

# Arash Tashakori

Halifax, NS | [Arash.Tashakori@dal.ca](mailto:Arash.Tashakori@dal.ca) | [LinkedIn](#) | [GitHub](#) | [Website](#)

## Education

---

<b>Bachelor of Computer Science</b>	Dalhousie University	GPA: <b>3.85/4.30</b>	2020-2025
-------------------------------------	----------------------	-----------------------	-----------

- Relevant Courses: Software Engineering, Data Mining and Warehousing, Foundations of Machine Learning, Data Structures and Algorithms, Algorithms Design and Analysis, Intro to Database Systems, Web Design and Development, Designing User Interfaces, Network Computing, Systems Programming, Principles of Programming Languages, Workplace Communications

## Skills

- 
- **Programming Languages:** Java, Python, HTML, CSS, JavaScript, C, C#, SQL
  - **Libraries:** React.js, Spring Boot, Node.js, Bootstrap, JUnit, PyTorch, NumPy, Pandas, Keras, Scikit-Learn, Unity
  - **Tools:** Git, Figma, Linux, Microsoft Office, MySQL Workbench, Visual Studio, Google Colab
  - **Interpersonal:** Teamwork, Presentation, Communication, Active Learning, Adaptability
  - **Other Relevant Skills:** Algorithm Design and Analysis, Database Design, User Interface Design, Object Oriented Programming, Test-Driven Development, Software Testing, Agile Practices, Machine Learning, Data Mining, Natural Language Processing, Generative AI, Web Development

## Relevant Experience

---

<b>Generative AI Extern</b>	Cognizant	06/2024 - 08/2024
		Toronto, Ontario

- Completed two projects focused on deep learning and natural language processing for Cognizant. (Python, NumPy, PyTorch, Pandas, PEFT)
- Utilized, tested and compared three pre-trained Convolutional Neural Network (CNN) models for animal image classification, evaluating and optimizing their performance based on accuracy and computational efficiency.
- Applied lightweight fine-tuning techniques to a GPT-2 model using Hugging Face's PEFT library, enhancing sequence classification accuracy from 36% to 59%.
- Acquired hands-on experience with Python, deep learning frameworks (i.e., PyTorch and NumPy), generative AI, and model optimization for image and text classification through completing the projects.
- Proactively self-taught multiple Python libraries for the projects, showcasing strong determination and quick learning abilities.

---

<b>Software Developer Intern</b>	AloDoctor	05/2022 - 08/2022
		Tehran, Iran

- Worked as a software developer intern, focusing on both back-end and front-end development to enhance the website's functionality and user interface. (HTML, CSS, JavaScript, Node.js, SQL)
- Collaborated with a team of 4 web developers, utilizing Agile methodology to implement new features, such as a physician rating system, patient login, tracking of favourite doctors from interactions, user recommendation forms, searching, filtering, and improved appointment booking.
- Enhanced the user interface, responsiveness and overall user experience of the website with a more intuitive and user-friendly design, receiving positive feedback from the users and the team.

**Web Developer Intern**      Sharif University of Technology

06/2019 - 09/2019

Tehran, Iran

- Worked on developing and maintaining a responsive and interactive website for Dr. Jawad A. Salehi, a professor at Sharif University of Technology. (HTML, CSS, JavaScript)
- Gathered the client's requirements to design the user interface to best suit the client's needs.
- Gained valuable skills in web development, time management, and client communication through developing the website in a timely manner.
- Completed the development and ongoing maintenance of the website successfully, ensuring it met the client's expectations and requirements.

## Projects

---

**Boardify - a collaborative list-making app**      Dalhousie University      06/2023 - 08/2023

- Developed a Trello-like web-based collaborative list-sharing app in collaboration with 2 teammates, using React.js, Spring Boot, MySQL and Git.
- Designed a relational database to organize and store project-related data.
- Employed Test-Driven Development (TDD) practices and wrote test cases before developing the backend, achieving over 70% testing line coverage.
- Utilized Git for collaborative development and ensured timely completion of each milestone and final delivery.
- Implemented key features such as user registration and authentication, creating project groups and task lists, managing to-do items, and filtering tasks by status and due date.

**Java Scheme Interpreter**      Dalhousie University

05/2024 - 06/2024

- Developed a Scheme language interpreter in Java, implementing core Scheme functionalities such as lambda expressions, conditionals, list manipulation, and logic and arithmetic operations.
- Applied object-oriented programming to enhance code modularity and maintainability, following a test-driven development process with rigorous unit and integration testing for quality assurance.
- Designed and implemented a parser, tokenizer, and evaluator for Scheme expressions, including handling recursive evaluation of nested expressions and lists.

**Simulation of Stochastic Hurricane Tracks**      Collaborative Project

09/2023 - 02/2024

- Collaborated with [Dr. Siamak Daneshvaran](#) on developing a simulation of stochastic hurricane tracks in the Atlantic Ocean.
- Utilized deep learning algorithms to simulate hurricane tracks and atmospheric pressure using Python and its libraries.
- Used 100 years of historical hurricane data from NOAA for training the model.

**Data Mining and Analysis on Multiple Datasets**      Dalhousie University

02/2024 - 04/2024

- Conducted data preprocessing and warehousing on COVID-19 data for Nova Scotia, focusing on data cleaning and structuring for analysis.
- Applied classification models such as Naive Bayes, Support Vector Machines, and Neural Networks to predict outcomes in various datasets.
- Demonstrated proficiency in Python and its libraries, including NumPy, Pandas, and Scikit-learn, for data manipulation, analysis, and machine learning implementation.
- Implemented the k-means clustering algorithm and compared its results with Scikit-learn's implementation on multiple datasets.

For more projects, please check my [GitHub page](#).