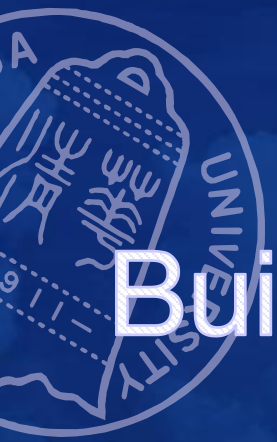


# tf.data
# Build TensorFlow input pipelines


黄世宇

huangsy1314@163.com

清华大学计算机科学与技术系

人工智能所


2019-11-20

# 目录

- 用tf.data读取数据
  - list numpy 文本 表格 图片 TFRecord
- 减小内存开销
- 打乱数据
- 数据预处理
- 对类别不平衡数据的处理

# 用tf.data读取list

## tf.data.Dataset.from_tensor_slices

```
dataset = tf.data.Dataset.from_tensor_slices([8, 3, 0, 8, 2, 1])
dataset
```

```
<TensorSliceDataset shapes: (), types: tf.int32>
```

# 用tf.data读取numpy数据

## tf.data.Dataset.from_tensor_slices

```
images, labels = train
images = images/255

dataset = tf.data.Dataset.from_tensor_slices((images, labels))
dataset
```

```
<TensorSliceDataset shapes: ((28, 28), ()), types: (tf.float64, tf.uint8)>
```

# tf.data减少内存开销

## tf.data.Dataset.from_generator

数据产生函数：

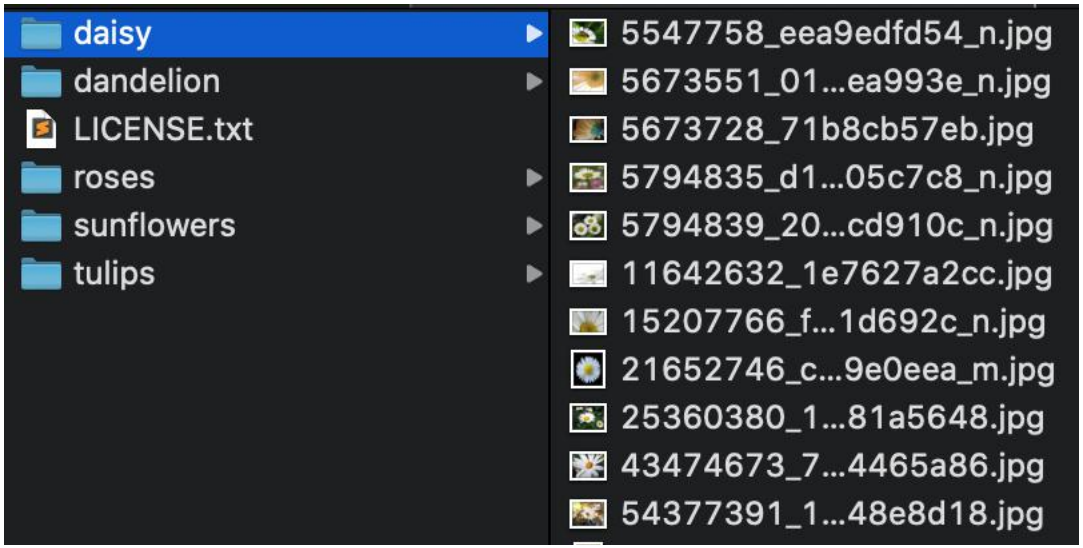
```
def count(stop):
  i = 0
  while i<stop:
    yield i
    i += 1
```

```
ds_counter = tf.data.Dataset.from_generator(count, args=[25])
```

# 利用高级API读取图片

tf.keras.preprocessing.image.ImageDataGenerator

图片数据：




```
img_gen = tf.keras.preprocessing.image.ImageDataGenerator(rescale=1./255, rotation_range=20)
```

```
images, labels = next(img_gen.flow_from_directory(flowers))
```

# 用tf.data读取TFRecord数据

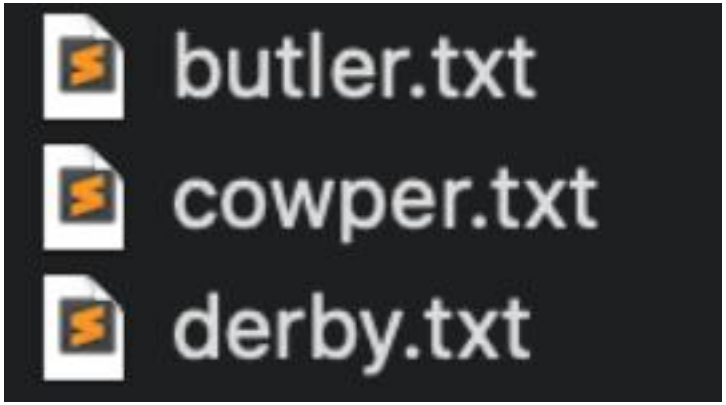
## tf.data.Dataset.from_generator

```
dataset = tf.data.TFRecordDataset(filenames = [fsns_test_file])
dataset
```

```
<TFRecordDatasetV2 shapes: (), types: tf.string>
```

# 用tf.data读取文本数据

## tf.data.TextLineDataset




butler.txt

```
Sing, O goddess, the anger of Achilles son of Peleus, that brought
countless ills upon the Achaeans. Many a brave soul did it send
hurrying down to Hades, and many a hero did it yield a prey to dogs and
vultures, for so were the counsels of Jove fulfilled from the day on
which the son of Atreus, king of men, and great Achilles, first fell
out with one another.
And which of the gods was it that set them on to quarrel? It was the
son of Jove and Leto; for he was angry with the king and sent a
pestilence upon the host to plague the people, because the son of
Atreus had dishonoured Chryses his priest. Now Chryses had come to the
ships of the Achaeans to free his daughter, and had brought with him a
great ransom: moreover he bore in his hand the sceptre of Apollo
wreathed with a suppliant's wreath, and he besought the Achaeans, but
most of all the two sons of Atreus, who were their chiefs.
"Sons of Atreus," he cried, "and all other Achaeans, may the gods who
dwell in Olympus grant you to sack the city of Priam, and to reach your
homes in safety; but free my daughter, and accept a ransom for her, in
reverence to Apollo, son of Jove."
On this the rest of the Achaeans with one voice were for respecting the
priest and taking the ransom that he offered; but not so Agamemnon, who
spoke fiercely to him and sent him roughly away. "Old man," said he,
"let me not find you tarrying about our ships, nor yet coming
hereafter. Your sceptre of the god and your wreath shall profit you
nothing. I will not free her. She shall grow old in my house at Argos
```

```
dataset = tf.data.TextLineDataset(file_paths)
```

# 用tf.data读取表格数据

## tf.data.experimental.make_csv_dataset

| | survived | sex | age | n_siblings_spouses | parch | fare | class | deck | embark_town | alone |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | male | 22.0 | 1 | 0 | 7.2500 | Third | unknown | Southampton | n |
| **1** | 1 | female | 38.0 | 1 | 0 | 71.2833 | First | C | Cherbourg | n |
| **2** | 1 | female | 26.0 | 0 | 0 | 7.9250 | Third | unknown | Southampton | y |
| **3** | 1 | female | 35.0 | 1 | 0 | 53.1000 | First | C | Southampton | n |
| **4** | 0 | male | 28.0 | 0 | 0 | 8.4583 | Third | unknown | Queenstown | y |

```
titanic_batches = tf.data.experimental.make_csv_dataset(
    titanic_file, batch_size=4,
    label_name="survived")
```

# 打乱数据

```
dataset.shuffle(buffer_size=100)
```

# 数据预处理

dataset.map(你的预处理函数)

```
import scipy.ndimage as ndimage

def random_rotate_image(image):
  image = ndimage.rotate(image, np.random.uniform(-30, 30), reshape=False)
  return image
```

# 对类别不平衡数据的处理

tf.data.experimental.sample_from_datasets

```
balanced_ds = tf.data.experimental.sample_from_datasets(
    [negative_ds, positive_ds], [0.5, 0.5]).batch(10)
```

# 对类别不平衡数据的处理

## 方法一：
## tf.data.experimental.sample_from_datasets

```
balanced_ds = tf.data.experimental.sample_from_datasets(
    [negative_ds, positive_ds], [0.5, 0.5]).batch(10)
```

## 方法二：
## tf.data.experimental.rejection_resample

```
resampler = tf.data.experimental.rejection_resample(
    class_func, target_dist=[0.5, 0.5], initial_dist=fractions)
```

```
resample_ds = creditcard_ds.unbatch().apply(resampler)
```

# 谢 谢！