# GPTology: The Impact of Fine-Tuning on the Geometry of GPT-2

Anonymous ACL submission


## Abstract

Although transformer decoders are quickly becoming the most prominent NLP models, little is known about how they embed text in vector space and make decisions on downstream tasks. In this study, we evaluate the impact of fine-tuning on *how* GPT-2 represents text in vector space. In particular, we demonstrate that fine-tuning refines the last half of the network, and that task specific information is encoded into what the literature refers to as "rogue dimensions". In contrast to previous work, we find that rogue dimensions that emerge when fine-tuning GPT-2 are influential to the model decision making process. By using a linear threshold on a single rogue dimension in space, we can complete downstream classification tasks with an error of 1.6% relative to the full 768-dimensional representations of GPT-2.


## 1 Introduction

Several studies have been dedicated to understanding what types of knowledge are encoded in BERT (Devlin et al., 2018) embeddings, from discovering patterns in attention matrices to demonstrating that BERT embeddings naturally perform word sense disambiguation (Rogers et al., 2020; Mickus et al., 2019; Kovaleva et al., 2019; Coenen et al., 2019). However, there have been far fewer studies investigating transformer-decoder-based models, such as GPT-1,2,3 (Radford et al., 2018, 2019; Brown et al., 2020). Previous studies examining the GPT-x family of models typically focus on bias contained in short passages produced by a language model (Bender et al., 2021; Bordia and Bowman, 2019), or on how small perturbations to input text can cause the quality of the output text to quickly degrade (Heidenreich and Williams, 2021).

Thus far, studies examining GPT-2 fail to investigate how the model embeds text in vector space. Further, there is a lack of literature on what features of the embedding space are important in determining how GPT-2 makes decisions when fine-tuned to complete a downstream task. In this paper, we examine: 1) the impact of fine-tuning on GPT-2 sentence embeddings and; 2) where task specific information is encoded during the process of fine-tuning. The contributions of this study are as follows:

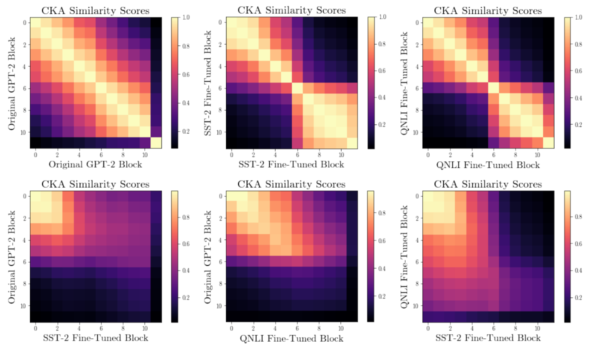


Figure 1: CKA similarity scores among fine-tuned SST-2 & QNLI GPT-2 models and the original GPT-2 model.

- Using Centered Kernel Alignment, we demonstrate that fine-tuning gives rise to a "bow-tie" pattern among decoder blocks where the last 6 decoder blocks specialize on the given tasks.
- We find that rogue dimensions emerge in the same location when fine-tuning for different tasks, and encode task specific knowledge.
- By comparing representations of fine-tuned GPT-2 and BERT, we show that rogue dimensions do not encode task specific information to the same degree in all models [1].

## 2 Distribution of Information Over Decoder Blocks

### 2.1 Methods & Related Works

We examine how GPT-2 representations change as a result of fine-tuning by 1) computing centered

[1]Program code is publicly available at: *Removed for anonymous review*

kernel alignment (CKA) of activations for each decoder-block; 2) visualizing sentence embeddings using t-SNE and; 3) exploring "outlier" (Kovaleva et al., 2021) or "rogue dimensions" Timkey and van Schijndel (2021) that exhibit high levels of variance compared to the rest of the vector space. We fine-tune GPT-2 on two GLUE tasks: SST-2 (Socher et al., 2013) and QNLI (Wang et al., 2018). SST-2 contains short movie reviews that a model must label as either positive or negative. QNLI tasks models to determine whether or not a given answer can be entailed from specified question. In both cases, we fine-tune the model for 10 epochs and achieve an accuracy of 92.8% and 88.2% on the hidden validation data for SST-2 and QNLI, respectively.

Intuitively, CKA is a dot-product-based, model agnostic tool that measures how similar representations are across different layers or networks (Kornblith et al., 2019). A CKA score of 0 indicates that representations are independent, while a score of 1 implies perfect correlation. Formally, CKA is based on the Hilbert Schmidt Independence Criterion (HSCI) (Gretton et al., 2005), which computes the square of the Frobenius norm between the cross-covariance matrix of two Gram matrices.

Previous works have used CKA to compare the outputs of layers in ViTs and CNNs to provide insights as to whether these two models learn significantly different representations for a given input image (Raghu et al., 2021). However, CKA analysis has not yet been applied to study the impact of fine-tuning language models. We compute CKA scores to evaluate the impact of fine-tuning on GPT-2 representations on both SST-2 and QNLI. Note that, to more easily interpret model outputs, we only compute CKA for activation maps on decoder blocks instead of every layer in the network. We compute CKA scores for each model on the hidden validation data for the respective task the models are fine-tuned on, and compare representations to a pre-trained GPT-2.

The literature overwhelmingly agrees that contextualized embedding models are anisotropic, meaning that they do not uniformly utilize the vector space they occupy (Ethayarajh, 2019; Rudman et al., 2022; Cai et al., 2021). Anisotropy in point clouds induced by contextualized embedding models stems from "rogue dimensions" that exhibit high levels of variance relative to other dimensions in space and dominate model representations

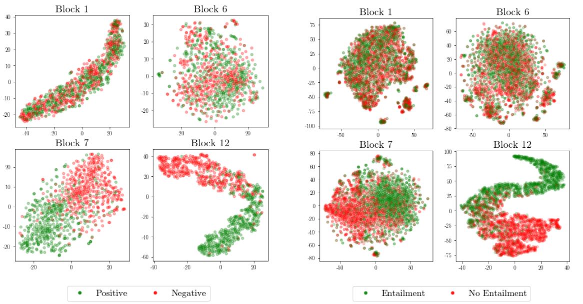


Figure 2: Last token t-SNE embeddings for fine-tuned SST-2 & QNLI GPT-2 models, respectively.

(Timkey and van Schijndel, 2021). In this study, we examine the impact of fine-tuning on rogue dimensions and characterize their role in the model's downstream decision making process. We visualize the impact of rogue dimensions by plotting the dimension index on the $x$-axis and the value of the specific dimension on the $y$-axis.

## 2.2 Results

### 2.2.1 Locality of Information

Computing CKA scores for GPT-2 provides us with a baseline of model behavior. In the original GPT-2 model, we see a block diagonal structure where early network layers are similar to one another, middle layers are similar to one another and the final layer is distinct from all other layers in the network (Figure 1). Fine-tuning GPT-2 causes the emergence of a bow-tie pattern in CKA matrices where the first 6 decoder blocks are similar to one another and the last 6 decoder blocks are similar to one another. We find that layers 7-12 produce similar activations to one another as they begin to encode task-specific knowledge. Figure 2 shows that, while none of the first 6 decoder blocks in the fine-tuned GPT-2 are able to separate input texts, layers 7-12 have clearly learned distinct subspaces that separate points by class label.

Previous work has used probing methods to argue that the process of fine-tuning encoder models primarily specializes the last few layers of the network (Merchant et al., 2020). Figure 1 empirically supports this intuition for transformer decoders. However, our results show that the process of fine-tuning in GPT-2 has a significant impact, not only on the last, but also on intermediate network layers which have been thought to be the "most transferable" for different tasks in BERT (Kovaleva et al., 2019). Figure 1 shows that the first 3 layers in all three models considered in this study exhibit CKA

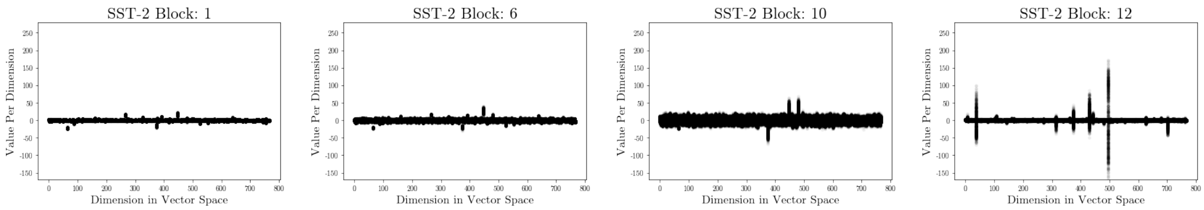


Figure 3: We visualize rogue dimensions for last-token representations across decoder blocks on the SST-2 validation data after fine-tuning. The horizontal axis tracks the dimension's index and the vertical tracks the value in the given dimension. The rogue dimensions can be clearly seen as "spikes" in the graph.

scores near 1, demonstrating that information in the first 3 decoder blocks is preserved across all fine-tuning tasks.

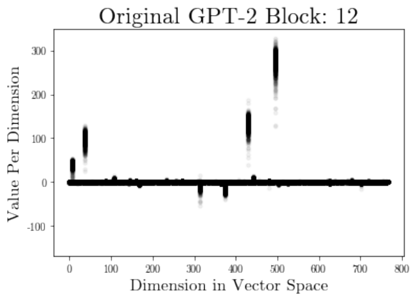


Figure 4: Rogue dimensions of GPT-2 on QNLI with no fine-tuning.

#### 2.2.2 Rogue Dimensions

We extend the understanding in the literature on rogue dimensions in several ways: 1) rogue dimensions emerge in later blocks of the network; 2) fine-tuning exacerbates existing rogue dimensions; 3) the same dimensions dominate the vector space in SST-2 and QNLI fine-tuned models; and 4) rogue dimensions encode far more class specific information in GPT-2 than BERT.

In Figure 3, we visualize how rogue dimensions change/emerge over time. Representations from earlier decoder blocks do not exhibit any prominent dimensions that deviate significantly from the distribution mean or exhibit exceedingly high variance. However, as we progress further through the network, the last token representations become dominated by rogue dimensions. In both the SST-2 and QNLI fine-tuned models, variance in the most prominent rogue dimensions increases. However, the mean in these dimensions is much closer to zero in the fine-tuned models compared to the pre-trained GPT-2 representations, as shown in Figure 4. Remarkably, fine-tuning impacts the same dimensions for GPT-2 in both SST-2 and QNLI. Eight of the top ten rogue dimensions are the same in both fine-tuned models.

Several authors have argued that the presence of anisotropy in the form of rogue dimensions is detrimental to model performance, and that by removing or mitigating rogue dimensions, we can improve performance on downstream tasks (Mu et al., 2017; Zhou et al., 2020; Timkey and van Schijndel, 2021; Liang et al., 2021; Zhang et al., 2022). However, studies examining the impact of rogue dimensions on model performance tend to focus either on static word embeddings or transformer encoders, such as BERT. In contrast to previous works that argue rogue dimensions "disrupt" model representations (Kovaleva et al., 2021), we find that rogue dimensions encode crucial task specific information in GPT-2. Further, Figure 5 shows that while class specific information is concentrated in rogue dimensions in GPT-2, task specific information is distributed across multiple dimensions in BERT.

## 3 Locality of Task-Specific Information

### 3.1 Methods

The purpose of this section is to determine where in the model task-specific information is encoded during the process of fine-tuning. We first compute what we refer to as the *principal* rogue dimension in space, i.e., the single dimension with the highest variance. Next, we use a simple linear 1-D SVM to find the optimal threshold value that linearly separates classes in the principal rogue dimension on the training data. We then make predictions for both SST-2 and QNLI based solely on the value of the principal rogue dimension on the hidden validation data for GPT-2 and BERT.

Additionally, we conduct a simple ablation experiment to determine how class specific information is distributed across multiple dimensions in GPT-2. Following a similar ablation strategy to Kovaleva et al. (2021), we ablate a dimension by setting the representations of GPT-2 in a given dimension to zero. Removing the $k$-bottom/top dimensions equates ablating the $k$-dimensions with the lowest/highest variance in the embedding space of GPT-2. After ablating the specified dimensions, we input the ablated representations of GPT-2 into the trained linear classification head for each task and evaluate performance.

### 3.2 Results

Using a simple linear threshold, we can predict the sentiment of a given input text in SST-2 with 91.3% accuracy (compared to 92.8% for the full model) and achieve a QNLI accuracy of 86.6% (compared to 88.2% for the full model). We re-run our threshold experiment on BERT fine-tuned on QNLI and SST-2 and find that rogue dimensions in transformer encoders do not encode task-specific information. The optimal decision boundary in the principal rogue dimension for BERT yields a mere 76.03% accuracy on SST-2 (compared to a 92.22% using the full representations) and 81.9% (compared to 89.69% using the full representations).

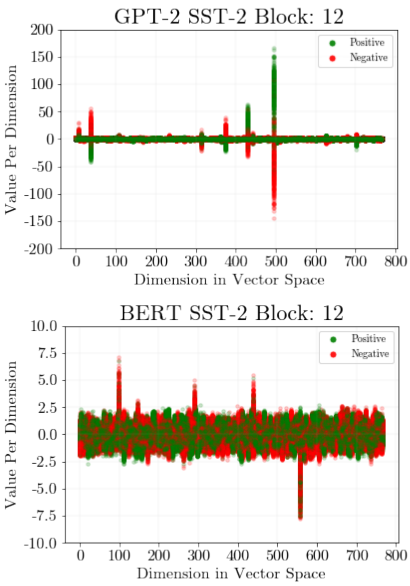


Figure 5: SST-2 sentence embedding representations from decoder block 12 for GPT-2 and BERT.

Figure 6 shows that ablating the 765 dimensions with the smallest variance minimally decreases accuracy. On QNLI, performance abruptly drops from ≈85% to ≈50% when we ablate all except the top 3 dimensions. We posit the classification head has learned to rely on information from the top 3 dimension, since QNLI is an inherently more difficult task than SST-2. Although model performance minimally increases when removing the top 92 dimensions on SST-2, performance quickly decays if we ablate more than 300 dimensions. This finding indicates that class specific information is stored in less than half of the top dimensions after fine-tuning GPT-2. Further, on QNLI, accuracy steadily decreases as we remove top dimensions.

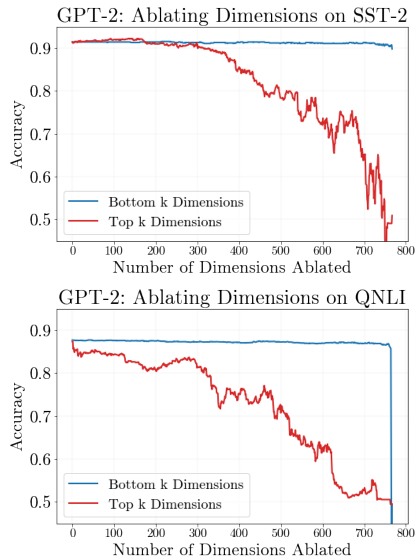


Figure 6: Performance after ablating dimensions from sentence embeddings in GPT-2.

## 4 Conclusions & Future Works

This paper examines the impact of fine-tuning on GPT-2 embeddings. Bow-tie patterns in CKA similarity heat maps demonstrate that fine-tuning specializes the last half of the network to adapt to a given task. We find that task specific knowledge acquired during the process of fine-tuning is encoded into what the literature refers to as *rogue dimensions*. In contrast to prior studies, we demonstrate that ablating rogue dimensions removes task specific information and can hurt model performance.

There are many promising directions for future work. Several studies have suggested that rogue dimensions may be detrimental for model performance. However, we posit that encouraging the formation of rogue dimensions may be beneficial for transformer decoder models. Given that the largest transformer decoder models rely on prompts, we will further examine how our methods can be applied to understand why certain prompts condition a model to perform well on few-shot tasks. We hope that this study will encourage other researchers to examine transformer-decoder architectures and give a more complete understanding of how these models represent text in space.

## 5 Limitations & Ethical Considerations

Although our study provides key insights on the impact that fine-tuning has on how GPT-2 represents text in space, there are several limitations. Firstly, increasingly large language models such as GPT-3 (Brown et al., 2020), Megatron-Turing NLG (Smith et al., 2022) and PALM (Chowdhery et al., 2022) have surpassed the capabilities of GPT-2 in recent years. Our methods can easily be adapted to larger, more advanced models, however, we are forced to restrict our analysis to GPT-2 given that the weights of these models are not publicly available. Secondly, we only analyze the impact that fine-tuning has on GPT-2 for classification tasks and not for the more common applications of transformer decoders such as natural language generation. Even though fine-tuning for classification tasks is less common for transformer decoders, our fine-tuned GPT-2 models are competitive with early transformer encoder models such as BERT. Thus, it is worth studying how transformer decoder models adapt when fine-tuned for classification tasks. Lastly, we restrict analysis to a single model: GPT-2. Our methodology can be applied to any transformer decoder and can be easily adapted to transformer encoders (by analyzing CLS tokens instead of last token representations). Future work should consider the presence of rogue dimensions in more advanced transformer encoder models such as RoBERTa (Liu et al., 2019) or sequence-to-sequence architectures such as T5 (Raffel et al., 2019).

## References


Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Shikha Bordia and Samuel R. Bowman. 2019. Identifying and reducing gender bias in word-level language models. *CoRR*, abs/1904.03035.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2021. Isotropy in the contextual embedding space: Clusters and manifolds. In *International Conference on Learning Representations*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. 2019. Visualizing and measuring the geometry of bert.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and GPT-2 embeddings. *CoRR*, abs/1909.00512.

Arthur Gretton, Olivier Bousquet, Alexander Smola, and Bernhard Schölkopf. 2005. Measuring statistical dependence with hilbert-schmidt norms. volume 3734.

Hunter Heidenreich and Jake Williams. 2021. The earth is flat and the sun is not a star: The susceptibility of gpt-2 to universal adversarial triggers. pages 566–573.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. 2019. Similarity of neural network representations revisited. *CoRR*, abs/1905.00414.

Olga Kovaleva, Saurabh Kulshreshtha, Anna Rogers, and Anna Rumshisky. 2021. BERT busters: Outlier dimensions that disrupt transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3392–3405, Online. Association for Computational Linguistics.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. *CoRR*, abs/1908.08593.

Yuxin Liang, Rui Cao, Jie Zheng, Jie Ren, and Ling Gao. 2021. Learning to remove: Towards isotropic pre-trained BERT embedding. *CoRR*, abs/2104.05274.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? *CoRR*, abs/2004.14448.

Timothee Mickus, Denis Paperno, Mathieu Constant, and Kees van Deemter. 2019. What do you mean, bert? assessing BERT as a distributional semantics model. *CoRR*, abs/1911.05758.

Jiaqi Mu, Suma Bhat, and Pramod Viswanath. 2017. All-but-the-top: Simple and effective postprocessing for word representations. *CoRR*, abs/1702.01417.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. 2021. Do vision transformers see like convolutional neural networks? *CoRR*, abs/2108.08810.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how BERT works. *CoRR*, abs/2002.12327.

William Rudman, Nate Gillman, Taylor Rayne, and Carsten Eickhoff. 2022. IsoScore: Measuring the uniformity of embedding space utilization. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3325–3339, Dublin, Ireland. Association for Computational Linguistics.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zheng, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. 2022. Using deepspeed and megatron to train megatron-turing NLG 530b, A large-scale generative language model. *CoRR*, abs/2201.11990.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

William Timkey and Marten van Schijndel. 2021. All bark and no bite: Rogue dimensions in transformer language models obscure representational quality. *CoRR*, abs/2109.04404.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Haode Zhang, Haowen Liang, Yuwei Zhang, Liming Zhan, Xiao-Ming Wu, Xiaolei Lu, and Albert Y. S. Lam. 2022. Fine-tuning pre-trained language models for few-shot intent detection: Supervised pre-training and isotropization.

Wenxuan Zhou, Bill Yuchen Lin, and Xiang Ren. 2020. Isobn: Fine-tuning BERT with isotropic batch normalization. *CoRR*, abs/2005.02178.

## A Model Hyperparameters and Training Details

In this section, we detail all model hyperparameters and expected training times. We used the HuggingFace implementations of GPT2ForSequenceClassification and BERTForSequenceClassification to conduct experiments. As the purpose of this paper is focused on analyzing model representations, we perform no hyperparameter sweeps and report results on a single run of the model. In order to speed up training we use gradient accumulation with a batch size of 32 and an accumulation step of 4. This creates an effective batch of 128. Fine-tuning GPT-2 and BERT took less than an hour for SST-2 and took less than 2 hours for QNLI.