Universal Language Model Fine-tuning for Text Classification

Suchith Kumar Suresh, Shivani Kukreti, Shadman Shoumik

Macquarie University

15 November 2019

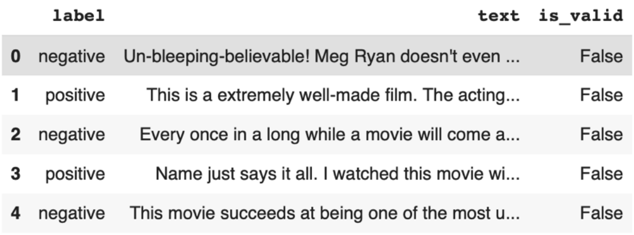
Abstract

Universal Language Model Fine-tuning (ULMFiT) builds on Transfer Learning, which is an NLP method, wherein the knowledge gained by solving a task is used as a starting point when solving another closely related task. The original paper was evaluated on 6 widely researched datasets namely:TREC-6, IMDb, Yelp-bi, Yelp-full, AG and DBpedia With varying numbers of documents and varying document length used as a state-of-the-art text classification and transfer learning approaches to reduce the error by 18-24 % on these datasets. Justification of Quality: The paper was published in May 2018 by author Jeremy Howard and Sebastian Ruder. Paper has been cited 446[1] times in about 1.5 years.

1 Description of Original Dataset

The IMDb dataset[2] is a popular sentiment analysis dataset available to public for personal and non-commercial use. It consists of 50,000 balanced reviews from Internet Movie Database (IMDb) labelled positive and negative .The dataset contains 3 variables: Label, Text and is_valid.

- Label: is the target column of the dataset with 2 polarities as positive and negative
- Text: contains the review text from different users for movies
- is_Valid: is a validation for the review based on polarity



	label	text	is_valid
0	negative	Un-bleeping-believable! Meg Ryan doesn't even ...	False
1	positive	This is a extremely well-made film. The acting...	False
2	negative	Every once in a long while a movie will come a...	False
3	positive	Name just says it all. I watched this movie wi...	False
4	negative	This movie succeeds at being one of the most u...	False

Figure 1: Format of the IMDb dataset

[1] https://scholar.google.com/scholar?q=Universal%20Language%20Model%20Fine-tuning%20for%20Text%20Classification%20Jeremy%20arXiv%202018

[2] http://ai.stanford.edu/ amaas/data/sentiment/

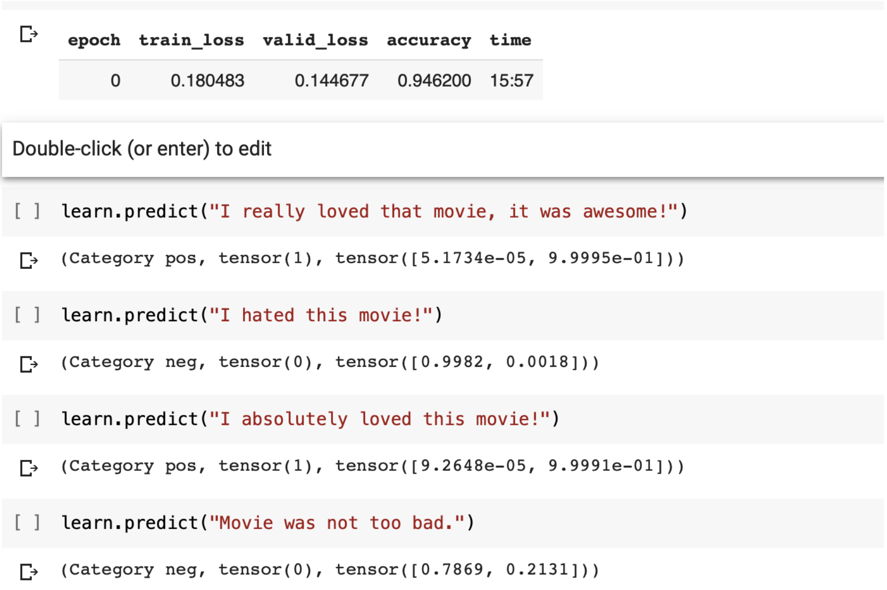
2 Replication of Original work

We got the IMDb datasets Source code used in original paper. Working on the original data and got the results similar to the original results.[3].Transfer Learning that can be applied to any task in NLP, Using Fast.ai[4] python library popular algorithms for natural language tasks.Uses pre-trained word embeddings (hypercolumns) , text.transform[5]for NLP data processing; tokenizes text and creates vocab indexes and language model AWD-LSTM.

ULMFiT follows of Three Steps

- 1 General Domain Language model Pre-training
- 2 Target task Language model fine tuning
- 3 Target task classifier fine tuning

We used GPU Runtime on Google Colab to run the code and it took 14 hours for the code to run in its entirety. The code has been written in Python 3 version.



epoch	train_loss	valid_loss	accuracy	time
0	0.180483	0.144677	0.946200	15:57

Double-click (or enter) to edit

```
[ ] learn.predict("I really loved that movie, it was awesome!")
```

```
(Category pos, tensor(1), tensor([5.1734e-05, 9.9995e-01]))
```

```
[ ] learn.predict("I hated this movie!")
```

```
(Category neg, tensor(0), tensor([0.9982, 0.0018]))
```

```
[ ] learn.predict("I absolutely loved this movie!")
```

```
(Category pos, tensor(1), tensor([9.2648e-05, 9.9991e-01]))
```

```
[ ] learn.predict("Movie was not too bad.")
```

```
(Category neg, tensor(0), tensor([0.7869, 0.2131]))
```

Figure 2: Original results

The figure above shows the results from running the original code on the original dataset. As expected, a high accuracy was achieved of 94.62% and the model was successfully able to identify the polarity of the reviews given.

3 Construction of New data

We decided to choose 3 datasets from 3 different domains to analyse if the model works equally significantly for all. The 3 datasets chosen were:

[3] https://github.com/Suchi1306/ADS-Final-Project/tree/master/ADS%20Project/Paper %20Novel%20project%20and%20Code%20excuted

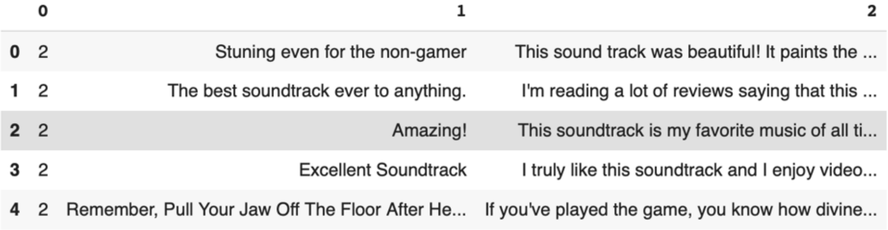
[4] https://docs.fast.ai

[5] https://docs.fast.ai/text.transform.html

3.1 Amazon Reviews

The Amazon Reviews[6] Polarity Dataset contains product reviews from amazon.com. This dataset was already pre-processed[7] and labelled with binary polarities as Class 1 and Class 2: class 1 being a negative review and class 2 being a positive review. The entire dataset contains 1.8 million reviews belonging to class 1 and 2 respectively, out of which I have used 25,000 samples for each class respectively due to system constraints.
The original dataset has 3 variables:

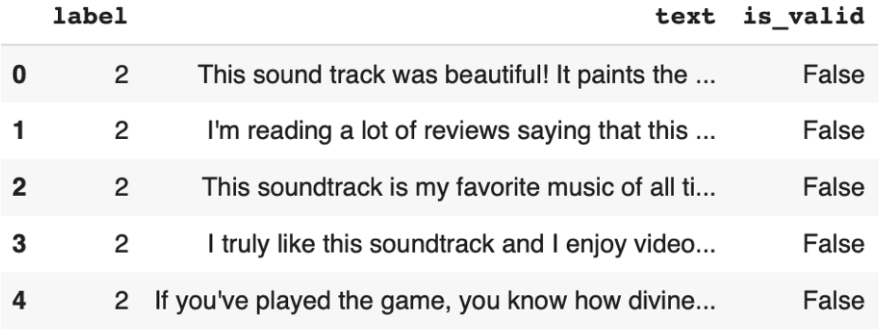
- 0: The target variable having polarities as 1 and 2. (1 = Negative review and 2 = Positive review).
- 1: The title of the review
- 2: The text of the review



	0	1	2
0	2	Stuning even for the non-gamer	This sound track was beautiful! It paints the ...
1	2	The best soundtrack ever to anything.	I'm reading a lot of reviews saying that this ...
2	2	Amazing!	This soundtrack is my favorite music of all ti...
3	2	Excellent Soundtrack	I truly like this soundtrack and I enjoy video...
4	2	Remember, Pull Your Jaw Off The Floor After He...	If you've played the game, you know how divine...

Figure 3: Format of Amazon Review data

Pre-processed the data as below and dropped the ”title” column as it was not required and added the is_valid column, which I used to divide the dataset into validation and training samples. For validation used 10% of the data.



	label	text	is_valid
0	2	This sound track was beautiful! It paints the ...	False
1	2	I'm reading a lot of reviews saying that this ...	False
2	2	This soundtrack is my favorite music of all ti...	False
3	2	I truly like this soundtrack and I enjoy video...	False
4	2	If you've played the game, you know how divine...	False

Figure 4: Pre-processed Amazon Review data

3.2 Drug Reviews

The Drug Reviews[8] dataset provides patient reviews on specific drugs along with related conditions and rating reflecting overall patient satisfaction.The data is split into a train (75%) and test (25%) partition with 200K observations,out of which I have used 50,000 random samples respectively due to system constraints.

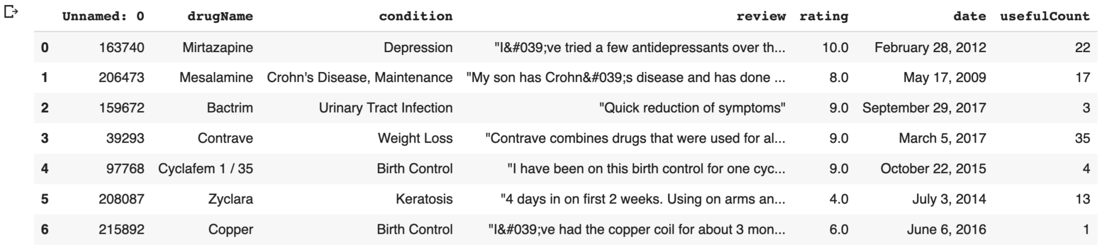
The original dataset has 7 variables:

[6] https://course.fast.ai/datasets

[7] https://arxiv.org/pdf/1509.01626.pdf

[8] https://archive.ics.uci.edu/ml/datasets/Drug+Review+Dataset+%28Drugs.com%29

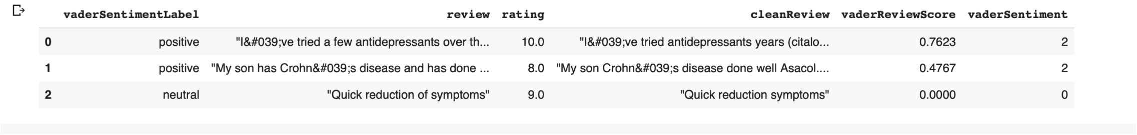
- Unamed : Reference number
- DrugName: Name of the drug
- Condition: Patient medical Condition
- Review : Patient review on Medcine/Drug used for cure
- rating : Drug/ medicine patient rating
- date : review date
- Usefullcount : helpful option selected by review viewers



	Unnamed: 0	drugName	condition	review	rating	date	usefulCount
0	163740	Mirtazapine	Depression	"I&#039;ve tried a few antidepressants over th...	10.0	February 28, 2012	22
1	206473	Mesalamine	Crohn's Disease, Maintenance	"My son has Crohn&#039;s disease and has done ...	8.0	May 17, 2009	17
2	159672	Bactrim	Urinary Tract Infection	"Quick reduction of symptoms"	9.0	September 29, 2017	3
3	39293	Contrave	Weight Loss	"Contrave combines drugs that were used for al...	9.0	March 5, 2017	35
4	97768	Cyclafem 1 / 35	Birth Control	"I have been on this birth control for one cyc...	9.0	October 22, 2015	4
5	208087	Zyclara	Keratosis	"4 days in on first 2 weeks. Using on arms an...	4.0	July 3, 2014	13
6	215892	Copper	Birth Control	"I&#039;ve had the copper coil for about 3 mon...	6.0	June 6, 2016	1

Figure 5: Format of Drug Review data

Pre-processed the data by removing stopwords and used Vader sentiment analysis[9] on review column to create the polarity labels (Positive, Negative and Neutral) based on the Vader-review score. Only 10% of data is used for validation of the model. And dropped the Columns(Unnamed, DrugName, Condition , Date and Usefulcount)as it is not required for this analysis. Added the VaderScore based on review column and generated Vader Sentiment polarity for each review observation based on VaderScore to train the language model.



	vaderSentimentLabel	review	rating	cleanReview	vaderReviewScore	vaderSentiment
0	positive	"I&#039;ve tried a few antidepressants over th...	10.0	"I&#039;ve tried antidepressants years (citalo...	0.7623	2
1	positive	"My son has Crohn&#039;s disease and has done ...	8.0	"My son Crohn&#039;s disease done well Asacol....	0.4767	2
2	neutral	"Quick reduction of symptoms"	9.0	"Quick reduction symptoms"	0.0000	0

Figure 6: Pre-processed Drug Review data

3.3 Coursera Reviews

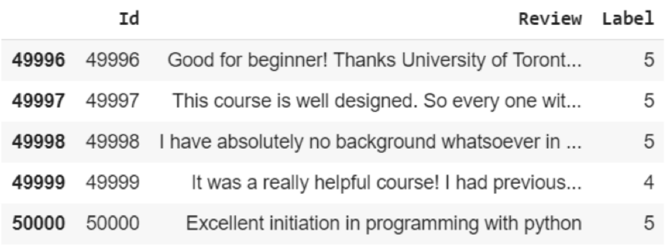
The Coursera Course[10] Review Dataset contains course reviews from Coursera Website.The data was collected from kaggle.The data needed some preprocessing. The dataset had rating instead of polarities.The ratings goes from 1-5 which was converted into binary polarities of positives and negatives.The total dataset contains over 100k data.We have used 50k of the data.

The original Dataset has 3 variables.

- 0: Id , the count number of the data.
- 1: Review , the text review of the courses.
- 2:Label , the rating given by the user

[9] https://medium.com/@oyewusiwuraola/opinion-mining-using-the-uci-drug-review-data-set-part-1-data-loading-and-pre-processing-using-49d3fb6025a8

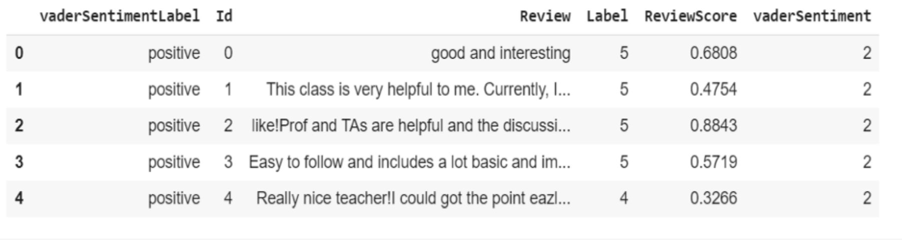
[10] https://www.kaggle.com/septa97/100k-courseras-course-reviews-dataset



	Id	Review	Label
49996	49996	Good for beginner! Thanks University of Toront...	5
49997	49997	This course is well designed. So every one wit...	5
49998	49998	I have absolutely no background whatsoever in ...	5
49999	49999	It was a really helpful course! I had previous...	4
50000	50000	Excellent initiation in programming with python	5

Figure 7: Format of Coursera Review data

The dataset was preprocessed as below. The Review were scored with vader sentiment analysis , score were used to set the values into binary Positive and Negative polarities .This is a snippet of the head in the dataset after the change.



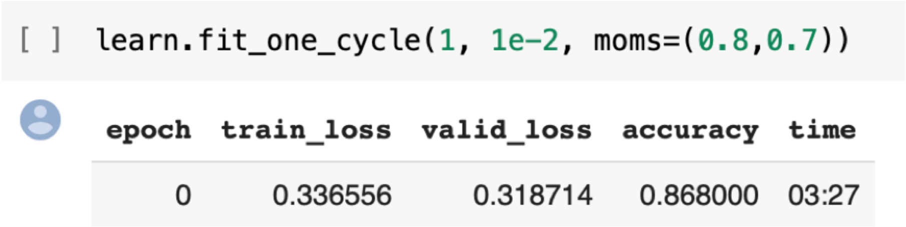
	vaderSentimentLabel	Id	Review	Label	ReviewScore	vaderSentiment
0	positive	0	good and interesting	5	0.6808	2
1	positive	1	This class is very helpful to me. Currently, I...	5	0.4754	2
2	positive	2	like!Prof and TAs are helpful and the discussi...	5	0.8843	2
3	positive	3	Easy to follow and includes a lot basic and im...	5	0.5719	2
4	positive	4	Really nice teacher!I could got the point eazl...	4	0.3266	2

Figure 8: Pre-processed Coursera Review data

4 Results on new data

4.1 Amazon Reviews

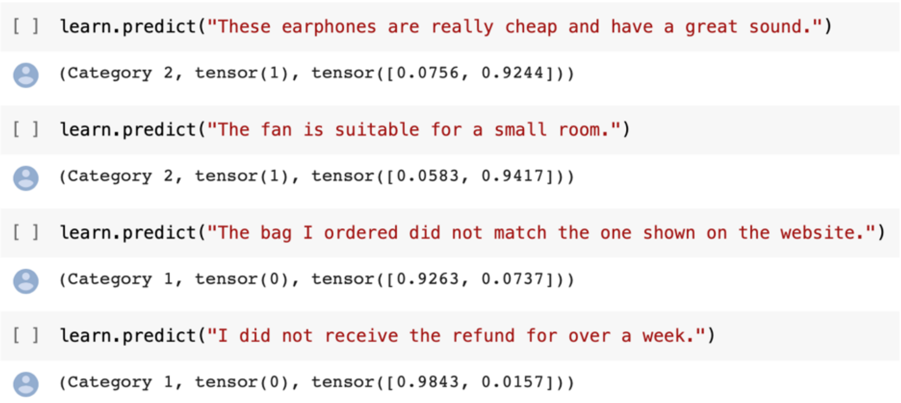
Used the same framework as the original dataset as described in description of the dataset and hence, the code ran successfully on this new dataset and gave an accuracy of 86.8%. (as a comparison, the accuracy we achieved when the code was run on the original dataset was 94.62%).



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

epoch	train_loss	valid_loss	accuracy	time
0	0.336556	0.318714	0.868000	03:27

Figure 9: Fine tuning Accuracy of Amazon Review data

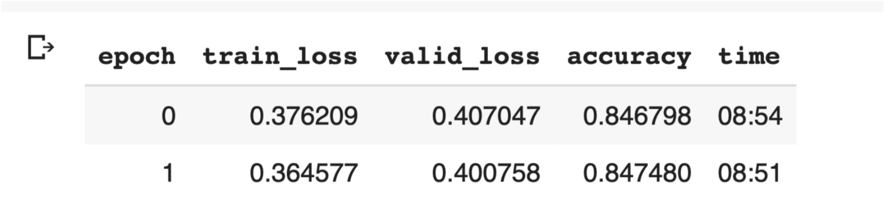


```
[ ] learn.predict("These earphones are really cheap and have a great sound.")
    (Category 2, tensor(1), tensor([0.0756, 0.9244]))

[ ] learn.predict("The fan is suitable for a small room.")
    (Category 2, tensor(1), tensor([0.0583, 0.9417]))

[ ] learn.predict("The bag I ordered did not match the one shown on the website.")
    (Category 1, tensor(0), tensor([0.9263, 0.0737]))

[ ] learn.predict("I did not receive the refund for over a week.")
    (Category 1, tensor(0), tensor([0.9843, 0.0157]))
```

Figure 10: ULMFiT Model Prediction results based on Amazon Review Data

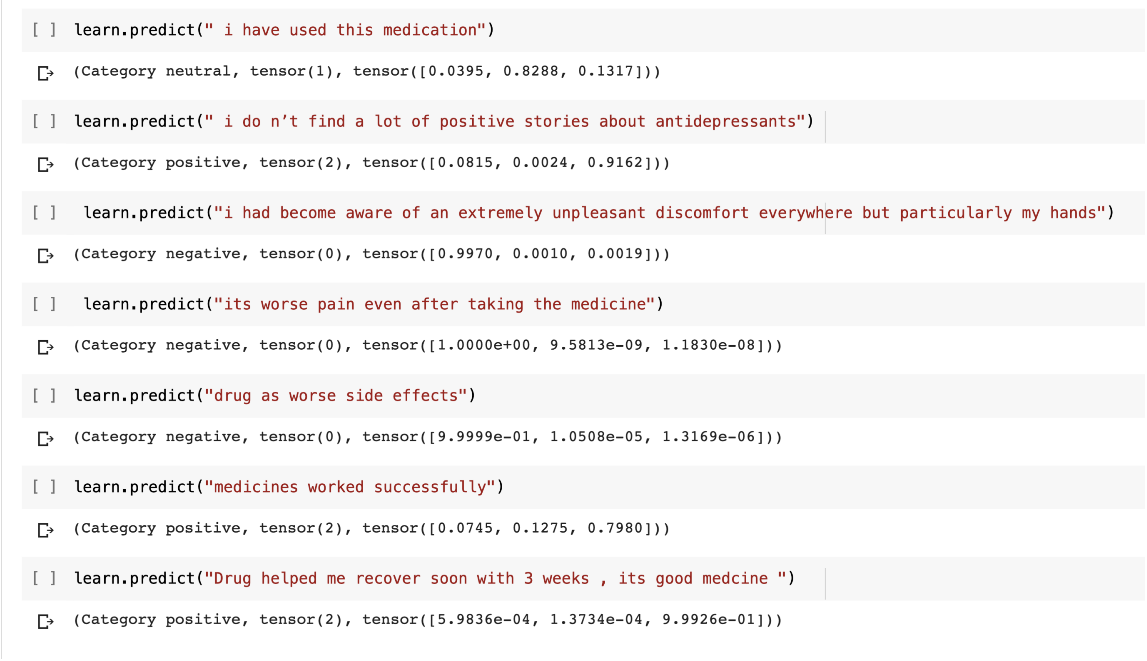
4.2 Drug Reviews

Used the same framework as the original dataset as described in description of the dataset and hence, the code ran successfully on this new dataset and gave an accuracy of 84.8%. (as a comparison, the accuracy we achieved when the code was run on the original dataset was 94.62%).



epoch	train_loss	valid_loss	accuracy	time
0	0.376209	0.407047	0.846798	08:54
1	0.364577	0.400758	0.847480	08:51

Figure 11: Fine tuning Accuracy of Drug Review data

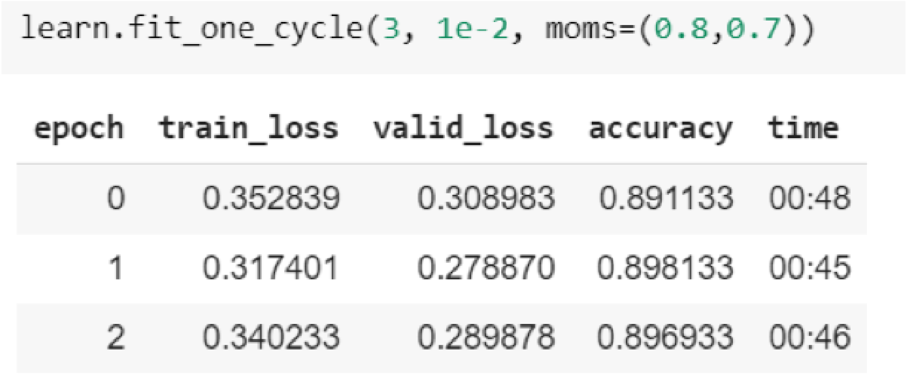


```
[ ] learn.predict(" i have used this medication")
    (Category neutral, tensor(1), tensor([0.0395, 0.8288, 0.1317]))

[ ] learn.predict(" i do n’t find a lot of positive stories about antidepressants")
    (Category positive, tensor(2), tensor([0.0815, 0.0024, 0.9162]))

[ ]  learn.predict("i had become aware of an extremely unpleasant discomfort everywhere but particularly my hands")
    (Category negative, tensor(0), tensor([0.9970, 0.0010, 0.0019]))

[ ]  learn.predict("its worse pain even after taking the medicine")
    (Category negative, tensor(0), tensor([1.0000e+00, 9.5813e-09, 1.1830e-08]))

[ ] learn.predict("drug as worse side effects")
    (Category negative, tensor(0), tensor([9.9999e-01, 1.0508e-05, 1.3169e-06]))

[ ] learn.predict("medicines worked successfully")
    (Category positive, tensor(2), tensor([0.0745, 0.1275, 0.7980]))

[ ] learn.predict("Drug helped me recover soon with 3 weeks , its good medcine ")
    (Category positive, tensor(2), tensor([5.9836e-04, 1.3734e-04, 9.9926e-01]))
```

Figure 12: ULMFiT Model Prediction results based on Drug Review Data

4.3 Coursera Reviews

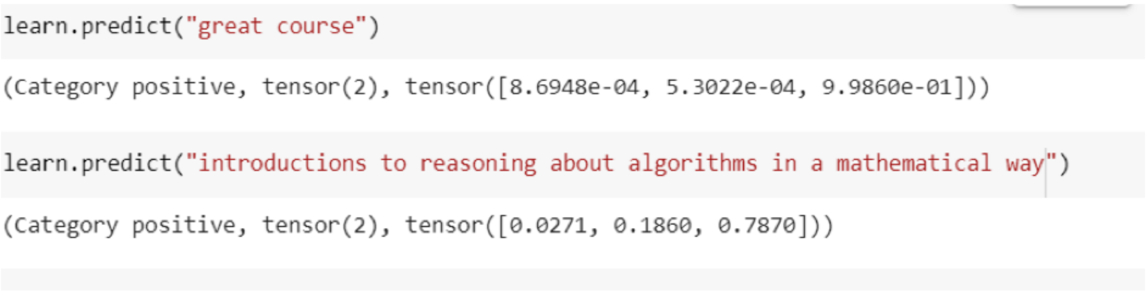
Used the same framework as the original dataset as described in description of the dataset and hence, the code ran successfully on this new dataset and gave an accuracy of 89.69%. (as a comparison, the accuracy we achieved when the code was run on the original dataset was 94.62%).



```
learn.fit_one_cycle(3, 1e-2, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	0.352839	0.308983	0.891133	00:48
1	0.317401	0.278870	0.898133	00:45
2	0.340233	0.289878	0.896933	00:46

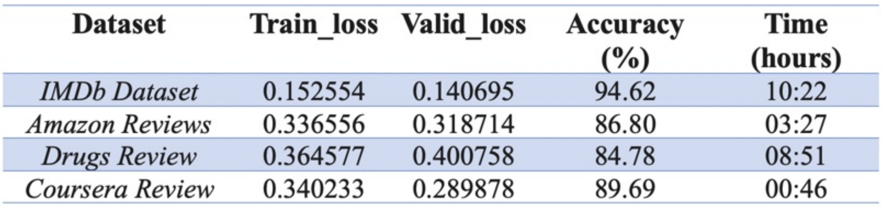
Figure 13: Fine tuning Accuracy of Coursera Review data



```
learn.predict("great course")

(Category positive, tensor(2), tensor([8.6948e-04, 5.3022e-04, 9.9860e-01]))

learn.predict("introductions to reasoning about algorithms in a mathematical way")

(Category positive, tensor(2), tensor([0.0271, 0.1860, 0.7870]))
```

Figure 14: ULMFiT Model Prediction results based on Coursera Review Data

4.4 Results Summary



Dataset	Train_loss	Valid_loss	Accuracy (%)	Time (hours)
IMDb Dataset	0.152554	0.140695	94.62	10:22
Amazon Reviews	0.336556	0.318714	86.80	03:27
Drugs Review	0.364577	0.400758	84.78	08:51
Coursera Review	0.340233	0.289878	89.69	00:46

Figure 15: New Data Results Accuracy comparing the Orginal Data (IMBb)

The ULMFit transfer learning method using novel fine-tuning techniques has produced significant results for the three new datasets chosen giving the highest accuracy of 89.69% on one of the datasets. The model has been able to successfully classify sentiments expressed in sentences.

Reference for Research paper selected [1] Reference for Original DataSet [2]

References

[1] Jeremy Howard and Sebastian Ruder. Fine-tuned language models for text classification. *CoRR*, abs/1801.06146, 2018.

[2] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.