Springboard Capstone Project Report #2

Using Machine Learning and Deep Learning techniques for text classification tasks.

Dmitri Kochubei (09.08.2020) | GitHub repository

Lex Fridman: So, what advice do you have for someone who wants to get started in deep learning?

Jeremy Howard: Train lots of models...

Motivation

The motivation behind this project is not to solve any particular problem. The project was completed to gain hands-on NLP and DL experience. A variety of DL and ML techniques were utilized in order to obtain performance metrics and compare efficiency of different approaches.

Dataset & Problem Statement

The dataset consists of texts of the reviews for 50000 movie titles and the corresponding labels (positive/negative).
Problem statement: Create a DL/ML model to correctly classify textual reviews according to the reviews' sentiment. This is a binary classification problem.

Pre-processing

Any ML/DL area (Computer Vision, NLP, Signal Processing etc.) has its own challenges at the stage of data pre-processing. In NLP the main problem is to find a way of representing textual data in the form of vectors. There are many ways in which it can be accomplished. For this project two techniques were used:

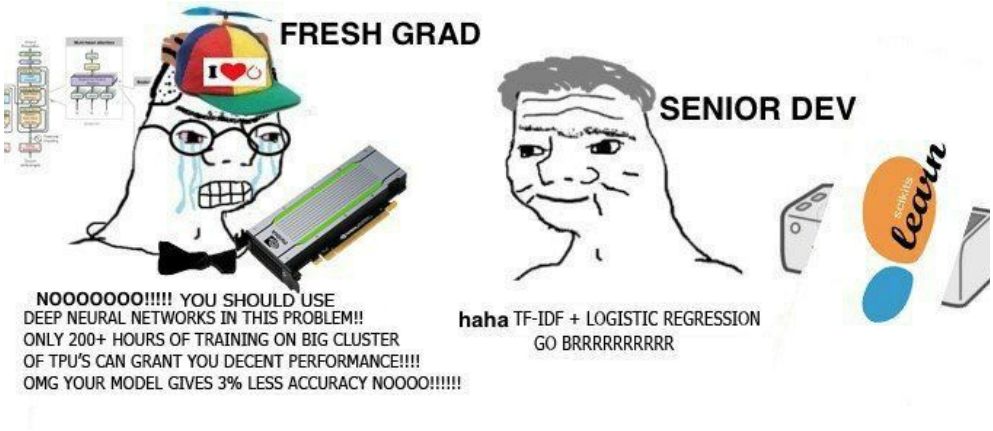
- For the ML approach tf-idf encoding was used.
- For the DL approach pre-trained word embeddings were used. In particular Stanford's GloVe representation of size 300 was used for the embedding layer.
  (Common Crawl (42B tokens, 1.9M vocab, uncased, 300d vectors, 1.75 GB):
  glove.42B.300d.zip)

There are 3 kinds of DL architectures that were used: Dense, LSTM and Convolutional networks. A total of six models were trained, every architecture has two instances with the only difference between them being whether the GloVe weights are updatable or not.

Modeling and performance

#	Model	Update GloVe	Val_Acc	AUC (test)	Recall(Neg)	Recall(Pos)	f1-score (Neg)	f1-score (Pos)
1	LSTM	TRUE	0.8691	0.8804	0.86	0.91	0.91	0.85
2	Conv1D	TRUE	0.8943	0.5053	0.5	0.54	0.93	0.08
3	Dense	TRUE	0.8943	0.8657	0.87	0.86	0.85	0.88
4	LSTM	FALSE	0.5706	0.8028	0.88	0.72	0.82	0.79
5	Conv1D	FALSE	0.8766	0.8820	0.87	0.89	0.88	0.88
6	Dense	FALSE	0.7914	0.8112	0.84	0.79	0.82	0.81
7	Logistic Regression	-	0.8873	0.8977	0.91	0.89	0.89	0.91

Taking into account models' metrics it can be concluded that in real-life scenario logistic regression is probably the best way to solve the problem. It's faster, simpler, uses less resources and it would be easier to integrate.



[Source]

Conclusion

Overall the results that came out of this research are in-line with what could be expected of this dataset and models. LSTM network with updatable GloVe weights has produced superior results although its domination is marginal. tf-idf encoding paired with simple logistic regression has outperformed some more complex DL architectures and took way less time to train than any of the deep neural networks.

Additionally it's worth noting that the problem chosen for this report is probably not complicated enough for DL-based approaches. However as it was stated in the motivation section the goal was to gain NLP experience by training different models, so in this respect the project can be considered successful.

Further steps:

There is a lot that has been done in the domain of binary classification tasks but it's very common to have more classe or even be presented with multi-label classification problems. So the next steps are to expand the toolkit to be able to work with more complex problems.