

```
In [1]: from fastai.text import *
```

```
/home/paperspace/anaconda3/envs/fastai/lib/python3.6/site-packages/sklearn/en  
semble/weight_boosting.py:29: DeprecationWarning: numpy.core.umath_tests is a  
n internal NumPy module and should not be imported. It will be removed in a f  
uture NumPy release.  
    from numpy.core.umath_tests import inner1d
```

```
In [ ]:
```

```
In [2]: PATH = Path("/home/paperspace/fastai/courses/dl2/data")
```

```
In [3]: data = pd.read_csv(PATH/"gop_sentiments.csv", encoding= "latin1")
```

```
In [4]: data = data[["sentiment", "text"]]
```

```
In [5]: data.columns = ['labels', 'text']  
data.head(200)
```

Out[5]:

	labels	text
0	Neutral	RT @NancyLeeGrah: How did everyone feel about...
1	Positive	RT @ScottWalker: Didn't catch the full #GOPdeb...
2	Neutral	RT @TJMShow: No mention of Tamir Rice and the ...
3	Positive	RT @RobGeorge: That Carly Fiorina is trending ...
4	Positive	RT @DanScavino: #GOPDebate w/ @realDonaldTrump...
5	Positive	RT @GregAbbott_TX: @TedCruz: "On my first day ...
6	Negative	RT @warriorwoman91: I liked her and was happy ...
7	Neutral	Going on #MSNBC Live with @ThomasARoberts arou...
8	Negative	Deer in the headlights RT @lizzwinstead: Ben C...
9	Negative	RT @NancyOsborne180: Last night's debate prove...
10	Negative	@JGreenDC @realDonaldTrump In all fairness #Bi...
11	Positive	RT @WayneDupreeShow: Just woke up to tweet thi...
12	Negative	Me reading my family's comments about how grea...
13	Neutral	RT @ArcticFox2016: RT @AllenWestRepub "Dear @J...
14	Positive	RT @pattonoswalt: I loved Scott Walker as Mark...
15	Negative	Hey @ChrisChristie exploiting the tragedy of 9...
16	Negative	RT @CarolCNN: #DonaldTrump under fire for comm...
17	Negative	RT @johncardillo: Guess who had most speaking ...
18	Negative	reason comment is funny 'in case you're ignora...
19	Negative	RT @PamelaGeller: Huckabee: Paying for transge...
20	Positive	RT @ChuckNellis: Cruz has class & truth, t...
21	Negative	RT @mchamric: RT @TeaTraitors: #GOPDebate w...
22	Negative	RT @erinmallorylong: No *I* hate Planned Paren...
23	Neutral	RT @thekevinryder: #GOPDebate (Vine by @dabull...
24	Negative	RT @MrPooni: Fox News trying to convince us yo...
25	Negative	#GOPDebate rankings: worst to be performance -...
26	Negative	RT @TheBaxterBean: Scott Walker's Abortion Ban...
27	Negative	RT @feministabulous: It's not a competition, b...
28	Negative	RT @mch7576: RT @TeaTraitors: #GOPDebate wa...
29	Negative	RT @HillaryClinton: Watch the #GOPdebate? Bet ...
...	...	...
170	Negative	RT @JohnKasich: Hey @realDonaldTrump, if you'r...
171	Negative	@megynkelly @BretBaier #ChrisWallace, #America...
172	Negative	@GovMikeHuckabee: Luntz: "As the lines go u...
173	Negative	RT @oftenimprudent: I wish we could teach teen...

	labels	text
174	Negative	RT @AmyMek: Conservatives are a people without...
175	Negative	RT @BHowl_: How come there's no talk about cli...
176	Neutral	RT @travisconsidine: .@GovernorPerry #GOPDebat...
177	Negative	RT @ChrisJZullo: Last night they could have ha...
178	Positive	#GOPDebate 2016 #teamdrbencarson @RealBenCarso...
179	Neutral	RT @SooperMexican: Ted Cruz doesn't appreciate...
180	Negative	RT @BenJamminWalker: Under no circumstances sh...
181	Positive	RT @DanScavino: #GOPDebate Winner!\nDrudge 50%...
182	Negative	RT @JoePrich: .@megynkelly In 2012 you chided...
183	Negative	#GOPDebate @FoxNews @realDonaldTrump in1996 y...
184	Negative	RT @renomarky: â never watch #KellyFile agai...
185	Positive	IMHO #GOPDebate \nwinners #MarcoRubio #Kasich ...
186	Positive	RT @JohnKasich: SMALLER GOVT: Ohio government ...
187	Neutral	RT @zelleimani: Twitter Users Takeover #GOPDe...
188	Negative	RT @JebBush: "Challenging the teacher's unions...
189	Neutral	RT @billyeichner: Thank you all for watching t...
190	Negative	@theblaze The REAL war on women!@realDonaldTrump...
191	Neutral	RT @GovernorPerry: .@GovernorPerry's "On Fire"...
192	Neutral	RT @blowticious: Pro-life Republicans be like ...
193	Neutral	RT @GovernorPerry: ICYMI: @GovernorPerry on Bo...
194	Negative	RT @AllThingsFlynn: This Trump vs. Fox News co...
195	Positive	RT @CarlyFiorina: I know how extraordinary thi...
196	Negative	RT @LAXnOREOS: Meanwhile, in the White House.....
197	Positive	#rubio still my preference but i can see point...
198	Positive	Did @realDonaldTrump ever tell @megynkelly to ...
199	Positive	RT @TheTexasPhoenix: He was on fire! .@Governor...

200 rows × 2 columns

```
In [6]: CLAS_PATH = Path("/home/paperspace/fastai/courses/dl2/data/gop_class")
CLAS_PATH.mkdir(exist_ok=True)
LM_PATH = Path("/home/paperspace/fastai/courses/dl2/data/gop_lm")
LM_PATH.mkdir(exist_ok=True)
```

```
In [7]: cleanup_nums = {"labels":      {"Negative": 0, "Neutral": 1, "Positive":2}
      }
      data.replace(cleanup_nums, inplace=True)
      data.head
```

Out[7]: <bound method NDFrame.head of labels

```
text
0      1 RT @NancyLeeGrahm: How did everyone feel about...
1      2 RT @ScottWalker: Didn't catch the full #GOPdeb...
2      1 RT @TJMShow: No mention of Tamir Rice and the ...
3      2 RT @RobGeorge: That Carly Fiorina is trending ...
4      2 RT @DanScavino: #GOPDebate w/ @realDonaldTrump...
5      2 RT @GregAbbott_TX: @TedCruz: "On my first day ...
6      0 RT @warriorwoman91: I liked her and was happy ...
7      1 Going on #MSNBC Live with @ThomasARoberts arou...
8      0 Deer in the headlights RT @lizzwinstead: Ben C...
9      0 RT @NancyOsborne180: Last night's debate prove...
10     0 @JGreenDC @realDonaldTrump In all fairness #Bi...
11     2 RT @WayneDupreeShow: Just woke up to tweet thi...
12     0 Me reading my family's comments about how grea...
13     1 RT @ArcticFox2016: RT @AllenWestRepub "Dear @J...
14     2 RT @pattonoswalt: I loved Scott Walker as Mark...
15     0 Hey @ChrisChristie exploiting the tragedy of 9...
16     0 RT @CarolCNN: #DonaldTrump under fire for comm...
17     0 RT @johncardillo: Guess who had most speaking ...
18     0 reason comment is funny 'in case you're ignora...
19     0 RT @PamelaGeller: Huckabee: Paying for transge...
20     2 RT @ChuckNellis: Cruz has class & truth, t...
21     0 RT @mchamric: RT @TeaTraitors: #GOPDebate w...
22     0 RT @erinmallorylong: No *I* hate Planned Paren...
23     1 RT @thekevinryder: #GOPDebate (Vine by @dabull...
24     0 RT @MrPooni: Fox News trying to convince us yo...
25     0 #GOPDebate rankings: worst to be performance -...
26     0 RT @TheBaxterBean: Scott Walker's Abortion Ban...
27     0 RT @feministabulous: It's not a competition, b...
28     0 RT @mch7576: RT @TeaTraitors: #GOPDebate wa...
29     0 RT @HillaryClinton: Watch the #GOPdebate? Bet ...
...
13841  0 RT @EthanObama: Did Trump just admit he gives ...
13842  1 Trump and most of the World! Just look at the ...
13843  0 @JessicaValenti they all need coif reform. #GO...
13844  0 Wow, Rubio's ear-to-head ratio is all messed u...
13845  0 RT @the818: Let's play "how fast can we distan...
13846  1 Me watching the GOP Debates. #gopdebates https...
13847  2 RT @mjtbaum: GOD is making an appearance at th...
13848  0 RT @PuestoLoco: .@NewDay Cancel the primaries....
13849  2 RT @kaylasmith4791: Really enjoyed everything ...
13850  0 RT @RWSurferGirl: The candidates don't have to...
13851  0 #GOPDebate in a nutshell via Vote True Blue 20...
13852  0 RT @erinmallorylong: No *I* hate Planned Paren...
13853  0 RT @BrendanKKirby: If @JohnKasich hasn't wrapp...
13854  0 RT @monaeltahawy: I'll tell you the one good t...
13855  0 while pro-life nonsense @ the #GOPDebate was n...
13856  0 GOP respects life....just not black ones. #GOP...
13857  0 This is why I don't watch Fox News, they're al...
13858  2 @marcorubio came out of the gate like a true l...
13859  2 Best line of #GOPDebate was "Immigration witho...
13860  0 People who say they are #prolife are usually a...
13861  0 RT @RWSurferGirl: Why should @realDonaldTrump ...
13862  2 RT @RWSurferGirl: Trump has got it right, nobo...
13863  0 So trans soldiers can die for you Huckabee but...
13864  0 RT @RWSurferGirl: Is it just me or does anyone...
13865  0 RT @RWSurferGirl: Fox is cherry picking the ca...
13866  0 RT @cappy_yarbrough: Love to see men who will ...
13867  2 RT @georgehenryw: Who thought Huckabee exceede...
13868  2 RT @Lrihendry: #TedCruz As President, I will a...
13869  0 RT @JRehling: #GOPDebate Donald Trump says tha...
13870  2 RT @Lrihendry: #TedCruz headed into the Presid...
```

```
[13871 rows x 2 columns]>
```

```
In [8]: split_index = int(data.shape[0] * .8)
train_df, test_df = np.split(data, [split_index], axis=0)
```

```
In [9]: print(train_df.shape)
print(test_df.shape)
print(train_df.head(10))
```

```
(11096, 2)
(2775, 2)
   labels text
0      1 RT @NancyLeeGrahm: How did everyone feel about...
1      2 RT @ScottWalker: Didn't catch the full #GOPdeb...
2      1 RT @TJMShow: No mention of Tamir Rice and the ...
3      2 RT @RobGeorge: That Carly Fiorina is trending ...
4      2 RT @DanScavino: #GOPDebate w/ @realDonaldTrump...
5      2 RT @GregAbbott_TX: @TedCruz: "On my first day ...
6      0 RT @warriorwoman91: I liked her and was happy ...
7      1 Going on #MSNBC Live with @ThomasARoberts arou...
8      0 Deer in the headlights RT @lizzwinstead: Ben C...
9      0 RT @NancyOsborne180: Last night's debate prove...
```

```
In [10]: ## write the clasification repo
train_df.to_csv(CLAS_PATH/"train.csv", header=False, index=False,encoding="utf-8")
test_df.to_csv(CLAS_PATH/"test.csv", header=False, index=False,encoding="utf-8")

train_texts, test_texts = sklearn.model_selection.train_test_split(
    np.concatenate([train_df["text"],test_df["text"]]), test_size = 0.1)

colnames = ["labels","text"]
df_train = pd.DataFrame({"text": train_texts, "labels": [0]*len(train_texts)},
    columns=colnames)
df_test = pd.DataFrame({"text": test_texts, "labels": [0]*len(test_texts)}, co
lums=colnames)

## write to lm repo
df_train.to_csv(LM_PATH/"train.csv", header=False, index=False, columns=colnam
es, encoding="utf-8")
df_test.to_csv(LM_PATH/"test.csv", header=False, index=False, columns=colnames
, encoding="utf-8")

#train_df.to_csv(LM_PATH/'train.csv', header=False, index=False, encoding='utf-8')
#test_df.to_csv(LM_PATH/'test.csv', header=False, index=False, encoding='utf-8')
```

```
In [11]: print(df_train.shape)
print(df_test.shape)
```

```
(12483, 2)
(1388, 2)
```

```
In [12]: CLASSES = ["Neutral","Positive","Negative"]
(CLAS_PATH/"classes.txt").open("w", encoding = "utf-8").writelines(f"{cls}\n"
for cls in CLASSES)
```

```
In [13]: #print(train_df.head(20))
```

```
In [14]: # chunksize for pandas so it doesn't run into any memory limits
chunksize=2400
```

```
In [15]: ## functions pulled from the fast.ai notebook for text tokenization

rel = re.compile(r' +')

def fixup(x):
    x = x.replace('#39;', '"').replace('amp;', '&').replace('#146;', '"').replace(
        'nbsp;', ' ').replace('#36;', '$').replace('\\n', '\n').replace('quo
t;', '"').replace(
        '<br />', '\n').replace('\\\"', '\"').replace('<unk>', 'u_n').replace('
@.@ ', '.').replace(
        '@-@ ', '-').replace('\\', '\\ ')
    return rel.sub(' ', html.unescape(x))
```

```
In [16]: def get_texts(df, n_lbls=1):
    labels = df.iloc[:, range(n_lbls)].values.astype(np.int64)
    texts = f'\n{BOS} {FLD} 1 ' + df[n_lbls].astype(str)
    for i in range(n_lbls+1, len(df.columns)): texts += f' {FLD} {i-n_lbls} '
+ df[i].astype(str)
    texts = list(texts.apply(fixup).values)

    tok = Tokenizer().proc_all_mp(partition_by_cores(texts))
    return tok, list(labels)
```

```
In [17]: def get_all(df, n_lbls):
    tok, labels = [], []
    for i, r in enumerate(df):
        print(i)
        tok_, labels_ = get_texts(r, n_lbls)
        tok += tok_
        labels += labels_
    return tok, labels
```

```
In [18]: df_train = pd.read_csv(LM_PATH/"train.csv", header=None, chunksize=chunksize)
df_test = pd.read_csv(LM_PATH/"test.csv", header=None, chunksize=chunksize)
```



```
In [19]: next(iter(df_train))
```

Out[19]:

	0	1
0	0	tillerylakelady: RT realDonaldTrump: "stinger_...
1	0	Not just Chris!-RT @RWSurferGirl: Is it just m...
2	0	RT @RWSurferGirl: Tonight we learned that @Gov...
3	0	RT @aqv21: How #Hillary Looked When Watching #...
4	0	RT @ericstonestreet: Trump has Cam hands. #GOP...
5	0	RT @Apathycase: So, Trump buys politicians, us...
6	0	RT @RWSurferGirl: Fox is cherry picking the ca...
7	0	Are they trying to knock out Paul and Trump by...
8	0	RT @DykstraDame: Candidates went after @Hillar...
9	0	There's visual evidence of @DaisyJDog's though...
10	0	RT @Debber66: Who do you think won the #GOPDeb...
11	0	RT @RWSurferGirl: Why doesn't Chris Wallace as...
12	0	RT @JimSterling: Donald Trump. Literally a fiv...
13	0	@FoxNews Please replace @megynkelly. Future de...
14	0	RT @curlyheadRED: Me watching the... #GOPDebat...
15	0	RT @sallykohn: The head of the Republican Part...
16	0	Circus Roundup #GOPDebates @nytimes How the Ca...
17	0	My family doesn't like John Kasich and they di...
18	0	Why did they have britney spears moderating th...
19	0	FOX #GOPDebate drew 10 million viewers. No Dem...
20	0	RT @jjauthor: The policies of Barack Obama hav...
21	0	Last night #Rand Paul show us that he will fac...
22	0	RT @CarolHello1: #GOPDebate TakeAway: https://...
23	0	YES HE DID!\n@RMConservative: #GOPDebate: Did ...
24	0	RT @sydneyvl24: Cautiously optimistic is how I...
25	0	RT @RWSurferGirl: FOX News won't admit who the...
26	0	RT @MarkDavis: #Perry, #Fiorina clips on #Iran...
27	0	Biggest takeaway from #GOPDebate, Rs love the ...
28	0	RT @ShawnDrurySC: Jeb: My father was...OH NEVE...
29	0	RT @larryelder: Trump should have said, "Megyn...
...	...	...
2370	0	I've seen more substance in a Michael Bay film...
2371	0	RT @drskyskull: One white woman killed by an i...
2372	0	Wait, did I watch a presidential debate last n...
2373	0	RT @NotBillWalton: The Straight Outta Compton ...

	0	1
2374	0	RT @ericstonestreet: Trump has Cam hands. #GOP...
2375	0	@FoxNews intentionally tried to shut @RandPaul...
2376	0	RT @jasper_deezy: Mike Huckabee quoting Limp B...
2377	0	RT @SupermanHotMale: Dear Gov Walker, you have...
2378	0	I'd rather have a boxing match & let the w...
2379	0	You guys, separation of church and state is al...
2380	0	ð©RT @Rik_Flair: X__X RT @kvxrdashian: when...
2381	0	RT @RWSurferGirl: I need to order another gree...
2382	0	The #GOPdebate summed up. <a href="http://t.co/fkYSDRwzS1">http://t.co/fkYSDRwzS1</a>
2383	0	proud of the @GOP field after #GOPDebate Took ...
2384	0	RT @chaunceydevega: Kasich is not enough of a ...
2385	0	RT @WendyWinston4: Florida graduation rate was...
2386	0	RT @RaisedByCulture: Unborn lives are more imp...
2387	0	Veterans for vice prez. #gopdebates
2388	0	There was a lot of 'God' going on last night. ...
2389	0	RT @RWSurferGirl: Trump has got it right, nobo...
2390	0	RT @TheBaxterBean: GOP Denied \$50B Infrastruct...
2391	0	@rhondalevaldo How is @UNITY_JFD working on po...
2392	0	RT @larryelder: Trump should have said, "Megyn...
2393	0	RT @EthanObama: Did Trump just admit he gives ...
2394	0	That @ScottWalker's zinger about Hillary's ema...
2395	0	RT @RWSurferGirl: I am wondering what Fox is u...
2396	0	RT @ScottLinnen: You screwed Atlantic City mor...
2397	0	Rubio, Cruz and Carson were the top dawgs i...
2398	0	RT @BenMank77: Well done, Fox...great question...
2399	0	@davidaxelrod @marcorubio Its outrageous to no...

2400 rows × 2 columns

```
In [20]: BOS = "xbos"
        FLD = "xfld"
```

```
In [287]: tok_train, train_labels = get_all(df_train,1)
        tok_test, test_labels = get_all(df_test,1)
```

```
0
1
2
3
4
0
```

```
In [22]: (LM_PATH/"tmp").mkdir(exist_ok = True)
tmp = Path("/home/paperspace/fastai/courses/dl2/data/gop_lm/tmp")
```

```
In [293]: np.save(tmp/"tok_trn.npy", tok_train)
np.save(tmp/"tok_test.npy", tok_test)
```

```
In [23]: tok_train = np.load(tmp/"tok_trn.npy")
tok_test = np.load(tmp/"tok_test.npy")
```

```
In [26]: len(tok_train) # len = 10083
len(tok_test)
```

```
Out[26]: 1388
```

```
In [27]: freq = Counter(p for o in tok_train for p in o)
freq.most_common(25)
```

```
Out[27]: [(' ', 15717),
          ('\n', 10624),
          ('1', 10186),
          ('xbos', 10083),
          ('xfld', 10083),
          (':', 9800),
          ('/', 8488),
          ('.', 7382),
          ('gopdebate', 6515),
          ('the', 6447),
          ('rt', 5725),
          ('gopdebates', 3794),
          (',', 3702),
          ('to', 3507),
          ('t_up', 3466),
          ('t.co', 2638),
          ('is', 2485),
          ('"', 2450),
          ('of', 2439),
          ('a', 2425),
          ('i', 2399),
          ('http', 2194),
          ('and', 2114),
          ('|', 2097),
          ('?', 2018)]
```

```
In [28]: max_vocab = 60000
min_freq = 2
```

```
In [399]: #itos = [o for o,c in freq.most_common(max_vocab) if c > min_freq]
itos = [o for o,c in freq.most_common(max_vocab)]

len(itos)
```

```
Out[399]: 16386
```

```
In [400]: itos.insert(0, "_pad_")
itos.insert(1, "_unk_")
```

```
In [401]: stoi = collections.defaultdict(lambda:0, {o:c for c,o in enumerate(itos)})  
len(stoi)
```

```
Out[401]: 16388
```

```
In [402]: trn_lm = np.array([[stoi[o] for o in p] for p in tok_train])  
test_lm = np.array([[stoi[o] for o in p] for p in tok_test])  
#"".join(str(o) for o in trn_lm[1])  
len(trn_lm)
```

```
Out[402]: 10083
```

```
In [403]: np.save(tmp/"trn_ids.npy", trn_lm)  
np.save(tmp/"test_ids.npy", test_lm)
```

```
In [404]: pickle.dump(itos, open(tmp/"itos.pkl", "wb"))
```

```
In [29]: trn_lm = np.load(tmp/"trn_ids.npy")  
test_lm = np.load(tmp/"test_ids.npy")  
itos = pickle.load(open(tmp/"itos.pkl", "rb"))
```

```
In [30]: vs=len(itos)  
vs,len(trn_lm)
```

```
Out[30]: (16388, 10083)
```

```
In [36]: em_sz, nh, nl = 400, 1150, 3
```

```
In [31]: PRE_PATH = Path("/home/paperspace/data/twitter/lm/models")  
PRE_LM_PATH = Path(PRE_PATH/"lm_tt.h5")
```

```
In [32]: wgts = torch.load(PRE_LM_PATH, map_location = lambda storage, loc: storage)
```

```
In [33]: enc_wgts = to_np(wgts["0.encoder.weight"])  
row_m = enc_wgts.mean(0)
```

```
In [34]: itos2 = pickle.load((PRE_PATH/"itos_tt.pkl").open("rb"))  
stoi2 = collections.defaultdict(lambda:-1, {v:k for k,v in enumerate(itos2)})
```

```
In [37]: new_w = np.zeros((vs, em_sz), dtype=np.float32)  
for i,w in enumerate(itos):  
    r = stoi2[w]  
    new_w[i] = enc_wgts[r] if r>=0 else row_m
```

```
In [38]: wgts['0.encoder.weight'] = T(new_w)  
wgts['0.encoder_with_dropout.embed.weight'] = T(np.copy(new_w))  
wgts['1.decoder.weight'] = T(np.copy(new_w))
```

```
In [408]: # PRE_PATH = Path("/home/paperspace/fastai/courses/dl2/data/aclImdb/models/wt1  
03")  
# PRE_LM_PATH = Path(PRE_PATH/"fwd_wt103.h5")
```

```
In [409]: # wgts = torch.load(PRE_LM_PATH, map_location = lambda storage, loc: storage)
```

```
In [410]: # enc_wgts = to_np(wgts["0.encoder.weight"])  
# row_m = enc_wgts.mean(0)
```

```
In [411]: # itos2 = pickle.load((PRE_PATH/"itos_wt103.pkl").open("rb"))
# stoi2 = collections.defaultdict(lambda:-1, {v:k for k,v in enumerate(itos
2)})
```

```
In [412]: # new_w = np.zeros((vs, em_sz), dtype=np.float32)
# for i,w in enumerate(itos):
#     r = stoi2[w]
#     new_w[i] = enc_wgts[r] if r>=0 else row_m
```

```
In [413]: # wgts['0.encoder.weight'] = T(new_w)
# wgts['0.encoder_with_dropout.embed.weight'] = T(np.copy(new_w))
# wgts['1.decoder.weight'] = T(np.copy(new_w))
```

## Language model

```
In [39]: wd=1e-7
bptt=70
bs=52
opt_fn = partial(optim.Adam, betas=(0.8, 0.99))
```

```
In [40]: trn_dl = LanguageModelLoader(np.concatenate(trn_lm), bs, bptt)
test_dl = LanguageModelLoader(np.concatenate(test_lm), bs, bptt)
md = LanguageModelData(PATH, 1, vs, trn_dl, test_dl, bs=bs, bptt=bptt)
```

```
In [41]: drops = np.array([0.25, 0.1, 0.2, 0.02, 0.15])*0.7
```

```
In [42]: learner= md.get_model(opt_fn, em_sz, nh, nl,
    dropouti=drops[0], dropout=drops[1], wdrop=drops[2], dropoute=drops[3], dr
    opouth=drops[4])

learner.metrics = [accuracy]
learner.freeze_to(-1)
```

```
In [43]: learner.model.load_state_dict(wgts)
```

```
In [44]: lr=1e-3
lrs = lr
```

```
In [45]: learner.fit(lrs/2, 1, wds=wd, use_clr=(32,2), cycle_len=1)
```

epoch	trn_loss	val_loss	accuracy
0	6.355974	5.82235	0.277013

```
Out[45]: [array([5.82235]), 0.2770133680767483]
```

```
In [46]: learner.save('lm_last_ft_tt')
```

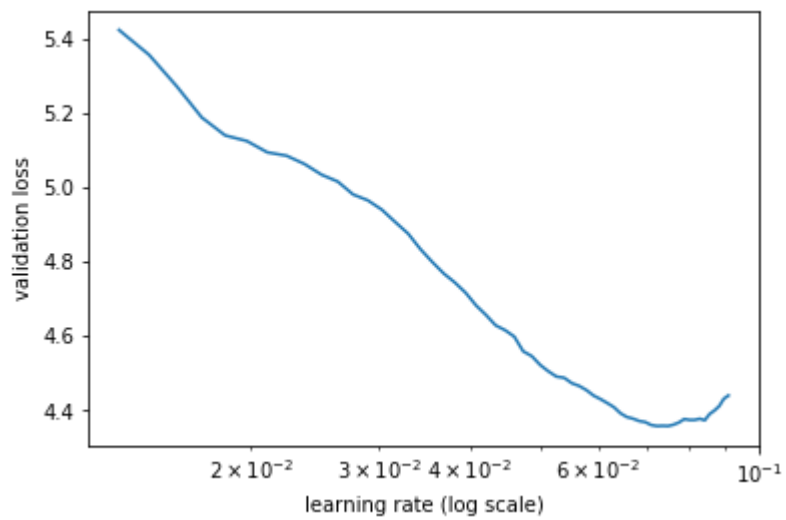
```
In [56]: learner.load('lm_last_ft_tt')
```

```
In [57]: learner.unfreeze()
```

```
In [62]: learner.lr_find(start_lr=lrs, end_lr=lrs*100, linear=True)
```

epoch	trn_loss	val_loss	accuracy
0	4.501976	4.750352	0.287467

```
In [63]: learner.sched.plot()
```



```
In [66]: learner.fit(1e-2, 1, wds=wd, use_clr=(20,10), cycle_len=10)
```



epoch	trn_loss	val_loss	accuracy
0	3.873435	3.423033	0.47803
1	3.301769	2.973426	0.525227
2	2.926314	2.986995	0.536954
3	2.667494	2.9305	0.548462
4	2.437687	2.92168	0.560738
5	2.223275	2.98898	0.566556

4%|█ | 3/76 [00:00<00:09, 7.70it/s, loss=2.22]

```

-----
KeyboardInterrupt                                Traceback (most recent call last)
<ipython-input-66-e1487cdff194> in <module>
----> 1 learner.fit(1e-2, 1, wds=wd, use_clr=(20,10), cycle_len=10)

~/fastai/courses/dl2/fastai/text.py in fit(self, *args, **kwargs)
    209
    210     def _get_crit(self, data): return F.cross_entropy
--> 211     def fit(self, *args, **kwargs): return super().fit(*args, **kwarg
s, seq_first=True)
    212
    213     def save_encoder(self, name): save_model(self.model[0], self.get_
model_path(name))

~/fastai/courses/dl2/fastai/learner.py in fit(self, lrs, n_cycle, wds, **kwar
gs)
    300         self.sched = None
    301         layer_opt = self.get_layer_opt(lrs, wds)
--> 302         return self.fit_gen(self.model, self.data, layer_opt, n_cycle
, **kwargs)
    303
    304     def warm_up(self, lr, wds=None):

~/fastai/courses/dl2/fastai/learner.py in fit_gen(self, model, data, layer_op
t, n_cycle, cycle_len, cycle_mult, cycle_save_name, best_save_name, use_clr,
use_clr_beta, metrics, callbacks, use_wd_sched, norm_wds, wds_sched_mult, us
e_swa, swa_start, swa_eval_freq, **kwargs)
    247         metrics=metrics, callbacks=callbacks, reg_fn=self.reg_fn,
clip=self.clip, fp16=self.fp16,
    248         swa_model=self.swa_model if use_swa else None, swa_start=
swa_start,
--> 249         swa_eval_freq=swa_eval_freq, **kwargs)
    250
    251     def get_layer_groups(self): return self.models.get_layer_groups()

~/fastai/courses/dl2/fastai/model.py in fit(model, data, n_epochs, opt, crit,
metrics, callbacks, stepper, swa_model, swa_start, swa_eval_freq, visualize,
**kwargs)
    139         batch_num += 1
    140         for cb in callbacks: cb.on_batch_begin()
--> 141         loss = model_stepper.step(V(x),V(y), epoch)
    142         avg_loss = avg_loss * avg_mom + loss * (1-avg_mom)
    143         debias_loss = avg_loss / (1 - avg_mom**batch_num)

~/fastai/courses/dl2/fastai/model.py in step(self, xs, y, epoch)
    55         if self.loss_scale != 1: assert(self.fp16); loss = loss*self.
loss_scale
    56         if self.reg_fn: loss = self.reg_fn(output, xtra, raw_loss)
---> 57         loss.backward()
    58         if self.fp16: update_fp32_grads(self.fp32_params, self.m)
    59         if self.loss_scale != 1:

~/anaconda3/envs/fastai/lib/python3.6/site-packages/torch/autograd/variable.p
y in backward(self, gradient, retain_graph, create_graph, retain_variables)
    165             Variable.
    166             """
--> 167         torch.autograd.backward(self, gradient, retain_graph, create_
graph, retain_variables)
    168
    169     def register_hook(self, hook):

~/anaconda3/envs/fastai/lib/python3.6/site-packages/torch/autograd/__init__.p
y in backward(variables, grad_variables, retain_graph, create_graph, retain_v

```

```

variables)
    97
    98     Variable._execution_engine.run_backward(
---> 99         variables, grad_variables, retain_graph)
    100
    101

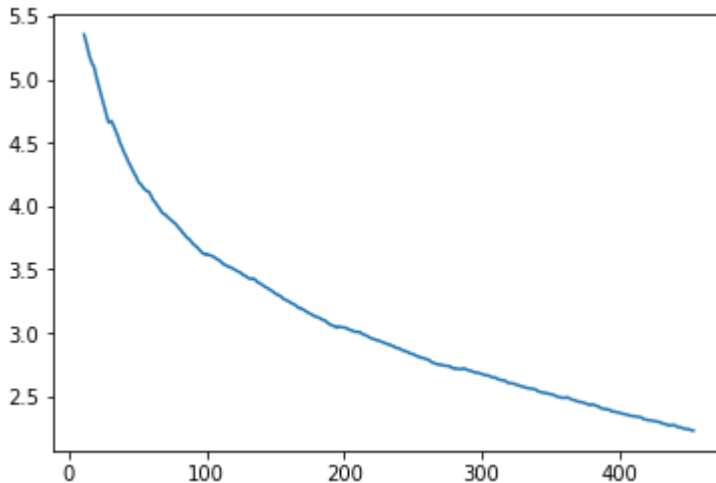
```

KeyboardInterrupt:

```
In [79]: learner.save('lml_tt')
```

```
In [80]: learner.save_encoder('lml_enc_tt')
```

```
In [67]: learner.sched.plot_loss()
```



```
In [68]: train_df = pd.read_csv(CLAS_PATH/'train.csv', header=None, chunksize=chunksize)
test_df = pd.read_csv(CLAS_PATH/'test.csv', header=None, chunksize=chunksize)
```

```
In [ ]:
```

```
In [431]: tok_trn, trn_labels = get_all(train_df, 1)
tok_val, val_labels = get_all(test_df, 1)
```

```

0
1
2
3
4
0
1

```

```
In [432]: (CLAS_PATH/'tmp').mkdir(exist_ok=True)

np.save(CLAS_PATH/'tmp'/'tok_trn.npy', tok_trn)
np.save(CLAS_PATH/'tmp'/'tok_val.npy', tok_val)

np.save(CLAS_PATH/'tmp'/'trn_labels.npy', trn_labels)
np.save(CLAS_PATH/'tmp'/'val_labels.npy', val_labels)
```

```
In [69]: tok_trn = np.load(CLAS_PATH/'tmp'/'tok_trn.npy')
tok_val = np.load(CLAS_PATH/'tmp'/'tok_val.npy')
```

```
In [70]: itos = pickle.load((LM_PATH/'tmp'/'itos.pkl').open('rb'))
stoi = collections.defaultdict(lambda:0, {v:k for k,v in enumerate(itos)})
len(itos)
```

```
Out[70]: 16388
```

```
In [71]: trn_clas = np.array([[stoi[o] for o in p] for p in tok_trn])
val_clas = np.array([[stoi[o] for o in p] for p in tok_val])
```

```
In [72]: np.save(CLAS_PATH/'tmp'/'trn_ids.npy', trn_clas)
np.save(CLAS_PATH/'tmp'/'val_ids.npy', val_clas)
trn_clas = np.load(CLAS_PATH/'tmp'/'trn_ids.npy')
val_clas = np.load(CLAS_PATH/'tmp'/'val_ids.npy')
```

## CLssification

```
In [97]: trn_labels = np.squeeze(np.load(CLAS_PATH/'tmp'/'trn_labels.npy'))
val_labels = np.squeeze(np.load(CLAS_PATH/'tmp'/'val_labels.npy'))
```

```
In [98]: bptt,em_sz,nh,nl = 70,400,1150,3
vs = len(itos)
opt_fn = partial(optim.Adam, betas=(0.8, 0.99))
bs = 48
```

```
In [99]: min_lbl = trn_labels.min()
trn_labels -= min_lbl
val_labels -= min_lbl
c=int(trn_labels.max())+1
```

```
In [100]: trn_ds = TextDataset(trn_clas, trn_labels)
val_ds = TextDataset(val_clas, val_labels)
trn_samp = SortishSampler(trn_clas, key=lambda x: len(trn_clas[x]), bs=bs//2)
val_samp = SortSampler(val_clas, key=lambda x: len(val_clas[x]))
trn_dl = DataLoader(trn_ds, bs//2, transpose=True, num_workers=1, pad_idx=1, sampler=trn_samp)
val_dl = DataLoader(val_ds, bs, transpose=True, num_workers=1, pad_idx=1, sampler=val_samp)
md = ModelData(PATH, trn_dl, val_dl)
```

```
In [101]: dps = np.array([0.4,0.5,0.05,0.3,0.4])*0.5
```

```
In [102]: m = get_rnn_classifier(bptt, 20*70, c, vs, emb_sz=em_sz, n_hid=nh, n_layers=nl
, pad_token=1,
          layers=[em_sz*3, 50, c], drops=[dps[4], 0.1],
          dropouti=dps[0], wdrop=dps[1], dropoute=dps[2], dropouth=dps[3])
```

```
In [103]: opt_fn = partial(optim.Adam, betas=(0.7, 0.99))
```

```
In [167]: learn = RNN_Learner(md, TextModel(to_gpu(m)), opt_fn=opt_fn)
learn.reg_fn = partial(seq2seq_reg, alpha=2, beta=1)
learn.clip=.25
learn.metrics = [accuracy]
```

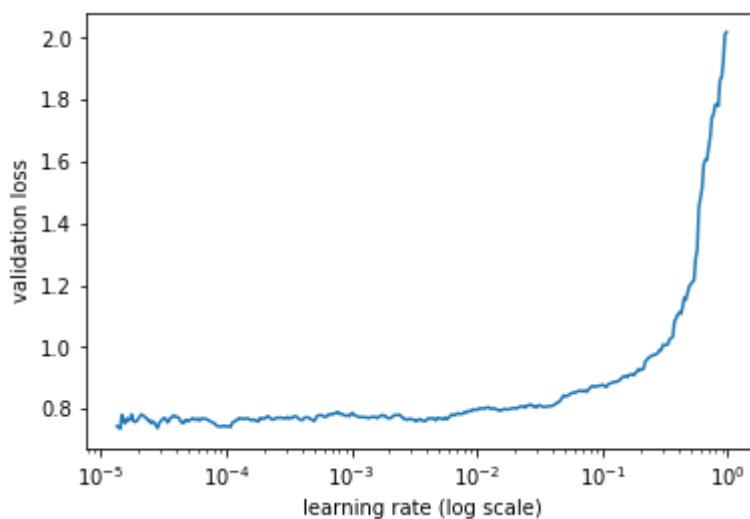
```
In [168]: lr=3e-3
          lrm = 2.6
          lrs = np.array([lr/(lrm**4), lr/(lrm**3), lr/(lrm**2), lr/lrm, lr])
```

```
In [169]: lrs=np.array([1e-4,1e-4,1e-4,1e-3,1e-2])
```

```
In [170]: wd = 1e-7
          wd = 0
          learn.load_encoder('lm1_enc_tt')
```

```
In [171]: learn.freeze_to(-1)
```

```
In [172]: learn.lr_find(lrs/1000)
          learn.sched.plot()
```



```
In [173]: learn.fit(lrs, 1, wds=wd, cycle_len=1, use_clr=(8,3))
```

epoch	trn_loss	val_loss	accuracy
0	0.785493	0.782897	0.704144

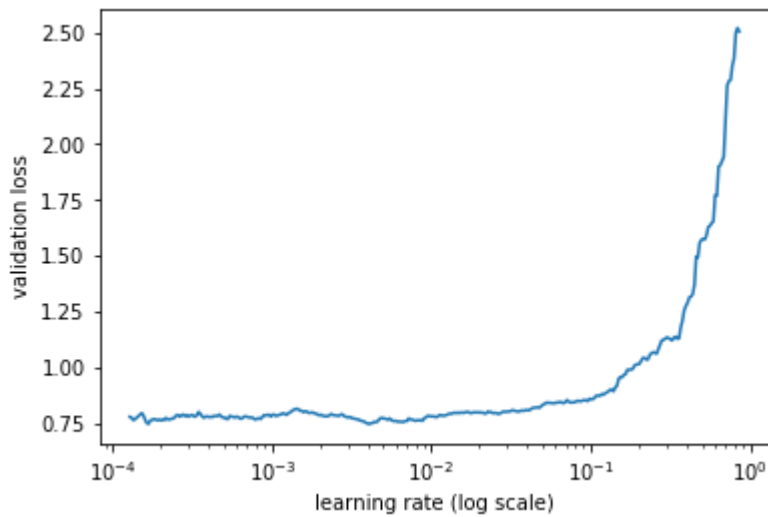
```
Out[173]: [array([0.7829]), 0.7041441384199504]
```

```
In [174]: learn.save('clas_0_tt')
```

```
In [175]: learn.load('clas_0_tt')
```

```
In [176]: learn.freeze_to(-2)
```

```
In [177]: learn.lr_find(lrs/100)
learn.sched.plot()
```



```
In [178]: learn.fit(lrs, 1, wds=wd, cycle_len=1, use_clr=(8,3))
```

epoch	trn_loss	val_loss	accuracy
0	0.786713	0.738291	0.715676

```
Out[178]: [array([0.73829]), 0.7156756741613955]
```

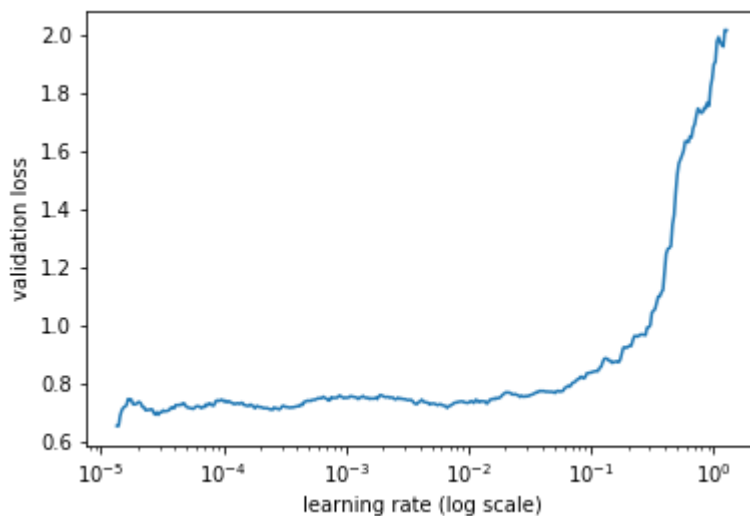
```
In [179]: learn.save('clas_1_tt')
```

```
In [185]: learn.load('clas_1_tt')
```

```
In [186]: learn.unfreeze()
```

```
In [188]: learn.lr_find(lrs/1000)
learn.sched.plot()
```

86% |██████████| | 399/463 [00:15<00:02, 24.81it/s, loss=2.13]

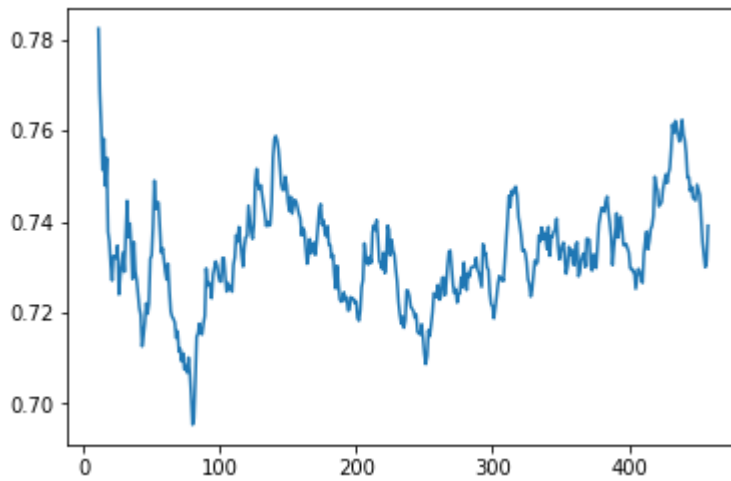


```
In [189]: #learn.fit(lrs, 1, wds=wd, cycle_len=5, use_clr=(32,10))  
learn.fit(lrs/1000, 1, wds=wd, cycle_len=1, use_clr_beta=(20,10,0.95,0.85))
```

epoch	trn_loss	val_loss	accuracy
0	0.723346	0.733949	0.716396

```
Out[189]: [array([0.73395]), 0.7163963964500942]
```

```
In [184]: learn.sched.plot_loss()
```



```
In [398]: learn.save('clas_2')
```