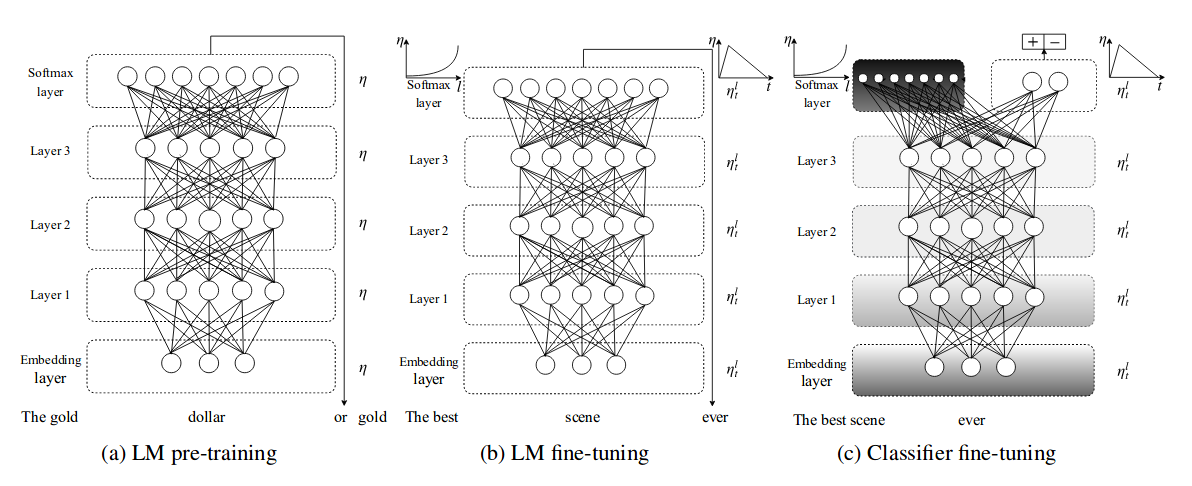
Employ Transfer Learning Approach for Sentiment Classification at TenPoint7

2020-02-27

Universal Language Model Fine-tuning (ULMFiT)[[1]](#footnote-0) is an effective transfer learning approach for solving variety tasks in Natural Language Processing (NLP). The authors shed some light on the three common tasks of text classification: sentiment analysis, question classification, and topic classification. Their work especially achieves state-of-the-art results on six text classification tasks.

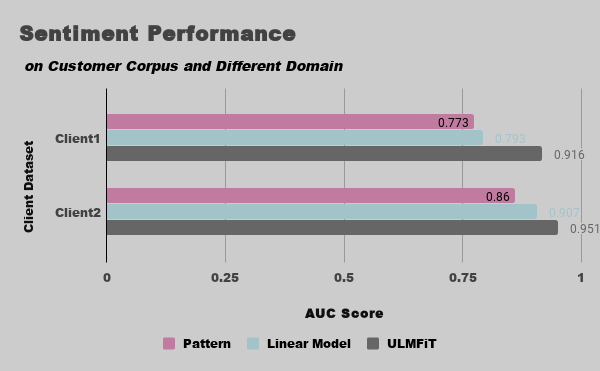
## Underlying ideas of ULMFiT



1. ULMFiT enabled robust inductive transfer learning approach for any NLP tasks with the same 3 layers architecture
2. ULMFiT pre-train a Language Model (LM) on large general corpus, and then fine tunes it on target task using novel technique. The model is universal in the sense that meet those practical criteria:
   * It works across tasks varying in document size, number, label type
   * Use a single architecture and training process
   * It requires no custom feature engineering or pre-processing
   * It does not require additional in-domain or label document
3. The famous AWD-LSTM LM was used for ULMFiT with various tuned dropout parameters.
4. ULMFiT consists of three following steps:
   * General domain LM pre-training (most expensive task, train once)
   * Target task LM fine-tuning
   * Target task classifier fine-tuning
5. The second (Target task LM fine-tuning) step: converges faster as it only needs to adapt to the target data, and allows to train a robust LM even for small corpus. The proposed discriminative fine-tuning and slangted triangular learning rate (STLR) for fine-tuning the LM
   * Discriminative fine-tuning: Different layers capture different types of information is the motivation behind the discriminative fine-tuning. Instead of using the same learning rate for all layers of the model, discriminative fine-tuning allows to tune different learning rate for different layers, for context of SGD.
   * STLR modifies the triangular learning rate with a short increase and a long decay period, which found key for good performance. Which is first linearly increases and then linearly decay its according to the predefined update schedule.
6. Beside discriminative fine-tuning and STLR, they proposed gradual unfreezing for fine-tuning the classifier. Gradual Unfreezing the model at the last layer as it contains least general knowledge. They first unfreeze the last layer and then fine-tune all the un-frozen layers for each epoch. Then unfreeze the next lower layer and repeat. Until they fine-tune all layers convergence at the last iteration.

## Employ ULMFiT at TenPoint7

At [TenPoint7](http://www.tenpoint7.com/), we provide data consulting services and we need to continuously employ the latest research and technologies by one way or another. One of those problems is the Sentiment Classification for many of our clients. They have built their own model before and wanted to improve the model performance. Initially, we've leveraged the lexicon-based[[2]](#footnote-1) approach to achieve the first benchmark performance. Afterward, the linear model with TF-IDF approach has been used to build a traditional classifier from the public data-sets. The linear model seems to fit right to the customers' need both in accuracy and time, along with its ease of deployment in production environment. Then one day, we heard the ULMFiT approach achieves impressive results and got a lot of attention from the community at the time, we intend to explore this transfer learning approach. The beauty of ULMFiT is its availability from fast.ai[[3]](#footnote-2), they open-source the pre-trained models and code[[4]](#footnote-3) with specific examples for fine-tuning on your target data-set. The modeling processes simply took the general domain AWD-LSTM language model at first and fine-tuned the language model in the second step then finally trained the classifier from the same public data-set that was used for the linear model.



We aim to compare the model's performance on different domain and for our internal customer corpus. Using the AUC metric to judge the model's performance in this experiment. The ULMFiT outperform the other benchmarks not only on the public data, but also on the other domain corpus from our clients. Motivating by this outcome, we took a step to employ this approach for Sentiment prediction feature of Addy.ai[[5]](#footnote-4) where it improves the accuracy to 18% compared to the previous method for this feature.

1. Jeremy Howard and Sebastian Ruder, [Universal Language Model Fine-tuning for Text Classification](https://arxiv.org/abs/1801.06146), 2018 [↑](#footnote-ref-0)
2. [pattern.en](https://www.clips.uantwerpen.be/pages/pattern-en#sentiment) [↑](#footnote-ref-1)
3. [fast.ai](https://www.fast.ai/) [↑](#footnote-ref-2)
4. http://nlp.fast.ai/ulmfit [↑](#footnote-ref-3)
5. [Addy.ai](https://addy.ai/), TenPoint7's data product that specialize for text analytics [↑](#footnote-ref-4)