RAGHAV UPADHYAY

West Lafayette, IN · (520) 343-6432 · raghav0408upadhyay@gmail.com
raghavupadhyay.org · linkedin.com/in/raghavupadhya04 · github.com/raghav-upadhyay2002

RESEARCH INTERESTS

Evaluation and reliability of large language models: hallucination detection, LLM-as-a-judge methodology, and the stability of model behavior under prompt, linguistic, and scale perturbations. Uncertainty quantification and explainability for deep learning. LLM-based social simulation.

EDUCATION

University of Arizona **Apr 2024 – May 2026**

M.S. in Data Science · GPA 3.78 / 4.00 · Conferred May 2026 · Tucson, AZ

Coursework: Machine Learning, Applied Natural Language Processing, Computational Linguistics, Data Mining and Discovery, Foundations of Data Science, Data Analysis and Visualization, Data Ethics.

SRM Institute of Science and Technology, India **2020 – 2024**

B.Tech in Computer Science and Engineering (Software Engineering) · CGPA 8.51 / 10.00

Coursework: Calculus and Linear Algebra, Advanced Calculus and Complex Analysis, Probability and Queueing Theory, Discrete Mathematics, Design of Experiments, Data Structures and Algorithms, Design and Analysis of Algorithms, Artificial Intelligence, Formal Languages and Automata, Compiler Design, Operating Systems, Database Management Systems.

RESEARCH EXPERIENCE

Research Assistant — Vision Systems and Intelligence (VSI) Lab, University of Arizona **Aug 2026 – Present**

Advisor: Prof. Eungjoo Lee, School of Electrical, Computing, and Software Engineering · Target venue: IEEE Robotics and Automation Letters (RA-L)

- Investigating whether a vision-only mobile robot can recognize when its visual evidence is unreliable for navigation and adapt its behavior before an unsafe or incorrect decision, generalizing a failure mode found in the Vision-Based Virtual Maze Navigator project.
- Designing a multi-environment Webots evaluation benchmark (10-20 maze layouts, randomized start/target positions) under six visual perturbation types (illumination, blur, occlusion, noise, contrast/brightness, field of view), informed by the RobustNav (ICCV 2021) evaluation protocol.
- Building an automated data-logging pipeline capturing per-frame camera imagery, perception outputs, actions, pose, and collision/success outcomes, with simulator ground truth reserved for training labels and evaluation rather than exposed to the navigation policy at test time.
- Establishing baseline comparisons (absolute-threshold Canny/HSV controller vs. a relative-threshold variant vs. a lightweight learned CNN policy) to characterize failure modes under visual corruption, as a first step toward a reliability-aware act / re-observe / escalate navigation framework.

PUBLICATIONS

[1] **Raghav Upadhyay**, Sai Hemanth Kilaru, Sriram Theerdh Manikyala, Sri Sai Kumar Ramavath, Srivika Nunavathu, Dalal Alharthi. *When Do LLMs Generate Realistic Social Networks? A Multi-Dimensional Study of Culture, Language, Scale, and Method.* arXiv:2605.12898 [cs.SI], May 2026. Under review at NeurIPS 2026.

- Formalized four LLM tie-formation mechanisms (sequential, global, local, iterative) as distinct conditional distributions over edge sets, grounded in homophily and structural balance theory.
- Generated 192 verified directed networks from 50 demographically grounded personas across four cultural contexts, four prompt languages, three GPT-4.1 variants, and four prompting architectures, two seeds per condition.
- Found that prompt architecture functions as a substantive sociological variable: political affiliation dominates tie formation under three of four methods, while the global method substitutes age — indicating that homophily findings reported under a single prompting scheme may be design artifacts.
- Established a model-scale divergence ranking that reproduced across all three studies, with the smallest variant differing qualitatively rather than only in noise.

RESEARCH AND TECHNICAL PROJECTS

AskMyDocs — Production RAG System with CI-Gated Evaluation **Apr 2026**

Python · LLaMA (Groq API) · ChromaDB · BM25 · Cross-Encoder · GitHub Actions · Gradio · Hugging Face Spaces

Live: huggingface.co/spaces/raghavupadhyay/askmydocs · Code: github.com/raghav-upadhyay2002/AskMyDocs

- Built and publicly deployed an end-to-end retrieval-augmented QA system answering questions over user-uploaded PDFs with grounded, citation-tagged responses, architected as swappable modules (loader, chunker, embedder, vectorstore, reranker, LLM, prompts) to support component-level ablation.

- Engineered hybrid retrieval combining ChromaDB vector similarity (60%) with BM25 keyword scoring (40%), re-scoring top-10 candidates to top-3 with an ms-marco-MiniLM-L-6-v2 cross-encoder reranker.
- Implemented hallucination detection flagging any answer lacking inline citations, with versioned prompt templates (default / strict / concise) selectable per query.
- Built an automated LLM-as-a-judge evaluation harness scoring faithfulness, relevance, and citation rate, wired into a GitHub Actions CI pipeline that fails the build below faithfulness 0.70, relevance 0.70, or citation rate 0.80.

Vision-Based Virtual Maze Navigator — Camera-Only Autonomous Robot NavigationAug 2026

Python · OpenCV · NumPy · Webots (e-puck) · Canny Edge Detection · HSV Detection · Reactive Control

Code: github.com/raghav-upadhyay2002/Vision-Based-Virtual-Maze-Navigator

- Built an autonomous e-puck robot that navigates an unmapped 3D maze and reaches a target in Webots from camera pixels only — no odometry, no map, and no distance sensors in the control loop — fusing three perception streams (three-zone Canny edge density, near-field brightness, HSV target detection) into a single obstacle-avoidance-prioritized control policy.
- Identified a counter-intuitive design principle — wall proximity registers as edge density collapsing toward zero rather than spiking — and built the front-camera wall detector around it, which then survived a blur perturbation test unmodified.
- Diagnosed six distinct failure modes (temporal noise, spatial-aggregation errors, near/far confusion, absolute-vs-relative brightness thresholds) via per-frame CSV telemetry, fixing five on the main branch and isolating that the remaining fragility traced specifically to the one detector keyed on absolute intensity rather than relative structure.
- Ran controlled robustness experiments across illumination, blur, and occlusion, and specified a relative-brightness reformulation that keeps the wall detector invariant to a 50% illumination drop while preserving baseline navigation performance.

Now the seed project for ongoing faculty-advised research targeting IEEE RA-L — see Research Experience.

Uncertainty-Aware RUL Prediction with LSTM ArchitecturesApr 2026

PyTorch · TensorFlow · scikit-learn · NASA C-MAPSS · ipywidgets

Code: github.com/raghav-upadhyay2002/RUL-prediction-using-LSTM-for-Aircraft-Engine

- Designed and benchmarked four recurrent architectures (VanillaLSTM, StackedBiLSTM, LSTM-Attention, CNN-LSTM) for turbofan remaining-useful-life prediction on NASA C-MAPSS.
- Implemented Monte Carlo Dropout and deep ensembles for uncertainty quantification, using a mean − 2σ lower bound for risk-adjusted maintenance decisions rather than point estimates.
- Applied gradient-based attribution and temporal attention heatmaps for auditable per-prediction explanations; evaluated zero-shot cross-dataset transfer across C-MAPSS subsets.

PROFESSIONAL EXPERIENCE

Data Analyst Intern — Sudhir Mehrotra & AssociatesJan 2025 – Aug 2025

Bareilly, India

- Built Python ETL pipelines automating financial workflows, reducing manual processing effort by 30%.
- Developed a time-series cash-flow forecasting module improving estimation accuracy by 15% over baseline.
- Automated recurring Excel reporting with Python and VBA and designed structured data-reporting systems for audit and compliance teams, improving traceability across reviews.

TECHNICAL SKILLS

LLMs & Retrieval: OpenAI API, LLaMA via Groq, Hugging Face Transformers, ChromaDB, sentence-transformers, hybrid search (vector + BM25), cross-encoder reranking, LLM-as-a-judge evaluation, hallucination detection

ML & Deep Learning: PyTorch, TensorFlow, Keras, scikit-learn, LSTM and attention architectures, deep ensembles, Monte Carlo Dropout, uncertainty quantification, explainability (XAI)

Computer Vision & Robotics: OpenCV, Canny edge detection, HSV color detection, Webots (e-puck), reactive control policies, robustness testing

Languages & Tools: Python, SQL, R, Bash; pandas, NumPy, NetworkX, Matplotlib, Gradio; Git, GitHub Actions (CI/CD), AWS, Linux, LaTeX