SOHIL LAL SHRESTHA

15705 Quorum Dr, Addison, TX 75001 | +1-817-557-7646 | sohil.shrestha@mavs.uta.edu | https://50417.github.io

EDUCATION

- **University of Texas, Arlington, Arlington, Texas**   2017 - December 2023 (Expected)
  PhD Candidate in Computer Science   CGPA: 4.00
- **Kathmandu University**, Dhulikhel, Nepal   2012 - 2016
  Bachelor of Engineering in Computer Engineering   CGPA: 3.85

AWARDS AND HONORS

WIPS Scholar	*2023*
UAI 2023 Student Scholarship	*2023*
ESEM NSF Travel Grant	*2023*
ESEM SIGSOFT CAPS Student Travel Grant	*2023*
ISSTA SIGSOFT CAPS Student Travel Grant	*2023*
Tapia Scholarship	*2023*
UT Arlington Dissertation Fellowship	*2023*
CodePath National Demo Day Top 5 iOS App Finalist[1]	*2021*
ICML Diversity & Inclusion Fellowhip	*2019*
SOCML Travel Grant	*2018*
Ncell App Camp Tourism Category Winner[2]	*2015*
Merit based Scholarship, Kathmandu University	*2013-2016*
Dean's list, School of Engineering, Kathmandu University	*2013-2016*

TECHNICAL SKILLS

- **Language:** Java (Expert), Python (Expert), MATLAB (Proficient), C/C++ (Intermediate), php
- **Databases & Frameworks:** SQL, Oracle, Vertica, REST, MVC, React, Django, Presto, Hack
- **Build/VCS Tools:** Mercurial (Intermediate), Maven, Git (Proficient), SVN
- **Tools & Libraries:** TensorFlow, Keras, Numpy, MatplotLib, Eclipse, Pytorch, VS Code, Jira, Anaconda, pandas, scipy, spacy, scikit-learn, Hive, Dataswarm, stream processing, docker
- **Cloud:** Amazon AWS, Google AutoML

INDUSTRY EXPERIENCE

Meta | Software Engineer Intern | Onsite   *May 2022 to August 2022*
- Designed and implemented dynamic step size algorithm of log barrier method based budget pacing system. Improved the budget pacer error rate from 1.5% to 0.007%
- Built new visualization tools and simulator to analyze issues relating to optimization of budget pacer.
- Designed and implemented an end-to-end A/B testing of budget pacing system.
- Learned 10+ internal tools, libraries and 2 additional unfamiliar programming languages.

Atos Syntel | R&D Machine Learning Intern | Remote   *May 2021 to August 2021*
- Prepared dataset and trained Google AutoML object detection model achieving around 90% precision and recall.
- Applied BERT based contextual spell correction on optical character recognition's output along with heuristic based correction. The approach produced 95% good sentences based on metric using Levenshtein ratio.
- Developed a web application using React and Django to visualize inference from deep learning models

Cotiviti Nepal Pvt. Ltd. | Intern/Associate Software Engineer   *July 2016 to May 2017*
- Performed root cause analysis on production issues, such as discrepancy in data, invalid sql queries, to reduce clients' downtime.
- Collaborated with senior member to reduce team's backlog by 50% involving Vertica SQL exception.

---

[1]among thousands of CodePath's students https://www.codepath.org/about/
[2]Nepal's annual national level competition

RESEARCH EXPERIENCE

Graduate Research Assistant, University of Texas at Arlington   *Fall 2017 - Present*
- Led a NSF funded project to test cyber-physical development toolchain using deep learning. My tool Deep-FuzzSL found a bug missed by the current state of the art fuzzer.
- Proposed and developed an approach to learn the specification of dataflow programming language (aka MATLAB/Simulink) using LSTM architecture to automatically generate Simulink models.
- Proposed a transfer learning approach leveraging a large pre-trained language model (GPT-2) for random Simulink model generation with high fidelity to training dataset and more bug finding capability than state of the art approach.
- Developed a fully automated tool to mine Simulink models from open source repositories. The tool alleviated non-trivial overhead of mining the repositories to sample Simulink models from open source projects
- Curated a dataset of Simulink models called SLNET which is 8 times larger than previous collections. Extracted and analyzed the Simulink model metrics to visualize modeling practices useful for future studies and tool development.

Publications
- **Shrestha, S. L.**, Boll, A., Kehrer T., Csallner, C. (2023, Oct).ScoutSL: An open-source Simulink search engine. MODELS
- **Shrestha, S. L.**, Boll, A., Chowdhury, S. A., Kehrer T., Csallner, C. (2023, Oct). EvoSL: A large open-source corpus of changes in Simulink models & projects. MODELS
- **Shrestha, S. L.**, Chowdhury, S. A., Csallner, C. (2023, May). Replicability Study: Corpora For Understanding Simulink Models & Projects. ESEM
- **Shrestha, S. L.**, Chowdhury, S. A., Csallner, C. (2022, May). SLNET: A Redistributable Corpus of 3rd-party Simulink Models. In Proc. 19th International Conference on Mining Software Repositories (pp. 237-241).
- **Shrestha, S. L.**, Csallner, C. (2021, June). SLGPT: Using Transfer Learning to Directly Generate Simulink Model Files and Find Bugs in the Simulink Toolchain. In Proc.2 5th International Conference on Evaluation and Assessment in Software Engineering (pp. 260-265).
- **Shrestha, S. L.** (2020, June). Automatic Generation of Simulink Models to Find Bugs in a Cyber-Physical System Tool Chain Using Deep Learning. In Proc. ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings (pp. 110-112).
- Chowdhury, S. A., **Shrestha, S. L.**, Johnson, T. T., Csallner, C. (2020, June). SLEMI: Equivalence Modulo Input (EMI) Based Mutation of CPS Models for Finding Compiler Bugs in Simulink. In Proc. ACM/IEEE 42nd International Conference on Software Engineering (pp. 335-346).
- Chowdhury, S. A., **Shrestha, S. L.**, Johnson, T. T., Csallner, C. (2020, June). SLEMI: Finding Simulink Compiler Bugs through Equivalence Modulo Input (EMI). In Proc. ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings (pp. 1-4).
- **Shrestha, S. L.**, Chowdhury, S. A., Csallner, C. (2020, May). DeepFuzzSL: Generating Models with Deep Learning to Find Bugs in the Simulink toolchain. In 2nd Workshop on Testing for Deep Learning and Deep Learning for Testing.
- **Shrestha, S. L.**, Panda, S., Csallner, C. (2018, May). Complementing Machine Learning Classifiers via Dynamic Symbolic Execution: " Human vs. Bot generated" tweets. In Proc. 6th International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering (pp. 15-20).

TEACHING EXPERIENCE

Graduate Teaching Assistant, University of Texas at Arlington   *Fall 2018, Spring/Summer 2019*
- Taught Algorithms and Database concepts to over 45 students each semester by solving practice problems and applications through recitations, office hours, and online discussions.
- Designed, tested and graded course assignments, projects and exams with professors and TAs.

PROJECTS

ScoutSL   *2023*
- Designed and implemented an end-to-end search engine for Simulink models, mining data from popular code hosting sites. The tool support search metrics not supported by other available search engines.
- The project is open-sourced at https://github.com/50417/ScoutSL

SLNET   *2022*
- Built a tool to automatically mine Simulink models from open source repositories.
- Extracted and analyzed SLNET's model metrics to visualize modeling practices and developed the largest corpus of Simulink models for future studies.
- The project is open-sourced at https://zenodo.org/record/5259648#.ZDRTHtLMIYs

WalletBud   *2021*
- Designed an iOS App that helps track expenses of an individual to better manage cashflows.
- Designed and implemented the business logic of the app using Parse Datastore, Back4App, Swift
- Nominated as one of the top 5 iOS app among hundreds of submissions. Presented the app in CodePath National Demo Day.
- The project is open-sourced at https://github.com/CodePathGroupOne/BudgetApp

SLGPT   *2021*
- Adapted Simulink model files to OpenAI's widely used GPT-2 language model to learn the structure of these Simulink model files.
- Designed and implemented the SLGPT tool that produced Simulink models that are both more similar to open-source models than its closest competitor, DeepFuzzSL, and found a super-set of the Simulink development toolchain bugs found by DeepFuzzSL.
- The project is open-sourced at https://github.com/50417/SLGPT

DeepFuzzSL   *2020*
- Proposed and developed an approach to learn specification of dataflow programming language (aka MAT-LAB/Simulink) using LSTM architecture to automatically generate Simulink models.
- DeepFuzzSL consistently generated over 90% valid Simulink models and found 2 confirmed bugs.
- The project is open-sourced at https://github.com/50417/DeepFuzzSL

VOLUNTEER

Student Volunteer at SIGMOD 2018 https://sigmod2018.org	*2018*
Student Volunteer at SC 2018 https://sc18.supercomputing.org/index.html	*2018*
Student Volunteer at NeurIPS 2018 https://nips.cc/Conferences/2018	*2018*
Student Volunteer at ICSE 2020 https://conf.researchr.org/home/icse-2020	*2020*
Student Volunteer at ISSTA 2023 https://conf.researchr.org/home/issta-2023	*2023*
Student Volunteer at PEARC 2023 https://pearc.acm.org/pearc23/	*2023*