Paper Proposal : Steps towards syntactic and semantic safety guarantees for voice assistants in autonomous vehicles

Warrick Macmillan
Marco, Matthew, Katya, ...

1 Abstract

We introduce a grammar for a controlled natural language (CNL) to give imperative commands to a voice assistant for a self-driving car to verify its behavior. We specifically seek to give the user of our system assurances against substitution-based attacks, whereby synonyms can be given the same meaning by imposing these conditions in how the parse trees (in our case, abstract syntax trees (ASTs)) are formed. In addition, we map our trees to a semantic form in an Agda implementation of linear temporal logic (LTL), which has many applications in the specification and verification of the behavior of robotics systems, particularly those with neural network components. We see that simple, formal systems provide a useful model for verifying systems with more a greater breadth of "knowledge" capable of more complex interactions.

2 Overview

Perhaps the most pervasive question pervading the use and application of natural language technologies, and one which exemplifies the boundary of the formal methods versus statistical methods approaches in designing such technologies, can be states as follows : How does one optimize for wide coverage of the system in addition to ensuring that system is robust?

For the statistical, or data-oriented machine learning NLP methods, which have made enormous gains over the past three decades, often take the pragmatic approach : compromise robustness for wide coverage, as this means the tools can be used by non-experts. The formal approaches in computational linguistics, instead, are often more concerned with theoretical explainability, and therefore develop systems which have seemingly more predictable and well-defined behavior for specific problem domains, yet fail to generalize without an explosion in complexity when presented with data outside their domain. The fact that natural language both seems structured with respect to rules, yet continuously breaks or introduces exceptions to these rules, makes it exceedingly hard penetrate from either the data oriented or formalist approach, leading many researchers to ask to what degree large amounts of data can be augmented with theoretical knowledge about language to create optimal systems with respect to both breadth and depth of coverage.

This problem acutely arises in when trying to design a voice assistant in the domain of commanding controllable robotics, specifically, autonomous vehicles. For the actions the vehicle takes, mostly the motion and path decisions, must be formally specified or at the very least controlled via some computer system subject to mathematical formalism. Assuming the user directing the vehicle isn't aware of these formalisms, it is incredibly difficult to design a controller capable of dealing with the breadth of language one may encounter in the wild.

The instructions an arbitrary user gives are not subject to the same formalities. For they may leave out detail ("go into the other lane" with multiple lanes on either side), say something wrong ("go into the other lane" on a single lane road), or give a command the controller should recognize as bad ("drive into the car ahead"). Additionally, the controller may need to recognize many ways many users may say "the same thing", that is the same with respect to some semantic formalism. Ambiguity and noise are other considerations to try to deal with (presumably through some dialogue system). Finally, there is the possibility of adversarial attacks, which we'll detail more below.

We therefore analyze our big-picture question above in the following sub-question : how can one map the manifold ways of presenting information to an autonomous robot into a rigorous and formally verifiable subsystem which the controlled can understand? Our proposed solution is to build a semantic parser from natural language commands to Linear Temporal Logic, whereby we can filter the many possible natural language commands into a "canoncial subset" which are equivalent to (sets of) temporal logic formulas. The "sets of" clause references the inevitable ambiguity of parses even from a big enough parser, even if the size of the canoncial expressions is vastly smaller than the domain of expressions mapping to them.

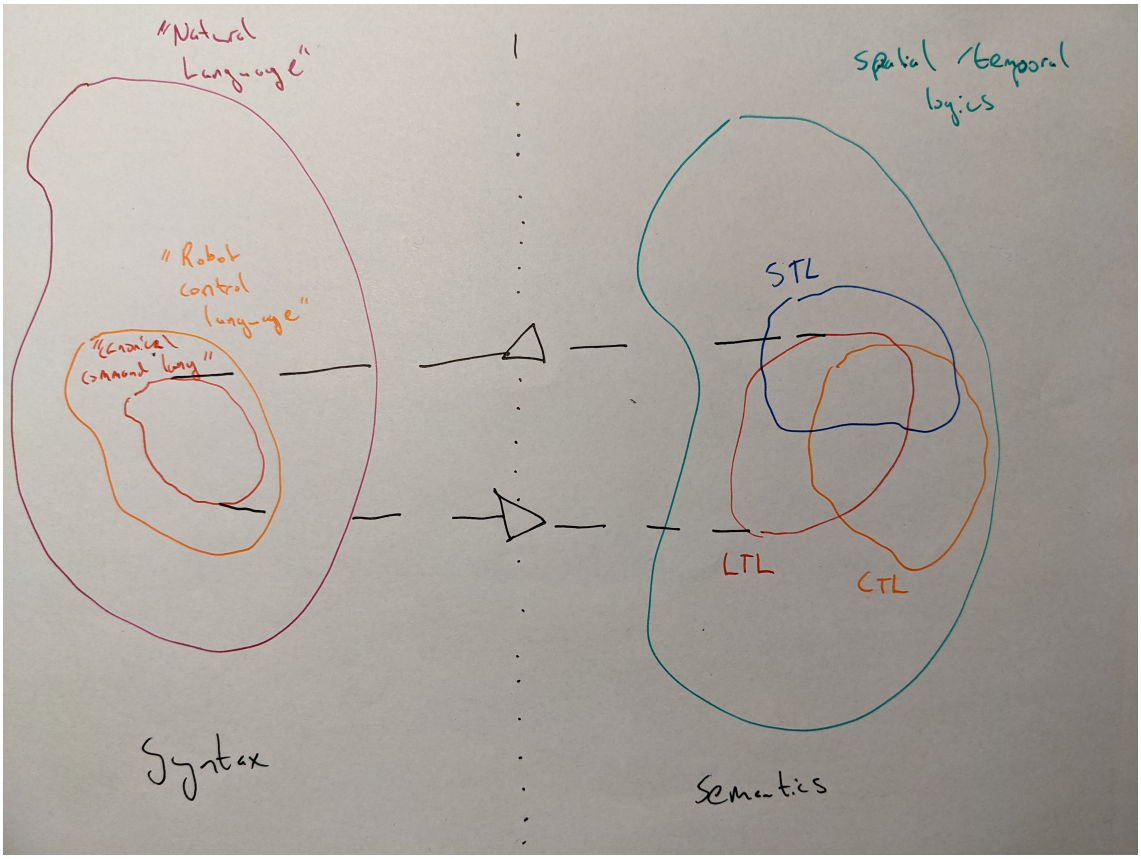

Figure 1: Language and Logical Spaces of Concern

We begin in Figure 1 with a high level overview of this semantic parsing system, whereby the space of natural language syntax can be mapped to some formal language semantic space (and possibly have some kind of inverse mapping). We note that "Natural Language", while an idealized notion, can be thought of the space of interpretable utterances. The relatively small subset of these utterances which one might give to a robot, labeled "Robot Control Language", is the ideal breadth our system would support, is still actually very large. We therefore applying another filter, to the "Canonical Command Language" which is inductively defined via some relatively thin set of grammar rules, which simultaneously generate and parse expressions in some logic. Although we target LTL because of its prominence in the literature and relatively straightforward implementation and interpretation, it should be noted that their are other temporal logics which may well be more expressive and better suited to the actual problem of synthesizing controllers.

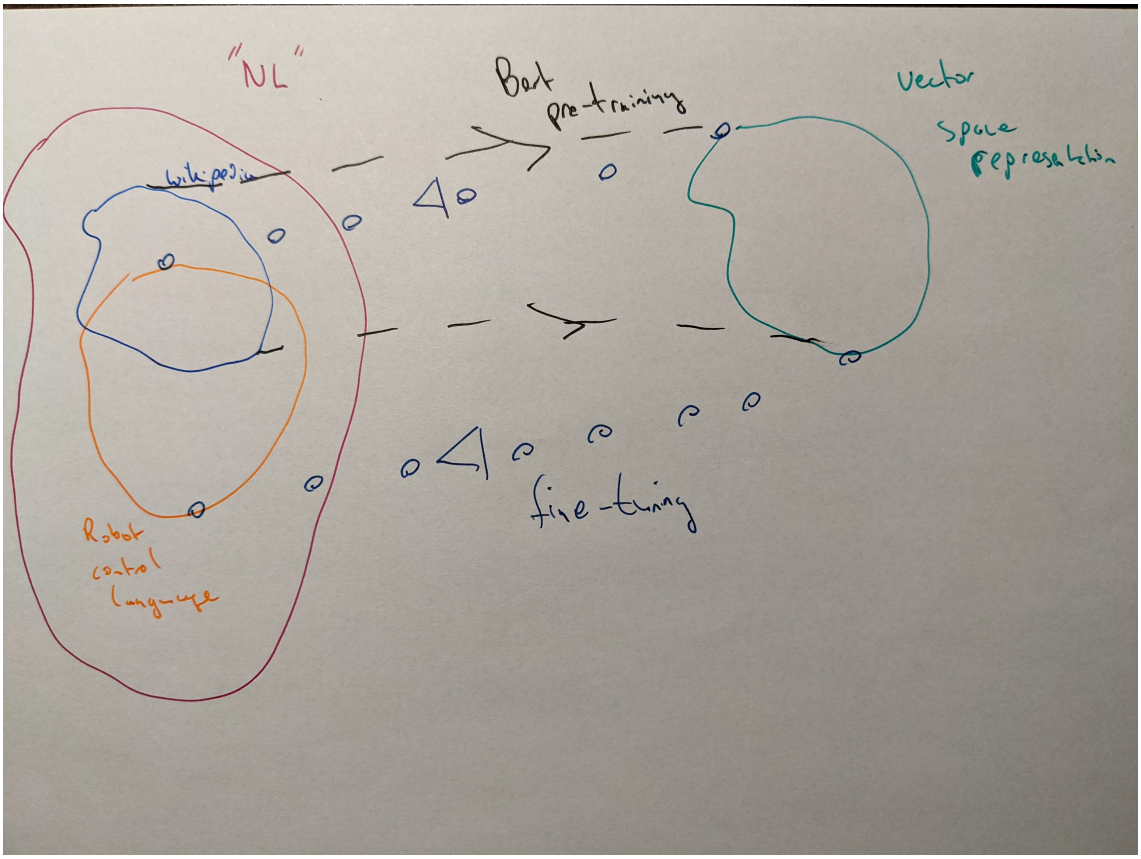

Figure 2: A simple caption

in Figure 2

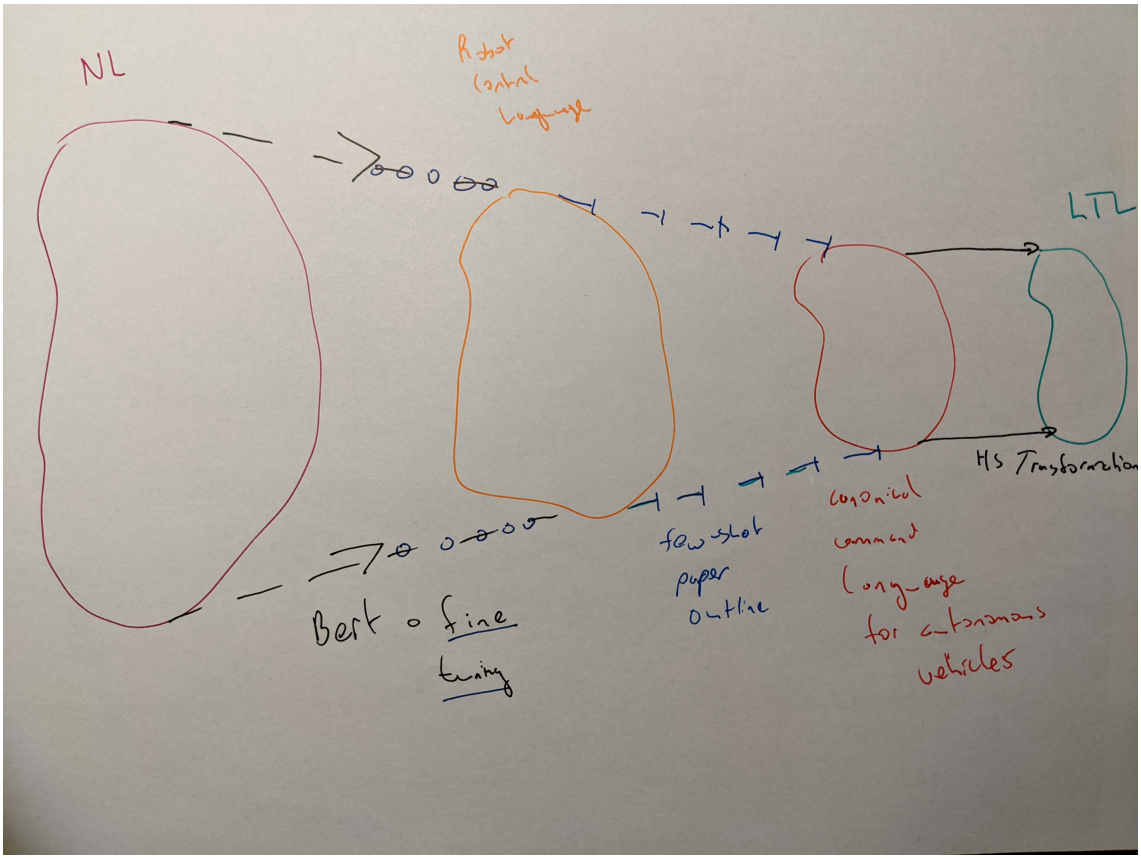

Figure 3: A simple caption

in Figure 3

3 Introduction

While the evaluation of machine learning systems provides assurances using different scores and metrics on different tasks assures one they may on average perform better than humans at certain tasks, the advent of adversarial attacks [24] with the intention of deceiving such a system by a hostile actor leads the system designer to desire, and possible require additional verification about the system's behavior. In the context of natural language processing (NLP), where data sources rely on strings of text, these attacks can focus an array of features from spellings of individual words to rearranging entire sentences []. So-called synonym attacks, which adversarially target the system at the lexical level, can cause traditional NLP models to [...] [].

In the context of designing a voice assistant for an autonomous vehicle, whereby one can give commands like "turn right after the woman with the big dog", we desire that the intensional belief a user has about her utterance is consistent with the extensional behavior of the vehicle. This can be done through an intermediary mapping to a formal semantic representation. Ensuring that the syntactic content of a voice director's (well-formed) utterance maps predictably to the logical form is important from the verificationist perspective : one wants to maximize the "syntactic completeness" of the system [17].

Aside from the user experience being compromised by a system which has been adversarially afflicted, there is also a possibility of physical danger for the passenger and other people in the vicinity. As voice directed robots have many possible points of failure, we focus on two types of verification for our system. Rather than focus on breadth of language coverage, which ML language models excel at of due to their reliance on statistical modeling and tons of data, our system is narrowly focused as a proof-of-concept, from which it could either be extended by

hand, or different components modified using other techniques and tools.

4 Current Landscape

4.1 Voice assistants for autonomous vehicles

The public company Cerence [] is already designing voice assistants for autonomous vehicles, for which it has a large software stack between the voice processing to actual control of current automative components. In addition to its technologies, many of which aren't accessible to external researchers due to intellectual property restrictions, Cerence has contracts with large automakers [..]. It is therefore natural to inquire, what a small team with varied backgrounds and not nearly the same expertise nor experience within the technological team at Cerence can provide.

First, we believe that the focus on verification, insofar as we envision it, is unlikely to be of current concern at Cerence due to the fact that their products are still being developed, and the primary goal of producing a working product is likely to precedence over preventing non-existent hostile actors.

Additionally, it is going to have to be determined by [verification of self-driving cars generally : software, hardware, behavior in a real environment, etc]

4.2 Natural Language and Robots, generally

4.3 Semantic Representations of NL for verification

Modal logics, specifically those dealing with time like LTL, CTL, STL, ..., have been used extensively in the specification and verification of properties of robotics systems, including autonomous vehicles [cite]

With verification being a core motivation of our work, we take for granted that these different logics have many manifestations in different systems. However, we hope that by choosing a domain with a lot of attention, that our system can be generalized in many possible directions :

- other logics
- other parsing formalisms (perhaps dependency for wide-coverage)
- other syntax -> semantic formalisms
- other robotics domains

4.4 Foundation Models

5 Work

5.1 GF Grammar

6 TODO

6.1 Grammar modulo wordnet

6.2 LTL in Agda

Along with colleagues from Singapore Management University, we have begun an Agda implementation [18] of LTL which will serve as the semantic space for our parsed utterances. Our method, uses a deep embedding, as opposed to the shallow embedding in [7], although the temporal encoding of paths as streams was directly adapted from this paper.

This implementation will hopefully allow us to prove decidability of LTL in a relatively straightforward manner. Other than the assurance that our implementation is correct, we hope this will allow us to feed the formula into some SAT or SMT solver so-as to actually allow verification of the behavior of a vehicle with respect to an utterance.

[TODO : Help from Matthew?]

6.3 AST -> Agda

6.4 ML training/verification stuff

Help from Marco, Nathalia if interested?

7 Publications Description

Realizing that the structure of the paper is ameanable to large changes, I'm posting a summary of relevant publications here.

7.1 Statistical (pre-trained) Language Models

The first set of publ

- In [23] [under review], the authors show how, using a *synchronous context-free grammar* (SCFG) to define a minified CNL with a parallel and dually parsible semantic form, that one can use a large pre-trained language model as a front-end to filter a much wider syntax into the CNL. I postulate GF's expressivity is more expressive than the SCFG, at least based off a tertiary reading in the index, and therefore if we carved out a subset of commands to cohere with our LTL (and maybe some other temporal or even spatial-temporal logics in the future), our model would be amenable to a similar "out-of-the box" semantic parser that could actually be used for verification. This paper borrows the idea of "semantic parsing as paraphrasing" from [2]
- In [22], the authors advocate for getting rid of parsers alltogether, although this naively takes for granted large public data-sets, none of which exist for an autonomous vehicle and temporal logic formalism
- [11] [under review] claims that Bert is robust, analyzing claims of four papers, including the one which uses a wordnet attack

7.2 NL to TL

Here we show mainly relevant research for NL to LTL.

The applications of LTL in machine learning are vast, and the scope of our specific application is still unclear, but nevertheless, we give a literature review of methods and applications relevant for our work.

- This paper [9] from 2009 uses a categorial grammar approach, but more or less can serve as an idea template for us, also nice pictures with grammar rules and formulas
- Also, a highly relevant template combines Natural Language, LTL, with the idea of having a verifiable pipeline [15]

  > The production of an on- tology of common actions and the type of formulas that they produce—for example, safety conditions, adding goals, constraining the initial state—in their negated and positive forms would be a step toward a more general solution to the problem of mapping natural language to LTL. Previous work has relied heavily on grammar formalisms to ease... [15]

- LTL formulas can be transformed into automata which can then be used as reward functions for reinforcement learners, as in [6]
- The following is one of the more relevant quotes from a paper reviewing the whole space of English to LTL translations

  > Overall, the typical approach followed by these studies can be summarized as follows: given an input English utterance, preprocess it to extract syntactical

> information, which may include part of speech tagging, dependency parsing, semantic role labelling, and so on. Then, enrich the input with these pieces of information. Finally, run an attribute grammar-based parser, or rely on some hand-made rules, to derive a translation into a target logical format. A notable exception is the work of [89], where a fully-supervised learning setting is considered. [5]

- Translating between English and STL can be done via a large language model [12] [under review], but the domain specificity of the problems are still significant enough to suggest that it will be years before an automated semantic parser is available, if it is even possible.
- Could ask Lapata in Edinburgh, whose work [8] is relevant and well-cited (although they use an encoder-decoder method)
-
-
-

7.3 Tellex

Stephanie Tellex has written extensively about natural language inputs and interfaces with robots. Although she has not specifically written about autonomous vehicles, the domains have enough intersection to warrant careful consideration of much of her work, especially the recent stuff.

- Grounding with an intermediate symbolic state, no LTL, but possibly relevant for paper generally. She also cites [16], a seminal paper in this area

  > Instruction following is a supervised learning problem where the agent must predict a trajectory that would satisfy an input natural language command. [10]

- The review paper [19] making recommendations has a section on robustness, but this is mostly for the sake of allowing sharing of interfaces and efficacy, no mention of verification (which is what we're primarily after)
- They design a NL -> LTL for drones that are grounded to actual landmarks [3]
- The group builds a trained pipeline that uses an object oriented template-instance methodology to generalize to different ontological categories in [13] [under review]
- In [20] build learn a semantic parser from NL to LTL (so that the language is grounded) where they collect executions of the LTL formulas in different environments using a weakly-supervised training method with reinforcement learning Part if the paper has to do with the execution of the command being dependent on the path taken by the robot executing the command, not just meeting the goal requirements, thereby giving a complexity bonus in comparison to previous work. She also evaluates the model on the [16] data set

7.4 Robot Motion Planning

Without getting to into the weeds, for the actual planning and control of the robot should, at least hypothetically, be at least somewhat

> Robot motion planning and control is the problem of automatic construction of robot control strategies from task specifications given in high-level, human-like language. The challenge in this area is the development of computationally efficient frameworks allowing for systematic, provably correct, control design accommodating both the robot constraints and the complexity of the environment, while at the same time allowing for expressive task specifications. [1]

- For instance, in [21] the authors indicate how to actually ground basic propositions from the language to paths in a space, while our model, outputting formulas un-grounded base predicates, is merely concerned with the logical structure.
-
- In [4], the authors borrow the English to LTL pipeline from the aforementioned [15], and synthesize grounded controllers

There is a group at MIT (Kuo, Katz, Barbu, ..) doing seemingly similar things to Tellex's group.

- In [14], the authors do the most general end-to-end task without intermediary states, namely, map natural language commands to navigation and manipulation tasks. While this "cutting out the middle man" mentality may be an idealistic long-term vision, it makes the system much too much of a black box - for the fine-tuned verification conditions we desire to express and impose via the intermediate symbolic representations, the intermediate states, we imagine give us a more explainable, predictable, and regulatable system
- More similar to Tellex et al's approach [20], [25] seek to train a model via images in a simulated world. Their work also uses a SCFG (to generate semantically inadequate sentences with corresponding LTL formulas) from which they can direct machines to follow the instructions, and then have users describe the instructions in more natural form.
- One of the

References

[1] Calin Belta et al. "Symbolic planning and control of robot motion [Grand Challenges of Robotics]". In: *IEEE Robotics Automation Magazine* 14.1 (2007), pp. 61–70.

[2] Jonathan Berant and Percy Liang. "Semantic Parsing via Paraphrasing". In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 1415–1425.

[3] Matthew Berg et al. "Grounding Language to Landmarks in Arbitrary Outdoor Environments". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, pp. 208–215.

[4] Adrian Boteanu et al. "A model for verifiable grounding and execution of complex natural language instructions". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2016, pp. 2649–2654.

[5] Andrea Brunello, Angelo Montanari, and Mark Reynolds. "Synthesis of LTL Formulas from Natural Language Texts: State of the Art and Research Directions". In: *26th International Symposium on Temporal Representation and Reasoning (TIME 2019)*. Ed. by Johann Gamper, Sophie Pinchinat, and Guido Sciavicco. Vol. 147. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019, 17:1–17:19.

[6] Alberto Camacho et al. "LTL and Beyond: Formal Languages for Reward Function Specification in Reinforcement Learning". In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, July 2019, pp. 6065–6073.

[7] Solange Coupet-Grimal. "An Axiomatization of Linear Temporal Logic in the Calculus of Inductive Constructions". In: *Journal of Logic and Computation* 13.6 (2003), pp. 801–813.

[8] Li Dong and Mirella Lapata. "Language to Logical Form with Neural Attention". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 33–43.

[9] Juraj Dzifcak et al. "What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution". In: *2009 IEEE International Conference on Robotics and Automation*. 2009, pp. 4163–4168.

[10] Nakul Gopalan et al. "Simultaneously Learning Transferable Symbols and Language Groundings from Perceptual Data for Instruction Following". In: *Robotics: Science and Systems XVI, Virtual Event / Corvalis, Oregon, USA, July 12-16, 2020*. Ed. by Marc Toussaint, Antonio Bicchi, and Tucker Hermans. 2020.

[11] Jens Hauser et al. *BERT is Robust! A Case Against Synonym-Based Adversarial Examples in Text Classification*. 2021. arXiv: 2109.07403 [cs.CL].

[12] Jie He et al. *From English to Signal Temporal Logic*. 2021. arXiv: 2109.10294 [cs.CL].

[13] Eric Hsiung et al. *Generalizing to New Domains by Mapping Natural Language to Lifted LTL*. 2021. arXiv: 2110.05603 [cs.CL].

[14] Yen-Ling Kuo, Boris Katz, and Andrei Barbu. "Deep compositional robotic planners that follow natural language commands". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 4906–4912.

[15] Constantine Lignos et al. "Provably Correct Reactive Control from Natural Language". In: *Auton. Robots* 38.1 (Jan. 2015), pp. 89–105.

[16] Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. "Walk the Talk: Connecting Language, Knowledge, and Action in Route Instructions". In: *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*. AAAI'06. Boston, Massachusetts: AAAI Press, 2006, pp. 1475–1482.

[17] Warrick Macmillan. "On the Grammar of Proof". MA thesis. University of Gothenburg, 2021, p. 90.

[18] Warrick Macmillan and Andreas Kallberg. *LTL in Agda*. https://github.com/wmacmil/LTL-Agda. 2021.

[19] Matthew Marge et al. "Spoken language interaction with robots: Recommendations for future research". In: *Computer Speech and Language* 71 (2022), p. 101255.

[20] Roma Patel, Stefanie Tellex, and Ellie Pavlick. "Learning to Ground Language to Temporal Logical Form". In: (2019).

[21] Erion Plaku and Sertac Karaman. "Motion planning with temporal-logic specifications: Progress and challenges". In: *AI communications* 29.1 (2016), pp. 151–162.

[22] Subendhu Rongali et al. "Don't Parse, Generate! A Sequence to Sequence Architecture for Task-Oriented Semantic Parsing". In: *Proceedings of The Web Conference 2020*. WWW '20. Taipei, Taiwan: Association for Computing Machinery, 2020, pp. 2962–2968.

[23] Richard Shin et al. "Constrained Language Models Yield Few-Shot Semantic Parsers". In: *CoRR* abs/2104.08768 (2021). arXiv: 2104.08768.

[24] Christian Szegedy et al. "Intriguing properties of neural networks". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2014.

[25] Christopher Wang et al. "Learning a natural-language to LTL executable semantic parser for grounded robotics". In: *CoRR* abs/2008.03277 (2020). arXiv: 2008.03277.