Khandaker Abrar Nadib

Email: abrar.nadib@gmail.com | Github: AbrarNad |LinkedIn: abrar-nadib | Web: abrarnadib.github.io

Research Interests

I study how people interpret and disseminate charts and data in interactive, online environments. My work sits at the intersection of **Information Visualization**, **Human-Computer Interaction (HCI)**, and **human-centered data science**. My goal is to improve comprehension of complex data and decision-making through designing better visualization and storytelling systems.

Publications

ReVISit 2: A Full Experiment Life Cycle User Study Framework
Zach Cutler, Jack Wilburn, Hilson Shrestha, Yiren Ding, Brian Bollen, Khandaker Abrar Nadib, Tingying He, Andrew McNutt, Lane Harrison, and Alexander Lex.
IEEE Transactions on Visualization and Computer Graphics (Proceedings of IEEE VIS 2025, **Best Paper Award**), 2025. **DOI:** 10.48550/arXiv.2508.03876

Interaction Based Credibility Analysis of News on Facebook Using Machine Learning Methodologies
Sadia Sharmin, Sudipa Saha, Tasin Hoque, and Khandaker Abrar Nadib.
In *Proceedings of the 16th International Conference on Signal Image Technology & Internet based Systems (SITIS)*, 2022. **DOI:** 10.1109/SITIS57111.2022.00077

Research Experience

Guardrail Selection in Line Charts for Persuasive Visualizations 2025
Visualization Design Lab & KORE Lab, University of Utah *Salt Lake City, Utah*

- Lead author on a preregistered mixed-design crowd-sourced study of guardrail sampling strategies in persuasive time-series dashboards (COVID-19 cases and stock performance), evaluating trust, performance, and perceived contextual completeness.
- Built an interactive visualization study using the reVISit framework, implementing various guardrail techniques for contextualizing line-chart comparisons and integrated a logging pipeline that captures provenance graphs, interaction events with Supabase/Firebase-backed storage for replay-based analysis of 500+ crowdsourced participants.
- **Tech:** TypeScript, React, Vite, Mantine UI, D3.js, reVISit, Supabase, Firebase.
- **Status:** Under Review in EuroVis 2026.

Ranking Visualizations of Correlation (ReVISit Replication Study) 2024
ReVISit 2 Replication Project *University of Utah*

- Designed and implemented a replication of Harrison et al.'s correlation JND study using reVISit's dynamic sequencing and staircase designs.
- Configured OSF preregistration and study materials, and helped manage crowdsourced data collection across multiple visualization conditions (scatterplots, PCPs, hexbins, heatmaps).
- **Status:** Published, also working in a first-authored short paper.

News Credibility Analysis on Facebook using User Interactions 2021 – 2022
Bangladesh University of Engineering and Technology (BUET) *Dhaka, Bangladesh*

- Proposed and evaluated an interaction-based approach for fake news detection on Facebook, using engagement signals rather than language features.
- Trained machine learning models to classify the authenticity of public posts; showed improved performance over content- and NLP-based baselines and language independence.
- **Tech:** Crowdtangle, scikit-learn, pandas, matplotlib.
- **Status:** Published

Education

University of Utah — Salt Lake City, Utah
Doctor of Philosophy in Computer Science — *August 2024 – Present*

Bangladesh University of Engineering and Technology (BUET) — Dhaka, Bangladesh
Bachelor of Science in Computer Science and Engineering — *Feb 2017 – May,2022*

Work Experience

Graduate Research Assistant — January 2025 – Present
University of Utah, Salt Lake City
Visualization Design Lab, KORE Lab

Graduate Research Fellow — August 2024 – December 2025
University of Utah, Salt Lake City
Visualization Design Lab, KORE Lab

Software Engineer — May 2022 – July 2024
Optimizely, Dhaka

Digital Asset Management (DAM) — *November 2022 – July 2024*

- Worked on a microservice-based large-scale Digital Asset Management platform (Google Drive–like system) supporting asset storage, organization, collaboration, and analytics for enterprise customers.
- Implemented Brand Template feature, which lets users create a Template for their brand and define Placeholders that other collaborators can edit.
- Implemented efficient Searching, Filtering, and Navigation within DAM Collection folders (a group of user-defined Assets, including Asset folders).
- Implemented various asset-specific features like meta information, asset relations, and bulk operations, which enhanced the user's ability to handle assets.
- Implemented breadcrumbs and routing in the DAM Library for navigation fluidity.
- Implemented various user activity tracking for analytics to gain useful insights.
- Improved backend and frontend components, focusing on performance, reliability, accessibility, and code quality, including refactoring error-prone APIs and adding/updating automated tests.
- Upgraded and integrated OpenAI models for AI content generation.
- Handled user roles and privileges for access to various features.
- Worked extensively with databases and data migrations, including schema migration, backpopulation, query optimization, and codebase migration.
- **Technologies:** Python, Flask, JavaScript, TypeScript, React.js, MySQL, MongoDB, Alembic, Celery, Elasticsearch

Asset Renditions (AR) — *May 2022 – October 2022*

- Implemented and maintained an Asset Rendition microservice. This feature allows users to pre-define "Rendition types", using which whenever users to define rendition types that automatically generate image and video variants on upload.
- Implemented logging schemes by combining multiple services to enable users and developers to diagnose and debug errors.
- Built three backend services for automated asset rendition generation, implementing image and video processing pipelines driven by predefined rendition specifications.
- Implemented stateless generators to scale horizontally and integrated asynchronous messaging for decoupling and scaling, for efficiency.
- Integrated the Rendition Service with the local development environment for developers.
- **Technologies:** Python, FastAPI, MySQL, PostgreSQL, Docker, Kubernetes, Message Queue

Projects

Interactive Guardrail-Integrated Line Chart Platform 2024–2025

- Implemented interactive dashboards used as stimuli in the guardrail selection study, including guardrail sampling logic, randomization.
- **Technologies:** TypeScript, React 18, Vite, Mantine UI, Redux Toolkit, D3.js, Vega/Vega-Lite, Arquero, reVISit, Supabase, Firebase.

Correlation JND Replication Experiments (ReVISit Framework) 2024

- Built a staircase-style experiment to measure just-noticeable differences in correlation across multiple visualization types using reVISit's dynamic sequencing.
- **Technologies:** TypeScript, React,D3.js, reVISit, Python (pandas, NumPy, SciPy, statsmodels), firebase.

HappyVis: Visualizing Happiness Around the Globe 2024

- Designed a web-based visualization dashboard for the World Happiness Report dataset, enabling exploration of global happiness scores and contributing factors across countries and years.
- Implemented linked views including an interactive choropleth map, trend line charts, and comparative views to analyze relationships between happiness, GDP per capita, life expectancy, and social support.
- **Technologies:** JavaScript, D3.js.

Awards and Honors

Best Paper Award, IEEE VIS 2025 2025
for "ReVISit 2: A Full Experiment Life Cycle User Study Framework"

Optimizely SPOT Awards (July & October) 2023
Two peer-nominated SPOT awards recognizing problem solving, team contribution, and performance

Board Merit Scholarships (SSC & HSC) 2014, 2016
Education Board Scholarships; ranked 6^{th} (male) in Dhaka Board in HSC

Technical Skills

Research Methods: Online controlled experiments, interviewing, qualitative coding (open-ended response analysis, thematic coding)
Analysis: Mixed-effects models, regression, hypothesis testing; Python (pandas, NumPy, SciPy, statsmodels, matplotlib, scikit-learn, PyTorch
Visualization & Frontend: D3.js, Vega, Vega-Lite, React.js, TypeScript, Mantine UI, Redux Toolkit, Vite
Experiment Platforms & Storage: Supabase, Firebase, reVISit, Qualtrics
Languages: Python, Java, C/C++, SQL, PL/SQL
Databases: MySQL, ElasticSearch, PostgreSQL, MongoDB, Oracle
Frameworks: Flask, FastAPI, Node.js, Bootstrap, Celery, Alembic
Tools/Software: Git, Docker, Jupyter, Playwright
Libraries: pandas, NumPy, Arquero (JS), Keras, Matplotlib, SciPy, scikit-learn, PyTorch, Enzyme (JS), React Testing Library
Scripting/Markup/Serialization: Bash, TCL, LaTeX, YAML, HTML, JSON