Classifying The Progression of Diabetic Retinopathy

Mark Cubi
GSAS Computer Science Department
Fordham University
New York, United States
mcubi@fordham.edu

***Abstract*—Diabetic Retinopathy is a leading cause of vision loss, and automated severity classification from retinal images can support clinical screening. A lightweight convolutional neural network was trained on 5,649 retinal images across five ordinal severity levels using class-weighted loss and data augmentation to address class imbalance. The model achieved a top-1 accuracy of 81% and top-2 accuracy of 95%. Tree-based models trained on CNN embeddings did not improve performance.**

I. Introduction

Diabetic retinopathy, according to the Mayo Clinic, is a complication of diabetes that causes damage to blood vessels in light-sensitive tissue at the back of the eye, called the retina [1]. It can develop in anyone that has type 1 or 2 diabetes. In advanced stages, the eye tries to grow new blood vessels, but they do not develop correctly and can bleed easily. The risk of developing it increases the longer someone has diabetes or if they fail to manage their blood sugar well. In milder cases, it may cause no symptoms or slight vision problems. In severe cases, it can lead to blindness. Symptoms include spots or dark strings floating in their sight, blurred vision, changes in vision, dark or empty areas in vision, or vision loss.

There are two stages of the condition. Non-proliferative diabetic retinopathy is the more common form. New blood vessels are not growing. Proliferative diabetic retinopathy is the more severe form. Damaged blood vessels close off, causing new irregular blood vessels to grow in the retina. These blood vessels can bleed into the clear jelly-like matter that fills the center of the eye, called the vitreous.

Those with diabetes are recommended to have yearly eye examinations to check for retinopathy. Those who are pregnant can also develop diabetes before or during pregnancy and are encouraged to have additional eye examinations during pregnancy. Early detection is crucial, so treatment can begin before the retina is damaged beyond repair. Manual screening using retinal imaging requires trained specialists and is time consuming. An automated approach, especially in areas without proper medical resources, can be very valuable. The goal of this project is to develop a convolutional neural network capable of predicting the severity of diabetic retinopathy from digital retinal imaging. We will also test if creating a pipeline that feeds the CNNs feature space into a Random Forest or XGBoost model can increase predictive power.

II. Dataset

The dataset used in this project is a combination of two publicly available diabetic retinopathy datasets. The first is a Kaggle diabetic retinopathy dataset published by Omar Essa with 1,986 retinal images, each sized at 256x256 pixels with three RGB channels [2]. The second one is a subset of a 2019 Diabetic Retinopathy Challenge dataset, curated by Sovit Ranjan Rath, which has 3,663 retinal images at a resolution of 224x224 pixels and three RGB channel [3]. The images from both dataset are visually different in terms of color, contrast and saturation, which may help the model generalize better by exposing it to a wider range of imaging conditions. Combined, these datasets provide 5,649 total samples.

Each image is labeled using one of five categories: healthy, mild, moderate, severe, and proliferative. A clinician assigned each label. “Healthy” represents individuals without diabetic retinopathy, while “mild”, “moderate”, and “severe” correspond to increasing severity levels within non-proliferative diabetic retinopathy. “Proliferative” represents patients with the advanced proliferative diabetic retinopathy. These labels are inherently ordinal rather than categorical, since each class represents an increase in severity of the disease. Methods for handling these ordinal labels will be discussed in the Methodology section.

Although around 5,000 images for a five-class CNN classification is relatively small, with the lack of GPU speed and memory, it was deemed sufficient for the scope of this project. The combined dataset however, is heavily unbalanced, largely driven by the distribution of Rath’s dataset. Healthy and moderate cases far exceed the number of the severe and proliferative cases. Essa’s dataset was much more balanced, but having more training samples was considered more valuable given the limited database size, even if it causes class imbalance. This class balance likely reflects real-world clinical distributions, since severe cases of diabetic retinopathy are rarer.

One notable observation is that the number of moderate samples far exceeds mild and severe samples in the combined dataset. There may be a possibility of labeling bias when the clinician identifies the image, since mild, moderate and severe represent different severities within non-proliferative diabetic retinopathy. Distinguishing between the boundaries of these categories can be ambiguous. This would make the intermediate “moderate” label a simpler, more general label for classifying the disease. Similar biases have been documented in clinical practice. For example, one study found that blood pressure recordings of patients with ischaemic heart diseases ended in the number zero 64% of the time, where it would

be expected to end with zero 10% of the time in practice [4]. This is a very alarming bias especially considering they are recording blood pressures for a very vulnerable group. This suggests a toward simplified or rounded documentation. While there is no way to prove labeling bias for this dataset, it is a factor worth considering when evaluating model predictions and interpreting results.

TABLE I
CLASS DISTRIBUTIONS ACROSS ESSA'S, RATH'S, AND THE COMBINED DATASETS

Class	Essa	Rath	Combined
Healthy	525	1805	2330
Mild	370	370	740
Moderate	599	999	1598
Severe	292	193	395
Proliferate	202	295	585

The dataset was divided into a 75/15/10 train/validation/test set. The training and validation sets were stratified such that each class was represented in proportion to its overall frequency in the combined dataset. As a result, the validation and testing set have the same class imbalance.

TABLE II
CLASS DISTRIBUTION ACROSS TRAINING, VALIDATION, AND TEST SETS

Class	Training	Validation	Test
Healthy	1747	349	234
Mild	555	111	74
Moderate	1198	239	161
Severe	296	59	40
Proliferate	438	87	60

The test set was not artificially balanced since that would either require decreasing the test and validation set significantly by removing healthy and moderate images, or would require removing severe and proliferative images from training. There are already so few proliferative and severe images in training that it would harm the models ability to learn these important classes. Upsampling the minority class (mild, severe, proliferate) by adding augmented versions to the training set was considered, but was avoided to prevent overfitting. Testing on a naturally occurring distribution also allows for results to reflect real-world performance.

III. METHODOLOGY

Given the dataset consists of retinal images, a convolutional neural network was selected as the primary model for classifying diabetic retinopathy severity. The task is inherently challenging due to the subjective nature of grading severity levels. There are also 5 labels and a small unbalanced dataset, so strong results will be difficult to achieve. To contextualize model performance, several baseline metrics were computed. Predicting only the majority class, healthy, will yield a 41.1% top-1 accuracy. Since the labels are ordinal, it is also important to consider the top-2 and top-3 accuracy. Top-2 and top-3 accuracy will identify if the model's second and third most likely predictions were the correct one. It will be a good determining factor for if the model is close to predicting the right severity. Predicting the two most likely classes, healthy and moderate, would yield a 69.4% top-2 accuracy. Predicting the three most likely classes, healthy, moderate and mild, would yield 82.4% top-3 accuracy. A reasonable goal would be to create a model that has 75% top-1, 90% top-2, and 98% top-3 accuracy. It is also important to consider the classification of rarer, more severe cases. Since there are much less samples of these cases, performance will likely be much lower than identifying a healthy case. The goal would be to get a .50 F-1 score for both severe and proliferative cases. Computational speed and memory are also severely limited to process images, preventing the use of deep CNN architectures. Therefore, another goal of the project is to seek an effective light-weight CNN that can be used to get effective results.

In addition to training CNNs directly for classification, the project explored if feature embeddings extracted from the penultimate layer of the CNN can be fed into ensemble methods like Random Forest or XGBoost model to achieve the same performance as a deeper CNN. Since CNNs examine different representations of image features like edges, line textures, and color patterns, their final activation layer can be interpreted as meaningful numerical features. While XGBoost and Random Forest would not be able to predict images from raw pixel data, they can effectively use learned feature vectors.

Data preprocessing was not extensive since images from the dataset were generally high quality. All the images were resized to 224x224 pixels, which only affected the 1,986 images in one of the datasets which were 256x256. Resizing was required so that all the inputs can be processed by the CNN, but it is unideal that many images had to be resized down since there are a lot of fine vascular structures that are important diagnostic features. Reducing the resolution may destroy some of the fine-grain detail that allows the model to identify diabetic retinopathy. The training images were augmented by random rotations of 10 degrees and a random horizontal flip. All pixels were normalized by subtracting the mean and dividing by the standard deviation, scaling all pixel values between -1 and 1.

A light-weight CNN was built and used as a feature extractor. Since the classes were unbalanced, weighting was implemented for each class. Class weights were calculated by taking the total number of samples and dividing by the number of classes multiplied by the number of samples of a particular class. This assigns larger weights to underrepresented classes, penalizing the model more heavily when predictions for this class are incorrect. This weighting encourages the model to generalize better to minority classes, particularly more severe stages of diabetic retinopathy. However, weights were not increased beyond the standard formula because heavily increasing the weight of minority classes can reduce precision and bias the model towards overpredicting rare cases. This would harm generalization. Some weighting does increase the recall of the model, which is especially beneficial since in a practical clinical environment it is more harmful to miss a severe illness than falsely classify an illness as severe.

The structure of the CNN consisted of five convolution layers that used a 3x3 kernel with padding of 1. After each convolution, the model would perform batch normalization to promote quicker training, followed by a ReLU activation function. Each layer also includes 2x2 max pooling, progressively reducing spatial resolution. After passing through the layers, the feature map had a shape of 512x7x7. Before passing the features into a dense layer, adaptive average pooling was used to shrink the spatial resolution to 1x1, producing a 512-dimensional feature vector. This largely decreases the computational cost when passing it to a dense layer of 256 neurons, which then outputs 5 logits for each of the classes. There is a dropout of 0.2 in the dense layer to prevent overfitting. Dropout is relatively small since the model is already lightweight and is unlikely to overfit due to the preprocessing steps taken in augmenting images and including images from two different datasets. Since the problem is multiclassification, cross-entropy loss was used as the loss function to encourage the model to assign high probability to the correct class and penalize confident incorrect predictions.

The CNN was trained over 100 epochs, with early stopping triggered after no improvement after 15 epochs. A larger early-stopping threshold was implemented because the validation set is small and class-weighted. Getting a few severe or proliferative images wrong can greatly influence the loss causing it to be unstable. A higher patience threshold gives the model more room to stabilize and reduces the likelihood of stopping early due to noise and not overfitting.

Early stopping was measured based on an aggregated "score" metric. The score metric was calculated by taking 30% of the top-1 validation accuracy and 70% of the macro average F1 score. Macro F1 was chosen to ensure that performance on minority classes like clinically critical, severe, and proliferate stages contributes equally to the score. Because macro F1 averages the F1 values for each class independently, poor performance on minority classes impacts the score just as much as misclassifying abundant classes such as "healthy." Accuracy was still included in the metric to ensure the model retained strong overall predictive ability and did not over-focus on rare classes at the expense of majority-class performance. It did not make sense to solely use validation accuracy because classes were weighed differently. Accuracy treats all samples equally and would therefore undervalue performance on rare classes.

Under the combined score metric, the model checkpoint with the highest score, whether reached before or at epoch 100, was saved. Since resources were limited, the CNN hyperparameters tested were batch sizes of 16 and 32 and learning rates of 0.001 and 0.0001. The validation set was very small, with only 87 proliferate and 59 severe images. This could lead to a lot of variation in validation results from epoch to epoch. This further justifies the need for a longer early-stopping window. Although K-fold cross-validation was considered to reduce variance and provide more robust evaluation, but its computational cost exceeded the available resources of the project.

The feature embeddings for each image were extracted by passing the input through the CNN and taking the 512×1 feature vector produced by the final adaptive pooling layer. These embeddings were then used as input to two tree-based models: a Random Forest classifier and an XGBoost classifier. The Random Forest employed balanced class weighting using the same formula applied in the CNN, and its hyperparameters were tuned by selecting the configuration that achieved the highest validation "score."

For the XGBoost model, hyperparameters were also tuned to maximize the validation "score," with multi-class log loss used as the evaluation metric. Unlike the CNN, XGBoost allows the loss function to be directly modified because prediction is optimized through gradient and hessian calculations rather than full backpropagation. This makes it significantly easier to incorporate ordinal information into training. To leverage the ordinal nature of the diabetic retinopathy labels, a custom loss function was implemented that penalizes the model more heavily for predicting classes farther from the true severity. This was accomplished by combining the softmax output probabilities with a class-distance matrix. The intention behind this ordinal-aware loss was to encourage higher top-2 and top-3 accuracy, so that even when the model's top prediction is incorrect, the correct label is still ranked among the next most likely severities.

IV. Discussion

TABLE III
AGGREGATED "SCORE" FOR DIFFERENT LEARNING RATE AND BATCH SIZE COMBINATIONS

Batch Size	LR = 0.001	LR = 0.0001
16	0.68112	0.70399
32	0.750493	0.72468

TABLE IV
MACRO AVERAGE F1 SCORE FOR DIFFERENT LEARNING RATE AND BATCH SIZE COMBINATIONS

Batch Size	LR = 0.001	LR = 0.0001
16	0.651473	0.673494
32	0.726232	0.698487

TABLE V
VALIDATION ACCURACY FOR DIFFERENT LEARNING RATE AND BATCH SIZE COMBINATIONS

Batch Size	LR = 0.001	LR = 0.0001
16	75.0296%	77.5148%
32	80.7101%	78.5799%

Across all hyperparameters, the validation accuracy was higher than the macro average F1 score. This outcome is expected given the class imbalance in the dataset. The model naturally performs better on majority classes like healthy and moderate. Classifying these more common classes would then dominate the accuracy score, giving the model the appearance that it is performing better than it actually is.

With respect to hyperparameters, performance depended on the interaction between learning rate and batch size. For a batch size of 16, reducing the learning rate from 0.001 to 0.0001 improved both the F1 score and the aggregated "score," likely due to smoother and more stable gradient updates. However, with a batch size of 32, the opposite trend was observed. The higher learning rate of 0.001 produced the strongest performance across all metrics. This suggests that the larger batch size benefited from more aggressive updates, whereas the smaller batch size required a more conservative learning rate to avoid instability. Overall, the combination of a batch size of 32 and a learning rate of 0.001 yielded the best results among the configurations tested.

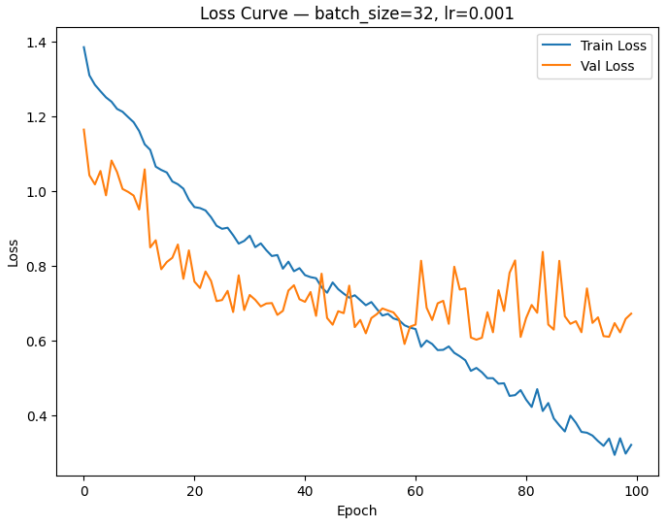

Fig. 1. The loss curve generated after training convolutional neural network with a batch size of 32 and learning rate of 0.001.

The loss curve for the model trained with a batch size of 32 and learning rate of 0.001 (shown in Fig. 1) starts off with a decreasing training and validation loss. Around 60 epochs however, the validation loss starts to show little to no improvement as the training loss continues to decrease. This conveys that the model is starting to overfit past this point, and for most hyperparameters, the best weights were restored to a value within 60-80 epochs. The validation loss curve also has a lot of noise. This is to be expected with the small validation set and with the weighting of the classes. Because individual severe or proliferative samples carry more weight, misclassifying just a few can cause noticeable fluctuations in the loss. Importantly, accuracy and macro F1 continues to improve even when the loss appears unstable, since those metrics are less sensitive to per-sample weighting. For the first 50 epochs, validation loss tended to be lower than training loss, which is a good sign because it means that the data augmentation was helping the model generalize.

The test set consists of only 569 images, with particularly limited representation for the minority classes (60 proliferate and 40 severe). The model trained with a learning rate of 0.001 was selected for final testing.

TABLE VI
CNN TEST ACCURACY RESULTS

Metric	Accuracy
Top-1 Accuracy	81.02%
Top-2 Accuracy	95.25%
Top-3 Accuracy	99.12%

From an accuracy perspective, the model performed very strongly given the lack of samples and the shallow CNN infrastructure. It predicted the correct severity 81% of the time and predicted had a 95% top-2 accuracy. This indicates that even when the model's first choice was incorrect, it almost always included the correct severity among its top two predictions. Considering that the task involves five ordinal classes, such performance suggests that the model has learned a meaningful representation of the progression of diabetic retinopathy. The top-3 accuracy of 99.12% further reinforces this observation, showing that the true label was nearly always ranked among the model's most likely predictions.

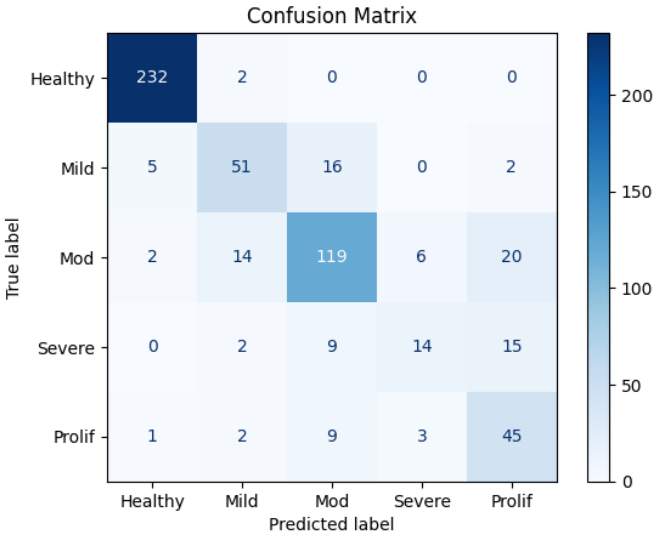

Fig. 2. Confusion matrix showing the predicted labels of the convolutional neural network compared with the true labels on the test set.

TABLE VII
CLASSIFICATION METRICS FOR THE CNN ON THE TEST SET

Class / Metric	Precision	Recall	F1-score	Support
Healthy	0.97	0.99	0.98	234
Mild	0.72	0.69	0.70	74
Moderate	0.78	0.74	0.76	161
Severe	0.61	0.35	0.44	40
Proliferative	0.55	0.75	0.63	60
Accuracy			0.81	569
Macro Avg	0.72	0.70	0.70	569
Weighted Avg	0.81	0.81	0.81	569

The model performs extremely well in classifying healthy images, as shown in Table VII, achieving a near perfect F1 score of 0.98. If the task was simplified to a binary decision where the goal is to classify retinal images as having diabetic

retinopathy or not, the model performance would be very high. Among the diabetic retinopathy classes, the predictive power drops in accordance with how often samples appear in the dataset. The model is able to predict moderate cases, which appears second most often, the second best. The same follows with mild, proliferative and severe. The drop in predicative power is not directly tied to the size of the class, however. There are less than half the amount of mild images as there are moderate images, yet only a 0.06 drop in F-1 score. The model classifies severe at a much weaker rate than all the other classes. The prediction for severe is on par with proliferate, but the recall is much worse. It is a bit confusing why this is the case, as severe and proliferate are weighed more. This is surprising because severe and proliferative were weighted more heavily during training. Such weighting usually boosts recall for underrepresented classes, which is reflected in the proliferate class but not severe. Overall, the model is highly effective at identifying healthy cases and reasonably good at distinguishing lower to moderate levels of DR. However, its performance declines for advanced stages, which is concerning from a clinical perspective, as severe cases are the most critical to detect accurately. For practical deployment, the decision threshold for Severe and Proliferate could be lowered to improve recall. In medical settings, it is often preferable to allow more false positives if it helps capture a larger proportion of dangerous, high-risk cases. However, the model frequently confuses Severe and Proliferative predictions with each other or with Moderate cases, as shown by the confusion matrix in Fig. 2. Adjusting thresholds to increase recall may not actually improve overall predictions as it can exacerbate misclassification between these similar classes.

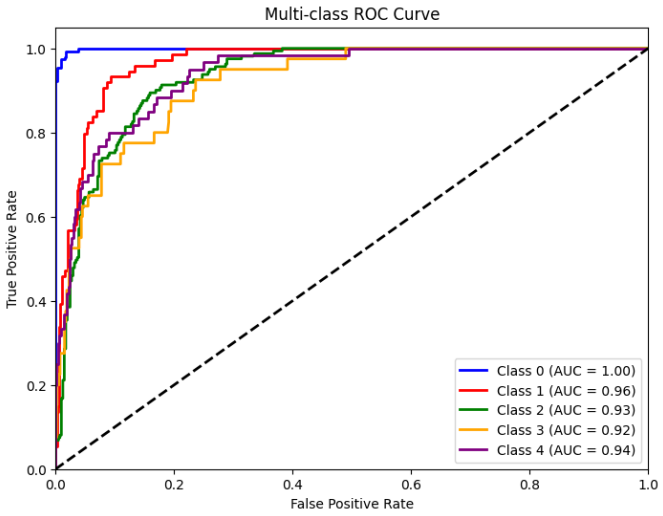

Fig. 3. One-vs-rest ROC-AUC curves for each class in the CNN, showing the model's ability to distinguish between Healthy, Mild, Moderate, Severe, and Proliferative retinal images.

One-vs-rest ROC-AUC scores were very high across all five classes (shown in Fig. 3). In a multiclass problem like this, ROC-AUC is computed by treating each class as "positive" and the remaining classes as "negative," so a high score means the model is very good at ranking true examples of that class above non-examples. It is like asking how well the model can predict healthy images versus images with diabetic retinopathy. This can seem confusing because the F1 scores for the severe and proliferative categories were low. The key difference is that ROC-AUC evaluates ranking, whereas the final CNN prediction is made using argmax, which picks only the single highest probability class. For rare classes, the model often assigns a moderately high probability (like 0.5-0.6), but another class, like moderate, gets an even higher probability. As a result, the model has a high AUC but low F1. It ranks the correct class well but doesn't choose it as the top-1 label. This also explains why top-2 and top-3 accuracy are extremely high, while top-1 accuracy is lower. The correct label is often the second or third highest predicted class probability.

After identifying and evaluating the best CNN, its learned feature embeddings were used as inputs to a Random Forest classifier. The Random Forest was tuned using a range of hyperparameters, and each configuration was evaluated using a composite score ($0.7 \times$ F1-score + $0.3 \times$ validation accuracy). Class weights were kept consistent with those used in the CNN to reflect the imbalance between majority and minority diabetic retinopathy classes. The optimal hyperparameters were found to be 400 estimators, a maximum depth of 40, square-root feature selection, a minimum of 2 samples required to split a node, and a minimum of 2 samples per leaf. The Random Forest was then trained and tested using these settings.

TABLE VIII
RANDOM FOREST TEST ACCURACY RESULTS

Metric	Accuracy
Top-1 Accuracy	81.90%
Top-2 Accuracy	95.08%
Top-3 Accuracy	99.12%

TABLE IX
CLASSIFICATION METRICS FOR THE RANDOM FOREST ON THE TEST SET

Class / Metric	Precision	Recall	F1-score	Support
0	0.97	0.99	0.98	234
1	0.85	0.62	0.72	74
2	0.70	0.87	0.78	161
3	0.60	0.30	0.40	40
4	0.64	0.60	0.62	60
Accuracy			0.82	569
Macro Avg	0.75	0.68	0.70	569
Weighted Avg	0.82	0.82	0.81	569

The Random Forest model achieved performance similar to the CNN, with nearly identical accuracy and macro average F1 Score (shown in Table. IX). Given the small size of the test set, especially for the minority classes, the differences in F1-scores between the two models are not large enough to justify selecting the Random Forest over the CNN-based classifier. Following this, XGBoost was trained on the same CNN embeddings. Hyperparameters were tuned on the validation

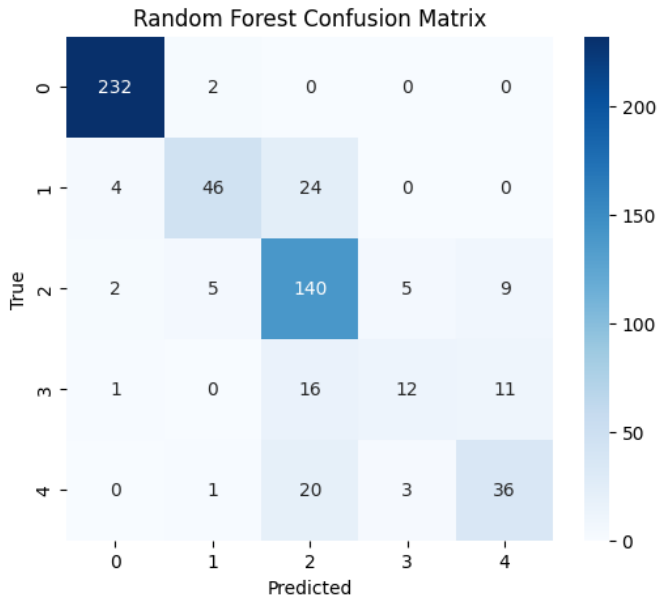

Fig. 4. Confusion matrix showing the predicted labels of the random forest model compared with the true labels on the test set.

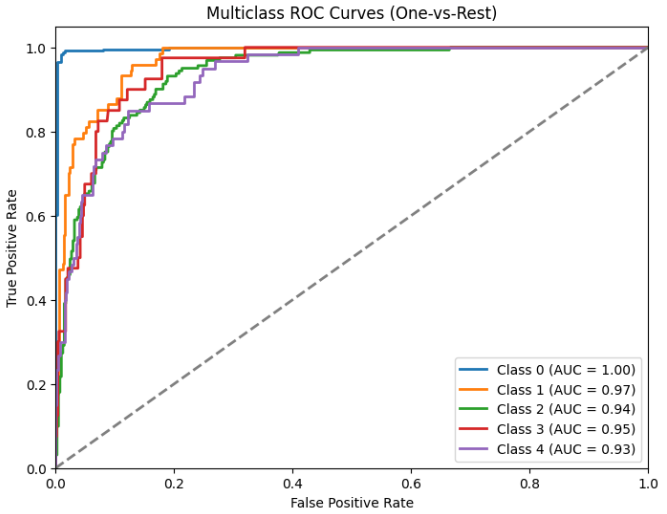

Fig. 5. One-vs-rest ROC-AUC curves for each class in the Random Forest, showing the model's ability to distinguish between Healthy, Mild, Moderate, Severe, and Proliferative retinal images.

set, and the optimal configuration consisted of a learning rate of 0.1, a maximum tree depth of 3, a subsample ratio of 0.7, a column (feature) subsample ratio of 0.7, and multiclass logarithmic loss as the evaluation metric. As with the CNN and Random Forest, class weighting was applied to compensate for the class imbalance.

The performance of XGBoost using standard multiclass logarithmic loss closely resembled the CNN and Random Forest models (shown in Table XI). Given the limited test set, particularly for the minority classes, none of the differences in F1-score or recall were substantial enough to suggest that

TABLE X
XGBOOST TEST ACCURACY RESULTS

Metric	Accuracy
Top-1 Accuracy	81.02%
Top-2 Accuracy	94.73%
Top-3 Accuracy	99.47%

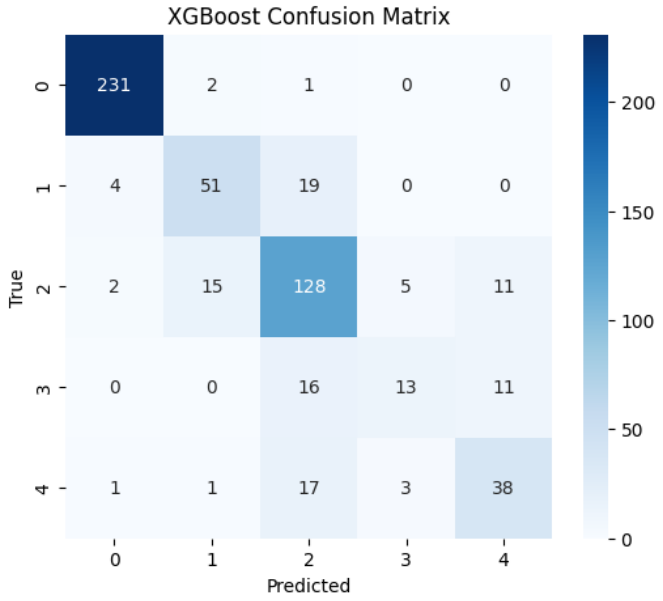

Fig. 6. Confusion matrix showing the predicted labels of the XGBoost model compared with the true labels on the test set.

TABLE XI
CLASSIFICATION METRICS FOR THE XGBOOST ON THE TEST SET

Class / Metric	Precision	Recall	F1-score	Support
0	0.97	0.99	0.98	234
1	0.74	0.69	0.71	74
2	0.71	0.80	0.75	161
3	0.62	0.33	0.43	40
4	0.63	0.63	0.63	60
Accuracy			0.81	569
Macro Avg	0.73	0.69	0.70	569
Weighted Avg	0.81	0.81	0.80	569

XGBoost meaningfully outperformed or underperformed the other approaches on any specific severity class.

To further leverage the ordinal structure of the diabetic retinopathy labels, another version of the XGBoost model was tested with an "ordinal aware" loss function. Multiclass logarithmic loss was used but it was multiplied by a penalty factor that was dependent on how far the prediction was from the correct severity. The penalty scales quadratically (1, 4, 9, 16, 25), meaning misclassifications farther from the correct class were punished more heavily than close-range errors. The goal of this approach was not necessarily to improve top-1 accuracy, but to encourage the model to make "nearby" predictions, thereby improving top-2 and top-3 accuracy. The model was tuned with the validation dataset to find the ideal hyperparameters: learning rate of 0.1, maximum depth of

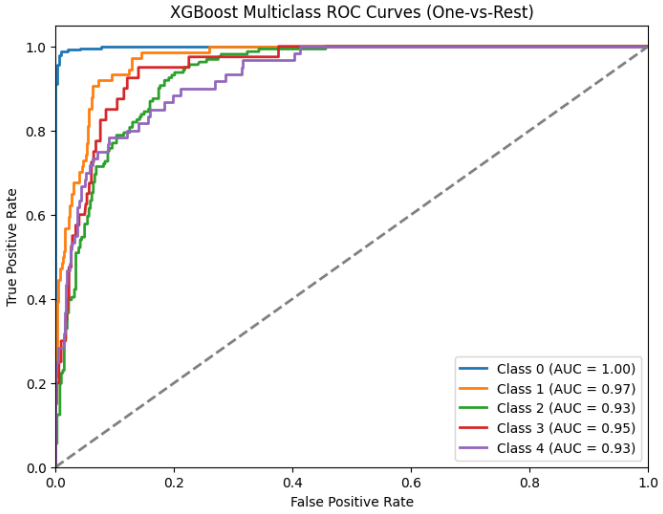

Fig. 7. One-vs-rest ROC-AUC curves for each class in XGBoost, showing the model's ability to distinguish between Healthy, Mild, Moderate, Severe, and Proliferative retinal images.

3, subsample rate of 0.7, column sampling rate of 0.7, and multiclass logarithmic loss for evaluation.

TABLE XII
XGBOOST (ORDINAL DISTANCE PENALTY) TOP-K ACCURACY

Metric	Accuracy
Top-1 Accuracy	81.55%
Top-2 Accuracy	95.08%
Top-3 Accuracy	98.95%

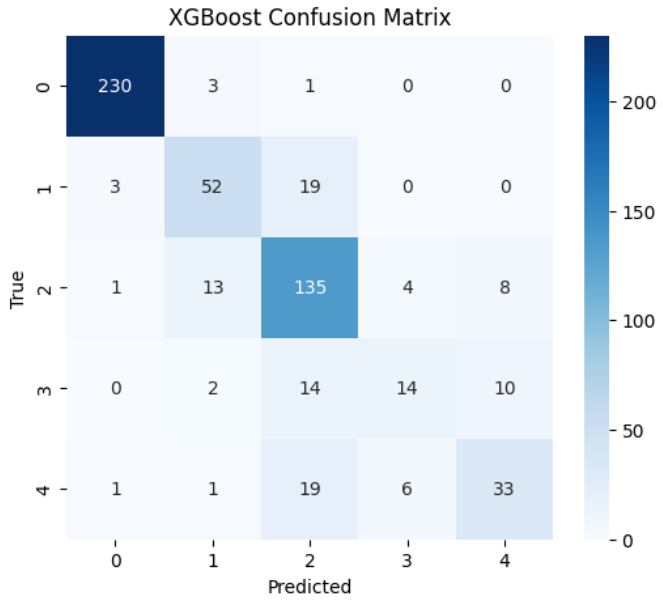

Fig. 8. Confusion matrix showing the predicted labels of the XGBoost model with an ordinal distance penalty compared with the true labels on the test set.

Surprisingly, results of the ordinal-aware XGBoost model were nearly identical to all the other models (shown in Table. XIII. Top-2 or Top-3 accuracy did not improve significantly either (Table. XII). Like all the models, there was a tendency to predict moderate even for severe or proliferate cases. Overall, however, all models generally produced predictions that fell within a reasonable range of the true severity, particularly for Healthy, Mild, and Moderate images.

These results indicate that feeding CNN embeddings into a Random Forest or XGBoost model did not improve prediction performance. There are several likely explanations for this. First, the embeddings themselves may not capture all critical features needed to accurately distinguish severity. With a top-1 accuracy of approximately 80%, the CNN may still miss key indicators of diabetic retinopathy severity. Since the Random Forest and XGBoost models are trained on these same embeddings, they are limited by the quality of the input features. Another contributing factor is the high dimensionality of the embeddings: the CNN produces 512 features, but the training set contains only around 4,000 images. This large feature-to-sample ratio can increase the risk of overfitting or produce instability in tree-based models, limiting their ability to improve upon the CNN's predictions.

TABLE XIII
CLASSIFICATION METRICS FOR THE ORDINAL-AWARE XGBOOST ON THE TEST SET

Class / Metric	Precision	Recall	F1-score	Support
0	0.98	0.98	0.98	234
1	0.73	0.70	0.72	74
2	0.72	0.84	0.77	161
3	0.58	0.35	0.44	40
4	0.65	0.55	0.59	60
Accuracy			0.82	569
Macro Avg	0.73	0.68	0.70	569
Weighted Avg	0.81	0.82	0.81	569

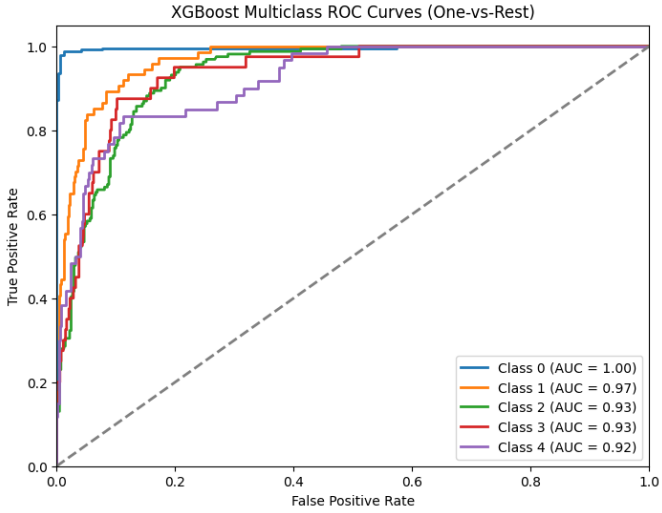

Fig. 9. One-vs-rest ROC-AUC curves for each class in XGBoost (with ordinal distance penalty), showing the model's ability to distinguish between Healthy, Mild, Moderate, Severe, and Proliferative retinal images.

V. Conclusion

For image classification, CNNs alone provided solid predictive power that was not improved upon with either ensemble method. Although the labels were ordinal and the CNN was not explicitly designed to leverage this relationship, it was still able to produce predictions that were generally within a reasonable range of the true severity for each class. The model's performance exceeded the initial expectations established at the start of the project. With additional computational resources and memory, the CNN could be trained deeper to potentially achieve even better results. However, increasing model depth would likely require a larger dataset to prevent overfitting.

Despite its strengths, the current CNN is not yet suitable for deployment in clinical practice. Its performance is most reliable for distinguishing Healthy images from those affected by diabetic retinopathy, but this binary classification alone is of limited utility to medical professionals. The ultimate goal is to accurately map severity and identify Severe cases of non-proliferative and proliferative diabetic retinopathy, which the current model does not achieve consistently. Another limitation of CNNs is their lack of interpretability; it is difficult to determine which features or blood vessel patterns the network relies on to assess severity. Incorporating interpretability methods would be beneficial to understand how the CNN evaluates retinal images and which patterns correspond to different levels of diabetic retinopathy. A technique that has been experimented was using image segmentation to identify blood vessels in retinal images, which would be a next step in exploring this project [5].

References

[1] Mayo Clinic Staff, "Diabetic retinopathy—Symptoms and causes," *Mayo Clinic*. [Online]. Available: https://www.mayoclinic.org/diseases-conditions/diabetic-retinopathy/symptoms-causes/syc-20371611. [Accessed: Dec. 13, 2025].

[2] O. Essa, "Diabetic retinopathy dataset," Kaggle. [Online]. Available: https://www.kaggle.com/datasets/jockeroika/diabetic-retinopathy/data. [Accessed: Dec. 13, 2025].

[3] S. R. Rath, "Diabetic retinopathy 224x224 (2019) dataset," Kaggle. [Online]. Available: https://www.kaggle.com/datasets/sovitrath/diabetic-retinopathy-224x224-2019-data. [Accessed: Dec. 13, 2025].

[4] S. de Lusignan, J. Belsey, N. Hague, and B. Dzregah, "End-digit preference in blood pressure recordings of patients with ischaemic heart disease in primary care," *Journal of Human Hypertension*, vol. 18, no. 4, pp. 261–265, 2004, doi: 10.1038/sj.jhh.1001663.

[5] S. Deari, İ. Öksüz, and S. Ulukaya, "Importance of data augmentation and transfer learning on retinal vessel segmentation," in *2021 29th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 2021, pp. 1-4. [Online]. Available: https://doi.org/10.1109/TELFOR52709.2021.9653400. [Accessed: Dec. 13, 2025].