The Coreference under Transformation Labeling Dataset: Entity Tracking in Procedural Texts Using Event Models

(Findings of ACL 2023)

Kyeongmin Rim(*), Jingxuan Tu, Bingyang Ye,
Marc Verhagen, Eben Holderness
and James Pustejovsky

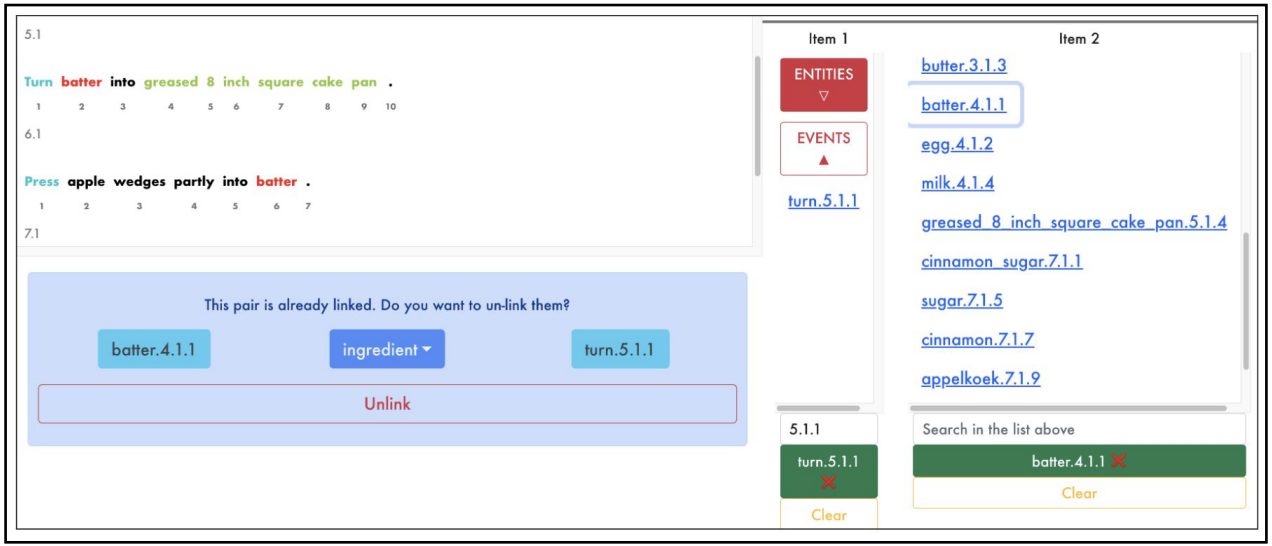
December 6, 2023
CRAC 2023

Background 1: R2VQ

- Tu et al. 2022. *SemEval-2022 Task 9: R2VQ – Competence-based Multimodal Question Answering*
- We tried two methods to capture the “roles” of the entities.
  a. Semantic Role:
     - Applied the SRL parser to tag the predicate-argument structure within each recipe sentence
     - Used VerbAtlas as the inventory of semantic roles
     - Output from SRL parser are manually validated by student annotators
  b. Cooking Role:
     - Simplified span-based argument relation annotation

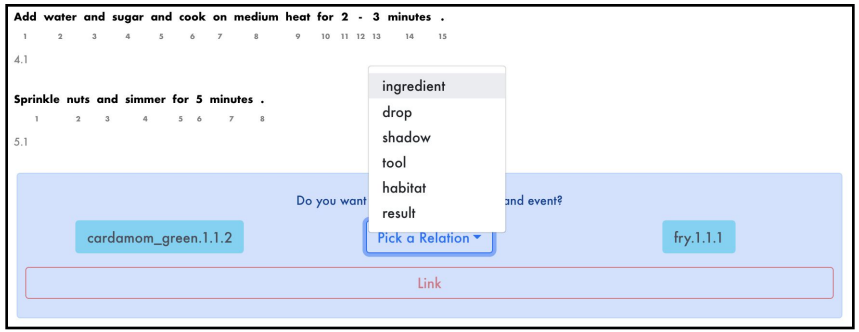
Deep Event-Entity Palette (DEEP)

- Annotation tool developed for cooking role annotation



Argument relations in CRL/DEEP

- 3 extent spans, 2 hidden spans, 1 “result” span



R2VQ annotation

- 8 university student (undergrad, master) annotators
- 3 month including DEEP development time
- 950 recipe documents, single annotation
- 50 recipe documents, dually annotated
- Inter-Annotator Agreement:

Entity	# of annotation from ANN1/ANN2	Cohen's κ
EXPLICIT	707 / 693	0.900
HIDDEN	1006 / 979	0.608
ALL	1713 / 1672	0.730
Coreference		CoNLL-F1
TOOL	178 / 172	58.08
HABITAT	319 / 310	56.35
INGREDIENT	783 / 785	58.27
ALL (EXPLICIT)	955 / 970	85.71
ALL	1280 / 1267	57.46

“Result” argument in CRL/DEEP

- From R2VQ guidelines;
  - The `result` relation is meant to identify relationships between the `event` and another entity **in the same sentence** (**`result` link cannot be a `hidden` relation,** see below for description of `hidden` arguments). In the sentence, *“Shape with hands into a ball”* the `ball` is the `result` of the `shape` action, which took place on a dropped plum (from an earlier sentence).
- This resulted in the majority of events missing extent result spans
- Note that SRL annotation only handles extent text spans

Coreference under Transformation Labeling (CUTL)

- Coreference Under Transformation Labeling as a multi-class labeling task
  - CuI: Traditional “full-identity” corefereces
    - Identity: strict coreference
    - Meronymy: two entities when one end is referring to an inseparable part of the other
    - Metonymy: btw ingredient and location when the location is used as a container.
    - Change-of-Location
  - CuT: Near-identity based on POEM
    - Transformation: one-to-one transformation (cut, cook, …)
    - Aggregation: many-to-one transformation (mix, add, …)
    - Separation: one-to-many transformation (split, remove, …)
- Annotation
  - 1 month, 7 annotators, double annotation, 4 rounds of adjudication
  - pairwise F1 as our primary IAA metric, mean F1 = 86.9
  - Developed in-house annotation environment, CULTER

CUTL Dataset

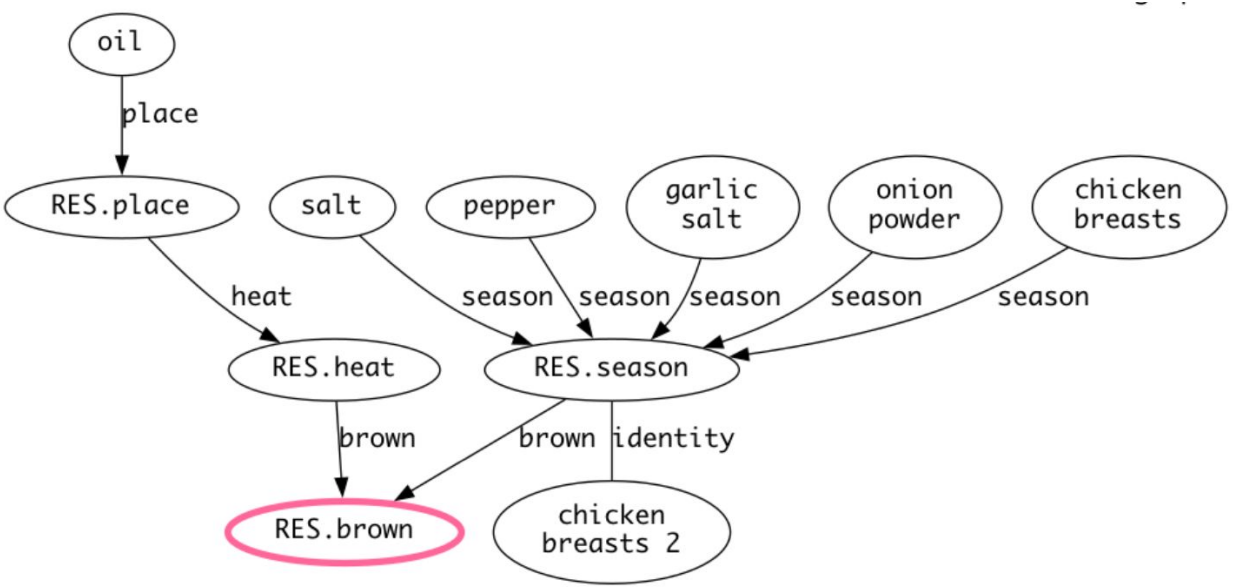
- Dataset statistics
  - Sampled and filtered from R2VQ dataset (r2vq.org)
  - 100 recipes, ~10 events per recipe in average
  - R2VQ annotates two relations (participant-of and result-of) between entity ⇔ event
  - POEM enabled a broader coverage of the hidden entities.

Avg. # of entities per recipe	Explicit	Hidden
EVENT	10.6	N/A
INGREDIENT (input)	12.0	**9.4**
INGREDIENT (output)	1.0	**10.4**
R2VQ		
INGREDIENT (participant)	11.5	5.7
INGREDIENT (result)	1.1	2.5

Background 2: Dense Paraphrasing

- Tu et al. 2023. *Dense Paraphrasing for Textual Enrichment*
- Prefix Paraphrase (rule-based)
  a. Categorize predicates into transformation, location-change, and etc.
  b. For transformation events, append adjectival form to the entity
     (e.g. “boiled” + water, “drained” + “soaked” peas)
  c. Insert .RES entities back into sentences using common-pattern templating
- Graph Paraphrase (LLM-based)
  a. Extract the subgraph rooted in the hidden entity mention node
  b. Linearize the subgraph into a string literal
  c. Prompt GPT3 to turn the subgraph into a natural language noun phrase

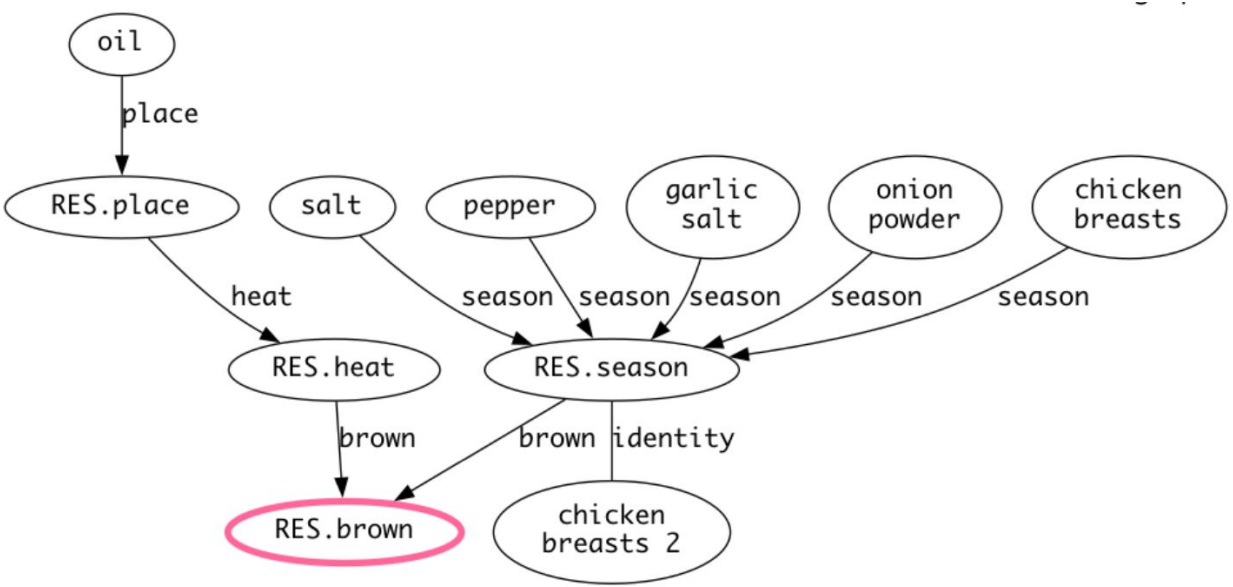
DP example



Implicit mention provenance:

Brown the chicken breasts on each side. → `RES.Brown`

DP example (rule-based)



"browned seasoned chicken breasts and heated oil'

DP example

Asking GPT

```
The task is to generate short and accurate paraphrase of the given noun phrases.
The input noun phrase describes the event state change of the food ingredients
through processing in a recipe, and the output paraphrase should summarize the
combination or state change of the ingredients.

input: stirred butter mixture and flour and cocoa and baking soda and salt
output: dough

[7 more exemplars]


input: squeezed horseradish
output:
```

```
The task is to generate short and accurate paraphrase of the given logical expression.
The input logical expression describes the cooking events and state change of the food ingredients
through processing in a recipe, and the output paraphrase should summarize the combination or state
change of the ingredients.

event types in the logical expression:
TRANSFORMATION: event that transforms the state, shape and etc. of an ingredient
AGGREGATION: event that combines multiple ingredients together
SEPARATION: event that separate an ingredient, or remove part of the ingredient
LOC: move the ingredient to another location

input: ['reserve-TRANSFORMATION', [['combine-AGGREGATION', ['onion', 'chilies', 'cilantro', 'salt']]]]
output: reserved onion mixture

[7 more exemplars]

input: ['squeeze-TRANSFORMATION', ['horseradish']]
output:
```

Experiment: Automatic Coreference Resolution

- Trained multi-class coreference resolution systems
  - As a mention-level pairwise label classification task (GLoVe + ELMo + windowed CNN)
    - Similar to Fang et al. 2022 and Lee et al. 2018
  - Two-stage classification: 1) mention pair assignment, 2) label assignment
- GRN/phantom nodes are paraphrased into natural language text
  - Using the “Dense Paraphrasing”
  - Different paraphrasing methods are compared for implicit output entities
- Label sets
  - Coarse setting: CuI, CuT binary
  - Fine setting: 3 CuI sub-categories, 3 CuT sub-categories.
    (“metonymy” is excluded due to very small number of instances)

Results

		CuI			CuT		
labels	Preprocessing	P	R	F1	P	R	F1
Coarse (binary)	PrefixP paraphrase	82.46 (±5.31)	9.31 (±6.81)	9.31 (±6.81)	86.05 (±1.92)	46.41 (±5.06)	60.29 (±4.01)
	SubgraphGPT paraphrase	85.68 (±9.81)	11.02 (±3.50)	**19.07 (±5.67)**	88.12 (±3.18)	47.25 (±2.84)	**61.09 (±2.88)**
Fine (all subcats)	PrefixP paraphrase	87.28 (±6.36)	11.60 (±0.83)	20.02 (±1.38)	85.19 (±1.10)	41.15 (±1.59)	54.89 (±1.30)
	SubgraphGPT paraphrase	89.57 (±4.37)	11.67 (±1.86)	**20.11 (±2.92)**	82.99 (±2.10)	44.50 (±2.72)	**57.33 (±1.95)**

Results

- No CoNLL metrics used due to existence of one-to-many, many-to-one relations
- Difficult to do coreference resolution with a more complex set of relation types - results of the coarse setting on both inputs are higher
- LLM-base paraphrases as inputs are higher than the inputs using rule-based paraphrasing
    - Partly due to LLM paraphrasing quite often results in a concise form, not the fully “dense” form
    - In other words, “too dense” augmentation of prompts might be adding unnecessary noises

Results

labels	Preprocessing	CuI-F1	CuT-F1
Coarse (binary)	PrefixP paraphrase	9.31 (±6.81)	60.29 (±4.01)
	SubgraphGPT paraphrase	**19.07 (±5.67)**	**61.09 (±2.88)**
Fine (all subcats)	PrefixP paraphrase	20.02 (±1.38)	54.89 (±1.30)
	SubgraphGPT paraphrase	**20.11 (±2.92)**	**57.33 (±1.95)**

CUTL *ours* (Rim et al. 2023, table 3)

labels	Training	F-anaphora	F-relation
Coref (Atom)	Coref only	19.7 (±0.9)	11.2 (±0.8)
	Joint labels	22.0 (±0.9)	14.4 (±0.7)
Bridging	Brgd only	33.2 (±0.6)	24.5 (±0.6)
	Joint labels	32.7 (±0.7)	24.7 (±0.6)

RecipeRef (Fang et al. 2022, table 6, anaphora resolution results with state changes)

Results

(VERY ROUGH MAPPING, NOT THE SAME EVALUATION DATASET)

labels	Preprocessing	CuI-F1	CuT-F1
Coarse (binary)	PrefixP paraphrase	9.31 (±6.81)	60.29 (±4.01)
	SubgraphGPT paraphrase	**19.07 (±5.67)**	**61.09 (±2.88)**
Fine (all subcats)	PrefixP paraphrase	20.02 (±1.38)	54.89 (±1.30)
	SubgraphGPT paraphrase	**20.11 (±2.92)**	**57.33 (±1.95)**

CUTL *ours* (Rim et al. 2023, table 3)

labels	Training	F-anaphora	F-relation
Coref (Atom)	Coref only	19.7 (±0.9)	11.2 (±0.8)
	Joint labels	22.0 (±0.9)	14.4 (±0.7)
Bridging	Brgd only	33.2 (±0.6)	24.5 (±0.6)
	Joint labels	32.7 (±0.7)	24.7 (±0.6)

RecipeRef (Fang et al. 2022, table 6, anaphora resolution results with state changes)

Limitation & future work

- CUTL scheme (v1) didn’t address
  - Complex temporal order (reverse, overlap) - *partly fixed in CUTLER v2*
  - Ellipsis under conjunction and/or disjunction
  - Event negation
- All documents in the CUTL release (v1)
  - Are temporally linear (very convenient with recipe data, but not so scalable to other genres)
  - Have a single terminal state (final dish, common in instruction text, not so much otherwise)
  - Have a high density of object transformations
  - Have a high density of entities referred explicitly throughout the text
- Event semantic types are generalized to three categories: 1) transformation, 2) change-of-location, and 3) neither.
  - *(future work)* Finer event type categorization by utilizing existing large lexical resources, such as VerbNet

Conclusion

- We started with problems in manually marking up entities that are non-existing in surface text form.
- We propose POEM, a computational version of generalized result nominal.
- We implemented POEM in an annotation tool for coreference annotation.
- Annotated GRNs can be fed to applications for inference and generation via dense paraphrasing technique.
- Tracking anaphora over state changes can be improved by adding deep semantic information such as GRN

More information available at

brandeis-llc/dp-cutl on GitHub