text classification

July 23, 2023

```python
[50]: import numpy as np
      import tensorflow as tf
      import tensorflow_hub as hub
      import tensorflow_datasets as tfds
```

```python
[51]: train, validation_data, test = tfds.load(name = 'imdb_reviews',
                                  split = ('train[:60%]','train[60%:]','test'),
                                  as_supervised = True) #:60% - 60% data for
        ↪training, 60%: - 40% for validation
```

```python
[52]: train
```

```
[52]: <_PrefetchDataset element_spec=(TensorSpec(shape=(), dtype=tf.string,
      name=None), TensorSpec(shape=(), dtype=tf.int64, name=None))>
```

```python
[53]: train_eg, train_labels = next(iter(train.batch(10)))
```

```python
[54]: train_eg
```

```
[54]: <tf.Tensor: shape=(10,), dtype=string, numpy=
      array([b"This was an absolutely terrible movie. Don't be lured in by Christopher
      Walken or Michael Ironside. Both are great actors, but this must simply be their
      worst role in history. Even their great acting could not redeem this movie's
      ridiculous storyline. This movie is an early nineties US propaganda piece. The
      most pathetic scenes were those when the Columbian rebels were making their
      cases for revolutions. Maria Conchita Alonso appeared phony, and her pseudo-love
      affair with Walken was nothing but a pathetic emotional plug in a movie that was
      devoid of any real meaning. I am disappointed that there are movies like this,
      ruining actor's like Christopher Walken's good name. I could barely sit through
      it.",
             b'I have been known to fall asleep during films, but this is usually due
      to a combination of things including, really tired, being warm and comfortable
      on the sette and having just eaten a lot. However on this occasion I fell asleep
      because the film was rubbish. The plot development was constant. Constantly slow
      and boring. Things seemed to happen, but with no explanation of what was causing
      them or why. I admit, I may have missed part of the film, but i watched the
      majority of it and everything just seemed to happen of its own accord without
      any real concern for anything else. I cant recommend this film at all.',
```

b'Mann photographs the Alberta Rocky Mountains in a superb fashion, and Jimmy Stewart and Walter Brennan give enjoyable performances as they always seem to do.

But come on Hollywood - a Mountie telling the people of Dawson City, Yukon to elect themselves a marshal (yes a marshal!) and to enforce the law themselves, then gunfighters battling it out on the streets for control of the town?

Nothing even remotely resembling that happened on the Canadian side of the border during the Klondike gold rush. Mr. Mann and company appear to have mistaken Dawson City for Deadwood, the Canadian North for the American Wild West.

Canadian viewers be prepared for a Reefer Madness type of enjoyable howl with this ludicrous plot, or, to shake your head in disgust.',

b'This is the kind of film for a snowy Sunday afternoon when the rest of the world can go ahead with its own business as you descend into a big arm-chair and mellow for a couple of hours. Wonderful performances from Cher and Nicolas Cage (as always) gently row the plot along. There are no rapids to cross, no dangerous waters, just a warm and witty paddle through New York life at its best. A family film in every sense and one that deserves the praise it received.',

b'As others have mentioned, all the women that go nude in this film are mostly absolutely gorgeous. The plot very ably shows the hypocrisy of the female libido. When men are around they want to be pursued, but when no "men" are around, they become the pursuers of a 14 year old boy. And the boy becomes a man really fast (we should all be so lucky at this age!). He then gets up the courage to pursue his true love.',

b"This is a film which should be seen by anybody interested in, effected by, or suffering from an eating disorder. It is an amazingly accurate and sensitive portrayal of bulimia in a teenage girl, its causes and its symptoms. The girl is played by one of the most brilliant young actresses working in cinema today, Alison Lohman, who was later so spectacular in 'Where the Truth Lies'. I would recommend that this film be shown in all schools, as you will never see a better on this subject. Alison Lohman is absolutely outstanding, and one marvels at her ability to convey the anguish of a girl suffering from this compulsive disorder. If barometers tell us the air pressure, Alison Lohman tells us the emotional pressure with the same degree of accuracy. Her emotional range is so precise, each scene could be measured microscopically for its gradations of trauma, on a scale of rising hysteria and desperation which reaches unbearable intensity. Mare Winningham is the perfect choice to play her mother, and does so with immense sympathy and a range of emotions just as finely tuned as Lohman's. Together, they make a pair of sensitive emotional oscillators vibrating in resonance with one another. This film is really an astonishing achievement, and director Katt Shea should be proud of it. The only reason for not seeing it is if you are not interested in people. But even if you like nature films best, this is after all animal behaviour at the sharp edge. Bulimia is an extreme version of how a tormented soul can destroy her own body in a frenzy of despair. And if we don't sympathise with people suffering from the depths of despair, then we are dead inside.",

b'Okay, you have:

Penelope Keith as Miss Herringbone-Tweed,

B.B.E. (Backbone of England.) She's killed off in the first scene - that's right, folks; this show has no backbone!

Peter O'Toole as Ol' Colonel Cricket from The First War and now the emblazered Lord of the Manor.

Joanna Lumley as the ensweatered Lady of the Manor, 20 years younger than the colonel and 20 years past her own prime but still glamourous (Brit spelling, not mine) enough to have a toy-boy on the side. It's alright, they have Col. Cricket's full knowledge and consent (they guy even comes 'round for Christmas!) Still, she's considerate of the colonel enough to have said toy-boy her own age (what a gal!)

David McCallum as said toy-boy, equally as pointlessly glamourous as his squeeze. Pilcher couldn't come up with any cover for him within the story, so she gave him a hush-hush job at the Circus.

and finally:

Susan Hampshire as Miss Polonia Teacups, Venerable Headmistress of the Venerable Girls' Boarding-School, serving tea in her office with a dash of deep, poignant advice for life in the outside world just before graduation. Her best bit of advice: "I've only been to Nancherrow (the local Stately Home of England) once. I thought it was very beautiful but, somehow, not part of the real world." Well, we can't say they didn't warn us.

Ah, Susan - time was, your character would have been running the whole show. They don't write 'em like that any more. Our loss, not yours.

So - with a cast and setting like this, you have the re-makings of "Brideshead Revisited," right?

Wrong! They took these 1-dimensional supporting roles because they paid so well. After all, acting is one of the oldest temp-jobs there is (YOU name another!)

First warning sign: lots and lots of backlighting. They get around it by shooting outdoors - "hey, it's just the sunlight!"

Second warning sign: Leading Lady cries a lot. When not crying, her eyes are moist. That's the law of romance novels: Leading Lady is "dewy-eyed."

Henceforth, Leading Lady shall be known as L.L.

Third warning sign: L.L. actually has stars in her eyes when she's in love. Still, I'll give Emily Mortimer an award just for having to act with that spotlight in her eyes (I wonder . did they use contacts?)

And lastly, fourth warning sign: no on-screen female character is "Mrs." She's either "Miss" or "Lady."

When all was said and done, I still couldn't tell you who was pursuing whom and why. I couldn't even tell you what was said and done.

To sum up: they all live through World War II without anything happening to them at all.

OK, at the end, L.L. finds she's lost her parents to the Japanese prison camps and baby sis comes home catatonic. Meanwhile (there's always a "meanwhile,") some young guy L.L. had a crush on (when, I don't know) comes home from some wartime tough spot and is found living on the street by Lady of the Manor (must be some street if SHE's going to find him there.) Both war casualties are whisked away to recover at Nancherrow (SOMEBODY has to be "whisked away" SOMEWHERE in these romance stories!)

Great drama.',

    b'The film is based on a genuine 1950s novel.

Journalist Colin McInnes wrote a set of three "London novels": "Absolute Beginners", "City of Spades" and "Mr Love and Justice". I have read all three. The first two are excellent. The last, perhaps an experiment that did not come off. But McInnes's work is highly acclaimed; and rightly so. This musical is the novelist's

ultimate nightmare - to see the fruits of one\'s mind being turned into a glitzy, badly-acted, soporific one-dimensional apology of a film that says it captures the spirit of 1950s London, and does nothing of the sort.

Thank goodness Colin McInnes wasn\'t alive to witness it.',
       b'I really love the sexy action and sci-fi films of the sixties and its because of the actress\'s that appeared in them. They found the sexiest women to be in these films and it didn\'t matter if they could act (Remember "Candy"?). The reason I was disappointed by this film was because it wasn\'t nostalgic enough. The story here has a European sci-fi film called "Dragonfly" being made and the director is fired. So the producers decide to let a young aspiring filmmaker (Jeremy Davies) to complete the picture. They\'re is one real beautiful woman in the film who plays Dragonfly but she\'s barely in it. Film is written and directed by Roman Coppola who uses some of his fathers exploits from his early days and puts it into the script. I wish the film could have been an homage to those early films. They could have lots of cameos by actors who appeared in them. There is one actor in this film who was popular from the sixties and its John Phillip Law (Barbarella). Gerard Depardieu, Giancarlo Giannini and Dean Stockwell appear as well. I guess I\'m going to have to continue waiting for a director to make a good homage to the films of the sixties. If any are reading this, "Make it as sexy as you can"! I\'ll be waiting!',
       b'Sure, this one isn\'t really a blockbuster, nor does it target such a position. "Dieter" is the first name of a quite popular German musician, who is either loved or hated for his kind of acting and thats exactly what this movie is about. It is based on the autobiography "Dieter Bohlen" wrote a few years ago but isn\'t meant to be accurate on that. The movie is filled with some sexual offensive content (at least for American standard) which is either amusing (not for the other "actors" of course) or dumb - it depends on your individual kind of humor or on you being a "Bohlen"-Fan or not. Technically speaking there isn\'t much to criticize. Speaking of me I find this movie to be an OK-movie.'],
      dtype=object)>

[55]: train_labels

[55]: <tf.Tensor: shape=(10,), dtype=int64, numpy=array([0, 0, 0, 1, 1, 1, 0, 0, 0, 0], dtype=int64)>

[56]:
```
embed = "https://tfhub.dev/google/tf2-preview/gnews-swivel-20dim/1"
hub_layer = hub.KerasLayer(embed, input_shape = [], dtype = tf.string,
    trainable = True)
```

[57]:
```
hub_layer(train_eg[:3])
```

[57]: <tf.Tensor: shape=(3, 20), dtype=float32, numpy=
array([[1.765786 , -3.882232 , 3.9134233 , -1.5557289 , -3.3362343 ,
        -1.7357955 , -1.9954445 , 1.2989551 , 5.081598 , -1.1041286 ,
        -2.0503852 , -0.72675157, -0.65675956, 0.24436149, -3.7208383 ,

```
        2.0954835 ,  2.2969332 , -2.0689783 , -2.9489717 , -1.1315987 ],
      [ 1.8804485 , -2.5852382 ,  3.4066997 ,  1.0982676 , -4.056685   ,
       -4.891284  , -2.785554  ,  1.3874227 ,  3.8476458 , -0.9256538 ,
       -1.896706  ,  1.2113281 ,  0.11474707,  0.76209456, -4.8791065 ,
        2.906149  ,  4.7087674 , -2.3652055 , -3.5015898 , -1.6390051 ],
      [ 0.71152234, -0.6353217 ,  1.7385626 , -1.1168286 , -0.5451594 ,
       -1.1808156 ,  0.09504455,  1.4653089 ,  0.66059524,  0.79308075,
       -2.2268345 ,  0.07446612, -1.4075904 , -0.70645386, -1.907037   ,
        1.4419787 ,  1.9551861 , -0.42660055, -2.8022065 ,  0.43727064]],
      dtype=float32)>
```

[58]:
```python
model = tf.keras.Sequential()
model.add(hub_layer)
model.add(tf.keras.layers.Dense(16, activation = 'relu'))
model.add(tf.keras.layers.Dense(1))
```

[59]:
```python
model.summary()
```

```
Model: "sequential_4"

_____
 Layer (type)                Output Shape              Param #
=================================================================
 keras_layer_3 (KerasLayer)  (None, 20)                400020

 dense_4 (Dense)             (None, 16)                336

 dense_5 (Dense)             (None, 1)                 17


=================================================================
Total params: 400373 (1.53 MB)
Trainable params: 400373 (1.53 MB)
Non-trainable params: 0 (0.00 Byte)
_____
```

[60]:
```python
model.compile(optimizer = 'adam', loss = tf.keras.losses.
 ↪BinaryCrossentropy(from_logits=True), metrics=['accuracy'])
```

[61]:
```python
history = model.fit(train.shuffle(10000).batch(100), epochs = 25,
                    validation_data = validation_data.batch(100),verbose = 1)
```

```
Epoch 1/25
150/150 [==============================] - 2s 11ms/step - loss: 0.6821 -
accuracy: 0.5955 - val_loss: 0.5657 - val_accuracy: 0.6978
Epoch 2/25
150/150 [==============================] - 2s 10ms/step - loss: 0.5016 -
accuracy: 0.7428 - val_loss: 0.4608 - val_accuracy: 0.7646
Epoch 3/25
150/150 [==============================] - 2s 10ms/step - loss: 0.4001 -
```

```
accuracy: 0.8160 - val_loss: 0.4011 - val_accuracy: 0.7924
Epoch 4/25
150/150 [==============================] - 2s 10ms/step - loss: 0.3278 -
accuracy: 0.8548 - val_loss: 0.3492 - val_accuracy: 0.8479
Epoch 5/25
150/150 [==============================] - 2s 10ms/step - loss: 0.2753 -
accuracy: 0.8836 - val_loss: 0.3238 - val_accuracy: 0.8599
Epoch 6/25
150/150 [==============================] - 2s 10ms/step - loss: 0.2347 -
accuracy: 0.9051 - val_loss: 0.3080 - val_accuracy: 0.8634
Epoch 7/25
150/150 [==============================] - 2s 10ms/step - loss: 0.2031 -
accuracy: 0.9200 - val_loss: 0.3039 - val_accuracy: 0.8657
Epoch 8/25
150/150 [==============================] - 2s 10ms/step - loss: 0.1767 -
accuracy: 0.9333 - val_loss: 0.3000 - val_accuracy: 0.8690
Epoch 9/25
150/150 [==============================] - 2s 10ms/step - loss: 0.1544 -
accuracy: 0.9434 - val_loss: 0.3051 - val_accuracy: 0.8669
Epoch 10/25
150/150 [==============================] - 2s 10ms/step - loss: 0.1337 -
accuracy: 0.9531 - val_loss: 0.3075 - val_accuracy: 0.8701
Epoch 11/25
150/150 [==============================] - 2s 10ms/step - loss: 0.1160 -
accuracy: 0.9601 - val_loss: 0.3155 - val_accuracy: 0.8707
Epoch 12/25
150/150 [==============================] - 2s 10ms/step - loss: 0.1009 -
accuracy: 0.9671 - val_loss: 0.3281 - val_accuracy: 0.8741
Epoch 13/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0871 -
accuracy: 0.9729 - val_loss: 0.3422 - val_accuracy: 0.8721
Epoch 14/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0751 -
accuracy: 0.9792 - val_loss: 0.3505 - val_accuracy: 0.8695
Epoch 15/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0638 -
accuracy: 0.9839 - val_loss: 0.3687 - val_accuracy: 0.8680
Epoch 16/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0542 -
accuracy: 0.9871 - val_loss: 0.3840 - val_accuracy: 0.8677
Epoch 17/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0455 -
accuracy: 0.9905 - val_loss: 0.4031 - val_accuracy: 0.8653
Epoch 18/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0383 -
accuracy: 0.9917 - val_loss: 0.4250 - val_accuracy: 0.8637
Epoch 19/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0320 -
```

```
accuracy: 0.9943 - val_loss: 0.4494 - val_accuracy: 0.8647
Epoch 20/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0267 -
accuracy: 0.9959 - val_loss: 0.4605 - val_accuracy: 0.8637
Epoch 21/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0224 -
accuracy: 0.9966 - val_loss: 0.4862 - val_accuracy: 0.8620
Epoch 22/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0184 -
accuracy: 0.9975 - val_loss: 0.5122 - val_accuracy: 0.8622
Epoch 23/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0152 -
accuracy: 0.9985 - val_loss: 0.5272 - val_accuracy: 0.8615
Epoch 24/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0125 -
accuracy: 0.9989 - val_loss: 0.5504 - val_accuracy: 0.8604
Epoch 25/25
150/150 [==============================] - 2s 10ms/step - loss: 0.0104 -
accuracy: 0.9995 - val_loss: 0.5708 - val_accuracy: 0.8593
```

[62]:
```python
results = model.evaluate(test.batch(100), verbose = 2)

for name, value in zip(model.metrics_names, results):
    print("%s: %.3f"%(name, value))
```

```
250/250 - 1s - loss: 0.6346 - accuracy: 0.8413 - 1s/epoch - 4ms/step
loss: 0.635
accuracy: 0.841
```

[ ]: