Aditya Silver Oak Institute of Technology
Department of Computer Engineering/Information Technology

Title of the event: 2 Days Workshop on "Deep Learning"			
Coordinator: Prof. Sagar Patel			
Mode of Webinar:	Zoom	**Date:**	5/7/2021 - 6/7/2021
Semester:	5th CE/IT		

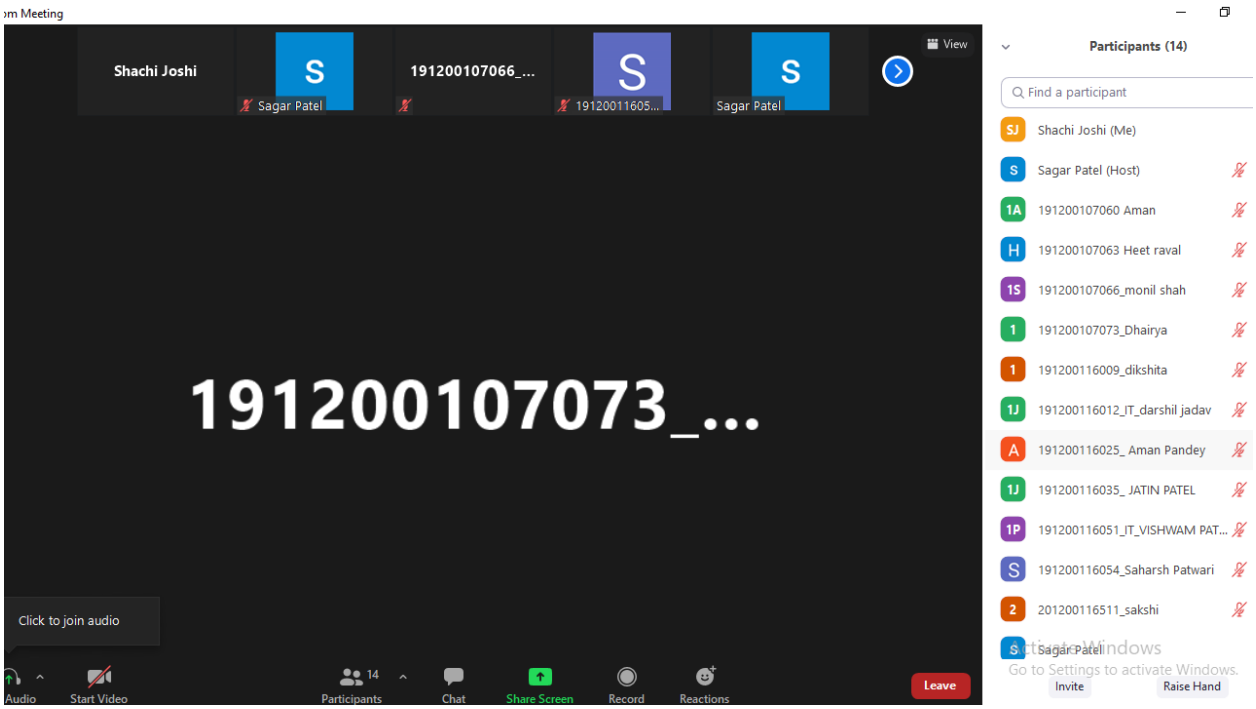
Abstract:
Machine Learning has nowadays become one of the most important parts of the technology. Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.The ability to process large numbers of features makes deep learning very powerful when dealing with unstructured data. So for students it is necessary to understand the importance of deep learning and through this workshop they were able to experience real time working and understanding in the field of Deep Learning.

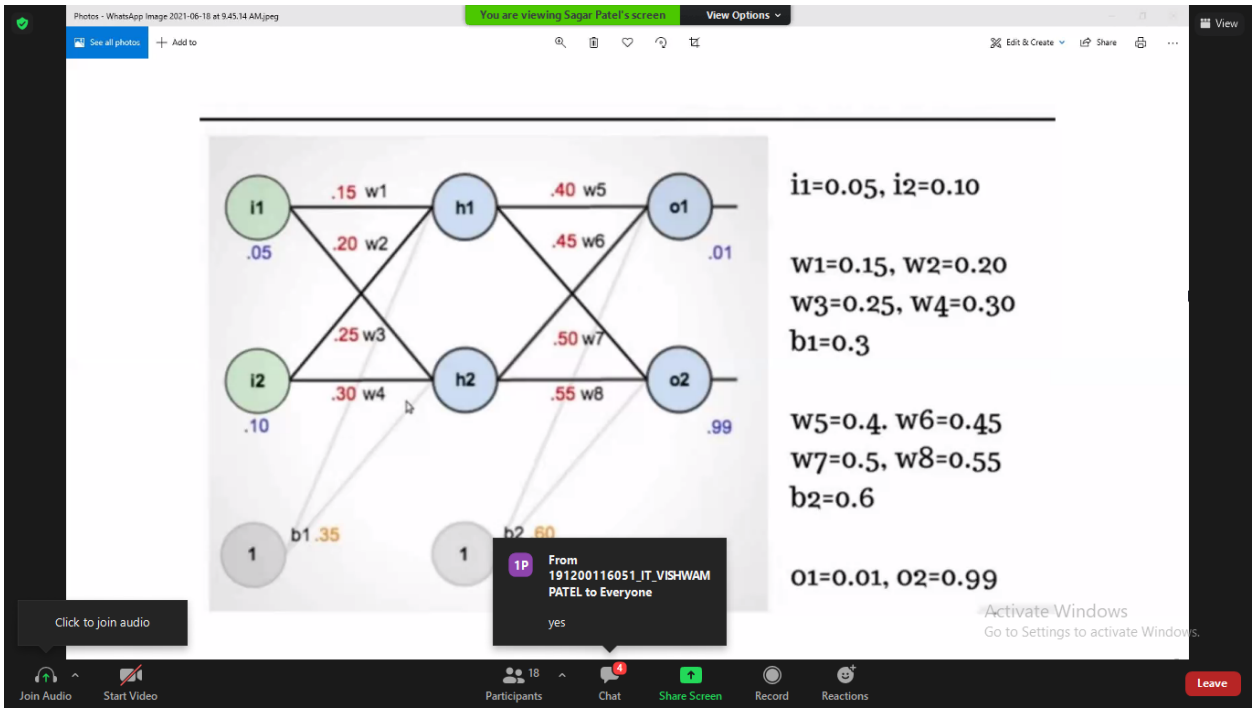
Expert or Guest Profile (In Detail):
Mr. Sagar Patel, Aditya Silver Oak Institute of Technology
More than 11 years of experience
Specialization: Deep Learning, Machine Learning, IOT

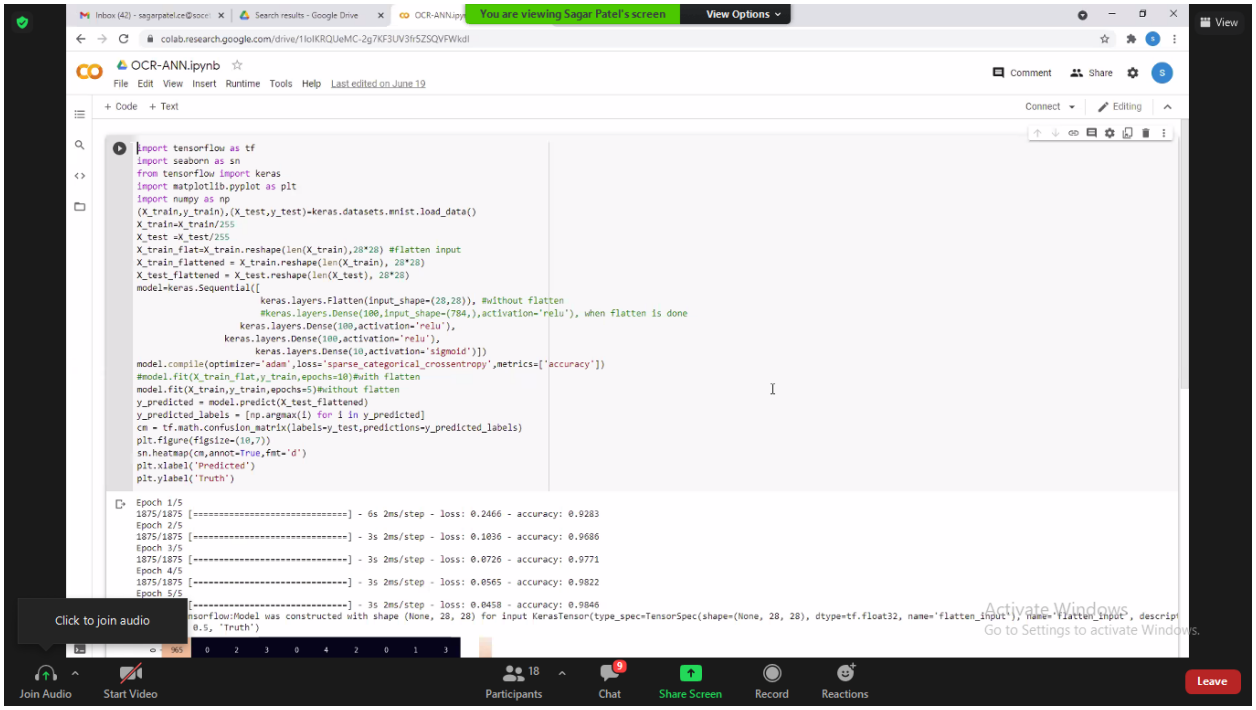
Topics delivered:
- What is Machine Learning and Deep Learning?
- Basics of Neural Networks
- What is Artificial Neural Networks ?
- Convolutional Neural Networks
- Difference between Neural networks vs Deep Learning.
- Optical Character Recognition
- How to train MNN Training
- Back propagation
- Vanishing and Exploding Gradient Problem

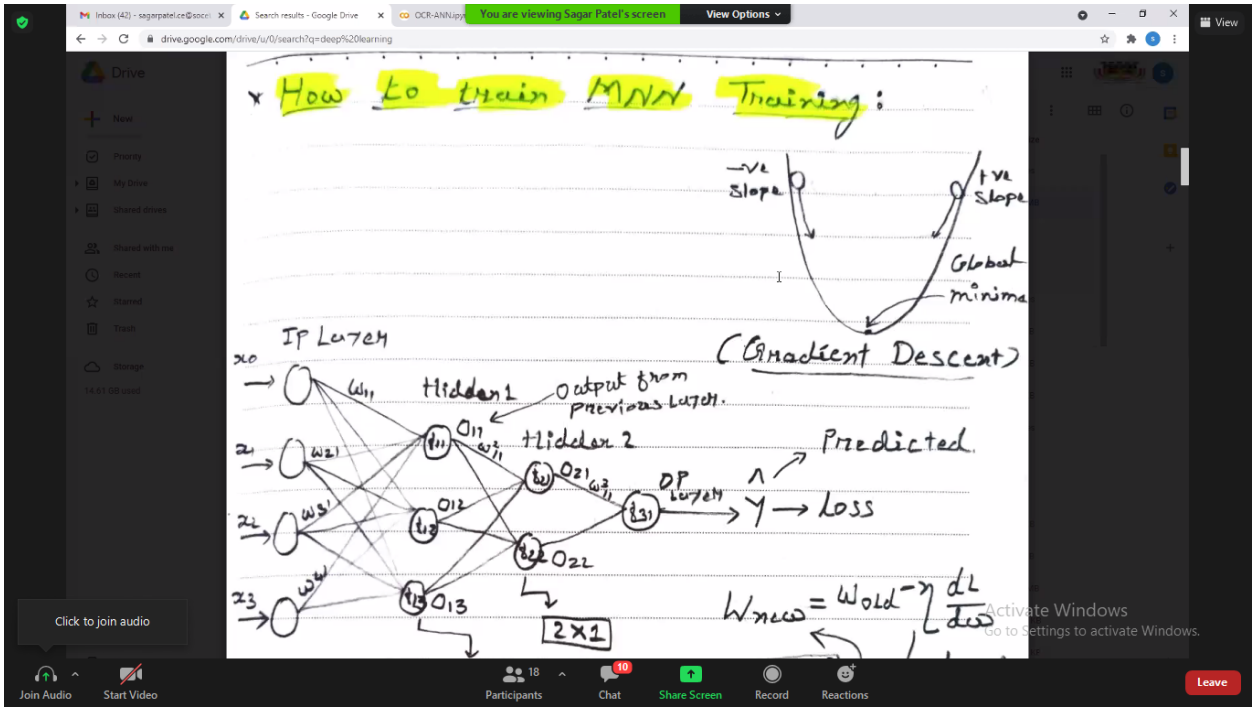
No. of Participants: 25
Impact of that event in terms of POs & PSOs:
- Importance of Deep Learning in upcoming era of technology
- Knowledge for Neural Networks
- Artificial Neural Networks
- Scope of Deep Learning
- Implementation of OCR-ANN
- Understanding of MNN Training

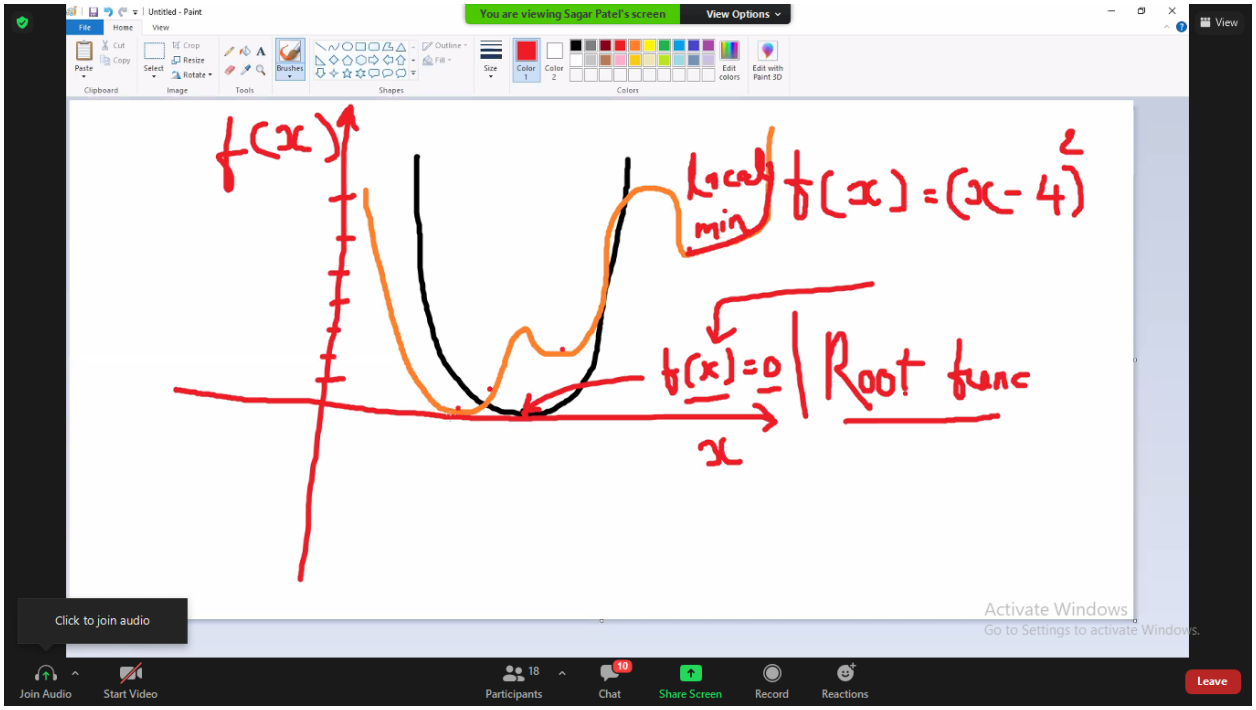
Photographs:

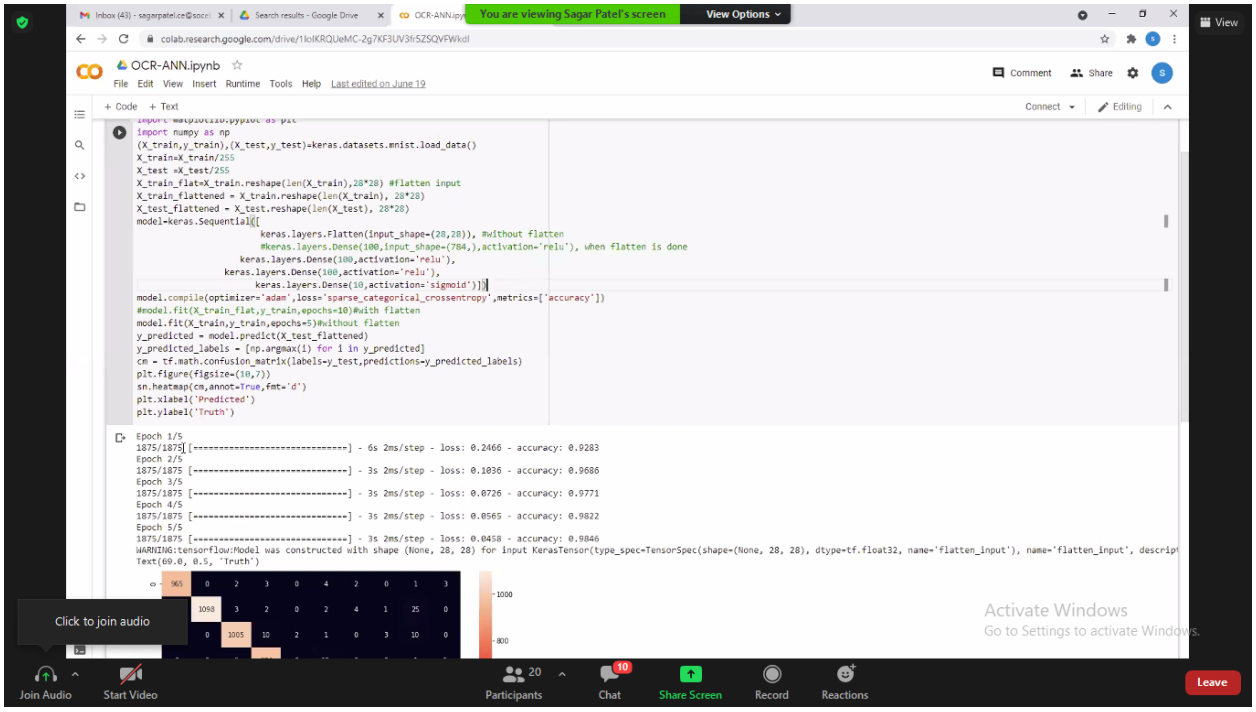





You are viewing Sagar Patel's screen
View Options
OCR-ANN.ipynb
File Edit View Insert Runtime Tools Help Last edited on June 19
Comment
Share
+ Code + Text
Connect
Editing
import tensorflow as tf
import seaborn as sn
from tensorflow import keras
import matplotlib.pyplot as plt
import numpy as np
(X_train,y_train),(X_test,y_test)=keras.datasets.mnist.load_data()
X_train=X_train/255
X_test =X_test/255
X_train_flat=X_train.reshape(len(X_train),28*28) #flatten input
X_train_flattened = X_train.reshape(len(X_train), 28*28)
X_test_flattened = X_test.reshape(len(X_test), 28*28)
model=keras.Sequential([
keras.layers.Flatten(input_shape=(28,28)), #without flatten
#keras.layers.Dense(100,input_shape=(784,),activation='relu'), when flatten is done
keras.layers.Dense(100,activation='relu'),
keras.layers.Dense(100,activation='relu'),
keras.layers.Dense(10,activation='sigmoid')])
model.compile(optimizer='adam',loss='sparse_categorical_crossentropy',metrics=['accuracy'])
#model.fit(X_train_flat,y_train,epochs=10)#with flatten
model.fit(X_train,y_train,epochs=5)#without flatten
y_predicted = model.predict(X_test_flattened)
y_predicted_labels = [np.argmax(i) for i in y_predicted]
cm = tf.math.confusion_matrix(labels=y_test,predictions=y_predicted_labels)
plt.figure(figsize=(10,7))
sn.heatmap(cm,annot=True,fmt='d')
plt.xlabel('Predicted')
plt.ylabel('Truth')
Epoch 1/5
1875/1875 [==========] - 6s 2ms/step - loss: 0.2466 - accuracy: 0.9283
Epoch 2/5
1875/1875 [==========] - 3s 2ms/step - loss: 0.1036 - accuracy: 0.9686
Epoch 3/5
1875/1875 [==========] - 3s 2ms/step - loss: 0.0726 - accuracy: 0.9771
Epoch 4/5
1875/1875 [==========] - 3s 2ms/step - loss: 0.0565 - accuracy: 0.9822
Epoch 5/5
1875/1875 [==========] - 3s 2ms/step - loss: 0.0458 - accuracy: 0.9846
WARNING:tensorflow:Model was constructed with shape (None, 28, 28) for input KerasTensor(type_spec=TensorSpec(shape=(None, 28, 28), dtype=tf.float32, name='flatten_input'), name='flatten_input', descrip
0.5, 'Truth')
Click to join audio
Activate Windows
Go to Settings to activate Windows.
Join Audio
Start Video
18
Participants
9
Chat
Share Screen
Record
Reactions
Leave


You are viewing Sagar Patel's screen
View Options
drive.google.com/drive/u/0/search?q=deep%20learning
How to train MNN Training :
-ve Slope
+ve Slope
Global minima
(Gradient Descent)
IP Layer
Hidden 1
output from previous Layer.
Hidden 2
Predicted
OP Layer
Y → Loss
2x1
Click to join audio
Activate Windows
Go to Settings to activate Windows.
Join Audio
Start Video
18
Participants
10
Chat
Share Screen
Record
Reactions
Leave


Untitled - Paint
You are viewing Sagar Patel's screen
View Options
f(x)
local min
f(x)=(x-4)²
f(x)=0
Root func
x
Click to join audio
Activate Windows
Go to Settings to activate Windows.
Join Audio
Start Video
18
Participants
10
Chat
Share Screen
Record
Reactions
Leave
View


You are viewing Sagar Patel's screen
View Options
OCR-ANN.ipynb
File Edit View Insert Runtime Tools Help Last edited on June 19
Comment
Share
Connect
Editing
+ Code + Text
import numpy as np
(X_train,y_train),(X_test,y_test)=keras.datasets.mnist.load_data()
X_train=X_train/255
X_test =X_test/255
X_train_flat=X_train.reshape(len(X_train),28*28) #flatten input
X_train_flattened = X_train.reshape(len(X_train), 28*28)
X_test_flattened = X_test.reshape(len(X_test), 28*28)
model=keras.Sequential([
keras.layers.Flatten(input_shape=(28,28)), #without flatten
#keras.layers.Dense(100,input_shape=(784,),activation='relu'), when flatten is done
keras.layers.Dense(100,activation='relu'),
keras.layers.Dense(100,activation='relu'),
keras.layers.Dense(10,activation='sigmoid')])
model.compile(optimizer='adam',loss='sparse_categorical_crossentropy',metrics=['accuracy'])
#model.fit(X_train_flat,y_train,epochs=10)#with flatten
model.fit(X_train,y_train,epochs=5)#without flatten
y_predicted = model.predict(X_test_flattened)
y_predicted_labels = [np.argmax(i) for i in y_predicted]
cm = tf.math.confusion_matrix(labels=y_test,predictions=y_predicted_labels)
plt.figure(figsize=(10,7))
sn.heatmap(cm,annot=True,fmt='d')
plt.xlabel('Predicted')
plt.ylabel('Truth')
Epoch 1/5
1875/1875 [==============================] - 6s 2ms/step - loss: 0.2466 - accuracy: 0.9283
Epoch 2/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.1036 - accuracy: 0.9686
Epoch 3/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0726 - accuracy: 0.9771
Epoch 4/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0565 - accuracy: 0.9822
Epoch 5/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0458 - accuracy: 0.9846
WARNING:tensorflow:Model was constructed with shape (None, 28, 28) for input KerasTensor(type_spec=TensorSpec(shape=(None, 28, 28), dtype=tf.float32, name='flatten_input'), name='flatten_input', descript
Text(69.0, 0.5, 'Truth')
Click to join audio
Activate Windows
Go to Settings to activate Windows.
Join Audio
Start Video
20
Participants
10
Chat
Share Screen
Record
Reactions
Leave
View

Attendance sheet with student name:
Available in Google Form Response Sheet

Student feedback:

"Genuinely one of the best workshops I've attended."

"I found it really interesting and was able to gain a lots of knowledge regarding Deep Learning"