1 Sentim

(https://colab.r
/TannerGilbert/T
Tutorials/6.
/Se

http://www.polyvista.com/b

1.1 What is

Sentiment Analysis
extraction and stud
text analysis, comp

1.2 Sentim Neural Netv

We will use a RNN
the data.

1.2.1 Imports

In [1]:
```
import re
import nu
import pa
from skle
import ma

from tens
from tens
from tens
from tens
from tens

from atte

# import
# from ke
# from ke
# from ke
# from ke
import te
from tens
```

In [2]:
```
from tens
from tens

config =
config.gp
config.gp
session =
```

In [3]:
```
from IPyt
Interacti
```

In [4]:
```
value = t
    cuda_
    min_c
)
print ('*
```

```
WARNING:tensor
_gpu_available
is deprecated
Instructions f
Use `tf.config
***If TF can a

 True
```

Contents

In [5]:
```
1 value = t
2 print(val
```

```
[PhysicalDevic
GPU')]
```

In [6]:
```
1 print(dev
```

```
[name: "/devic
device_type: "
memory_limit:
locality {
}
incarnation: 7
, name: "/devi
device_type: "
memory_limit:
locality {
}
incarnation: 1
physical_devic
, name: "/devi
device_type: "
memory_limit:
locality {
}
incarnation: 1
physical_devic
, name: "/devi
device_type: "
memory_limit:
locality {
  bus_id: 1
  links {
  }
}
incarnation: 4
physical_devic
us id: 0000:02
]
```

In [7]:
```
1 tf.debugg
```

In [8]:
```
1  tf
2 print("Nu
3
```

Out[8]:
```
<module 'tenso
nv-tensorflow/
_.py'>

Num GPUs Avail
```

Contents

2 Place te

3 with tf.

a = tf.constant([[1.0
[5.0, 6.0]])

c = tf.matmul(a, b)

3.0.1 Loading

In [9]:

```
1 data1 = p
2 data2 = p
3 # data1 =
4 # data2 =
5 print(dat
6 print(dat
7
8 data1.hea
9 data2.hea
```

```
(14640, 15)
(1600000, 2)
```

Out[9]:

	twe
0	57030613367776
1	57030113088812
2	57030108367281
3	57030103140762
4	57030081707446

Out[9]:

	sentiment	
0	negative	@
1	negative	is
2	negative	@Ke
3	negative	
4	negative	@

Removing all colum

In [10]:
```
1 data1 = d
2 new_colum
3 data1.col
4 data1.hea
```

Out[10]:

	sentiment	
0	neutral	
1	positive	@Vi
2	neutral	@
3	negative	
4	negative	

In [11]:
```
1 df = data
2 print(df.
3 df
```

```
(1614640, 2)
```

Out[11]:

	sentimen
0	neutra
1	positive
2	neutra
3	negative
4	negative
...	..
1614635	positive
1614636	positive
1614637	positive
1614638	positive
1614639	positive

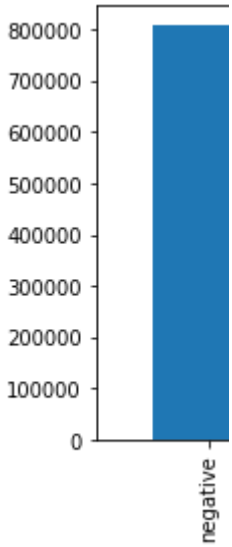
1614640 rows × 2 c

3.0.2 Data exp

Contents

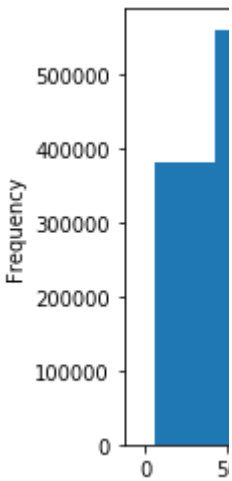
In [12]:
```
1 df['senti
```

Out[12]: <matplotlib.ax



In [13]:
```
1 df['text'
```

Out[13]: <matplotlib.ax



3.0.3 Preproc

In [14]:
```
1 # How much
2 frac = 0
```

In [15]:
```
# data['t
# data.he
df = df.s
df
```

Out[15]:

	sentiment
0	positive
1	positive
2	negative
3	positive
4	negative
...	...
807315	positive
807316	negative
807317	positive
807318	positive
807319	negative

807320 rows × 2 co

In [16]:
```
df['text'
df['text'
df['text'
```

Out[16]:
```
0          it d
1          @nev
2          what
3
4          @kat

807315
807316    @cai
(http://myloc.
807317    i so
807318    @kel
807319    @and
Name: text, Le
```

Out[16]:
```
0    It doesnt
1    NeverSlee
2    what a sa
3
4    katecamer
Name: text, dt
```

In [17]:
```
1 df['senti
```

Out[17]:
```
0          posi
1          posi
2          nega
3          posi
4          nega
             ..
807315     posi
807316     nega
807317     posi
807318     posi
807319     nega
Name: sentimen
```

In [18]:
```
1 df = df[d
```

In [19]:
```
1 df
```

Out[19]:

	sentiment
0	positive
1	positive
2	negative
3	positive
4	negative
...	...
807315	positive
807316	negative
807317	positive
807318	positive
807319	negative

805787 rows × 2 co

In [20]:
```
1 vocabular
```

Contents

In [21]:
```
1 tokenizer
2 tokenizer
3
4 X = token
5 X = pad_s
6 X[:5]
```

Out[21]:
```
array([[    0,
0,     0,
           0,
0,     0,
           0,
0,     0,
           0,
81,     4,
          29,
       [    0,
0,     0,
           0,
0,     0,
           0,
0,     0,
           0,
88,   313,
           7,
       [    0,
0,     0,
           0,
0,     0,
           0,
0,     0,
           0,
8,  1679,
          33,
       [    0,
0,     0,
           0,
0,     0,
           0,
0,     0,
           0,
0,   117,
          29,
       [    0,
0,     0,
           0,
0,     1,
          48,
30,    31,
       19171,
3,  5210,
           2,
```

3.0.4 Creating

Contents

Contents

In [22]:

```
1 model = S
2 model.add
3 model.add
4 model.add
5 model.add
6 model.add
7 model.add
8 model.add
9 model.add
```

```
Executing op R
0/task:0/devic
Executing op S
evice:CPU:0
Executing op M
evice:CPU:0
Executing op A
evice:CPU:0
Executing op V
task:0/device:
Executing op V
lica:0/task:0/
Executing op L
ask:0/device:C
Executing op A
0/device:CPU:0
Executing op A
ca:0/task:0/de
Executing op R
0/task:0/devic
Executing op S
evice:GPU:0
Executing op M
evice:GPU:0
Executing op A
evice:GPU:0
Executing op V
task:0/device:
Executing op V
lica:0/task:0/
Executing op L
ask:0/device:G
Executing op A
0/device:GPU:0
Executing op A
ca:0/task:0/de
Executing op R
eplica:0/task:
Executing op Q
vice:GPU:0
Executing op D
k:0/device:GPU
Executing op S
device:GPU:0
Executing op T
sk:0/device:GP
```

Contents

In [23]:
```
1 model.com
2 model.sum
```

```
Executing op V
task:0/device:
Model: "sequen
_________________
_____
Layer (type)
#
==============
=====
embedding (Emb
00
_________________
_____
dropout (Dropo
_________________
_____
bidirectional
4
_________________
_____
dropout_1 (Dro
_________________
_____
bidirectional_
20
_________________
_____
attention (Att
8
_________________
_____
batch_normaliz
_________________
_____
dense (Dense)
==============
=====
Total params:
Trainable para
Non-trainable
_________________
_____
```

In [24]:
```
1 y = pd.ge
2 [print(df
```

```
positive [0 1]
positive [0 1]
negative [1 0]
positive [0 1]
negative [1 0]
```

Out[24]: [None, None, N

In [25]:
```
1 X_train
```

3.0.5 Training

In [26]:
```
1  batch_siz
2  epochs =
3  import ti
4  from date
5  datetime
6  csv_logge
7  start = t
8  history =
9  end = tim
10 elapsed =
11 print(ela
12
```

```
Executing op R
/task:0/device
Executing op R
0/task:0/devic
Executing op M
ask:0/device:C
Executing op P
a:0/task:0/dev
Executing op F
a:0/task:0/dev
Executing op T
0/task:0/devic
Executing op R
0/task:0/devic
Executing op Z
ask:0/device:C
Executing op P
lica:0/task:0/
Executing op D
```

3.0.6 Plotting

In [27]:
```
1 # print(h
```

Contents

In [28]:
```
 1 import ma
 2
 3 # Plot tr
 4 plt.plot(
 5 # plt.plo
 6 plt.title
 7 plt.ylabe
 8 plt.xlabe
 9 plt.legen
10 plt.show(
11
12 # Plot tr
13 plt.plot(
14 # plt.plo
15 plt.title
16 plt.ylabe
17 plt.xlabe
18 plt.legen
19 plt.show(
20
```

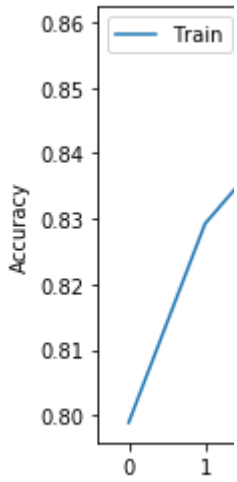
Out[28]: [<matplotlib.l

Out[28]: Text(0.5, 1.0,

Out[28]: Text(0, 0.5, '

Out[28]: Text(0.5, 0, '

Out[28]: <matplotlib.le



Out[28]: [<matplotlib.l

Out[28]: Text(0.5, 1.0,

Out[28]: Text(0, 0.5, '

Out[28]: Text(0.5, 0, '

Out[28]: <matplotlib.le

3.0.7 Testing

In [29]:
```
1 predictio
2
3 [print(df
```

```
Executing op R
/task:0/device
Executing op R
0/task:0/devic
Executing op M
ask:0/device:C
Executing op P
a:0/task:0/dev
Executing op F
a:0/task:0/dev
Executing op T
0/task:0/devic
Executing op R
0/task:0/devic
Executing op Z
ask:0/device:C
Executing op P
lica:0/task:0/
Executing op M
/task:0/device
Executing op A
plica:0/task:0
Executing op _
ce /job:localh
It doesnt get
79671234 0.203
NeverSleeps76
10   [0.970198
what a sad dep
290171  0.2708
honeym549 very
katecameron200
hucked off Twi
[0.54774606 0.
```

Out[29]:
```
[None, None, N
```

Contents

In [30]:
```
accurate_
for i, pr
    if np
        a
    else:
        i

total_pre
print('Nu
print('Nu
print('Nu
print('Ac
```

```
Number of pred
Number of accu
Number of fals
Accuracy:  0.8
```

In [31]:
```
name = 'S
```

In [32]:
```
model_sav
```

```
Executing op R
a:0/task:0/dev
Executing op I
k:0/device:GPU
Executing op R
a:0/task:0/dev
Executing op R
a:0/task:0/dev
Executing op I
k:0/device:GPU
```

In [33]:

```
# pos_cou
# real_po
# for i,
#     if
#
#     eli
#
#     els
#

#     if
#
#     eli
#
#     els
#

# print('
# print('
# print('
# print('
# print('
# print('
```

In [34]:

```
# !jupyte
```

3.1 Improv

- Weight classes
- Train more epo
- Use bigger net
- Try other word

3.2 Resour

- Recurrent Neu (https://prograr explained/recu
- Sentiment Ana /wiki/Sentimen
- What is the be (https://www.q Python-I%E2% an-emoticon-d me-with-that)
- How to Do Ser /watch?v=si8z

Contents