Towards Open Domain English to Logic Translation

Adam Pease and Josef Urban

[1] Articulate Software, San Jose, CA, USA
apease@articulatesoftware.com
[2] CIIRC, Czech Technical University, Prague, Czechia
jurban@cvut.cz

Introduction

Logic-based systems have high degrees of trustworthiness due to their ability to document their reasoning and sources. It is possible that logic-based systems may be used in the future to provide plausible explanations of answers provided by machine learning systems or to test their outputs against known facts. However, there have been several barriers to language to logic translation. Often very simple resulting logics are used, limiting the generality and power of the portion of natural language semantics that can be captured. Approaches that have used linguistic elements as though they were logical terms suffer from the absence of background knowledge that anchors the meaning of those terms and ensures that machine inference conforms to human understanding of linguistically-expressed concepts. Rule-based parsing has the difficulty of scaling up manual creation of language-to-logic interpretation rules. Approaches to training a machine learning based system have been hampered by the challenge of creating training pairs of language and their equivalent logical translations. Previous work in *auto-formalization* of mathematics has shown how it is possible to convert informal descriptions into fully formal logic expressions [13].

Our approach attempts to address these issues by using an expressive higher order logic and a very large theory of world knowledge with a comprehensive mapping to linguistic tokens, and a synthetic corpus. Our focus here is on the generation of training data of language and logic pairs.

We utilize the Suggested Upper Merged Ontology (SUMO)[6, 8], a comprehensive ontology of around 20,000 concepts and 80,000 hand-authored logical statements in a higher-order logic, that has an associated tool set called Sigma[10], integrated with leading theorem provers such as Eprover [11] Vampire [5] and LEO-III [1], and manually-created links[7] to the WordNet lexico-semantic database[4]. We have described [10] elsewhere how to translate SUMO to the strictly first order language of TPTP [12], as well as TF0/TFA [9] and TH0/THF[2, 3].

Synthetic Corpus

We create a simple frame structure of linguistic elements that can be turned into a sentence and a logical expression. We started with a simple subject-verb-object structure that corresponds to the most common English sentences, and then added extra features incrementally. SUMO has such a large set of concepts and their corresponding linguistic equivalents that we can generate millions of sentences even for some of the simplest variations. Thanks to SUMO's collection of higher-order relationships we can include statements of authorship, belief, normative force and many other constructs that have been conspicuously absent in prior efforts at language to logic translation.

Our conceptual library, along with lexical presentations of each of the concepts, allows us to generate 1323 **Process** types - roughly equivalent to verbs, describing types of actions; 67 **CaseRole**s that describe the roles that entities play in processes; 930 **Object** types that can be subjects, direct objects or indirect objects; 323 **SocialRole**s that refer to people by their professions or other social characteristics; and 100 names of people.

We generate a wide variety of linguistic features, some of which are:

- You understood - imperatives - "Chop some wood!"
- epistemics - believes, knows - "John knows that Mary chops some wood."
- modals - possibility, necessity - "John may chop some wood."
- normative force - obligation, permission, prohibition - "John ought to chop some wood."
- numbers and units, quantities, qualifiers - "some" - "John chops 100 pounds of wood."
- times and dates - "On Tuesday, John chops some wood."
- politeness - "Please chop some wood."
- negation - "John doesn't chop some wood."
- desires - "Mary hopes that John chops some wood."
- authorship - said, wrote, quoted or unquoted - "Mary said 'John chops some wood.'"

Just for subject-verb-object-indirectObject sentences we can theoretically generate 200 trillion combinations, and that does not include most of the additional linguistic features we can generate as listed above. However, not all combinations make sense. While SUMO has logical definitions that restrict many such spurious combinations (for example, that "John" can't be `Eating` a `Table`) it is impractical in terms of the time required to run theorem proving to test all combinations. So we use SUMO's relation `capability` which relates types of processes to the types of things that can play specific roles in those processes. We also added the relation `prohibitedRole` to express combinations that are non-sensical. Each of these relations is defined axiomatically so it can also support theorem proving, but is in a standard form that is read into a table that can be checked very quickly during sentence generation. Reviewing generated sentences for bad combinations has been an important part of this work and creates a useful byproduct - preventing nonsensical sentences from being generated requires an understanding of why these combinations do not accord with common sense, thereby adding more knowledge to SUMO about how the world does or does not work. Finally, the generation of a certain percentage of nonsense sentences has an impact only on the efficiency of the data set, rather than the resulting correctness of the trained system. It simply allows the neural network to learn plausible logical equivalents for nonsense sentences. As long as those examples are not so prevalent as to dominate training time, there is no impact.

In addition, SUMO has language generation templates for all relations. We generate arguments for these relations according to the type signature of the relation. This provides a wide variety of sentence types beyond the notion of action sentences that our parameterized sentence generation covers. Yet another generation step is provided by creating natural language paraphrases of all statements in SUMO.

While there is no way we can create a template or process to generate all possible types of sentences through manual anticipation of their structure, this does give us a very large and varied corpus of sentences with a deep formal semantic equivalent.

Conclusion and Future Work

We are using Google's Neural Machine Translation system[1] to train our translator. In 5000 epochs we achieve a perplexity of 1.01[2] on a corpus of 10 million sentences and their logical equivalent. Our next step is to derive measures of success and coverage. Sigma can tell us whether a given formula correct syntax, and relation argument types. Those tests are quick. A more expensive test is use theorem proving to see whether each new statement entails a contradiction with respect to existing statements in SUMO. This however is not necessarily a sign of a flaw in translation from language to logic, since all human-authored texts are not factually consistent with one another. We will begin by processes statements in the COCA news corpus, which is the largest freely available corpus of American English.

[1] https://github.com/tensorflow/nmt

[2] code at https://github.com/ontologyportal/sigmakee and https://github.com/JUrban/sumonlp

References

[1] Christoph Benzmüller, Laurence Paulson, Frank Theiss, and A. Fietzke. (2008). *LEO-II - A Cooperative Automatic Theorem Prover for Higher-Order Logic. In Proceedings of the Fourth International Joint Conference on Automated Reasoning (IJCAR'08), LNAI volume*, 5195:162–170, 2008.

[2] Christoph Benzmüller and Adam Pease. Progress in automating higher-order ontology reasoning. In Boris Konev, Renate Schmidt, and Stephan Schulz, editors, *Workshop on Practical Aspects of Automated Reasoning (PAAR-2010)*. CEUR Workshop Proceedings, Edinburgh, UK, 2010.

[3] Chad Brown, Adam Pease, and Josef Urban. Translating SUMO-K to Higher-Order Set Theory. submit/4892900, 2023.

[4] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[5] Laura Kovács and Andrei Voronkov. First-order theorem proving and vampire. In *Proceedings of the 25th International Conference on Computer Aided Verification*, volume 8044 of *CAV 2013*, pages 1–35, New York, NY, USA, 2013. Springer-Verlag New York, Inc.

[6] Ian Niles and Adam Pease. Toward a Standard Upper Ontology. In Chris Welty and Barry Smith, editors, *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, pages 2–9, 2001.

[7] Ian Niles and Adam Pease. Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology. In *Proceedings of the IEEE International Conference on Information and Knowledge Engineering*, pages 412–416, 2003.

[8] Adam Pease. *Ontology: A Practical Guide*. Articulate Software Press, Angwin, CA, 2011.

[9] Adam Pease. Arithmetic and inference in a large theory. In *AI in Theorem Proving*, 2019.

[10] Adam Pease and Stephan Schulz. Knowledge Engineering for Large Ontologies with Sigma KEE 3.0. In *The International Joint Conference on Automated Reasoning*, 2014.

[11] S. Schulz. E – A Brainiac Theorem Prover. *Journal of AI Communications*, 15(2/3):111–126, 2002.

[12] Steven Trac, Geoff Sutcliffe, and Adam Pease. Integration of the TPTPWorld into SigmaKEE. In *Proceedings of IJCAR '08 Workshop on Practical Aspects of Automated Reasoning (PAAR-2008)*. CEUR Workshop Proceedings, 2008.

[13] Qingxiang Wang, Chad E. Brown, Cezary Kaliszyk, and Josef Urban. Exploration of neural machine translation in autoformalization of mathematics in mizar. In *CPP*, pages 85–98. ACM, 2020.