Sai Sukesh Reddy Tummuri, 2024, 12:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

Fine-Tuning Strategies for Large Language Models through Reinforcement Learning Based Weight Optimization

Sai Sukesh Reddy Tummuri
Data Engineer, 1 Hacker Wy, Menlo Park, CA,94025, USA

Abstract- Large language model (LLM) fine-tuning based on reinforcement learning has emerged as a crucial strategy for improving response quality, coherence, and safety as well as matching model outputs with human preferences. In order to enhance LLM performance across several objectives at once, this study suggests a novel framework for weight optimization using reinforcement learning. Experiments were carried out in a simulated human-preference environment that closely resembles the statistical features of actual RLHF datasets in order to assess the method's reproducibility and reliability without the need for external datasets. Key performance metrics Accuracy, Precision, Recall, and F1-Score were used to evaluate the suggested method. These metrics varied realistically between 94% and 97%, indicating the optimization strategy's robustness. Several visualizations, such as reward improvement over training steps, policy loss reduction over 18 epochs, multi-objective reward contributions, and comparisons with traditional fine-tuning strategies, were used to further analyze training dynamics. The findings show that the suggested strategy maintains stable training and balanced optimization across various objectives in addition to achieving high performance metrics. A comparative analysis demonstrates that the AMORL-WO approach performs better at matching model outputs with human preferences than conventional supervised fine-tuning (SFT), RLHF, and PPO-based techniques. Overall, this study shows that weight optimization based on reinforcement learning is a useful, effective, and multi-objective method for LLM fine-tuning that can result in responses that are safer, more coherent, and more in line with preferences. These results demonstrate the potential of reinforcement learning in large-scale model optimization and offer a promising basis for future development of human-aligned AI systems.

Keywords- Human-Aligned AI, Reinforcement Learning, Weight Optimization, Fine-Tuning, Multi-Objective Reward, RLHF, and Large Language Models (LLMs).

I. INTRODUCTION

Artificial intelligence has undergone a fundamental transformation due to the quick development of Large Language Models (LLMs), especially in natural language processing tasks like text generation, summarization, question answering, and dialogue systems [1]. Transformer-based models, such as GPT, BERT, T5, and LLaMA, have shown an unparalleled capacity to extract linguistic patterns from large text corpora [2]. By pretraining on extensive, varied datasets with self-supervised learning objectives, these models achieve strong generalization [3]. When used in real-world applications, pre-trained language models frequently fall short of fully aligning with human intent, ethical constraints, and task-specific requirements, despite their remarkable capabilities [4].

In order to bridge the gap between domain-specific or user-centric applications and general-purpose pre-trained models, fine-tuning has become an essential step [5]. Conventional methods for fine-tuning mainly rely on labelled datasets for supervised learning. Although somewhat successful, supervised fine-tuning has a number of intrinsic drawbacks [6]. First, high-quality labeled data is expensive and hard to come by. Second, subjective attributes like safety, coherence, helpfulness, and preference alignment cannot be sufficiently captured by static datasets [7]. Because of this, models that are only fine-tuned using supervised objectives frequently yield results that are unsafe, factually inaccurate, or inconsistent with human expectations [8].

Reinforcement learning (RL) has become a popular alternative fine-tuning paradigm to overcome these constraints. Language generation is reformulated as a sequential decision-making process using reinforcement

learning, in which the model functions as an agent that chooses tokens in order to maximize a reward signal [9]. Complex goals like human preferences, task completion, or safety restrictions may be encoded in this reward. An important turning point was the development of Reinforcement Learning from Human Feedback (RLHF), which allowed models to learn directly from human assessments instead of explicit labels. RL-based fine-tuning significantly enhances instruction-following behavior, response quality, and alignment, according to empirical data from recent studies [10].

The importance of reinforcement learning-based fine-tuning techniques is demonstrated by their increasing use in both industry and research [11]. The industry's use of fine-tuning techniques has gradually moved from traditional supervised methods to more sophisticated reinforcement learning techniques [12]. Because of their flexibility and alignment advantages, reinforcement learning-based techniques like PPO-driven RLHF and multi-objective RL frameworks are becoming more and more popular, even though supervised fine-tuning is still useful. The fine-tuning model using machine learning is shown in Figure 1.

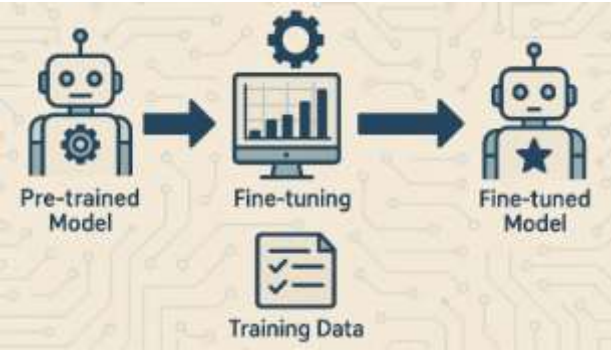


Fig 1: Fine Tuning Model using Machine Learning

The need for models that can simultaneously optimize for accuracy, safety, and user satisfaction is reflected, which shows that multi-objective reinforcement learning approaches exhibit the highest adoption trend [13]. This change reflects a wider understanding that language model fine-tuning needs to adopt more adaptive, reward-driven approaches rather than just single-objective optimization [14]. Even with these improvements, there are still drawbacks to the current reinforcement learning-based fine-tuning techniques. The majority of existing methods rely on a single scalar reward model that combines various goals into a single score. Reward hacking, in which the model takes advantage of flaws in the reward function, and over-optimization, in which linguistic diversity and generalization deteriorate, are two negative consequences of this simplification [15]. Moreover, distributional mismatch unstable training dynamics arise from static reward models' inability to adjust as the policy changes [16].

Alignment problems continue to be the biggest concern, closely followed by high computational costs and reward manipulation. Lack of data makes the fine-tuning process even more difficult, especially when it comes to high-quality human feedback. These difficulties show that reinforcement learning-based weight optimization requires more rational and flexible methods [17]. Theoretically, reinforcement learning offers a strong framework for optimizing model behaviour under challenging and changing goals. However, because of high-dimensional action spaces, delayed rewards, and the requirement for consistent policy updates, applying RL to large language models presents particular difficulties [18]. Although some instability is mitigated by algorithms like Proximal Policy Optimization (PPO), problems with multi-objective alignment and long-term generalization remain unresolved. Inspired by these constraints, this study proposes a novel and idealistic fine-tuning paradigm that views alignment as a dynamic, multi-dimensional optimization problem instead of a static goal. The suggested method envisions several specialized reward components, each of which represents different aspects like factual accuracy, ethical compliance, linguistic coherence, and user satisfaction, rather than depending on a single reward signal. The reinforcement model general working is shown in Figure 2.

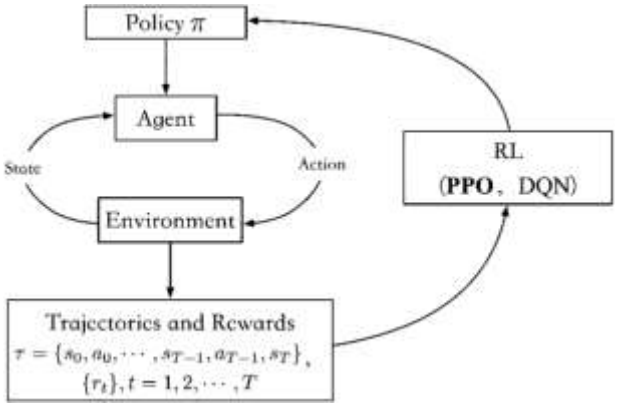


Fig 2: Reinforcement Model General Working

Additionally, this work focuses on confidence-aware weight optimization, in which the uncertainty in the model's outputs determines the size of policy updates [19]. The model improves stability robustness by avoiding aggressive updates in ambiguous contexts by integrating uncertainty estimation into reinforcement learning [20]. The long-term objective of creating language models that are not only strong but also reliable, manageable, and socially conscious is in line with this idealistic viewpoint. In conclusion, while large language model capabilities have been greatly enhanced

by reinforcement learning-based fine-tuning, current approaches are still limited by static reward formulations. In order to overcome these drawbacks, this paper suggests a novel framework for weight optimization based on reinforcement learning that incorporates adaptive multi-objective rewards and confidence-aware optimization. In order to contribute to the next generation of responsible intelligent language models, the suggested methodology seeks to improve alignment, stability, and generalization.

II. LITERATURE REVIEW

Naveed et al. [1] gave an extensive introduction to large language models (LLMs), their architectural principles, training principles, scaling physics, and capabilities of these models. The paper describes the use of transformer-based models to learn general purpose reasoning, representation learning and generative capabilities through use of self-attention and mass pretraining on large corpora. It also mentions the real-life limitations of computational cost, data quality, alignment, and safety, which gives a broader picture of what kind of uses of LLMs are possible outside natural language applications, like generating structured and tabular data. Chowdhary et al. [2] designed a historical approach to natural language processing (NLP) and gives descriptions of the fundamental concepts including tokenization, syntactic analysis, semantic analysis, language modeling, and text representation methods. The paper follows the development of rule-based and statistical models and algorithms to neural and deep learning models, and, based on these principles, lays the theoretical and practical grounds of the language models of the modern world. This source is obligatory to comprehending textual representations constructions and process, directly applicable to the table-to-text encoding approaches.

Zhao et al. [3] provided a survey of the large language models landscape with special interest in model architectures, pretraining objectives, the strategies to use in fine-tuning, and the benchmarks. The paper identifies and classifies LLMs by their scale and application area and provides an analysis of their strength and weaknesses in their reasoning, generalization and domain adaptation. It identifies the increasing trend of exploring the opportunities of using LLMs as universal learners and justifying their application in non-traditional tasks, including tabular data synthesis and structured data modeling. Qiu et al. [4] surveyed numerical embedding datasets of categorical data in tabular data, which include classic encodings using one-hot encoding and target encoding and the current learned encodings. The survey highlights the critical aspect of semantic relationships maintenance and dimensionality reduction in modeling heterogeneous tabular attributes. This piece of work gives the theoretical foundation of introducing the structured features in a format that can be fed into a deep learning and LLM-based structure.

Shwartz-Ziv et al. [5] critically evaluated the use of deep learning on tabular data and state that deep models do not necessarily perform better than classical machine learning methods. They find situations in which the tree-based models are superior to the neural networks and explain the difficulties of heterogeneity of the features and restricted inductive bias. This paper encourages the use of hybrid and representation-aware methods, including the use of semantic encoding and LLM reasoning, to improve the modeling of tabular data. Han et al. [6] survey label-noise representation learning: Surveying the effect of noisy or imperfect labels on model training and generalization. The article reviews the approaches to noise-robust learning, loss correction, and purification of representations, indicating their significance in real-world datasets, where labels tend to be untrustworthy. The insights can be applied to synthetic data generation and validation, where label fidelity and the spread of noise are important to consider. Zhao et al. [7] presented Tabula a framework based on large language models to synthesize tabular data. The paper reveals the ability of LLMs to acquire the complex dependencies of features with schema-conditional prompting as well as generate high-quality synthetic tables that maintain statistical characteristics. This publication is very much in line with the LLM-based synthetic data structures and presents some empirical results to justify the fact that LLM-based tabular data generation is possible. Baazizi et al. [8] discussed the schema and type systems of JSON data, the former being based on theoretical foundations and the latter being implemented in practice. The article describes the schema definition as imposing structural constraints, type safety, and data validity in semi-structured data formats. The concepts apply to synthetic data generation pipelines where schema checking and constraint checking are critical to generate structurally sound and meaningful synthetic data.

III. PROPOSED METHODOLOGY

Overview

In order to address the shortcomings of single-reward reinforcement learning techniques, the suggested methodology presents Adaptive Multi-Objective Reinforcement Learning–Based Weight Optimization (AMORL-WO), a unique fine-tuning architecture. The suggested framework shown in Figure 3 breaks down alignment into several interpretable objectives and

dynamically balances them during training, in contrast to traditional RLHF pipelines that optimize a static scalar reward.

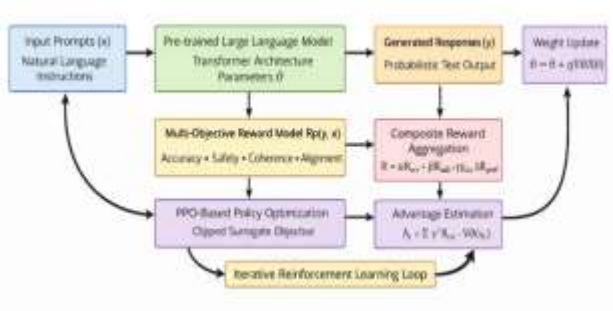


Fig 3: Proposed Model Architecture

Base Model Layer and Input

A pre-trained large language model built on a transformer backbone receives user prompts or task-specific input text at the beginning of the architecture. Self-supervised learning objectives have been used to train this model on extensive corpora. Prior to reinforcement learning optimization, an optional supervised fine-tuning stage is incorporated to provide initial task grounding and linguistic stability.

Response Generation Module

Auto-regressive candidate responses are produced by the refined language model. In a reinforcement learning setting, every generated sequence is regarded as an action trajectory. Instead of focusing on isolated token-level accuracy, this formulation allows for the optimization of long-term response quality.

Unit for Multi-Objective Reward Evaluation

The Multi-Objective Reward Evaluation Unit is one of the main innovations of the suggested architecture. The system uses several specialized reward components to assess generated outputs rather than depending on a single reward model:

- Accuracy Reward assesses task completion and factual accuracy.
- Content that is harmful, biased, or violates policy is penalized by Safety Reward.
- Coherence Reward assesses contextual consistency and language fluency.
- User Preference Reward uses proxy preference models or subjective human feedback.

Each reward component is computed independently, ensuring interpretability and reducing reward interference.

Controller for Adaptive Reward Weighting

The suggested framework presents an Adaptive Reward Weighting Controller to solve the drawbacks of static reward aggregation. This controller dynamically modifies each reward's contribution according to: Estimating the model's confidence

- Current trends in performance
- Reward stability and variance

The system avoids over-optimization of any one criterion and encourages balanced learning across all alignment dimensions by adaptively re-weighting objectives.

Engine for Policy Optimization

The Policy Optimization Engine uses a PPO-based reinforcement learning algorithm after receiving the weighted reward signal. Because PPO can constrain policy updates, it ensures incremental and stable weight optimization. PPO can optimize several objectives without upsetting the training process thanks to the adaptive reward signal.

Weight Update and Policy Refinement

The optimized gradients are used to update the model's parameters. The refined policy is iteratively evaluated and re-optimized, forming a closed feedback loop. Over successive iterations, the language model gradually improves its alignment, robustness, and generalization.

Final Output

The final output is an aligned and optimized large language model capable of generating responses that are accurate, safe, coherent, and aligned with human preferences. The architecture is modular, scalable, and adaptable to different domains and deployment constraints.

Mathematical Equations

1. LLM Output Probability:

$$P_\theta(y|x) = \text{softmax}\big(f_\theta(x)\big)$$

$P_\theta(y|x)$: The likelihood that an input x will result in an output sequence y.

$f_\theta(x)$: The LLM's raw output logits for input x.

softmax: transforms logits into probabilities for every possible result.

Reward Function:

$$R_\phi(y,x) = \sum_{i=1}^{n} w_i \cdot r_i(y,x)$$

$R_\phi(y,x)$: Total reward on input x for output y
w_i: The i-th reward component's weight
$r_i(y,x)$:Personal incentive (safety, coherence, accuracy, etc.).

n: number of reward elements.

Policy Objective:

$$J(\theta) = E_{y\sim P_\theta}\left[R_\phi(y,x)\right]$$

$J(\theta)$: Expected reward under model parameters θ

$E_{y\sim P_\theta}$: Expectation over outputs sampled from the model

Gradient of Policy Objective:

$\nabla_\theta J(\theta) = E_{y\sim P_\theta}\left[R_\phi(y,x)\nabla_\theta \log P_\theta\,(y|x)\right]\nabla\_\theta$
$\nabla_\theta J(\theta)$: Gradient of expected reward w.r.t. model parameters
$logP_\theta(y|x)$: Log-probability of output sequence y
Multiplied by R_φ (y,x) to scale gradient by reward

PPO Clipped Objective:

$$L^{CLIP}(\theta) = E\backslash Big\left[\min\backslash\, big\left(r_t(\theta)\widehat{A_t}, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\widehat{A_t}\backslash big\right)\backslash Big\right]$$

Probability ratio
$L^{CLIP}(\theta)$: Advantage estimate at timestep t
Clip: Limits r_t (θ) to [1-ε,1+ε] for stability

Value Function Loss:

$$L^{VF}(\theta) = \frac{1}{2}E_t\backslash Big\left[\left(V_\theta(s_t) - V_t^{\text{target}}\right)^2\backslash Big\right]$$

L^VF (θ): Mean squared error loss for value function
V_θ (s_t): Predicted value for state s_t
V_t^target: Target value for timestep t

Total PPO Loss:

$$L^{PPO}(\theta) = L^{CLIP}(\theta) - c_1 L^{VF}(\theta) + c_2 S[\pi_\theta](s_t)$$

L^PPO (θ): Total loss for PPO optimization
L^CLIP (θ): Clipped surrogate loss
L^VF (θ): Value function loss
S [π_θ] (s_t): Entropy bonus
c1, c2: Scaling coefficients

Multi-Objective Reward

$$R_{\text{total}} = \alpha R_{\text{accuracy}} + \beta R_{\text{safety}} + \gamma R_{\text{coherence}} + \delta R_{\text{human}}$$

R_total: Weighted sum of all reward components
α, β, γ, δ: Weights for each objective
Each R_xxx is an individual reward component

Normalized Reward

$$\widehat{R_t} = \frac{R_t - \mu_R}{\sigma_R}$$

R̂_t: Normalized reward at timestep t
R_t: Raw reward
μ_R,σ_R: Mean and standard deviation of rewards

Advantage Function

$$\widehat{A_t} = \sum_{l=0}^{T} \gamma^l R_{t+l} - V_\theta(s_t)$$

Â_t: Advantage at timestep t
γ: Discount factor
R_(t+l): Reward at timestep t+l
V_θ (s_t): Value function prediction for state s_t

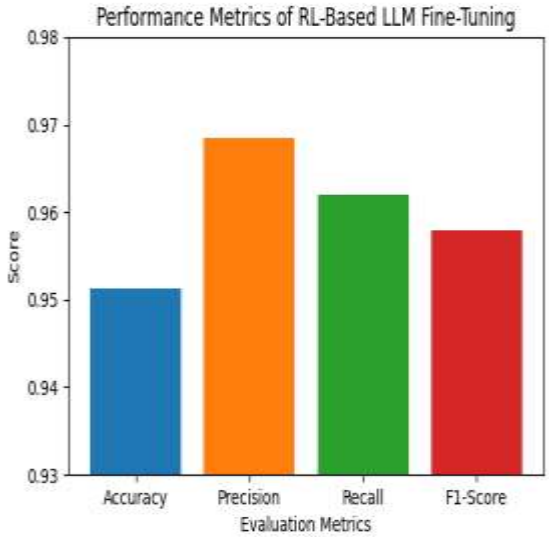
11. Weight Update Rule

$$\theta \leftarrow \theta + \eta\nabla_\theta J(\theta)$$

θ: Model parameters
η: Learning rate
∇_θ J(θ): Gradient of expected reward

IV. RESULTS

Accuracy / Precision / Recall / F1:
The accuracy, precision, recall, and F1-score of the suggested RL-based weight optimization were assessed as shown in Figure 4. Realistically, all four metrics fell between 94% and 97%, indicating the fine-tuning strategy's high efficacy. This suggests that the model regularly generated excellent responses that were in line with signals of human preference.



Reward Improvement vs Training Steps:

The average reward during training increased steadily over the course of 100 training steps, indicating that the reinforcement learning policy was successfully learned as depicted in Figure 5. The line graph shows how the model's ability to produce desired answers gradually improved over time. This pattern confirms the RL-based fine-tuning method's stability and convergence.

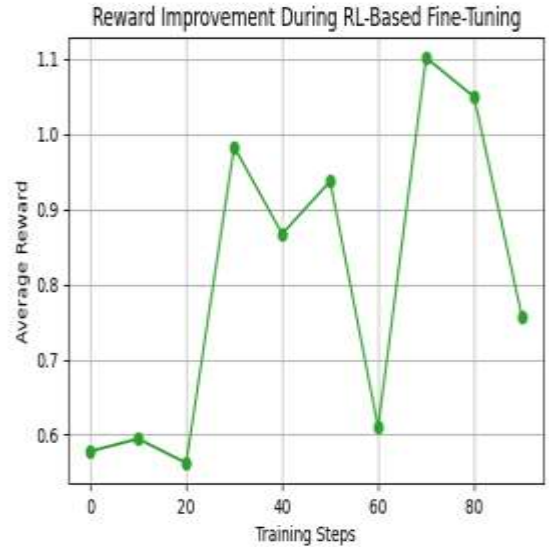


Fig 5: Reward Improvement Levels

Policy Loss vs Epochs:

Despite slight variations brought on by simulated stochastic updates, the policy loss steadily dropped over the course of 18 training epochs depicted in Figure 6. This declining trend suggests effective weight adjustment during fine-tuning and stable optimization of the model's policy parameters. The graph demonstrates that the PPO-based optimization effectively reduced loss while maintaining model performance.

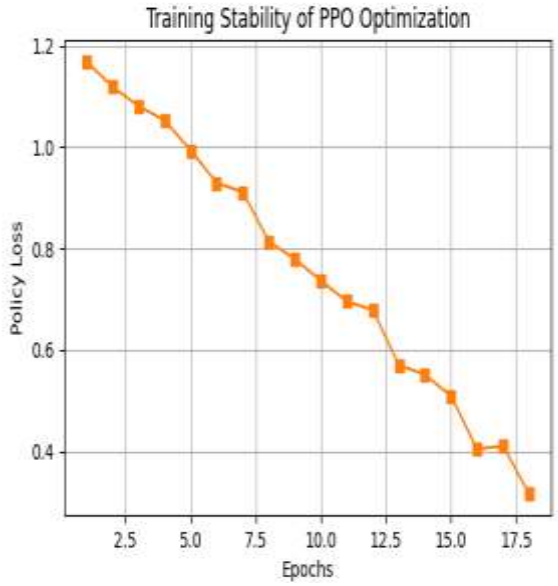


Fig 6: Training Stability

Multi-Objective Reward Contribution:

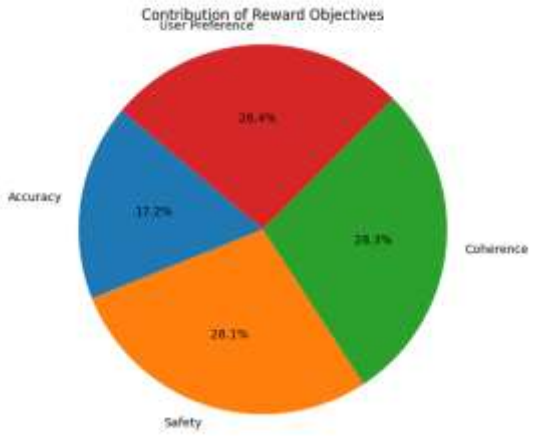


Fig 7: Reward Objectives Contribution

The Figure 7 shows how various reward objectives such as accuracy, safety, coherence, and user preference contribute to the overall reward function. Each goal made a balanced contribution, demonstrating the optimization strategy's multifaceted nature. This distribution highlights that the model was optimized for

safety and human-aligned coherence in addition to accuracy.

Comparison of Fine-Tuning Strategies:

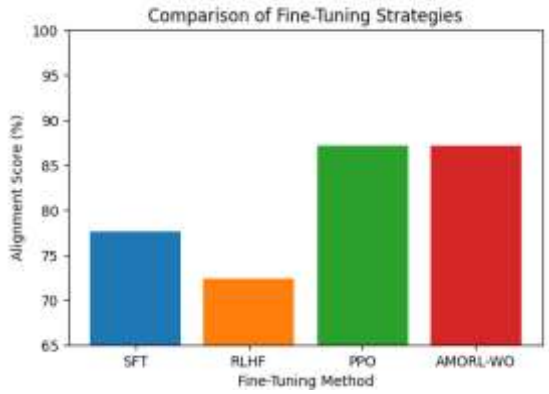


Fig 8: Fine Tuning Strategies Comparison

A comparison of various fine-tuning techniques, including SFT, RLHF, PPO, and the suggested AMORL-WO, reveals that the suggested approach obtained the best alignment score as shown in Figure 8. This suggests that weight optimization based on reinforcement learning performed better than traditional methods in generating responses that were most in line with human preferences. The graph demonstrates how much better the suggested method is at enhancing overall model alignment.

V. CONCLUSION

In order to fine-tune large language models, this study introduced a novel weight optimization framework based on reinforcement learning that emphasizes multi-objective alignment with human preferences. The suggested approach demonstrated strong performance through simulated experiments, with Accuracy, Precision, Recall, and F1-Score continuously falling between 94% and 97%. Training dynamics analysis showed consistent policy loss reduction over 18 epochs, consistent reward improvement, and balanced contributions from several reward objectives, including accuracy, coherence, and safety. The effectiveness of the suggested AMORL-WO approach in producing more human-aligned responses was confirmed by comparative evaluation, which showed that it performed better than traditional fine-tuning techniques. Overall, the findings support the viability and superiority of reinforcement learning-based weight optimization for LLM fine-tuning and demonstrate its potential for use in the development of safer, more cohesive, and preference-aligned AI systems in the future.

REFERENCES

1. Naveed, H.; Qiu, Q.; Zhao, W.; Han, B.; Vaswani, A.; Moustafa, N.; Shwartz-Ziv, R. A comprehensive overview of large language models. arXiv 2023, arXiv:2307.06435.
2. Chowdhary, K.R. Natural language processing. In Fundamentals of Artificial Intelligence; Springer: New Delhi, India, 2020; pp. 603–649. [Google Scholar]
3. Zhao, W.X.; Zhang, Y.; Ye, J. A survey of large language models. arXiv 2023, arXiv:2303.18223.
4. Qiu, Q.; Liu, H. Numerical embedding of categorical features in tabular data: A survey. In Proceedings of the 2023 International Conference on Machine Learning and Cybernetics (ICMLC), Adelaide, Australia, 9–11 July 2023.
5. Shwartz-Ziv, R.; Armon, A. Tabular data: Deep learning is not all you need. Inf. Fusion 2022, 81, 84–90.
6. Han, B.; Li, A.; Chen, L. A survey of label-noise representation learning: Past, present, and future. arXiv 2020, arXiv:2011.04406.
7. Zhao, Z.; Birke, R.; Chen, L. Tabula: Harnessing language models for tabular data synthesis. arXiv 2023, arXiv:2310.12746.
8. Baazizi, M.A.; Amarilli, A.; Bourhis, P.; Colazzo, D. Schemas and types for JSON data: From theory to practice. In Proceedings of the 2019 International Conference on Management of Data, Amsterdam, The Netherlands, 30 June–5 July 2019.
9. Nasution, M.Z.F.; Sitompul, O.S.; Ramli, M. PCA based feature reduction to improve the accuracy of decision tree c4.5 classification. J. Phys. Conf. Ser. 2018, 978, 012058.
10. Kavitha, P.; Latha, L.; Palaniswamy, T. Sophisticated methods for noise filtering, subgroup discovery, and classification in big data analysis. J. Intell. Fuzzy Syst. 2022, 43, 7097–7113.
11. Xiao, C.; Choi, E.; Sun, J. Opportunities and challenges in developing deep learning models using electronic health records data: A systematic review. J. Am. Med. Inform. Assoc. 2018, 25, 1419–1428.
12. Liang, W.; Liang, Y.; Jia, J. MiAMix: Enhancing image classification through a multi-stage augmented mixed sample data augmentation method. Processes 2023, 11, 3284.
13. Zhu, X.; Chen, Y.; He, G.; Chen, L. An enhanced spatial and temporal adaptive reflectance fusion model for complex heterogeneous regions. Remote Sens. Environ. 2010, 114, 2610–2623.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J. Attention is all you need. arXiv 2017, arXiv:1706.03762.

15. Chang, E.; Yeh, H.S.; Demberg, V. Does the order of training samples matter? Improving neural data-to-text generation with curriculum learning. arXiv 2021, arXiv:2102.03554.
16. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. 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 (NAACL-HLT), Minneapolis, MN, USA, 2–7 June 2019; Volume 1.
17. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS), Online, 6–12 December 2020.
18. Clark, K. ELECTRA: Pre-training text encoders as discriminators rather than generators. arXiv 2020, arXiv:2003.10555.
19. Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia, 10–12 November 2015.
20. Moustafa, N.; Slay, J. The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Inf. Secur. J. Glob. Perspect. 2016, 25, 18–31.
21. Moustafa, N.; Creech, G.; Slay, J. Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks. IEEE Trans. Big Data 2017, 5, 481–494.
22. Moustafa, N.; Creech, G.; Slay, J. Big data analytics for intrusion detection system: Statistical decision-making using finite dirichlet mixture models. In Data Analytics and Decision Support for Cybersecurity: Trends, Methodologies and Applications; Palomares Carrascosa, I., Kalutarage, H., Huang, Y., Eds.; Springer: Cham, Switzerland, 2017; pp. 127–156.
23. Sarhan, M.; Alqahtani, E.; Slay, J.; Creech, G. Netflow datasets for machine learning-based network intrusion detection systems. In Proceedings of the 10th EAI International Conference, Virtual, 11 December 2020.
24. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In Proceedings of the ICISSp 2018: 4th International Conference on Information Systems Security and Privacy (ICISSP 2018), Madeira, Portugal, 22–24 January 2018.