Nanye (Willis) Ma

Email Website Linkedin Github

Education

New York University **New York, NY**

Honors B.A. in Mathematics; B.A. in Computer Science; 3.9/4.0 GPA *September 2020 - Expected May 2024*

- **Selected Coursework:**
  - **Graduate Level**: Machine Learning, Computer Vision, Probability Theory I, Portfolio and Risk Management, Programming Language.
  - **Undergrad Level**: Honors Analysis, Honors Algebra, Honors Theory of Probability, Honors Numerical Analysis. Operating System, Basic Algorithm, Applied Internet Technology.

Publications / Preprints

- **N. Ma**, M. Goldstein, M. S. Albergo, N. M. Boffi, E. Vanden-Eijnden, S. Xie. SiT: Exploring Flow and Diffusion-based Generative Models with Scalable Interpolant Transformers. *In submission to CVPR 2024.*

Research Experiences

SiT: Exploring Flow and Diffusion-based Generative Models with Scalable Interpolant Transformers

May 2023 - Present

- Proposed Scalable Interpolant Transformers (SiT), a novel family of generative models built on dynamical transport, effectively merging Flow and Diffusion-based models, and engineered the models in `JAX` and `Flax`.
- Conducted a study on SiT design space in class-conditional ImageNet256, compared various component combinations against the baseline DiT model, and identified configurations that yielded a **19%** improvement in FID score.
- Innovated a novel backward SDE construction that allows customized diffusion coefficients, enhancing flexibility in the diffusion sampling process while also leading to an extra **5%** performance boost.
- Without guidance, SiT models matched DiT performance using only **50%** of the computational resources and exceeded DiT's FID score by **22%** under similar budgets.
- The largest SiT model achieved a FID score of **2.06** with guidance, surpassing the DiT model's score of **2.27**.

Text-to-Image Scalable Diffusion Models with Transformers

Feb 2023 - May 2023

- Integrated a pre-trained DistilBERT encoder with Diffusion Transformers (DiT) in `PyTorch` to enable rapid and high-quality text embedding.
- Achieved a **78%** reduction in memory consumption by gradient checkpointing and pre-extracted Variational Autoencoder(VAE) image features, enhancing training speed to 0.32 iters/sec on a single A100 GPU.
- Fine-tuned text-to-image DiT from pre-trained class-conditional DiT checkpoint by unfreezing `Embedders` and inject randomly initialized weights to `adaLN` modules.
- Benchmarking on MS-COCO, the fine-tuned model attained a competitive FID score of **15.49** with the LDM-KL-8-G model with only 200K training steps.

Deep Marching Tetrahedra on Non-Watertight Models

Oct 2022 - Dec 2022

- Re-engineered Deep Marching Tetrahedra (DMTet) network to enable direct training on non-watertight 3D meshes from ShapeNet.
- Replaced the PVCNN Encoder with a fine-tuned Inverse Distance Weighted KD Tree, enabling detailed and precise spatial encoding of non-watertight models' point cloud.
- Enhanced the Surface Subdivision module with a learnable Signed Distance Field threshold to adapt to different surface topologies, extracting more refined surfaces.
- The optimized model outperformed the baseline DMTet in both L1 and L2 Chamfer Distance on non-watertight meshes, attaining best scores of **5.46** and **3.89**, respectively.

Modeling for Green GDP

Feb 2023 - Mar 2023

- Built the Predicted GDP model using the Cobb-Douglas function, incorporating policy rewards based on carbon neutrality and sustainability.
- Devised a Green GDP model by integrating the Predicted GDP model output with a fine-tuned decaying factor.
- Optimized Green GDP using Euler-Lagrangian conditions in `scipy`, and determined the optimized policy rewards by constructing a convex optimization problem with affine constraints in carbon emission and sustainability factors.
- The optimized model outperformed traditional GDP model, improving the growth rate by **7.1%** and reducing global CO2 emission by **14.9%** over a ten-year period.

Optimization for Road Cycling Power Distribution

Mar 2022 - Apr 2022

- Constructed a non-linear model based on existing data from top cyclists and achieved a deviation of less than 3.6% compared to the average podium finish times in each time trial.
- Implemented Genetic Algorithm with `scikit-learn` to find the optimal power distribution curve over all time trial courses, enabled the predicted finish time to decrease by an average of **9.3%**.

Projects

***Dream Diffusion*, Full Stack Development Project** **Jan 2023 - May 2023**

- Orchestrated the development of a full stack application using `Next.js`, designed specifically for recording dreams and writing dream journals with DallE backbone.
- Engineered session management from scratch to handle authentication and authorization, utilizing `Redis` as session store for efficient information retrieval.
- Designed robust authentication middleware using `Passport.js` and `JWT`-based strategies to bolster application security and safeguard user privacy.
- Optimized Serverless API architecture to support large-size (1024×1024) images transfers between frontend and backend, maintaining a low latency of up to 120ms for seamless user experience.

***HyperEX*, Software Development Internship Project** **May 2022 - Jan 2023**

- Played a pivotal role in the development team of HyperEx, an innovative online immersive social platform supporting hundreds of users in a shared virtual space.
- Designed a Finite State Machine library and Inverse Kinematics algorithms in JavaScript, enhancing the player's motion system, rag-doll system, and inventory system.
- Developed RESTful API endpoints using `express` and `Socket.IO` to facilitate communications between game engine and the application, resulting in seamless interactions between user interfaces and game renderings with a low latency of up to 50ms.
- Engineered a performance optimization module with `Draco` to ensure the platform's performance on outdated devices, and further increases the game's frame rate by over 300

***Weensy OS*, Operating System Development Project** **Sep 2021 - Nov 2021**

- Implemented a small OS in `Assembly`, `C` and `C++`, supporting 2 MB physical and 3MB virtual memory, one basic shell command ls, multithreaded kernel and userspace, paging and address allocation, and a FUSE file system.

Technical Skills

Coding Languages: Python, JavaScript, C++, Java, C, Standard ML, Scheme, Prolog, Assembly, Julia
Tools/Frameworks & Libraries: Github, Linux / JAX, Flax, Optax, PyTorch, Numpy, Scipy, Pandas, Node.js, React, Express, MongoDB, Next.js

Honors

Summer Undergraduate Research Experience (SURE) Fund
Dean's Undergraduate Research Fund
Dean's List for Academic Year 2020-2021, 2021-2022, 2022-2023
MCM Success Award 2022, 2023