

NAREN SENGODAN

INFORMATION SCIENCE & ENGINEERING

+917708173360

narensengodan@gmail.com

github.com/narensen

SKILLS:

Technical Skills

Programming Languages:

- C++
- Python
- Rust

Frameworks and Libraries:

- PyTorch
- TensorFlow
- Pandas
- Matplotlib
- Scikit-learn

Domain Expertise:

- Computer Vision
- Natural Language Processing (NLP)
- Reinforcement Learning
- Generative Models

Software Development:

- Git
- Version Control Systems

EDUCATION:

- 2023 - Present | JAIN University
- B.Tech - Electrical Engineering
- Grade - 8.15

EXPERIENCE:

Google Development Student Clubs AI/ML Facilitator (2023 - Present)

- **Technical Contributions:**
Developed and delivered hands-on sessions on topics like web development, machine learning, and cloud computing, impacting over 2000 students.
- **Community Engagement:**
Collaborated with industry professionals to host guest lectures and mentoring sessions, enhancing the practical skills of club members.

Courses:

- Generative Adversarial Networks (GANs) Specialization - DeepLearningAI
- State Estimation and
- Localization- UToronto
- Visual Perception - UToronto
- Python for Research - Harvard
- Computer Graphics - UCSD

RELEVANT PROJECTS:

Image Captioning using EfficientCBAM and BERT

- Developed an Image Captioning model using PyTorch, integrating a hybrid Efficientnet and CBAM as encoder with a Transformer-based decoder and attention mechanisms.
- Applying CBAM has enhanced the feature extraction capabilities of the layers and resulted in improved feature maps, increasing the mAP by 3%
- Achieved a 15% improvement mean Average Precision and feature extraction using the pretrained EfficientNet-based encoder compared to baseline CNNs.
- Enhanced caption quality with a Transformer-based decoder, resulting in a 7-point increase in BLEU score over SOTA models with similar amount of parameters
- Trained and evaluated on the COCO 2017 dataset, attaining a CIDEr score of 0.8, indicating high-quality caption generation.
- Optimized with CUDA support, reducing training time by approximately 50% compared to CPU-only training.
- Supports custom datasets for flexible application in various image captioning tasks.

Deformable Attention Transformer for Image Restoration

- Developed an advanced image processing method using a Deformable Attention Transformer to effectively remove moiré patterns from images.
- Improved visual clarity and overall image quality, achieving a 30% reduction in visual artifacts compared to traditional vision transformers (ViTs).
- Optimized resource usage, reducing computational costs by approximately 25% while maintaining a high level of detail preservation.
- Enhanced the efficiency of high-quality image processing, demonstrating the technique's practical applicability in resource-constrained environments.

Sparse Attention Mechanism with U-Net for Medical Image Segmentation

- By lowering the retrieval of unnecessary features, a pretrained U-Net model and a sparse attention mechanism were combined to increase processing speed by 20%.
- 92.39% accuracy while reducing needless feature processing, enhancing the overall performance of the model.
- About 15% less computational overhead, proving the effectiveness of sparse attention in deep learning applications in environments with limited resources.

Minimal implementation of BPE using Rust

- Developed a minimalistic implementation of the Byte Pair Encoding (BPE) algorithm using Rust, prioritizing performance and security.
- Achieved a substantial performance improvement, with the Rust implementation executing in 3.2 milliseconds, compared to 5.4 milliseconds in a Python-based implementation—a 40% reduction in execution time.
- Leveraged Rust's strong memory safety guarantees to prevent common vulnerabilities, making the implementation both faster and more secure.
- Demonstrated the effectiveness of Rust for high-performance text processing tasks

Contributions

- Developed and integrated the Quasi-Recurrent Neural Network (QRNN) module into the PyTorch open-source library.
- Implemented Visual Information Fidelity (VIF) metric into the PyTorch library, to measure the quality for Image Restoration tasks