COMPSCI 682, Fall 24 Project Spotlights

Day 1 – Dec 3, 2024

Umass, Amherst

Instructions

Speakers will have 2 mins to present their work

We will warn you at when 1 min, 30 sec, 0 sec remain

Must wrap up at 0

We will ask questions during grading

Thanks!

But first attendance

Presentation order

2	improving semi-supervised fine grained classification using curriculum learning	Ishita Chakravarthy (ichakravarth@umass.e Matthew James (matthewjames@umass.ec Mihir Thalanki (mthalanki@umass.edu)
3	comparitive analysis of modern image anomaly detection methods	Anish Gupta (anishgupta@umass.edu), Neel Joshi (nsjoshi@umass.edu)
4	domain adaptation for OCR models	Swathy Anand (swathyanand@umass.edu) Swetha Mohan (swethamohan@umass.edu
8	enhancing identity consistency in video generation	Haoyu Zhen (hzhen@umass.edu), Jiaben Chen (jiabenchen@umass.edu), Zixin Wang (zixinwang@umass.edu)
14	layer-selective rank reduction in VLM	Jeevana Karnuthala (jkarnuthala@umass.e Srujana Pillarichety (spillarichet@umass.ed Vishnu Vardhan Reddy Kandi (vkandi@uma
17	Improving fashion recommendation relevance via fit aware neural re-ranking	Harshitha Kolukuluru (hkolukuluru@umass. Sai Sreenivas Chintha (saisreenivas@umas Vishal G (vishalg@umass.edu)
19	Fairness in kNNs	Alexander Yeung (ayeung@umass.edu), Harold Thidemann (hthidemann@umass.e Shreyans Babel (sbabel@umass.edu)
22	Fair neural unlearning	Cameron Mattheson (cmattheson@umass. Riddho Ridwanul Haque (rhaque@umass.
26	LLM/uncertainity/reasoning (?)	Jenna Russell (jrussell@umass.edu), Rishanth Rajendhran (rrajendhran@umass. Vu Le (vdle@umass.edu)
30	mitigating bias in facial recognition using seldonion fraework	Frances Chang (fchang@umass.edu), Priya Balakrishnan (pbalakrishna@umass.e Roshini Sanikop (rsanikop@umass.edu)
32	evaluating effectiveness of GNN for node classification in sanitation infrastructure mapping	Nana Agyemang Sereboo (nagyemangser@ Sylvia Imanirakiza (simanirakiza@umass.e
34	Knowledge distillation for efficient neural network compression	Arav Adikesh Ramakrishnan (aravadikeshr(Siddhartha Jaiswal (sjaiswal@umass.edu)
40	llm-routing maximizing output quality while minimizing cost	Aparajith Raghuvir (araghuvir@umass.edu) Jeet Devendra Sharma (jeetdevendra@uma Tarun Gowda (tgowda@umass.edu)
43	predicting hospital readmission rates	Eniola Adegbegha (yadegbegha@umass.e Itir Sayar (isayar@umass.edu)

44	density estimation and crowd counting	Balachandra Devarangadi Sunil (bdevarang Rakshith Venkatesh (rakshith@umass.edu) Shantanu Todmal (stodmal@umass.edu)
45	optimizing video QA with heirarchical frame extraction and video-LLama	Reshma Ashok (rashok@umass.edu), Satya Srujana Pilli (spilli@umass.edu), Varshini Venkataraman (vvenkatarama@un
47	Compact Diffusion Policy for Robot Control	Jeongah Lee (jeongahlee@umass.edu), Shauna Choi (seohyunchoi@umass.edu), Suyoung Kang (suyoungkang@umass.edu)
50	Long Video Question Answering	Katie Zhang (kzhang@umass.edu), Vani Gupta (vanigupta@umass.edu)
51	Photofix: Restoring image quality through neural networks	Jeremy Hummel (jhummel@umass.edu), Nicole Kaldus (nkaldus@umass.edu)
58	compact diffusion models for cifar-10	Manan Abbott (mhabbott@umass.edu), Varun Jhaveri (vjhaveri@umass.edu)
61	Fine-Grained Brain Tumor Segmentation Using an Attention-Enhanced U-Net Model	Rishitha Guttapalle Mohan (rguttapallem@u Sreya Talasila (stalasila@umass.edu), Thashmitha B S (tbs@umass.edu)
63	emotion based style transfer	Aaryan Mohindru (amohindru@umass.edu) Antara Parmar (antaraparmar@umass.edu)
73	data augumentation with diffusion models using latent space exploration	Harshavardhan Reddy Rajanala (hrajanala(Vin Tyagi (vtyagi@umass.edu)
75	visual grounding in unseen domains	Jyothirmai Puram (jpuram@umass.edu), Keerthy Kaushik Dasoju (kdasoju@umass.e Supraj Bachawala (sbachawala@umass.ed
76	Neuro-symbollic solver	Frank Chiu (fchiu@umass.edu), James Barrett (jamesbarrett@umass.edu), Shamanth Kuthpadi Seethakantha (skuthpa
81	Deep learning for plant health monitoring	Greeshma Gudapati (ggudapati@umass.ed Meghana Sai Veligatla (mveligatla@umass.
86	occlusion aware module for 2d object tracking for autonomus vehicles	Alauddeen Shaikh (alauddeensha@umass. Amit Sarker (asarker@umass.edu), Roshini Pulishetty (rpulishetty@umass.edu)
89	Evaluating Open World Agents using Generative Models	Justin Clarke (jclarke@cs.umass.edu)

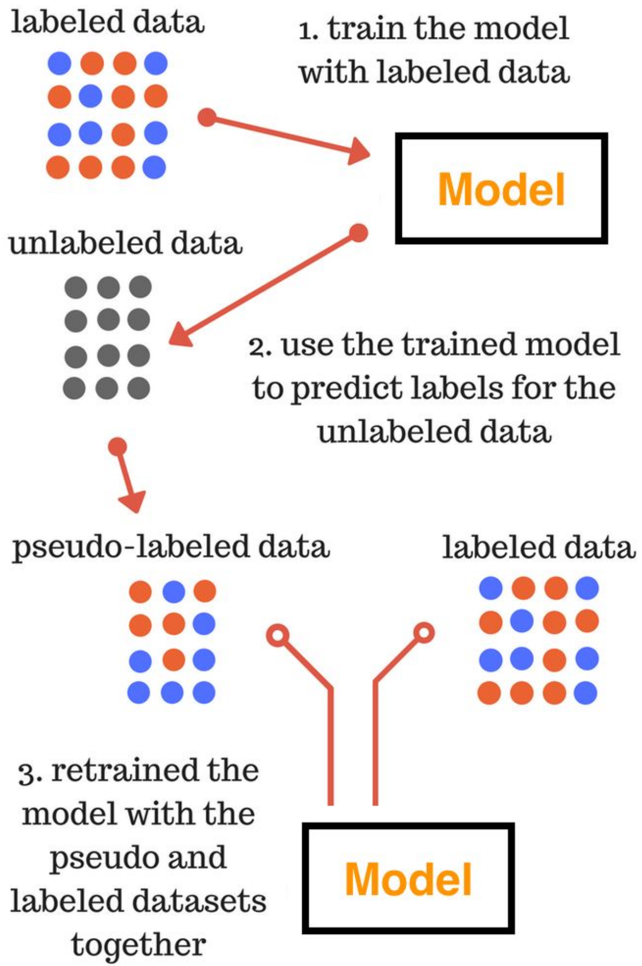
Semi-Supervised Learning for Fine-Grained Classification with Adaptive Pseudo-Labeling

Ishita Chakravarthy (ichakravarth@umass.edu),
Matthew James (matthewjames@umass.edu),
Mihir Thalanki (mthalanki@umass.edu)

Project #2

Pseudo-Labeling

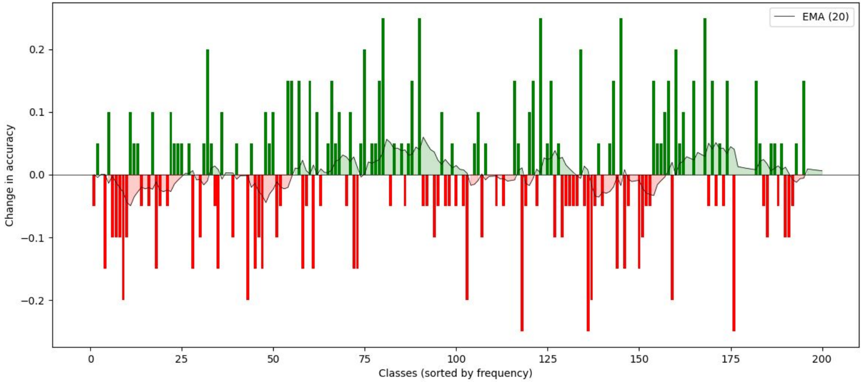
- Use a base model's confident predictions on unlabeled images as labels
- If the maximum probability of a class is greater than a threshold τ, create "pseudo-label"
- Uses a fixed threshold τ
    - for all iterations (as the model learns, threshold could be changed)
    - for all classes (does not leverage the inherent taxonomy information)

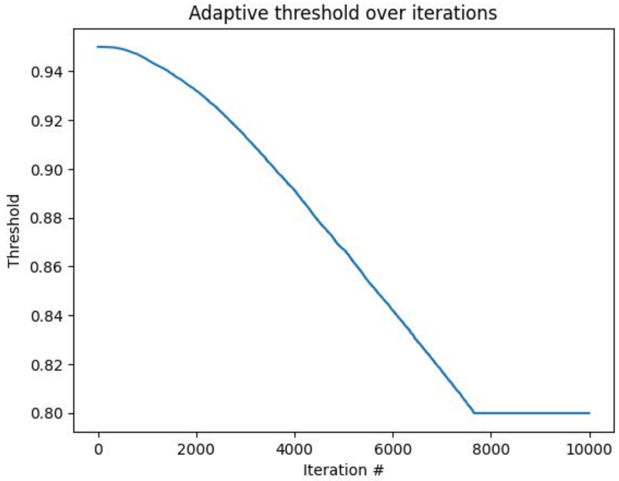


Source: https://datawhatnow.com/pseudo-labeling-semi-supervised-learning/

Adaptive global threshold

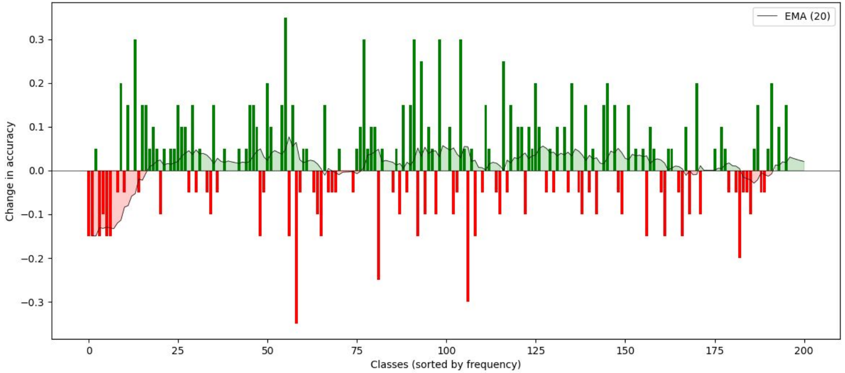
$$G_i = G_{i-1} - (1 - \lambda)(\Delta_{confidence})$$

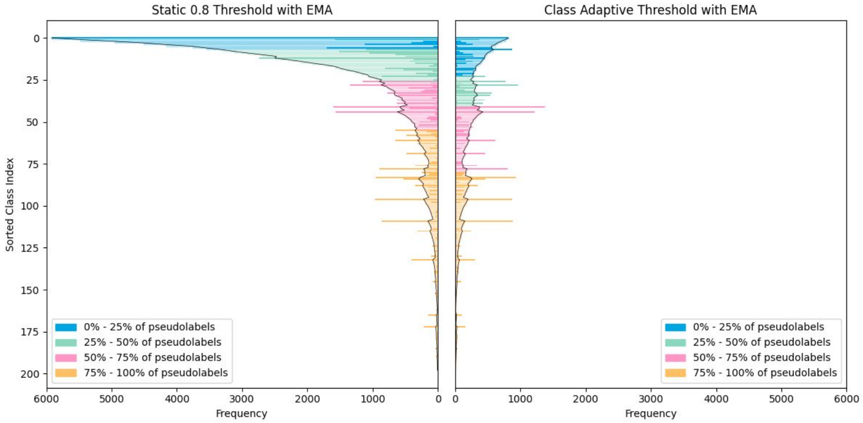




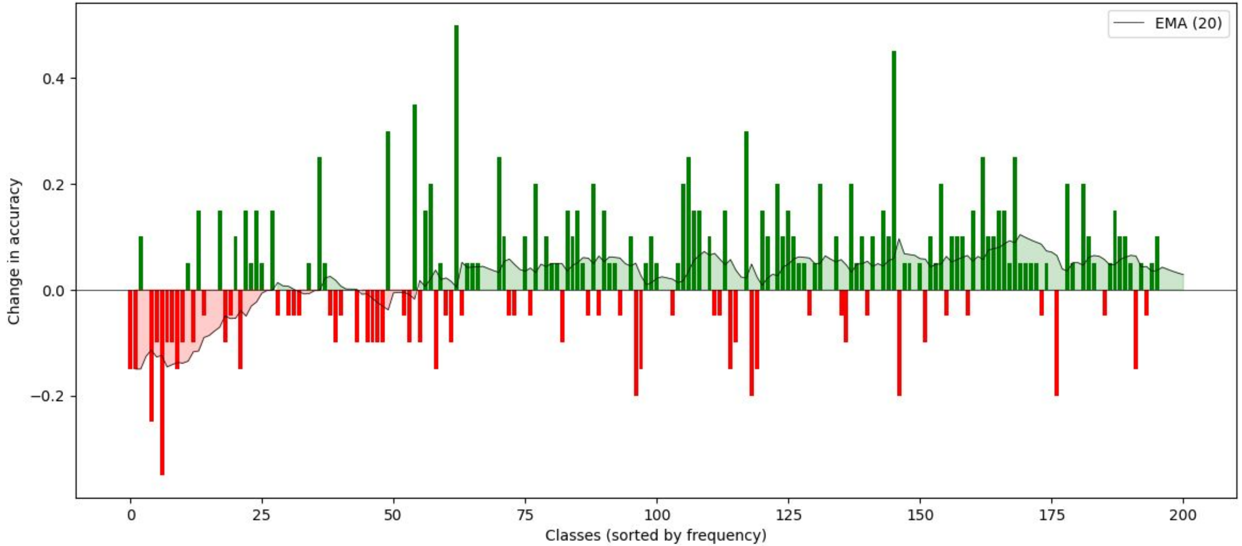
Adaptive class-specific threshold

$$C_i(c) = C_{i-1}(c) - (1 - \lambda)(\Delta_{confidence})$$





Final Model - Combined threshold



$$T_i(c) = \frac{2 * C_i(c) * G_i}{C_i(c) + G_i}$$

Method	Top1 Acc (%)	Top5 Acc (%)
Static (0.8)	40.45	71.65
Combined threshold	43.70	74.45

Iterative Class Based	Top1 Acc (%)	Top5 Acc (%)
ResNet18	41.77	73.10
ResNet50	47.55	80.17

Comparative analysis of modern image anomaly detection methods

Anish Gupta (anishgupta@umass.edu),
Neel Joshi (nsjoshi@umass.edu)

Project #3

Problem

Problem: Anomaly detection models are not good at generalizing to images that have real-world perturbations (eg: brightness/contrast shifts or lens distortion)

Solution:

1.) Create new datasets with perturbed images using image augmentation and measure performance on SOTA AD model (PatchCore)

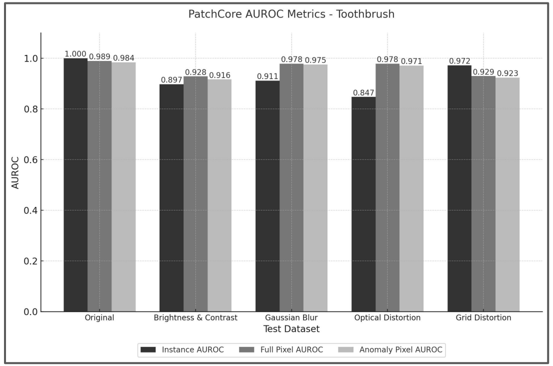
2.) Integrate TENT (test time entropy minimization) to optimize the model on confidence with aim to adapt to data it struggles on and increase performance

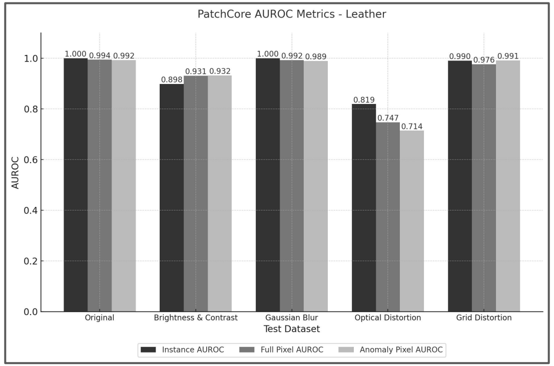
Dataset: We used a subset of the popular AD dataset MVTec-AD (leather and toothbrush objects) for our experiments

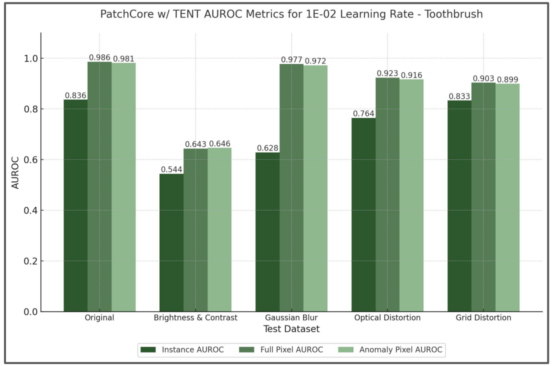
Original Image

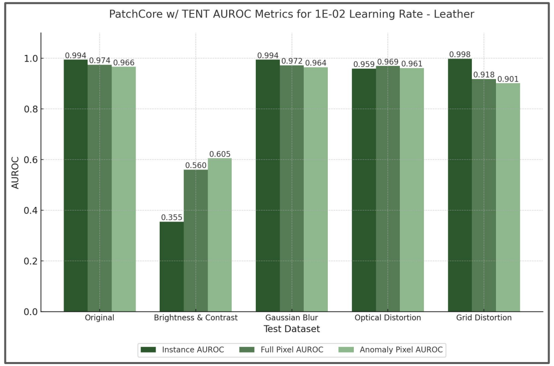
Random Brightness & Contrast

Results









TENT: Optimizing the model on confidence; entropy loss is calculated on the probability scores of inference on each image. Then, a backwards pass is ran only updating the batch normalization weights, gamma and beta.

Benefits:

- No need to retrain
- Low overhead since it only updates batchnorm variables

Conclusions

- The SOTA AD model, PatchCore, performed worse on the perturbed test datasets
  - Generally, we saw better performance for the perturbed images on the leather dataset (texture) vs the toothbrush dataset (item)
- Our addition of TENT to Patchcore failed to improve the performance of the model on the perturbed datasets
- Running these experiments on the rest of the MVTec-AD datasets and attempting to add TENT to other models like SimpleNet and EfficientAD might be some productive next steps

Domain adaptation for OCR models

Swathy Anand (swathyanand@umass.edu),
Swetha Mohan (swethamohan@umass.edu)

Project #4

Motivation

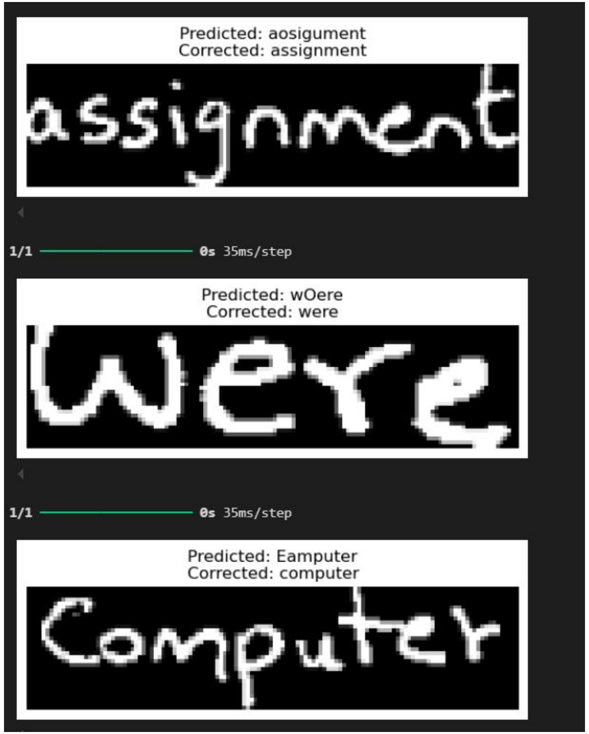
Optical Character Recognition (OCR) systems have excelled in recognizing printed text but encounter significant challenges with handwritten text. Models trained on specific datasets perform well within their domain but struggle to generalize effectively to diverse, real-world handwriting, especially in scenarios where labeled data is scarce or unavailable.

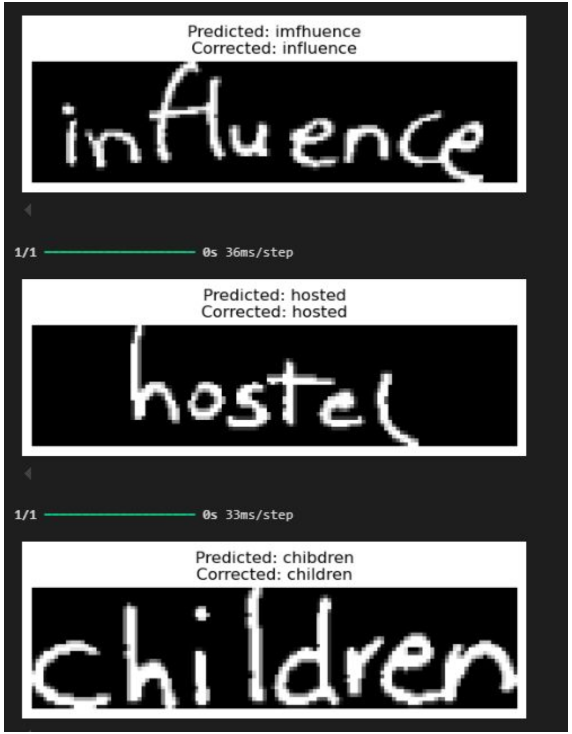
Project Summary

Developed a **Convolutional Recurrent Neural Network (CRNN)** model with Connectionist Temporal Classification (CTC) loss. Enhanced the performance by using the SymSpell Algorithm.

The **SymSpell algorithm** uses a dictionary to perform corrections for misspelled words within a specified **edit distance** (number of insertions, deletions, substitutions required to transform one word into another).

Results





Evaluation Metrics:

Character Error Rate (OCR Predicted): 14.291%

Word Error Rate (OCR Predicted): 75.862%

Character Error Rate (new Model): 18.198%

Word Error Rate(new Model): 47.932%

Conclusion

The SymSpell correction process helped reduce WER significantly but might have overly adjusted correct characters, leading to a higher CER. This trade-off suggests focusing on refining correction rules to balance these metrics.

Other Methods

- SymSpell Checker - Greedy Decoding, Beam Encoding
- PySpell Checker
- Pseudo Labeling

Future Improvement

In the post-processing phase, refining the dictionary-based correction methods by integrating context-aware language models, such as BERT or GPT, could provide more accurate predictions by leveraging semantic context.

Data augmentation techniques, such as elastic transformations or random distortions, can help the model generalize better to varied handwriting styles

Cut, Action! Generating Multi-Shot Human Speech Videos with Identity Consistency

Haoyu Zhen (hzhen@umass.edu),
Jiaben Chen (jiabenchen@umass.edu),
Zixin Wang (zixinwang@umass.edu)

Project #8

Motivation & Quick Summary

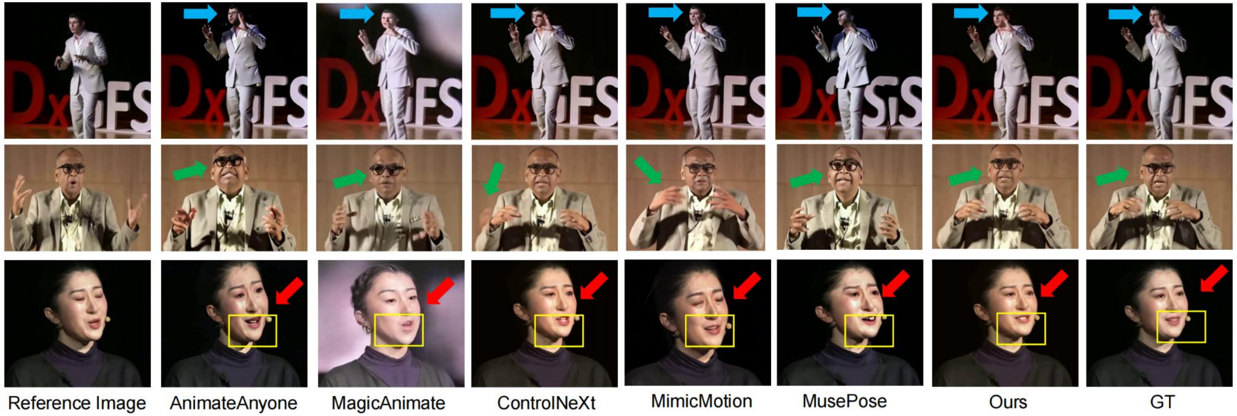
Results

Video Generation Comparison

Method	Video Generation Quality					ID Preser.
	SSIM↑	PSNR↑	LPIPS↓	FID↓	FVD↓	ArcFace Dis. ↓
MagicAnimate [43]	0.731	18.397	0.235	125.500	893.230	0.552
Animate Anyone [18]	0.754	20.468	0.176	93.230	789.360	0.450
MusePose [37]	**0.771**	19.468	0.191	106.760	823.020	0.513
ControlNeXt [31]	0.746	21.584	0.149	63.150	485.118	0.409
MimicMotion [49]	0.759	20.572	0.168	81.820	702.410	0.435
Ours	0.763	**21.959**	**0.146**	**62.550**	**480.210**	**0.372**

LLM Director Comparison

Method	Accuracy↑	SMA↑	IOU↑
Embedding Model	35.60%	30.42%	35.60%
Llama 3.1 Z.S.	20.41%	23.72%	10.50%
Llama 3.1 R.F.	24.63%	44.01%	13.28%
Llama 3.1 RAG	21.65%	47.15%	15.33 %
Llama 3.1 Tune	79.09%	49.40%	30.06%
GPT-4o Z.S.	64.34%	48.34%	40.59%
GPT-4o R.F.	67.50%	58.12%	42.46%
GPT-4o RAG (Ours)	70.66%	64.06 %	48.10%



Conclusion and Future Works

Summary

- We developed an automated pipeline **for multi-shot speech video** generation from text scripts and reference images
- We address **identity inconsistency** across different camera views through an ID adapter.

Future Works:

- Improve **motion continuity** to achieve smoother transitions
- Develop an **end-to-end system** by directly connecting the LLM and video diffusion model

Layer-selective rank reduction in VLM

Jeevana Karnuthala (jkarnuthala@umass.edu),
Srujana Pillarichety (spillarichet@umass.edu),
Vishnu Vardhan Reddy Kandi (vkandi@umass.edu)

Project #14

Motivation

- **LLMs** - LLMs showed promising accuracy improvement using a specific type of rank reduction
- **Vision-Language Models** (VLMs)
    - Excel at tasks like image captioning and visual question answering
    - **Computationally expensive** due to large parameter counts
    - Challenging to scale for real-world applications.

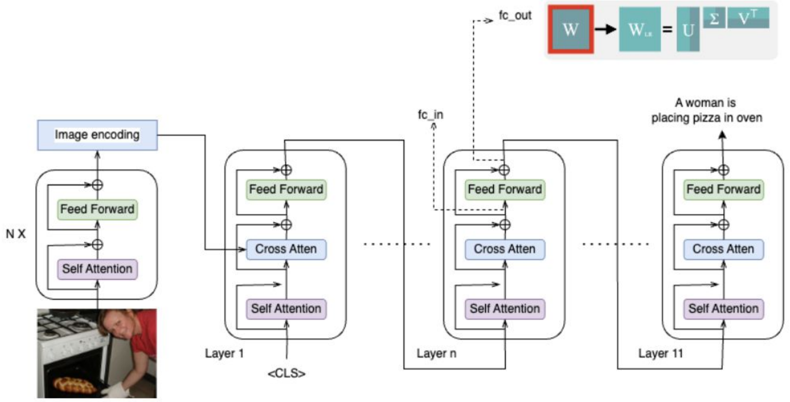
Can similar rank based reduction improve the performance in VLMs?

Problem Statement

The problem addressed in this study is to improve the performance of VLMs by applying **rank reduction techniques** to key weight matrices while making the model less computationally expensive

Method

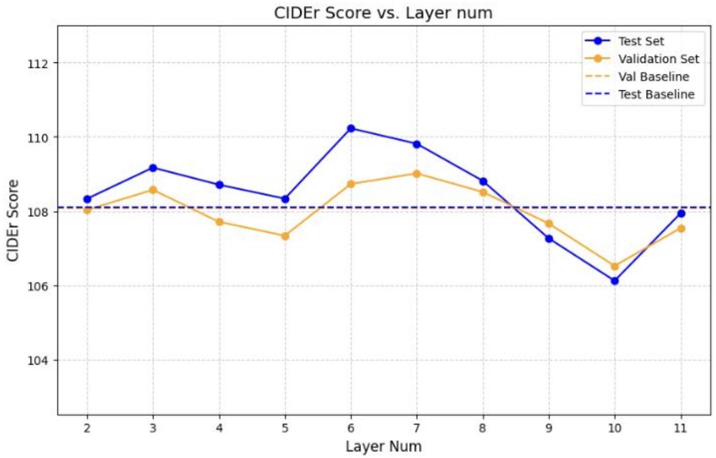
- Model - BLIP
- Tasks - Image Captioning, Visual Question Answerin
- Intervention Parameters - (l, n, r)
    - l - layer name (fc_in, fc_out)
    - n - layer number (text : [1, 11], visual : [1, 23])
    - r - intervention rate [0.1, 0.9]



Results

Dataset	CIDEr / (Base)	BLEU4 / (Base)	SPICE / (Base)	METEOR / (Base)
COCO Val set	**110.35** / (108.07)	27.95 / (27.59)	22.33 / (21.62)	27.71 / (27.56)
COCO Test set	**110.14**/ (108.10)	28.03 / (27.62)	22.33 / (21.91)	27.70 / (27.60)
NoCaps OOD set	**99.36** / (96.21)	30.90 / (30.95)	13.87 / (13.65)	26.76 / (26.93)
NoCaps Test set	**98.54** / (96.43)	32.9 / (31.27)	13.96 / (13.83)	28.42 / (28.31)

Dataset	Accuracy / (Baseline)
VQA Val set	86.35 / (86.19)
VQA Test set	86.28 / (86.22)
GQA Val set	63.66 / (62.86)
GQA Test set	63.163 / (62.77)



Gold Answer: Skewered pieces of meat are lying on the table next to a cup and pink bright flower

Base Model Answer: There a skewered meat on skewers on a wooden plate

After intervention: there are skewered meat on a wooden plate with a cup of sauce

Conclusion

- CIDEr score improved by 2.5% on COCO dataset and 3% on NoCaps OOD datasets.
- Rank-reduction on middle layers performed better compared to initial and later layers
- Intervention parameters optimized on BLIP with COCO showed better performance on the NoCaps OOD dataset, highlighting the model's generalizability
- Rank reductions on the text encoder boosted image captioning performance, while reductions on the visual encoder improved VQA results.
- Applying multiple interventions on the BLIP model resulted in promising outcomes, reducing model size while maintaining accuracy improvements

Intervention(l, n, r)	**CIDEr**	**BLEU4**	**METEOR**	**SPICE**
Baseline	96.21	30.95	26.93	13.65
(6, fc_out, 0.2)	99.36	30.90	26.76	13.87
(8, fc_in, 0.3)(6, fc_out, 0.2)	99.31	30.57	26.77	13.79
(8, fc_in, 0.3)(7, fc_out, 0.2)(6, fc_out, 0.2)	98.79	30.72	26.93	13.63
(8, fc_in, 0.3)(7, fc_out, 0.2)(6, fc_out, 0.2)(5, fc_in, 0.3)	98.43	31.56	27.04	13.51
(9, fc_in, 0.4)(8, fc_in, 0.3)(7, fc_out, 0.2)(6, fc_out, 0.2)(5, fc_in, 0.3)	98.95	30.36	26.72	13.89

Results of multiple interventions on NoCaps OOD dataset

Improving fashion recommendation relevance via fit aware neural re-ranking

Harshitha Kolukuluru (hkolukuluru@umass.edu),
Sai Sreenivas Chintha (saisreenivas@umass.edu),
Vishal G (vishalg@umass.edu)

Project #17

This project enhances **fashion recommendations** by leveraging **fine-tuned CLIP embeddings** and **custom features** like bounding box dimensions(using Faster R-CNN) and dominant colors(using KNN) to deliver precise, personalized suggestions. A custom **neural network** re-ranks top-k results, bridging gaps in traditional systems by aligning recommendations with user preferences for fit and style. **By fine-tuning CLIP for fashion-specific nuances, integrating custom features, and optimizing rankings,** the approach ensures accurate and user-focused results.

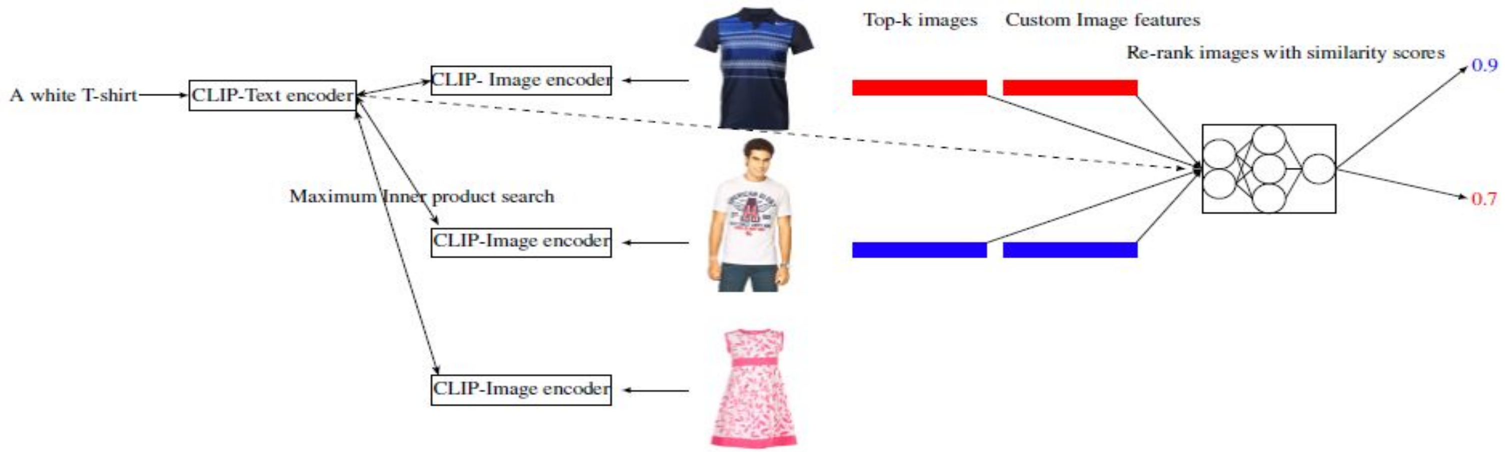


Figure 1. Diagram of our pipeline to first retrieve images with high recall and use re-ranking with custom image features such as bounding box, sleeve length etc. along with the embeddings of original image and query to achieve better precision

RESULTS

Model	R@1	R@3	R@5	R@10	R@20
CLIP(finetune)	0.42	0.66	0.78	0.90	0.95
CLIP(zero-shot	0.21	0.40	0.50	0.65	0.80
FashionVLP*(previous work)	0.26	-	-	-	-

CLIP baseline vs CLIP fine-tune performance

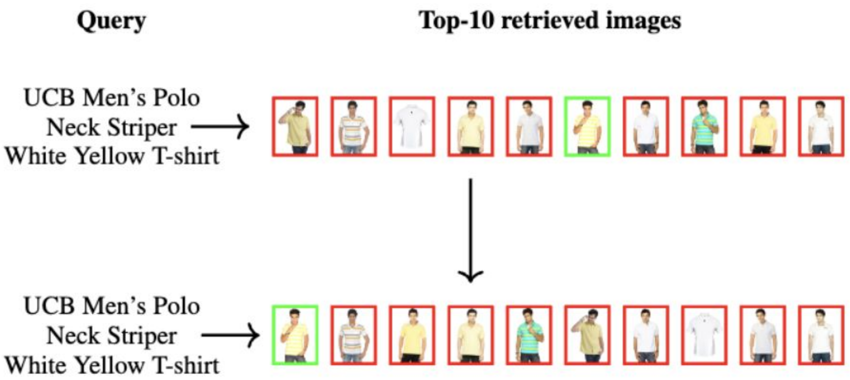
Model	R@1	R@3	R@5	R@10	R@20
CLIP(zero -shot)	0.16	0.34	0.45	0.61	0.86
NN(with zero-shot CLIP)	0.06	0.20	0.32	0.53	0.82

CLIP baseline vs Custom NN performance

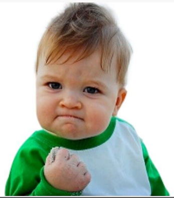
Model	R@1	R@3	R@5	R@10	R@20
CLIP(fine-tune)	0.30	0.49	0.58	0.75	0.93
NN(with fine-tune CLIP)	0.13	0.31	0.44	0.64	0.86

CLIP fine-tune vs Custom NN performance

*evaluated on Fashion200K dataset while our models were evaluated on the Fashion30K dataset



Before re-ranking (top) vs After re-ranking (bottom)



CONCLUSION

- We evaluated fashion recommendation methodologies on the Fashion30K dataset, where our **fine-tuned CLIP model significantly outperformed the baseline** in R@k and qualitative metrics, emphasizing the benefits of domain-specific tuning.

- Incorporating custom features like bounding box dimensions and dominant colors refined embeddings. However, the **custom neural network delivered mixed results**, underscoring the need for better feature integration and training strategies. Hard-negative sampling and fine-tuned embeddings improved alignment but revealed robustness limitations.

- **These findings highlight the importance of domain-specific engineering, structured prompts, and targeted sampling for enhancing recommendation accuracy, paving the way for future improvements.**

Fairness in kNNs

Alexander Yeung (ayeung@umass.edu),
Harold Thidemann (hthidemann@umass.edu),
Shreyans Babel (sbabel@umass.edu)

Project #19

Quick Summary

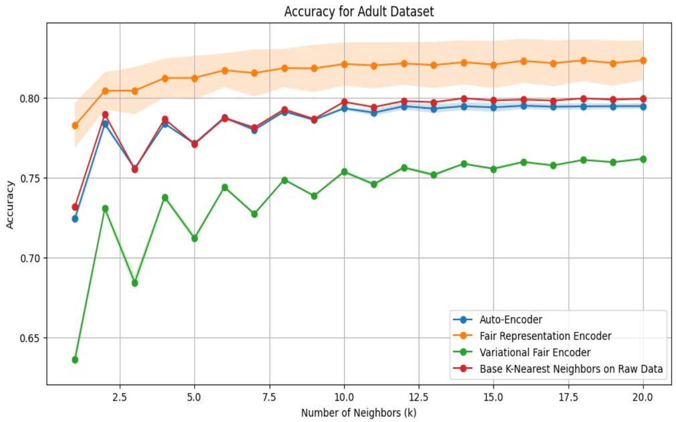
Problem: Machine learning models are prone to biased predictions when trained on datasets containing sensitive (e.g. age, sex, race…) demographic information

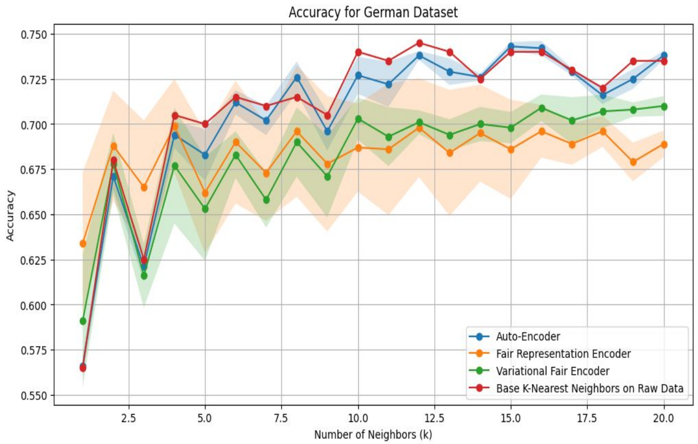
- We focused on mitigating biases in k-Nearest Neighbor (KNN) classifiers using different embedding techniques

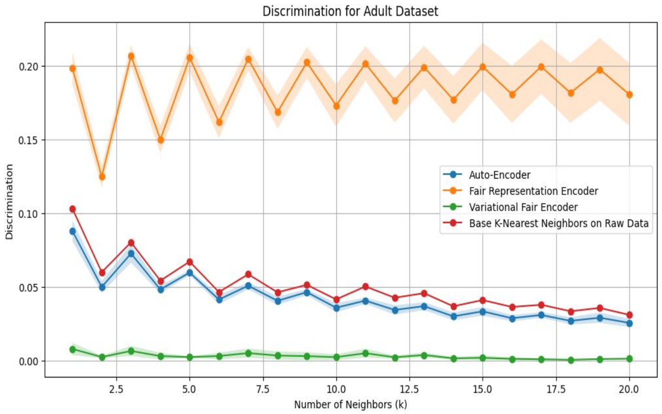
Goal: Develop a novel embedding generation method that minimizes sensitive demographic information while retaining prediction accuracy

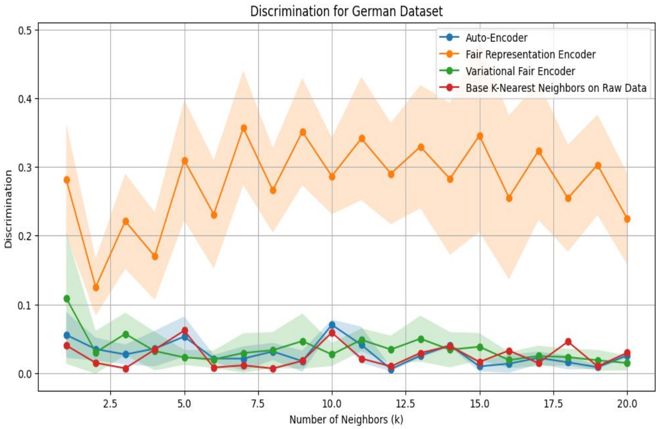
Approach: Train a variational autoencoder to generate biased minimized embeddings for kNN using a loss function to prioritize compression of sensitive attributes

Results


Accuracy for Adult Dataset
Accuracy
Number of Neighbors (k)
Auto-Encoder
Fair Representation Encoder
Variational Fair Encoder
Base K-Nearest Neighbors on Raw Data


Accuracy for German Dataset
Accuracy
Number of Neighbors (k)
Auto-Encoder
Fair Representation Encoder
Variational Fair Encoder
Base K-Nearest Neighbors on Raw Data


Discrimination for Adult Dataset
Discrimination
Number of Neighbors (k)
Auto-Encoder
Fair Representation Encoder
Variational Fair Encoder
Base K-Nearest Neighbors on Raw Data


Discrimination for German Dataset
Discrimination
Number of Neighbors (k)
Auto-Encoder
Fair Representation Encoder
Variational Fair Encoder
Base K-Nearest Neighbors on Raw Data

Conclusion

- Our model was able to preserve accuracy to an extent while greatly reducing discrimination
- Comparing to more baselines, different datasets, and expanding the hyperparameter search would provide more insight into its performance

Fair neural unlearning

Cameron Mattheson (cmattheson@umass.edu),
Riddho Ridwanul Haque (rhaque@umass.edu)

Project #22

OVERVIEW | NEURAL UNLEARNING

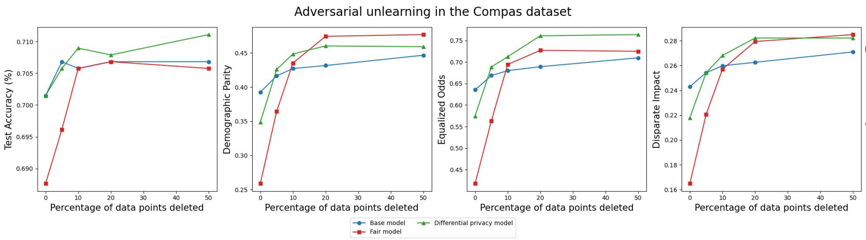
- Safe removal of user data from data networks is becoming a prominent issue in industry
- This can stem other issues, such as underrepresentation of minority categories and classification bias
- Question: How do the removal of key data points impact classification bias of sensitive groups?

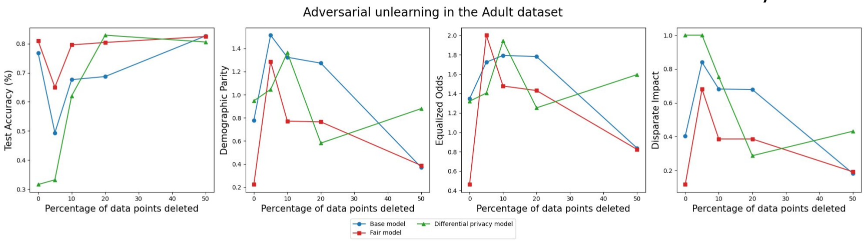
Models Utilzed

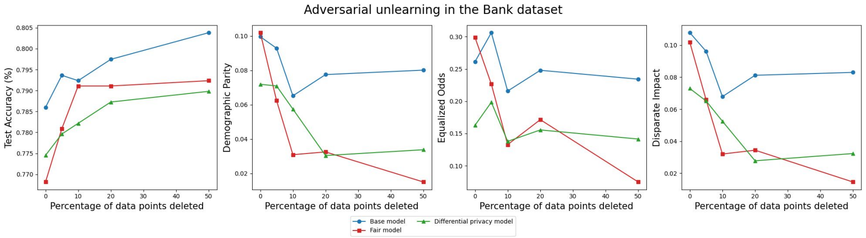
- Baseline
- Fair Learning Model: add classification bias into loss function
- Differential Privacy Model: normalize individual record gradients and add small noise during backpropagation

RESULTS

- Removing under-represented groups will consistently increase accuracy
- Correlated increase in unfairness of predictions of under-represented groups
- Models incorporating fairness into backpropagation see slight decrease in accuracy compared to base model, but see vase improvements in fairness metrics as more data points are removed







TAKEAWAYS

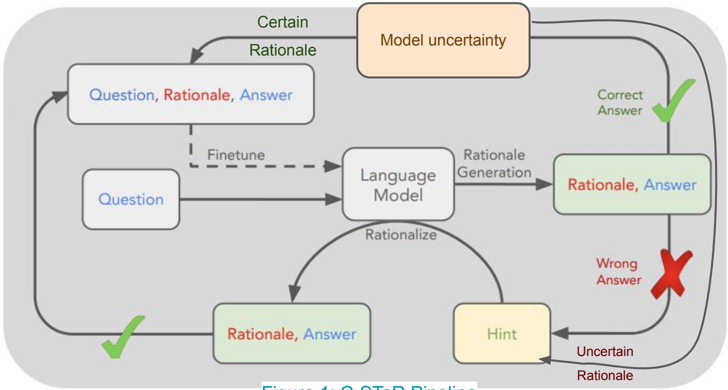
- Trade-off between prediction accuracy and classification bias in model optimization is significant depending on the task
- Data deletion of under-represented groups will naturally create more bias
- Incentivizing fairness during initial loss and backpropagation can minimize bias of skewed/sensitive features

C-STaR: Certain Self-Taught Reasoner

Jenna Russell (jrussell@umass.edu),
Rishanth Rajendhran (rrajendhran@umass.edu),
Vu Le (vdle@umass.edu)

Project #26

C-STaR Motivation: Improve the STaR methodology by only fine-tuning over rationales that the model is confident about, theoretically improving the quality of training data used for fine-tuning at each iteration




```
Q: What can be used
to carry a small dog?
Answer Choices:
(a) swimming pool
(b) basket
(c) dog show
(d) backyard
(e) own home
A: The answer must be
something that can be
used to carry a small
dog. Baskets are
designed to hold things.
Therefore, the answer
is basket (b).
```

Figure 1: C-STaR Pipeline (extension of STaR pipeline)

Algorithm: C-STaR

Input: M: a pretrained LLM; dataset $D = \{(x_i, y_i)\}_{i=1}^{|D|}$ (with few-shot prompts)

1: $M_0 \leftarrow M$ # Copy the original model

2: for n in $1, \ldots, N$ **do** # Outer loop

3: $(\hat{r}_i, \hat{y}_i) \leftarrow M_{n-1}(x_i)\ \forall i \in [1, |D|]$ # Perform rationale generation

4: $(\hat{r}_i^{\text{rat}}, \hat{y}_i^{\text{rat}}) \leftarrow M_{n-1}(\text{add\_hint}(x_i, y_i))\ \forall i \in [1, |D|]$ # Perform rationalization

5: $D_n \leftarrow \{(x_i, \hat{r}_i, y_i) \mid i \in [1, |D|] \wedge \hat{y}_i = y_i\}$ # Filter rationales using ground truth answers

6: $D_n^{\text{rat}} \leftarrow \{(x_i, \hat{r}_i^{\text{rat}}, y_i) \mid i \in [1, |D|] \wedge \hat{y}_i \neq y_i \wedge \hat{y}_i^{\text{rat}} = y_i\}$ # Filter rationalized rationales

7: $D_n^{certain} \leftarrow \{(x_i, r_i, y_i) | (x_i, r_i, y_i) \in D_n \cup D_n^{\text{rat}}$ s.t. $Uncertainty(M, r_i, (x_i, y_i)) < Threshold\}$ #Prune uncertain rationales

8: $M_n \leftarrow \text{train}(M, D_n^{certain})$ # Finetune the original model on correct solutions - inner loop

9: end for

Uncertainty Metrics

Perplexity: *How well a probabilistic model predicts a sample of data*

Entropy: *Measure of randomness within a rationale*

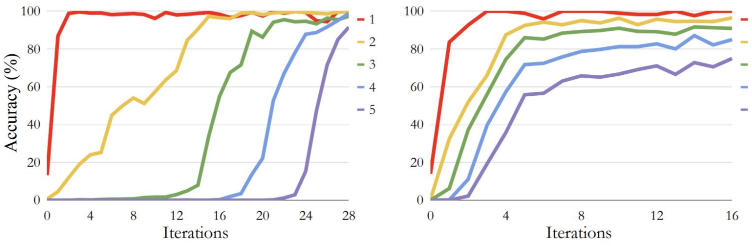
Prediction Confidence: *If model is confident in prediction, probably also confident in rationale*

Rationale Usefulness: *If there is a large difference in probability, rationale makes model more confident*

Paraphrase Consistency: *If model generates similar rationale many times, it must be confident*

Results

STaR results on CommensenseQA (GPT-J)



(a) Without rationalization (b) With rationalization

C-STaR results on CommensenseQA (Llama-3.1-8B-Instruct)

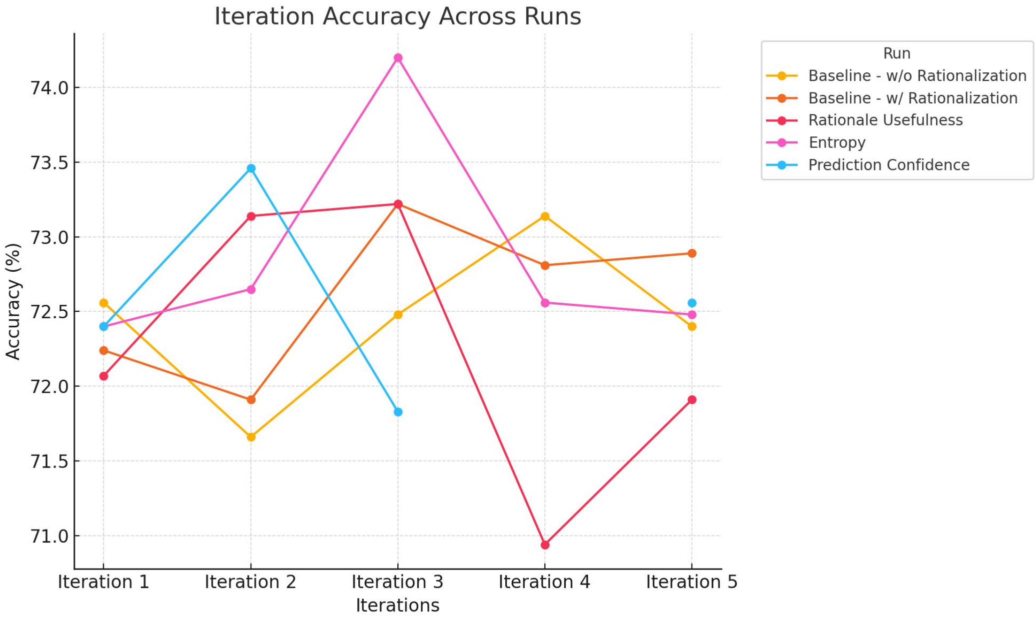


Table 3. Iteration Accuracy Across Methods

Run	Iteration 1 (%)	Iteration 2 (%)	Iteration 3 (%)	Iteration 4 (%)	Iteration 5 (%)
Baseline - w/o Rationalization	72.56	71.66	72.48	73.14	72.40
Baseline - w/ Rationalization	72.24	71.91	73.22	72.81	72.89
Rationale Usefulness	72.07	73.14	73.22	70.94	71.91
Entropy	72.40	72.65	74.20	72.56	72.48
Prediction Confidence	72.40	73.46	71.83	73.22	72.56

Conclusion

- Model uncertainty, as quantified in our experiments, does not seem to improve STaR pipeline significantly; Threshold a possible confounder.
- Remains unclear if uncertainty better or worse than fine-tuning on entire dataset
- Baseline Llama3.1-8b-Instruct performs as well as fine-tuned GPT-J on CommonsenseQA which could mean that CQA is not a good dataset to test on anymore.

Future Work

- Run over the GSM8k dataset (other commonly tested reasoning dataset) and other model families
- Expand evaluation framework to include more robust metrics than accuracy
- Research further methods of calculating uncertainty
- Explore impact of uncertainty threshold for rationale inclusion on reasoning capabilities

Mitigating bias in facial recognition using seldonion framework

Frances Chang (fchang@umass.edu),
Priya Balakrishnan (pbalakrishna@umass.edu),
Roshini Sanikop (rsanikop@umass.edu)

Project #30

Summary

Motivation

- Reliance on facial recognition (FR) technologies is growing
- **Biases** against underrepresented groups are common
  - → negatively impacts minorities, leading to **ethical and societal concerns**
  - → undermines **technologies' effectiveness**, sometimes in fields with higher stakes (e.g. security, healthcare, criminal justice, finance)
- We are motivated to reduce biases in FR models to:
  - enhance fairness and reduce harm to underrepresented groups and society
  - improve models' reliability and effectiveness

Background

- **Seldonian Framework**
  - Developed by Philip S. Thomas, Bruno Castro da Silva, Andrew G. Barto, Stephen Giguere, Yuriy Brun, and Emma Brunskill
  - Lets users without technical expertise define a **probabilistic constraint against biases**; can be applied across diverse applications
  - Expressed as:

$$\Pr(g(a(D)) \leq 0) \geq 1 - \delta$$

  - Incorporation of the framework into our model lets us identify solutions that maximize our objective while controlling biases across demographic groups

Problem

- We aim to integrate the Seldonian Framework into a facial recognition model (e.g., an age prediction model) to mitigate biases.
- The objective is to build a facial recognition model with high probabilistic guarantees on fairness constraints. This involves reducing the accuracy difference between predictions for Caucasian faces and non-Caucasian faces, including Black, Hispanic, Southeast Asian, Indian, and other racial groups, ensuring equitable performance across demographics.

Prior Works

- Seldonian framework applications: RL, classification, RobinHood and ELF algorithms
- Some other works mitigating unfairness: adversarial debiasing, reweighting or resampling training data

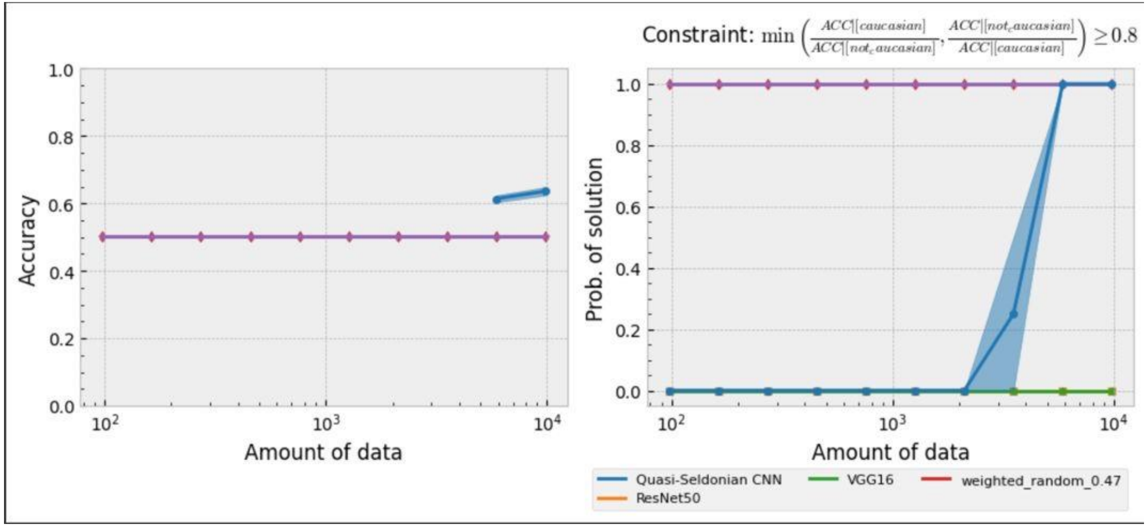
Methods & Results

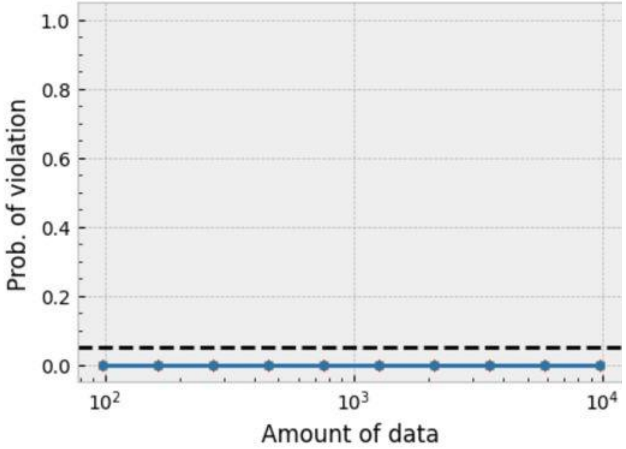
Methods

- **Data:** FairFace Dataset
  - contains images across 7 racial groups
  - balanced, curated specifically to reduce racial bias in FR systems
- **Models** being trained and compared:
  - Baseline model 1 (fairness-unaware): ResNet50
  - Baseline model 2 (fairness-unaware): VGGFaceNet
  - Core model with integration of the Seldonian model (Definition of fairness constraint: model's accuracy remains within 20% of each other for two chosen demographic groups, with confidence level δ = 95%)
- $$\min\left(\frac{\text{Accuracy} \mid \text{race} = 3}{\text{Accuracy} \mid \text{race} \neq 3}, \frac{\text{Accuracy} \mid \text{race} \neq 3}{\text{Accuracy} \mid \text{race} = 3}\right) \geq 0.8 \tag{6}$$

  We will enforce this constraint with a confidence level of $\delta = 0.05$, meaning we require the fairness condition to hold with at least 95% confidence.

Results





Evaluating effectiveness of GNN for node classification in sanitation infrastructure mapping

Sylvia Imanirakiza (simanirakiza@umass.edu),
Nana Agyemang Sereboo (nagyemangser@umass.edu)

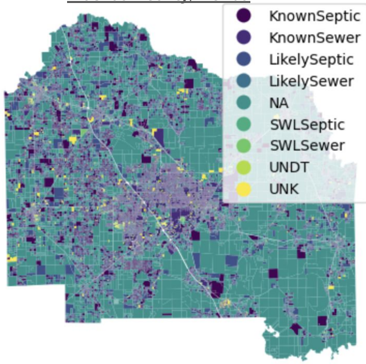
Project #32

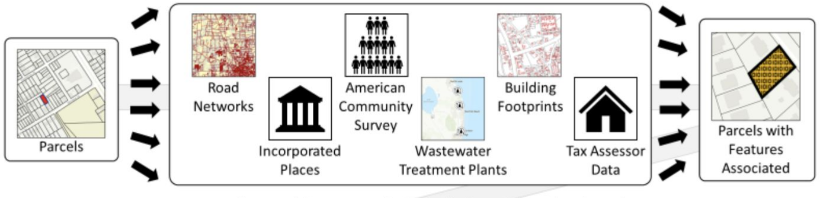
Quick Summary

- America's wastewater infrastructure lacks updated mapping data since the 1990s.
- Maintenance issues (leaks, system failures, environmental contamination) persists due to outdated information.
- Recent works employ **traditional machine learning models, but they fail to capture the complex relationships in urban infrastructure data. Graph Neural Networks (GNNs)** excel at **capturing complex spatial relationships** making them suitable for predictive maintenance and management across interconnected sanitation networks.

How effective are Graph Neural Networks for node classification in mapping sanitation systems?

Alachua County, Florida



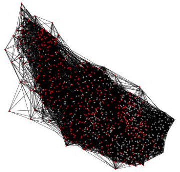


1. We assign features (characteristics) to individual parcels based on these datasets.

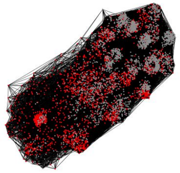
Image Credit: Nelson da Luz et al.
Developing methods to predict buried sanitation infrastructure. work in progress, 2023.

Results

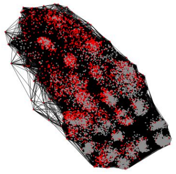
We created graph representations for different subsets of parcels **(1,000, 5,000, and 10,000), sampling across five distinct folds using a similarity network approach**.



(a) Sample(1) nrows1k graph



(b) Sample(1) nrows5k graph



(c) Sample(1) nrows10k graph

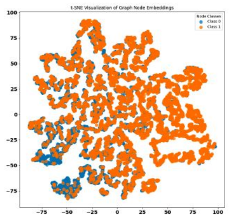
- For the transductive setting, the GNN models (GraphSAGE, GCN, GAT) consistently outperform the baseline MLP across all graph sizes.
- Particularly For 50k nodes, ***the GraphSAGE model achieves an accuracy comparable to the random forest proposed by prior work.***
- For the inductive setting, GraphSAGE consistently outperforms both GCN and GAT across all graph sizes likely **due to its ability to learn representations that generalize well to unseen graph structures**.
- GCN exhibits the poorest performance across all graph sizes due to its structural dependency,
- In both settings, the GraphSAGE performance saturates slightly on larger graphs, suggesting potential limitations in scalability.
- In both settings, GAT struggles to generalize across varying graph structures.
- The node embeddings visualized using t-SNE show the GraphSAGE **models capacity to learn and leverage the inherent structure of the graph to improve its predictions.**

Table 2. Performance of the different GNN models and the baseline MLP in the **transductive setting**. The values indicate the mean and standard deviation accuracy across 5 distinct samples of a given number of parcels(nodes), excluding 50k nodes, where we used one sample. Bold values indicate best performance.

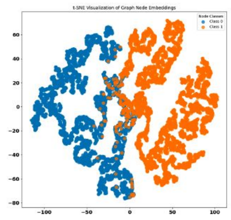
Model	1k Nodes	5k Nodes	10k Nodes	50k Nodes
GCN	**83.80% ± 6.83**	93.12% ± 1.04	94.62% ± 0.61	91.72%
GAT	81.20% ± 1.92	88.68% ± 2.15	93.60% ± 2.83%	-
GraphSAGE	83.20 % ± 2.59	**93.60% ± 0.58**	**94.72% ± 0.53**	**96.42**
Baseline MLP	83.20% ± 7.05	83.84% ± 1.66	83.64% ± 0.53	84.52% ± 0.76
RF [3]	-	-	-	93.5%

Table 3. Performance of the different GNN models and the baseline MLP in the **inductive setting**. The values represent the accuracy of each model when trained on a graph structure with n nodes from Sample SF=1, validated on a graph structure from Sample SF=2, and tested on a graph structure from Sample SF=3. Bold values indicate the best performance.

Model	1k Nodes	5k Nodes	10k Nodes
GCN	64.50	78.54%	76.56%
GAT	69.30%	75.22%	74.47%
GraphSAGE	**81.60 %**	**89.52%**	**91.49%**



(e) Untrained GraphSAGE



(f) Trained GraphSAGE

Conclusion

- We demonstrate the effectiveness of Graph Neural Networks (GNNs) for node classification in mapping sanitation infrastructure, addressing a critical gap in wastewater mapping data.
- Our findings reveal **GraphSAGE as a highly effective and scalable model, outperforming GCN and GAT in inductive and transductive settings.**
- This research shows the ability of GNNs to **extract and leverage complex spatial and functional relationships in urban infrastructure data, offering a significant step toward supporting wastewater mapping systems.**
- Future work will focus on optimizing **GNN architectures for scalability, improving generalization across diverse graphs,** and advancing **explainability to support real-world deployment and stakeholder decision-making.**

Efficient Knowledge Distillation on YOLOv8

Arav Adikesh Ramakrishnan (aravadikeshr@umass.edu),
Siddhartha Jaiswal (sjaiswal@umass.edu)

Project #34

Project Overview and Motivation

- **Motivation:**
  - Investigate effectiveness of KD on a model architecture as complex as YOLO
  - Address computational constraints in deploying deep learning models for classification tasks.
- **Background:**
  - Knowledge distillation transfers knowledge from large "teacher" models (YOLOv8Large) to compact "student" models (YOLOv8Nano).
  - Utilized **Kullback Leiber Divergence**, **Logit Standardization**, and **Curriculum Temperature Scheduling**
- **Goal:**
  - Achieve high accuracy with resource-efficient models while maintaining computational efficiency

Main Results

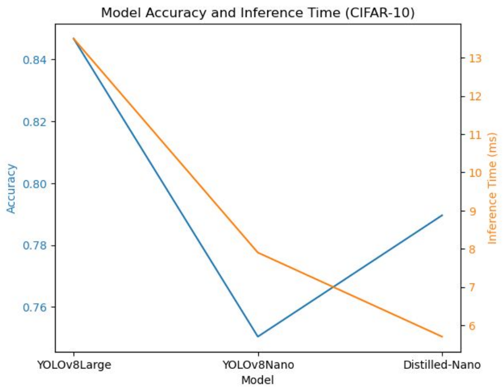
Datasets: CIFAR-10, Tiny-ImageNet, Oxford-IIIT Pets.

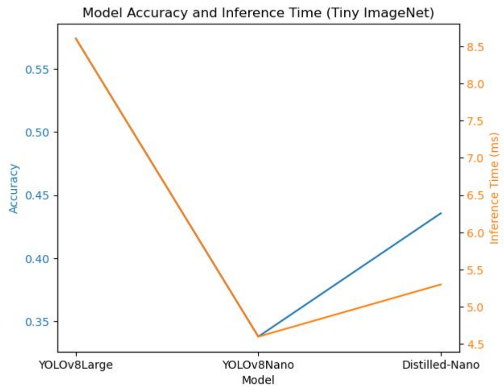
- **Key Metrics Improvements:**
    - **CIFAR-10:** Accuracy improved by 5.2% with reduced inference time.
    - **Tiny-ImageNet:** Accuracy increased by 35.36%.
    - **Oxford-IIIT Pets (Fine-grained):** Accuracy boosted by 37.61%.

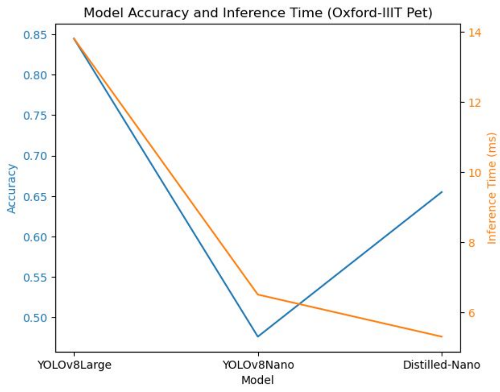
Dataset	Model	Params (M)	FLOPs (B)	Accuracy (%)	Inference Time (ms)
CIFAR-10	YOLOv8l Baseline	35.7	99.7	84.67	13.5
	YOLOv8n Baseline	2.7	4.3	75.04	7.9
	YOLOv8n Distilled	2.7	4.3	**78.96**	**5.7**
Tiny-ImageNet	YOLOv8l Baseline	35.7	99.7	57.38	8.6
	YOLOv8n Baseline	2.7	4.3	33.80	4.6
	YOLOv8n Distilled	2.7	4.3	**43.56**	**5.3**
Oxford Pets	YOLOv8l Baseline	35.7	99.7	84.46	13.8
	YOLOv8n Baseline	2.7	4.3	47.62	6.5
	YOLOv8n Distilled	2.7	4.3	**65.49**	**5.3**

Conclusion

- **Summary:**
  - Successfully distilled YOLOv8Large to YOLOv8Nano with novel techniques, achieving significant accuracy improvements without increasing resource demands.
  - Demonstrated robustness for the distilled model across standard and fine-grained classification tasks.







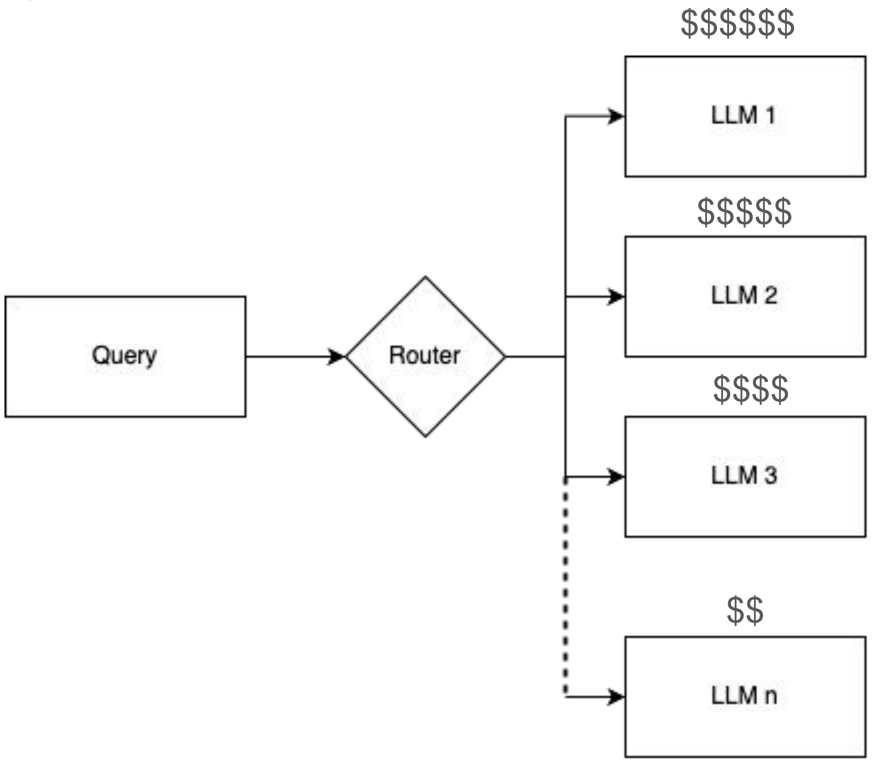
LLM-routing maximizing output quality while minimizing cost

Aparajith Raghuvir (araghuvir@umass.edu),
Jeet Devendra Sharma (jeetdevendra@umass.edu),
Tarun Gowda (tgowda@umass.edu)

Project #40

Overview, Motivation and Background

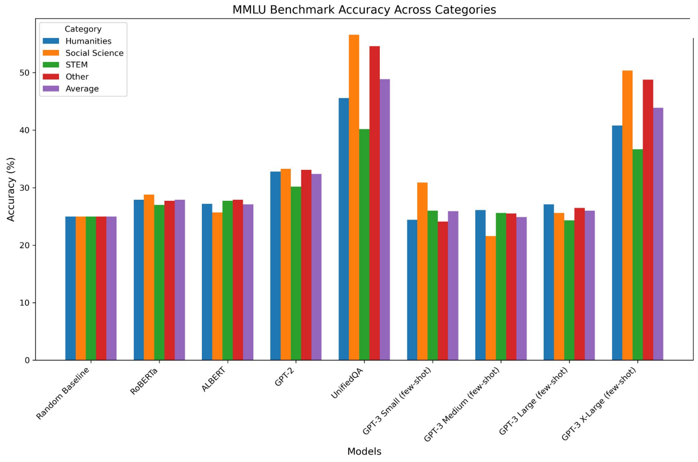
- Design an intelligent system to route prompts to the most suitable LLM, balancing cost efficiency and output quality.
- We take 3 approaches, route for category based questions, general prompts and coding queries.
- Democratize high-quality AI access to avoid monopolies by ensuring state-of-the-art results at reduced costs.

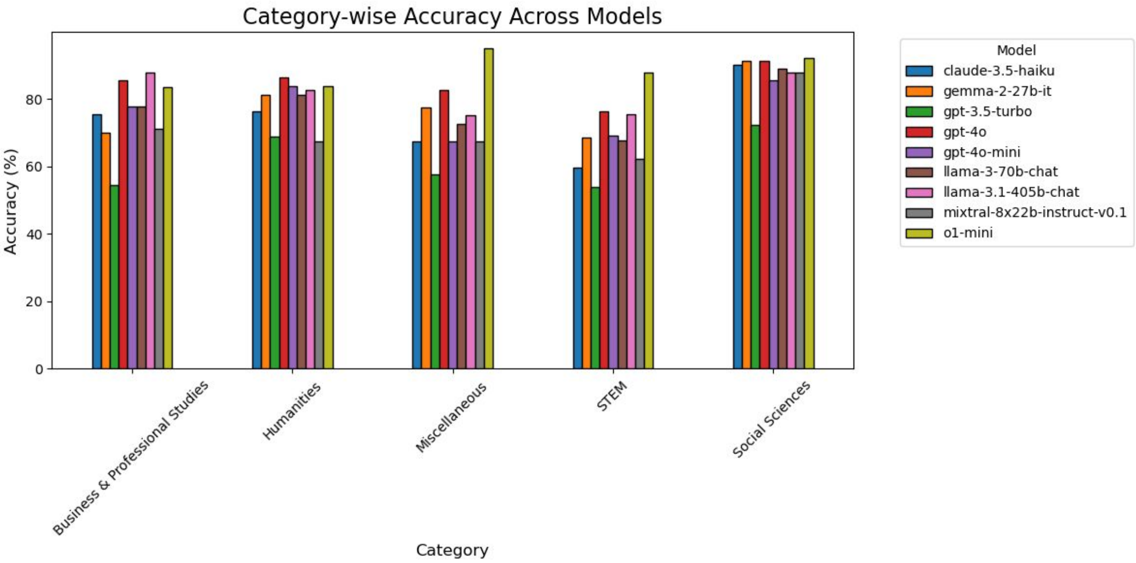


Results & Conclusion

Cost per million token($)

O1 mini	Gemma
5.25	0.25





- Categories like Social Sciences everything performs well.
- For Business and Social Studies llama-3.1-405b performs the best.

Model	GPT-4o calls	Other calls	Cost ($/M Tokens)
Base	100.00	0.00	2.50
Router	74.64	25.36	2.09

Coding specific router

General prompt router

- Saves money with better output.
- More data and more diverse data is needed for better router - clearly seen by the overfitting in the general prompt router.
- Consistent generalization gap.

Predicting hospital readmission rates

Eniola Adegbegha (yadegbegha@umass.edu),
Itir Sayar (isayar@umass.edu)

Project #43

Motivation:

- Hospital readmissions within 30 days are costly and indicate potential care quality issues.
- Reducing readmissions improves patient outcomes and reduces healthcare costs.

Background:

-> Traditional methods (e.g., logistic regression) struggle with the complexity of EHR data.

-> Advances in machine learning enable more accurate predictions using neural networks.

-> We aimed to compare the performance of CNNs, LSTMs, and ANNs for predicting readmissions of CHF patients.

Objective:

-> Evaluate neural networks for 30-day readmission prediction.

-> Benchmark predictive performance and resource efficiency.

Main Results

	Precision	Accuracy	ROC-AUC
ANN	0.45	0.78	0.63
CNN	0.50	0.79	0.62
LSTM	0.44	0.79	0.61

Conclusions and Next Steps

- ANN performed slightly better in ROC-AUC but had lower precision compared to CNN.
- CNN's higher precision suggests it may reduce false positives, making it more suitable for real-world use.
- LSTM, while expected to excel due to its temporal capabilities, underperformed, likely due to dataset limitations.

Future Work:

- Optimize hyperparameters for all models to improve performance.
- Incorporate additional features, such as social determinants of health, to enhance predictive power.
- Compare results with hybrid models (e.g., CNN + LSTM) for improved accuracy and precision.

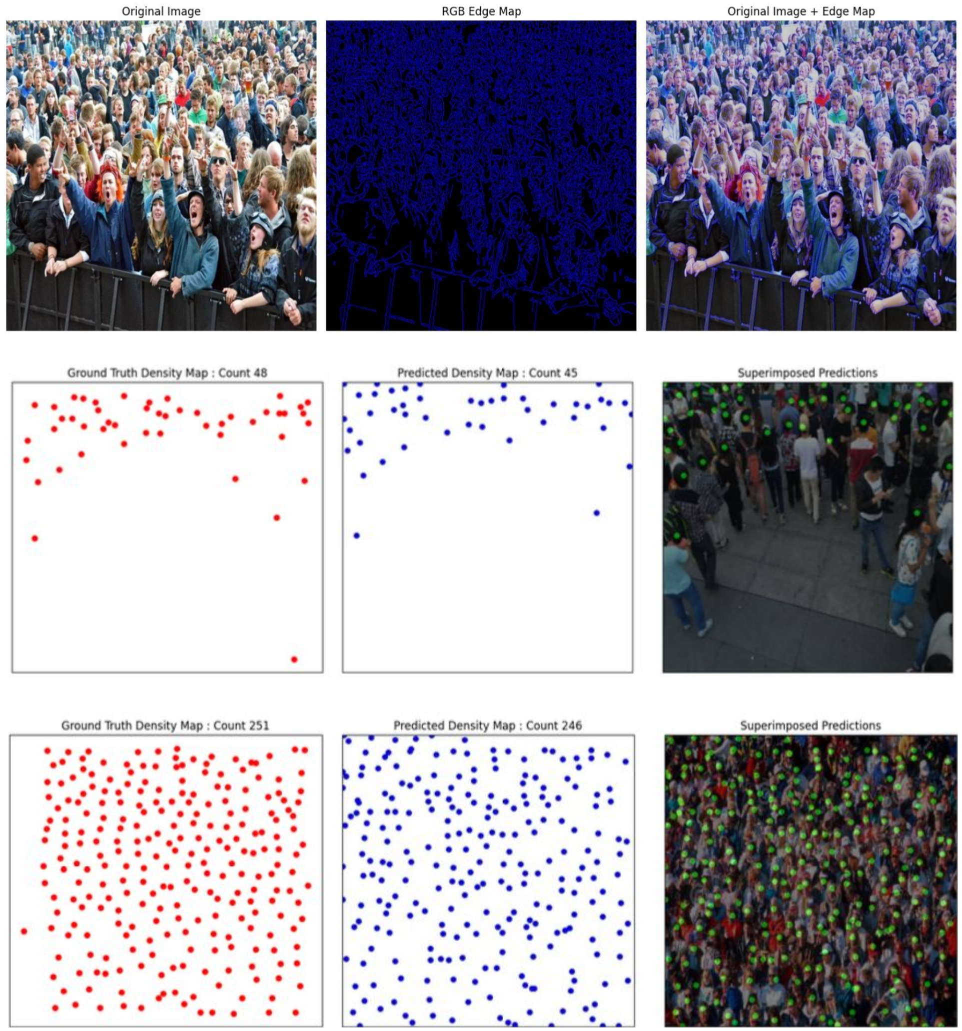
Density estimation and crowd counting

Balachandra Devarangadi Sunil (bdevarangadi@umass.edu),
Rakshith Venkatesh (rakshith@umass.edu),
Shantanu Todmal (stodmal@umass.edu)

Project #44

Motivation and background

- **Crowd Density Estimation Challenges**:
  - Essential for public safety, event management, and strategic crowd control.
  - Current methods struggle with variability, occlusions, and high-density environments.
- **Proposed Solution**:
  - Leverages **Diffusion models** for high-fidelity density map generation and narrow Gaussian kernels for enhanced precision.
  - Introduces **Event-driven video sampling** using Farneback optical flow to focus on significant frames, reducing computational load while processing videos.
- **Efficiency Gains**:
  - **Single pass and Superimposed edge detection** maintains precision while reducing inference time, enabling real-time video analysis.
  - Efficiently integrates **spatial and temporal analysis** for real-time crowd monitoring.
- **Datasets**:
  - ShanghaiTech datasets test the approach across diverse scenarios.



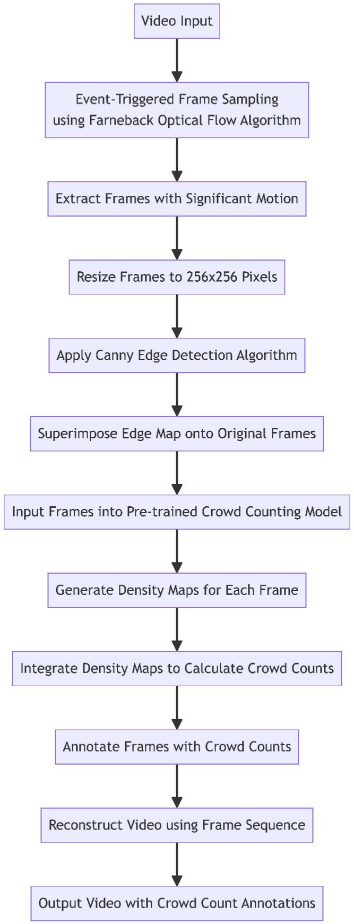


Figure 1. Flowchart for proposed solution

Sampling Method	Saved Frames	Total Frames	Correct Frames	Correct Frame Rate (%)
Uniform Sampling	40	1,177	15	37.5%
Random Sampling	10	1,177	4	40.0%
Adaptive Sampling	50	1,177	28	56.0%
Keyframe Sampling	10	1,177	6	60.0%
Stratified Sampling	15	1,177	7	46.7%
Event-Based Sampling	**34**	**1,177**	**30**	**88.2%**

Table 1: Sampling Technique

Video	TP	FP	FN	Precision	Recall	F1
Video 1	30	4	5	0.882	0.857	0.869
Video 2	9	3	1	0.750	0.900	0.818
Video 3	33	7	8	0.825	0.805	0.815
Video 4	34	10	8	0.773	0.810	0.791
Video 5	54	11	4	0.831	0.931	0.878
Video 6	21	2	1	0.913	0.955	0.934

Table 2: Event-Based Sampling Stats

Method	ShTech A	ShTech B
TopoCount	61.2	7.8
SUA	68.5	14.1
ChFl	57.5	6.9
MAN	56.8	NA
GauNet	54.8	6.2
CLTR	56.9	6.5
CrowdHat	51.2	5.7
STEERER	54.5	5.8
PET	49.3	6.2
CrowdDiff	47.4	5.7
Proposed Approach	**54.7**	**6.4**

Table 3: MAE Metrics for Diverse Approaches

Conclusion

- **Event-Driven Sampling Strategy**
  Introduced sampling based on Farneback optical flow algorithm.
  Reduced computational overhead while preserving critical crowd dynamics.
  Achieved an average **F1 score of 0.92** across test videos.
- **Extended Crowd Density Estimation to Videos**
  Leveraged conditional diffusion models to handle temporal dynamics and computational efficiency.
- **High-Fidelity Density Maps**
  Generated accurate density maps capturing crowd distributions in sampled frames.
  Estimated crowd counts closely aligned with ground-truth data.
- **Efficiency for Real-Time Applications**
  Essential for scenarios with constraints on processing speed and storage capacity.
  Valuable for public safety, disaster response, and event management.
- **Contributions**
  Developed a scalable, accurate, and efficient framework for monitoring crowd behavior in dynamic environments.
  Real-time capable for crowd monitoring and event management.

Optimizing video QA with hierarchical frame extraction and video-LLama

Reshma Ashok (rashok@umass.edu),
Satya Srujana Pilli (spilli@umass.edu),
Varshini Venkataraman (vvenkatarama@umass.edu)

Project #45

Project Overview

- Long-form videos generate vast amounts of data, making traditional QA systems resource-intensive and prone to missing key contextual information. Existing methods like fixed-interval frame sampling often fail to capture meaningful frames, reducing QA accuracy. There is a need for an efficient system that balances computational overhead with performance.

- **Dataset** used is **CinePile**.

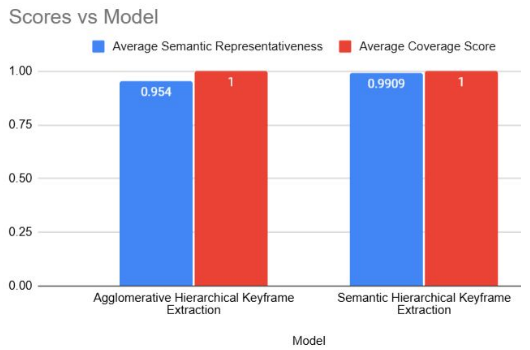
- **Proposed Solution**

  1.**Hierarchical Keyframe Extraction:**We have used two approaches for this keyframe extraction: **Agglomerative Clustering with Visual Features** and **Agglomerative Clustering along with Semantic Features.** In both cases, we identify 16 key frames that best represent the video's core content using the clustering methods. By incorporating both visual and semantic features, the method ensures that the selected frames effectively capture the essence of the video, while also optimizing computational efficiency.

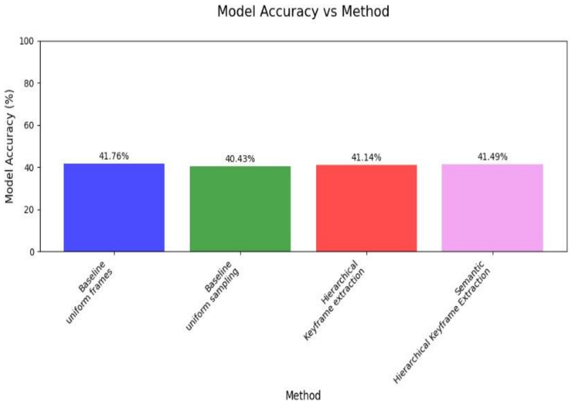
  2.**Video-LLaMA Integration:** The selected keyframes, along with their corresponding subtitles, are processed by the Video-LLaMA model for video question-answering. The model efficiently integrates both visual and textual data to provide accurate answers to multiple-choice questions.

Results

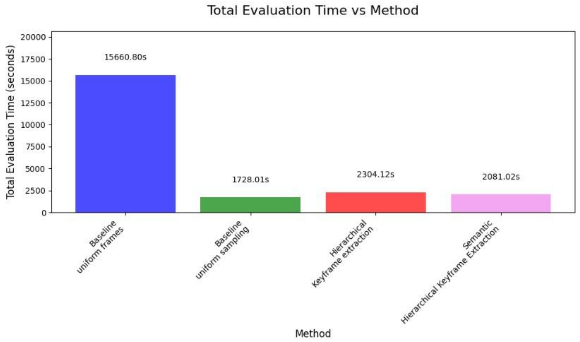
Quality of Hierarchical Keyframe Extraction Approaches Using Semantic Representativeness and Coverage Scores



Comparison of Model Accuracy Across Keyframe Extraction Methods and the Baseline



Evaluation Time Comparison Across Keyframe Extraction Methods and the Baseline



Conclusion

- **Efficiency Redefined:**
  Reduced processing time from approximately **5 hours** to just **30 minutes** by extracting **16 keyframes per video**.
- **Better Performance:**
  Almost achieves **baseline-level accuracy** with dramatically lower computational costs. Combines **semantic features** and **visual features** for superior keyframe extraction and clustering process.
- **Future Enhancements:** Enhancing accuracy by introducing further dimensions-such as more detailed semantic features, temporal dependencies, and better fusion techniques between textual and visual inputs.Enhance cluster separation with contrastive learning, adapt to scene complexities with dynamic thresholds, and scale for impactful video summarization and moderation.

Compact Diffusion Policy for Robot Control

Jeongah Lee (jeongahlee@umass.edu),
Shauna Choi (seohyunchoi@umass.edu),
Suyoung Kang (suyoungkang@umass.edu)

Project #47

Summary

Motivation

Limitation of **Diffusion Models**:
Prolonged inference time,
caused by the numerous denoising
steps required to generate
high-quality outputs.

Solution

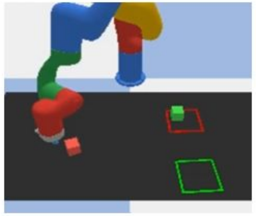
Apply **Knowledge Distillation**
(teacher-student model)

Result

Our goal is to **reduce inference time** while maintaining comparable task success rates

(1) Task

Push-T



Block pushing

(2) Evaluation Metric: Inference time, Mean score, IoU for Push-T and px score for Block pushing

Table 1. Inference Time

	Push-T	BlockPush
Teacher model	0.829	0.816
Our	**0.059**	**0.049**

"15 times faster"

Methodology & Implementation

Baseline paper (RSS 2023)

Diffusion Policy: Visuomotor Policy Learning via Action Diffusion

Cheng Chi*[1], Zhenjia Xu*[1], Siyuan Feng[2], Eric Cousineau[2], Yilun Du[3], Benjamin Burchfiel[2], Russ Tedrake[2,3], Shuran Song[1,4]

- We utilize a pre-trained diffusion visuomotor policy as the teacher model. It is implemented based on Denoising Diffusion Probabilistic Models (DDPM).
- We adopt the consistency distillation approach to enable single-step inference. This meleverages an ordinary differential equation (ODE) solver and a pre-trained teacher diffusion model.

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^{-}; \phi) := \mathbb{E}[\lambda(k)\, d(\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{A}^{k+1}, \mathbf{O}, k+1), \boldsymbol{f}_{\boldsymbol{\theta}^{-}}(\hat{\mathbf{A}}^{k}_{\phi}, \mathbf{O}, k))]$$

$\hat{\mathbf{A}}^{k}_{\phi}$: One-step Euler ODE solver / $\mathbf{A}$: Actions / $\boldsymbol{s}_{\phi}$: Score model

Result & Conclusion

(1) Significant Speed Improvement: 15 times faster

Table 1. Inference Time

	Push-T	BlockPush
Teacher model	0.829	0.816
Our	**0.059**	**0.049**

(2) Critical Role of Distillation: compared to one step

Table 2. Mean Score

	Push-T		BlockPush	
	train	test	train	test
Teacher model	0.997	0.917	0.497	0.198
Teacher model (one step)	0.115	0.120	0.082	0.039
Our	0.786	0.509	0.167	0.079

(3) Task Complexity Impact on Performance

Table 3. IoU score for Push-T

	Push-T	
	train	test
Teacher model	0.750	0.728
Teacher model (one step)	0.0	0.0
Our	0.621	0.343

Table 4. PX score for block pushing

	Train		Test	
	p1	p2	p1	p2
Teacher Model	0.666	0.333	0.3	0.1
Teacher Model (one step)	0.166	0.0	0.08	0.0
Ours	0.167	0.167	0.12	0.04

Improving Video-LLaMa’s performance with low-quality video input

Katie Zhang (kzhang@umass.edu),
Vani Gupta (vanigupta@umass.edu)

Project #50

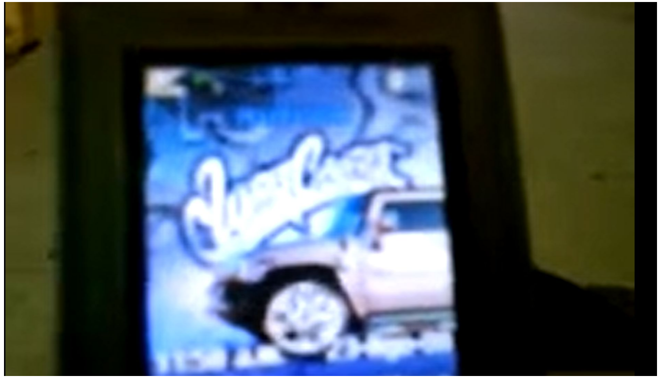
Summary

- Video LLaMa is a powerful multimodal LLM, but it is limited by the quality of the input received, as it was trained and tested on high-quality data (stock footage)
- It shows poorer performance with Q&A when presented with videos of lower video quality/resolution, garbled audio, or other imperfections, even though it is somewhat comprehensible to humans
- Re-training the model to account for poor quality videos is impractical, so we approach the problem from a preprocessing standpoint
- There appears to be no literature dedicated to investigating video LLM performances with poor quality input/suggesting possible improvements
- We employ **video super-resolution (VSR)** to preprocess the input - generating higher-quality frames from low-quality videos
- Use Real-ESRGAN algorithm to enhance individual frames, then stitch them back together

Results

Human understanding: Someone is booting up a phone with a Windows OS and is scrolling through different screens and menus and listening to music.

Before



After

“Can you describe what’s happening in this video?”

Sure! The video shows a person using a phone to listen to music and browse through different options on the screen.

Sure! The video shows a welcome screen on a device, followed by a menu screen with options to choose from. Then we see a TV screen displaying a picture of a car and a person holding a cell phone showing a picture of a car.

“Can you be more specific about what’s on screen?”

Unfortunately, I cannot provide more information about what's on the screen as the video does not show any clear images of the screen content.

The screen displays various images and videos, including a welcome message on a black background, a menu with options for settings and other features, a picture of a car, a video of a person holding a cell phone, and a blurry image of a TV screen showing a blue sky.

Conclusion

- After preprocessing, descriptions tend to be more specific
- Sometimes preprocessing doesn’t really help improve video quality much
- Still sometimes prone to hallucinations or misinterpretations
- Consideration must be given to processes outside the model in order to make models more robust
- Future considerations:
    - Is preprocessing a cost-/time-effective method for improving performance overall?
    - Would other VSR algorithms work more efficiently?
    - Is it possible to preprocess too much?

CoGIR: ControlNet-Guided Image Refinement

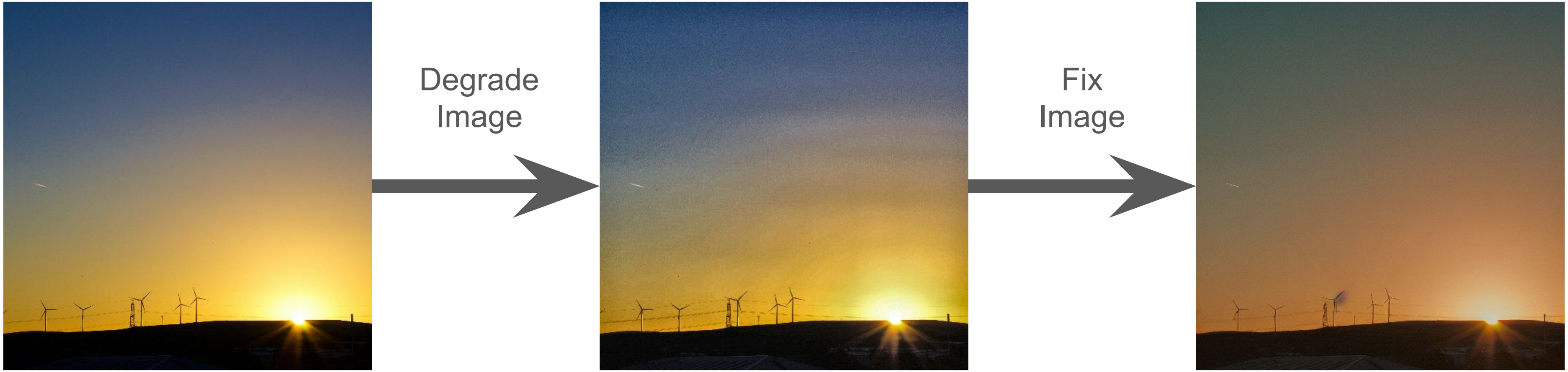
Jeremy Hummel (jhummel@umass.edu),
Nicole Kaldus (nkaldus@umass.edu)

Project #51

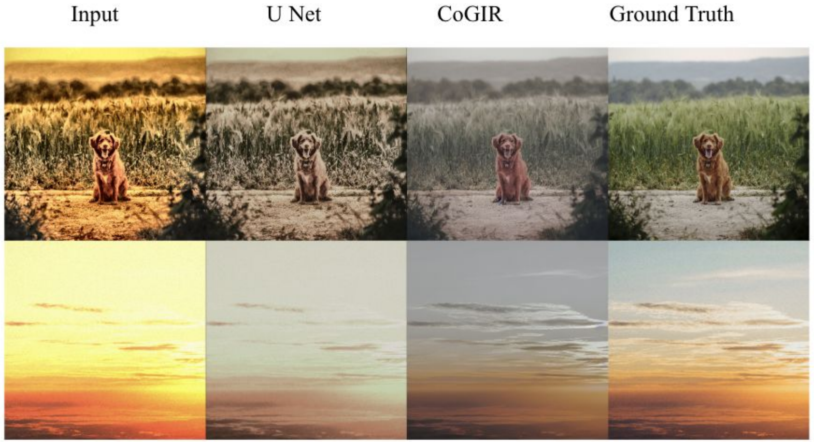
Motivation and Background

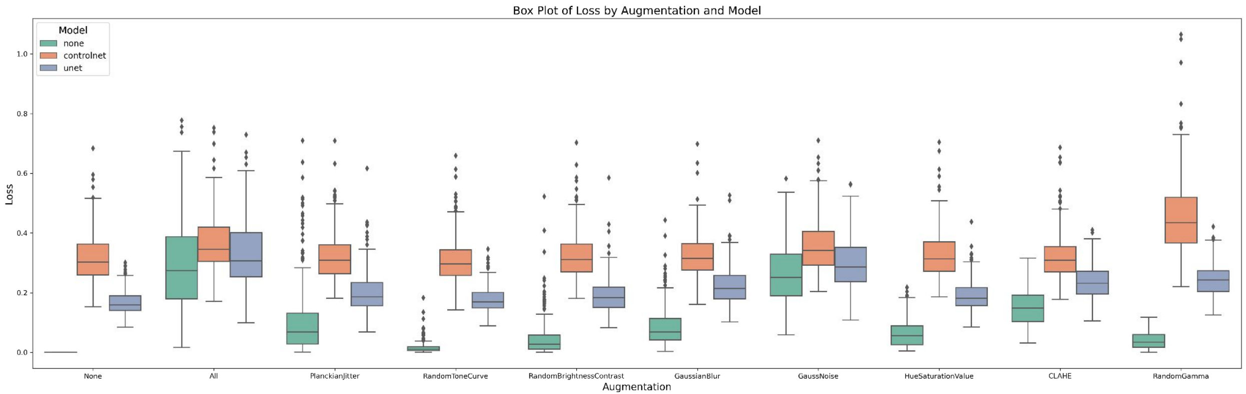
Goal: Provide quality image restoration, comparable to expensive softwares or professional work without using a simple network architecture

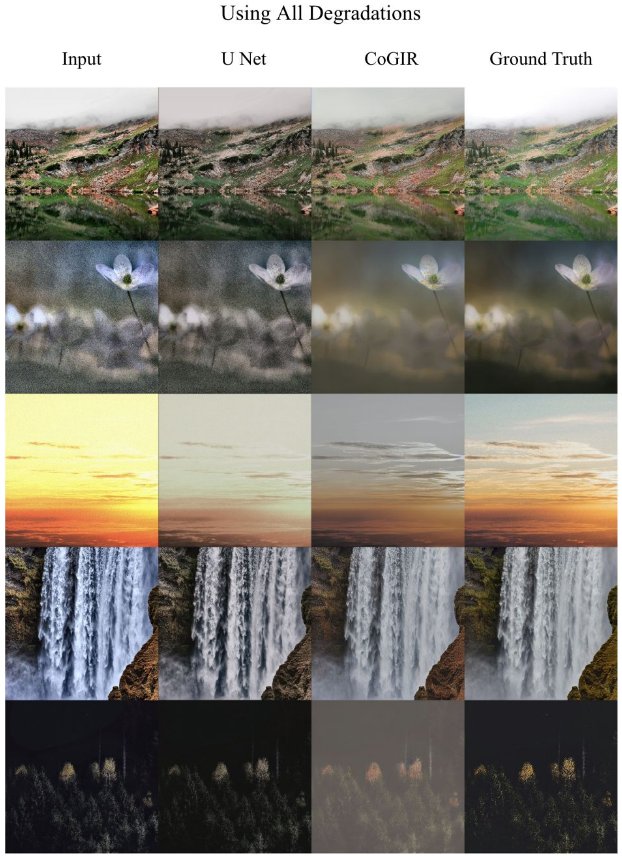
Summary: Train an image-to-image ControlNet integrated with a pre-trained text-to-image stable diffusion model. Get training data from the Unsplash lite dataset, augment images using the Albumenations library, generate a sample prompt through an open-source LLM, attempt to recreate ground truth



Main Results & Comparison







Conclusion

To conclude, there are many areas of improvement for the ControlNet model developed in this paper, including more accurate evaluation for the task at hand using different quantitative evaluation functions and a more advanced baseline model for comparison. Furthermore, future work should perform more experiments to see the impact that the provided text prompts have on the final "style" of the ControlNet model.

With this said, the CoGIR was able to produce impressive results as a lightweight model. While the actual produced images may not have been similar to the target image as calculated by LPIPs + L1 loss, they were able to match the average aesthetic within the dataset. CoGIR was able to seamlessly handle common degradations that images on the internet may face, and was particularly good at handling more extreme input, which the baseline model could not effectively do.

Compact diffusion models for Cifar-10

Manan Abbott (mhabbott@umass.edu),
Varun Jhaveri (vjhaveri@umass.edu)

Project #58

Summary:
This work addresses the computational challenges in **Diffusion Probabilistic Models (DPMs)** for CIFAR-10 image generation. By leveraging **structured pruning** and **knowledge distillation**, we developed MiniDiff-C10, through a self supervised approach, an efficient model with reduced inference time and memory footprint while retaining high image generation quality.

Motivation:

- Diffusion models like DDPMs deliver state-of-the-art generative performance but demand substantial computational resources
- Real-world deployment in resource-constrained environments requires efficient alternatives

Background:

- Structured pruning systematically removes unimportant weights, reducing model complexity
- Knowledge distillation transfers knowledge from a larger "teacher" model to a smaller "student" model to retain performance
- This project combines both techniques to optimize diffusion models for practical use cases

Baseline

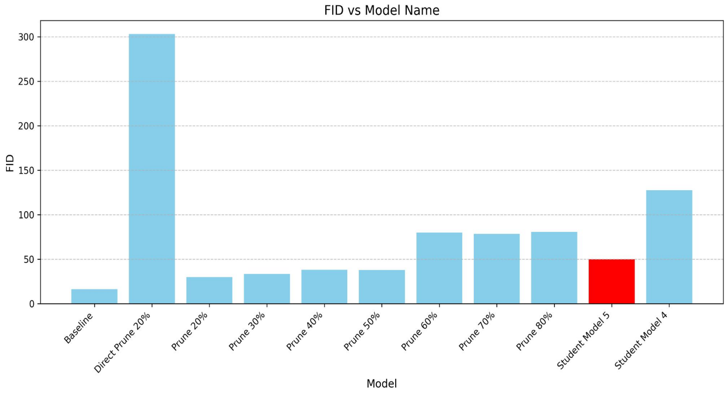
Minidiff-C10

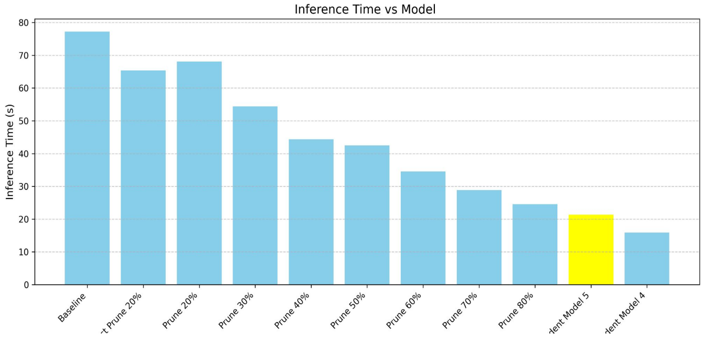
Inference Efficiency: MiniDiff-C10 achieved a significant reduction in inference time and memory usage (21.35 s, 36 Mb) compared to the baseline DDPM (77.25 s, 137 Mb).

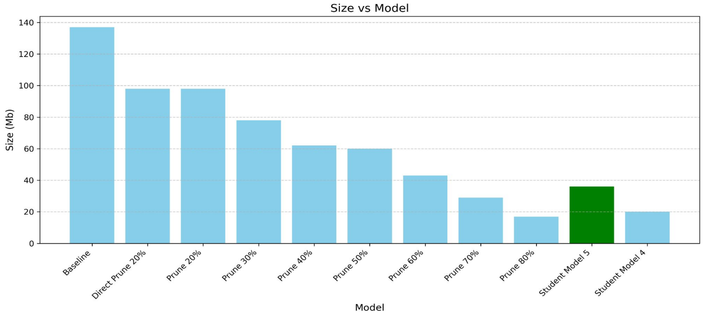
FID Score Performance: With 20% pruning and fine-tuning after knowledge distillation, MiniDiff-C10 maintained an FID score (50) close to the baseline (16.25), demonstrating 68% image quality retention.

Comparison to Prior Work:

- Unlike generic pruning strategies, our approach integrates structured pruning with distillation, ensuring the model retains essential generative capabilities.
- Previous work has focused on pruning individual weights or using only KD; our combined approach optimizes for both efficiency and generative performance.







Key Takeaways:

- MiniDiff-C10 demonstrates that structured pruning, paired with knowledge distillation, is a viable strategy for making DDPMs more efficient.
- The trade-off between model size and image quality can be effectively managed, allowing for practical use in resource-constrained environments.

Future Directions:

1. **Architectural Enhancements:** Explore deeper UNet models with larger input images to capture finer details.
2. **Low-Rank Approximation:** Incorporate rank-reduction techniques for additional parameter efficiency.
3. **Broader Applications:** Apply these methods to other generative models (e.g., GANs, VAEs) and datasets.

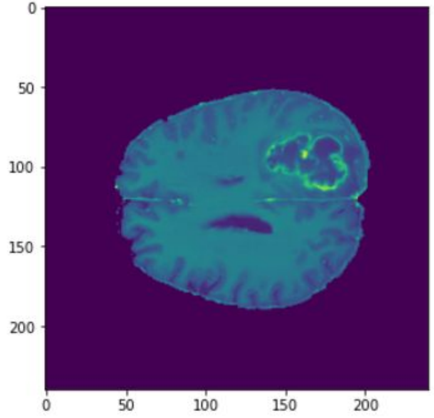
Fine-Grained Brain Tumor Segmentation Using an Attention-Enhanced U-Net Model

Rishitha Guttapalle Mohan (rguttapallem@umass.edu),
Sreya Talasila (stalasila@umass.edu),
Thashmitha B S (tbs@umass.edu)

Project #61

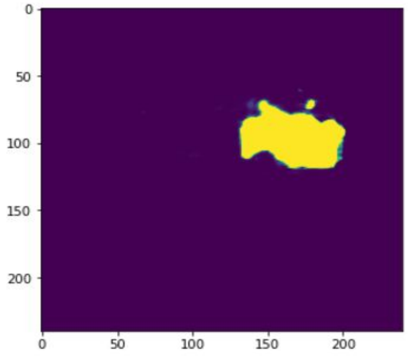
Motivation and Background

- Brain tumor segmentation is vital for diagnosis and treatment, but manual methods are time-consuming and inconsistent, necessitating automated solutions. This research aims to enhance segmentation accuracy and efficiency with attention-based U-Net models.
- MRI is widely used for brain tumor assessment, but accurate segmentation is challenging due to tumor variability and complex patterns. While U-Net shows promise, it struggles with weak tumor boundaries, and attention mechanisms help improve segmentation by focusing on relevant regions.



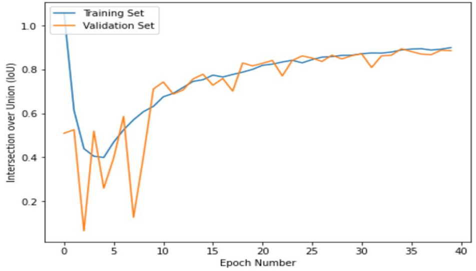
Summary

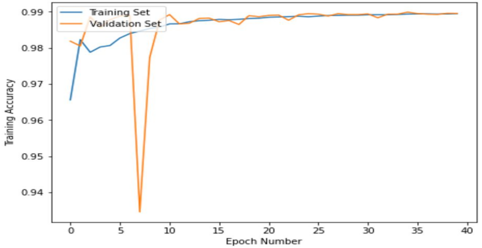
This study leverages the BraTS 2020 dataset, comprising multimodal MRI scans (T1, T1ce, T2, FLAIR), to develop an attention-based U-Net model. The architecture incorporates attention gates to focus on tumor regions, improving segmentation accuracy. Evaluation metrics show competitive results, and the model demonstrates significant improvements over the baseline U-Net.

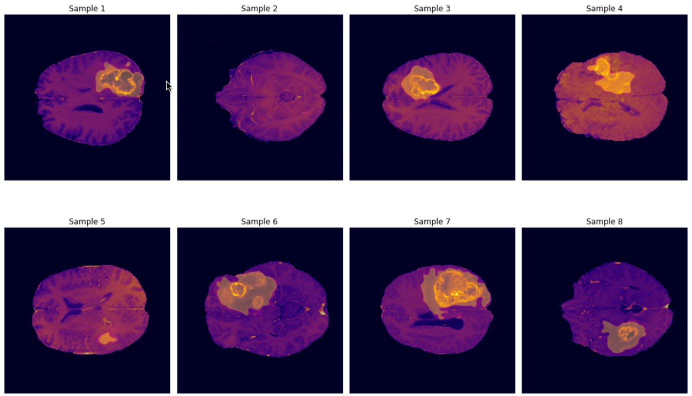


Comparison to Prior Work and Main Results

- **Segmentation using U-Net**: demonstrated U-Net models significantly improves focus on tumor boundaries, particularly in handling complex and irregular structures in medical imaging.
- **Robust Handling of Class Imbalance**: The preprocessing pipeline and multimodal MRI integration address challenges such as class imbalance and heterogeneous tumor characteristics more effectively than earlier methods.
- **State-of-the-Art Metrics**: The model achieves competitive Dice coefficient and IoU metrics, surpassing earlier traditional U-Net and models in precision and robustness.
- **Our Contribution**: Unlike traditional U-Net models, our architecture incorporates attention gates, which dynamically prioritize relevant spatial regions, leading to enhanced segmentation accuracy.

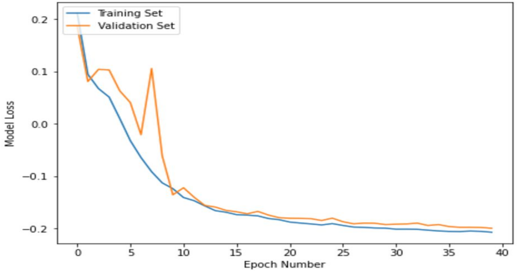






Evaluation Metrics and Results

- Dice Similarity Coefficient (DSC)
- Intersection over Union (IoU)
- Sensitivity
- Specificity



Conclusion

This study introduces an attention-enhanced U-Net model for brain tumor segmentation using the BraTS 2020 dataset.

- **Improved Segmentation Accuracy**:
  - Attention mechanisms significantly enhance the model's ability to accurately segment tumors, particularly in complex cases with subtle boundaries and heterogeneous features.
- **Focused Attention**:
  - The model prioritizes critical regions in MRI scans, improving segmentation precision and demonstrating robustness against noise and tumor variability.
- **Impact on Medical Imaging**:
  - The promising results underline the potential of attention-based architectures to address challenges in medical image segmentation, specifically in brain tumor analysis.
- **Clinical Implications**:
  - This work paves the way for more accurate, automated tumor detection, offering the potential for more efficient and reliable diagnosis in clinical practice.
- **Future Directions**:
  - Future research will focus on:
    - Optimizing model performance.
    - Exploring lightweight architectures to reduce computational overhead.
    - Leveraging **transfer learning** using pre-trained models on similar medical datasets.

EmotionLess: Optimizing Emotional Style Transfer Through Customized Loss Functions

Aaryan Mohindru (amohindru@umass.edu),
Antara Parmar (antaraparmar@umass.edu)

Project #63

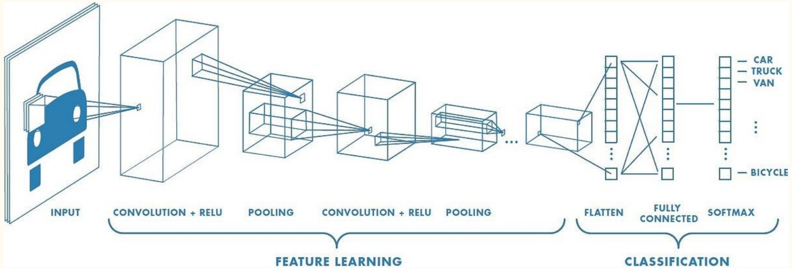
Introduction

Background: Existing style transfer models struggle to balance maintaining image's original style while transferring a certain emotion, especially when working with many emotions.

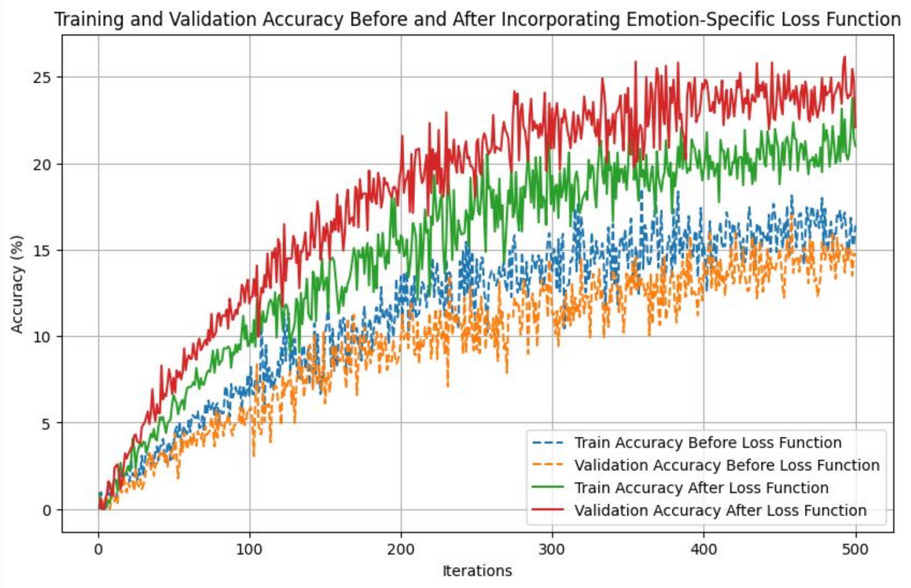
Motivation: Improve the balance between transference of emotion and maintaining original style with an increased number of emotions, promote more emotion-based art that society can resonate with

Objective: Introduce an additional custom loss function focused on reducing differences between the base and stylized images to the style transfer to improve performance by preventing the original style from being lost.

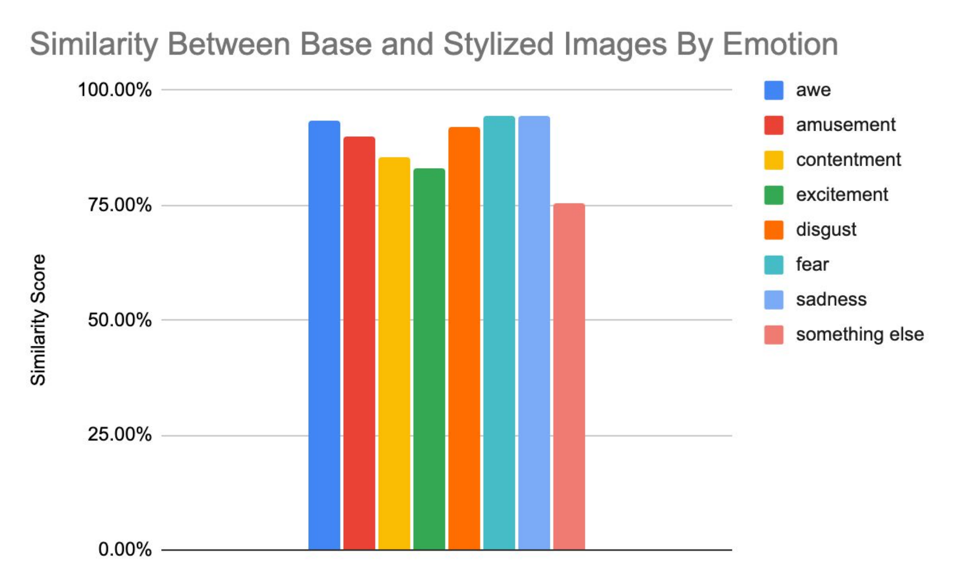
Models/Datasets: Trained image-to-emotion classifier on 80k+ images from Artemis dataset. Generated 1000+ images through style transfer.



Results

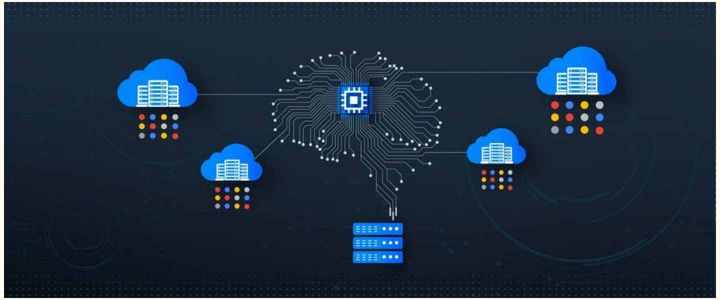


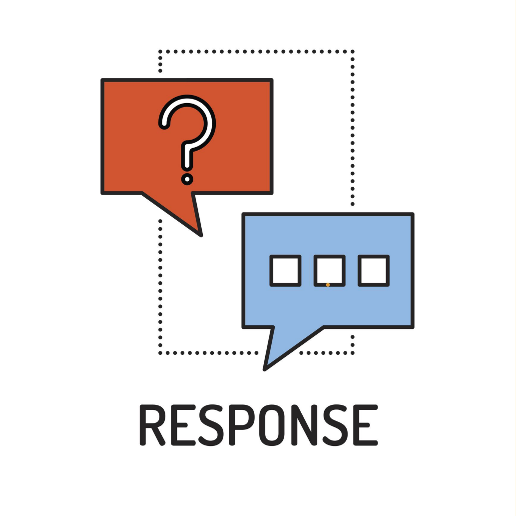
Training and Validation Accuracy Before and After Incorporating Emotion-Specific Loss Function
Accuracy (%)
Iterations
0
5
10
15
20
25
0
100
200
300
400
500
Train Accuracy Before Loss Function
Validation Accuracy Before Loss Function
Train Accuracy After Loss Function
Validation Accuracy After Loss Function



Similarity Between Base and Stylized Images By Emotion
Similarity Score
100.00%
75.00%
50.00%
25.00%
0.00%
awe
amusement
contentment
excitement
disgust
fear
sadness
something else

Conclusion





Challenges:

- Managing compute resources
- Obtaining survey results

Key Findings:

- Implementing a custom loss function helps emotion-based style transfer models perform generally well on a wider scale of emotions
- Classifying between many emotions can be balanced with retaining style

Future Improvement:

- Enhance data collection methods and dataset selection
- Incorporate additional models trained on diverse datasets for emotion classification.

Data augmentation with diffusion models using latent space exploration

Harshavardhan Reddy Rajanala (hrajanala@umass.edu),
Vin Tyagi (vtyagi@umass.edu)

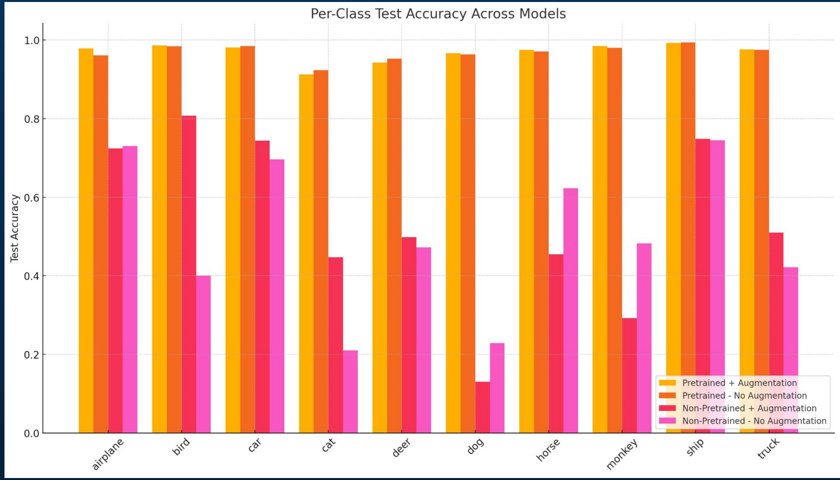
Project #73

Introduction

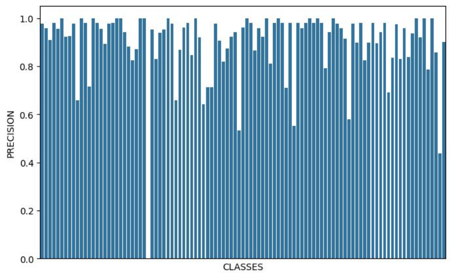
Summary:
we implement CLIP in order to develop image embeddings in between two images. Additionally, our additional goal is to be able to explore possible perturbation within the latent space through single image embeddings in order to fine-tune the output of our image generation process

Motivations:
Our investigation into this challenge is specifically motivated by the potential of latent feature space exploration in generating synthetic data that can effectively test and enhance the robustness of classification models.

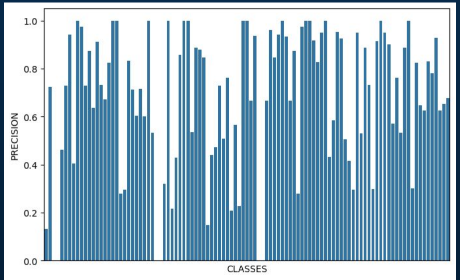
Results



Per-Class Test Accuracy Across Models
Test Accuracy
Pretrained + Augmentation
Pretrained - No Augmentation
Non-Pretrained + Augmentation
Non-Pretrained - No Augmentation
airplane
bird
car
cat
deer
dog
horse
monkey
ship
truck



PRECISION
CLASSES



PRECISION
CLASSES

Image 1
t = 0.00
t = 0.11
t = 0.22
t = 0.33
t = 0.44
t = 0.56
t = 0.67
t = 0.78
t = 0.89
t = 1.00
Image 2

Conclusion

Analysis of results:

SLERP’s benefits are less pronounced when models already have strong feature representations through pre-training. On the other hand, diffusion models were not able to uplift the performance of the previous pre-trained model, as they oftentimes perturbed the original image to the point of semantic loss, and harmed the quality of the data.

What now?

We can now attempt to provide optimal values of strength and similarity parameters used to find semantically different images. Our goal then would be to find a balance between maintaining the content of the image while generating elements different from the source.

Visual grounding in unseen domains

Jyothirmai Puram (jpuram@umass.edu),
Keerthy Kaushik Dasoju (kdasoju@umass.edu),
Supraj Bachawala (sbachawala@umass.edu)

Project #75

Quick Summary

- Visual grounding: linking natural language expressions to corresponding regions in an image. (think object detection with natural language)
- We investigate the ability of pretrained visual grounding models to adapt to new, unseen domains
- Evaluates one-stage and transformer-based grounding methods.
- We curated a dataset choosing retail stores as our domain
- Investigated Low-Rank Adaptation (LoRA) for fine-tuning

Experiments and Results

	Accuracy@0.5		Mean IOU	
	Model A	Model B	Model A	Model B
Baseline	23.52	21.43	34.39	37.24
With LoRA	26.31	27.59	35.16	38.72
Cluttered	11.42	16.56	29.77	21.41
Uncluttered	27.37	23.63	38.21	39.10

Table 1

	Accuracy@0.5	Mean IOU
Baseline	25.67	35.34
With LoRA	26.13	39.22
Cluttered	17.42	15.16
Uncluttered	31.25	41.73

Table 2

- **Table 1** shows our results with Yang et al. [22], pre-trained models on Retail Dataset. Model A: Pretrained BERT on RefCOCO, Model B: Pretrained BERT on Flickr

- **Table 2** shows our results using TransVG on Retail Datase

Conclusion

- Pre-trained models struggle with unseen domains (Acc@0.5 < 30%).
- Visual clutter significantly impacts accuracy and IoU.
- LoRA offers marginal improvement; alternative domain adaptation methods needed.
- Domain-specific pre-training may improve generalization.
- Future work: explore advanced adaptation techniques and data augmentation.

Neuro-symbolic solver (NASR++)

Frank Chiu (fchiu@umass.edu),
James Barrett (jamesbarrett@umass.edu),
Shamanth Kuthpadi Seethakantha (skuthpadi@umass.edu)

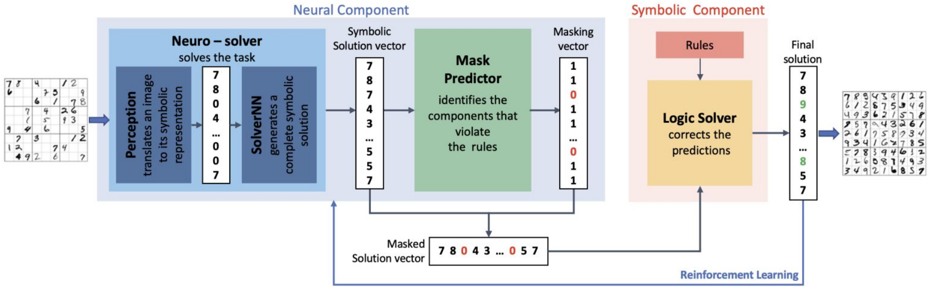
Project #76

Summary, Motivation, Background

- Motivation:
  - Improve the pipeline’s (NASR) performance on challenging datasets without RL to ultimately reduce the computational cost
  - Cornelio et al. trained multiple models (instead of a general purpose model) for each sudoku datasets of different characteristics.
- Architecture
- Performance
- Training Time

	big_kaggle		minimal_17		multiple_sol		satnet_data	
Perception	99.64	74.56	99.84	87.70	99.48	65.70	99.32	63.20
SolverNN	100-98.33	62.68	100-61.56	0.00	100-93.72	46.70	100-94.84	24.00
Mask-Predictor	99.92-99.71		99.54-35.26		99.02-76.12		99.90-96.06	
Neuro-Solver		47.03		0.00		28.80		14.40
NASR w/o RL		80.02		1.59		60.00		76.40
NASR with RL		**84.24**		**87.00**		**73.00**		**82.20**
NASR-Heur.Mask		73.11		66.99		55.60		42.80

	big_kaggle	minimal_17	multiple_sol	satnet_data
Perception	108.84	57.71	11.68	14.14
SolverNN	85.58	41.97	9.01	26.51
Mask-Predictor	81.43	40.15	8.87	128.98
RL	121.07	241.31	24.86	16.44
NASR w/o RL	275.85 (108.84)	139.83 (57.71)	29.56 (11.68)	169.63 (128.98)
NASR with RL	396.92 (229.91)	381.14 (299.02)	54.42 (36.54)	186.07 (145.42)
Symb. Baseline	108.84	57.71	11.68	14.14
SatNet	1151.25	582.13	41.66	108.51
NaurASP-Infer	108.84	57.71	11.68	14.14
NeurASP	94.34	91.53	99.96	~ 90



Tables and architecture sourced from Cornelio et al.

Results

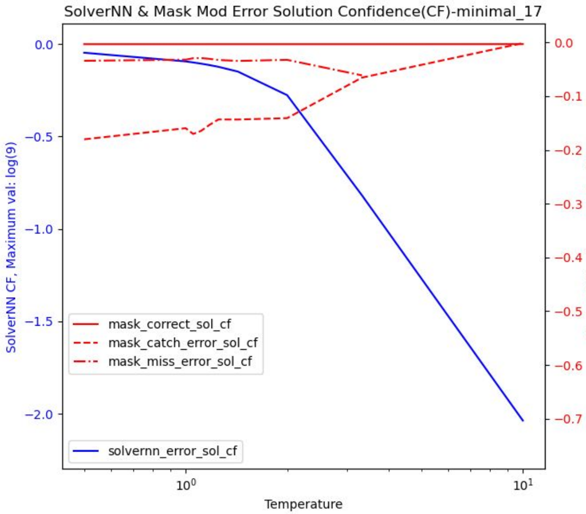
- Robustness
  - Two models: model trained on big_kaggle (***Mb***), model trained on minimal_17 (***Mm***)
  - Evaluated ***Mb*** on ***Mm*** and vice versa → found that both the models were not robust
  - Composed a representative & robust dataset to train a new model
    - Could not complete training in the interest of time
- Constraint Loss
  - Functions: cosine similarity, KL divergence, Jensen-Shannon divergence
  - Training Time
  - Performance
- Modular Interactions between SolverNN and Mask Predictor
  - SolverNN confidence vs Mask Predictor confidence & performance
  - Manipulate latent representations: Softmax with vary temperature
  - Confidence: average negative entropy of distributions across Sudoku boards

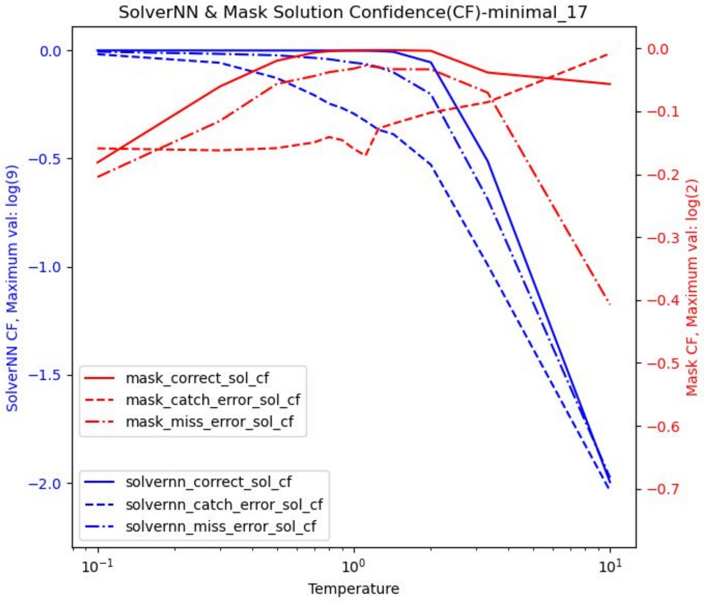
Training Time	Big Kaggle	Minimal 17
Standard	200 epochs, 145 mins	200 epochs, 73 mins
Cosine	138 epochs, 240 mins (timed out)	200 epochs, 120 mins

Performance (correctness on all cells)	Big Kaggle	Minimal 17
Standard	98.42%	61.65%
Cosine	97.84%	61.38%

Conclusion

- Now that we have a composed dataset that includes a wide range of hints per board, we hope to train a new model with the NASR pipeline on this dataset
- We expected to…
  - Decrease training time
  - Improve the performance
- Unfortunately, we were not able to achieve either of these goals
- Future work, GNN?





Deep learning for plant health monitoring

Greeshma Gudapati (ggudapati@umass.edu),
Meghana Sai Veligatla (mveligatla@umass.edu)

Project #81

Problem Statement:

- Plant diseases represent a significant challenge to global agriculture, causing substantial crop losses and financial difficulties for farmers. Traditional manual inspection methods are inefficient, inconsistent, and impractical for large-scale operations, highlighting the need for an automated and scalable approach to ensure timely and accurate disease detection.

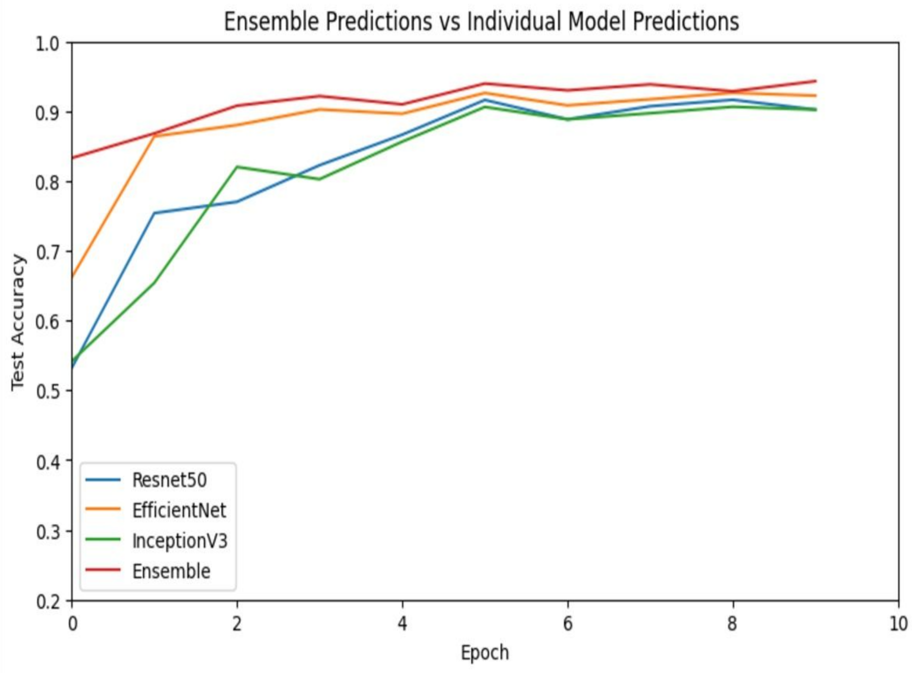
Goal

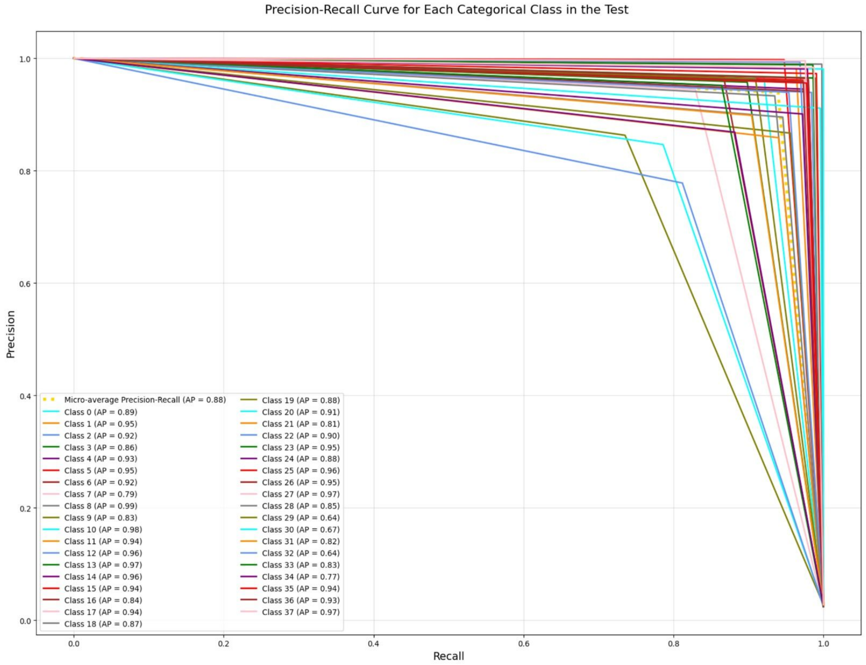
- To demonstrate that the DeepLeaf ensemble model, combining ResNet50, EfficientNet, and InceptionNet, outperforms individual models in automating plant disease detection, achieving superior classification accuracy, reliability, and scalability.

Approach

- DeepLeaf employs an ensemble of advanced CNN models (ResNet50, EfficientNet, InceptionNet) combined with LSTM-based time-series analysis to enhance plant disease classification and progression tracking, supported by data augmentation techniques to address class imbalances and improve model generalization.

Results


Ensemble Predictions vs Individual Model Predictions
Test Accuracy
Epoch
Resnet50
EfficientNet
InceptionV3
Ensemble


Precision-Recall Curve for Each Categorical Class in the Test
Precision
Recall
Micro-average Precision-Recall (AP = 0.88)
Class 0 (AP = 0.89)
Class 1 (AP = 0.95)
Class 2 (AP = 0.92)
Class 3 (AP = 0.86)
Class 4 (AP = 0.93)
Class 5 (AP = 0.95)
Class 6 (AP = 0.92)
Class 7 (AP = 0.79)
Class 8 (AP = 0.99)
Class 9 (AP = 0.83)
Class 10 (AP = 0.98)
Class 11 (AP = 0.94)
Class 12 (AP = 0.96)
Class 13 (AP = 0.97)
Class 14 (AP = 0.96)
Class 15 (AP = 0.94)
Class 16 (AP = 0.84)
Class 17 (AP = 0.94)
Class 18 (AP = 0.87)
Class 19 (AP = 0.88)
Class 20 (AP = 0.91)
Class 21 (AP = 0.81)
Class 22 (AP = 0.90)
Class 23 (AP = 0.95)
Class 24 (AP = 0.88)
Class 25 (AP = 0.96)
Class 26 (AP = 0.95)
Class 27 (AP = 0.97)
Class 28 (AP = 0.85)
Class 29 (AP = 0.64)
Class 30 (AP = 0.67)
Class 31 (AP = 0.82)
Class 32 (AP = 0.64)
Class 33 (AP = 0.83)
Class 34 (AP = 0.77)
Class 35 (AP = 0.94)
Class 36 (AP = 0.93)
Class 37 (AP = 0.97)

Future Scope:

- DeepLeaf can be improved by integrating explainable AI for enhanced interpretability, optimizing it for deployment on edge devices, training it on more diverse real-world datasets, and incorporating advanced methods like attention mechanisms to boost accuracy and scalability for widespread agricultural use.

Conclusion :

- DeepLeaf establishes the effectiveness of ensemble learning by achieving superior accuracy over individual models in plant disease detection. With its integration of LSTM-based temporal analysis, it provides a comprehensive solution for monitoring disease progression, offering a transformative tool for precise and sustainable agricultural practices.

Occlusion-aware module for 2d object detection for autonomous vehicles

Alauddeen Shaikh (alauddeensha@umass.edu),
Amit Sarker (asarker@umass.edu),
Roshini Pulishetty (rpulishetty@umass.edu)

Project #86

Quick Snapshot of Our Project

- **Motivation**: *Object detection* is a crucial component of critical downstream tasks like *object tracking for autonomous driving* and *motion planning of robots*. Yet, occlusion poses a severe challenge for detections.

  Robustness to occlusions is required for safe autonomous vehicles.

- **Goal**: Improve the performance of existing object detection framework in occluded cases, especially localizing objects in autonomous driving scenarios.

- **Approach**: We propose a two-stage object detector that enhances YOLO v5m with ResNet-18 secondary classifier and further, show that overlap with initial predictions (IoU) and aspect ratios serve as an important information for predicting object dimensions when occluded.

Results

YOLO v5m detections

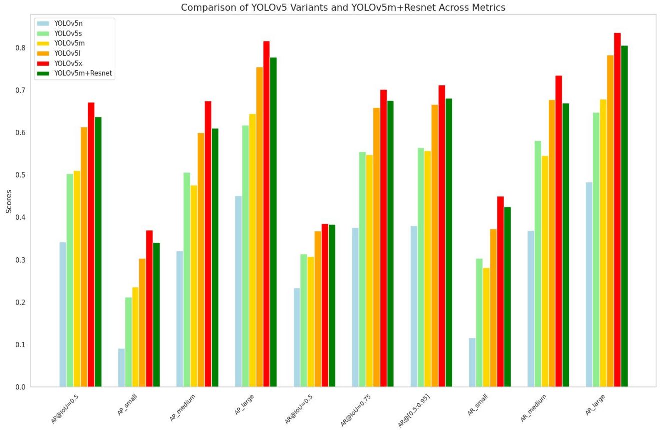
YOLO v5m+ResNet detections

YOLO v5m + ResNet and occlusion-aware module detections

Conclusion

Summary of Contributions:

- Developed a two-stage detection pipeline enhancing YOLO v5m with a ResNet-18 secondary classifier.
- Implemented an occlusion-aware bounding box enhancement method to improve localization under occlusions.
- Reduced false positives and misclassifications without significant computational overhead.



Key Results:

- Significant improvement in detection accuracy across all vehicle categories (car, bus, motorcycle, truck).
- Achieved higher Average Precision (AP) and Average Recall (AR) compared to baseline YOLO v5 models.
- Enhanced detection of small and medium-sized vehicles and better handling of occluded objects.

Future Work:

- Explore integration of more advanced classifiers and machine learning techniques.
- Extend approach to additional object categories and dynamic scenes.

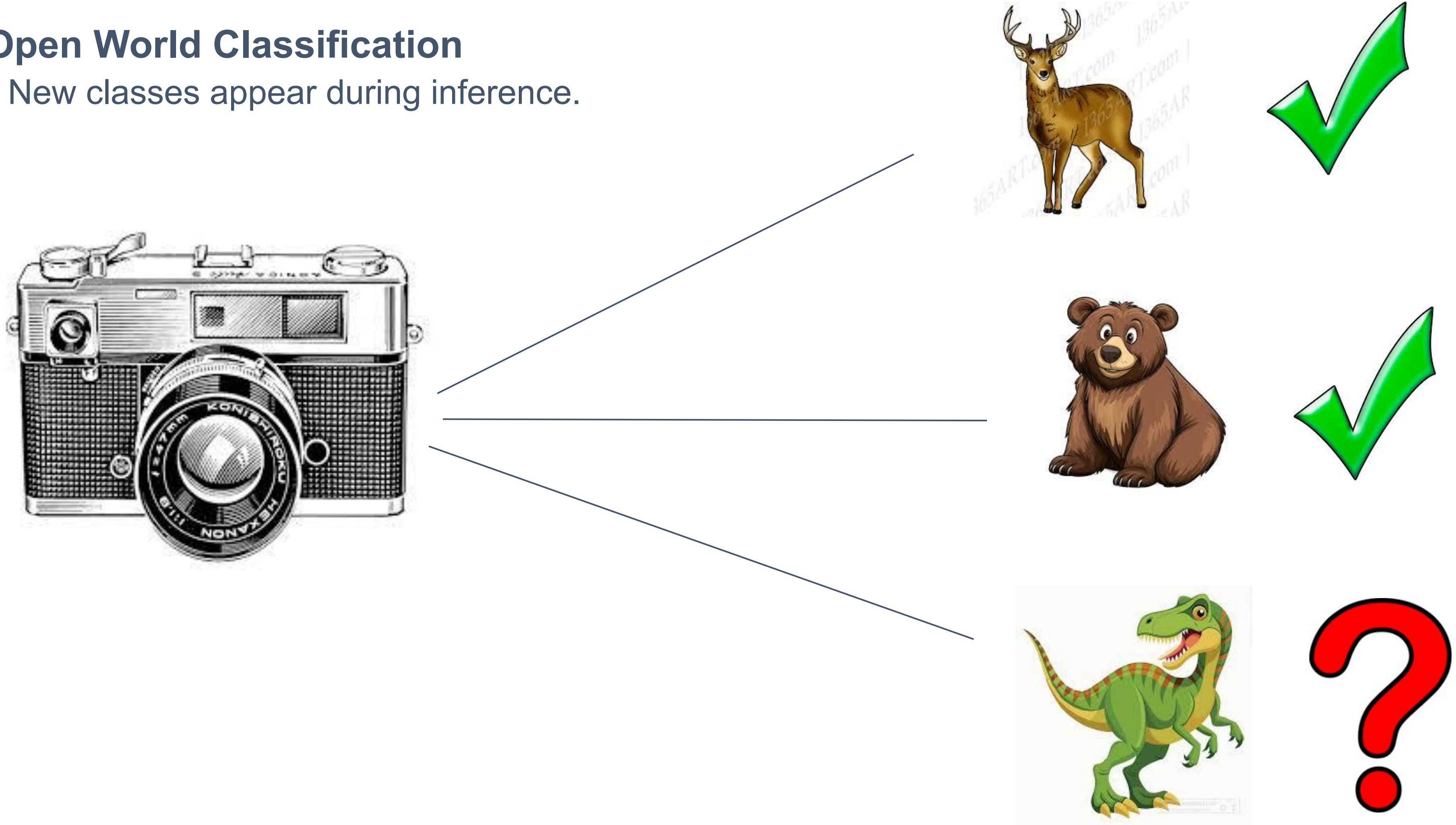
Evaluating Open World Agents using Generative Models

Justin Clarke (jclarke@cs.umass.edu)

Project #89

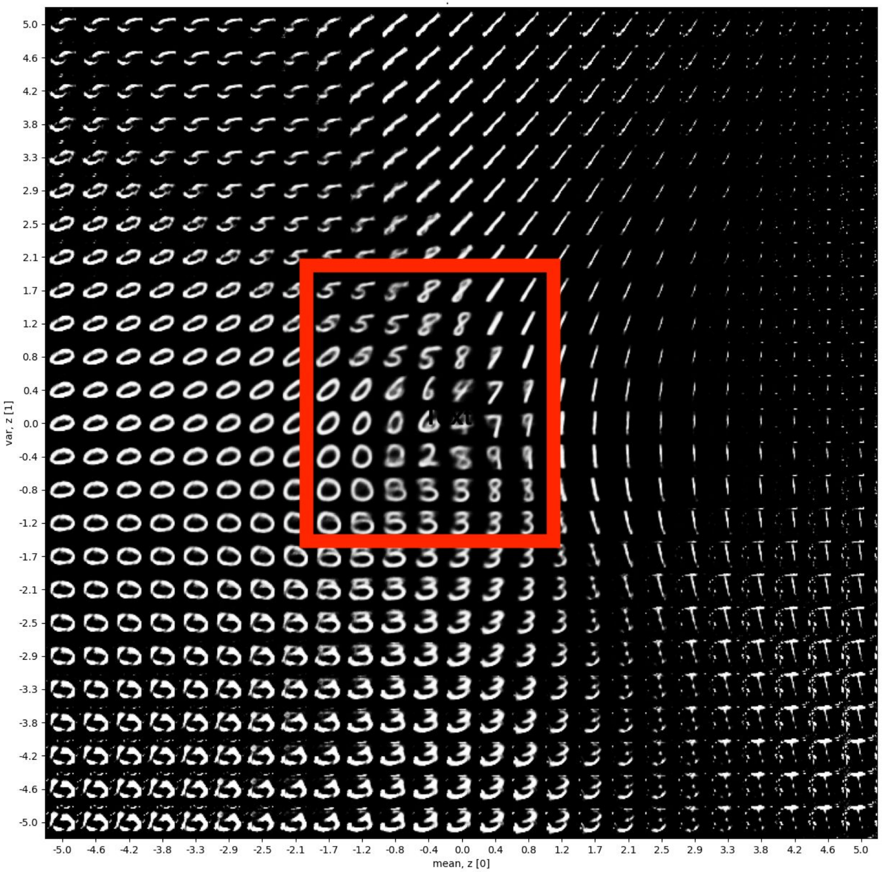
Open World Classification

- New classes appear during inference.



Generative models

- Learned latent representation



Results