Model Distillation for Sentiment Analysis given Tiny Labelled Dataset

Jiali Lu
jialilu.physics@gmail.com

1. Motivation

State of the art accuracy can be obtained for many natural language processing (NLP) tasks by fine-tuning large language models (LLMs). Since LLMs are slow, model distillation is often performed to obtain a smaller, faster model with similar accuracy. The smaller model is then used for production.

In this project, I study model distillation for sentiment analysis task in a practical setting, where the labelled dataset is small but unlabelled data is abundant. This is often the case as labelling is usually expensive. It is unclear if model distillation is still beneficial in this scenario. After all, if the labelled dataset is too small and the fine-tuning of LLM becomes unstable, we might obtain better performance if we fine-tune a smaller model directly.

2. Analysis and Results

Here I study and compare the two approaches on sentiment analysis task on Yelp review dataset [1]. To make the study reflect a more realistic setting, I assume only 11,000 reviews have sentiment labels (in terms of 1 to 5 stars), though the text of all 650,000 reviews are available. Among the labelled reviews, 10,000 would be used for training and 1,000 for validation.

2.1. Fine-tuning BERT-base

A BERT-base cased model is fine-tuned on the 10,000 reviews with sentiment labels. The main difficulty lies in choosing the hyper-parameters. Inappropriate choice can have major negative impact on the model accuracy. For example, setting too high a learning rate leads to catastrophic forgetting. Here I follow the recommendation by Mosbach et al [2] and the specific values of the hyper-parameters are listed in the appendix.

2.2. Training BERT-small by Model Distillation

The fine-tuned BERT-base model is then used as a teacher to train a more compact student model, based on BERT-small. The student model is trained on the soft labels provided by the teacher, i.e. the probability that the specific review belongs to each class according to the teacher model. The specific hyper-parameters can be found in appendix.

2.3. Fine-Tuning BERT-small directly

As an alternative to the model distillation approach, we could obtain a BERT-small model by fine-tuning on the labelled dataset directly, without involving a BERT-base model. I carefully followed this approach, again following the suggestion of Mosbach et al [2]. The model obtained from direct fine-tuning can be seen as a baseline, and any performance gap between the baseline and the distilled model should be attributed to the model distillation procedure.

TABLE 1. TRAINING OF THE FOUR BERT BASED MODELS

	language model	training procedure	sample size	training time (h)
1	BERT-base	fine-tuning	10k	4.05
2	BERT-small	model distillation	10k / 639k *	12.2
3	BERT-small	fine-tuning	10k	0.58
4	BERT-small	fine-tuning	20k	1.18

*: trained on 639k reviews soft-labelled by BERT-base above

2.4. Controlling for Budget Difference

While model distillation indeed improves the model accuracy substantially, it also takes longer, and thus incurs a higher computational cost. If the cost were spent toward labelling more data, the accuracy would also increase for the direct fine-tuning approach. To take this into account, I fine-tuned BERT-small on 20,000 labelled reviews instead, 10,000 more than in model distillation. Details on how this number is determined can be found in appendix.

If we compare the distilled BERT-small model and BERT-small fine-tuning on increased amount of data, the accuracy gap would be the gain associated with the model distillation technique, controlling for the same budget.

2.5. Performance Comparison

The performance of the four models mentioned above is summarized in Table 2. Here 5-class accuracy stands for the rate when the model correctly predicts the number of stars associated with the review. The accuracy is calculated in the test set included in the original dataset, and thus is comparable with other results[1].Polarity of a review can be negative (1 or 2 stars) or positive (4 or 5 stars). The polarity accuracy calculates the rate when the predicted polarity matches the truth, and 3-star reviews are excluded from calculation. This is in line with majority of the publications. Mean Absolute Error (MAE) is the absolute difference between the predicted number of stars and the true value. Inference speed is calculated from the time the model takes to make predictions on the testing set.

By comparing model No. 2 and model No. 3, we can see model distillation improves accuracy by 2.5% (5-class) and 2.1% (polarity). Given 10,000 more labelled reviews, model No. 4 has increased accuracy compared to No. 3. However, the accuracy is still substantially lower than that from model distillation. Thus, model distillation is still beneficial even when we have a tiny labelled dataset.

TABLE 2. PERFORMANCE OF THE FOUR BERT BASED MODELS

	model	accuracy (%, 5-class)	accuracy (%, polarity)	MAE	inference speed (review/s)
1	BERT-base (fine-tuned, 10k)	60.0	94.9	0.450	48.7
2	BERT-small (distillation, 10k)	61.5	95.6	0.428	337.8
3	BERT-small (fine-tuned, 10k)	59.0	93.5	0.489	335.9
4	BERT-small (fine-tuned, 20k)	60.4	94.2	0.470	339.4

3. Further Discussion

Model distillation is often used to obtain a more compact deep learning model while retaining most the accuracy. In this project, I consider model distillation under realistic scenario when only a tiny fraction of the whole dataset is labelled. In this case, we can fine-tune a large model on the labelled data and perform model distillation using the unlabelled ones. If the large model is fine-tuned properly, model distillation would still result in a substantial gain in accuracy for the smaller model.

There are other techniques when the labelled dataset is tiny. For example, one can obtain a paraphrased version of the original text by first translating it into another language, and then translating it back (for example, English to French and back to English). This is sometimes called back-translation, although the original idea in neural translation context is somewhat different.

One factor not considered in the correct study is the fluctuation in model accuracy resulting from the random initialization of neuron weights. This can be taken into account by repeating the study for many times to estimate the fluctuation. However, I chose not to take such additional steps due to the computational budget constraint.

4. Conclusion

In this project I studied model distillation under the scenario when the fraction of labelled dataset is tiny. Compared to the large teacher model, the distilled model is smaller, and faster. Comparing to the same sized model directly obtained from fine-tuning, I show that model distillation provides substantial gain in accuracy. This is still the case even if the additional computational cost for distillation is spent towards labelling more data.

1. For example, the state-of-the-art accuracy on the whole 650k training set is 73.28% from [3]. Note, however, our models are trained only on 1.5% of the whole dataset.

Appendix

Fine-tuning

Following the suggestion in Section 6 of [2], I used AdamW with bias correction and a maximal learning rate of 2×10^{-5}. The fine-tuning lasts for 20 epochs with learning rate linearly increasing to maximal learning rate in the first 10% of iterations and then linearly dropping to zero afterward. Given limited memory capacity, data is loaded in batch of 8 and gradient accumulation is used to obtain an effective batch size of 16. The model is evaluated every 100 steps and we keep track of the best model in terms of accuracy on the validation set.

Controlling for the Difference in Model Budget

To obtain a BERT-small by model distillation, I need to first fine-tune a BERT-base and then use it as a teacher to train BERT-small. The training time is much longer (12.2 hours vs 0.58 hour). The question is how many more reviews can we label given the additional budget we spent on training.

Given additional training time T, price of training time per hour r_{time} and price per label r_{label}, the additional number of labels we could have obtained is $N = \frac{T \times r_{time}}{r_{label}}$. To get an upper bound, we can use $T = 12$, $r_{time} = 8$ (often the price per compute is more like \$3 per GPU per hour, for example on AWS) and $r_{label} = 0.01$ (which is the minimal reward per hit on Amazon Mechanical Turk). This results in $N = 9,600$, which is then rounded up to 10,000.

References

[1] Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. Advances in neural information processing systems, 28.

[2] Mosbach, M., Andriushchenko, M., & Klakow, D. (2020). On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. arXiv preprint arXiv:2006.04884.

[3] Abreu, J., Fred, L., Macêdo, D., & Zanchettin, C. (2019, September). Hierarchical attentional hybrid neural networks for document classification. In International Conference on Artificial Neural Networks (pp. 396-402). Cham: Springer International Publishing.