

Министерство образования Республики Беларусь
Учреждение образования
"Белорусский Государственный университет информатики
и радиоэлектроники"

Лабораторная работа №4
“Реализация приложения по распознаванию номеров домов”
по учебной дисциплине “Машинное обучение”

Выполнил:

Студент гр. 956241 Дубовик Н.О.

Минск 2020

Данные: Набор изображений из Google Street View с изображениями номеров домов, содержащий 10 классов, соответствующих цифрам от 0 до 9.

- 73257 изображений цифр в обучающей выборке;
- 26032 изображения цифр в тестовой выборке;
- 531131 изображения, которые можно использовать как дополнение к обучающей выборке;
- В двух форматах:
 - Оригинальные изображения с выделенными цифрами;
 - Изображения размером 32×32 , содержащих одну цифру;
- Данные первого формата можно скачать по ссылкам:
 - <http://ufldl.stanford.edu/housenumbers/train.tar.gz> (обучающая выборка);
 - <http://ufldl.stanford.edu/housenumbers/test.tar.gz> (тестовая выборка);
 - <http://ufldl.stanford.edu/housenumbers/extra.tar.gz> (дополнительные данные);
- Данные второго формата можно скачать по ссылкам:
 - http://ufldl.stanford.edu/housenumbers/train_32x32.mat (обучающая выборка);
 - http://ufldl.stanford.edu/housenumbers/test_32x32.mat (тестовая выборка);
 - http://ufldl.stanford.edu/housenumbers/extra_32x32.mat (дополнительные данные);
- Описание данных на английском языке доступно по ссылке: <http://ufldl.stanford.edu/housenumbers/>

Результат выполнения заданий опишите в отчете.

Задание 1.

Реализуйте глубокую нейронную сеть (полносвязную или сверточную) и обучите ее на синтетических данных (например, наборы MNIST (<http://yann.lecun.com/exdb/mnist/>) или notMNIST).

Ознакомьтесь с имеющимися работами по данной тематике: англоязычная статья (<http://static.googleusercontent.com/media/research.google.com/en//pubs/archive/42241.pdf>), видео на YouTube (https://www.youtube.com/watch?v=vGPI_JvLoN0).

Для решения поставленной задачи была создана следующая архитектура нейронной сети:

- 8 сверточных слоев;
- 2 полносвязных слоя;
- 6 выходящих слоев;
- первый слой определяет количество цифр в номере дома;

- второй, третий, четвёртый, пятый и шестой выходящий слой определяет первую цифру, вторую, третью, четвёртую и пятую цифру в номере соответственно;
- использовался оптимизатор Adam;
- функция потерь “sparse_categorical_crossentropy”.

На рисунке 1 представлена таблица, описывающая получившуюся архитектуру. Данная таблица была получена с помощью метода model.summary().

Layer (type)	Output Shape	Param #	Connected to
input (InputLayer)	[(None, 128, 128, 3)]	0	
conv2d (Conv2D)	(None, 128, 128, 48)	3648	input[0][0]
batch_normalization (BatchNormaliza	(None, 128, 128, 48)	192	conv2d[0][0]
max_pooling2d (MaxPooling2D)	(None, 64, 64, 48)	0	batch_normalization[0][0]
dropout (Dropout)	(None, 64, 64, 48)	0	max_pooling2d[0][0]
conv2d_1 (Conv2D)	(None, 64, 64, 64)	76864	dropout[0][0]
batch_normalization_1 (BatchNor	(None, 64, 64, 64)	256	conv2d_1[0][0]
max_pooling2d_1 (MaxPooling2D)	(None, 64, 64, 64)	0	batch_normalization_1[0][0]
dropout_1 (Dropout)	(None, 64, 64, 64)	0	max_pooling2d_1[0][0]
conv2d_2 (Conv2D)	(None, 64, 64, 128)	204928	dropout_1[0][0]
batch_normalization_2 (BatchNor	(None, 64, 64, 128)	512	conv2d_2[0][0]
max_pooling2d_2 (MaxPooling2D)	(None, 32, 32, 128)	0	batch_normalization_2[0][0]
dropout_2 (Dropout)	(None, 32, 32, 128)	0	max_pooling2d_2[0][0]
conv2d_3 (Conv2D)	(None, 32, 32, 160)	512160	dropout_2[0][0]
batch_normalization_3 (BatchNor	(None, 32, 32, 160)	640	conv2d_3[0][0]
max_pooling2d_3 (MaxPooling2D)	(None, 32, 32, 160)	0	batch_normalization_3[0][0]
dropout_3 (Dropout)	(None, 32, 32, 160)	0	max_pooling2d_3[0][0]
conv2d_4 (Conv2D)	(None, 32, 32, 192)	768192	dropout_3[0][0]
batch_normalization_4 (BatchNor	(None, 32, 32, 192)	768	conv2d_4[0][0]
max_pooling2d_4 (MaxPooling2D)	(None, 16, 16, 192)	0	batch_normalization_4[0][0]
dropout_4 (Dropout)	(None, 16, 16, 192)	0	max_pooling2d_4[0][0]
conv2d_5 (Conv2D)	(None, 16, 16, 192)	921792	dropout_4[0][0]

Рисунок 1 – Архитектура созданной нейронной сети

max_pooling2d_5 (MaxPooling2D)	(None, 16, 16, 192)	0	batch_normalization_5[0][0]
dropout_5 (Dropout)	(None, 16, 16, 192)	0	max_pooling2d_5[0][0]
conv2d_6 (Conv2D)	(None, 16, 16, 192)	921792	dropout_5[0][0]
batch_normalization_6 (BatchNor	(None, 16, 16, 192)	768	conv2d_6[0][0]
max_pooling2d_6 (MaxPooling2D)	(None, 8, 8, 192)	0	batch_normalization_6[0][0]
dropout_6 (Dropout)	(None, 8, 8, 192)	0	max_pooling2d_6[0][0]
conv2d_7 (Conv2D)	(None, 8, 8, 192)	921792	dropout_6[0][0]
batch_normalization_7 (BatchNor	(None, 8, 8, 192)	768	conv2d_7[0][0]
max_pooling2d_7 (MaxPooling2D)	(None, 8, 8, 192)	0	batch_normalization_7[0][0]
dropout_7 (Dropout)	(None, 8, 8, 192)	0	max_pooling2d_7[0][0]
flatten (Flatten)	(None, 12288)	0	dropout_7[0][0]
dense (Dense)	(None, 4096)	50335744	flatten[0][0]
dense_1 (Dense)	(None, 4096)	16781312	dense[0][0]
length (Dense)	(None, 6)	24582	dense_1[0][0]
digit1 (Dense)	(None, 11)	45067	dense_1[0][0]
digit2 (Dense)	(None, 11)	45067	dense_1[0][0]
digit3 (Dense)	(None, 11)	45067	dense_1[0][0]
digit4 (Dense)	(None, 11)	45067	dense_1[0][0]
digit5 (Dense)	(None, 11)	45067	dense_1[0][0]
=====			

Рисунок 1, лист 2

Задание 2.

После уточнения модели на синтетических данных попробуйте обучить ее на реальных данных (набор Google Street View). Что изменилось в модели?

После обучения на реальных данных были получены следующие результаты:

Точность на обучающей выборке составляет 98.64% для первого слоя; 95.60% для второго слоя; 94.35% для третьего слоя; 96.99 % для четвертого слоя; 99.48% для пятого слоя; 99.97% для шестого слоя.

Точность на валидационной выборке составляет 89.47% для первого слоя; 85.35% для второго слоя; 81.51% для третьего слоя; 88.48% для четвертого слоя; 97.72% для пятого слоя; 99.98% для шестого слоя.

Точность на тестовой выборке составляет 80.76% для первого слоя; 82.15% для второго слоя; 80.77% для третьего слоя; 86.03% для четвертого слоя; 99.12% для пятого слоя; 99.98% для шестого слоя.

Задание 3.

Реализуйте приложение для ОС Android, которое может распознавать цифры в номерах домов, используя разработанный ранее классификатор. Какова доля правильных классификаций?

Нейронная сеть была использована для распознавания четырех изображений с камеры телефона.

В результате только два изображения из четырех были распознаны верно, рисунок 2.



Рисунок 2 – Распознавание сфотографированных номеров домов