Assessing the Macro and Micro Effects of Random Seeds on Fine-Tuning Large Language Models

Nghia T. Bui[1], **Guergana Savova**[2], **Lijing Wang**[1]
[1]New Jersey Institute of Technology
{ntb23, lijing.wang}@njit.edu
[2]Boston Children's Hospital and Harvard Medical School
guergana.savova@childrens.harvard.edu

Abstract

The impact of random seeds in fine-tuning large language models (LLMs) has been largely overlooked despite its potential influence on model performance. In this study, we systematically evaluate the effects of random seeds on LLMs using the GLUE and SuperGLUE benchmarks. We analyze the macro impact through traditional metrics like accuracy and F1, calculating their mean and variance to quantify performance fluctuations. To capture the micro effects, we introduce a novel metric, *consistency*, measuring the stability of individual predictions across runs. Our experiments reveal significant variance at both macro and micro levels, underscoring the need for careful consideration of random seeds in fine-tuning and evaluation.

1 Introduction

The impact of random seeds in neural network training has long been recognized across various domains, such as general machine learning classification and regression tasks (Ganesh et al., 2023) (Madhyastha and Jain, 2019), computer vision (Picard, 2021) (Åkesson et al., 2024), natural language processing (NLP)(Bethard, 2022),(Lucic et al., 2022).

In the field of NLP, large language models (LLMs) have achieved state-of-the-art results on benchmarks like GLUE and SuperGLUE, which are now standard for evaluating language understanding and reasoning. However, pretrained transformers such as BERT (Devlin et al., 2019) and RoBERTa (Liu, 2019) are highly sensitive to random seeds (Risch and Krestel, 2020; Dodge et al., 2020; Mosbach et al., 2020), often leading to significant performance variation that complicates experimental interpretation and benchmarking. While other sources of randomness, such as prompt formatting (He et al., 2024), in-context example selection (Gupta et al., 2023), and how learnable weights are initialized (Hayou et al., 2024), have also been explored, seed variation remains a fundamental and underaddressed issue. A recent analysis of 85 papers from the ACL Anthology (Bethard, 2022) revealed risky practices in the use of random seeds: over 50% of the papers exhibited potential misuse, with 24 using a single fixed random seed. *This highlights that random seeds sensitivity in LLM fine-tuning remains insufficiently understood, motivating the need for more systematic investigation.*

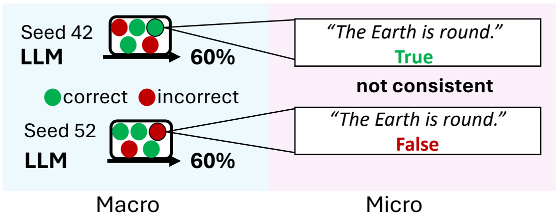

Figure 1: Macro and micro performance. A pretrained LLM is fine-tuned with random seed 42 and 52. The accuracy for both models is 60%, but the overlapping of individual predictions is 20%.

Existing studies examining the impact of random seeds (Ganesh et al., 2023; Madhyastha and Jain, 2019; Picard, 2021) typically evaluate performance variations by measuring the variance of standard metrics, such as accuracy score for classification tasks, or MAE (mean absolute error) for regression tasks, across multiple seeds. These evaluations focus on the **macro** agreement of model performance across the entire test set, offering insights into overall variability. However, *they overlook the **micro** impact of how individual test points are influenced by seed-induced variability*. As shown in Figure 1, model performance is robust to random seed 42 and 52 at the macro level (both achieve 60% accuracy) but lacks consistency at the micro level (only 20% overlapping predictions). This micro inconsistency can have severe consequences in real-world applications, especially in fields where model predic-

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 41–46
December 20-24, 2025 ©2025 Association for Computational Linguistics

tions are highly sensitive to individual test points, such as medical diagnosis and autonomous driving. Understanding this micro effect is crucial for assessing model robustness at the level of individual predictions, ensuring that specific test samples are not inconsistently predicted due to seed-induced variability. Additionally, it helps pinpoint specific areas where models may exhibit significant instability, such as consistently misclassifying certain types of data points or showing highly variable predictions for similar inputs. Recognizing these areas of instability can guide targeted improvements in both model design and evaluation practices, ensuring that assessments account for seed-induced variability in performance.

Major contributions: To address these gaps, in this work, (1) we analyze the impact of random seeds on pretrained LLMs using the GLUE and SuperGLUE benchmarks, covering both macro and micro variability; (2) We introduce a novel **consistency** metric to assess prediction stability on individual test points, capturing the micro effects of random seeds; (3) Our extensive experiments reveal significant variability in both standard and consistency metrics, underscoring the need to consider seed-induced variability in fine-tuning and evaluation, and incorporate random seeds sensitivity into benchmarking and reporting for more reliable and reproducible results.

2 Macro Metric: Variance

To measure the macro impact of random seeds on LLM performance, we calculate the variance of a standard evaluation metric across multiple seeds. Let $[\zeta_1, \cdots, \zeta_S]$ represent the values of model performance for an LLM fine-tuned with S random seeds, the *variance* is calculated by:

$$\text{VAR}(\zeta) = \sqrt{\frac{1}{S}\sum_{i=1}^{S}(\zeta_i - \bar{\zeta})^2} \qquad (1)$$

where $\bar{\zeta} = \frac{1}{S}\sum_{i=1}^{S}\zeta_i$. ζ can be any standard metrics, such as accuracy score for classification tasks or MAE for regression tasks. A smaller VAR indicates less variation in macro performance.

3 Micro Metric: Consistency

'Consistency' can have varying definitions across domains. Building on prior work, Wang et al. (2020) formally defined the *consistency* of a deep learning model as its ability to produce consistent predictions for the same input when periodically retrained with streaming data in deployment settings. Extending this idea, we define the *consistency* of an LLM as its ability to generate consistent predictions for the same input across models fine-tuned with different hyperparameter settings, with *correct-consistency* further specifying its ability to make consistent correct predictions in this context.

More specifically, consider two LLMs A and B, given a dataset $\mathcal{D} = d_1, \cdots, d_N$ of N data points, y_i^A and y_i^B are the prediction of A,B for a data point d_i with ground truth r_i. The *consistency* of A and B can be calculated as follows:

$$\text{CON:} \frac{1}{N}\sum_{t=1}^{N}\pi_{A,B}(t) \qquad (2)$$

where $\pi_{A,B}(\cdot)$ is the scoring function that quantifies the alignment between predictions y_t^A and y_t^B, with higher values indicating smaller variations in micro-level predictions; it can be either binary (e.g., 1 for match, 0 for mismatch) or probabilistic based on different NLP tasks. And the *correct-consistency* is calculated by:

$$\text{CCON:} \frac{1}{N}\sum_{t=1}^{N}\pi_{A,B,r}(t) \qquad (3)$$

where $\pi_{A,B,r}(\cdot)$ is the scoring function that quantifies the alignment between predictions y_t^A, y_t^B, and ground truth r_t.

In this paper, we focus solely on classification tasks because the GLUE and SuperGLUE benchmarks primarily consist of classification problems. The scoring function $\pi(\cdot)$ is defined as an indicator function that equals 1 if $y_t^A = y_t^B (= r_t)$, otherwise 0. We summarize the standard metric ζ and the possible corresponding scoring functions π for various NLP tasks in Table 2 in the Appendix A.1. We hope this summary offers useful context for interpreting our results and supports future extensions of our evaluation to a broader range of NLP tasks.

While consistency metrics can generally be used for quantifying the agreement of individual predictions from any two LLMs with different architectures, hyperparameters, or training settings, in our study, they are specifically used to serve as metrics to evaluate the micro impact of random seeds on the same pretrained LLM.

RoBERTa-large							Llama3.2-3B						
GLUE	**MRPC**	**QNLI**	**QQP**	**SST2**	**RTEG**	**CoLA**	GLUE	**MRPC**	**QNLI**	**QQP**	**SST2**	**RTEG**	**CoLA**
ζ	90.34	94.00	92.00	95.59	85.02	65.61	ζ	84.02	94.20	89.37	96.78	85.92	61.33
VAR	0.89	0.38	0.06	0.55	1.48	1.32	VAR	0.56	0.14	0.09	0.27	3.34	0.88
CON	92.89	95.64	95.57	96.83	91.09	93.95	CON	90.34	96.85	96.63	98.32	88.80	94.86
CCON	86.79	91.80	89.79	94.01	80.55	82.76	CCON	79.19	92.63	87.69	95.95	80.32	81.36
SuperGLUE	**BoolQ**	**CB**	**RTES**	**MultiRC**	**WiC**	**COPA**	SuperGLUE	**BoolQ**	**CB**	**RTES**	**MultiRC**	**WiC**	**COPA**
ζ	83.05	94.64	72.89	76.46	68.57	73.00	ζ	72.49	73.92	68.66	80.14	68.71	84.20
VAR	7.35	3.95	16.67	13.30	2.77	16.83	VAR	1.62	2.03	2.77	0.57	1.58	5.17
CON	86.79	91.50	70.05	74.63	80.36	64.88	CON	81.10	89.82	71.69	86.73	85.86	86.8
CCON	76.37	89.84	64.58	63.77	58.75	55.44	CCON	63.09	70.35	54.51	73.50	61.64	77.6

Table 1: Macro and micro impact of ten random seeds. ζ is the average of 10 values, which is MCC for CoLA and accuracy for the other tasks. VAR is the variance of ζ calculated using Equation 1. CON and CCON are the average of 45 consistency values calculated using Equation 2 and 3. ζ, CON, and CCON are expressed as percentages.

4 Experimental Setup

4.1 Benchmarks and pretrained models

In this study, we conduct experiments on a range of NLP tasks including CoLA, SST2, MRPC, QQP, QNLI, and RTE from GLUE (Wang et al., 2018) benchmark; RTE, CB, WiC, BoolQ, MultiRC, and COPA from SuperGLUE (Wang et al., 2019) benchmark. STSB, WSC, and MNLI tasks are omitted from our experiment. We specify the reason in Section A.4. All tasks use accuracy as the standard evaluation metric ζ, except for CoLA, which uses Matthews Correlation Coefficient (MCC). In our paper, we use RTEG to denote RTE task from GLUE and RTES for SuperGLUE.

To examine the effects on various scales of LLMs, we experiment with RoBERTa-large (~350M trainable parameters), as well as a larger LLM Llama3.2-3B (~3.21B trainable parameters) using LoRA (Hu et al., 2022) fine-tuning, enabling us to assess whether our findings generalize across model scales.

4.2 Settings

Our experiment is implemented using Hugging Face Transformers (v4.30.0) and PyTorch (v2.0), conducted on NVIDIA two A100 GPUs with 80GB of memory each. Based on the empirical findings in (Wang et al., 2023; Dodge et al., 2020; Mosbach et al., 2020), 5–10 seeds are sufficient to estimate variance of LLMs in NLP tasks. We perform full fine-tuning for each task with ten randomly chosen seeds: 42, 52, 62, 72, 82, 92, 102, 112, 122, 132 (i.e., $S = 10$). We calculate CON and CCON on each unique pair of seeds and report the average of 45 values as the final consistency score. Our fine-tuning process is based on the PyTorch script `run_glue.py`, and the best previously reported settings were applied unless otherwise specified.

To ensure proper experimental setup and reproducibility, we refer the configurations in (Liu, 2019) and replicate state-of-the-art (SOTA) scores reported using RoBERTa-large with full fine-tuning. A comparison of our implementation with the reference SOTA scores and detailed data and learning settings are provided in Appendix Table 3, Table 4, and Table 5. Since we could not find reference configurations or performance reports for Llama3.2-3B on the two benchmarks, we use the SOTA scores (Table 3) as a reference and conduct experiments using our own settings, detailed in Table 6.

5 Results and Discussion

5.1 Macro impact

Table 1 presents the averaged accuracy (ζ) and variance (VAR) for all tasks across ten random seeds using two LLMs. Significant VAR is observed in many tasks using RoBERTa-large, such as RTES (16.67), COPA (16.83), and MultiRC (13.30), reflecting sensitivity to random seed selection. High variability at the macro level undermines the reliability of single-seed evaluations, emphasizing the need for robust evaluation methods and stability-enhancing techniques. In contrast, tasks like QQP (0.06), QNLI (0.38), SST2 (0.55), and MRPC (0.89) show much greater stability, likely due to their inherent properties such as larger datasets or simpler decision boundaries. This also helps explain why SuperGLUE tasks generally show higher VAR than GLUE tasks.

Compared to RoBERTa-large, Llama3.2-3B with LoRA fine-tuning exhibits significantly lower VAR across most tasks. This is likely because only a small subset of parameters (~2.3 million) is updated during LoRA fine-tuning, which constrains the variance introduced by random seeds and results in greater stability and robustness to seed-induced fluctuations. Furthermore, the decoder-only, autoregressive architecture of `Llama3.2-3B` may be inherently less sensitive to minor parameter perturbations during task adaptation compared to the encoder-only structure of `RoBERTa`, contributing to its robust performance. A deeper analysis of how these factors govern performance variance is a promising avenue for future work.

5.2 Micro impact

Table 1 reports consistency (CON) and correct-consistency (CCON) for all tasks over ten random seeds. For `RoBERTa-large`, high CON values in tasks like SST2 (96.83), QNLI (95.64), and QQP (95.57) indicate stable predictions, while lower values in tasks like COPA (64.88) and RTES (70.05) highlight their sensitivity to random seeds, potentially due to smaller training sizes or task complexity. Tasks with large CON–CCON gaps (e.g., WiC with a 21.61 difference and CoLA with 11.19) suggest that *consistent predictions are not always accurate, emphasizing the need to evaluate both stability and correctness*. Furthermore, tasks like WiC, which show low VAR alongside low CON and CCON, demonstrate that *similar macro accuracy can mask underlying instability*, reinforcing the importance of micro-level evaluation beyond traditional metrics. Results from `Llama3.2-3B` show consistent trends, albeit with task-specific variations.

Unlike the macro impact where `Llama3.2-3B` with LoRA fine-tuning shows significantly lower VAR than `RoBERTa-large`, no such trend is observed for CON and CCON. This implies that *using a parameter-efficient method like LoRA with a modern LLM helps mitigate macro variance but has limited effect on micro consistency.*

5.3 Discussion

Will increasing training data size improve variance and consistency in general?

To answer the question, we present Pearson correlation analysis in Figure 2, showing the relationship between training size, variance, and consistency, with tasks sorted by increasing dataset size. It reveals a weak negative correlation (-0.3918) between training size and VAR, indicating that smaller datasets tend to increase macro variance. However, the effect is not pronounced or consistent across all tasks, as MultiRC exhibits high VAR despite a relatively large dataset. A weak or moderate positive correlation is observed between training size and both CON (0.4257) and CCON (0.4259), suggesting that larger datasets generally improve consistency and prediction stability across random seeds, but with no guarantee. *Increasing training size can reduce macro and micro variability in random seeds, but its effectiveness depends on factors like data quality, task complexity, and label noise (Shahinfar et al., 2020; Althnian et al., 2021; Bailly et al., 2022).*

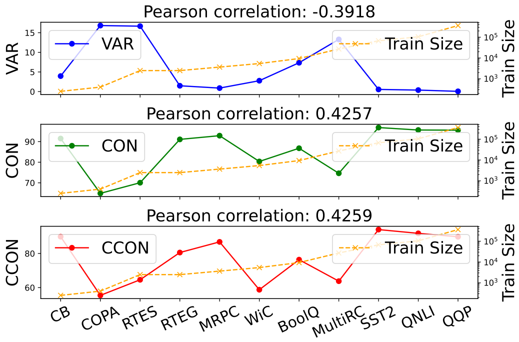

Figure 2: Correlation between training size (log scale), VAR, CON, and CCON while using `RoBERTa-large`. Tasks are arranged in ascending order of training size, with exact sizes detailed in Appendix 4.

Outlook: mitigating seed-induced variability.

Our analysis reveals substantial seed-induced variability in model performance at both the macro and micro levels. The findings indicate that increasing the training size can reduce both macro and micro variability in random seeds, but it's not a guaranteed solution, especially for complex tasks.

Prior studies have proposed several strategies to mitigate macro variability, including model ensembling (Risch and Krestel, 2020; Wang et al., 2023, 2020), stability-aware training (Dodge et al., 2020), and more robust evaluation protocols (Mosbach et al., 2020). Among these methods, only Wang et al. (2020) explicitly addresses consistency through snapshot ensembles. While this method guarantees improvement, it is computationally prohibitive for LLMs, as it requires training numerous models to form a sufficient ensemble.

Inspired by our observation that individual predictions are highly sensitive to randomness, a promising future direction is the development of novel optimization algorithms. Specifically, one could dynamically weight the loss of individual examples during gradient propagation to stabilize training and reduce both macro and micro variance, without incurring significant computational overhead. We leave the exploration of such methods for future work.

6 Conclusion

In conclusion, this work highlights the significant impact of random seeds on pretrained LLMs, revealing variability at both macro and micro levels. By introducing a novel consistency metric, we emphasize the importance of considering seed-induced variability in individual predictions in model evaluation. Our findings stress the need for incorporating random seed sensitivity into benchmarking for more reliable and reproducible results.

7 Limitations

Our work focuses on classification tasks which are the main consists of GLUE and SuperGLUE benchmarks. While this provides a solid foundation, it may limit the generalizability of our findings to other NLP task types. Incorporating a broader set of benchmark datasets would allow for a more comprehensive evaluation using our proposed macro and micro metrics across diverse task categories (as summarized in Table 2). Experimenting with greater task diversity would better capture variability in model behavior, ultimately enhancing the robustness and applicability of our analysis. We leave this broader exploration as future work.

Acknowledgement

The authors would like to thank the anonymous reviewers for feedback that improved the paper, the New Jersey Institute of Technology (NJIT) and the US National Institutes of Health (NIH) for providing funding. This research is supported by NJIT New Faculty Grant and NIH Grant R01LM013486. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NJIT or NIH.

References

Julius Åkesson, Johannes Töger, and Einar Heiberg. 2024. Random effects during training: Implications for deep learning-based medical image segmentation. *Computers in Biology and Medicine*, 180:108944.

Alhanoof Althnian, Duaa AlSaeed, Heyam Al-Baity, Amani Samha, Alanoud Bin Dris, Najla Alzakari, Afnan Abou Elwafa, and Heba Kurdi. 2021. Impact of dataset size on classification performance: an empirical evaluation in the medical domain. *Applied Sciences*, 11(2):796.

Alexandre Bailly, Corentin Blanc, Élie Francis, Thierry Guillotin, Fadi Jamal, Béchara Wakim, and Pascal Roy. 2022. Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models. *Computer Methods and Programs in Biomedicine*, 213:106504.

Steven Bethard. 2022. We need to talk about random seeds. *arXiv preprint arXiv:2210.13393*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Noah Smith, and Hannaneh Hajishirzi. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*.

Prakhar Ganesh, Hongyan Chang, Martin Strobel, and Reza Shokri. 2023. On the impact of machine learning randomness on group fairness. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, pages 1789–1800.

Shivanshu Gupta, Matt Gardner, and Sameer Singh. 2023. Coverage-based example selection for in-context learning. *arXiv preprint arXiv:2305.14907*.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. 2024. The impact of initialization on lora finetuning dynamics. *Advances in Neural Information Processing Systems*, 37:117015–117040.

Jia He, Mukund Rungta, David Koleczek, Arshdeep Sekhon, Franklin X Wang, and Sadid Hasan. 2024. Does prompt formatting have any impact on llm performance? *arXiv preprint arXiv:2411.10541*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Ana Lucic, Maurits Bleeker, Samarth Bhargav, Jessica Forde, Koustuv Sinha, Jesse Dodge, Sasha Luccioni, and Robert Stojnic. 2022. Towards reproducible machine learning research in natural language processing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial*

Abstracts, pages 7–11, Dublin, Ireland. Association for Computational Linguistics.

Pranava Madhyastha and Rishabh Jain. 2019. On model stability as a function of random seed. *arXiv preprint arXiv:1909.10447*.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2020. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. *arXiv preprint arXiv:2006.04884*.

David Picard. 2021. Torch. manual_seed (3407) is all you need: On the influence of random seeds in deep learning architectures for computer vision. *arXiv preprint arXiv:2109.08203*.

Julian Risch and Ralf Krestel. 2020. Bagging BERT models for robust aggression identification. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 55–61, Marseille, France. European Language Resources Association (ELRA).

Saleh Shahinfar, Paul Meek, and Greg Falzon. 2020. “how many images do i need?” understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring. *Ecological Informatics*, 57:101085.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint arXiv:1905.00537*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Lijing Wang, Dipanjan Ghosh, Maria Gonzalez Diaz, Ahmed Farahat, Mahbubul Alam, Chetan Gupta, Jiangzhuo Chen, and Madhav Marathe. 2020. Wisdom of the ensemble: Improving consistency of deep learning models. *Advances in Neural Information Processing Systems*, 33:19750–19761.

Lijing Wang, Yingya Li, Timothy Miller, Steven Bethard, and Guergana Savova. 2023. Two-stage fine-tuning for improved bias and variance for large pretrained language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15746–15761, Toronto, Canada. Association for Computational Linguistics.