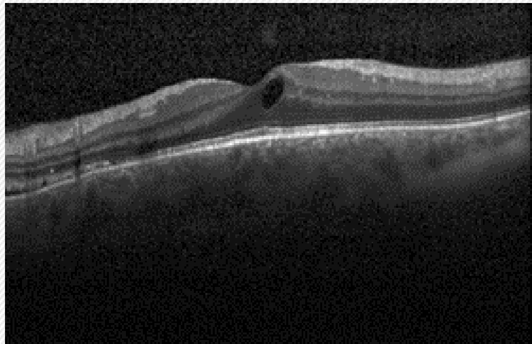
Using Artificial Intelligence for Analyzing the Retinal Images (OCT) in People with Diabetes

Tahani A. Daghistani

Introduction

- Diabetic macular edema (DME) is a common disease of diabetic retinopathy.
- Due to the infection of DME disease, many patients’ vision is lost.
- To cure DME eye disease, early detection and treatment are very important and vital steps.
- Medical imaging developed rapidly to play a central role in medicine today by supporting diagnosis and treatment of a disease.
- To automatically diagnosis DEM disease, strategies involving Artificial Intelligence (AI) could provide a solution.



Goal

- The main goal of this project is develop an Artificial Intelligence solution that can help to classify OCT.
- To achieve our goal, Deep-Learning (DL) model will be used to predict the risk of patients with diabetes developing diabetic retinopathy.
- The outcomes would improve patient access to treatment and ease pressures on time and resources in ophthalmology clinics.

Data

- The dataset downloaded from the Kaggle website.
- A total of (3,500) OCT images included in this dataset.
- The ratio of splitting the original dataset (3,500 images) is 50/20/30.
    - Training data made up of 50% training (1,960 images)
    - Devote 20% to the validation set (490 images) to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters
    - Test dataset made up of remaining 30% (1,050 images) to evaluate the accuracy of the model on the data it has never seen.

Feature Engineering

- Resize image to establish a base size for all images fed into our machine learning algorithms.
- Classic augmentation techniques like flips and rotations will applied to each image in the training set without manually processing each image.
  - Shifting, Flipping, Rotating

Models

- Classification is the most widely used technique for machine learning.
- CNN is a powerful neural network that uses filters to extract features from images.
- All popular frameworks support Convolutional Neural Networks like Tensorflow-Keras.
- Five models with different Convolutional layers built then the best one selected based on evaluation metrics.

Models

- Each input image will pass it through a series of convolution layers with filters (Kernals), Pooling, fully connected layers and apply activation functions to classify an object with probabilistic values between 0 and 1.

- Pool feature map applied then flatten our pooled feature map into one-dimensional to be ready for Dense layers.

- An epoch is how many times the model trains on our whole data set.
- Batch can be explained as taking in small amounts, train and take some more.
  - Each epoch must finish all batch before moving to the next epoch.

- Early Stopping used to reduce the computer calculation, program running time and avoid overfitting.
  - The model at the time that training is stopped to have good generalization performance.

- Model Consists of input layers, two hidden layers, and one output layer

Model Evaluation and Selection

- Confusion Matrix
  - True Positive =True Positive/(True Positive+False Negative)
  - False Positive (FPR)=False Positive/(False Positive+True Negative)
  - Precision=True Positive/(True Positive+False Positive)
  - Recall=True Positive/(True Positive+False Negative)
  - F-measure: F_score =2*(precision*recall)/(precision+recall)

- Area Under the ROC curve (AUROC)

- Loss to evaluate and diagnose model optimization.

Tools

- The Jupyter Notebook
- Python programming language
- Libraries: A library is a collection of pre-combined codes that can be used iteratively to reduce the time required to code.
  - Tensorflow-Keras for developing deep learning models.
  - Numpy and pandas for data manipulation
  - Matplotlib and seaborn for visualization

Results

Table1: The Performance of the Models on the Test Dataset

CNN Model/Metrics	Accuracy	precision	recall	f1-score	AUC
5-Convolutional Layer	[illegible]	[illegible]	[illegible]	[illegible]	0.82
4-Convolutional Layer	[illegible]	[illegible]	[illegible]	[illegible]	0.79
3-Convolutional Layer	[illegible]	[illegible]	[illegible]	[illegible]	0.75
2-Convolutional Layer	0.74	0.74	0.74	0.74	0.74
1-Convolutional Layer	0.63	0.70	0.63	0.60	0.63

Table2: precision and recall of the Models on the Test Dataset

CNN Model/Metrics	precision	recall
DME	0.87	0.74
Normal	0.77	0.89

Results

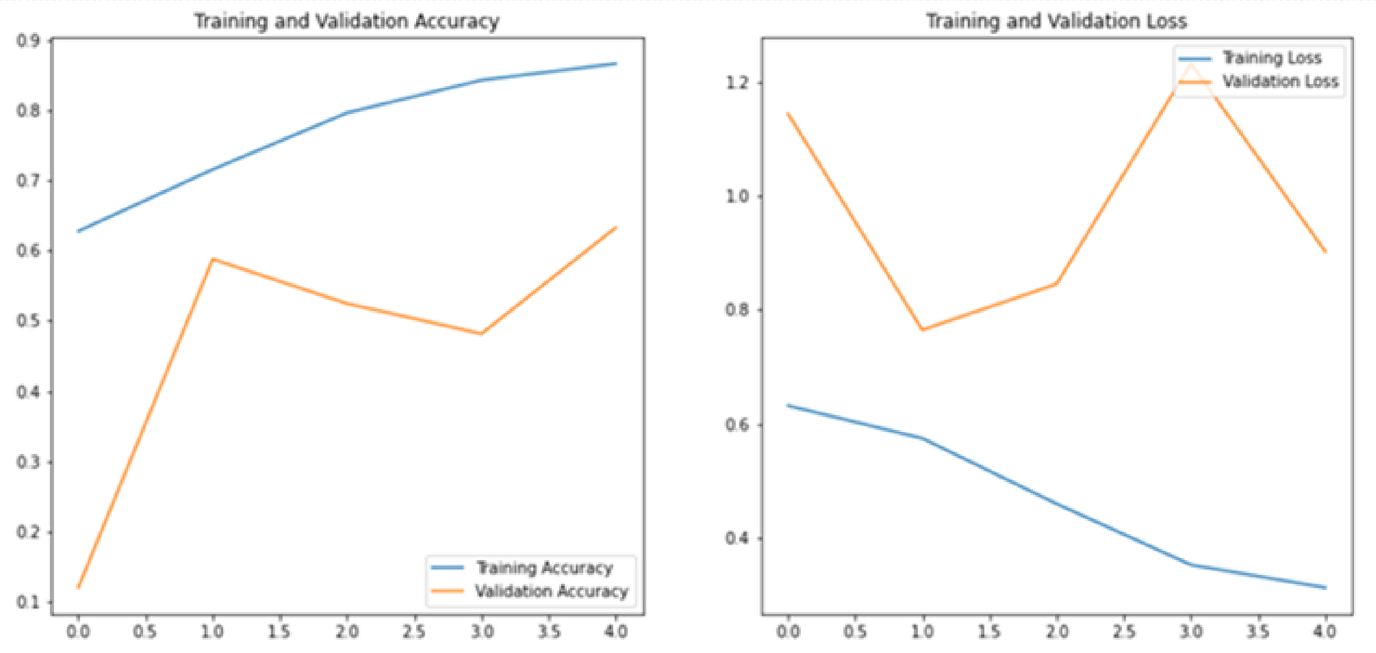
Model: "sequential_6"

Layer (type)	Output Shape	Param #
conv2d_16 (Conv2D)	(None, 222, 222, 32)	896
max_pooling2d_16 (MaxPooling	(None, 111, 111, 32)	0
conv2d_17 (Conv2D)	(None, 109, 109, 32)	9248
max_pooling2d_17 (MaxPooling	(None, 55, 55, 32)	0
conv2d_18 (Conv2D)	(None, 53, 53, 32)	9248
max_pooling2d_18 (MaxPooling	(None, 27, 27, 32)	0
conv2d_19 (Conv2D)	(None, 25, 25, 32)	9248
max_pooling2d_19 (MaxPooling	(None, 13, 13, 32)	0
conv2d_20 (Conv2D)	(None, 11, 11, 32)	9248
max_pooling2d_20 (MaxPooling	(None, 6, 6, 32)	0
flatten_6 (Flatten)	(None, 1152)	0
dense_16 (Dense)	(None, 64)	73792
dense_17 (Dense)	(None, 32)	2080
dense_18 (Dense)	(None, 1)	33

Total params: 113,793
Trainable params: 113,793
Non-trainable params: 0

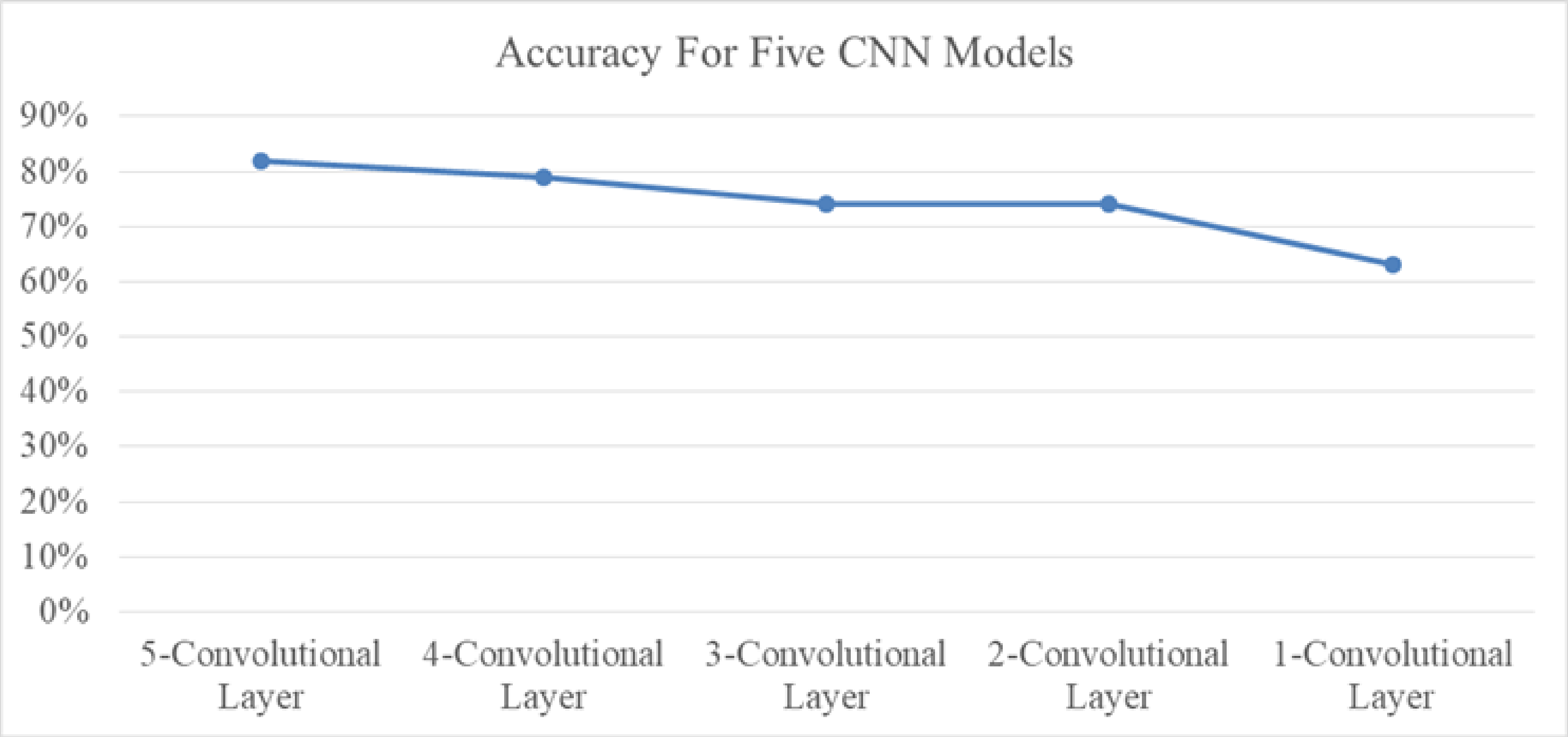
Summary of 5-Convolutional Layer (CNN) Model

Results



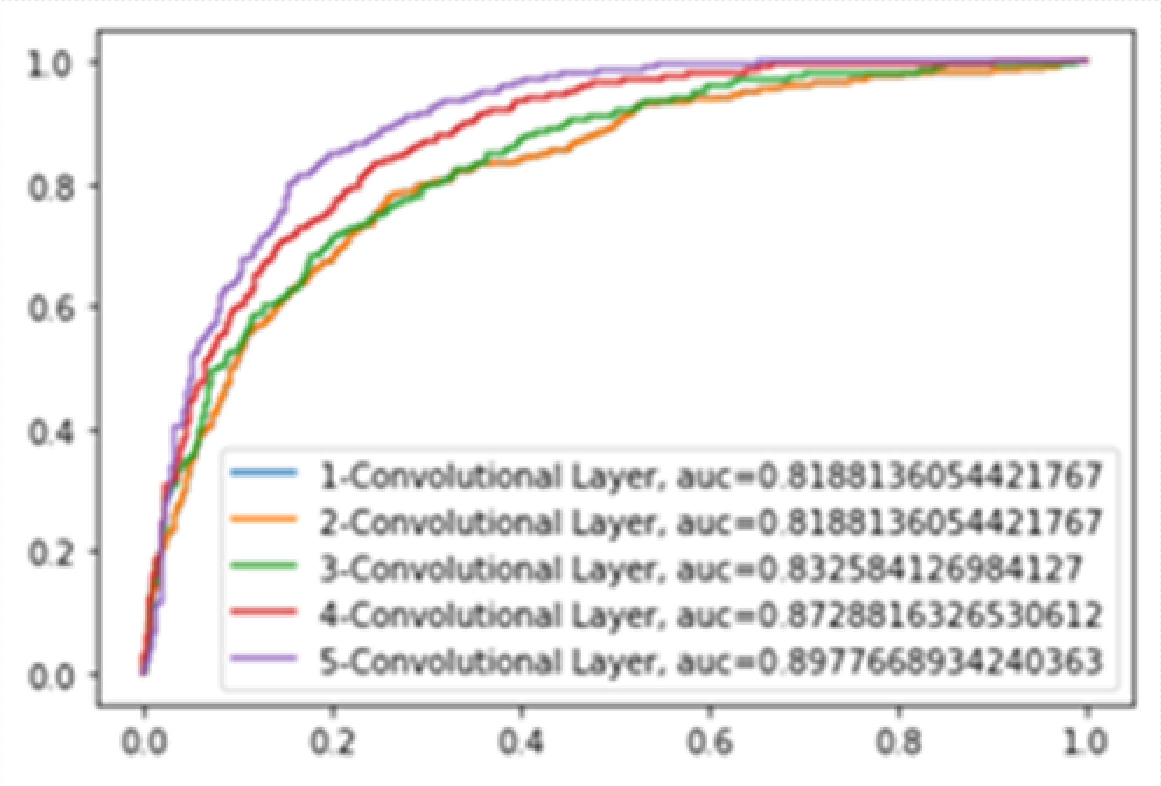
Accuracy and loss for training and validation sets

Results



Accuracy achieved for five CNN models

Results



AUC achieved for five CNN models

Conclusion

- We propose a deep learning based method to solve the and classification problem in OCT images.
- Results show that the proposed method has been successful in predicting the presence of DME in OCT images.

Thank you