Final Project Report

Chow Jun Wei and Esraa Sultan
Group name: Jun and Esraa

24 November 2021

Requirements

No requirements on black-box or non-black-box model based on business requirements, as long as it gives good results.

Basic Model

Preprocessing

These are preprocessing pipelines in order.

- Regex Cleaning
- Tokenizers
  - Removal of stop words
  - Lemmatization
- Restricting the result from tokenizers to be between a range (used 3 – 15 words for its length)
- Count Vectorizer transformation.
- TF-IDF transformation.

Models

- Random Forest Classifier acquires 0.75 accuracy with default configurations.
- K-nearest neighbor classifier with number of neighbors = 3 and weights = “distance” achieves 0.45 accuracy.
- Grid-search CV which ensemble TF-IDF Vectorizer and KNN gives score of 0.26.

Data Glacier
Your Deep Learning Partner

Advanced Model

Cleaning Data

- Thresholding line number: picking only those data that are larger than a certain line number. We use 7 lines as threshold (excluding extra newlines).
- OOV article removal: remove all articles with OOV occupying more than threshold percentage of data. We use 10%.
- Regex cleaning of data.
- Spelling mistakes cleaning: clean those we saw mistakes in. Unfortunately they’re not necessarily all the mistakes made in the files.
- British to American english: « color » and « colour » should have the same meaning with same embeddings, so convert everything that conflicts to one of it.

Model: AWD-LSTM[1]

- Uses DropConnect and a variant of Average-SGD (NT-ASGD) along with several other regularization techniques.
- Default of fastai's NLP model and was state of the art three years ago, giving acceptable results.

Training Method: ULMFiT

- Three steps for ULMFiT are:
- AWD-LSTM pretrained on WikiText-103 corpus (made available already).
- (**Pretraining stage**) Transfer learning to your corpus of words (corpus predict what the next word is).
- Remove the head and replace it with classification head, and train for classification task.

Pretraining Stage

- Train a few epoch. **Not required for best accuracy**. We only want the body of the model to get used to our corpus of words, not necessarily to predict the next word nor overfitting.
- Better accuracy here doesn't necessarily gives better accuracy in classification task downstream. However under-training at this phase might means not yet warm up for classification task, hence might give worse classification results downstream.
- Depending on number of epochs train, one manages to get about 0.33 for 3+1 epochs and about 0.34 for 10+1 epochs.
- +1 because 1 additional epoch is trained with the model frozen except the head, and the rest being fine-tuning the whole model.

Classification Training Stage

- Gradual Unfreezing technique: slowly unfreeze from the head towards the body, one layer by one layer, each training for some epochs. This gives better result.
- Gradual decrement of learning rate: as unfreezing goes, learning rate decrease to not perturb the pretrained model weights by too much.
- Results around 0.82-0.83 for several epochs of training.
- Best benchmarking result are at 0.886. [2] Our result would let us be in the top 11.

References

[1] https://yashuseth.blog/2018/09/12/awd-lstm-explanation-understanding-language-model/

[2] https://paperswithcode.com/sota/text-classification-on-20news