

Handwritten English Letters Recognition

PYERS3

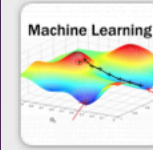
孟凡佛
孙阿敏
束玥璘

About our project

- ▶ What is wonder words?
 - ▶ a small program called Wonder Words. Wonder Words can identify printed or clear handwritten English letters.
- ▶ Why?
 - ▶ Handwriting recognition has a long history and is widely used in modern society.
 - ▶ By applying what we have learned in the class to get a deeper understanding about machine learning.

Preparation:

- ▶ Extensive reading on machine
- ▶ Viewed tons of tutorials and in videos
 - ▶ E.g. 3blue1brown's series on neu



A Quick Introduction to Neural Networks

ujjwalkarn.me

An Artificial Neural Network (ANN) is a computational mod...



OCR of Hand-written Data using kNN

opencv-python-tutroals.readthedocs.io

Goal In this chapter We will use our knowledge on kNN to...



keras/examples at master · keras-team/keras

github.com

GitHub is where people build software. More than 27 million...



A Guide to TF Layers: Building a Convolutional Ne...

tensorflow.org

The TensorFlow layers module provides a high-level API that...



Character Recognition in Natural Images

ee.surrey.ac.uk

[jump to download] Character recognition is a classic patter...

Challenges

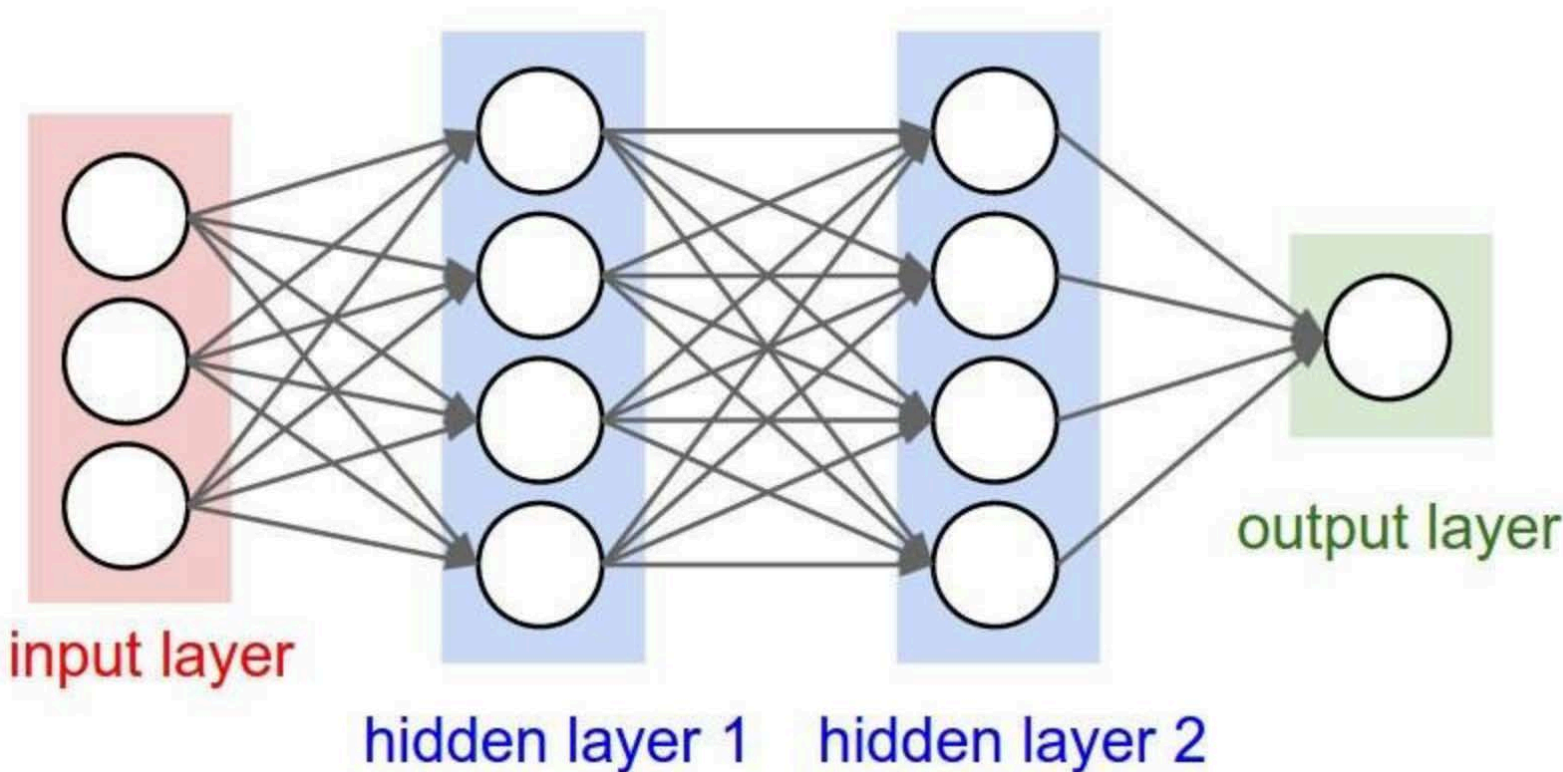
- ▶ Understanding deep learning
- ▶ Finding data on handwritten English letters
 - ▶ UCI Machine learning repository, Github, kaggle...
 - ▶ Narrowed down to 3 ideal training sets
 - ▶ 1、 Png picture set 2、 a csv file 3、 natural images
 - ▶ Finally we chose the csv file as our data
- ▶ Processing the data
 - ▶ Figure out the structure and content

Computing techniques

- ▶ Deep Learning
- ▶ CNN
- ▶ Keras (A Python Deep Learning library)
- ▶ Tensorflow(A open-source machine learning framework)
- ▶ Our resources:
 - ▶ Kaggle

What is Deep Learning????

- By
- Ar
- ur
- al
- br
- Th
- yc
- at
- a



arning in
arning
s
of the

od is that
outer will
hen give
yers.

CNN

- ▶ Effective in image recognition
- ▶ CNN's can **drastically reduce the number of parameters** that need to be tuned. Therefore, CNN's can **efficiently** handle the high dimensionality of raw images.



CODES

#step 1 Import libraries and modules

```
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Dropout
from keras.layers import Flatten
from keras.layers.convolutional import Conv2D
from keras.layers.convolutional import MaxPooling2D
from keras import backend as K
from keras.utils import np_utils
from sklearn.model_selection import train_test_split
import numpy as np
```

```
# seed for reproducing same results
seed = 785
np.random.seed(seed)
```

#Step 2 Load CSV file

```
# load dataset  
dataset = np.loadtxt('A_Z Handwritten Data.csv', delimiter=',')
```

```
# split into input and output variables  
X = dataset[:,0:784]  
Y = dataset[:,0]  
  
(X_train, X_test, Y_train, Y_test) = train_test_split(X, Y, test_size=0.50, random_state=seed)
```

Step 3 Preprocessing input data for keras

```
# reshape the data
X_train = X_train.reshape(X_train.shape[0], 28, 28, 1).astype('float32')
X_test = X_test.reshape(X_test.shape[0], 28, 28, 1).astype('float32')

# normalize data values to the range [0, 1]
X_train = X_train / 255
X_test = X_test / 255
|
Y_train = np_utils.to_categorical(Y_train)
Y_test = np_utils.to_categorical(Y_test)

num_classes = Y_test.shape[1]
```

Step 4 Define model structure

```
# create model
model = Sequential()
model.add(Conv2D(32, (5, 5), input_shape=(28, 28, 1), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.2))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dense(num_classes, activation='softmax'))
```

#Step 5 Compile model and fit model on training sets

```
# Compile model and fit models on training datas  
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])  
model.fit(X_train, Y_train, validation_data=(X_test, Y_test), epochs=10, batch_size=200, verbose=2)
```

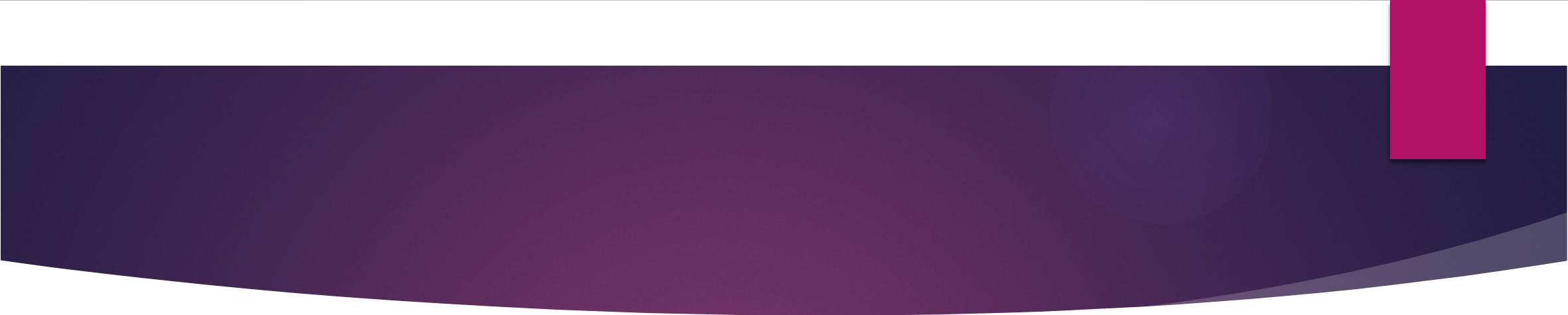

Step 6 Evaluate the model and save the model

```
# Final evaluation of the model
scores = model.evaluate(X_test,Y_test, verbose=0)
print("CNN Error: %.2f%%" % (100-scores[1]*100))

model.save('weights.model')
```

Let's Try recognizing some English letters!

```
from PIL import Image
s = "abcdefghijklmnopqrstuvwxyz"
img=Image.open('v.png')
img = np.array(img)
y_pred = model.predict_classes(img.reshape((1, 28, 28, 1)))
print("The Letter Is"+" "+s[y_pred[0]]+"!"+":")
```



SUMMARY



THANKS FOR LISTENING

孟凡佛
孙阿敏
束玥璘