Logical Story Representations via FrameNet + Semantic Parsing

Anonymous ACL submission

Abstract

We present a means of obtaining rich semantic representations of stories by combining neural FrameNet identification, a formal logic-based semantic parser, and a hierarchical event schema representation. The final schematic representation of the story abstracts constants to variables, preserving their types and relationships to other individuals in the story. All identified FrameNet frames are incorporated as temporally bound "episodes" and related to one another in time. The semantic role information from the frames is also incorporated into the final schema's type constraints. We describe this system as well as its possible applications to question answering and open-domain event schema learning.

1 Introduction

Story understanding requires deep, non-textual representations of textual information. The human brain, neural language models, and formal logic engines all transduce textual input into some other format in order to perform semantic tasks on that input. While formal logical representations of language admit more reliable and explainable inference procedures on text than, for example, the vector representations used by transformers, they suffer from characteristic brittleness when attempting to parse the true logical *meaning* of text: paraphrases and idioms stymie the logical capture of true semantics at best, and actively lead to incorrect understanding at worst.

The FrameNet project (Baker et al., 1998) attempts to provide a taxonomy of event "frames" (sometimes also called "schemas" or "scripts"), including their actors and objects, that one might observe in the real world, and thus in texts discussing the real world. These frames are not tied to any one means of expression: many different constructions, e.g. "she wolfed down the meal" and "she ate her

```
EPI-SCHEMA ((?X_C (COMPOSITE-SCHEMA.PR ?X_D)) ** ?E)
      :ROLES
            !R1  (?X_A FRIEND.N)
            !R2  (?X_A (PERTAIN-TO ?X_B))
            !R3  (?X_B AGENT.N)
            !R4  (?X_C MOM.N)
            !R5  (?X_C (PERTAIN-TO ?X_B))
            !R6  (?X_C MOTION-THEME.N)
            !R7  (?X_C INGESTION-INGESTOR.N)
            !R8  (?X_D HOUSE.N)
            !R9  (?X_D (PERTAIN-TO ?X_A))
            !R10 (?X_D MOTION-GOAL.N)
            !R11 (?X_E FOOD.N)
            !R12 (?X_E INGESTION-INGESTIBLES.N)

      :STEPS
            ?E1 (?X_C MOTION-GO.1.V ?X_D)
            ?E2 (?X_C INGESTION-EAT.2.V ?X_E)

      :EPISODE-RELATIONS
            !W2 (?E1 BEFORE ?E2)
```

Figure 1: An example of an Episodic Logic schema representing the story *"Jenny's mom went to her friend's house. She ate food there."* Noun predicates taken from single story tokens, e.g. FRIEND.N, are color-coded with their variables. Noun and verb predicates obtained from FrameNet matches are underlined, and prefixed with the name of the FrameNet frame before the hyphen. Additional information on the syntax and semantics of the schema is given by Lawley et al. (2021).

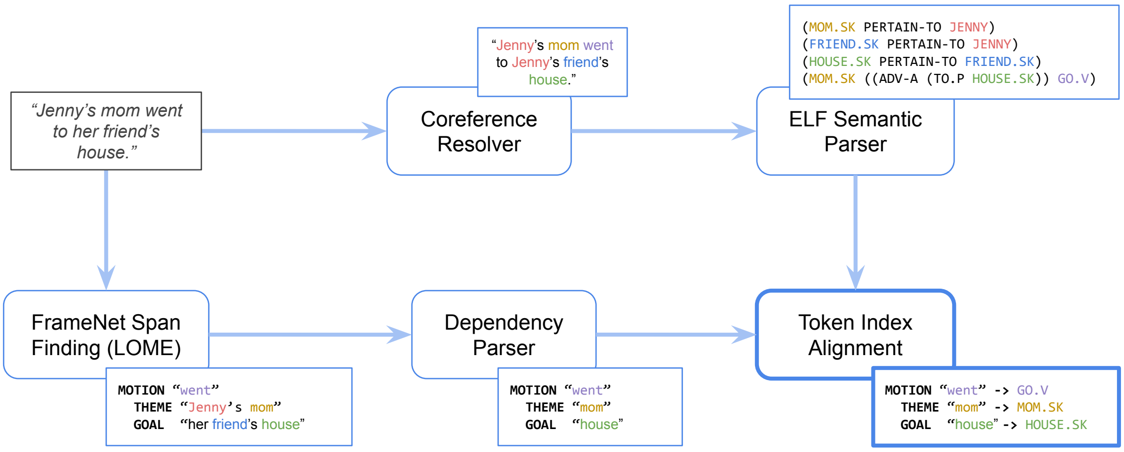

Figure 2: The architecture of the system. Raw story text is fed along two tracks: the logical-semantic parsing track, shown along the top, and the FrameNet parsing track, shown along the bottom. The FrameNet text spans are reduced to direct object tokens and correlated with logical individuals in the ELF parse via token index matching.

food", can express the same frame, e.g. "ingestion". These frames are constructed manually, however, rather than learned automatically from texts, and are defined in terms of natural language rather than a more manipulable representation. FrameNet parsing of text generally consists of the mapping of spans of text to FrameNet roles; these text spans, being natural language, are difficult to manipulate programmatically and draw inferences from.

In this paper, we present a means of producing expressive, semantically manipulable, formal logical "schema" representations of stories using a state-of-the-art FrameNet parsing system, LOME (Xia et al., 2021), as a jumping-off point. By augmenting FrameNet parses with logical semantic representations of the text, we obtain schema-like story representations that mitigate both the brittleness inherent to literal semantic parsing and the difficulty of manipulation inherent to natural language frames. We also discuss the potential application of these representations to the task of automatically acquiring event schema knowledge from natural text corpora.

2 Semantic Representation

The semantic representation we provide is based on Episodic Logic (EL) (Hwang and Schubert, 1993), a formal logical representation of language that enables efficient inference while maintaining a surface resemblance to the English language. One feature of EL that is well suited to story representation is its *characterizing* operator, $\ast\ast$, as seen in the example in Figure 2, which relates an Episodic Logic Formula (ELF) to an *episode*. Informally, $(\phi \ast\ast \texttt{E})$ means that `E` is "an episode of" some formula ϕ. These episodes, characterized by formulas derived from sentences, have temporal bounds, and can be related to each other in time using relations derived from the Allen Interval Algebra (Allen, 1983). Episodes are first-class citizens in Episodic Logic, and may be used as arguments to predicates, such as in the temporal relation formula `(E1 BEFORE E2)`.

ELFs, like those seen in the schema in Figure 1, often have predicates derived from nouns or verbs. For example, the schema ELF `(?X_A FRIEND.N)` asserts that the variable `?X_A` satisfies the predicate `FRIEND.N`, and the ELF `(?X_C MOTION-GO.1.V ?X_D)` can be read as a subject-verb-object verb phrase, where the arguments to the verb predicate, `MOTION-GO.1.V`, are the variables `?X_C` and `?X_D`.

2.1 EL Schemas

To represent frames identified by the FrameNet parser, as well as the story as a whole, we use the schema system built atop EL by Lawley et al. (2021). An example schema, produced by the system presented in this paper, is shown in Figure 1. This schema format allows declaration of entity types, and of relationships between entities, via EL propositions in the `Roles` section. The `Steps` section contains ELFs, and their characterizing episodes, for the schema's constituent events. These episodes are related in the `Episode-relations` section, and the entire schema may itself be embedded by the ELF formula known as its *header*, visible at the top of the schema, and characterizing an episode itself.

The EL schema framework we use allows for other section types, such as goals, preconditions, and postconditions, and was designed as part of a larger schema acquisition project. In this work, however, we primarily make use of the `Roles`, `Steps`, and `Episode-relations` sections for frame and story representation.

3 Architecture

Our system's architecture, illustrated in Figure 2, is divided into two main information pipelines: the EL track, responsible for semantic parsing, and the FrameNet track, responsible for frame identification and span selection. The information from both of these pipelines is unified into a final schematic representation at the end using token indices from the input text.

3.1 EL Track

To produce an EL semantic parse of the story, we first perform span mapping on the input text using the AllenNLP coreference resolver (Gardner et al., 2017). Co-referring token indices are saved, and story sentences are then converted into ELFs by first parsing them into ULF—an underspecified variant of EL (Kim and Schubert, 2019)—and then processing the ULFs into full ELFs by converting grammatical tense information into temporal relations and scoping quantifiers. More information on the ELF parser can be found in (Lawley et al., 2021).

Coreference resolution on the ELFs is performed by cross-referencing the token index clusters with token index tags placed on individuals in the EL parse. Co-referring individuals in the EL parse are then combined into one individual and substitutions are made throughout the parse.

3.2 FrameNet Track

To identify basic behavioral frames invoked by the raw text, we make use of the LOME information extraction system (Xia et al., 2021). LOME outputs invoked frames, and text spans that fill frame roles, as CONCRETE data files. Once we extract the invoked frames and text spans, we perform a syntactic dependency parse on the input text using spaCy (Honnibal and Montani, 2017) and identify the first token in each span with a `NSUBJ`, `DOBJ`, or `POBJ` tag. This allows any span of text containing tokens for multiple individuals, e.g. *her friend's house*, to be reduced to, e.g., *house*, which will be the token used to identify the logical individual in the EL parse during the alignment phase.

3.3 Token Index Alignment and Schema Formation

To represent the identified FrameNet frames as EL formulas, the text spans that fill the semantic roles for each frame must first be bound to logical individuals. After the dependency parser identifies the token to cross-reference with the EL parse, the noun predicate with the same token index is retrieved from the EL parse, and the individual satisfying that predicate is identified as the bound value for the frame role.

The verb that invoked the frame is identified in a similar fashion, and a schema is created with that verb's formula from the EL parse as its header, and with the names of the FrameNet semantic roles applied to the relevant individuals as semantic types in the new schema's `Roles` section. When multiple frames are converted to schemas in this way, they may all be embedded together in a *composite schema*, such as the one shown in Figure 1, with their header formulas as steps and with each of their inner type constraints shown in the composite schema's `Roles` section for clarity. This composite schema forms our final semantic representation of the story.

4 Discussion

The goal of our representation, and of semantic story representations in general, is to enable a variety of reasoning tasks. We discuss two interesting potential applications of this representation here: question answering and event schema acquisition.

4.1 Applications

4.1.1 Question Answering

Episodic Logic has been used for question answering (Morbini and Schubert, 2009), as has its underspecified variant, ULF (Platonov et al., 2020). EL formulas can be *unified* with one another, binding variables in one formula to constants or variables in another. Many questions about events or types can be formulated as EL propositions with variables to be bound to potential answers. For example, to answer the question of whose house the mom went to in the story represented in Figure 1, we could create the question formulas with new variables for the house and its owner: `(?X_C MOTION-GO.1.V ?house)` and `(?house (PERTAIN-TO ?who))`. The only valid unification of these formulas with the story binds the house `?X_D` to `?house` and the friend `?X_A` to `?who`. Identifying FrameNet frames when parsing the question into EL would allow many different phrasings of the same core frame of motion to receive the same answer, e.g., "whose house did the mom run off to?".

This form of question answering may also be used for semantic information retrieval based on multiple separate type, relational, and event occurrence constraints, for example, finding sets of stories where a person buys something edible at a store.

4.1.2 Schema Learning

When information about stereotypical situations is packaged up into event schemas, those schemas may be partially matched to new stories, and inferences may then be drawn from the unmatched pieces of those schemas: upon observing someone sitting down at a restaurant, for example, you might infer that they would then receive a menu, or that the restaurant probably serves food that they like. Automatic acquisition of a large and diverse corpus of event schemas has been pursued for decades, and researchers have employed symbolic (Mooney, 1988; Lebowitz, 1980) as well as statistical (Chambers and Jurafsky, 2008; Pichotta and Mooney, 2016) techniques.

The event schema syntax we use, taken from (Lawley et al., 2021), was conceived as part of a system for learning rich, logical event schemas from texts by using a set of simple behavioral *protoschemas*—concepts children are familiar with, like asking for assistance with a task or eating food to alleviate hunger—to bootstrap the acquisition of more complex schemas. We believe that our conversion of identified FrameNet frames to canonicalized logical formulas could aid this process: many FrameNet frames resemble simple behavioral protoschemas, and a mapping between them could greatly enhance the linguistic coverage of protoschemas.

4.2 Limitations

While our system produces useful representations, extant Episodic Logic parsing software, especially ULF parsing, is still somewhat error-prone. Work on EL parsing is ongoing, and notably includes an application of the cache transition parsing system developed by Peng et al. (2018) to ULF parsing (Kim, 2019), which is the initial step in converting English text into a logical form.

We also note that we do not leverage the full schema syntax of Lawley et al. (2021), and in particular have not added stated goals, preconditions, and postconditions from FrameNet frames into the relevant sections from that schema system. This is due, in large part, to the lack of availability of those particular semantic roles in current FrameNet parses.

Finally, we note that our system was developed using only stories from the ROCstory corpus (Mostafazadeh et al., 2016), and that grammatically and conceptually complex texts may require additional parsing techniques, better parser performance, a larger initial corpus of *learned* schemas instead of hand-created frames, or any subset of these.

5 Conclusion

We have presented a system for obtaining rich, formal logic-based, schema-like representations of stories from text by combining the frame identification power of LOME and FrameNet with the semantic representation power of Episodic Logic schemas. We showed that these representations normalize language into propositions based on semantic frames; model type, relational, and temporal constraints; and allow for hierarchical nesting of situations. Finally, we discussed their potential application, in future work, to tasks that neither FrameNet nor EL parsing alone is trivially capable of, such as paraphrase-resistant question answering, information retrieval, and automatic acquisition of event schemas from text.

References

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98/COLING '98, pages 86–90. Association for Computational Linguistics.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. *Proceedings of ACL-08: HLT*, pages 789–797.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Chung Hee Hwang and Lenhart K Schubert. 1993. Episodic logic: A situational logic for natural language processing. *Situation Theory and its Applications*, 3:303–338.

Gene Kim and Lenhart Schubert. 2019. A type-coherent, expressive representation as an initial step to language understanding. In *Proceedings of the 13th International Conference on Computational Semantics*, Gothenburg, Sweden. Association for Computational Linguistics.

Gene Louis Kim. 2019. Towards parsing unscoped episodic logical forms with a cache transition parser. In *the Poster Abstracts of the Proceedings of the 32nd International Conference of the Florida Artificial Intelligence Research Society*.

Lane Lawley, Benjamin Kuehnert, and Lenhart Schubert. 2021. Learning general event schemas with episodic logic. In *Proceedings of the 1st and 2nd Workshops on Natural Logic Meets Machine Learning (NALOMA)*, pages 1–6, Groningen, the Netherlands (online). Association for Computational Linguistics.

Michael Lebowitz. 1980. *Generalization and Memory in an Integrated Understanding System*. Ph.D. thesis, New Haven, CT, USA. AAI8109800.

Raymond J. Mooney. 1988. *A General Explanation-Based Learning Mechanism and its Application to Narrative Understanding*. Ph.D. thesis.

Fabrizio Morbini and Lenhart Schubert. 2009. Evaluation of Epilog: A reasoner for Episodic Logic. In *Proceedings of the Ninth International Symposium on Logical Formalizations of Commonsense Reasoning*, Toronto, Canada.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Xiaochang Peng, Daniel Gildea, and Giorgio Satta. 2018. AMR parsing with cache transition systems. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-18)*.

Karl Pichotta and Raymond J Mooney. 2016. Learning statistical scripts with lstm recurrent neural networks. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Georgiy Platonov, Lenhart Schubert, Benjamin Kane, and Aaron Gindi. 2020. A spoken dialogue system for spatial question answering in a physical blocks world. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 128–131, 1st virtual meeting. Association for Computational Linguistics.

Patrick Xia, Guanghui Qin, Siddharth Vashishtha, Yunmo Chen, Tongfei Chen, Chandler May, Craig Harman, Kyle Rawlins, Aaron Steven White, and Benjamin Van Durme. 2021. LOME: Large ontology multilingual extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 149–159, Online. Association for Computational Linguistics.