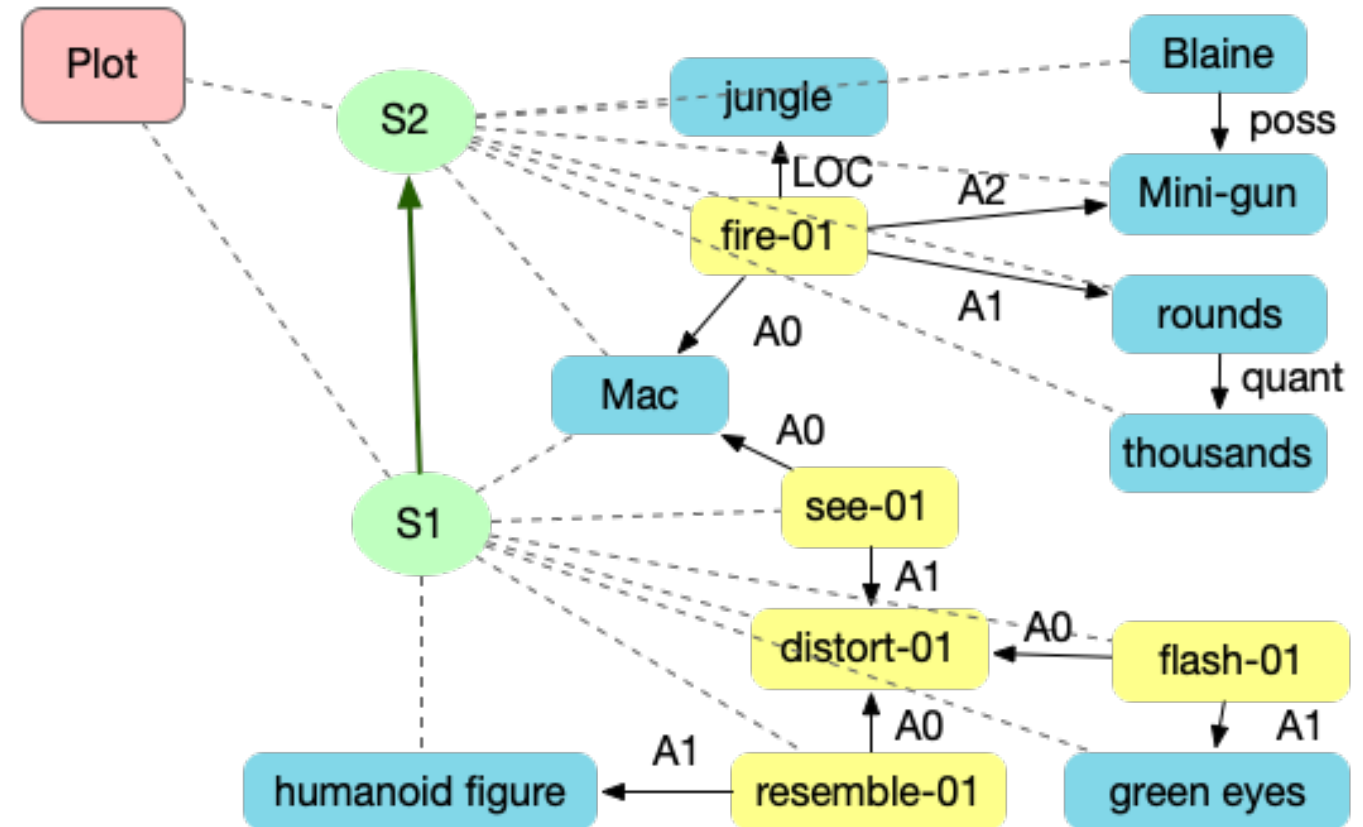
StructBART: Structure-aware fine-tuning of BART for transition-based AMR parsing

LA(1,ARG0)

Document-Level AMR Graphs

- Extend AMR graphs by:
 - Cross-sentence Coreference: merge entities that are coreferential across sentences
 - Sentences Nodes: add edges between the sentence node and the concepts that appear in the sentence
 - Narrative order: Add edges between adjacent sentence nodes
 - Beyond a single document: use source nodes to represent documents

*Plot: Mac sees a humanoid-like distortion that flashes green eyes.
Mac opens fire with Blaine's mini-gun, firing thousands of rounds into the jungle.
The rest of the team rushes to the spot and also opens fire.*



Grounding Dialog Systems in Knowledge

Do you like reading?



Yes, reading is fun.

What you think about the Dune series?

Not informative

I love it.

Dune is a science fiction novel.

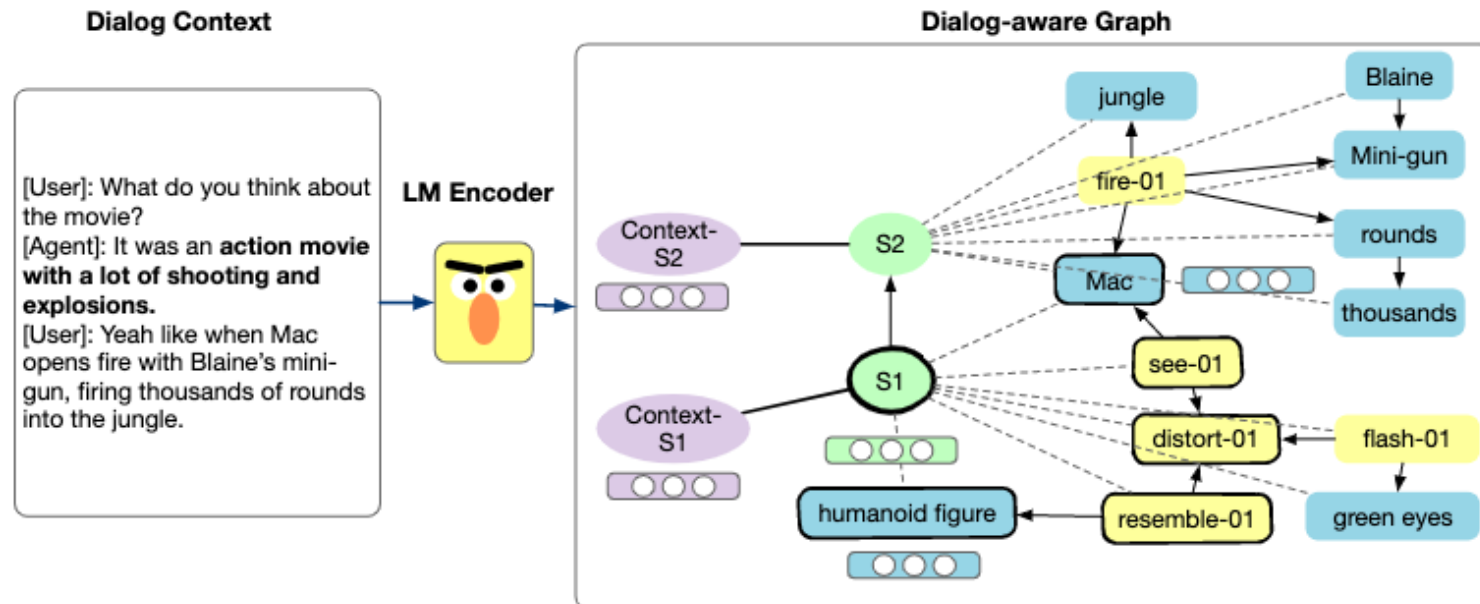
Hallucination

Yes, it was also one of the first works to be published with a cover designed by George Lucas.

- ❑ Without knowledge, dialog responses can be non-informative or suffer from hallucination
- ❑ We need to inject the most relevant knowledge to the dialog context -> the knowledge selection task

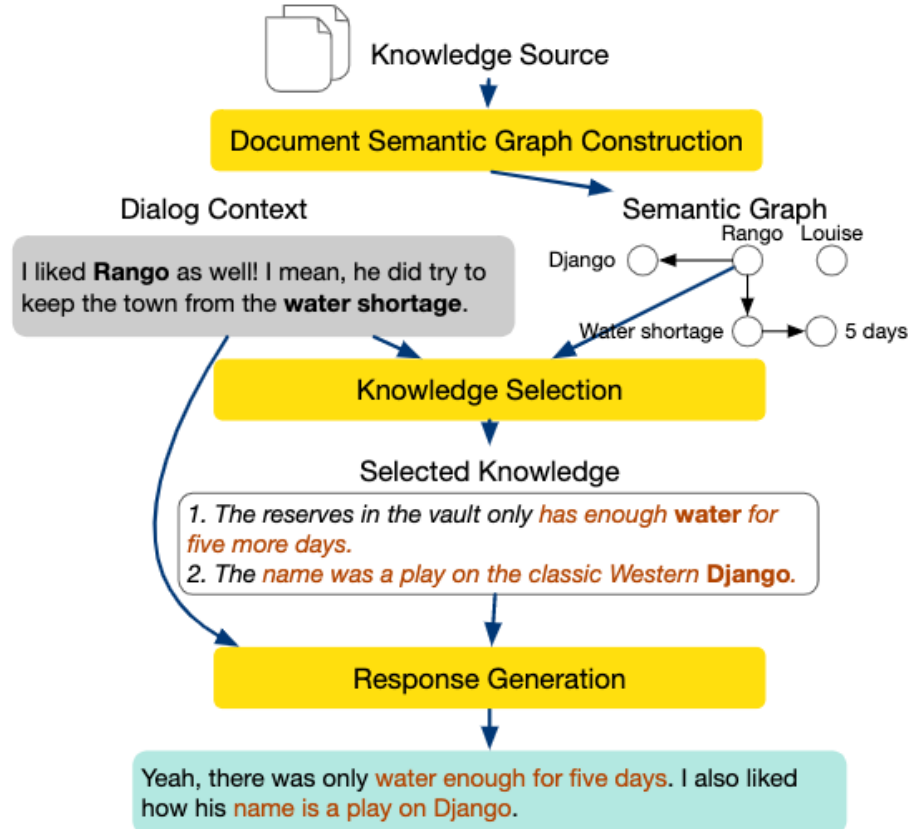
AMR Graphs for Knowledge Selection

- We formulate the knowledge selection problem as **node selection** on the **document semantic graph**
 - Enables knowledge selection on both the sentence-level and the concept-level by selecting different types of nodes
- Contextualize document semantic graph with the dialog
 - For each dialog turn, we encode the dialog context along with each candidate sentence with BERT and then add the context node (purple) to the document semantic graph.



AMR Graphs for Knowledge Selection

- Selected relevant knowledge can be plugged into a generative LM for response generation



Dataset	Method	MAP	Acc
HolIE	Ranking	0.493	0.343
	Graph paths	0.497	0.350
	DocGraph	0.513	0.377
WoW unseen split	Ranking	0.436	0.263
	Graph paths	0.436	0.264
	DocGraph	0.486	0.308

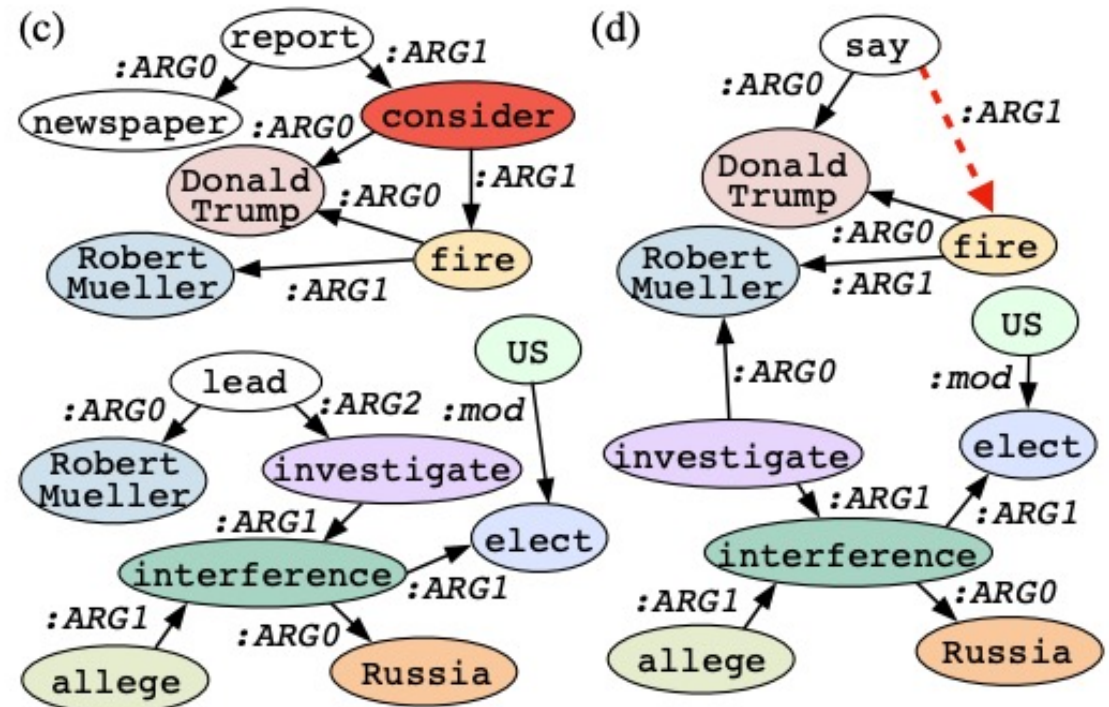
AMR Graphs for Factuality Evaluation

- Evaluate the fine-grained factuality of summaries
 - Main idea: convert the document and the summary to AMR graphs and compare the graphs
 - The red node “consider” is missing from the summary, indicating an error

(a) [...] Mr Mueller was given the role of special counsel by the justice department to lead its investigation into alleged Russian interference in last year's US election [...] The NYT has reported that Mr Trump has considered firing Mr Mueller [...].

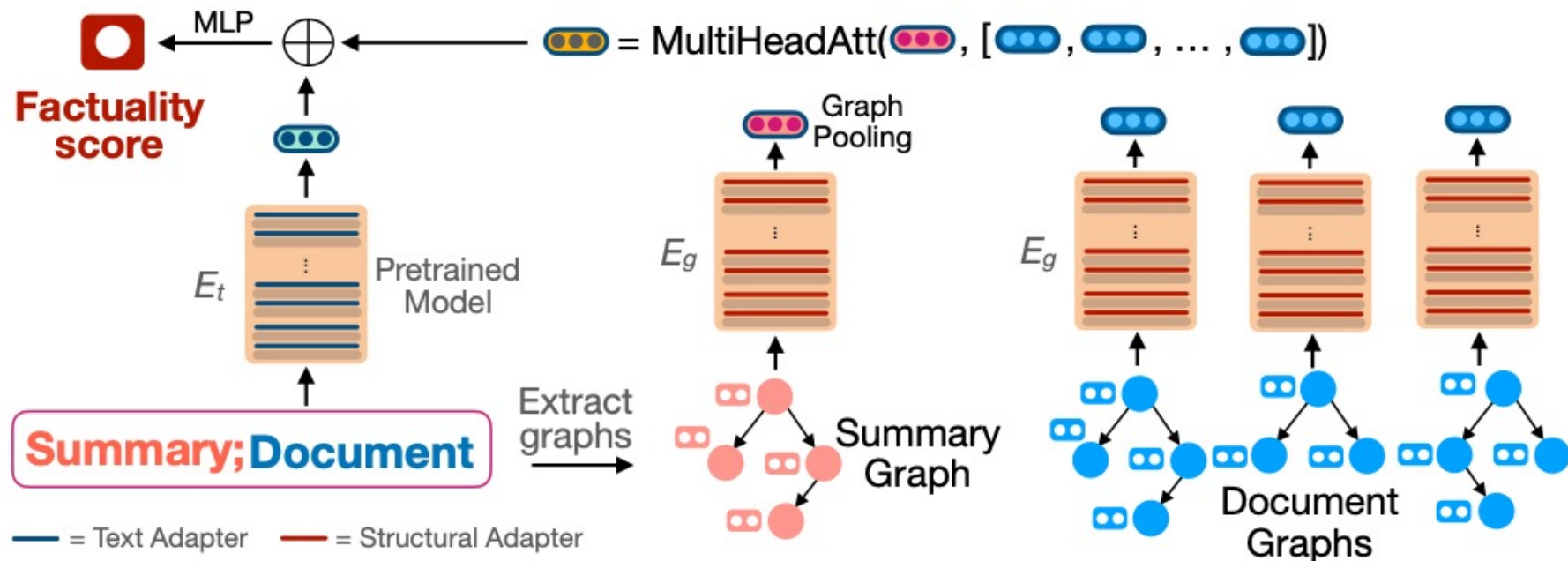
(b) US President Donald Trump has said he will fire special counsel Robert Mueller, who is investigating alleged Russian interference in the US election.

- (a) The original document
- (b) The generated summary
- (c) The AMR graph for the document
- (d) The AMR graph for the summary



AMR Graphs for Factuality Evaluation

- Using AMR graphs to help evaluate the fine-grained factuality of summaries
 - Summary and document graphs are encoded by the graph encoder with structure-aware adapters
 - Both the text-based representation and the graph-based representation are fed into the MLP to predict the factuality score




AMR Graphs for Factuality Evaluation

Model	All data		CNN/DM		XSum	
	BACC	F1	BACC	F1	BACC	F1
QAGS (Wang et al., 2020)	79.8	79.7	64.2	76.2	59.3	85.2
QUALS (Nan et al., 2021)	78.3	78.5	60.8	76.2	57.5	82.2
FACTCC (Kryscinski et al., 2020)	76.0	76.3	69.0	77.8	55.9	73.9
FACTCC+	83.9 (0.4)	84.2 (0.4)	68.0 (1.0)	83.7 (0.5)	58.3 (2.2)	84.9 (1.0)
FACTGRAPH	86.3 (1.3)	86.7 (1.1)	73.0 (2.3)	86.8 (0.8)	68.6 (2.3)	86.6 (2.0)
FACTGRAPH (pretrained structural adapters)	86.4 (0.6)	86.8 (0.5)	74.1 (1.0)	87.4 (0.3)	70.4 (1.9)	85.9 (1.4)
FACTGRAPH (pretrained structural and text adapters)	87.6 (0.7)	87.8 (0.7)	76.0 (2.8)	87.5 (0.4)	69.9 (2.3)	88.4 (1.2)

- ❑ FactCC+ was pretrained on synthetic data
- ❑ FactGraph > FactCC+: semantic graph representations are beneficial for factuality evaluation
- ❑ Further pretraining the structural adapters and text adapters boosts performance

Outline

- Syntactic and Semantic Parse Graphs
- Information Extraction Graphs 
 - Relation Extraction
 - Event Extraction
 - Coreference Resolution
- Procedure and Schema Graphs
- Belief and Reasoning Graphs

Relation Extraction

- Given a head entity, tail entity and the context containing both entities, classify the relation between them (could be NULL)

Task Setting:

Sentence: *It's a meeting of L.C.K., a civil rights organization founded by Shawn.*

Head Entity: *L.C.K.*

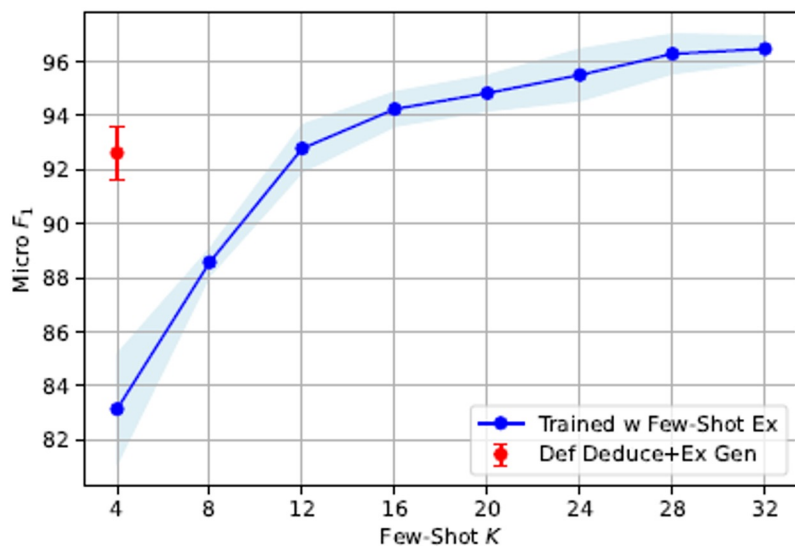
Tail Entity: *Shawn*



Relation between Head & Tail Entities: org:founded_by

RePaL: Definition Based Relation Extraction

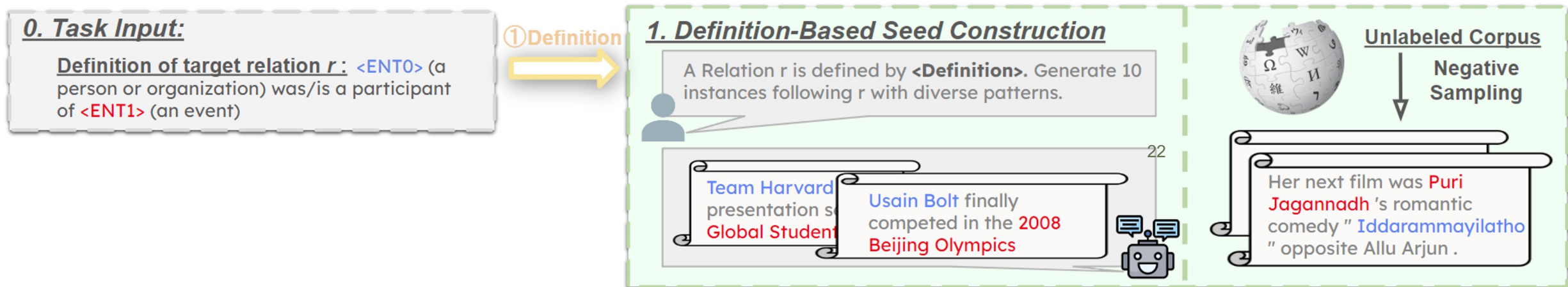
- Definitions are often more available than few-shot examples
- A single definition is worth many examples
- Define-and-then-generate extends the patterns conveyed by few-shot instances



Gold Definition	Gold Few-Shot Instances For Derivation	Derived Definition
<ENT1> was/is the occupation of <ENT0> (a person)	<ol style="list-style-type: none"> <ENT0>Pierre Maudru</ENT0> (1892\u20131992) was a French <ENT1>screenwriter</ENT1> . Goble p.189 He also directed three films . WWF Hall of Famer Bob Backlund and Extreme Championship Wrestling <ENT1>manager</ENT1> <ENT0>Bill Alfonso</ENT0> also made surprise appearances during the event . In May 2010 , Paratici moved from Sampdoria to Juventus , along with Director General Giuseppe Marotta and <ENT1>Manager</ENT1> <ENT0>Luigi Delneri</ENT0> . <ENT0>Else Reval</ENT0> (14 June 1893 \u2013 25 January 1978) was a German <ENT1>film actress</ENT1> . Giesen p.210 	<ENT1> is the profession in which <ENT0> (a person) works or has worked.
<ENT0> (a person or organization) was/is a participant of <ENT1> (an event)	<ol style="list-style-type: none"> He only saw limited action in <ENT1>Euro 2000</ENT1> as cover for left - back <ENT0>Arthur Numan</ENT0> . <ENT0>Francesco Cameli</ENT0> was a sailor from Italy , who represented his country at the <ENT1>1928 Summer Olympics</ENT1> in Amsterdam , Netherlands . <ENT0>Giannin Andreossi</ENT0> (born July 2 , 1902 , date of death unknown) was a Swiss ice hockey player who competed in the <ENT1>1928 Winter Olympics</ENT1> . <ENT0>Ren\u00e9 Sch\u00fcrmann</ENT0> (born February 3 , 1962) is a German speed skater who competed for East Germany in the <ENT1>1984 Winter Olympics</ENT1> . 	<ENT1> is the major international sports competition in which <ENT0> (an athlete) has competed.

Methodology: Definition-based seed construction

- ❑ Query LLM for initial positive examples given the definition
- ❑ Obtain negative examples by random sampling over an unlabeled corpora
 - ❑ Hypothesis: *in a large-scale unlabeled corpus, the proportion of target relation instances is relatively small*



Methodology: Training RE-specialized SLM

- ❑ Leverage small language models (SLMs) as task-specialized extractors for better performance with low cost
 - ❑ We formulate RE as an NLI task for fine-tuning

Premise_j : = s^j,
 Hypothesis_j : = d(E₀ = e₀^j, E₁ = e₁^j).

Given a SLM model \mathcal{M} , we obtain the encoded sequence hidden states \mathbf{H} by:
 $\mathbf{H} = \mathcal{M}(\text{Premise}_j [\text{SEP}][\text{SEP}] \text{Hypothesis}_j)$

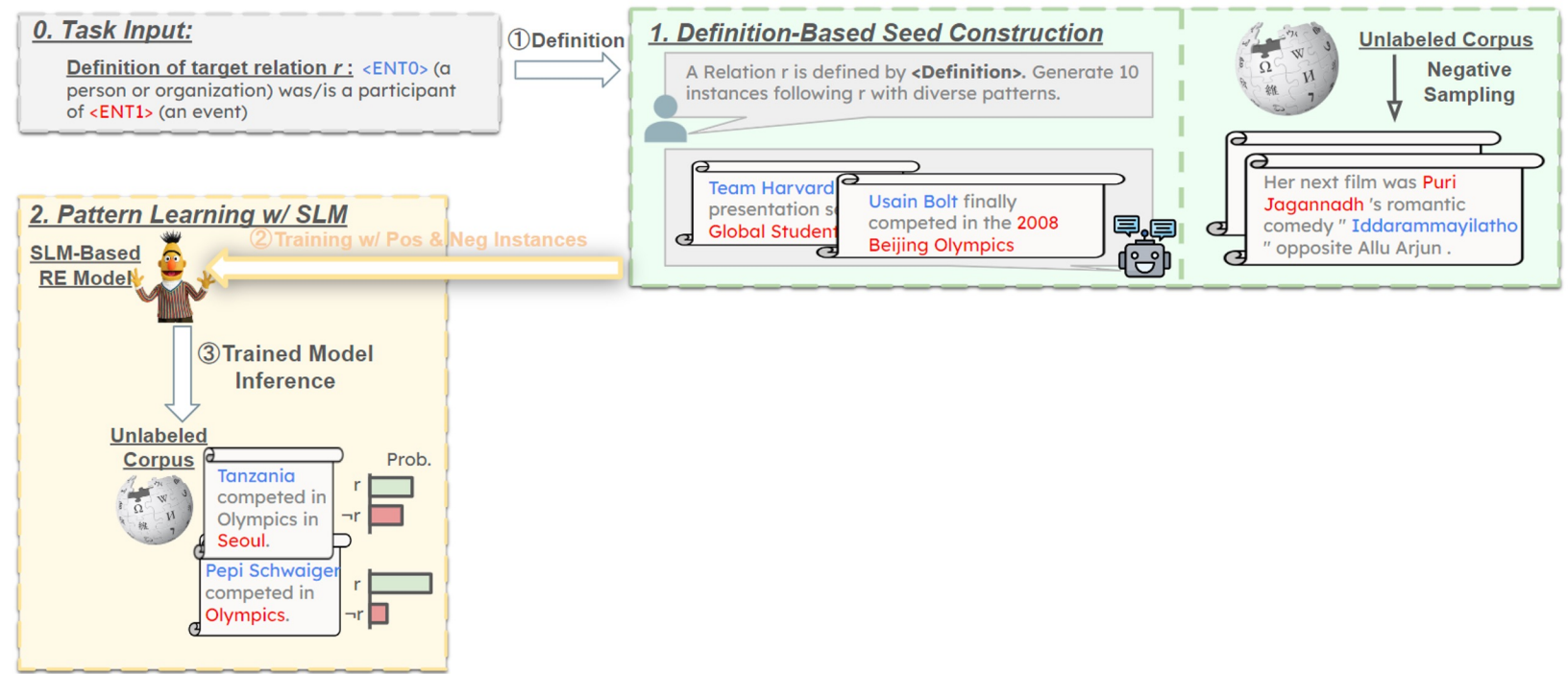
and the NLI logits $\mathbf{z} = [z_E, z_N, z_C] \in \mathbb{R}^3$ is computed as:

$$\mathbf{z} = \mathbf{W} \cdot \mathbf{H}_{[\text{CLS}]} + \mathbf{b}.$$

Finally, P_j , the probability of instance (s^j, e_0^j, e_1^j) following relation $r(E_0, E_1)$, is computed as the normalized logit of ENTAILMENT label:

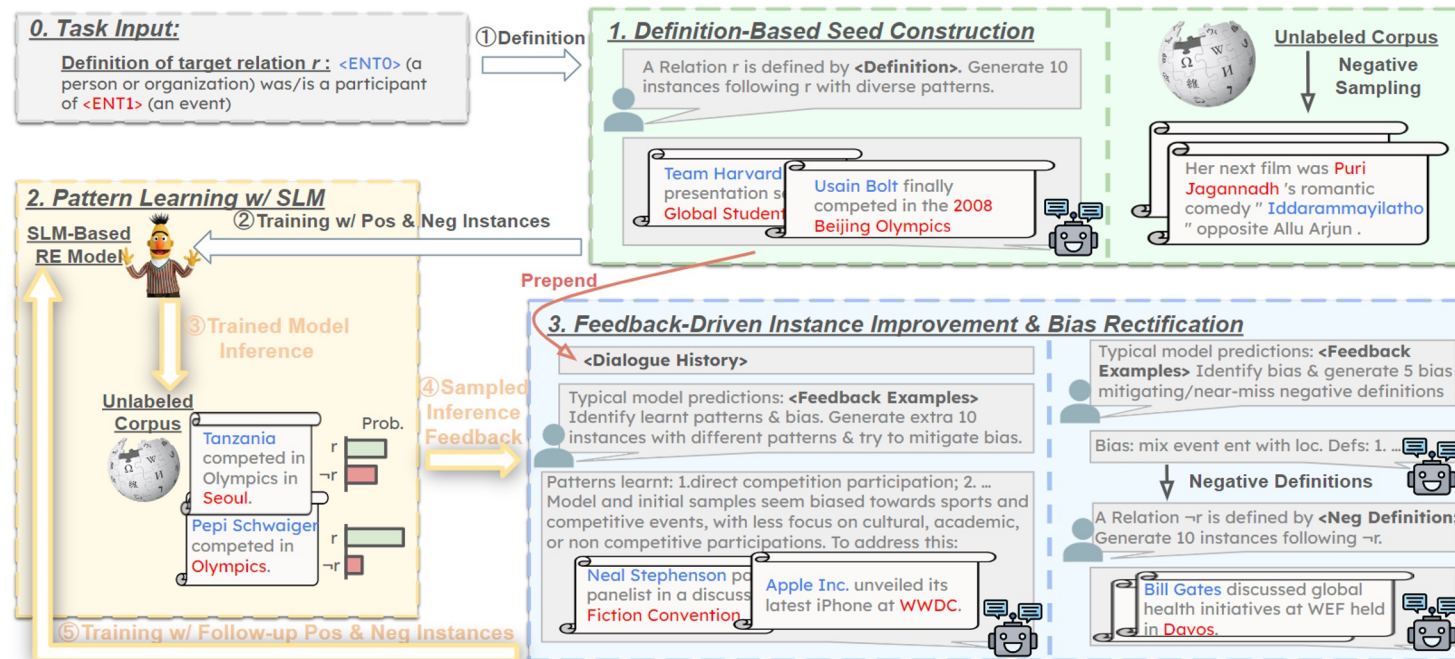
$$P_j = \frac{e^{z_E}}{\sum_{c \in \{C, N, E\}} e^{z_c}},$$

where C, N, E denote logits \mathbf{z} 's indices for NLI label CONTRADICTION, NEUTRAL, ENTAILMENT respectively.



Methodology: Feedback-driven improvement

- Inference on the unlabeled corpora with trained SLM
- Query LLM to generate feedback
 - Sample positive and negative instances within certain confidence intervals
 - Ask the LLM to generate instances with different patterns to mitigate bias



RePaL: Performance

Model	DefOn-FewRel			DefOn-ReTACRED		
	Precision	Recall	F ₁	Precision	Recall	F ₁
<i>Fully-Supervised</i>						
ROBERTA NLI	77.34	99.06	85.89	77.61	98.35	86.03
<i>Few-Shot</i>						
GPT-3.5 ICL	46.65	72.86	54.97	40.28	64.55	48.72
<i>Zero-Shot</i>						
RANDOM GUESS	7.14	49.76	12.49	9.10	50.02	15.37
GPT-3.5	50.79	69.76	54.23	49.58	38.48	42.52
ROBERTA NLI	39.67	94.34	49.40	26.94	97.55	40.64
ZS-BERT*	41.00	40.51	40.73	20.20	17.70	18.81
RELATIONPROMPT*	75.18	66.08	70.34	51.67	51.26	51.40
RELATIONPROMPT (NoGEN)*	0.00	0.00	0.00	2.98	0.83	1.25
RE-MATCHING*	76.47	71.80	74.05	54.43	50.22	52.16
REPAL (Ours)	78.19	82.25	77.93	68.47	80.52	68.42

- Zero-shot models marked by * are trained on 61 seen relation instances from FewRel and require all negative test relations to be known
- Better overall performance against existing zero-shot methods
- Larger margin for ReTACRED, which require transfer-learning based zero-shot models to generalize across domains

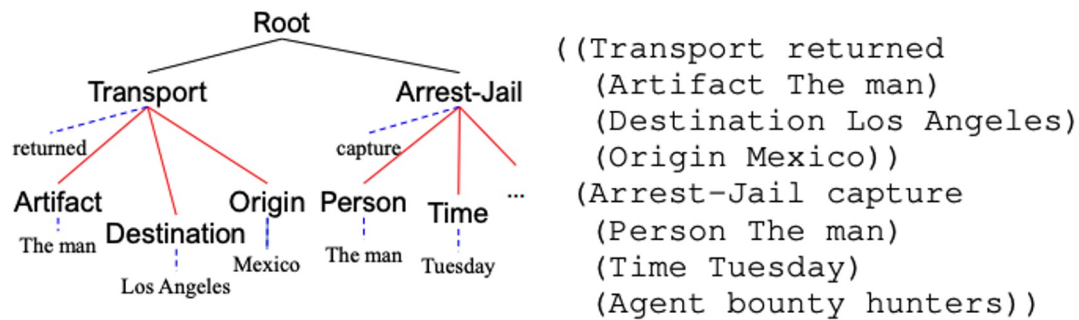
Event Extraction Overview

- Events represent dynamic changes of state
 - While relation extraction involves two entities, events typically involve 4-5 entities (agent, patient, instrument, time, location etc.)
 - The arguments of an event depend on the event type
- Event extraction is typically separated into 2 stages of **event detection** and **argument extraction**

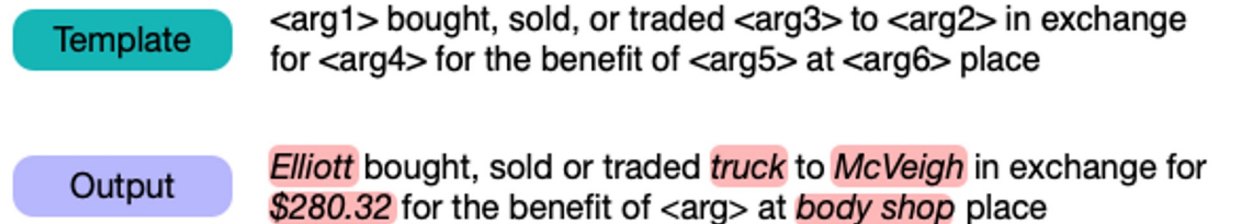
In May, after a 12-week trial, Tsarnaev [Killer] was found guilty of killing [Life.Die] three people [Victim] and injuring [Life.Injure] 264 in the April 15, 2013 bombing at the world-renowned race, where he, and his brother, 26, set off [Conflict.Attack.DetonateExplode] two pressure-cooker bombs near the finish line [Place].

LLMs for Event Extraction

- ❑ Generative language models are trained to predict the next token
- ❑ IE tasks require translating text to structures
- ❑ How do we reconcile the difference?



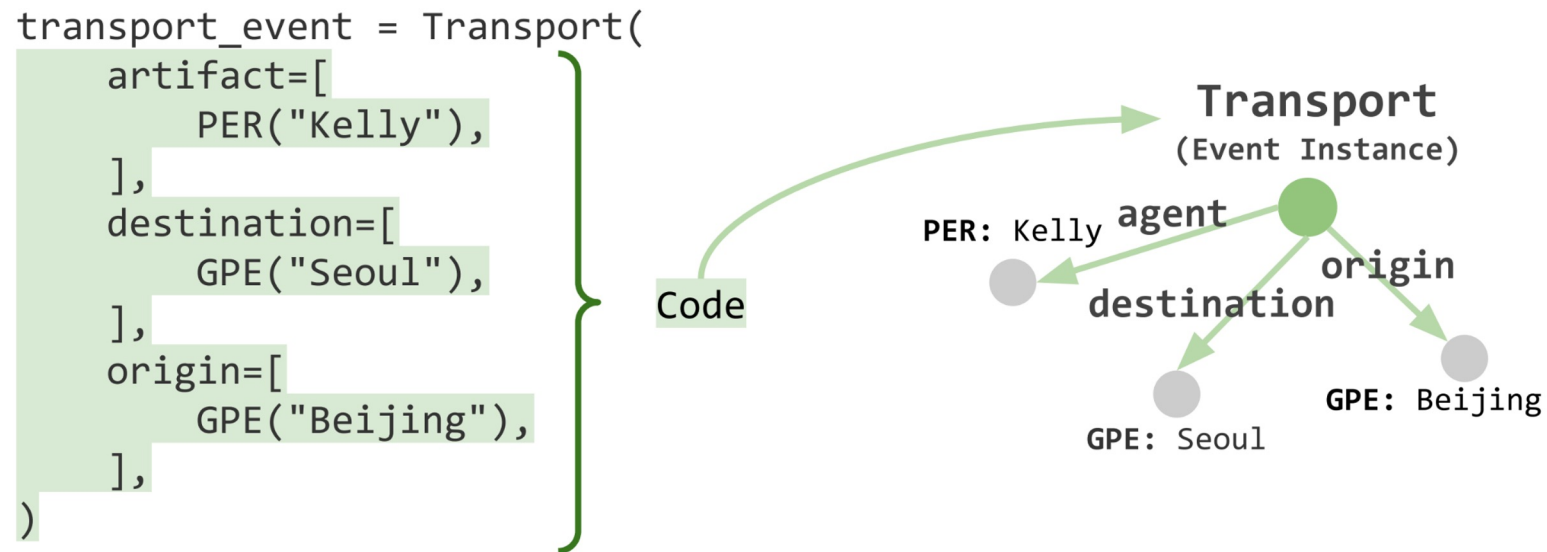
Linearized output



Templated output

Code Language Models for Event Extraction

The output structure of some NLP tasks (e.g., Event Argument Extraction) can be mapped to code in a more straightforward way compared to natural language.



Insight: Leverage such text-to-code capability of LLM to solve structured prediction problems in NLP

Mapping Between Event Argument Extraction and Programming Language

EAE task closely aligned with features of Python Programming Language.

Event Argument Extraction	Programming Language (Python)
Event / Entity Type Transport, VEH	Class definition <pre>class Transport, class VEH</pre>
Hierarchical Events Movement:Transport	Inheritance Inheritance is a way to create a hierarchy of classes in PL. A child class can base upon another class, retaining similar implementation. <pre>class Transport(Movement)</pre>
Event Arguments vehicle	Function arguments <pre>def function(vehicle=...)</pre>
Argument Constraint Each argument can has multiple entities; Argument <i>vehicle</i> should be entities of type VEH .	Type Annotation Type annotations are used by developers to indicate the data types of variables and input/outputs of functions. <pre>def function(vehicle: List[VEH] = [], ...)</pre>
Weakly-supervised Information Transport Event describes <i>someone</i> transporting <i>something</i> in a <i>vehicle</i> from <i>one place</i> to <i>another place</i> .	Docstring or Comments <pre>class Transport(Movement): """ self.agent transported self.artifact in self.vehicle vehicle from self.origin place to self.destination place. """</pre>

We can use **Inheritance** to represent hierarchical event relationships

We can also leverage **type annotation** to annotate the entity types accepted for each argument

How to Prompt LLM for EAE?

Each prompt has 3 components:

- (1) Ontology context;
- (2) K -shot examples for in-context learning
- (3) Task prompt

```
from typing import List
class Entity:
    def __init__(self, name: str):
        self.name = name
class Event:
    def __init__(self, name: str):
        self.name = name
```

Base Class
Definition

```
class ORG(Entity):
    def __init__(self, name: str):
        super().__init__(name=name)
class GPE(Entity):
    """Geopolitical entities such as countries, provinces,
    states, cities, towns, etc. GPEs are composite entities,
    consisting of ..."""
    def __init__(self, name: str):
        super().__init__(name=name)
```

Ontology
Context

Relevant Entity Definition(s)

Event Definition

(optional) k In-context Examples

```
"""
Translate the following sentence into an instance of
Transport. The trigger word(s) of the event is marked
with **trigger word**.
"Kelly , the US assistant secretary for East Asia and
Pacific Affairs , **arrived** in Seoul from Beijing
Friday to brief Yoon , the foreign minister ."
"""
transport_event = Transport(
```

Task
Prompt

How does Code4Struct Perform on EAE?

Model	Data	Arg-I F1	Arg-C F1
DyGIE++	Full	66.2	60.7
BERT-QA	Full	68.2	65.4
OneIE	Full	73.2	69.3
TANL	Full	65.9	61.0
BART-Gen	Full	69.9	66.7
DEGREE	Full	76.0	73.5
DEGREE	50-shot*	40.8	37.3
CODE4STRUCT	50-shot*	62.0	58.1
CODE4STRUCT	0-shot	50.0	35.7

50-shot Code4Struct rivals fully-supervised approaches trained on 4,202 instances of the training data

In 50-shot setting, it **surpass current SOTA by a large margin** (20.8% absolute F1 difference on Arg-C)

0-shot can already achieve higher Arg-I performance than 50-shot DEGREE

Is Code Prompt any better than Text Prompt?

To compare our code-based prompt with text-style GPT-3 prompt, we design a text prompt mimicking our code prompt.

We compare the performance of **text prompt** and **code prompt** on GPT-3 (text-davinci-002) and Codex (code-davinci-002) through OpenAI API.

Description of base entity types:

GPE: Geopolitical entities such as countries, provinces, states, cities, towns, etc. GPEs are composite entities, consisting of a physical location, a government, and a population. All three of these elements must be present for an entity to be tagged as a GPE. A GPE entity may be a single geopolitical entity or a group.
... (other types omitted for space)

(1) Entity Definition(s)

Role definition of event type Transport (Parent type: Movement):

1. agent (need to be one of GPE or ORG or PER)
2. artifact (need to be one of FAC or ORG or PER or VEH or WEA)
3. destination (need to be one of FAC or GPE or LOC)
4. origin (need to be one of FAC or GPE or LOC)
5. vehicle (need to be one of VEH)

Multiple entities can be extracted for the same role, each entity is a double-quote enclosed string.

Each extracted entity should look like: (Base Entity Type) "content of extracted string"

If entity is not present in the text, write: () ""

Different entities are delimited by a comma.

In this event: [agent] transported [artifact] in [vehicle] vehicle from [origin] place to [destination] place.

(2) Event Definition

Translate the following sentence into an instance of Transport event. The

trigger word(s) of the event is marked with ****trigger word****.

Sentence: "Renowned Hollywood madam Heidi Fleiss has been ****flown**** to Melbourne as guest of honour at Thursday's market debut and , according to Harris , has already played a key role in attracting worldwide media attention to the event ."

1. agent: () ""
2. artifact: (PER) "Heidi Fleiss"
3. destination: (GPE) "Melbourne"
4. origin: () ""
5. vehicle: () ""

(3) In-context Examples

Translate the following sentence into an instance of Transport event. The trigger word(s) of the event is marked with ****trigger word****.

Sentence: "Kelly , the US assistant secretary for East Asia and Pacific Affairs , ****arrived**** in Seoul from Beijing Friday to brief Yoon , the foreign minister ."

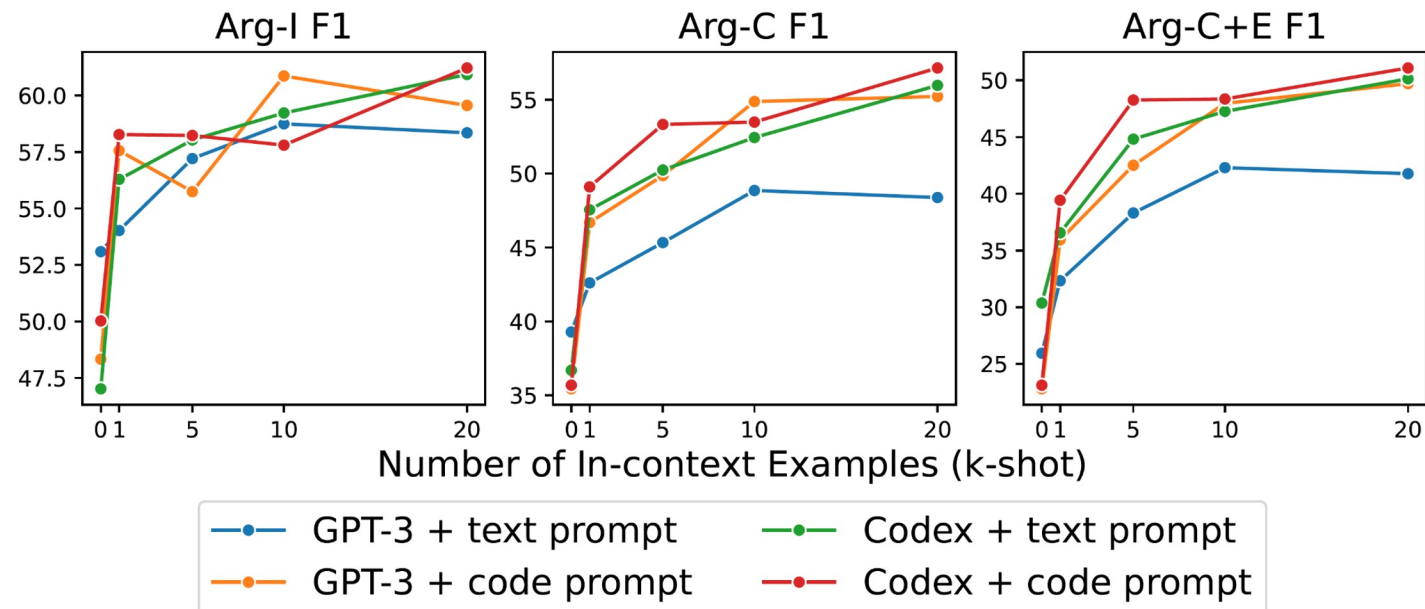
1. agent: () ""
2. artifact: (PER) "Kelly"
3. destination: (GPE) "Seoul"
4. origin: (GPE) "Beijing"
5. vehicle: () ""

(4) Event Instantiation

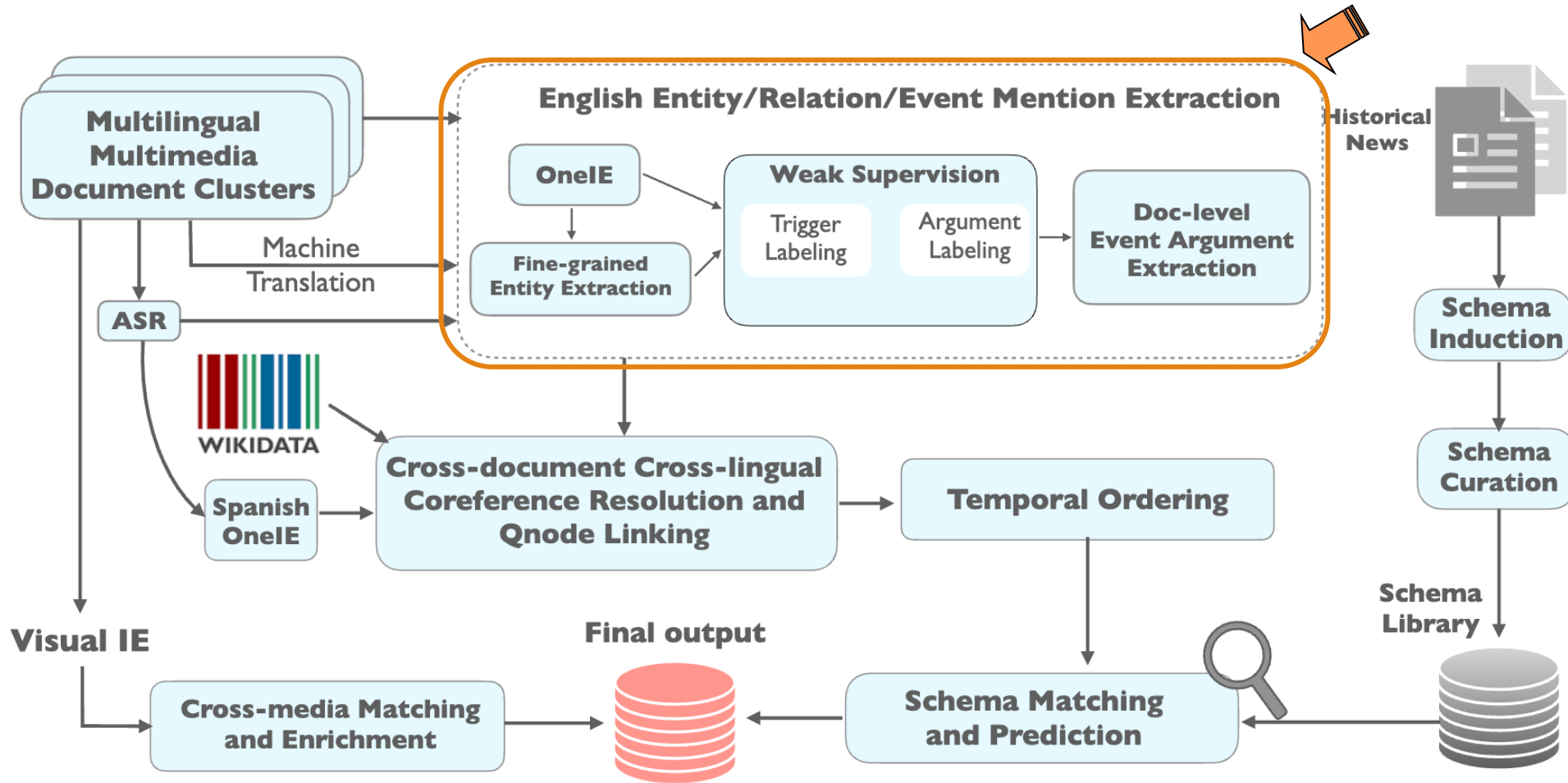
Is Code Prompt any better than Text Prompt?

Our findings


- (1) Codex + code outperforms all text prompts on Arg-C and Arg-C+E under few-shot settings. The performance gap is most significant on Arg-C F1 (8.7% absolute F1 difference when compared to GPT-3 + text prompt).
- (2) Zero-shot code prompt underperforms text prompt on Arg-C for both Codex and GPT-3.
- (3) GPT-3 + code prompt quickly catch-up with Codex + code prompt performance given adequate training examples.



Putting it all together

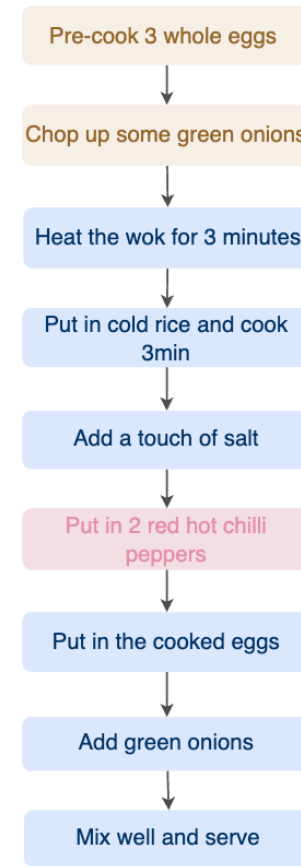
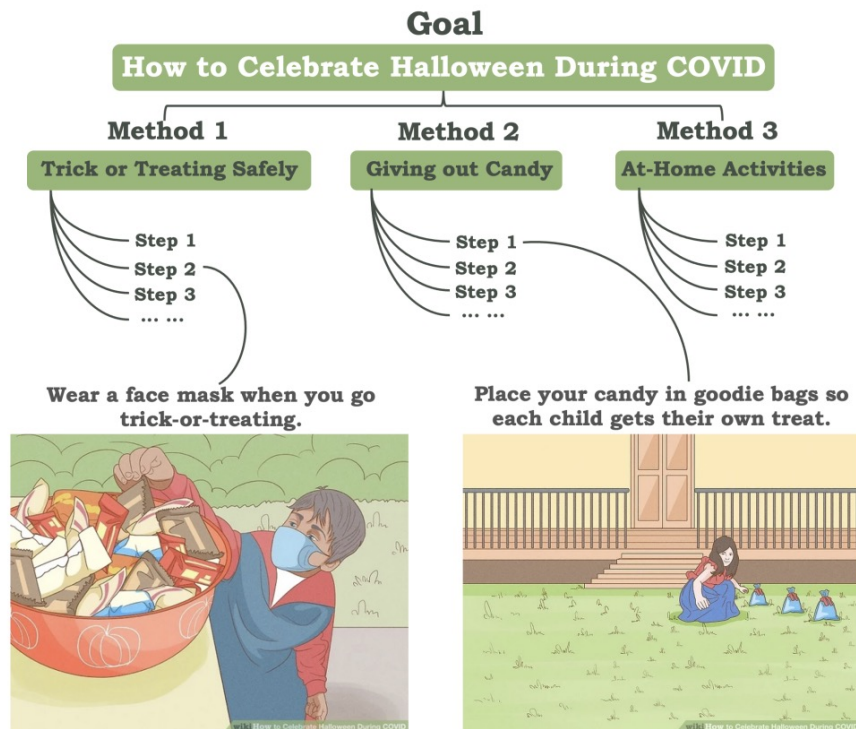


Outline

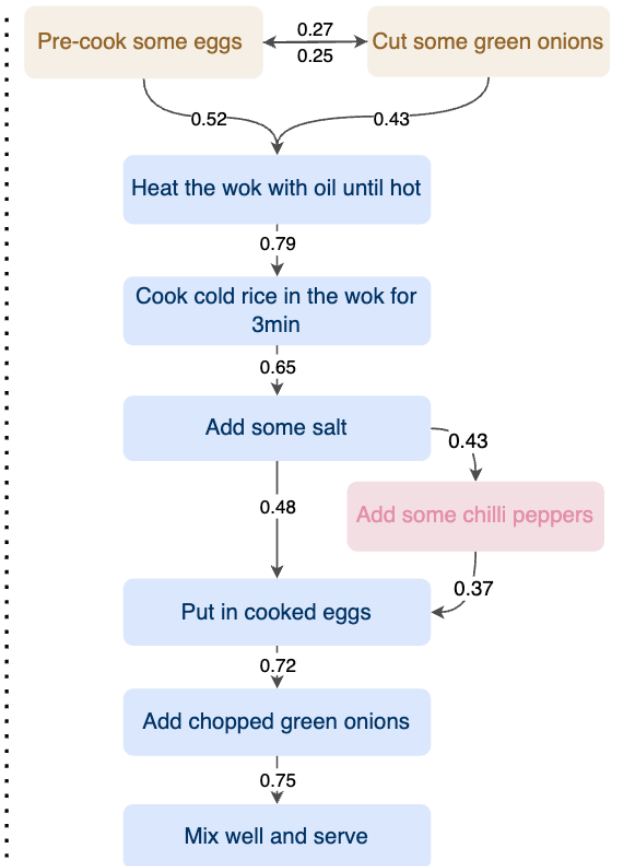
- ❑ Syntactic and Semantic Parse Graphs
- ❑ Information Extraction Graphs
- ❑ Procedure and Schema Graphs 
 - ❑ Procedure graph induction through multimodal alignment
 - ❑ Schema graph induction from data
 - ❑ Schema graph construction from LLMs
- ❑ Belief and Reasoning Graphs

Procedure Graphs

- A procedure consists of a **goal** and a **sequence of steps** that can be carried out to achieve the goal.
- Each node in this graph is a step (short phrase or sentence)

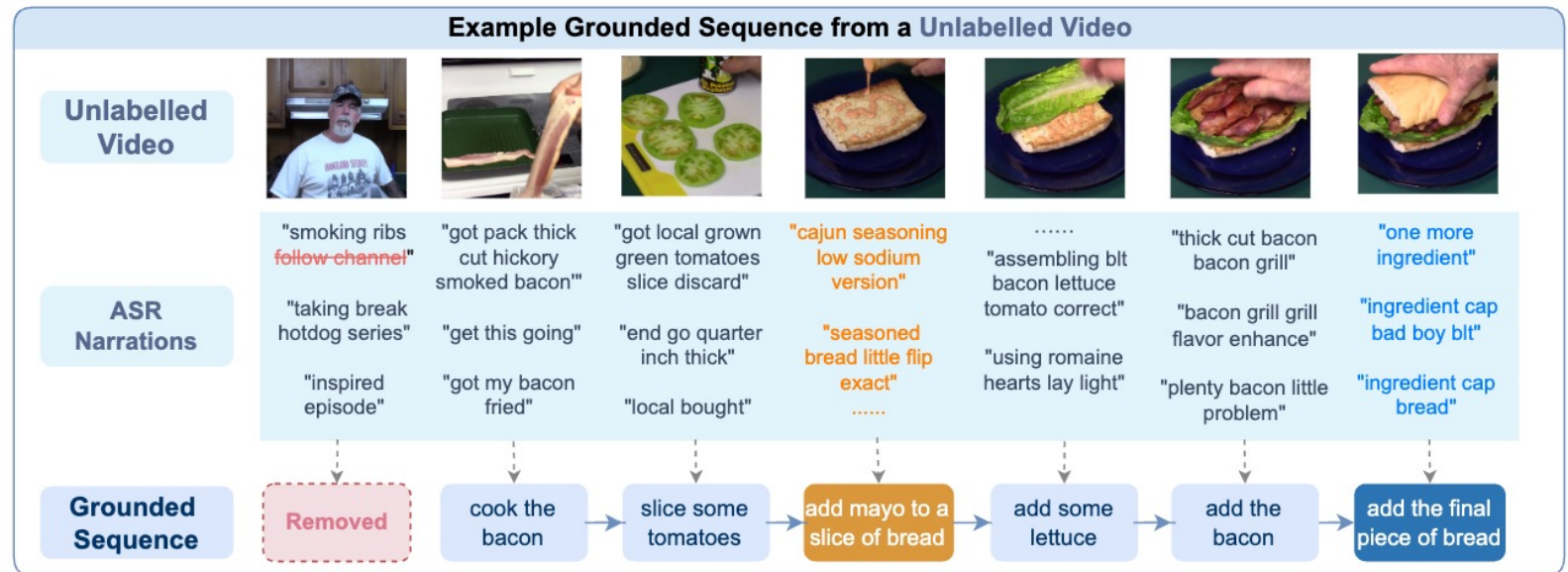
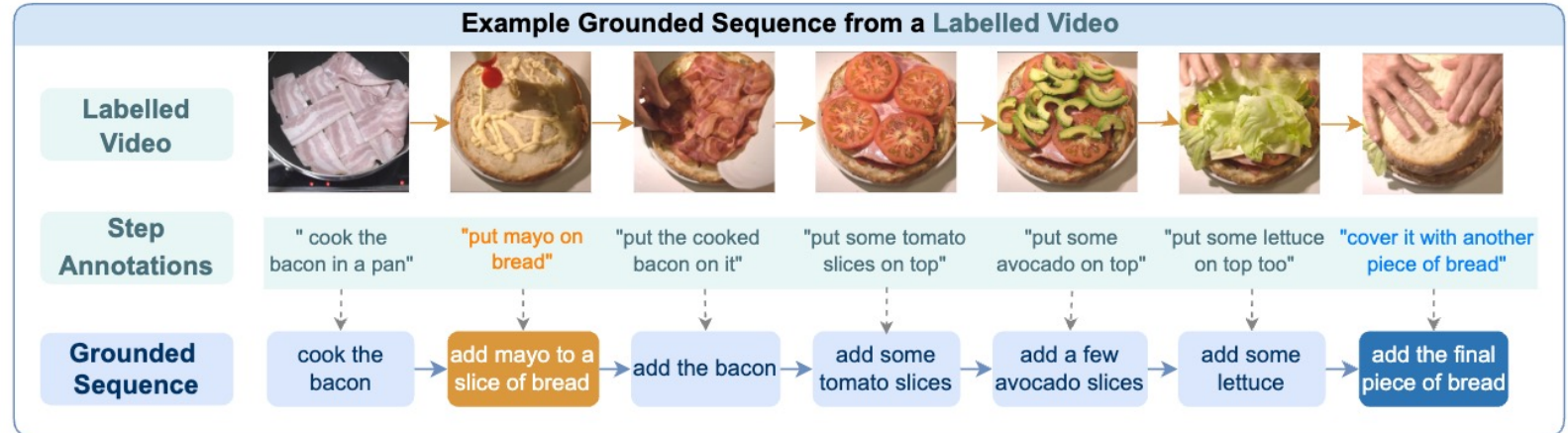
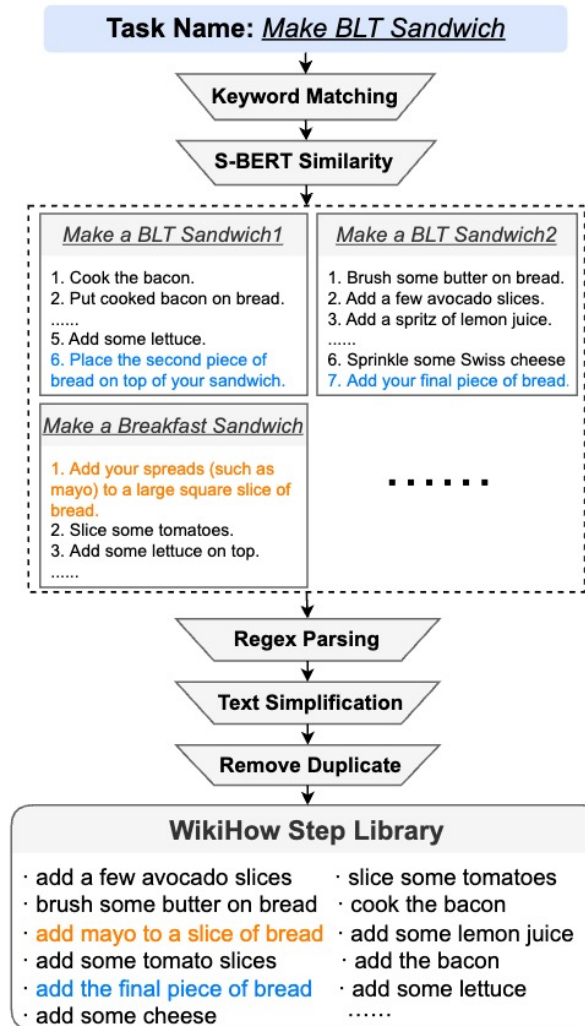


wikiHow
to do anything



Non-sequential Graph Script

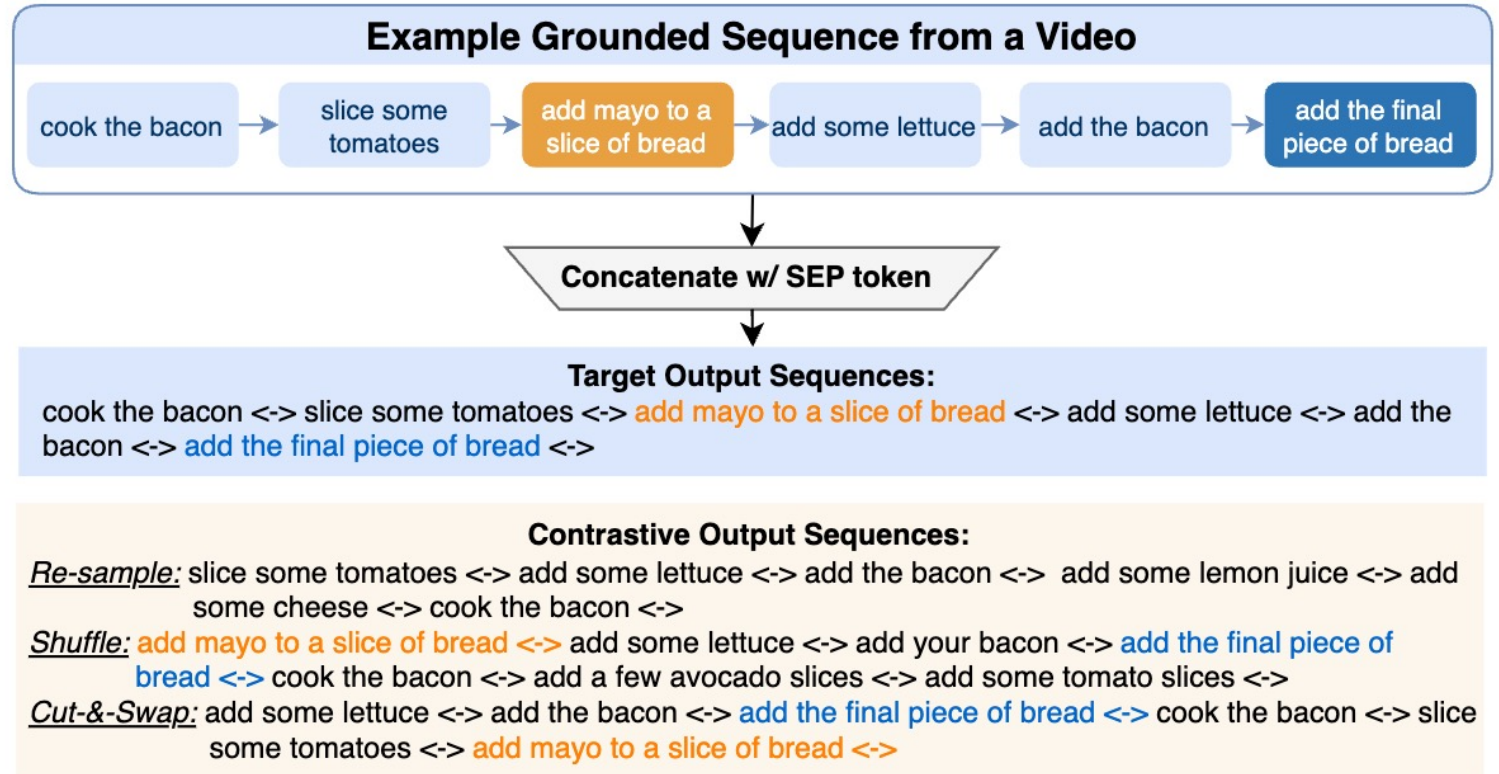
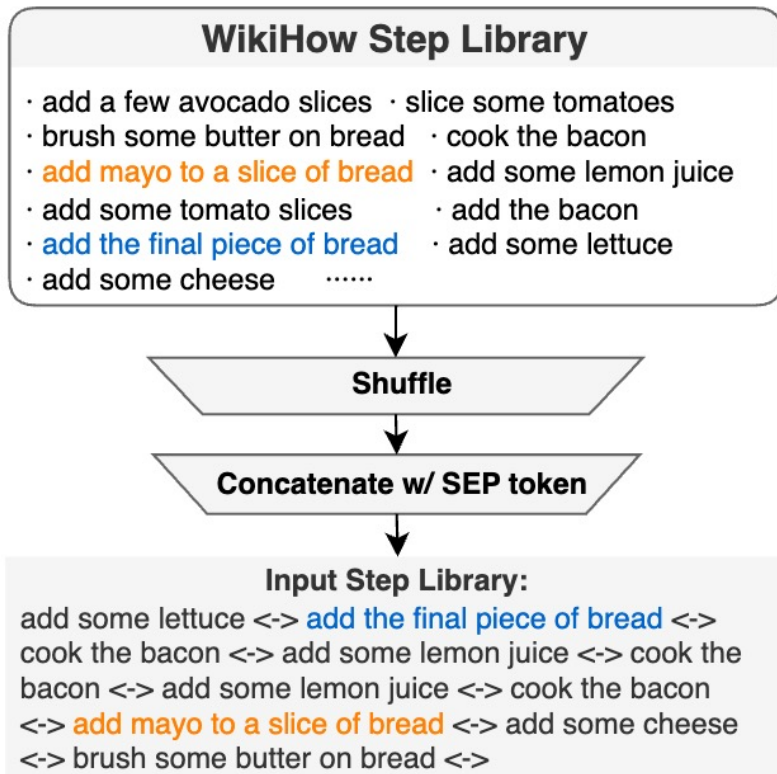
Procedure Graph Induction by Multimodal Alignment



Two-level alignment on the task-level and the step-level.

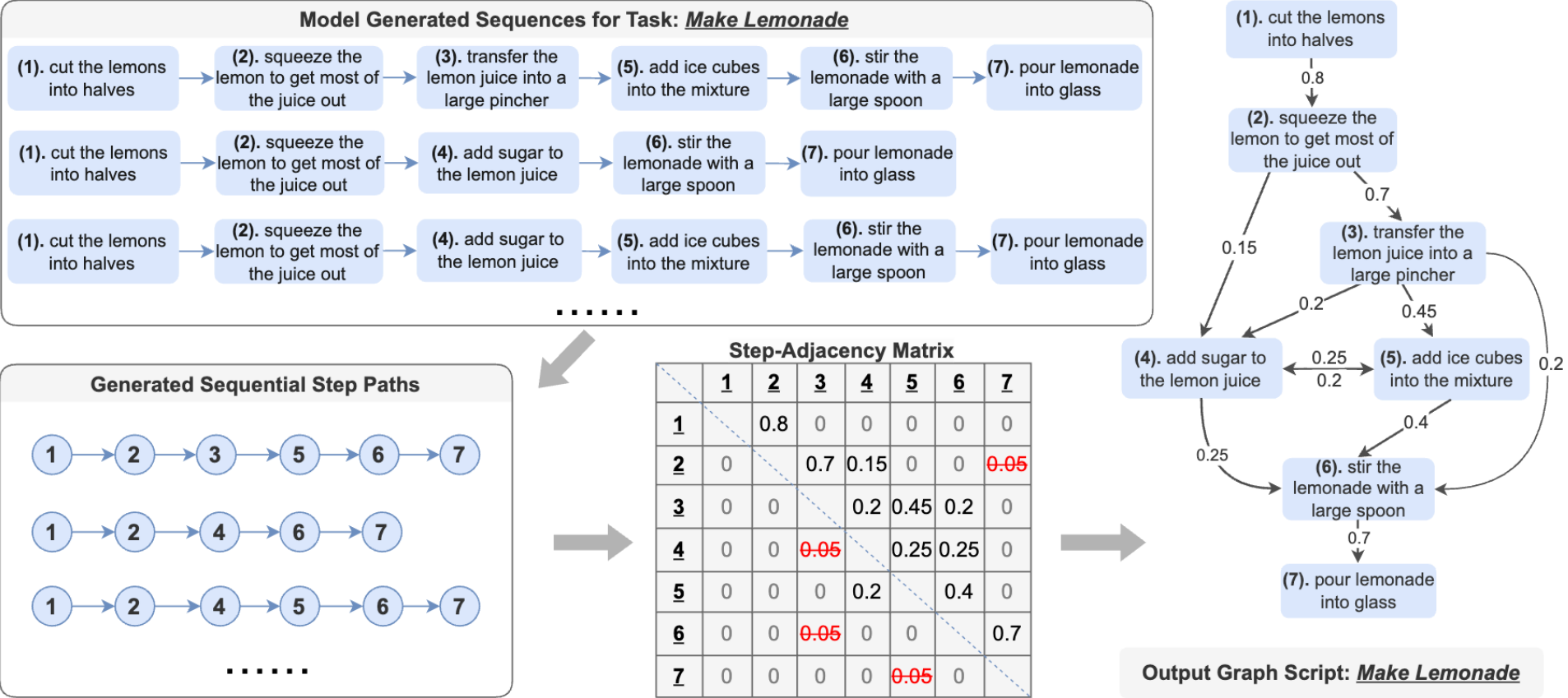
Procedure Graph Learning

- Besides the positive sequences from multimodal alignment, we also generate negative sequences by random selection, shuffling and swapping positive sequences for contrastive learning



Procedure Graph Construction

- After training the path generation model, we can reconstruct the procedure graph by sampling a large number of paths to estimate the adjacency matrix



Procedure Graph Results

Model	HT100M	Next Step Prediction				Partial Sequence Completion		
		Acc@1 ↑	Acc@3 ↑	Rec@3 ↑	F ₁ @3 ↑	Acc@1 ↑	Edit Dist. ↓	Normalized Edit Dist. ↓
TimeSformer+DS	✗	59.91	60.82	52.98	43.83	-	-	-
Random	✗	31.34	50.32	28.84	38.04	1.20	2.398	.6935
wikiHow Linear	✗	44.05	59.51	54.02	42.14	11.74	1.872	.6061
ReBART	✗	49.07	58.00	61.39	44.38	18.28	1.802	.4411
Direct NSP (Grounding)	✗	68.89	63.02	79.01	53.85	-	-	-
Direct PSC (Grounding)	✗	-	-	-	-	29.17	1.214	.4118
Ours (Grounding)	✗	75.59	67.50	83.17	58.29	20.12	1.639	.4296
Ours (Grounding)	✓	70.97	74.68	74.14	61.52	29.34	1.193	.4093
Ours (Grounding + PLC)	✗	75.49	71.89	72.51	58.48	26.70	1.228	.4267
Ours (Grounding + PLC)	✓	76.09	73.72	78.22	61.90	32.08	1.123	.3849

- ❑ Ours > wikiHow Linear: By utilizing graph representations, we can achieve much better prediction performance
- ❑ Pre-training on HowTo100M Videos helps
- ❑ The contrastive objective (PLC) helps with the partial sequence completion task

Event Graphs

Complex event: A collection of atomic events, their participants and relations.
Generally corresponds to a “news story”.

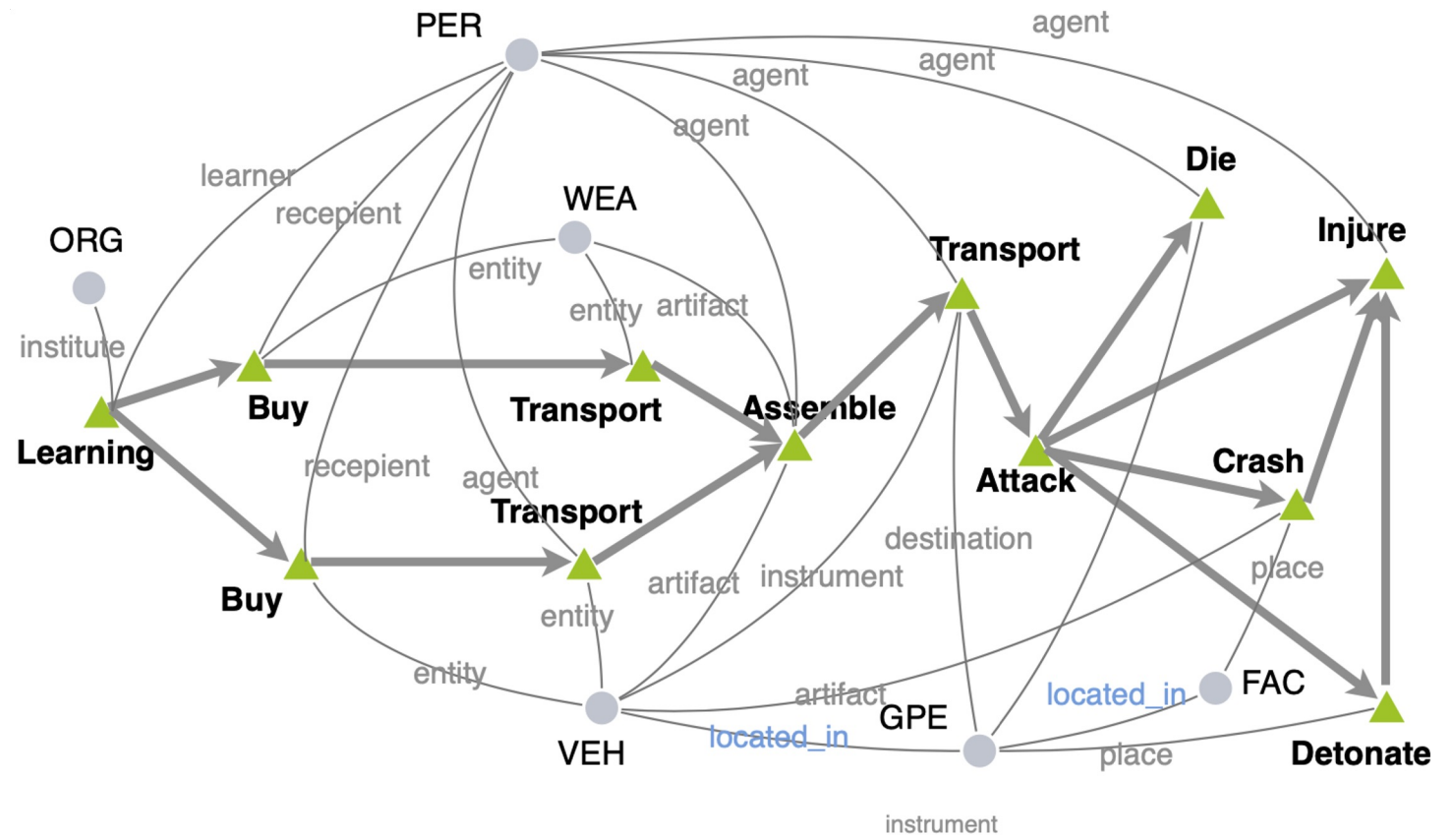
Nodes

- Events
- Entities

Coreferential events (entities) are merged into a single node.

Edges

- Event-entity argument edges
- Entity-entity relation edges
- Event-event temporal edges



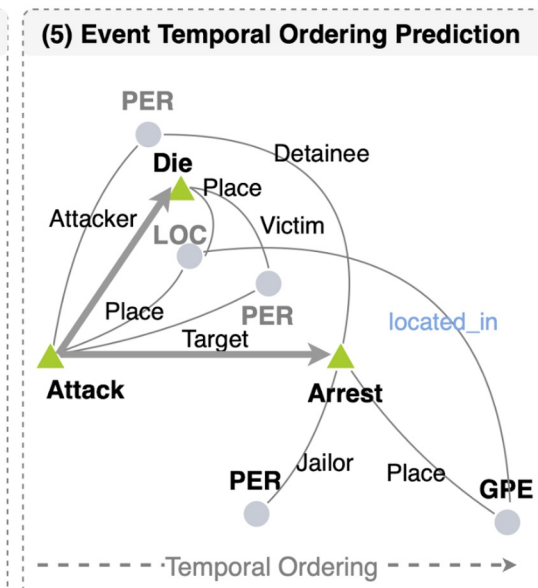
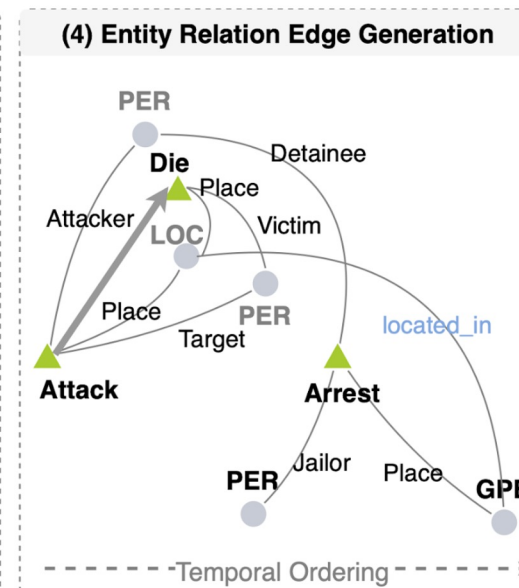
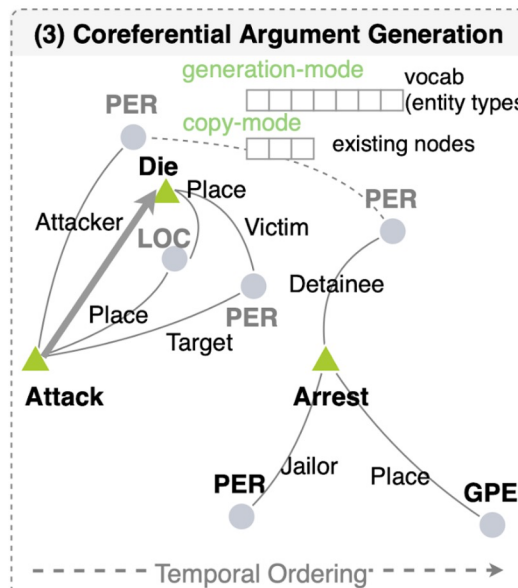
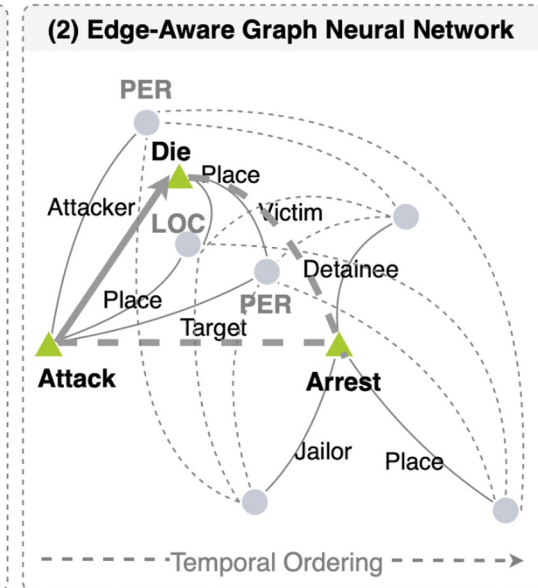
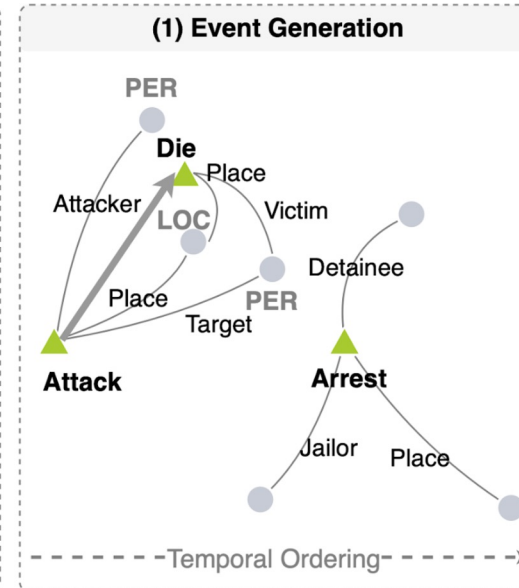
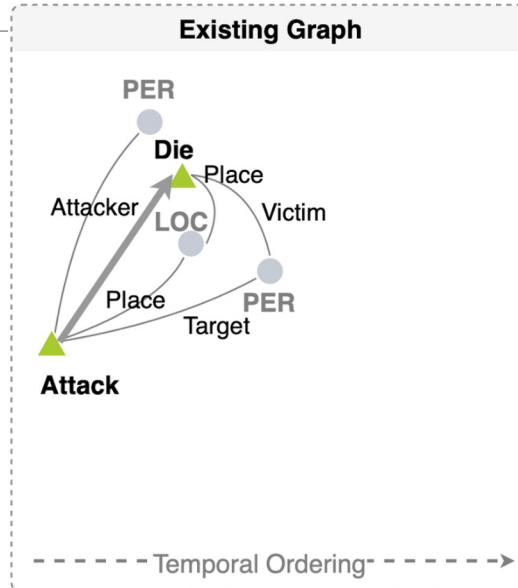
From Event Graph to Event Schema

- ❑ Original definition: “Structure for defining the appropriate sequence of events in a context” (Schank and Abelson 1977)
- ❑ A more modern interpretation: “Model for defining the probable sequence of events in a context” (Weber et al. 2018)

Generative Event Graph Model

$$p(G_i|G_{<i}) = p(e_i|G_{<i}) \prod_{a_j \in \mathcal{A}(e_i)} p(\langle e_i, a_j, v_j \rangle | e_i, a_j) \prod_{v_k \in G_{<i}} p(\langle v_j, r, v_k \rangle | v_j, v_k) \prod_{e_l \in G_{<i}} p(\langle e_i, e_l \rangle | e_i, e_l). \quad (1)$$

- Step 1. Event Node Generation
- Step 2. Message Passing
- Step 3. Argument Node Generation
- Step 4. Relation Edge Generation
- Step 5. Temporal Edge Generation



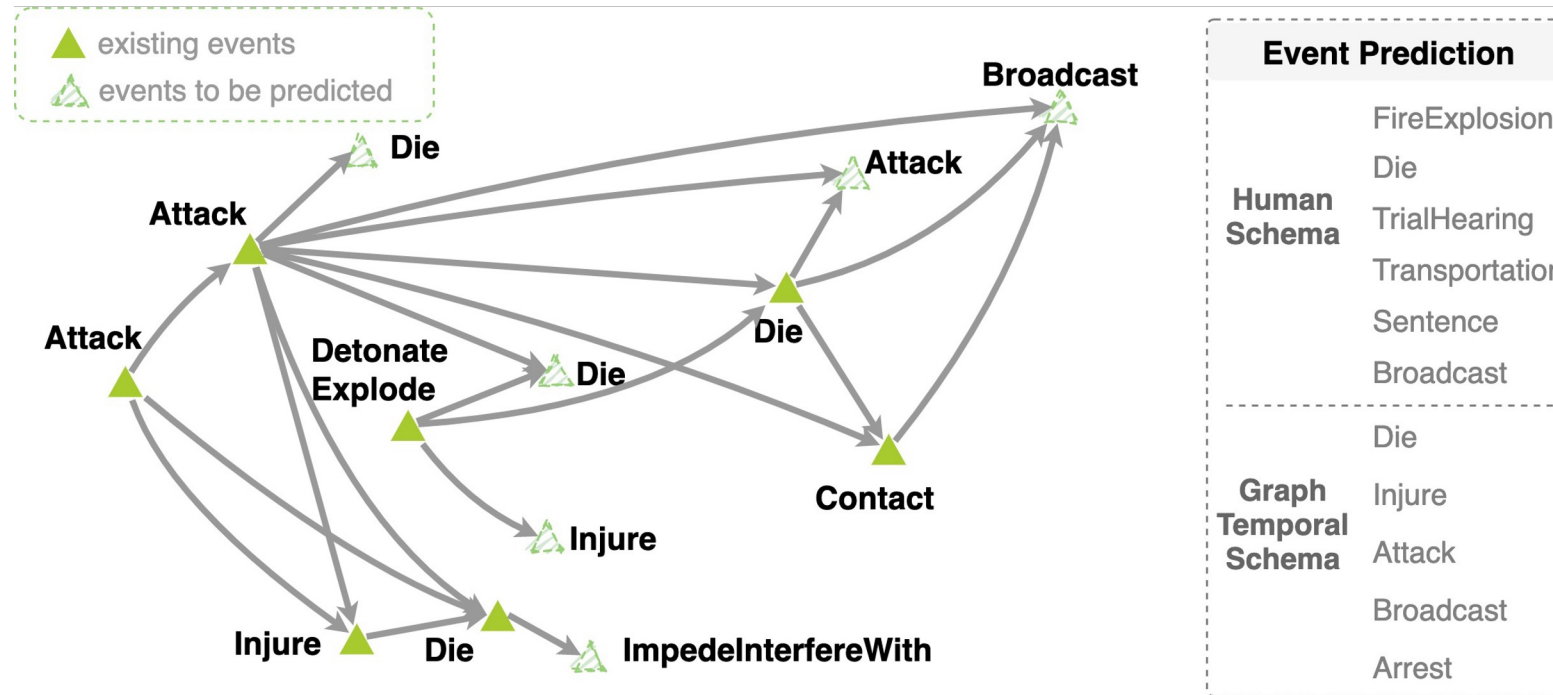
Intrinsic Evaluation

- ❑ **Schema Matching Evaluation:** We compare the generated schemas with the ground truth schemas based on the overlap between them.
- ❑ **Instance Graph Perplexity Evaluation:** We compute perplexity by predicting the instance graphs in the test set.

Dataset	Models	Event Match	Ordering Match	Sequence Match			Connection Match	Event Perplexity	Full Perplexity
				$l = 3$	$l = 5$	$l = 7$			
General	Event Language Model	0.392	0.578	0.397	0.239	0.132	-	-	-
	Sequential Pattern Mining	0.371	0.567	0.412	0.236	0.097	0.314	-	-
	Event Graph Model	0.451	0.612	0.479	0.298	0.181	0.391	1.104	3.798
	w/o Argument	0.423	0.601	0.469	0.271	0.173	-	1.982	-
IED	Event Language Model	0.701	0.815	0.679	0.417	0.301	-	-	-
	Sequential Pattern Mining	0.703	0.810	0.687	0.421	0.297	0.517	-	-
	Event Graph Model	0.812	0.881	0.718	0.432	0.321	0.567	0.585	2.307
	w/o Argument	0.803	0.872	0.712	0.422	0.309	-	0.956	-

Extrinsic Evaluation

- **Schema-guided Event Prediction:** The task aims to predict ending events of each graph.
 - Considering that there can be multiple ending events in one instance graph, we rank event type prediction scores and adopt MRR and HITS@1 as evaluation metrics.

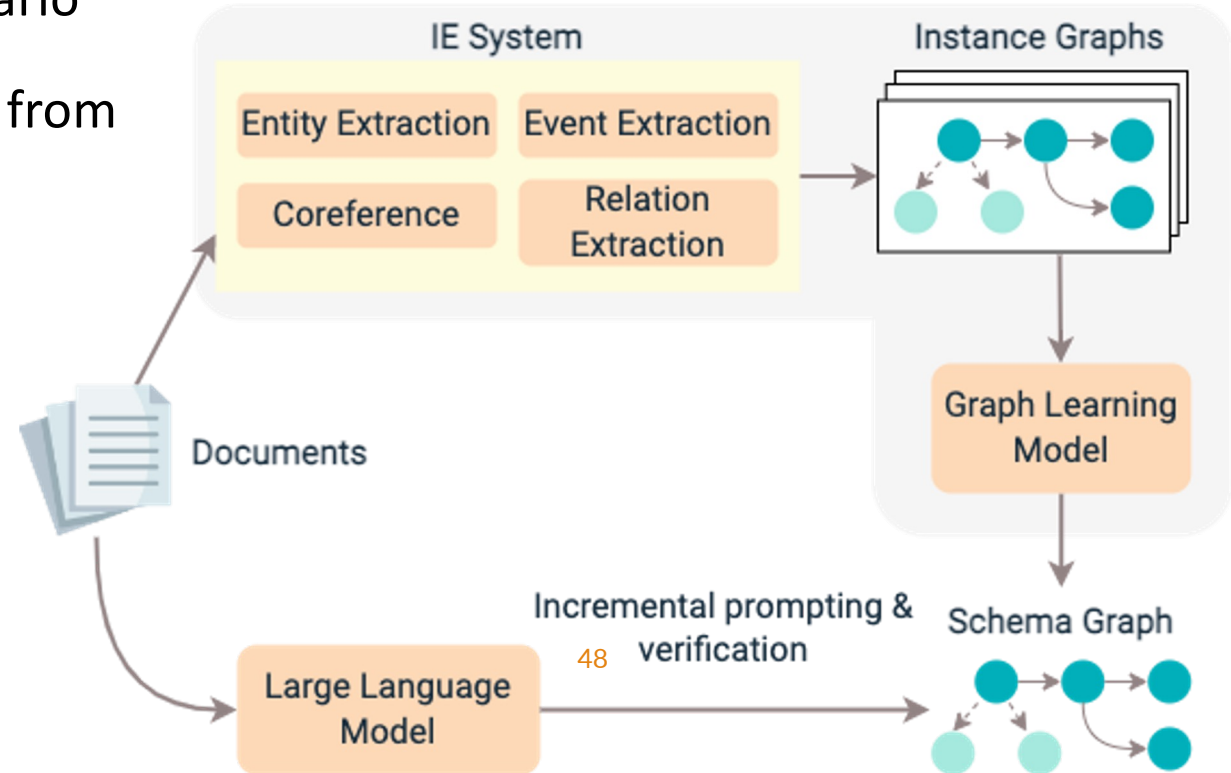


Dataset	Models	MRR	HITS@1
General	Human Schema	0.173	0.205
	Event Graph Model	0.401	0.520

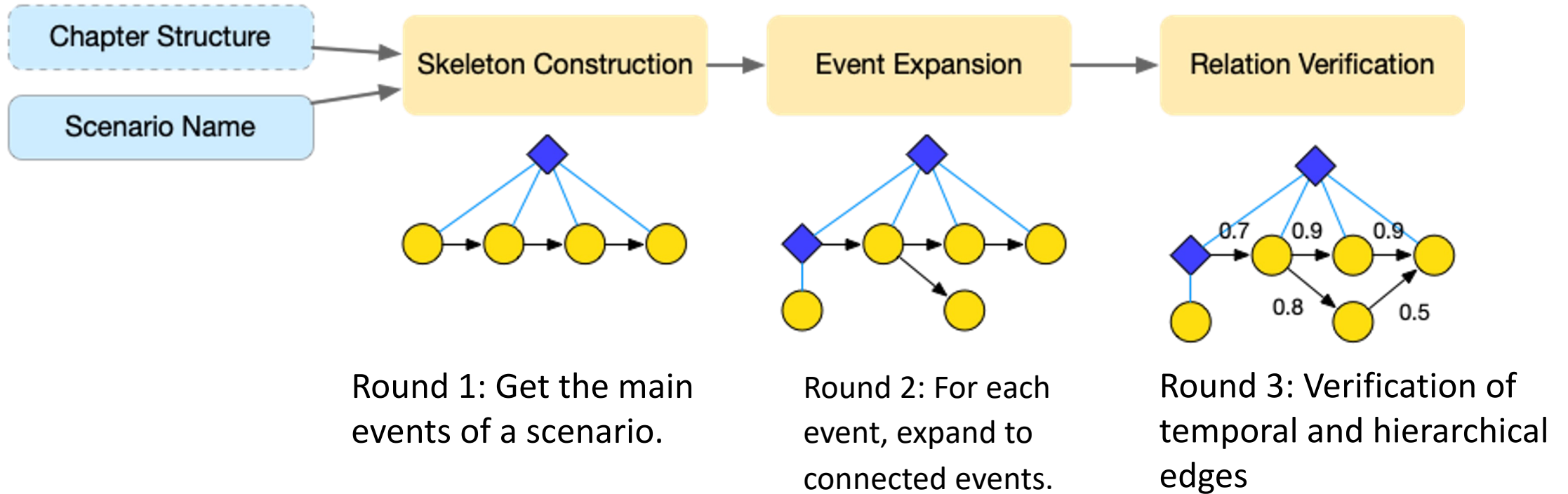
Dataset	Models	MRR	HITS@1
IED	Human Schema	0.072	0.222
	Event Graph Model	0.223	0.691

Schemas as Event-Related Commonsense Knowledge

- Event schemas can be viewed as structured *commonsense knowledge* about a given scenario
- We can probe such commonsense knowledge from an LLM
 - **Open-Domain:** our model can induce schemas for any scenario given the scenario name.
 - **Extensible:** our paradigm can support new event-event relations by adding new prompt templates.
 - **Interpretable:** by representing events with sentences, human assessors consider our schemas to be more readable than prior approaches.



IncSchema Framework



Retrieval-Augmented Prompting

- ❑ When humans curate schemas, they often refer to related news articles or Wikipedia
- ❑ Whenever our prompt is related to an event, we simulate this process by using the GPT3-generated event description to retrieve related passages to serve as extra context to the language model
 - ❑ To encourage the model to output a general answer, we retrieve 3 passages per prompt using a pretrained TCT-ColBERT model.
 - ❑ Retrieving multiple passages (ideally about different instances) is important for guiding the model to produce a **generalized answer**.

Retrieval-Augmented Prompt

```
Based on the following passages
{retrieved passages},
{prompt}
```

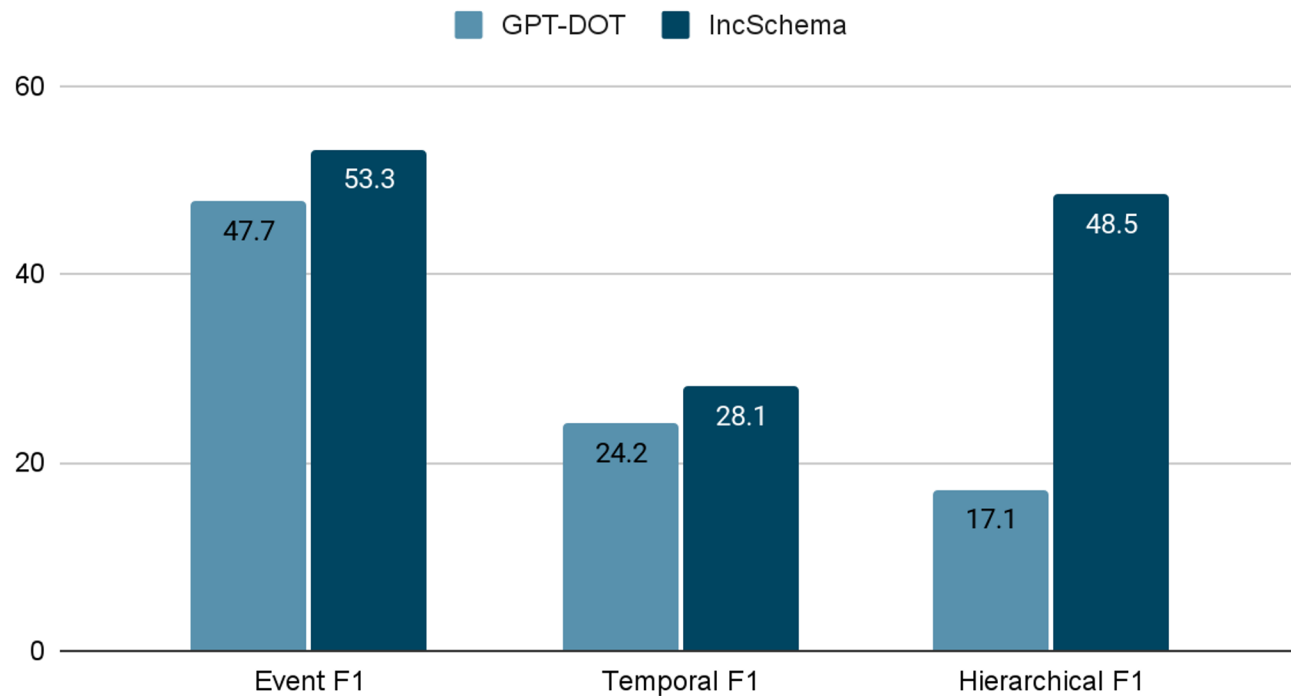
Question Decomposition for Relations

- Instead of directly asking for “Does event A happen before event B?”, we ask 3 questions about start time, end time and duration.
- This allows us to avoid conflicting hierarchical and temporal relations since the hierarchical relation can be defined as spatial-temporal containment.

Relation	Allen's base relations	e_1 starts before e_2 ?	e_1 ends before e_2 ?	Duration
$e_1 \prec e_2$	e_1 precedes e_2 , e_1 meets e_2	Yes	Yes	-
$e_1 \succ e_2$	e_1 is preceded by e_2 , e_1 is met by e_2	No	No	-
$e_1 \subset e_2$	e_1 starts e_2 , e_1 during e_2 , e_1 finishes e_2	No	Yes	$d(e_1) < d(e_2)$
$e_1 \supset e_2$	e_1 is started by e_2 , e_1 contains e_2 , e_1 is finished by e_2	Yes	No	$d(e_1) > d(e_2)$
$e_1 \parallel e_2$	e_1 overlaps with e_2 , e_1 is equal to e_2	Yes	No	$d(e_1) \leq d(e_2)$
$e_1 \parallel e_2$	e_1 is overlapped by e_2	No	Yes	$d(e_1) > d(e_2)$

Main Result: Hierarchical Schema Quality

Schema Quality Evaluation



The baseline GPT-DOT, directly asks GPT3 to output a linearized graph format of the final schema given some in-context examples.

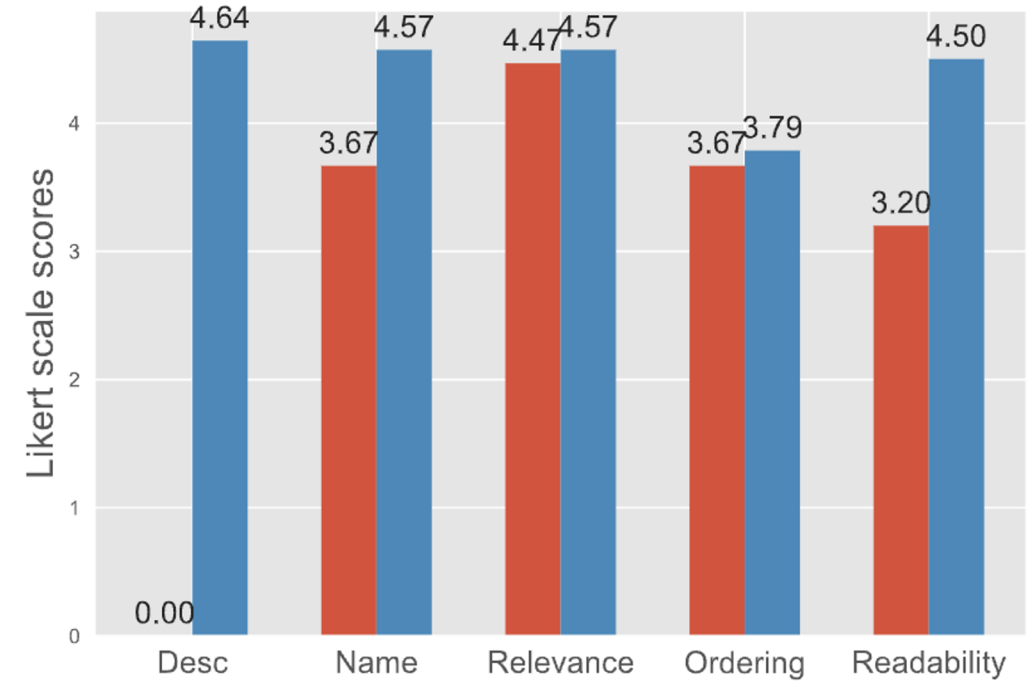
Compared to our model, GPT-DOT generates much fewer events (10.11 events for GPT-DOT VS 52.6 events on ODIN) which leads to high precision but low recall.

GPT-DOT struggles with hierarchical relations, especially when hierarchical relations co-exist with temporal relations.

Interpretability Evaluation Results


Model	Coverage↑	Len(words)↑	Time(mins)↓
Double-GAE	79.8	9.62	0.998
INCPROMPT	89.7	15.53	1.137

- Human assessors are able to compose a longer story with better event coverage using our schema while taking roughly the same amount of time.
- Human assessors rate our event descriptions and event names to be very helpful (>4.5 score) and our schemas are easier to understand compared to the baseline.



Double-GAE's scores are shown in red and our model's scores are shown in blue.

Outline

- ❑ Syntactic and Semantic Parse Graphs
- ❑ Information Extraction Graphs
- ❑ Procedure and Schema Graphs
- ❑ Belief and Reasoning Graphs 

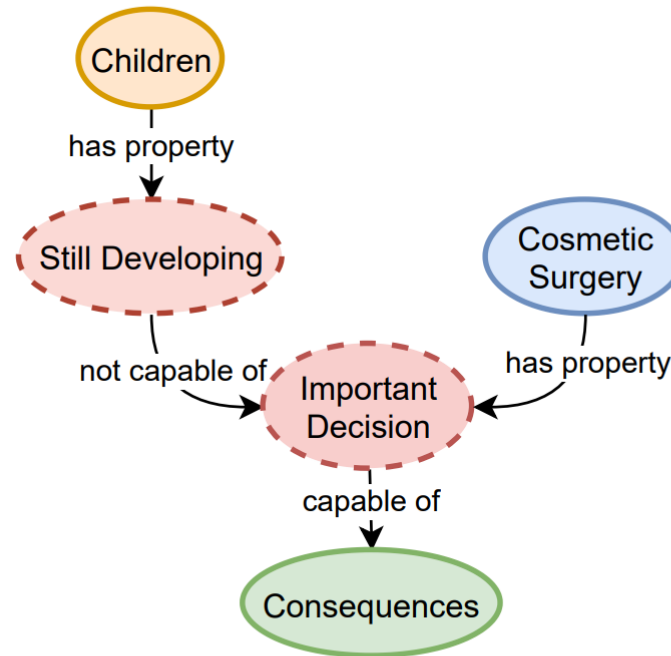
Explanation Graphs

- Given a belief and an argument, predict the stance of the belief and the reasoning process
- The reasoning process can be represented as a graph of concepts and their relations

Belief: Children should be able to consent to cosmetic surgery.

Argument: Children do not have the mental capacity to understand the consequences of medical decisions.

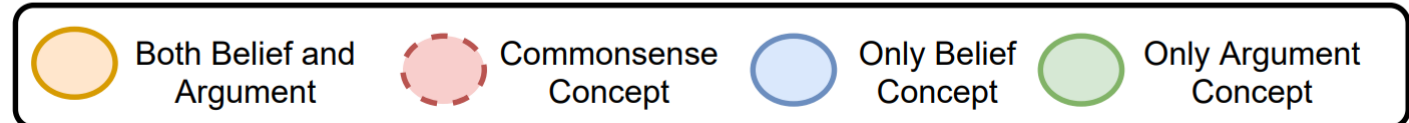
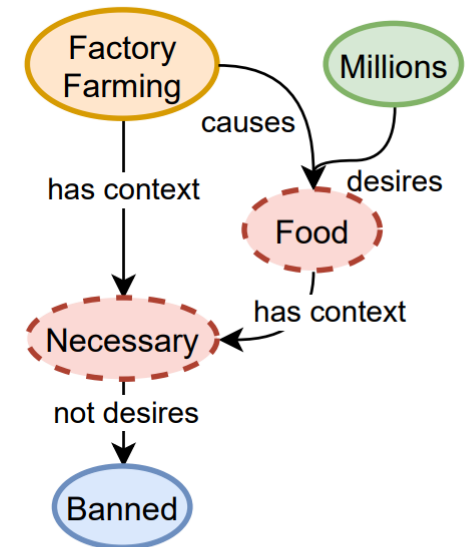
Stance: Counter



Belief: Factory farming should not be banned.

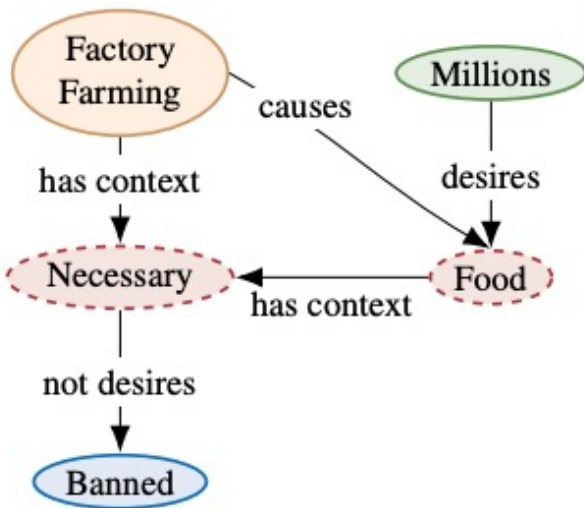
Argument: Factory farming feeds millions.

Stance: Support



Generating ExplaGraphs with Code LLMs

- ❑ The task of generating an ExplaGraph can be converted into generating a piece of code with a list of `add_edge` function calls.
- ❑ By using this formulation, few-shot CoCoGen outperforms fine-tuned T5 across all metrics.



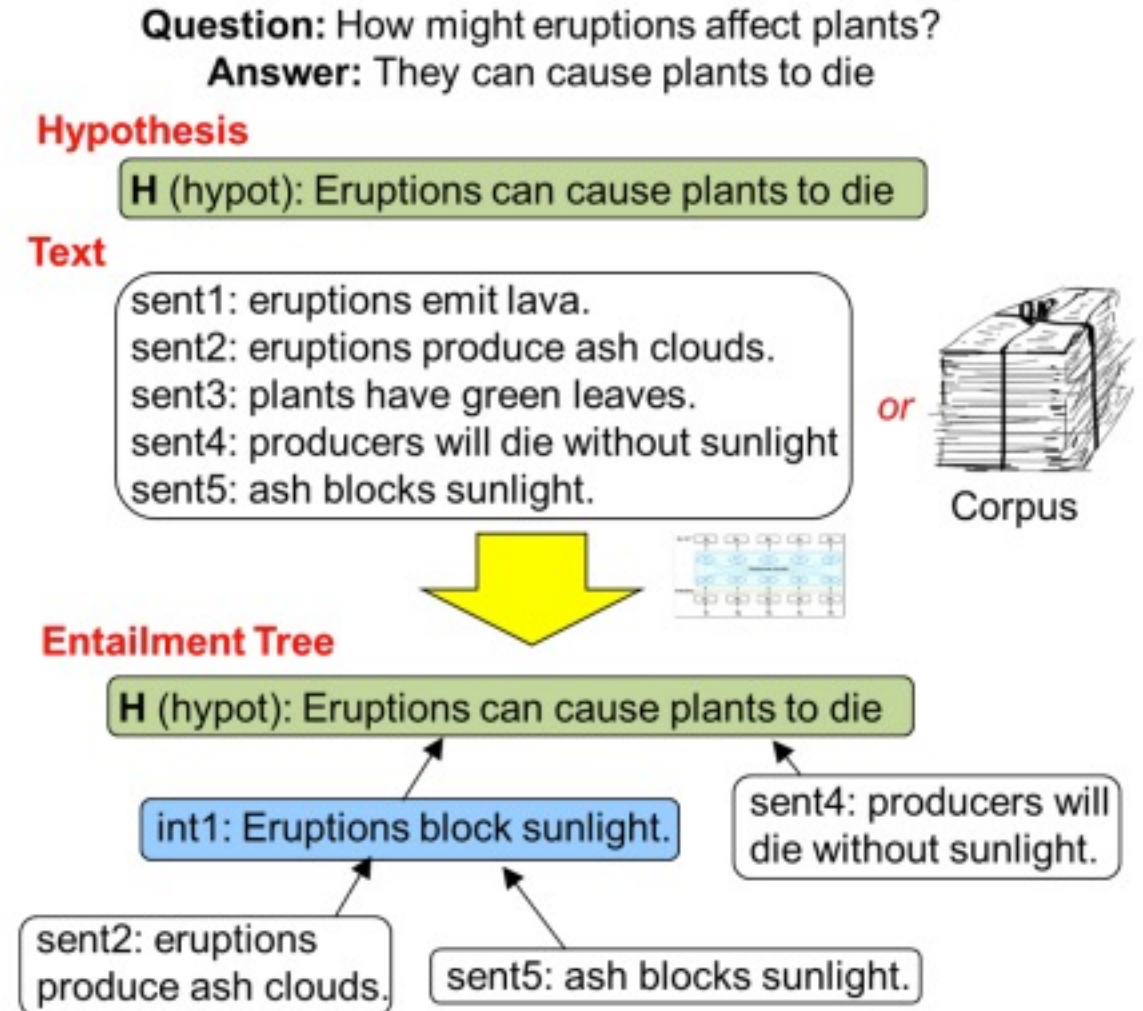
```
class ExplanationDAG:
    def __init__(self):
        belief = "factory farming should not be banned."
        argument = "Factory farming feeds millions."
        stance = "support"

    # Edges
    begin = ["factory farming", "millions"]
    add_edge("factory farming", "causes", "food")
    add_edge("factory farming", "has context", "necessary")
    add_edge("food", "has context", "necessary")
    add_edge("necessary", "not desires", "banned")
    add_edge("millions", "desires", "food")
```

		StCA (↑)	SeCA (↑)	G-BS (↑)
fine-tuned	T5 (150)	12.56	6.03	9.54
	T5 (1500)	38.19	21.86	29.37
	T5 (2500)	43.22	29.65	33.71
few-shot	CURIE (30)	5.03	1.26	3.95
	DAVINCI (30)	23.62	10.80	18.46
	CoCoGen (30)	45.20	23.74	34.68

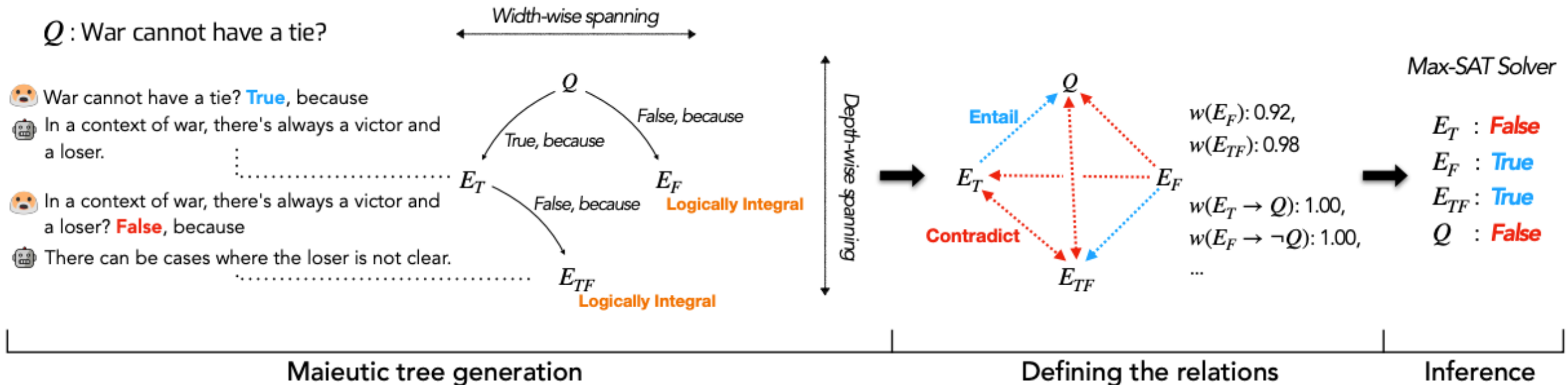
Entailment Trees

- Entailment tree: a tree of multi-premise entailment steps from facts that are known, through intermediate conclusions, to the hypothesis of interest



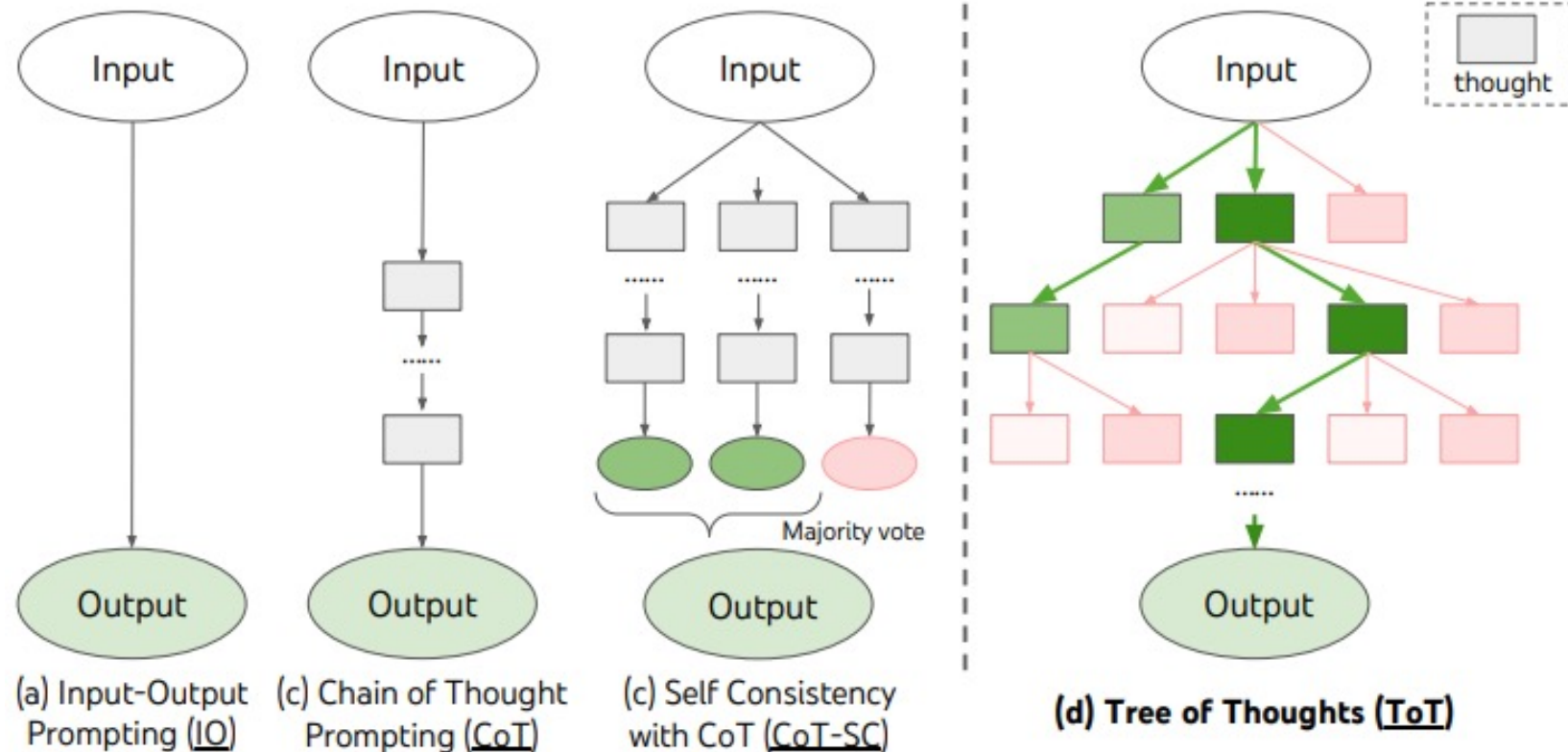
Improving LLM Reasoning with Recursive Prompting

- Induce a tree of explanations recursively by prompting the LM with “X is true, because...”
- Compute the belief (X is true) for each statement and the consistency (can X and Y both be true) between every two statements
- Solve the truth values of the statements using a MAX-SAT solver



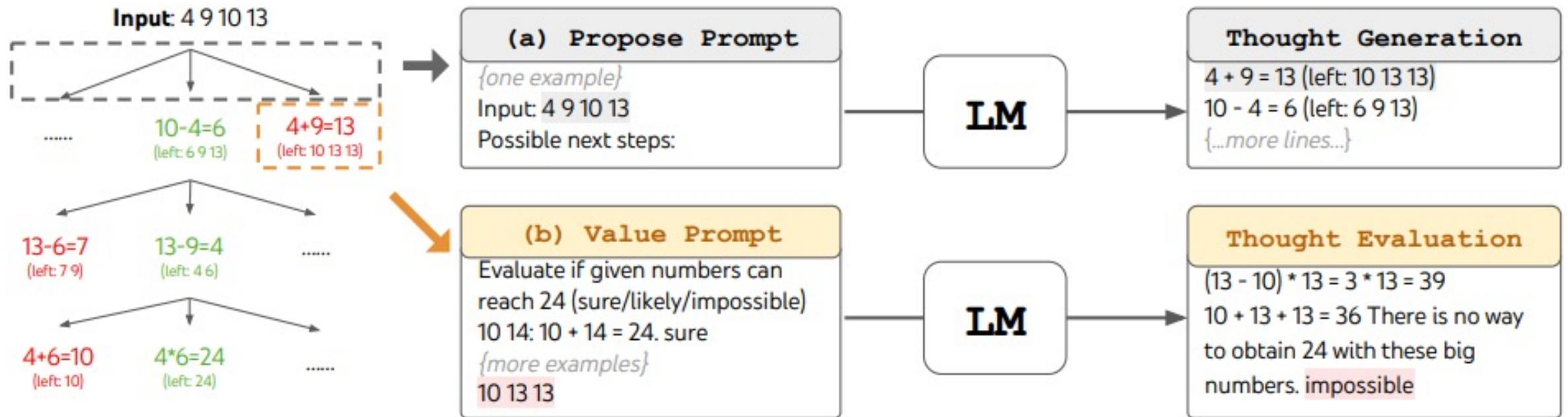
From Chain-of-Thought to Tree-of-Thought

- Compared to the commonly-used CoT, ToT explores multiple possibilities, evaluates the intermediate output and picks the best move forward. If all possibilities are bad, the model backtracks to the last state.



Tree-of-Thought Example

- ❑ Task: Game of 24 is a mathematical reasoning challenge, where the goal is to use 4 numbers and basic arithmetic operations (+-*/) to obtain 24.
- ❑ The LLM is used for generating the possible next step and also evaluating the current solution.



References

- ❑ Jiawei Zhou, Tahira Nasee, Ramón Fernandez Astudillo, Young-Suk Lee, Radu Florian, Salim Roukos. Structure-aware Fine-tuning of Sequence-to-sequence Transformers for Transition-based AMR Parsing. EMNLP 2021.
- ❑ Leonardo F. R. Ribeiro , Mengwen Liu , Iryna Gurevych , Markus Dreyer , Mohit Bansal. FACTGRAPH: Evaluating Factuality in Summarization with Semantic Graph Representations. NAACL 2022.
- ❑ Sha Li, Mahdi Namazifar, Di Jin, Mohit Bansal, Heng Ji, Yang Liu, and Dilek Hakkani-Tur. 2022. Enhancing Knowledge Selection for Grounded Dialogues via Document Semantic Graphs. NAACL 2022.

References

- ❑ Jiaxin Huang, Yu Meng, and Jiawei Han. “Few-Shot Fine-Grained Entity Typing with Automatic Label Interpretation and Instance Generation” KDD 22.
- ❑ Sizhe Zhou, Yu Meng, Bowen Jin, & Jiawei Han. (2024). Grasping the Essentials: Tailoring Large Language Models for Zero-Shot Relation Extraction. arXiv preprint arXiv:2402.11142.
- ❑ Xingyao Wang, Sha Li, and Heng Ji. 2023. [Code4Struct: Code Generation for Few-Shot Event Structure Prediction](#). ACL 2023.
- ❑ Xinya Du, Zixuan Zhang, Sha Li, Pengfei Yu, Hongwei Wang, Tuan Lai, Xudong Lin, Ziqi Wang, Iris Liu, Ben Zhou, Haoyang Wen, Manling Li, Darryl Hannan, Jie Lei, Hyounghun Kim, Rotem Dror, Haoyu Wang, Michael Regan, Qi Zeng, et al.. 2022. [RESIN-11: Schema-guided Event Prediction for 11 Newsworthy Scenarios](#). NAACL 2022 Demo.
- ❑ Coreference Resolution through a seq2seq Transition-Based System. Bernd Bohnet, Chris Alberti, Michael Collins. TACL 2023.

References

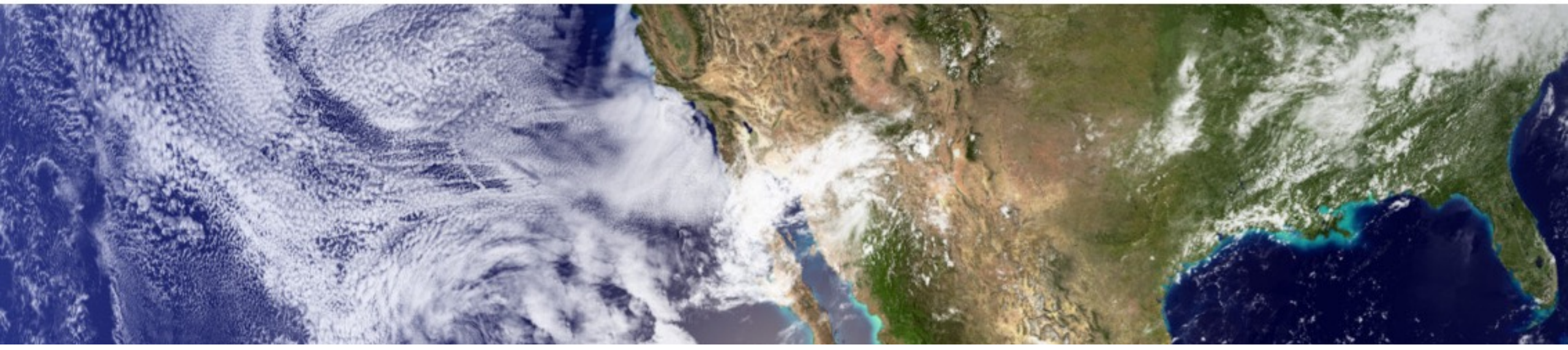
- ❑ Non-Sequential Graph Script Induction via Multimedia Grounding. Yu Zhou, Sha Li, Manling Li, Xudong Lin, Shih-Fu Chang, Mohit Bansal, Heng Ji. ACL 2023.
- ❑ Manling Li, Sha Li, Zhenhailong Wang, Lifu Huang, Kyunghyun Cho, Heng Ji, Jiawei Han, and Clare Voss. 2021. The Future is not One-dimensional: Complex Event Schema Induction by Graph Modeling for Event Prediction. EMNLP 2021.
- ❑ Sha Li, Ruining Zhao, Manling Li, Heng Ji, Chris Callison-Burch, and Jiawei Han. 2023. [Open-Domain Hierarchical Event Schema Induction by Incremental Prompting and Verification](#). ACL 2023.

References

- ❑ Swarnadeep Saha, Prateek Yadav, Lisa Bauer, and Mohit Bansal. 2021. [ExplaGraphs: An Explanation Graph Generation Task for Structured Commonsense Reasoning](#) EMNLP 2021
- ❑ Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. 2021. [Explaining Answers with Entailment Trees](#). EMNLP 2021
- ❑ Aman Madaan, Shuyan Zhou, Uri Alon, Yiming Yang, and Graham Neubig. 2022. [Language Models of Code are Few-Shot Commonsense Learners](#). EMNLP 2022.
- ❑ Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. 2022. [Maieutic Prompting: Logically Consistent Reasoning with Recursive Explanations](#).EMNLP 2022.
- ❑ Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, Karthik Narasimhan. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. Neurips 2023.



Q&A

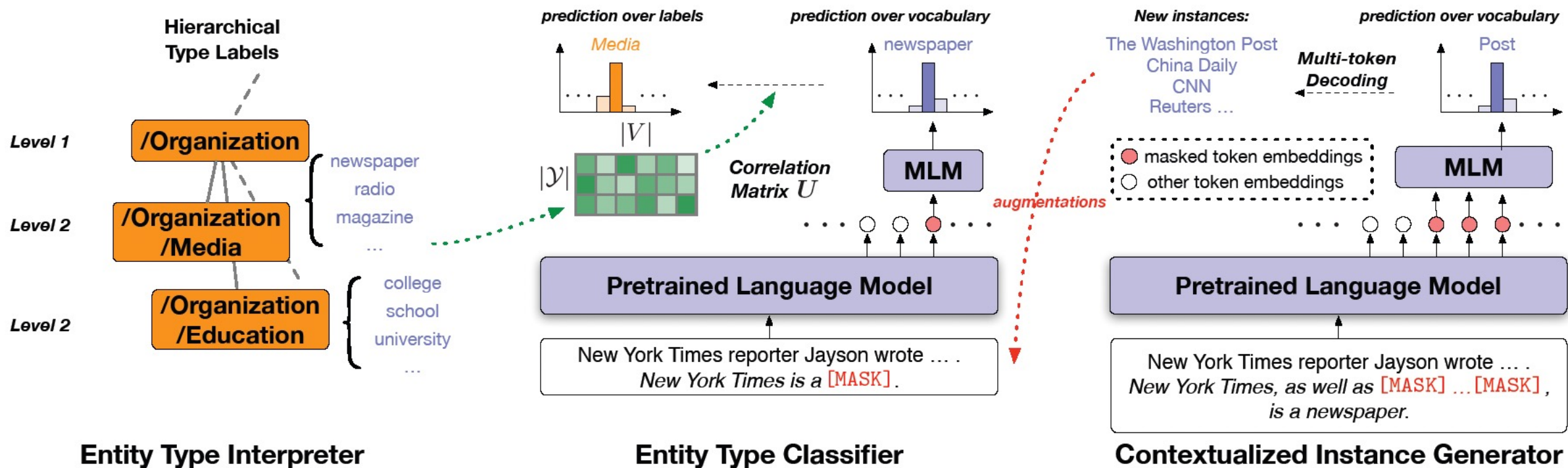


Backup Slides

Named Entity Recognition (NER)

- ❑ A **named entity** typically refers to a sequence of words that correspond to a specific entity in the real world (i.e., an entity with a *name*) (e.g., “*Bill Clinton*”)
- ❑ **Named-entity recognition (NER)** seeks to **locate** and **classify named entities** in text into **pre-defined categories**
 - ❑ Given a sentence, NER is to first *segment which words are part of entities*, and then *classify each entity by type* (person, organization, location, and so on)
 - ❑ Example
 - ❑ Input: Jim bought 300 shares of Acme Corp. in 2006
 - ❑ Output: [Jim]_{Person} bought 300 shares of [Acme Corp.]_{Organization} in [2006]_{Time}

ALIGNIE (Automatic Label Interpretation and Generating New Instance for Entity typing)



(Left): With a given type label hierarchy, an entity type interpretation module relates all the words in the vocabulary with the label hierarchy by a correlation matrix.

(Middle): An entity typing classifier maps the word probability at the [MASK] position to type probability using the correlation matrix.

(Right): A type-based contextualized instance generator uses an entity mention and its predicted type to construct a template for new instance generation to augment the training set.

PLM-based Instance Generator

- E.g., a *newspaper* entity “New York Times” → more newspaper names

Generation Template :

[Context]. **New York Times**, as well as [MASK] [MASK] [MASK], is a *newspaper*.

Entity Mention

ranges from 1 to the length of original entity mention

Predicted by Entity Type Classifier

Multi-Token Instance Generation

- We randomly choose one [MASK] token at each step, and sample from its output token probability to fill in a word.

E.g.
New York Times, as well as the₁ [MASK] [MASK] is a newspaper.
New York Times, as well as the₁ Washington₂ [MASK] is a newspaper.
New York Times, as well as the₁ Washington₂ Post₃ is a newspaper.

The next blank to be filled in is randomly selected, therefore the order is not always from left to right.

$$\text{Score}(\tilde{m}) = \sum_{i=1}^{|\tilde{m}|} \log(s_i)$$

The conditional probability
at each step

Generated New instances based on predicted types of example entities

□ Multi-token instances

Generation from multi-token entities		
Context & entity mention	MLM predicted type	Generated new instances
The album also included the song “Vivir Lo Nuestro,” a duet with Marc Anthony .	singer	Beyonce, Jennifer Lopez, Rihanna, Taylor Swift, Lady Gaga, Michael Jackson, ...
The film was released on August 9, 1925, by Universal Pictures .	company	Warner Brothers, Paramount Pictures , Columbia Pictures, Lucasfilm, Hollywood Pictures, ...
Everland hosted 7.5 million guests in 2006, ranking it fourth in Asia behind the two Tokyo Disney Resort parks and Universal Studios Japan, while Lotte World attracted 5.5 million guests to land in fifth place.	park	Lotte World, Universal Studios Japan, Shanghai Disney World , Orlando Universal Studios, ...
The site of Drake’s landing as officially recognised by the U.S. Department of the Interior and other agencies is Drake’s Cove.	government agency	the Department of Homeland Security, the Bureau of Land Management, the Federal Bureau of Investigation, the United States Forest Service, the National Institutes of Health, ...
Pikmin also make a cameo during the process of transferring downloadable content from a Nintendo DSi to a 3DS, with various types of Pikmin carrying the data over.	handheld	3DS, 2DS, Wii U, Nintendo Switch, the PSP, PlayStation Vita, ...

Main Results

Method	OntoNotes			BBN			Few-NERD		
	(Acc.)	(Micro-F1)	(Macro-F1)	(Acc.)	(Micro-F1)	(Macro-F1)	(Acc.)	(Micro-F1)	(Macro-F1)
5-Shot Setting									
Fine-tuning	28.60	50.70	51.60	51.03	60.03	58.22	36.09	48.56	48.56
Prompt-based MLM	32.62	60.97	61.82	67.00	75.23	73.55	44.69	59.24	59.24
PLET	48.57	70.63	75.43	71.23	79.22	78.93	56.94	68.81	68.81
ALIGNIE (- hierarchical reg.)	52.74	77.55	79.72	72.15	80.35	80.40	59.01	70.91	70.91
ALIGNIE (- new instances)	51.10	72.91	76.88	73.50	81.62	81.31	57.41	69.47	69.47
ALIGNIE	53.37	77.21	80.68	75.44	82.20	82.30	59.72	71.90	71.90
Fully Supervised Setting									
Fine-tuning	56.70	75.21	78.86	78.06	82.39	82.60	79.75	85.74	85.74
Prompt-based MLM	55.18	74.57	77.47	77.10	81.77	82.05	77.38	85.22	85.22

- Prompt-based results have higher performance than vanilla fine-tuning in few-shot settings. In fully supervised settings, however, fine-tuning performs a little better than prompt-based MLM.
- ALIGNIE can even outperform fully supervised setting on OntoNotes and BBN, but cannot on Few-NERD. This is because the training set of OntoNotes and BBN are automatically inferred from external knowledge bases, and can contain much noise.

Schema Induction Dataset

- IED Scenario-aware Instance Graph Construction
 - **Scenario-aware data collection based on Wikipedia:** For each scenario, we find the associated Wikipedia category, and we collect the major events under it. For each major event, we crawl the reference news articles as input news articles.
 - **Instance graph construction:** We order events using temporal relations, and ignore events that are not connected to other events.

Dataset	Split	# Document	# Event	# Argument	# Relation
General	Train	383	60,40	10,720	6,858
	Dev	72	1,044	1,762	1,112
	Test	83	1,211	2,112	1,363
IED	Train	5,247	4,1672	136,894	122,846
	Dev	575	4,661	15,404	13,320
	Test	577	5,089	16,721	14,054