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 or why 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.

As my first step towards this goal, I set out to **interpret how modern language models handle syntactic agreement** while an undergrad at Harvard, advised by Stuart Shieber and Yonatan Belinkov. In our ACL 2021 paper [1] (also advised by Sebastian Gehrmann and Tal Linzen) we apply causal analysis to understand how transformers handle number agreement between nouns and verbs. We find that transformers generally learn two distinct mechanisms for agreement, depending on whether the relevant tokens are adjacent or not. Subsequent work has used causal analysis for analyzing other linguistic phenomena such as distributivity [2], and the effect of relative clauses on agreement [3]. Strengthening our understanding of how (and whether) language models deal with well-studied linguistic phenomena can give insight into what inductive biases these models learn, and can guide research towards methods for improvement.

Excited by our findings, and starting as a pre-doctoral researcher at AI2, I decided to next explore how another field, formal language theory, can increase our understanding of the capabilities of transformer models. In particular, I found formal language to be an excellent fit as a test-bed for studying how well language models **generalize on highly compositional tasks**. Compositional generalization is one area in which neural methods have been shown to consistently underperform humans [4]. This becomes particularly problematic as **instruction following** emerges as a prominent paradigm [5, 6] for building general-purpose large language models. Instruction following is where a language model is expected to perform a novel task (one not seen in training) given only a description of the task. Since this paradigm has emerged relatively recently little is known about what kinds of tasks current models can be expected to learn. To evaluate this and provide a synthetic sandbox for future research, I introduced RegSet [7] in my EMNLP 2022 paper advised by Kyle Richardson, Ashish Sabharwal and Peter Clark. In our work we propose a highly controllable proxy for studying instruction learning by studying a language models ability to learn to interpret regular expressions. Regular expressions are succinct representations of regular languages, a well studied and highly compositional class of formal languages. As a result of our experiments, we develop a handful 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), less precise instructions, and tracking long contexts.

Subsequent work [8] has built on our ideas by using formal language theory and computation to formally characterize transformers ability to learn from instructions. I am currently working with Ashish Sabharwal to further this idea. My hope is that a better theoretical understanding of the limitations of transformers will inform our understanding of the architectural requirements for modeling language.

While we uncovering shortcomings of neural networks as general instruction followers, I became interested in also evaluating neural networks as **general-purpose math reasoners**. In a contrasting approach, I led the effort to compile a **comprehensive math reasoning benchmark** to evaluate language models' abilities on diverse math problem types, posed in natural language. I introduce the benchmark, LĪLA [9], in my EMNLP 2022 paper (advised by Ashwin Kalyan). Current evaluation schemes fail to comprehensively capture **general-purpose math reasoning skills** in language models 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 language model 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 language models, in their current form, are woefully deficient when it comes to math reasoning. My hope is that our high quality, comprehensive evaluation will serve as a point of reference as modeling approaches improve and become more general-purpose.

As a future direction, I am interested in **learning how to leverage and control language models beyond fine-tuning**. In our preprint [10], working closely with Tushar Khot and advised by Ashish Sabharwal, we develop methods for using large language models 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 language model 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 these types of problems as a PhD student:

- Tackling problems that deal with generality, from compositional generalization, to general purpose reasoning.
- Improving our fundamental understanding of our methods rather than merely improving performance.
- Pushing beyond the task-specific fine-tuning paradigm by exploring new ways to leverage existing general-purpose models.

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] 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.

[3] 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.

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

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

[6] 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.

[7] 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.

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

[9] 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.

[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.