A Study of Compressed Language Models in Social Media Domain

Linrui Zhang, Belinda Copus
University of Central Missouri
W.C. Morris 222, Warrensburg, Missouri
{lzhang, copus}@ucmo.edu

Abstract

Transfer learning from large-scale language models is witnessing incredible growth and popularity in natural language processing (NLP). However, operating these large models always requires a huge amount of computational power and training effort. Many applications leveraging these large models are not very feasible for industrial products since applying them into power-scarce devices, such as mobile phone, is extremely challenging. In this case, model compression, i.e. transform deep and large networks to shallow and small ones, is becoming a popular research trend in NLP community. Currently, there are many techniques available, such as weight pruning and knowledge distillation. The primary concern regarding these techniques is how much of the language understanding capabilities will be retained by the compressed models in a particular domain? In this paper, we conducted a comparative analyses between several popular large-scale language models, such as BERT, RoBERTa, XLNet-Large and their compressed variants, e.g. Distilled BERT, Distilled RoBERTa and etc, and evaluated their performances on three datasets in the social media domain. Experimental results demonstrate that the compressed language models, though consume less computational resources, are able to achieve approximately the same level of language understanding capabilities as the large-scale language models in the social media domain.

Introduction

As transfer learning becomes more prevalent in natural language processing, large-scale pre-trained language models, such as BERT (Devlin et al. 2018), XLNET (Yang et al. 2019), and GPT (Radford et al. 2019), are becoming fundamental tools for many NLP tasks. These models normally contain millions of parameters. Despite the fact that larger parameters can lead to better performance, it also raises several concerns. First, these models are very difficult to be fine-tuned, especially under constrained computational power. For instance, the latest NVIDIA Megatron LM (Shoeybi et al. 2019) contains 8.3 billion parameters. Even with cloud computing, it will take several weeks to fine-tune this model before researchers can get a satisfactory result. Second, it is not feasible to run these models on devices like smartphones. In this case, building light-weight, responsive and energy-efficient models is becoming an urgent need in the NLP community.

Hinton (Hinton, Vinyals, and Dean 2015) introduced knowledge distillation, a compression technique in which a small model is trained to reproduce the behavior of a larger model. This technique has been successfully used to compress many large-scale models in different fields, such as computer version (Cho and Hariharan 2019) and video analysis (Nie et al. 2019). Hugging Face (Sanh et al. 2019) made the first attempt to introduce knowledge distillation into the NLP field. They compressed the popular BERT language model and proposed a distilled version — DistilBERT and assessed it on GLUE Benchmark. The experimental results showed that the DistilBERT model is able to reduce the size of BERT model by 40%, while retaining 97% of its language understanding capabilities and being 60% faster.

Though the GLUE dataset (Wang et al. 2018) is a well-known benchmark for language understanding, it lacks data from the social media domain, while social media is ubiquitous in human communication of thoughts and opinions. In order to fill this gap, we continued their research and extended the assessment process on two different tasks in the social media domain — sentiment analysis and irony detection. Specifically, we fine-tuned and evaluated a variety of pre-trained large-scale language models and their alternatives (e.g. BERT, RoBERTa and etc) as the baseline, and then conducted a thorough investigation and comparison with their compressed variants (e.g. DistilBERT, MobileBERT and etc) on the above-mentioned tasks. In addition, we also compared these pre-trained language models with the traditional trained (RNN and CNN-based) models selected from the SemEval-2018 top performers. The primary goal of this paper is to investigate whether the energy-efficient compressed language models could still retain high-level language understanding capabilities while being extended to social media domain.

Task Description

In this section, we will introduce the NLP tasks and the corresponding corpora. We conducted our experiments on two downstream tasks (1) sentiment analysis and (2) irony detection of Twitter.

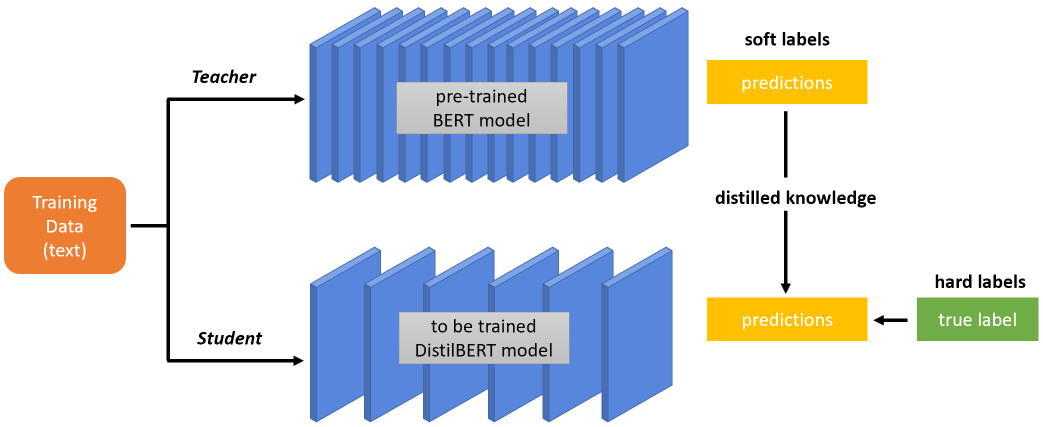

Figure 1: The training process of the DistilBERT model

Sentiment Analysis Task

This task is a multiclass classification task. Given a tweet, classify it into one of seven ordinal classes (from -3 to 3), corresponding to various levels of positive and negative sentiment intensity, that best represents the mental state of the tweeter. For example:

- Tweet: *And here we go again*
- Label: *-2 (moderately negative emotion)*

Irony Detection Task

This task contains two subtasks. The first subtask (subtask A) is a binary classification task where the system has to predict whether a tweet is ironic or not. For example:

- Tweet: *I just love when you test my patience!!#not*
- Label: *ironic*

The second subtask (subtask B) is a multiclass classification task where the system has to predict one out of four labels describing i) verbal irony realized through a polarity contrast, ii) verbal irony without such a polarity contrast, iii) descriptions of situational irony, iv) non-irony. For example:

- Tweet: *I really love this year's summer; weeks and weeks of awful weather*
- Label: *Verbal irony realized through a polarity contrast*

Task Corpora

For the irony detection task, we selected the corpora from SemEval-2018 Task 3: Irony Detection in English Tweets (Van Hee, Lefever, and Hoste 2018). It contains two subtasks, binary irony detection task (subtask A) and multiclass irony detection task (subtask B). For the sentiment analysis task, we chose the corpus from SemEval-2018 Task 1: Affect in Tweets (Mohammad et al. 2018). All the corpora have already been divided into three sets - train, dev, and test. The statistics of the corpora are shown in Table 1.

Task	Set	Number of tweets
Irony Detection	Train	3067
	Dev	767
	Test	784
Sentiment Analysis	Train	1181
	Dev	449
	Test	937

Table 1: The statistics of the corpora

Approach

In this section, we will briefly introduce the knowledge distillation technique and the DistilBERT model. A more detailed explanation about distillation can be found in the original Hugging Face paper (Sanh et al. 2019). In addition, we will also introduce how to fine-tune the DistilBERT model and re-use it in our downstream tasks.

Model Structure

Knowledge distillation (Hinton, Vinyals, and Dean 2015), also known as teacher-student learning, is a compression technique in which a compact (or student) model is trained to reproduce the behavior of a larger (or teacher) model.

In Hugging Face's proposal (Sanh et al. 2019), the student - DistilBERT - has the same general architecture as the teacher - BERT, but with a reduced number of layers, specifically, the token-type embeddings and the pooler are removed while the number of layers is also reduced by a factor of two. In addition, most of the operations used in the Transformer architecture are also highly optimized. With these optimization and architectural choices, the sub-network DistilBERT can reduce the size of the BERT model by 40%.

Model Training

In a traditional classification problem, a standard training objective is to minimize the cross-entropy between the model's predicted distribution and the hard targets (one-hot

Model	Subtask A				Subtask B			
	ACC	**Precision**	**Recall**	F_1	**ACC**	**Precision**	**Recall**	F_1
DistilBERT(uncased)	0.684	0.677	0.683	0.677	0.654	0.451	0.474	0.456
BERT(uncased)	**0.717**	0.704	0.696	0.699	0.667	0.456	0.492	0.470
DistilRoBERTa	0.633	0.664	0.546	0.489	0.644	0.478	0.328	0.327
RoBERTa	0.607	0.575	0.515	0.434	0.621	0.385	0.319	0.314
BERT-Large	0.627	0.661	0.659	0.627	0.603	0.151	0.250	0.189
MobileBERT	0.650	0.665	0.670	0.650	0.648	0.490	0.477	0.460
XLNET	0.707	0.705	0.714	0.703	0.667	0.440	0.452	0.445
XLNET-Large	0.707	**0.737**	**0.738**	**0.707**		Resource Exhausted		
DistilBERT (cased)	0.691	0.684	0.691	0.685	0.672	0.615	0.428	0.431
BERT (cased)	0.689	0.673	0.667	0.669	**0.691**	**0.492**	**0.494**	**0.484**

Table 2: Comparison of the large-scale language models and their compressed versions on the irony detection task

empirical distribution of the gold class). In contrast to training with a cross-entropy over the hard targets, Hugging Face trained the student model with a distillation loss over the soft target (probability distribution of the teacher's output).

$$L_{ce} = \sum_i t_i * \log(s_i) \tag{1}$$

where *t* is the logits from the teacher and *s* is the logit of the student. In this case, the student model can leverage the knowledge of the teacher model.

In addition, following Hinton (Hinton, Vinyals, and Dean 2015), they also used a *softmax-temperature* to further expose the mass of the distribution over the classes. The formula for *softmax-temperature* is as follows:

$$p_i = \frac{exp(\frac{z_i}{T})}{\sum_j exp(\frac{z_j}{T})} \tag{2}$$

where *T* controls the smoothness of the output distribution and z_i is the model score for the class *i*. The same temperature *T* is applied to the student and the teacher at training time, while at inference, *T* is set to 1 to recover a standard *softmax*.

The entire model structure is shown in Figure 1.

Fine-tuning the Compressed Models

After the training process described in the previous section, the BERT model could be distilled into the compacted version DistilBERT. With similar approach, Hugging Face (Sanh et al. 2019) also compressed other large-scale language models, such as XLNET-Large, RoBERTa and etc. We leveraged their model implementations (TensorFlow Version [1]) and modified the output layer to adapt it to our downstream tasks. Then used the standard *cross-entropy* loss as the training objective to fine-tune those models.

The experiments are performed on the Google Colab, specifically on a Tesla T4 machine.

[1]The implementation can be found at `https://github.com/huggingface/transformers`

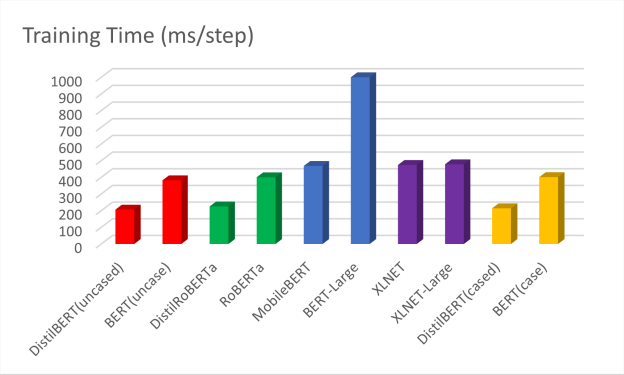

Figure 2: The training time of different language models on binary irony dectection task

Experiment and Analysis

Experiment 1 — Large-scale Language Models VS Compressed Language Models on Irony Detection Task

In this section, we compared the performances of the large-scale language models with their compressed versions on the irony detection task using the standard metrics as in the SemEval competition — accuracy, precision, recall and F_1. Table 2 illustrates the experimental results. From the table, we failed to observe a superior advantage of the large-scale language models over their compressed versions. For example, the performance of DistilBERT is only 0.02 below regular BERT in F_1 measure. Similar observation could be found in other models, such as RoBERTa and XLNET. This means the compressed models, even with a reduced size, can still retain most of the language understanding capabilities.

In addition to the performances, we also compared the time efficiency of each model on the binary irony detection task (subtask A), and the results are demonstrated in Figure 2. The y-axis shows the training time in millisecond per step. From the figure, we could observe that the training time of the DistilBERT is 206 ms / step while BERT is 382 ms / step. This indicates that DistilBERT saves approximately 45 % of training efforts. The same trend could be found in

other models as well, such as RoBERTa and MobileBERT. Though not shown in the paper, we also evaluated the training time of each model on subtask B and noted similar results, e.g. the training time of DistilBERT for subtask B is 155 ms / step which is 46 % faster than BERT (training time 289 ms / step).

Task	Pearson r
DistilBERT (uncased)	73.1
BERT (uncased)	71.2
DistilRoBERTa	67.9
RoBERTa	70.0
MobileBERT	74.8
DistilBERT (cased)	71.7
BERT (cased)	73.2

Table 3: Comparison of the large-scale language models and their compressed versions on the sentiment analysis task

Experiment 2 — Large-scale Language Models VS Compressed Language Models on Sentiment Analysis Task

In this section, we compared the performances of the large-scale language models with their compressed versions on the sentiment analysis task using the Pearson correlation coefficient ($Pearson\ r$) which is the original evaluation manner in SemEval. Table 3 illustrates the experimental results. Similar to the irony detection task, we observed a neck-to-neck performance between the large-scale and compressed models. In some cases, the compressed models even surpass the performance of the large-scale models. For instance, the uncased DistilBERT beats the regular BERT by 2.2 in Pearson r and the best performer in this experiment is MobileBERT, exceeding regular BERT by 3.6 in Pearson r. We believe this is due to the domain transferring issue caused by the limited size of the fine-tuning data. BERT is a pre-trained model, which already contains some pre-existing knowledge about the original domain. It will be very difficult to adapt it into a new domain, if we don't have enough fine-tuning data, since BERT will still be influenced by the previous knowledge. On the contrary, DistilBERT has a reduced size in structure with less parameters [2], which makes it much easier to forget the former knowledge and be transferred to a new domain.

Experiment 3 — Large-scale Pre-trained Models VS Trained Models

In this section, we compared the performances between the pre-trained models and the traditional (RNN or CNN based) trained models. The trained models are selected from the top performers in SemEval-2018. Table 4 shows the comparison results on the irony detection and sentiment analysis task respectively.

[2]DistilBERT has around 66M parameters as opposed to 110M of BERT

From Table 4, we could observe that (1) in the binary irony detection (subtask A), the pre-trained model XLNET-Large model outperforms the NTUA-SLP team (Baziotis et al. 2018) (ranked 3^{rd} place in SemEval); (2) in the multiple irony detection (subtask B), the BERT (cased) model outperforms the NIHRIO team (Vu et al. 2018) (4^{th} place in SemEval); (3) the performance of the MobileBERT is also comparable to the top-ranked teams SeerNet (Duppada, Jain, and Hiray 2018) and Amobee (Rozental and Fleischer 2018) in the sentiment analysis task.

Task	Model	prec.	recall	F_1
Binary	XLNET-Large	0.737	0.738	0.707
Irony	UCDCC	0.788	0.669	0.724
Detection	NTUA-SLP	0.654	0.691	0.672
Multiclass	BERT(cased)	0.492	0.494	0.484
Irony	UCDCC	0.577	0.504	0.507
Detection	NIHRIO	0.545	0.448	0.444
Task	**Model**	**Pearson r**		
Sentiment	MobileBERT	74.8		
Analysis	SeerNet	83.6		
Task	Amobee	81.3		

Table 4: The comparison between the top performers of the pre-trained and traditional trained models on the irony detection task and sentiment analysis task

Though the results of the large-scale pre-trained models are promising, there is not much that stands out performance-wise over the traditional trained models. For instance, in the sentiment task, the MobileBERT model is 8.8 in Pearson r below the top 1 performer SeerNet (Duppada, Jain, and Hiray 2018). We believe this is due to the overfitting issue. The total size of the data we could use for fine-tuning is only 1181. This is really a small size considering that MobileBERT contains 25M trainable parameters. The data will overfit the model easily and induce the performance loss. In the future, we plan to employ data augmentation techniques to increase the amount of fine-tuning data and investigate whether this could alleviate the overfitting problem and improve the system performance.

Conclusion

In this paper, we continued the research of Hugging Face and extended their work about knowledge distillation to a new semantic domain. We conducted experiments to compare the performance of a variety of pre-trained large-scale language models with their compressed variants. Experimental results show that compared with large-scale language models, compressed language models are still able to retain their language understanding capabilities and are faster in the social media domain. One drawback about our experiments is that the task corpora contain a very limited amount of fine-tuning data, which caused the domain transferring issue and overfitting issue. In the future, we plan to provide a part-way solution by applying data augmentation to artificially enlarge the training set so as to improve the model performance.

References

[Baziotis et al. 2018] Baziotis, C.; Athanasiou, N.; Chronopoulou, A.; Kolovou, A.; Paraskevopoulos, G.; Ellinas, N.; Narayanan, S.; and Potamianos, A. 2018. Ntua-slp at semeval-2018 task 1: Predicting affective content in tweets with deep attentive rnns and transfer learning. *arXiv preprint arXiv:1804.06658*.

[Cho and Hariharan 2019] Cho, J. H., and Hariharan, B. 2019. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4794–4802.

[Devlin et al. 2018] Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[Duppada, Jain, and Hiray 2018] Duppada, V.; Jain, R.; and Hiray, S. 2018. Seernet at semeval-2018 task 1: Domain adaptation for affect in tweets. *arXiv preprint arXiv:1804.06137*.

[Hinton, Vinyals, and Dean 2015] Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

[Mohammad et al. 2018] Mohammad, S.; Bravo-Marquez, F.; Salameh, M.; and Kiritchenko, S. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, 1–17.

[Nie et al. 2019] Nie, X.; Li, Y.; Luo, L.; Zhang, N.; and Feng, J. 2019. Dynamic kernel distillation for efficient pose estimation in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6942–6950.

[Radford et al. 2019] Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1(8):9.

[Rozental and Fleischer 2018] Rozental, A., and Fleischer, D. 2018. Amobee at semeval-2018 task 1: Gru neural network with a cnn attention mechanism for sentiment classification. *arXiv preprint arXiv:1804.04380*.

[Sanh et al. 2019] Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

[Shoeybi et al. 2019] Shoeybi, M.; Patwary, M.; Puri, R.; LeGresley, P.; Casper, J.; and Catanzaro, B. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

[Van Hee, Lefever, and Hoste 2018] Van Hee, C.; Lefever, E.; and Hoste, V. 2018. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, 39–50.

[Vu et al. 2018] Vu, T.; Nguyen, D. Q.; Vu, X.-S.; Nguyen, D. Q.; Catt, M.; and Trenell, M. 2018. Nihrio at semeval-2018 task 3: A simple and accurate neural network model for irony detection in twitter. *arXiv preprint arXiv:1804.00520*.

[Wang et al. 2018] Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

[Yang et al. 2019] Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.; and Le, Q. V. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.