Evaluation, Interpretability, and Deployment of an Image Classification Model for Autism Prediction

1. Problem Statement

In the domain of computer vision, particularly in healthcare, accurately classifying images can significantly impact diagnosis and intervention strategies. This report focuses on a deep learning model designed to classify images of children as either 'Autistic' or 'Non_Autistic.' Additionally, it addresses the interpretability of the model's predictions and the deployment of the model in a user-friendly interface.

2. Objectives

- To train a convolutional neural network (CNN) for classifying images into 'Autistic' and 'Non_Autistic' categories.
- To evaluate the performance of the trained model using metrics such as accuracy, precision, recall, and F1-score.
- To visualize and interpret the model's predictions using Grad-CAM, LIME, Integrated Gradients, and other techniques.
- To deploy the model using Gradio, providing an interactive interface for end-users to upload images and receive predictions.

3. Methodology

- Data Preparation: The dataset was divided into training, validation, and test sets. Images were loaded and their dimensions verified.
- Data Visualization: A bar chart was created to visualize the class distribution in the training set.
- Model Training: A CNN model (ResNet34) was trained using the Fastai library. The learning rate was optimized, and the model was fine-tuned for 30 epochs.
- Model Evaluation: The model's performance was evaluated through a confusion matrix, classification report, and ROC curve.
- Interpretability Techniques:
    a) Grad-CAM: Visualizes the regions of an image that influenced the model's predictions.
    b) Grad-CAM++: An enhanced version of Grad-CAM providing more precise localization.
    c) LIME: Explains individual predictions by perturbing the input image and observing changes in predictions.
    d) Integrated Gradients: Attributes the output of a model to its input features.
    e) Saliency Maps: Visualizes the gradient of the output with respect to the input image.

f) DeepDream: Enhances and visualizes features learned by the model.

- Deployment: The model was deployed using Gradio, allowing users to upload images and receive real-time predictions.

4. Screenshots

Screenshots of the model training process, evaluation metrics, interpretability techniques, and the Gradio interface are provided below:

4.1 Training Process

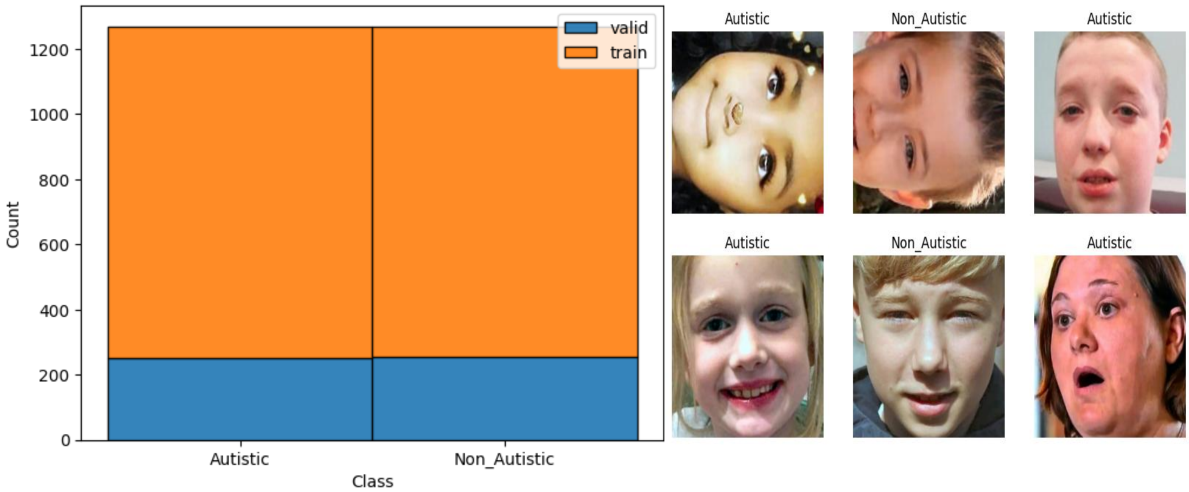
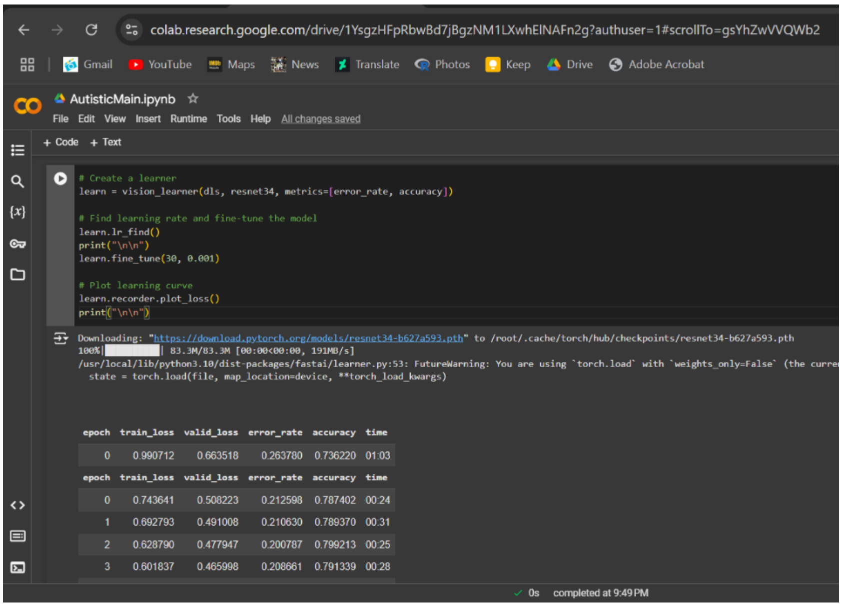
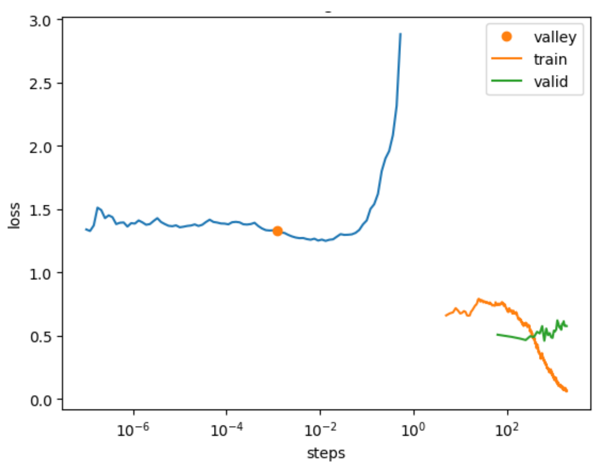


Fig. Train and Validation Set Distribution



1	0.692793	0.491008	0.210630	0.789370	00:31
2	0.628790	0.477947	0.200787	0.799213	00:25
3	0.601837	0.465998	0.208661	0.791339	00:28
4	0.542295	0.496641	0.198819	0.801181	00:28
5	0.478927	0.489571	0.185039	0.814961	00:26
6	0.431195	0.531292	0.214567	0.785433	00:29
7	0.362889	0.518232	0.169291	0.830709	00:33
8	0.359500	0.576325	0.190945	0.809055	00:27
9	0.304362	0.460520	0.171260	0.828740	00:26
10	0.270472	0.556806	0.179134	0.820866	00:25
11	0.245062	0.504839	0.177165	0.822835	00:25
12	0.219119	0.520598	0.169291	0.830709	00:25
13	0.218850	0.488191	0.147638	0.852362	00:28
14	0.193200	0.482852	0.155512	0.844488	00:29
15	0.173854	0.540194	0.159449	0.840551	00:26
16	0.164083	0.534212	0.145669	0.854331	00:26
17	0.156142	0.550627	0.159449	0.840551	00:28
18	0.126580	0.621396	0.167323	0.832677	00:27
19	0.123979	0.596423	0.153543	0.846457	00:28
20	0.110367	0.566541	0.153543	0.846457	00:25
21	0.103977	0.570429	0.145669	0.854331	00:27
22	0.100438	0.547056	0.157480	0.842520	00:24
23	0.095879	0.592966	0.157480	0.842520	00:24
24	0.089032	0.593104	0.163386	0.836614	00:27
25	0.082330	0.614751	0.165354	0.834646	00:22
26	0.079709	0.578509	0.159449	0.840551	00:26
27	0.075474	0.582551	0.153543	0.846457	00:26
28	0.070697	0.576414	0.155512	0.844488	00:32
29	0.064113	0.576404	0.155512	0.844488	00:27

Fig. Screenshots of Epochs ran



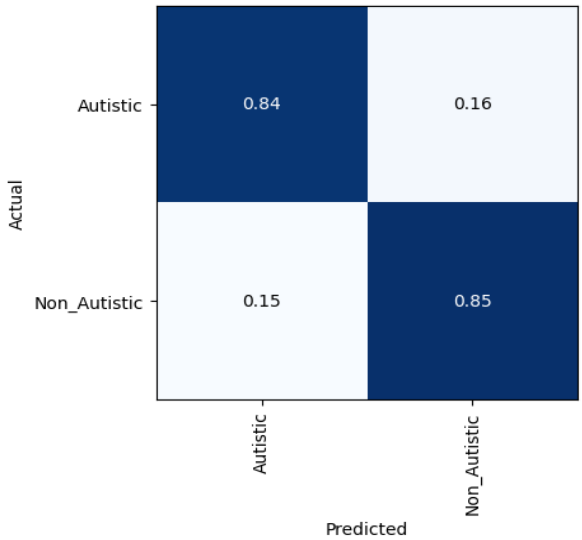
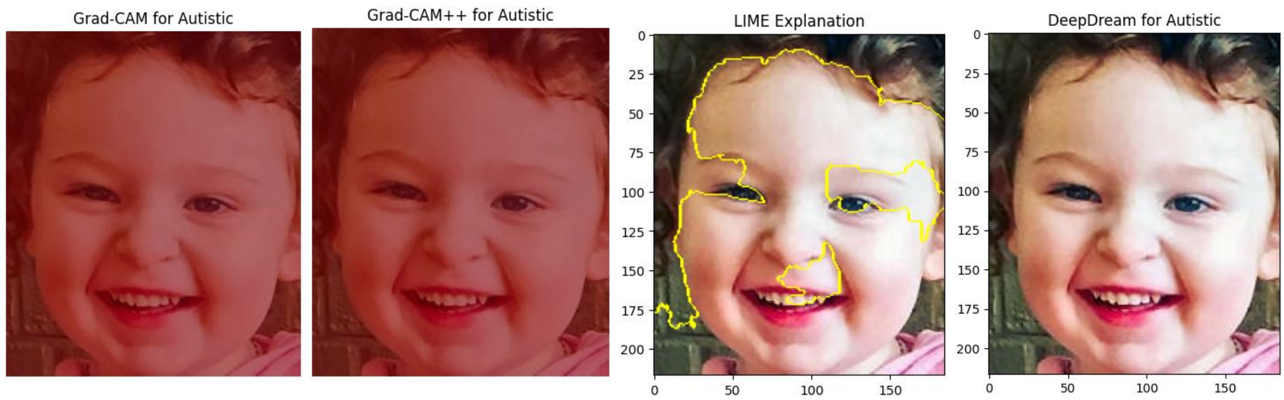
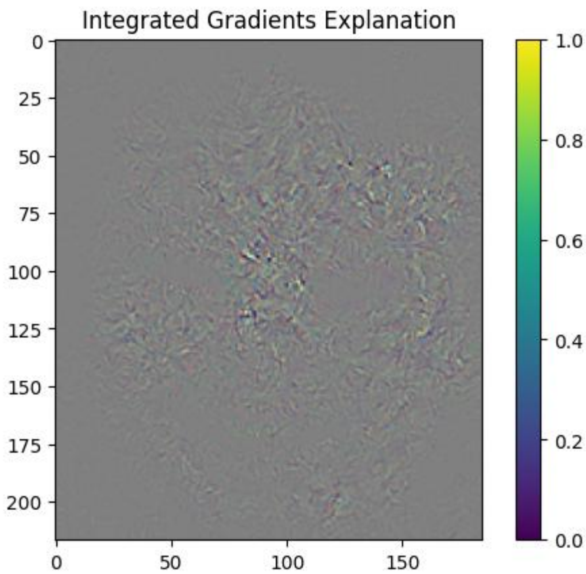


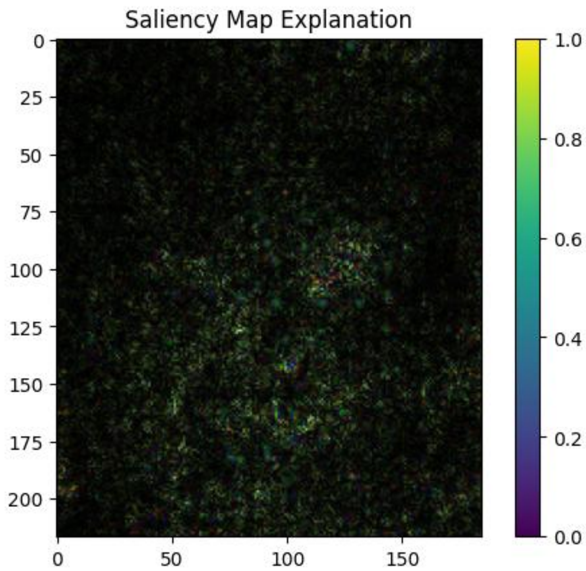
Fig: Learning Curve

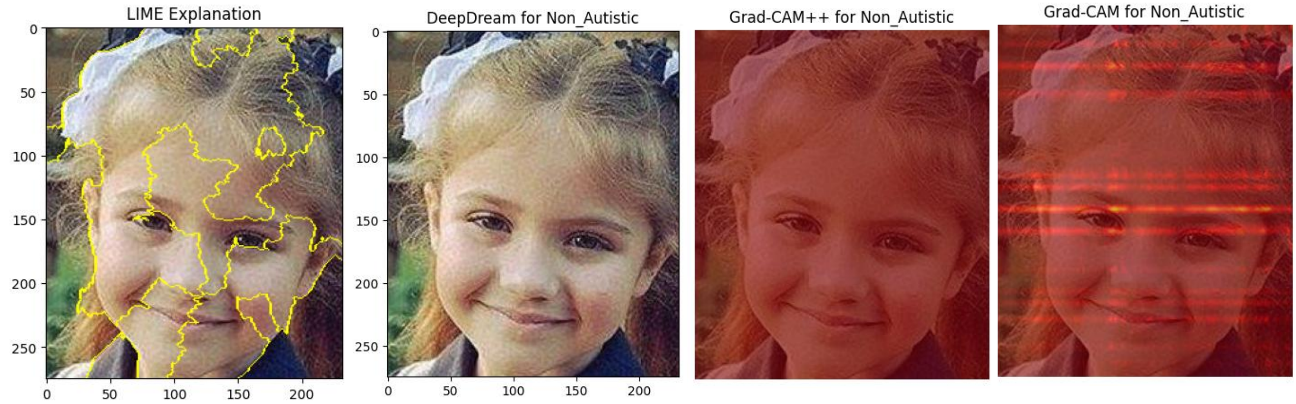
Fig: Confusion Matrix

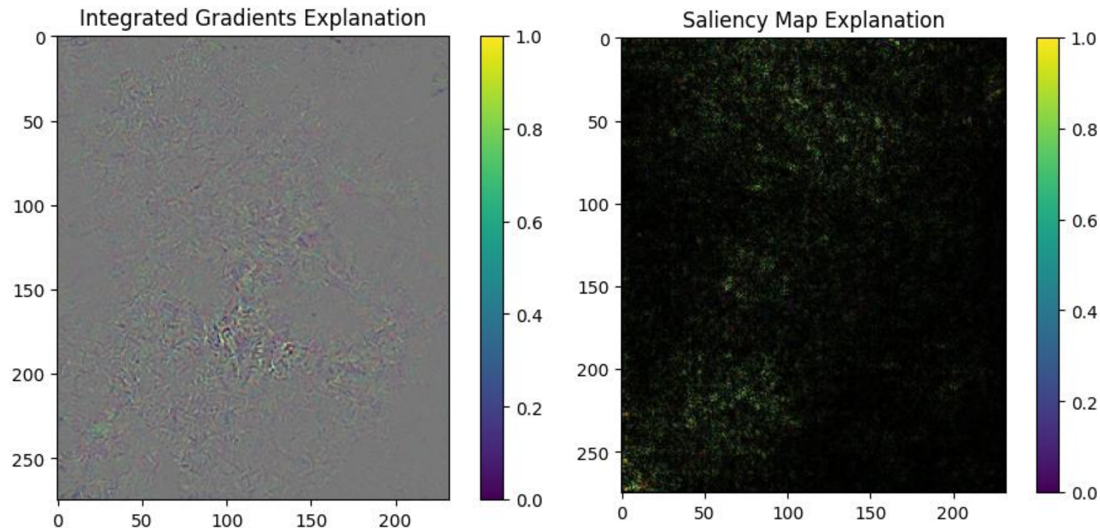
4.2 Interpretability techniques on an image from each category







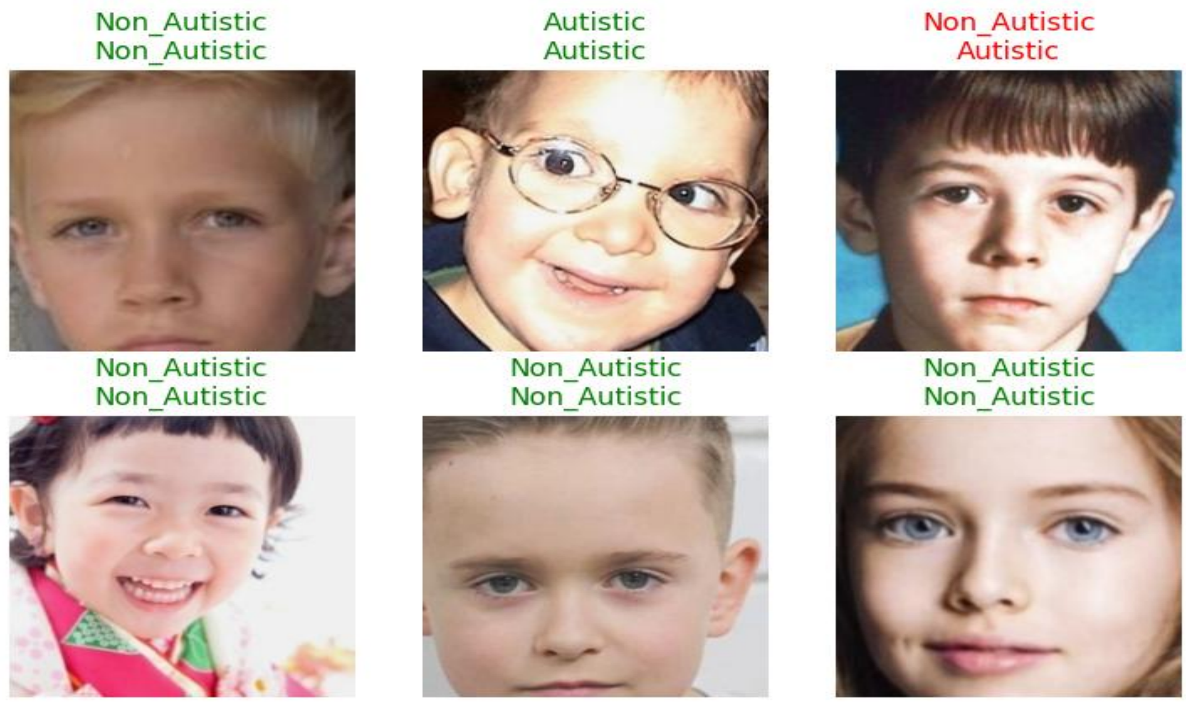




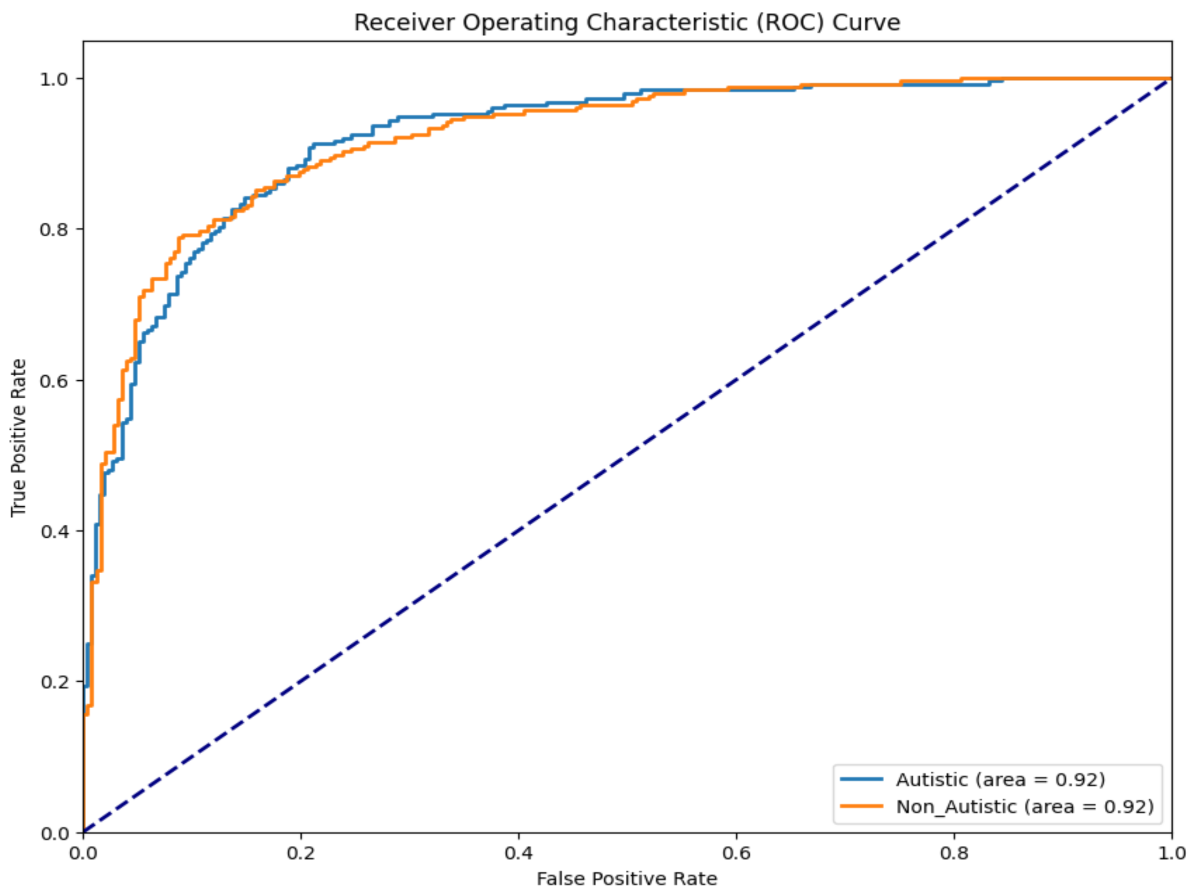
5. Results

The trained model achieved an accuracy of 84% on the test set, with the following metrics:

	PRECISION	RECALL	F1-SCORE	SUPPORT
AUTISTIC	0.85	0.84	0.84	252
NON_AUTISTIC	0.84	0.85	0.85	256
ACCURACY			0.84	508
MACRO AVG	0.84	0.84	0.84	508
WEIGHTED AVG	0.84	0.84	0.84	508



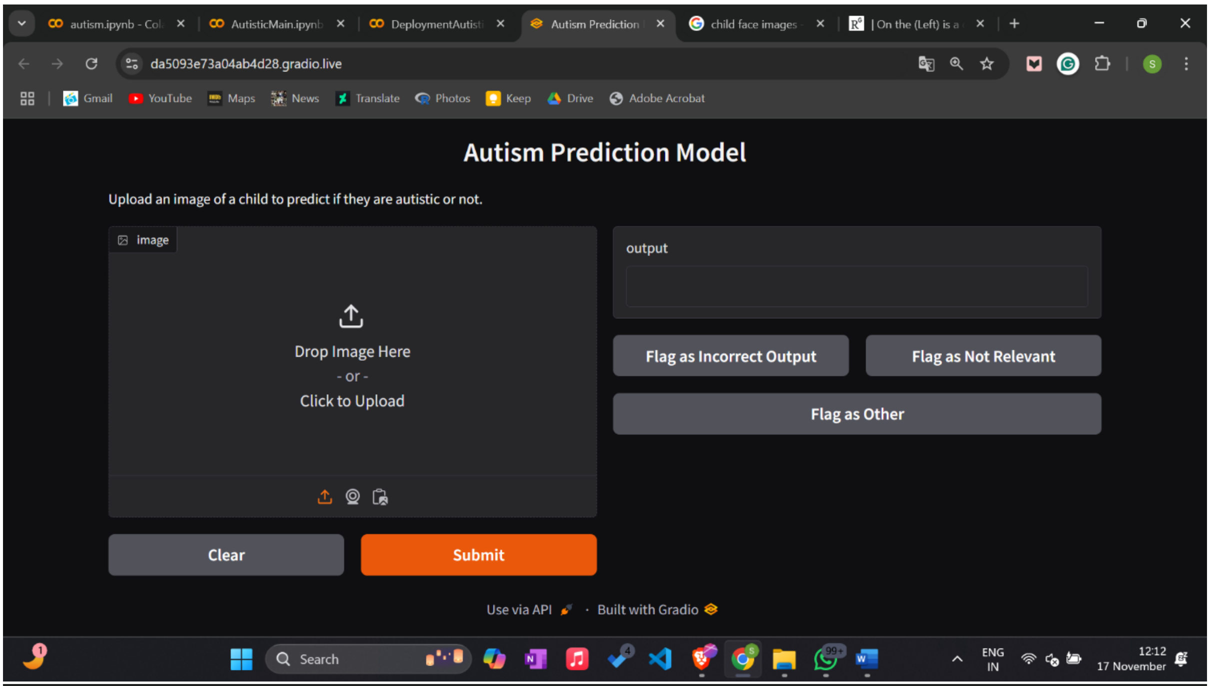
The ROC curve illustrated the model's ability to distinguish between the two classes effectively.

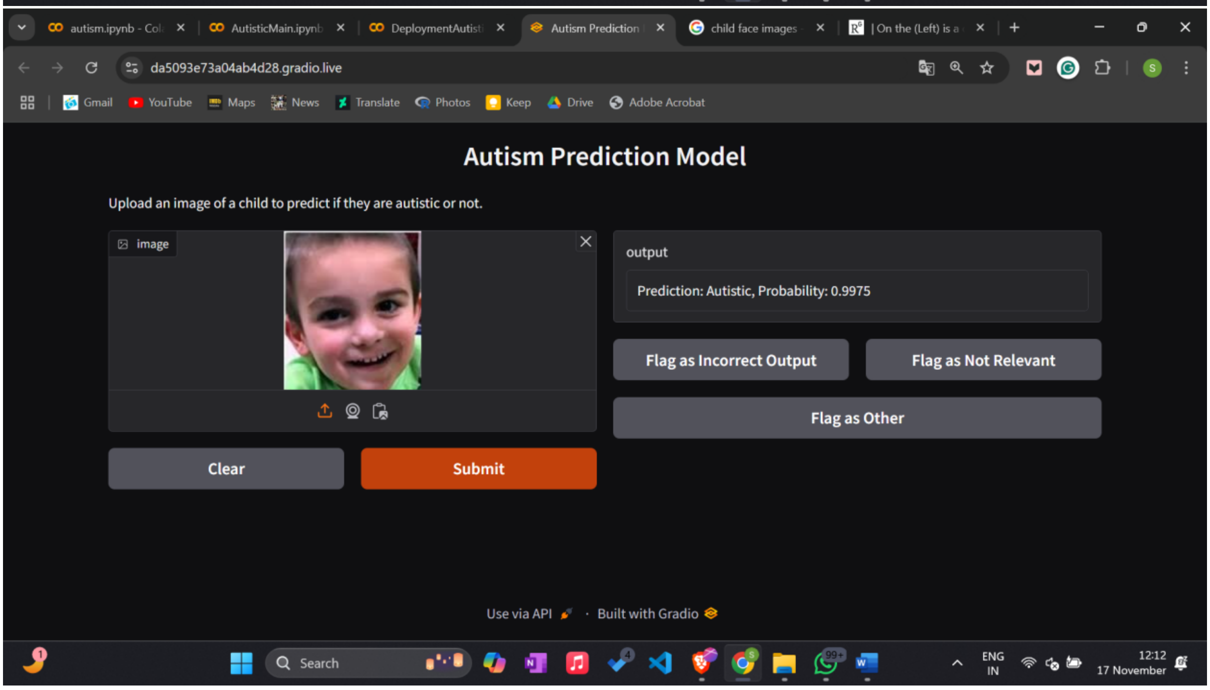


7. Deployment

The model was deployed using the Gradio library, creating an interactive web interface where users can upload an image of a child. The model processes the image and returns a prediction along with the associated probability. The interface allows users to flag predictions as "Incorrect Output," "Not Relevant," or "Other," enabling continuous improvement of the model based on user feedback. The deployment code is as follows:

Gradio Interface:





8. Conclusion

The image classification model demonstrated satisfactory performance in distinguishing between 'Autistic' and 'Non_Autistic' categories. The interpretability techniques employed provided valuable insights into the model's decision-making process, allowing for a better understanding of which features influenced predictions. The deployment of the model through Gradio offers a user-friendly interface, enhancing accessibility for end-users. This work emphasizes the importance of model interpretability and deployment in building trust in AI systems, especially in sensitive applications.

9. Self Reflection

Reflecting on the work done, I found the process of training the model, applying interpretability techniques, and deploying the model to be both challenging and rewarding. The hands-on experience with various libraries and frameworks enhanced my understanding of deep learning, model evaluation, and deployment. I learned the significance of visualizing model predictions and the importance of interpretability in machine learning. Moving forward, I aim to explore more advanced interpretability techniques and their applications in real-world scenarios. This project has strengthened my skills and knowledge in machine learning, and I look forward to applying these concepts in future projects.