ADITHYA BHASKAR

Princeton University, NJ, USA | Computer Science

+1 (609) 721-0839
github.com/testzer0
@ adithyabcse@gmail.com
@ ab4197@princeton.edu
35 Olden St, Princeton, NJ 08540
`adithyabh.github.io`

Education

2023-*Ongoing*	Ph.D. in Computer Science, Princeton University, USA *Advised by Prof. Danqi Chen* (Specialization : Natural Language Processing)	4.00/4.00
2019-23	Bachelor of Technology in Computer Science and Engineering (Honors), IIT Bombay, India	9.67/10.00
2017-19	High School, Central Board of Secondary Education, India	97.2/100.0
2017	Senior Secondary School, Central Board of Secondary Education, India	98.0/100.0

Publications

2025 Unintentional Unalignment : Likelihood Displacement in Direct Preference Optimization, *ICLR 2025*
Noam Razin, Sadhika Malladi, **Adithya Bhaskar**, Danqi Chen, Sanjeev Arora, and Boris Hanin

2024 Finding Transformer Circuits With Edge Pruning, *NeurIPS 2024* **(Spotlight)**
Adithya Bhaskar, Alexander Wettig, Dan Friedman, and Danqi Chen

2024 The Heuristic Core : Understanding Subnetwork Generalization in Pretrained Language Models, *ACL 2024* **(Oral)**
Adithya Bhaskar, Dan Friedman, and Danqi Chen

2023 Benchmarking and Improving Text-to-SQL Generation under Ambiguity, *EMNLP 2023*
Adithya Bhaskar*, Tushar Tomar*, Ashutosh Sathe, and Sunita Sarawagi

2023 Prompted Opinion Summarization with GPT-3.5, *ACL 2023 (Findings)*
Adithya Bhaskar, Alexander R. Fabbri, and Greg Durrett

2023 Performance Bounds for LASSO under Multiplicative Noise : Applications to Pooled RT-PCR Testing, *Signal Processing, Vol. 214*
Richeek Das, Aaron Jerry Ninan, **Adithya Bhaskar**, and Ajit Rajwade

Preprints

2025 Cache Me If You Can : How Many KVs Do You Need for Effective Long-Context LMs?, *arXiv preprint, arXiv:2506.17121*
Adithya Bhaskar*, Alexander Wettig*, Tianyu Gao, Yihe Dong, and Danqi Chen

2024 Continual Memorization of Factoids in Language Models, *arXiv preprint, arXiv:2411.07175*
Howard Chen, Jiayi Geng, **Adithya Bhaskar**, Dan Friedman, and Danqi Chen

2024 Improving Language Understanding from Screenshots, *arXiv preprint arXiv:2402.14073*
Tianyu Gao, Zirui Wang, **Adithya Bhaskar**, and Danqi Chen

Invited Talks

April 2024 The Heuristic Core : Understanding Subnetwork Generalization in Pretrained Language Models
Host : Mathew Monfort *Amazon AWS*

Scholastic Achievements

2024 Recipient of the **Hisashi and Masae Kobayashi *67 Fellowship**.

2023 Recipient of the **Thomas Dooie Class of 1974 Research Award**.

2019 **All India Rank 18** in JEE Advanced 2019 among **240 thousand** candidates.

2019 **All India Rank 114** in JEE Mains 2019 among **1.1 million** candidates.

2018 Secured a position in the **top 39 ranks** in the **Indian National Physics Olympiad** and was **invited to the Orientation-cum-Selection-Camp in Physics** held in May-June 2018.

2018 Secured a position in the **top 49 ranks** in the **Indian National Chemistry Olympiad** and was **invited to the Orientation-cum-Selection-Camp in Chemistry** held in May-June 2018.

2016 Among the **39 students** to clear the **Indian National Mathematical Olympiad**, becoming **one of the youngest** to **ever** be invited to the Orientation-cum-Selection-Camp in Mathematics **aged 14**.

Experience

UT Austin
Summer 2022

Research Intern, Natural Language Processing, USA
Advisor : Prof. Greg Durrett
Very Large Language Models for Multi-Document Summarization

- Developed metrics to measure **factuality**, **faithfulness** and **specificity** (whether it is correct, prefers major viewpoints, and is not too generic) for a summary of multi-document text such as hotel reviews.
- Utilized the above along with an n-gram abstractiveness metric to benchmark **GPT-3.5**, and showed that simple hierarchical summarization of large text performs poorly on faithfulness and specificity.
- Investigated various **pre-clustering** and **pre-summarization** methods and illustrated that pre-summarization with a pretrained keyword-based extractive model improves correctness, faithfulness and specificity, while only marginally affecting abstractiveness.

Uppsala University
Summer 2021

Research Intern, Formal Verification,
Advisor : Prof. Parosh Abdulla
Model Checking for Programs Running under the ARMv8 Memory Model

- Developed a **model** and **simulator** for programs running under the ARMv8 memory model, and demonstrated the equivalence of the model to the ARM specification using **7500+** litmus tests.
- Used **Context Bounded Model Checking** to perform **State Reachability Analysis** for programs under the ARMv8 memory model, achieving **up to an order of magnitude** of speedup over existing checkers.

Other Projects

Robust Models
Spring 2023

Bachelor's Project, Natural Language Processing, Guide : Prof. Sunita Sarawagi
Automated data augmentation for robustness.

- Demonstrated that training a Text-to-SQL model on partially masked (underspecified) inputs leads to **diversity** in the model outputs, including in **columns/tables, string literals, integers**, and **aggregates**.
- Filtered the outputs by model probabilities relative to the output with the unmasked question.
- Furnished questions for the generated queries via an SQL-to-Text model. Data augmentation with the pairs led to increases in accuracy on the SPIDER dataset and the robustness benchmark, Dr. SPIDER.
- The increases exceeded those obtained by augmenting with Dr. SPIDER style perturbations.

Group Testing
Fall 2022

R&D Project, Compressive Sensing, Guide : Prof. Ajit Rajwade
Applying compressive sensing to improve COVID-19 Group Testing.

- Proved the theoretical applicability of Compressive Sensing with Weighted LASSO for **any general noise model** with an asymptotically well-defined Moment-Generating-Function.
- Derived values of the optimal weights for the case of **Multiplicative Gaussian Noise** as in RT-PCR tests.
- Demonstrated improvements in **sensitivity**, **specificity**, **MCC** and **RMSE** by **Monte Carlo Simulations**.

C Decompiler
Fall 2020

Course Project, Software Systems, Guide : Prof. Amitabha Sanyal
Recovering Code From Compiled RTL

- Built a **decompiler** to convert **Register Transfer Language** to **C** for portability across architectures.
- Utilized **lex** and **bison** to parse source code in RTL and identify program elements like **assignments**, **basic arithmetic operations**, **conditional/looping constructs**, **function calls** and **memory accesses**.
- Performed **local & global data flow analysis** and **control flow analysis** to contextualize parsed code.

Teaching

Spring 2025	Graduate Teaching Assistant, *COS 484 : Natural Language Processing* Instructors : Danqi Chen, Vikram Ramaswamy, and Tri Dao	Princeton University
Fall 2024	Graduate Teaching Assistant, *COS 597R : Deep Dive into Large Language Models* Instructors : Danqi Chen, and Sanjeev Arora	Princeton University

Service

2025	NeurIPS 2025	Reviewer
2025	ICML 2025 MOSS Workshop	Reviewer
2024	NeurIPS 2024 ATTRIB Workshop	Reviewer