

```
In [1]: %reload_ext autoreload
        %autoreload 2
        %matplotlib inline
```

```
In [2]: from fastai.text import *
```

```
In [3]: path = untar_data(URLs.YELP_REVIEWS_POLARITY)
```

Downloading https://s3.amazonaws.com/fast-ai-nlp/yelp_review_polarity_csv

```
In [4]: path.ls()
```

```
Out[4]: [PosixPath('/home/jupyter/.fastai/data/yelp_review_polarity_csv/test.csv'),
         PosixPath('/home/jupyter/.fastai/data/yelp_review_polarity_csv/readme.txt'),
         PosixPath('/home/jupyter/.fastai/data/yelp_review_polarity_csv/train.csv')]
```

```
In [27]: test = pd.read_csv(path/'test.csv', header=None)
```

```
In [28]: train = pd.read_csv(path/'train.csv', header=None)
```

```
In [29]: print(train.shape)
         print(test.shape)
```

```
(560000, 2)
(38000, 2)
```

```
In [21]: f = open((path/'readme.txt'), 'r')
```

```
In [22]: x = f.readlines()
         f.close()
```

In [23]: x

```
Out[23]: ['Yelp Review Polarity Dataset\n',
'\n',
'Version 1, Updated 09/09/2015\n',
'\n',
'ORIGIN\n',
'\n',
'The Yelp reviews dataset consists of reviews from Yelp. It is extract
ed from the Yelp Dataset Challenge 2015 data. For more information, ple
ase refer to http://www.yelp.com/dataset_challenge\n',
'\n',
'The Yelp reviews polarity dataset is constructed by Xiang Zhang (xian
g.zhang@nyu.edu) from the above dataset. It is first used as a text cla
ssification benchmark in the following paper: Xiang Zhang, Junbo Zhao,
Yann LeCun. Character-level Convolutional Networks for Text Classificat
ion. Advances in Neural Information Processing Systems 28 (NIPS 201
5).\n',
'\n',
'\n',
'DESCRIPTION\n',
'\n',
'The Yelp reviews polarity dataset is constructed by considering stars
1 and 2 negative, and 3 and 4 positive. For each polarity 280,000 train
ing samples and 19,000 testing samples are take randomly. In total ther
e are 560,000 trainig samples and 38,000 testing samples. Negative pola
rity is class 1, and positive class 2.\n',
'\n',
'The files train.csv and test.csv contain all the training samples as
comma-sparated values. There are 2 columns in them, corresponding to cl
ass index (1 and 2) and review text. The review texts are escaped using
double quotes ("), and any internal double quote is escaped by 2 double
quotes ("). New lines are escaped by a backslash followed with an "n"
character, that is "\\n".\n']
```

In [30]: train.head()

```
Out[30]:
```

	0	1
0	1	Unfortunately, the frustration of being Dr. Go...
1	2	Been going to Dr. Goldberg for over 10 years. ...
2	1	I don't know what Dr. Goldberg was like before...
3	1	I'm writing this review to give you a heads up...
4	2	All the food is great here. But the best thing...

```
In [37]: data = (TextList.from_csv(path=path, csv_name='train.csv', cols=1)
                .filter_by_rand(0.2)
                .split_by_fname_file(fname="test.csv")
                .label_from_df(cols=0)
                .databunch())
```

```
In [39]: train.head()
```

```
Out[39]:
```

	0	1
0	1	Unfortunately, the frustration of being Dr. Go...
1	2	Been going to Dr. Goldberg for over 10 years. ...
2	1	I don't know what Dr. Goldberg was like before...
3	1	I'm writing this review to give you a heads up...
4	2	All the food is great here. But the best thing...

```
In [44]: train = train.assign(is_valid=0)
```

```
In [51]: train.shape
```

```
Out[51]: (560000, 3)
```

```
In [46]: test = test.assign(is_valid=1)
```

```
In [52]: test.shape
```

```
Out[52]: (38000, 3)
```

```
In [47]: df = train.append(test)
```

```
In [50]: df.shape
```

```
Out[50]: (598000, 3)
```

```
In [53]: data = (TextList.from_df(df=df, path=path, cols=1)
               .split_from_df(col=2)
               .label_from_df(cols=0)
               .databunch())
```

```
In [77]: data_lm = (TextList.from_df(df, cols=1)
                  .filter_by_rand(0.2)
                  .split_by_rand_pct(0.2)
                  .label_for_lm()
                  .databunch(bs=48))
```

```
In [78]: data_lm.show_batch()
```

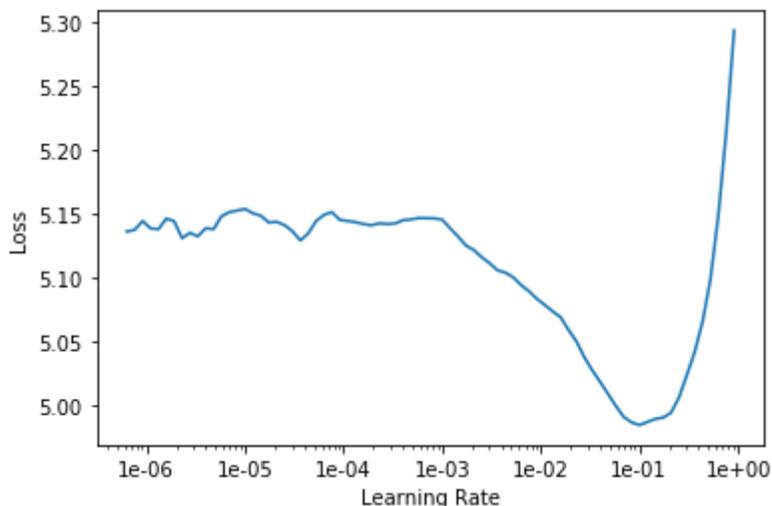
idx	text
0	xxunk . xxmaj he xxunk all options with you and is very patient and understanding . xxmaj he does n't judge and asks all the right questions . xxmaj very thorough and wants to be kept in the loop on every aspect of your medical health and your life . xxbos xxmaj all the food is great here . xxmaj but the best thing they have is their wings .
1	the carol . xxmaj of course my all time favorite is no longer on the menu , i am told if requested when not slammed , if all ingredients are there , they will make that too ! xxmaj goodness i love this place . xxmaj if you have n't eaten here , what are you waiting for ? xxbos xxmaj while driving around town my friend and i came
2	slightly wet . xxmaj the coasters are about 30 seconds long and about as scary as driving on xxup i-17 ; the more tame of the two , called " xxmaj the xxmaj patriot , " gives you two go - rounds in one sitting , but that 's not really any consolation . \n \n xxmaj most of the other stuff is pretty stock , from the arcade
3	. xxmaj then we split the xxmaj xxunk de xxmaj puerco xxmaj fritas - the pork " morsels " in a garlic - lime sauce . xxmaj the pork was very moist and tender with the right amount of crispness on the exterior . \n \n xxmaj of course we had to try dessert too , and split the xxmaj carribean xxmaj candy xxmaj bar - coconut flan with
4	it had good reviews . i got the deluxe pedicure cuz my feet really needed it . xxmaj it was only suppose to be 40, <i>butbythetimeiwasonitxxunkupbeing</i> 53 not including tip . xxmaj she paid more attention to my legs than my feet . xxmaj the massage was n't that good . xxmaj she kept massaging the top

```
In [111]: learn = language_model_learner(data_lm, AWD_LSTM, drop_mult=0.3)
```

```
In [80]: learn.lr_find()
```

LR Finder is complete, type {learner_name}.recorder.plot() to see the graph.

```
In [81]: learn.recorder.plot(skip_end=12)
```



```
In [84]: learn.fit_one_cycle(1, 3e-2, moms=(0.8, 0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	4.065816	3.930086	0.280139	12:09

```
In [85]: learn.save('fit_head')
```

```
In [87]: learn.load('fit_head');
```

```
In [88]: learn.unfreeze()
```

```
In [89]: learn.fit_one_cycle(6, 1e-3, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	3.679810	3.650271	0.309919	13:59
1	3.583807	3.553628	0.322064	13:59
2	3.498367	3.494323	0.329822	13:58
3	3.396103	3.452759	0.335527	14:01
4	3.297024	3.433400	0.338392	14:04
5	3.234129	3.435297	0.338619	14:02

```
In [90]: learn.save('fine_tuned')
```

```
In [112]: learn.load('fine_tuned');
```

```
In [101]: learn.export('fine_tuned_enc')
```

```
In [113]: learn.save_encoder('fine_tuned_enc')
```

```
In [94]: path = untar_data(URLs.YELP_REVIEWS_POLARITY)
```

```
In [95]: data_clas = (TextList.from_df(df=df, path=path, cols=1, vocab=data_lm.vocab)
               .split_from_df(col=2)
               .label_from_df(cols=0)
               .databunch(bs=32))
```

```
In [96]: data_clas.show_batch()
```

	text	target
<p>xxbos a hands down must see attraction . xxmaj the first time i saw it in 2006 they played xxmaj my xxmaj heart xxmaj will xxmaj go xxmaj on by xxmaj celine xxmaj dion . xxmaj it was so beautiful i almost cried ! xxmaj the strip side offers the best views in my opinion . \n \n xxmaj the fountains are set in a 9 acre xxunk lake</p>		2
<p>xxbos i see that i have been beaten to the punch by 3 fellow ramen aficionados already , but here is my take . \n \n i would have given 5 starts if this were 4 years ago , but as the competition heats up for local ramen - ya 's , i have to be a bit more stingy with my ratings . \n \n xxmaj before</p>		2
<p>xxbos xxmaj please read this carefully . \n xxmaj if you have elderly parents or a disabled person in you family , do not order service from xxup direct xxup alert !! \n \n xxmaj terrible quality of device as well as customer service ! \n xxmaj since 22nd of xxmaj may , till now xxmaj june xxunk , their customer service has not returned my</p>		1
<p>xxbos xxmaj please save yourself . xxmaj do n't do business with these people . \n \n xxmaj if i could give a negative star i would . i write this review to warn others and to try and spare you the headaches that we have experienced . xxmaj you are welcome to contact me with any questions and i am happy to share any documentation you would care</p>		1
<p>xxbos xxmaj wow yelp reviewers , literally the worst hotel i 've ever stayed at on the xxmaj strip . xxmaj the service was horrific . i should have know by the ridiculous hold times when i called prior to my arrival that the service of the hotel was not up to par . xxmaj this hotel is nothing like the xxmaj venetian . xxmaj do n't be fooled .</p>		1

```
In [110]: learn = text_classifier_learner(data_clas, AWD_LSTM, drop_mult=0.5)
learn.load_encoder('fine_tuned_enc')
```

```
-----
----
FileNotFoundError                                Traceback (most recent call 1
ast)
<ipython-input-110-58fb872cb4e2> in <module>
      1 learn = text_classifier_learner(data_clas, AWD_LSTM, drop_mult=
0.5)
----> 2 learn.load_encoder('fine_tuned_enc')

/opt/anaconda3/lib/python3.7/site-packages/fastai/text/learner.py in lo
ad_encoder(self, name, device)
      67         if device is None: device = self.data.device
      68         if hasattr(encoder, 'module'): encoder = encoder.module
--> 69         encoder.load_state_dict(torch.load(self.path/self.model
_dir/f'{name}.pth', map_location=device))
      70         self.freeze()
      71         return self

/opt/anaconda3/lib/python3.7/site-packages/torch/serialization.py in lo
ad(f, map_location, pickle_module, **pickle_load_args)
     420         elif (sys.version_info[0] == 3 and isinstance(f, pathlib.Pa
th)):
     421             new_fd = True
--> 422             f = f.open('rb')
     423         try:
     424             if sys.version_info >= (3, 0) and 'encoding' not in pic
kle_load_args.keys():

/opt/anaconda3/lib/python3.7/pathlib.py in open(self, mode, buffering,
encoding, errors, newline)
     1191             self._raise_closed()
     1192             return io.open(self, mode, buffering, encoding, errors,
newline,
-> 1193                             opener=self._opener)
     1194
     1195         def read_bytes(self):

/opt/anaconda3/lib/python3.7/pathlib.py in _opener(self, name, flags, m
ode)
     1044         def _opener(self, name, flags, mode=0o666):
     1045             # A stub for the opener argument to built-in open()
-> 1046             return self._accessor.open(self, flags, mode)
     1047
     1048         def _raw_open(self, flags, mode=0o777):

FileNotFoundError: [Errno 2] No such file or directory: '/home/jupyter
r/.fastai/data/yelp_review_polarity_csv/models/fine_tuned_enc.pth'
```

```
In [109]: ! ~/.fastai/models
```

```
/bin/sh: 1: /home/jupyter/.fastai/models: Permission denied
```

```
In [ ]: learn.lr_find()
```

```
In [ ]: learn.recorder.plot()
```