I first took an interest in computational linguistics while learning to speak Tagalog. Through the process, I learned a deep appreciation for the language's rich morphology, and the immense complexity of language in general. My enthusiasm for language has not diminished since that time and I am excited to pursue a PhD in natural language processing (NLP).

Though some subfields of NLP research have advanced by leaps and bounds in the past few years, our understanding of how these advances have come about remains unsatisfactory. While transformers may get impressive results on, e.g., question answering, we do not know what the fundamental limits of what current architectures can compute or approximate, or what these models are doing internally. Do they learn syntactic rules? Can they generalize to new problems composed of known concepts? In my PhD I hope to answer some of these questions by **drawing on fields such as formal language theory and syntax to develop practical and theoretical frameworks for understanding modern NLP methods**.

Understanding LMs through linguistics and formal language theory. Modern NLP has become largely divorced from our understanding of linguistics and cognition in humans. I am interested in reconciling our understanding of both. As my first foray into this effort, I set out to discover how modern LMs handle syntactic agreement. In my first-author ACL 2021 paper and oral presentation [1] we intervene on individual neurons in large transformers to observe their causal effect [2] on syntactic agreement. We find that transformers learn two distinct mechanisms for number agreement, depending on whether the relevant tokens are adjacent or not. Subsequent research has built on our work in order to analyze other linguistic phenomena such as distributivity [3], and the effect of relative clauses on agreement [4]. During my PhD I hope to continue this line of research by exploring how LMs handle (or fail to handle) other linguistic phenomena. I am particularly interested in linguistic generalization. For instance, perhaps causal analysis could help us understand how transformers handle and acquire new words inference time by intervening on contextualized embeddings within the model. Strengthening our understanding of how LMs deal with well-studied linguistic phenomena can give insight into what inductive biases these models learn and how to make them better. I also believe that insights from computational methods can contribute to the field of linguistics by empirically lower-bounding the computational ingredients for producing various linguistic phenomena that we see in human speech.

As noted earlier, I am also excited by projects that borrow from computation and formal language theory to generate insights about LMs. This approach makes it possible to answer very general questions that might otherwise be very difficult to approach. For instance, I used regular languages to measure the capabilities of **transformers as general instruction followers** in RegSet, my first-author EMNLP 2022 paper and oral presentation [5]. Large, pre-trained LMs can solve some NLP tasks by conditioning their generations on natural language instructions for the task [6, 7]. However, the complexity of natural language makes difficult to predict what types of instructions may be beyond the grasp of Transformers. To solve this predicament, we propose a controllable proxy for studying instruction learning by studying LMs ability to learn interpret regular expressions and recognize their strings. Our experiments lead us to a number of intriguing hypotheses about what makes instruction learning hard, including evidence that even large transformers struggle with modular counting (e.g., determining whether something is even or odd). Subsequent work [8] has built on our formalization to obtain important theoretical results by using circuit complexity theory to formally characterize transformers ability to execute instructions in the

form of a circuit. In my PhD, I hope to continue developing both theoretical and empirical frameworks for understanding neural networks through formal languages and computation theory. For instance, perhaps we can prove theoretical upper bounds for what transformers can learn from instructions, or characterize the additional computational power afforded to transformers when they are allowed to generate multiple intermediate tokens before producing an output. On the empirical side, we can use formal languages to develop benchmarks to isolate and measure progress towards specific abilities in transformers that we hope to see in natural language settings. This approach takes advantage of the well studied properties of formal language and gives us fine-grained control over the data.

Understanding generalization in LMs My work on RegSet bridges to another area of research I hope to continue in during my PhD. Research has shown that neural models consistently to struggle with compositional generalization [9], which bodes poorly for the instruction following regime I studied with RegSet, where the space of task descriptions is large and highly compositional. I am specifically interested in studying and developing models and methods that tackle issues of generalization.

In a preprint currently under submission to ICLR 2023 [10], I develop a modular prompting method for recursively prompting large LMs in order to vastly improve length generalization on an algorithmic task compared to chain-of-thought prompting [].

While we uncovering shortcomings of neural networks as general instruction followers, I became interested in also evaluating neural networks as **general-purpose math reasoners**. To accomplish this, I led a team of 11 researchers in compiling a **comprehensive and diverse natural language math reasoning benchmark.** I introduce the benchmark, LĪLA [11], in my first-authored EMNLP 2022 paper and oral presentation. Current evaluation schemes fail to holistically evaluate general-purpose math reasoning skills in LMs because they are far too narrow in scope. As a result, they often overestimate the ability of particular models optimized for a single type of math reasoning. In our large scale effort we draw together a diverse set of mathematical tasks and unite them under a single benchmark of over 140K math problems. We provide valuable annotations for mathematical reasoning via program synthesis, where the LM has access to a Python interpreter. Our experiments show that multi-task learning, combined with augmenting the model with a Python interpreter, improves general-purpose math reasoning and the resulting model is an effective starting point for downstream fine-tuning. At the same time, our benchmark shows LMs, in their current form, are woefully deficient when it comes to math reasoning. I led and contributed to all aspects of the LĪLApaper, including collecting and annotating the datasets, the experimental design, running the experiments, and writing the paper. Our high quality, comprehensive evaluation will serve to unify evaluation and further the effort towards developing general-purpose math reasoning models.

Building off of my work on learning from instructions and general-purpose reasoning, I am currently interested in **learning how to leverage and control LMs beyond fine-tuning**. In our preprint [10], working closely with Tushar Khot and advised by Ashish Sabharwal, we develop methods for using large LMs as problem decomposers and modular sub-problem solvers to improve performance over other prompting techniques such as chain-of-thought []. In a current project, I am developing a decoding method for using a LM to self-generate optimal task-specific prompts relying only on inference-time sequence probability estimates from the model.

From these research experiences I have found that I am interested in pursuing the following types of problems as a PhD student:

- Problems that deal with generality, from compositional generalization, to general purpose reasoning.
- Improving our fundamental understanding of modern NLP methods.
- Leveraging our theoretical understanding of language and machine learning to uncover novel methods and paradigms for NLP.

Eventually, I hope to become a PI at an institution where I can continue to pursue my passion for mentorship and teaching. I have thoroughly enjoyed my teaching experiences as an undergrad and look forward to continuing to develop these skills as a PhD.

References

[1] Matthew Finlayson, Aaron Mueller, Stuart M. Shieber, Sebastian Gehrmann, Tal Linzen, and Yonatan Belinkov. Causal analysis of syntactic agreement mechanisms in neural language models. *ArXiv*, abs/2106.06087, 2021.

[2] Judea Pearl. Direct and indirect effects. *Probabilistic and Causal Inference*, 2001.

[3] Pangbo Ban, Yifan Jiang, Tianran Liu, and Shane Steinert-Threlkeld. Testing pre-trained language models' understanding of distributivity via causal mediation analysis. *ArXiv*, abs/2209.04761, 2022.

[4] Shauli Ravfogel, Grusha Prasad, Tal Linzen, and Yoav Goldberg. Counterfactual interventions reveal the causal effect of relative clause representations on agreement prediction. *ArXiv*, abs/2105.06965, 2021.

[5] Matthew Finlayson, Kyle Richardson, Ashish Sabharwal, and Peter Clark. What makes instruction learning hard? an investigation and a new challenge in a synthetic environment. *ArXiv*, abs/2204.09148, 2022.

[6] Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *ACL*, 2022.

[7] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *ICLR*, 2022.

[8] William Cooper Merrill and Ashish Sabharwal. Log-precision transformers are constant-depth uniform threshold circuits. *ArXiv*, abs/2207.00729, 2022.

[9] Brenden M. Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *ICML*, 2018.

[10] Tushar Khot, H. Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. Decomposed prompting: A modular approach for solving complex tasks. *ArXiv*, abs/2210.02406, 2022.

[11] Matthew Finlayson, Swaroop Mishra, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, and Ashwin Kalyan. LĪLA: A unified benchmark for mathematical reasoning. 2022.