Cross Modality Pre-training and the Intrinsic Values of Data Modality

Nick Wall
Southern Methodist University
walln@smu.edu

Andrew Havard
Southern Methodist University
ahavard@smu.edu

Jonathan Ebrahimian
Southern Methodist University
jebrahimian@smu.edu

***Abstract*—We evaluate the impact of different data modalities for pre-trained transformers to investigate the inherit value of different data types in large models. We find that given the correct situation, pre-trained models can sometimes be used to bootstrap effectively for certain tasks even when the pre-training modality is different than the downstream modality. Furthermore, it appears in our evaluations that natural language more consistently performs well across modalities. While this is not a surefire trick that always works or can always be useful, it is an interesting finding that certain model architectures can learn representations that may be useful across different data modalities.**

I. Introduction

Recently, large transformer models have seen an abundance of success when used for downstream classification or generation tasks of the same modality. We are interested in evaluating if large transformer models can be used for downstream tasks across different modalities. In other words, can a large language transformer model such as GPT2 that was trained on natural language tasks, be used to pre-train an image classifier? This question has been explored by Lu et al. [4], in their work: Transformers as Universal Computation Engines. We will be attempting different types of transfer learning to see if this is consistently reproducible and, if the modality really matters.

The value of pre-training transformers has already been made abundantly clear when fine-tuning a large transformer model to perform a downstream task for models trained on the same modality as the downstream task. This has become a common practice in the NLP community as larger and larger language models prove to generalize well to many different natural language tasks. Some examples of this are fine-tuning models such as T5 for text to SQL generation or using BERT for sentiment analysis. We want to explore if this same idea can be used across modalities to enable generalizable pre-training.

The hypothesized theory behind this technique is that for certain modalities, or potentially any modality, transformers can learn feature representations in the attention layers than can translate to other arbitrary downstream modalities. If the model can accurately learn these feature representations, it could potentially be used to bootstrap other models. This would make the case that the pre-trained models could potentially be used as a more effective initialization method regardless of the pre-training dataset.

We explore this idea by adding our own input and output layers to two different transformer models and testing different pre-training and fine-tuning strategies.

II. Background

In 2017, Vaswani et al. [1] published their work regarding transformer networks. The transformer network consists of an encoder and decoder stack, where a stack refers to a set of identical encoder or decoder layers. Both the encoder and decoder layers use multi-headed self-attention to allow the model to focus on certain combinations of data. Since the transformer is an auto-regressive model, meaning that it deals with sequences of data, this architecture allows it focus on certain combinations of data across the sequence of data (eg. different words in a sentence) and learn representations of that data in such a way that it can be interpreted downstream. With this architecture, Vaswani et al. achieved new state-of-the-art performances in language translation tasks, which lead to transformers' rise in popularity in the field of natural language processing.

In 2020, Qi et al. [2] introduced ImageBERT, a type of transformer which examines data across different modalities in order to try to understand the relationship between those modalities and make some sort of prediction based upon that relationship. In their case, they used a combination of images with corresponding text descriptions to try to predict a description of a given image. After fine-tuning their model, they achieved a new state-of-the-art performance for this task, showing promise for research involving cross-modality networks.

In 2022, Reid et al. [5] investigated whether they could pre-train a transformer on one modality to improve training speed in another. They used a pre-trained GPT2 model, a model trained on natural language, to bootstrap a reinforcement learning model to play several Atari games. They found that their pre-trained transformer model converged roughly 3-6 times faster than a different, state-of-the-art model trained from scratch, while still achieving similar performance. This research was similar to work done by Lu et al. [4] in 2021, in which they used a pre-trained GPT2 model to train for several tasks, including image classification, bit memory, and protein folding. They found that despite the difference in modality, they still saw advantages both computationally and in terms of performance from pre-training on a modality completely

separate from the given task, compared to an LSTM and fully retraining the model.

In 2022, Li et al. [3] published their research on BLIP, a vision transformer dedicated to both understanding information across modalities (in their case, images and text) and also generative tasks. They do this by training their vision transformer across multiple tasks: encoding across modalities, encoding only images, and decoding text. They found that by doing this, they were able to equal or outperform other state-of-the-art vision transformers in tasks between modalities such as image-to-text retrieval and image captioning with less training data. They also outperformed or closely equaled state-of-the-art models in tasks across modalities such as visual question answering (given an image and a question, supply an answer to the question).

III. Methodology

We evaluate whether different modalities of data have differing pre-training and bootstrapping capabilities when compared to each other through a series of tests. The main set of evaluations comes from comparing two different models that were trained on two different modalities. We chose two models, GPT2 and ViT-16-224k, for the following reasons:

- There is evidence from previous works that these models can be trained effectively across modalities.
- These models are small enough to fit within our training budgets.
- The models are not too dissimilar in architecture to where it may be a completely unfair comparison.

Ideally we would have used the same model trained on completely different modalities; however, we could not find quality transformer models that have been trained in such a way that also meet our other selection criteria.

We train each model type in 3 different ways:

1) We train with no pre-training at all, such that the weights are completely untrained.
2) We train the model with the pre-trained weights and all of the layers unfrozen, such that we are fully fine-tuning the model.
3) We train the model in the fashion outlined in [4] where only input, output, positional embeddings, and layer-norms are fine-tuned. This is hypothesized to preserve feature representations across modalities in attention layers where relations in arbitrary data sequences are learned.

For our experiments, we only train for 250 epochs because non-trivial datasets require an extremely long training duration with enormous computational requirements to saturate the large number of weights and fully train to convergence. As a result of this, the results do not display fully converged models. Significant emphasis should be placed on the training speed and starting performances instead of only end results.

We evaluated on three main datasets to explore different modalities and to inspect whether the relative difficulty of the task for the modality can have a significant impact. For these tests we selected MNIST and CIFAR10 as image-based datasets to specifically evaluate how different pre-training works with images, and how varying image classification difficulties perform. We also attempted a sentiment analysis task with the IMDB movie review dataset to evaluate an NLP task.

For all of our experiments we attempted to keep configuration and hyperparameters as consistent as possible to make fair comparisons, even at the cost of performance. Using better hyperparameters and configurations would surely create better results, but we want to be able to clearly identify where gains and losses come from. More specifically, we used an input layer that connects to an embedding of size 768 for all models, and an output layer that is based on the number of output classes. The models are all trained with Adam optimization and a patch size of 4. The learning rate is the only parameter that is tuned based on the methodology. This is because the different methodologies will have drastically different training objectives. For the models without pre-training we used a learning rate of 1×10^{-3}, with pre-training and partial freezing we used 1×10^{-5}, and a learning rate of 1×10^{-7} for the fully unfrozen pre-trained models. Each model is trained on a specific dataset with the same relative batch size for a given dataset, 100 steps per epoch, and 250 epochs total. Ideally we could run experiments across the same models with different pre-training and not have to worry about hyperparameters, but we prioritized consistency.

IV. Empirical Evaluation

In this section, we look to determine whether benefits from pre-training exist, to what extent, and if they can be preserved across modalities.

A. Final Training Performance

After completing the training for 250 epochs for each model and dataset we can see some interesting results in Table 1. Pre-training clearly has some benefits when doing short term training on a simple dataset even when the pre-training modality is completely different. While the difference between the two models is small in all cases, it is clear that when you compare the training methodologies the results are dramatic. It should be noted however, that when training for significantly longer all of these results will eventually converge and get extremely high performance given how trivial the task of MNIST is. Nonetheless, pre-training here seems to be a decent way of bootstrapping this kind of task.

Moving on to looking at CIFAR at the same time frame, we can see some more variation. Clearly the more challenging task is going to take significantly longer for the model to saturate and converge so these results are fairly unimpressive. However, we can still see that pre-training looks like it has a significant leg up on the other training methodologies; however, it would stand to reason that such results are somewhat meaningless when the peak accuracy is so poor. Future work would likely benefit from evaluating much smaller transformer models such

TABLE 1
MNIST FINAL TRAINING PERFORMANCE

Model Type	Training Methodology		
	Pre-trained and Partially Frozen	***Pre-trained and Fully Unfrozen***	***No Pre-training***
GPT2	74%	33%	53%
ViT	69%	34%	46%

TABLE 2
CIFAR10 FINAL TRAINING PERFORMANCE

Model Type	Training Methodology		
	Pre-trained and Partially Frozen	***Pre-trained and Fully Unfrozen***	***No Pre-training***
GPT2	28%	13%	19%
ViT	26%	19%	19%

TABLE 3
IMDB FINAL TRAINING PERFORMANCE

Model Type	Training Methodology		
	Pre-trained and Partially Frozen	***Pre-trained and Fully Unfrozen***	***No Pre-training***
GPT2	52%	49%	52%
ViT	49%	53%	51%

as minGPT to determine how parameter scaling is affected by cross-modality pre-training.

The last dataset to evaluate is IMDB movie reviews. For this task we are classifying whether movie reviews are positive or negative based on natural language. This task, in our opinion, is significantly more complex than the others. However, it is challenging to curate a fair example across domains. There are no NLP tasks that are comparably trivial when compared to MNIST or potentially even CIFAR10. Regardless, we present these results in Table 3 to indicate how the methodologies we explored are not without significant fault. Clearly none of the models have learned enough to be fairly evaluated as none of them are significantly better than random choice. It seems that in more challenging tasks, there is a less significant, if any, performance boost in bootstrapping the model. It would be interesting to reexamine this after many thousands of epochs to see if it is possible to get the Vision Transformer to learn this downstream task.

Interestingly, across the board, if the model can start to effectively learn in this short time span, the pre-training has a significant boost in performance. The pre-training modality is less significant than we would expect as well. GPT2 seems to hold up fairly competitively and is even better in certain cases than the Vision Transformer in the image-classification tasks.

B. Bootstrapping Performance

As suggested earlier, another evaluation criteria that may give insight into the potential benefits of pre-training is the model's performance immediately and earlier into the training. The question really is, does pre-training give the model a head start even when the downstream modality is different? This effect can be measured at different points in time but for the sake of this research we will be evaluating the performance within 25 epochs. We also exclude the IMDB dataset from this evaluation as we have already discussed that nothing interesting was learned in this short training duration, and so there are no interesting insights to extract from the early performance on this task.

This cutoff for evaluating early performance was selected because it should be early enough in the training to display gains that are primarily driven by the starting values of the weights without giving too much of an advantage to shared modality experiments where we would expect better performance in the first handful of epochs.

Looking at the results in Tables 4 and 5, we can see that pre-training can be largely hit or miss. We would expect a significant boost in pre-training performance for the vision transformer for both tasks. However, the results in MNIST and CIFAR10 contrast starkly. In the MNIST experiment we can see a significant early performance gain for the vision transformer, but in the CIFAR experiment the gap is much smaller. We hypothesize that in more complex downstream tasks, models cannot efficiently fine-tune large parameter counts, due to an increased total modification to the base weights being required.

While the initial gains are not what might be expected, how is it possible that in some cases the pre-training results in better final results? It seems as though given a bit of time to start fitting to the data, models learn to fit around the existing weights rather than just fitting to the task like would be expected when training from scratch. We suppose this because the input and output layers will always start as complete noise since they have never been pre-trained, so the first few epochs saturate these layers and then the model begins to tune around the contrast between the existing and new layers. This theory is not fully supported by our testing and would require further research, but the idea that arbitrary feature representations could potentially be across tasks and

TABLE 4
MNIST EARLY TRAINING PERFORMANCE

Model Type	Training Methodology		
	Pre-trained and Partially Frozen	***Pre-trained and Fully Unfrozen***	***No Pre-training***
GPT2	22%	15%	32%
ViT	53%	11%	27%

TABLE 5
CIFAR10 EARLY TRAINING PERFORMANCE

Model Type	Training Methodology		
	Pre-trained and Partially Frozen	***Pre-trained and Fully Unfrozen***	***No Pre-training***
GPT2	13%	10%	17%
ViT	15%	12%	20%

modalities is interesting.

C. Training Speedups

The pre-training with partial freezing that applies the methodology of the Frozen Pre-trained Transformer [4] has some benefits outside of potential bootstrapping for early improved performance on downstream tasks. Since only a fraction of the total parameters are receiving updates during the fine-tuning process, the training is significantly faster. This effect only increases should we scale to larger and larger models. As compute demands have grown dramatically in recent years to be able to leverage increasingly large models, this technique enables faster training with less total compute. If the task can be effectively bootstrapped with pre-training it may prove to be beneficial to try the largest possible model regardless of the modality of the data that the model was pre-trained on. If this scenario is viable for a given task it can enable more efficient use of compute resources and energy, as well as more centralized model usage which results in auxiliary benefits such as improved tooling and infrastructure. This is a driving motivation of this research. Should techniques to transfer large transformers across tasks and modalities display promise in reducing compute and energy needs, then the centralization of effort into creating general-purpose large transformers to use as pre-training for downstream tasks could potentially be a method for universal models. For this to be viable, two conditions must be met. Firstly, the performance on the tasks need to be high, which our results are inconclusive towards. Additionally, the training requirements should be low.

TABLE 6
TRAINING DURATION (SECONDS/EPOCH)

Method	*Mean*	*Standard Deviation*
Pre-trained Partially Frozen	4.946	1.442
Pre-trained Fully Unfrozen	8.123	2.862
No Pre-training	8.395	3.742

Looking at the training times for different tasks in Table 6, we can see that pre-training with frozen attention and feed-forward layers results in pretty dramatic reductions in training time per epoch. The potential to gain a near 50% speedup in training time per epoch is very enticing. Additionally, the time per epoch becomes more stable as the number of parameters receiving updates during training decreases.

V. CONCLUSION

Our experimentation resulted in some fairly inconclusive results. While there were some promising results showing that it is sometimes possible to gain an advantage when pre-training large models regardless of the pre-training modality, we had lots of conflicting results that make it impossible to claim that this method works or is consistent across all or many tasks and pre-training modalities. Nonetheless, we suggest that it may be worthwhile evaluating these strategies for different architectures and datasets to see if there can be any benefits gained.

Moving forward, we would continue this research by evaluating on a larger variety of data and model architectures and experimenting by actually evaluating each model with more specific hyperparameters that encourage success for the specific model, rather than just enforcing consistency for the sake of fairness. Another potential avenue that could yield interesting results is the exploration of fusion models that are pre-trained on multiple modalities at once as a baseline model. As these models, such as those outlined by Li et al. [3], learn feature representations that are more flexible, we could potentially see greater pre-training potential and more universal models.

ACKNOWLEDGMENT

We would like to thank Dr. Larson for his guidance and encouragement, and Clay Harper for his insights.

REFERENCES

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. arXiv preprint arXiv:1706.03762, 2017.

[2] Di Qi, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. Imagebert: Cross-modal pre-training with large-scale

[3] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. arXiv preprint arXiv:2201.12086, 2022. weak-supervised image-text data. arXiv preprint arXiv:2001.07966, 2020.

[4] Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. arXiv preprint arXiv:2103.05247, 2022.
[5] Machel Reid, Yutaro Yamada, and Shixiang Shane Gu. Can wikipedia help offline reinforcement learning?. arXiv preprint arXiv:2201.12122, 2022.