CSE578 Data Visualization (Fall 2019)
Project Proposal

Title: DeepVID: Deep Visual Interpretation and Diagnosis for Image Classifiers via Knowledge Distillation [Accepted for PacificVis'19]

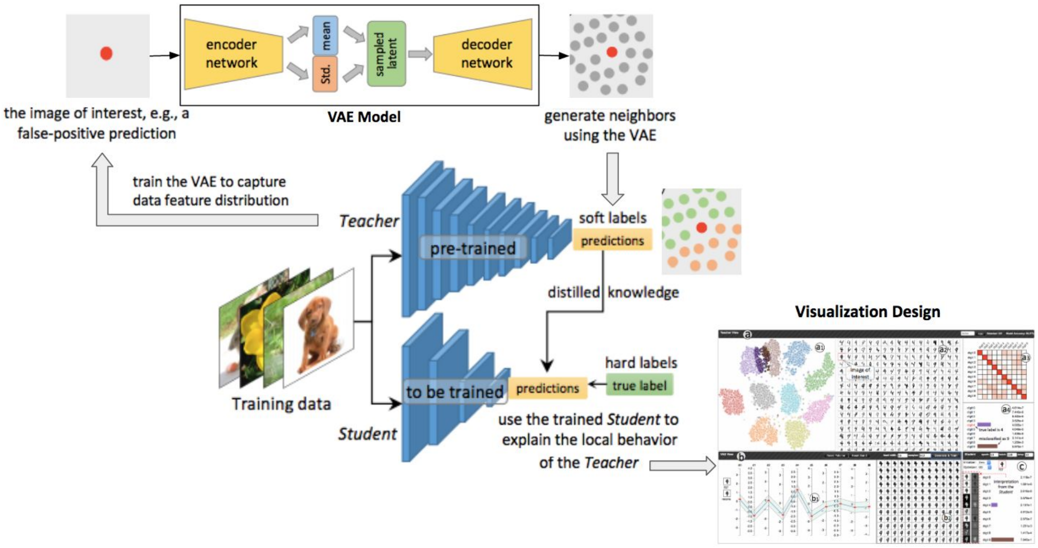
Team Members: 1) Srihari Jayakumar (1217164455) 2) Sandipan Choudhuri (1213340672) 3) Tanveer Ghousia (1217099793)
4) Abhishek Muralikrishnan (1217200439) 5) Shailaja Sampat (1213348433)

Executive Summary of DeepVID:

DeepVID is a tool for visual interpretation of deep models based image classifiers, inspired from the fact that with more layers we lose the tractability of neural models. It leverages the Student-Teacher method to create a small interpretable classifier that mimics the original cumbersome model using knowledge distillation. Finally, they design interactive visualization to get deeper insights about misclassified examples, along with numerous options of trainable parameters to observe their impact in near real-time.

Design:

Figure 1: Detailed system design of DeepVID


encoder network
mean
Std.
sampled latent
decoder network
VAE Model
the image of interest, e.g., a false-positive prediction
generate neighbors using the VAE
train the VAE to capture data feature distribution
Teacher
pre-trained
soft labels
predictions
distilled knowledge
Training data
to be trained
predictions
hard labels
true label
Student
use the trained Student to explain the local behavior of the Teacher
Visualization Design

Techniques:

The technique addresses a classification task where a sophisticated network (Teacher) is trained for classifying instances in a dataset, and is to be interpreted. It also involves two other network models: (a) a pre-trained Variational Auto-Enoder (VAE), which is a self-supervised encoding-decoding model that learns a low-dimensional latent representation of an input and tries to generate it back as output, with the help of the bottleneck latent representations and (b) a linear Student model. Now, given a false-positive data (based on the Teacher model's output), the pretrained VAE is utilized to generate neighbors around the given data. The generated instances are sent as input to the Teacher network for its knowledge distillation. This is followed by training the student network using the generated instances and their soft-labels from the Teacher model. Eventually, an interactive interpretation that throws light on "why the Teacher model misclassified the data sample" can be obtained by visualizing the Student network.

Interactions:

1. **Teacher View:** Overview of test data and Teacher's prediction on them
   - t-SNE view: lasso selections over a 2D scatter plot that alters displays in image-grid view
   - Confusion matrix view: clickable matrix cells that alters displays in image-grid view
   - Image-grid view: scrollable display of images, mouse hover maps to a corresponding point in t-SNE view, mouse click updates the probability distribution view for the clicked data instance
   - Probability distribution view: N/A
2. **VAE View:** Interactive exploration of feature space and generated neighbours for the datapoint of interest
   - Parallel coordinates plot: dragging interaction to visualize the effect of different latent vectors over an image, dragging interaction to specify perturbation band for generating neighbours, facilitates morphing between two images to see the underlying reason for misclassification
   - Neighbours view: N/A
3. **Student View:** zoomed image on hover

Input boxes, buttons and drop-down widgets are common interactions among all views.

Proposed improvements in existing implementation:

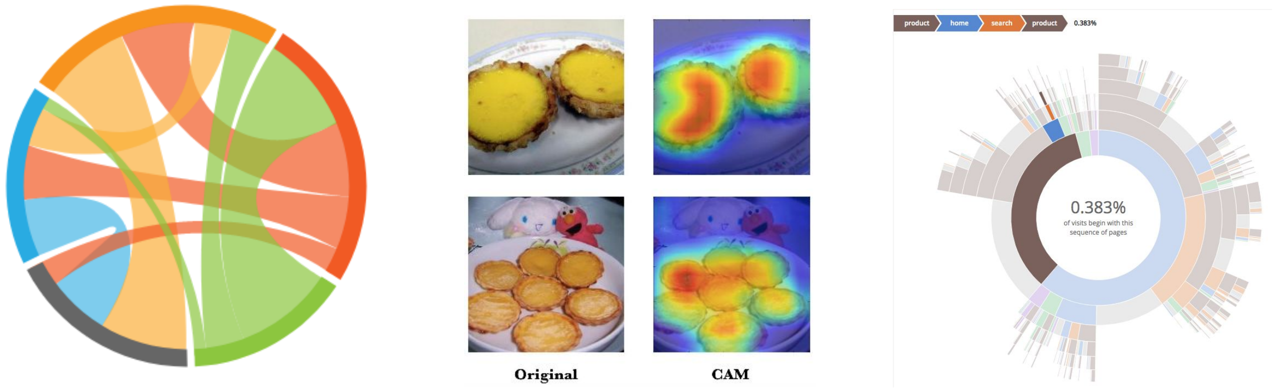
1. Display a progress bar to a user while models are retrained with new parameters
2. Support more optimizers, mechanisms for hyperparameter tuning and/or regularizations
3. For datasets with very high latent space, VAE view becomes untidy therefore we will try to simplify it
4. Instead of lasso selection interaction from cluttered 2D scatterplot, we would like to see if 3D t-SNE view separates classes better
5. In addition to confusion matrix, we believe that Chord diagram (Figure 2a) can provide a better relative comparison of false positives and false negatives in classification.

Additional Datasets:

1. CIFAR-10: Classification of 10 daily life objects/animals [airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck]
2. Fashion-MNIST: Classification of 10 fashion objects [t-shirt, trouser, pullover, dress, coat, sandal. shirt, sneaker, bag, ankle boot]

Proposed Extension:

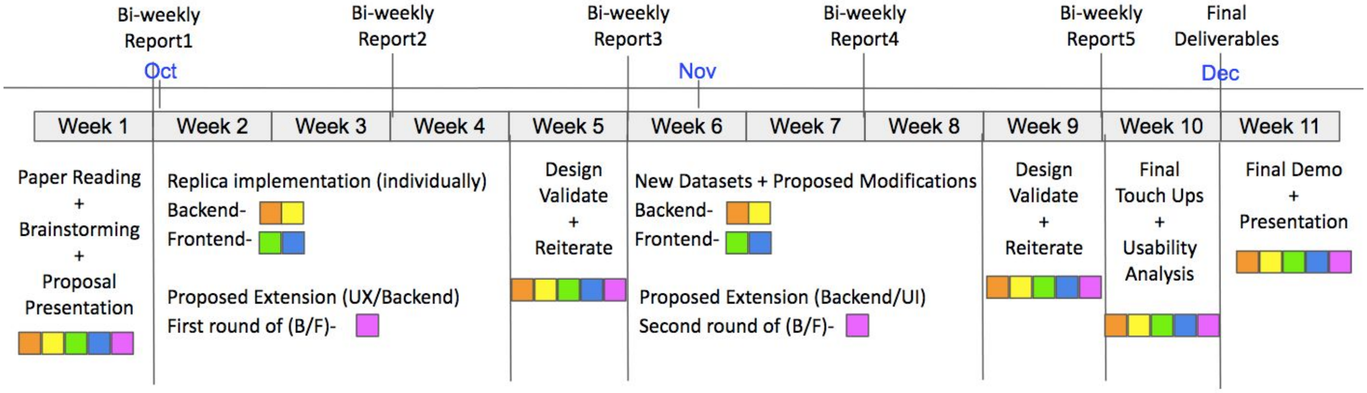
1. Class Activation Maps (CAMs) are useful in discriminating image regions in classifiers to identify a specific class in the image. We will extend existing visualization to incorporate CAMs (as in Figure 2b) which will provide better explanations of reasoning behind misclassified examples.
2. Classification can be thought of as a hierarchical decision making from various features observed in an image. To represent this idea in form of visualizations, we believe Sequence Sunburst graph will be a suitable choice (as in Figure 2c).



(a) (b) (c)

Figure 2: (a) Chord Diagram for better representation of false positives and false negatives (b) Class Activation Maps (c) Sequence Sunburst graph to represent hierarchical decision making in classification process

Implementation timeline and work allotment for team members:



Bi-weekly Report1 | Bi-weekly Report2 | Bi-weekly Report3 | Bi-weekly Report4 | Bi-weekly Report5 | Final Deliverables

Oct | Nov | Dec

Week 1 | Week 2 | Week 3 | Week 4 | Week 5 | Week 6 | Week 7 | Week 8 | Week 9 | Week 10 | Week 11

Week 1: Paper Reading + Brainstorming + Proposal Presentation 🟧🟨🟩🟦🟪

Weeks 2–4: Replica implementation (individually)
Backend- 🟧🟨
Frontend- 🟩🟦
Proposed Extension (UX/Backend)
First round of (B/F)- 🟪

Week 5: Design Validate + Reiterate 🟧🟨🟩🟦🟪

Weeks 6–8: New Datasets + Proposed Modifications
Backend- 🟧🟨
Frontend- 🟩🟦
Proposed Extension (Backend/UI)
Second round of (B/F)- 🟪

Week 9: Design Validate + Reiterate 🟧🟨🟩🟦🟪

Week 10: Final Touch Ups + Usability Analysis 🟧🟨🟩🟦🟪

Week 11: Final Demo + Presentation 🟧🟨🟩🟦🟪

🟧 Srihari (DL) 🟨 Sandipan (DL) 🟩 Tanveer (D3) 🟦 Abhishek (D3) 🟪 Shailaja (DL/D3)

Each person will write one of the bi-weekly reports, based on updates from everyone through a brief meeting before submission. Final report, presentation and demo video, we will all put together by dividing work uniformly among ourselves.