InvestAlign: Align LLMs with Investor Decision-Making under Herd Behavior

Huisheng Wang* **Zhuoshi Pan*** **Hangjing Zhang***
Mingxiao Liu* **Yiqing Lin** **H. Vicky Zhao**
Department of Automation, Tsinghua University
{whs22,pzs23,hangjing23,mx-liu21,linyq20}@mails.tsinghua.edu.cn
vzhao@tsinghua.edu.cn

Abstract

Large Language Models (LLMs) can be leveraged to assist in solving complex investment problems. However, the investment decisions generated by existing LLMs often deviate from real-user data. One method to align LLMs with investor decision-making processes is Supervised Fine-Tuning (SFT), which requires a substantial amount of real-user data that is costly to collect and raises concerns about privacy and security. In this work, we propose **InvestAlign**, an efficient method that constructs large-scale SFT training datasets based on the theoretical solution to a similar and simpler optimal investment problem, rather than the original complex one. We theoretically demonstrate that fine-tuning LLMs with these datasets leads to faster parameter convergence compared to using real-user data. By fine-tuning LLMs, we obtain **InvestAgent**s, which align more with real-user data than pre-SFT LLMs in both the simple and original complex problems. This highlights **InvestAlign** as a promising approach with the potential to address complex optimal investment problems and align LLMs with investor decision-making processes in economics and finance.

1 Introduction

Large Language Models (LLMs) have been widely adopted in various domains as generative agents to assist with specific tasks [1, 2]. There is an emerging trend that LLM agents are equipped with human-like intelligence to simulate human decision-making processes [3]. In economics and finance, substantial works have been done on aligning LLMs with human values and decisions, particularly in models for market behavior prediction and the analysis of complex economic data for policy-making [4, 5]. These efforts often focus on macroeconomic issues, such as information dissemination and collective decision-making in global markets [6]. To the best of our knowledge, little attention has been paid to LLMs' performance in microeconomics and behavioral finance, especially concerning investor decision-making under herd behavior, and current LLMs are shown to not fully align with investors' behavior in micro-level financial decision-making, as demonstrated in Section 2.

Achieving the alignment of LLMs to investors' decision-making processes often relies on large-scale real-user data in Supervised Fine-Tuning (SFT) [7]. Fine-tuned with specific training datasets, LLMs can better reflect investor behavior in complex problems. However, it faces the following obstacles. Collecting real-user data can be costly due to the wide variation in investors' attributes, such as risk preference and herd behavior degree [8]. Additionally, many investors are reluctant to share their investment decisions due to privacy and security concerns. To address the data scarcity problem, note that for some simple problems such as the one in [9], we have already found its theoretical solution, using which we can generate a large amount of data. Therefore, one possible solution is,

*The four authors contributed equally to this work.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

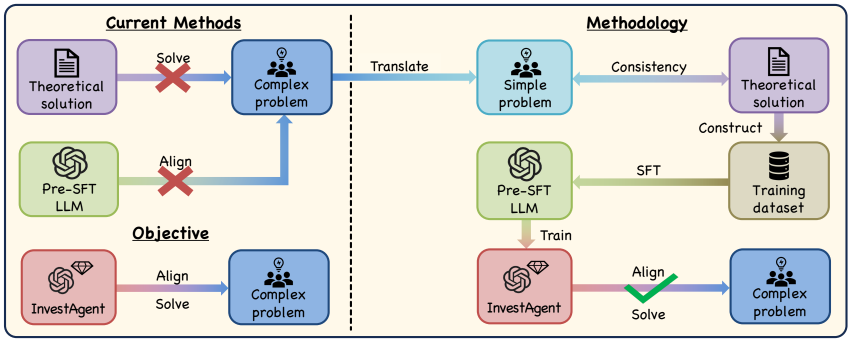


Figure 1: Overview of **InvestAlign**.

given a complex problem, we first identify a similar and simpler problem with a theoretical solution, construct the SFT training dataset using this theoretical solution, and then fine-tune LLMs to solve the original complex problem. There are several issues to be addressed when following this approach:

QA: Given the complex problem, how to identify a similar and simpler problem?
QB: Do the theoretical solution of the simpler problem align with real users' investment decisions, and can they be used to construct a training dataset that mirrors investor decision-making processes?
QC: How can we construct the training dataset based on the theoretical solution of the simpler problem? How does it perform in aligning with real-user data?
QD: How to adapt the fine-tuned LLMs to solve the complex problem, and what is its performance?

To verify the feasibility of the proposed approach and to address the above four issues, in this work, we consider the following simple scenario of optimal investment as an example. Assume that there are two agents, one is an investment assistant and the other is an investor whose investment decisions are unilaterally influenced by the assistant under the herd behavior. For the original complex problem to solve, we consider the relative herd behavior in [10] where the investor adjusts his/her investments in response to changes made by the investment assistant within the same time interval and imitates the changing rate of the investment assistant's decision. Note that for this problem, even though we find its theoretical solution, its computational complexity is very high. To answer **QA**, we use the absolute herd behavior in [9] as the simpler problem where the investor replicates the entire portfolio of the investment assistant, and its established theoretical solution can be derived more easily. Then, to answer **QB**, we collect real-user data on the simpler problem using interviews and questionnaires, and apply statistical methods to validate the consistency between real-user data and the theoretical solution. Next, to answer **QC**, we construct SFT training datasets based on the theoretical solution, and theoretically prove that fine-tuning LLMs on the training datasets leads to faster parameter convergence than using real-user data. Then, to answer **QD**, given the training dataset, we fine-tune the LLMs and develop the **InvestAgent**s, which can make decisions similar to the theoretical solution, thus aligning with real-user data. Finally, we conduct another real-user test to verify the performance of **InvestAgent**s on solving the complex problem. The experimental results show that **InvestAgent**s exhibit better alignment performance than pre-SFT LLMs.

2 Problem Simplification and Real-User Data Verification

2.1 Optimal Investment Problems under Herd Behavior

Following the prior work in [11], we consider the scenario where an investor and an investment assistant invest in the period $\mathcal{T}$ in a financial market consisting of a deposit and a stock. We define the funds invested in the stock by the investor and investment assistant as their *investment decisions*, denoted by $\{P(t)\}_{t\in\mathcal{T}}$ and $\{Q(t)\}_{t\in\mathcal{T}}$, respectively. We denote r as the interest rate of the deposit, v and σ as the excess return rate and volatility of the stock, and T as the terminal time. Given the above parameters, the investor's fund $\{X(t)\}_{t\in\mathcal{T}}$ satisfies $\mathrm{d}X(t) = [rX(t) + vP(t)]\mathrm{d}t + \sigma P(t)\mathrm{d}W(t)$, where $X(0) = x_0$ is his/her initial fund, and $\{W(t)\}_{t\in\mathcal{T}}$ is a standard Brownian motion modeling the randomness of the stock price. We assume that the investment assistant is rational and tries to maximize his/her expected utility of the terminal wealth, and from [12], we assume that the investment assistant's decision $\{Q(t)\}_{t\in\mathcal{T}}$ satisfies $Q(t) = \frac{v}{A\sigma^2}\exp[r(t-T)], t \in \mathcal{T}$, where A is the investment assistant's risk aversion coefficient [13]. Considering the herd behavior, the investor jointly maximizes his/her expected utility of the terminal fund $\mathbb{E}\phi[X(T)]$ and minimizes the distance between his/her own and the investment assistant's decisions $D(P, Q)$. Following the prior work in [12], we assume that the investor's utility of the terminal fund is $\phi[X(T)] = -\frac{1}{\alpha}\exp[-\alpha X(T)]$, where α is his/her risk aversion coefficient. In summary, the optimal investment problem under herd behavior is $\sup_{\{P(t)\}_{t\in\mathcal{T}}} \mathbb{E}\phi[X(T)] - \theta D(P, Q)$, where θ is the influence coefficient to address the tradeoff between the two different objectives. We call the risk aversion coefficient α and the influence coefficient θ the investor's *investment attribute*. Following the prior work in [10], when considering the relative herd behavior, the distance is defined as $\delta(P, Q) = \frac{1}{2}\int_0^T [P'(t) - Q'(t)]^2 \mathrm{d}t$. In this case, the optimal investment problem is ***P1*:** $\sup_{\{P(t)\}_{t\in\mathcal{T}}} \mathbb{E}\phi[X(T)] - \theta\delta(P, Q)$. For the case of absolute herd behavior, the distance is defined as $\Delta(P, Q) = \frac{1}{2}\int_0^T [P(t) - Q(t)]^2 \mathrm{d}t$, and the optimal investment problem is ***P2*:** $\sup_{\{P(t)\}_{t\in\mathcal{T}}} \mathbb{E}\phi[X(T)] - \theta\Delta(P, Q)$. The theoretical optimal decision for ***P2*** is in Appendix A.1. The parameter values in ***P1*** and ***P2*** are in Appendix A.2.

2.2 Validation of Pre-SFT LLMs and the Theoretical Solution

To verify whether the theoretical solution matches users' real investment decisions, we collect real-user data from 119 participants using interviews and questionnaires when facing the investment problem ***P2***. We denote the index set of participants as $\mathcal{I} = \{1, 2, \ldots, 119\}$. To reduce bias and noise in the collected data, we primarily recruit professionals and students in the fields of economics and finance, and we treat this real-user data as a proxy for the ground truth. The questionnaire we use is in Figure 4 in Appendix A.6. Next, to verify whether pre-SFT LLMs align with real-user data, we collect the pre-SFT LLMs' investment decisions. In this work, we choose a variety of LLMs, including API-based model `GPT-3.5-Turbo` [14], as well as open-source models like `GLM-4-9B-CHAT` [15], `Qwen2-7B-Instruct` [16], and `Meta-Llama-3.1-8B-Instruct` [17]. To obtain these pre-SFT LLMs' investment decisions in ***P2***, we construct a prompt, as shown in Figure 5 in Appendix A.6.

The real-user data shows that the participants' risk aversion coefficients $\{\alpha_i\}_{i\in\mathcal{I}}$ and influence coefficients $\{\theta_i\}_{i\in\mathcal{I}}$ fall within the ranges of $\tilde{\mathcal{S}}_\alpha = [0.09, 0.38]$ and $\tilde{\mathcal{S}}_\theta = [0, 1\times 10^{-7}]$, respectively. For the convenience of data processing, we discretize these two sets into $\tilde{\mathcal{S}}_\alpha = \bigcup_{m\in\mathcal{M}} \tilde{\mathcal{S}}_\alpha^m$ and $\tilde{\mathcal{S}}_\theta = \bigcup_{n\in\mathcal{N}} \tilde{\mathcal{S}}_\theta^n$, and treat values that fall within the same interval as the same value. We then group the participants according to these subsets, with participants sharing the same investment attributes forming a class. Specifically, the class of participants with risk aversion coefficient $\alpha \in \tilde{\mathcal{S}}_\alpha^m$ and influence coefficient $\theta \in \tilde{\mathcal{S}}_\theta^n$ for all $m \in \mathcal{M}$ and $n \in \mathcal{N}$ is denoted as $\mathcal{I}^{mn} = \{i | \alpha_i \in \tilde{\mathcal{S}}_\alpha^m, \theta_i \in \tilde{\mathcal{S}}_\theta^n\}$ for all $m \in \mathcal{M}$ and $n \in \mathcal{N}$. For each participant class $\mathcal{I}^{mn}$, we calculate the mean of the real-user data, the mean of the pre-SFT LLMs' investment decisions based on 10 repeated trials with the same investment attribute, and the corresponding theoretical solution. The experimental results in Figure 2 show that there is a significant discrepancy between the pre-SFT LLMs' investment decisions and the real-user data, indicating that pre-SFT LLMs fail to align with real-user data in optimal investment under absolute herd behavior. This underscores the necessity of supervised fine-tuning to bridge the gap between pre-SFT LLMs' investment decisions and real-user data. On the contrary, the theoretical solutions are closer to the real-user data than pre-SFT LLMs' investment decisions. We validate the statistical consistency between the theoretical solutions and real-user data in Appendix A.4

3 Methodology: InvestAlign

3.1 Constructing SFT Training Dataset with Theoretical Solution

The SFT training dataset comprises input-output pairs used for fine-tuning LLMs, which are generated based on a custom prompt template. The prompt for SFT is in Figure 6 in Appendix A.6. When constructing the SFT training dataset, we need to vary the investment attribute, i.e., the risk aversion coefficient α and the influence coefficient θ. Following the work in [9], we set the values of α and θ in $\hat{\mathcal{S}}_\alpha = \{0.05, 0.10, \ldots, 0.50\}$ and $\hat{\mathcal{S}}_\theta = \{1\times 10^{-8}, 2\times 10^{-8}, \ldots, 1\times 10^{-7}\}$, respectively. We set the above investment attributes through two questions expressed in natural language that are easy for LLMs to understand, rather than directly telling them the values of these parameters. For each investment attribute, we first calculate the theoretical optimal decision $\{\hat{P}(t)\}_{t\in\mathcal{T}}$, and then calculate the investment proportion $\{\hat{P}(t)/X(t)\}_{t\in\mathcal{T}}$. We repeat 10 trials for each investment attribute. In summary, the SFT training dataset contains 1000 training samples.

3.2 Analysis of the Parameter Convergence Rate in Fine-Tuning

We theoretically show that fine-tuning LLMs on the training datasets constructed from theoretical solutions leads to faster parameter convergence compared to using real-user data.

To gain insights and ensure mathematical tractability, we make the following assumptions. First, when calculating the loss function, we only consider the values of the LLM's investment decision, theoretical solution, and real-user data, excluding the natural language parts. Second, we assume that the sample size of the training dataset constructed from the theoretical solution and real-user data are both sufficiently large. Third, we assume that the output layer of the LLM is a Sigmoid layer. We denote the ranges of the LLM's investment decision $P(t)$, theoretical solution $\hat{P}(t)$, and real-user data $\tilde{P}(t)$ as $\mathcal{P}(t)$, $\hat{\mathcal{P}}(t)$, and $\tilde{\mathcal{P}}(t)$, respectively. We can express the cross-entropy loss function as $\hat{L}(\mathbf{w}) = -\sum_{t\in\mathcal{T}}\int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x)\log f_{P(t)}(x)\mathrm{d}x$, where $f_{P(t)}(x)$ and $f_{\hat{P}(t)}(x)$ represent the probability density functions of $P(t)$ and $\hat{P}(t)$ in the training dataset, respectively. Similarly, we can define the cross-entropy loss function $\tilde{L}(\mathbf{w})$ for the case when fine-tuning the LLM using the real-user data. We can further prove that $\|\nabla\hat{L}(\mathbf{w})\| > \|\nabla\tilde{L}(\mathbf{w})\|$. Details are in Appendix A.3. That is, the gradient norm when using the training dataset constructed from the theoretical solution is higher than when using real-user data. This is because, once the parameters are given, the real-user data are noisy, while the theoretical solution is deterministic. According to [18], the gradient descent algorithm converges faster when the gradient norm is larger. Thus, we can conclude that the gradient descent algorithm converges faster when using the training dataset compared to using real-user data. The experiment results in Appendix A.5 validate our above analysis on open-source models.

4 Experiments and Performance Validation

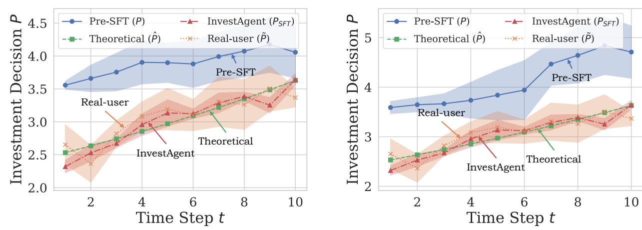
4.1 Performance of InvestAgent in *P2*

Experimental Setup. To compare the alignment performance of pre-SFT LLMs and **InvestAgent**s with real-user data, we develop a Python-based simulation environment. The prompt is in Figure 5 in Appendix A.6. For different investment attributes, we select α from the set $\mathcal{S}_\alpha = \{0.09, 0.13, 0.19, 0.26, 0.38\}$ and θ from the set $\mathcal{S}_\theta = \{0, 1\times10^{-8}, 2\times10^{-8}, \ldots, 1\times10^{-7}\}$. We use 10 random seeds for each investment attribute, producing a total of 550 trials.

Experimental Results. We plot the mean and the 95% confidence interval of the real-user data, denoted by $\tilde{P}$, and the pre-SFT LLMs' and **InvestAgent**s' investment decisions based on 10 repeated trials with the corresponding investment attribute, denoted by P, and P_{SFT}, respectively. We also plot the theoretical solutions, denoted by $\hat{P}$. The experimental results are in Figure 2. Here, we take the investment attribute $\alpha = 0.13$ and $\theta = 7\times10^{-8}$ with `GPT-35` and `Llama-3.1` as examples, and we observe the same trend for other values and LLMs. As shown in Figure 2, **InvestAgent**s' investment decisions are significantly closer to real-user data and theoretical solutions compared to pre-SFT LLMs across different LLMs.

Additionally, to quantitatively evaluate how **InvestAlign** can help pre-SFT LLMs align with real-user data in ***P2***, we calculate the overall MSE between the mean of pre-SFT LLMs' investment decisions and real-user data, i.e., Overall MSE$(P,\tilde{P}) = \frac{1}{|\mathcal{M}||\mathcal{N}||\mathcal{T}|}\sum_{m\in\mathcal{M}}\sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}}[P_{mn}(t)-\tilde{P}_{mn}(t)]^2$, and the overall MSE between the mean of **InvestAgent**s' investment decisions and real-user data, i.e., Overall MSE$(P_{SFT},\tilde{P}) = \frac{1}{|\mathcal{M}||\mathcal{N}||\mathcal{T}|}\sum_{m\in\mathcal{M}}\sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}}[P_{SFT,mn}(t)-\tilde{P}_{mn}(t)]^2$, where $\{\tilde{P}_{mn}(t)\}_{t\in\mathcal{T}}$ represents the mean of the real-user data in class $\mathcal{I}^{mn}$, $\{P_{mn}(t)\}_{t\in\mathcal{T}}$ and $\{P_{SFT,mn}(t)\}_{t\in\mathcal{T}}$ represents the mean of the pre-SFT LLMs' and **InvestAgent**s' investment decisions with the corresponding investment attribute, respectively. The experimental results are in Table 1. As shown in Table 1, **InvestAlign** helps reduce the overall MSEs by $45.59\% \sim 61.26\%$.

The experimental results validate the effectiveness of our proposed method **InvestAlign**, i.e., fine-tuning LLMs using the SFT training dataset constructed from the theoretical solution can align them better with investor decision-making under herd behavior.



(a) `GPT-35` (b) `Llama-3.1`

Figure 2: Comparison of real-user data ($\tilde{P}$), pre-SFT LLMs' investment decision (P), **InvestAgent**s' investment decision (P_{SFT}), and theoretical solution ($\hat{P}$).

Table 1: Comparison of the overall MSE between pre-SFT LLMs' and **InvestAgent**s' investment decisions with real-user data in optimal investment problems ***P2*** and ***P1***.

Overall MSE		`GPT-35`	`GLM-4`	`Qwen-2`	`Llama-3.1`
P2	**Pre-SFT LLM**	4.44	4.20	3.97	4.08
	InvestAgent	1.72	2.26	2.16	1.59
	Reduction from Pre-SFT (%)	-61.26%	-46.19%	-45.59%	-61.03%
P1	**Pre-SFT LLM**	14.03	13.85	17.22	13.07
	InvestAgent	7.46	6.14	7.46	7.25
	Reduction from Pre-SFT (%)	-46.84%	-55.66%	-56.69%	-44.52%

4.2 Performance of InvestAgent in *P1*

Experimental Setup. This experiment shows the alignment performance of our proposed **InvestAlign**, i.e., using LLMs fine-tuned from ***P2*** to solve ***P1***. The prompt we use is in Figure 7 in Appendix A.6. The investment attributes are set the same as those in Section 4.1. We collect 90 real-user data using interviews and questionnaires, and the participants are also primarily professionals and students in the fields of economics and finance to reduce bias and noise in collected data.

Experimental Results. Using the same method in Section 4.1, we report the overall MSE between the mean of pre-SFT LLMs' investment decisions with real-user data, Overall $\mathrm{MSE}(P, \tilde{P})$, and the overall MSE between the mean of **InvestAgent**s' investment decisions with real-user data, Overall $\mathrm{MSE}(P_{SFT}, \tilde{P})$, in Table 1. As shown in Table 1, **InvestAlign** helps reduce the overall MSEs by $44.53\% \sim 56.68\%$. The experiment results validate the effectiveness of our proposed **InvestAlign**, and show that the **InvestAgent**s fine-tuned using the theoretical solution in a similar and simpler problem can better align with human decision-making processes in a complex problem than pre-SFT LLMs. It demonstrates the potential of **InvestAlign** to solve complex optimal investment problems and align LLMs with investor decision-making processes in economics and finance.

5 Conclusion

LLMs can be leveraged to assist in solving complex investment problems. To fine-tune LLMs for alignment with human decision-making processes, a substantial amount of real-user data is required. However, the cost of collecting the real-user data is high, and there are concerns regarding privacy and security. To address these challenges, we propose **InvestAlign**, a novel method that constructs training datasets using the theoretical solution of a similar and simple problem to align LLMs with investor behavior under herd behavior. We demonstrate that fine-tuning LLMs on these training datasets leads to faster parameter convergence compared to using real-user data. The experimental results indicate that **InvestAgent**s, fine-tuned with **InvestAlign**, achieves superior alignment performance in the original complex problem.

References

[1] Grgur Kovač, Rémy Portelas, Peter Ford Dominey, and Pierre-Yves Oudeyer. The socialai school: Insights from developmental psychology towards artificial socio-cultural agents. *arXiv preprint arXiv:2307.07871*, 2023.

[2] Andres M Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D White, and Philippe Schwaller. Chemcrow: Augmenting large-language models with chemistry tools. *arXiv preprint arXiv:2304.05376*, 2023.

[3] Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. Large language models empowered agent-based modeling and simulation: A survey and perspectives. *arXiv preprint arXiv:2312.11970*, 2023.

[4] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[5] Jean Lee, Nicholas Stevens, Soyeon Caren Han, and Minseok Song. A survey of large language models in finance (finLLMs). *arXiv preprint arXiv:2402.02315*, 2024.

[6] Nian Li, Chen Gao, Mingyu Li, Yong Li, and Qingmin Liao. Econagent: Large language model-empowered agents for simulating macroeconomic activities. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15523–15536, 2024.

[7] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023.

[8] Tyler Abbot. Heterogeneous risk preferences in financial markets. In *Paris December 2016 Finance Meeting EUROFIDAI-AFFI*, 2017.

[9] Huisheng Wang and H Vicky Zhao. Optimal investment with herd behaviour using rational decision decomposition. In *2024 43rd Chinese Control Conference (CCC)*, pages 1645–1651. IEEE, 2024.

[10] Huisheng Wang and H. Vicky Zhao. Optimal investment under the influence of decision-changing imitation. *arXiv preprint arXiv:2409.10933*, 2024.

[11] Robert C Merton. Lifetime portfolio selection under uncertainty: The continuous-time case. *The Review of Economics and Statistics*, pages 247–257, 1969.

[12] Leonard CG Rogers. *Optimal investment*, volume 1007. Springer, 2013.

[13] John W Pratt. Risk aversion in the small and in the large. In *Uncertainty in economics*, pages 59–79. Elsevier, 1978.

[14] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[15] Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024.

[16] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

[17] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[18] Jianting Chen and Xiang Yang. Survey of unstable gradients in deep neural network training. *Journal of Software*, 29(7):2071–2091, 2018.

[19] Alfréd Rényi. *Probability theory*. Courier Corporation, 2007.

[20] Jun Shao. *Mathematical statistics*. Springer Science & Business Media, 2008.

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

A Appendix

A.1 Theoretical Optimal Decisions of *P2*

From the prior work in [9], the investor's optimal decision for ***P2*** is

$$\hat{P}(t) = \frac{A\sigma^2\eta\exp[2r(T-t)]+\theta}{\alpha\sigma^2\eta\exp[2r(T-t)]+\theta} \cdot \frac{v}{A\sigma^2}\exp[r(t-T)],\ t \in \mathcal{T}, \tag{1}$$

where the parameter η can be numerically calculated using Algorithm 1.

Algorithm 1: Numerical Method of the Parameter η in ***P2***.

Input: Interest rate: r;
Excess return rate: v;
Volatility: σ;
Initial fund: x_0;
Risk aversion coefficients: α and A;
Investment period: T;
Influence coefficient: θ;
Tolerance: ε.

Output: The parameter η.

$\eta_0 = \exp\left[-\alpha x_0 \mathrm{e}^{rT} - \frac{v^2 T}{2\sigma^2}\right], \Delta\eta_0 = +\infty, k = 0, \vartheta = \frac{\theta}{\alpha\sigma^2}$;

while $\Delta\eta_k \geqslant \varepsilon$ **do**

  $\eta_{k+1} = \eta_0 \exp\left\{\int_0^T \frac{\vartheta^2 v^2 (\alpha/A-1)^2 \mathrm{d}t}{2\sigma^2\left\{\eta_k \mathrm{e}^{2r(T-t)}+\vartheta\right\}^2}\right\}$;

  $\Delta\eta_{k+1} = |\eta_{k+1} - \eta_k|$;

  $k \leftarrow k+1$;

end

$\eta \approx \eta_k$.

A.2 Parameter Setting

Following the prior work in [9], we set the parameter values as follows.

- Interest rate: $r = 0.04$;
- Excess return rate: $v = 0.03$;
- Volatility: $\sigma = 0.17$;
- Initial fund: $x_0 = 10$;
- Investment assistant's risk aversion coefficient: $A = 0.02$;
- Investment period: $T = 10$.

A.3 Gradient Norms of the Loss Function

We assume the parameters α and θ satisfy two uniform distributions, denoted by $U(\underline{\alpha}, \overline{\alpha})$ and $U(\underline{\theta}, \overline{\theta})$, respectively. Therefore, their probability distribution functions are

$$f_\alpha(x) = \tfrac{1}{\overline{\alpha}-\underline{\alpha}},\ x \in [\underline{\alpha}, \overline{\alpha}],\ \text{and}\ f_\theta(x) = \tfrac{1}{\overline{\theta}-\underline{\theta}},\ x \in [\underline{\theta}, \overline{\theta}]. \tag{2}$$

Therefore, we have

$$\begin{aligned} f_{\hat{P}(t)}(x) = \tfrac{1}{\overline{\theta}-\underline{\theta}} \int_{\underline{\theta}}^{\overline{\theta}} f_\alpha \left(\tfrac{1}{\sigma^2 \eta \mathrm{e}^{2r(T-t)}} \left[\tfrac{A\sigma^2 \eta \mathrm{e}^{2r(T-t)}+y}{x} \cdot \tfrac{v}{A\sigma^2} \mathrm{e}^{r(t-T)} - y \right] \right) \\ \cdot \tfrac{A\sigma^2 \eta \mathrm{e}^{2r(T-t)}+y}{\sigma^2 \eta \mathrm{e}^{2r(T-t)} x^2} \cdot \tfrac{v}{A\sigma^2} \mathrm{e}^{r(t-T)} \mathrm{d}y. \end{aligned} \tag{3}$$

Here, following the prior work in [10], we assume that η remains constant when α and θ change slightly. Because $\hat{P}(t) \in \hat{\mathcal{P}}(t)$, we can rewrite (3) as

$$f_{\hat{P}(t)}(x) \approx \tfrac{\min[\hat{\mathcal{P}}(t)] \cdot \max[\hat{\mathcal{P}}(t)]}{\max[\hat{\mathcal{P}}(t)] - \min[\hat{\mathcal{P}}(t)]} \cdot \tfrac{1}{x^2},\ x \in \hat{\mathcal{P}}(t). \tag{4}$$

That is, the theoretical optimal decision $\hat{P}(t)$ approximately satisfies a Pareto distribution. To simplify the notation, we denote the normalization parameter $c = \frac{\min[\hat{\mathcal{P}}(t)] \cdot \max[\hat{\mathcal{P}}(t)]}{\max[\hat{\mathcal{P}}(t)] - \min[\hat{\mathcal{P}}(t)]}$. Using the convolution formula [19], we have

$$\begin{aligned} f_{\tilde{P}(t)}(x) &\approx \tfrac{1}{2\varepsilon} \int_{-\varepsilon}^{\varepsilon} f_{\hat{P}(t)}(x-y) \mathrm{d}y \\ &= \begin{cases} \frac{c}{2\varepsilon} \int_{\min[\hat{\mathcal{P}}(t)]-x}^{\varepsilon} \frac{1}{(x-y)^2} \mathrm{d}y, & x \in [\min[\hat{\mathcal{P}}(t)] - \varepsilon, \min[\hat{\mathcal{P}}(t)] + \varepsilon) \\ \frac{c}{2\varepsilon} \int_{-\varepsilon}^{\varepsilon} \frac{1}{(x-y)^2} \mathrm{d}y, & x \in [\min[\hat{\mathcal{P}}(t)] + \varepsilon, \max[\hat{\mathcal{P}}(t)] - \varepsilon) \\ \frac{c}{2\varepsilon} \int_{-\varepsilon}^{\max[\hat{\mathcal{P}}(t)]-x} \frac{1}{(x-y)^2} \mathrm{d}y, & x \in [\max[\hat{\mathcal{P}}(t)] - \varepsilon, \max[\hat{\mathcal{P}}(t)] + \varepsilon] \end{cases} \\ &= \tfrac{c}{2\varepsilon} \left(\tfrac{1}{\max\{\min[\hat{\mathcal{P}}(t)], x-\varepsilon\}} - \tfrac{1}{\min\{\max[\hat{\mathcal{P}}(t)], x+\varepsilon\}} \right),\ x \in \tilde{\mathcal{P}}(t). \end{aligned} \tag{5}$$

Therefore, we have

$$\begin{aligned} \nabla \hat{L}(\mathbf{w}) &= -\textstyle\sum_{t \in \mathcal{T}} \int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) \nabla \log f_{P(t)}(x) \mathrm{d}x \\ &= -\textstyle\sum_{t \in \mathcal{T}} \int_{\hat{\mathcal{P}}(t)} \frac{f_{\hat{P}(t)}(x)}{f_{P(t)}(x)} \nabla \mathrm{Sigmoid}(\mathbf{z}) \mathrm{d}x \\ &= -\mathbf{z} \textstyle\sum_{t \in \mathcal{T}} \int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) [1 - f_{P(t)}(x)] \mathrm{d}x \\ &= -\mathbf{z} \textstyle\sum_{t \in \mathcal{T}} \left[\int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) \mathrm{d}x - \int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) f_{P(t)} \mathrm{d}x \right] \\ &= -\mathbf{z} \textstyle\sum_{t \in \mathcal{T}} \left[1 - \int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) f_{P(t)} \mathrm{d}x \right]. \end{aligned} \tag{6}$$

Thus, the gradient norm is

$$\|\nabla \hat{L}(\mathbf{w})\| = \|\mathbf{z}\| \textstyle\sum_{t \in \mathcal{T}} \left[1 - \int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) f_{P(t)}(x) \mathrm{d}x \right]. \tag{7}$$

Next, we compare the two gradient norms $\|\nabla \hat{L}(\mathbf{w})\|$ and $\|\nabla \tilde{L}(\mathbf{w})\|$. We only need to compare the following two integrals: $\int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) f_{P(t)}(x) \mathrm{d}x$ and $\int_{\tilde{\mathcal{P}}(t)} f_{\tilde{P}(t)}(x) f_{P(t)}(x) \mathrm{d}x$. Because the investment decisions of the LLM without SFT can be arbitrary due to the randomness of model parameters, we have $\hat{\mathcal{P}}(t) \subset \tilde{\mathcal{P}}(t) \subset \mathcal{P}(t)$. Because $f_{P(t)}(x)$ is monotonically decreasing, we can prove that

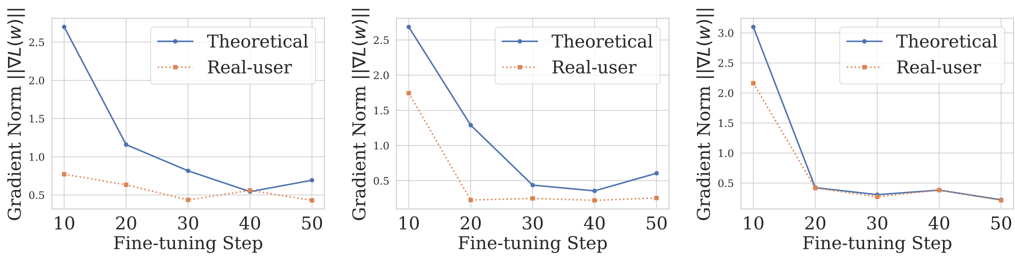
$$\int_{\hat{\mathcal{P}}(t)} f_{\hat{P}(t)}(x) f_{P(t)}(x) \mathrm{d}x < \int_{\tilde{\mathcal{P}}(t)} f_{\tilde{P}(t)}(x) f_{P(t)}(x) \mathrm{d}x < 1, \tag{8}$$

and thus we have

$$\|\nabla \hat{L}(\mathbf{w})\| > \|\nabla \tilde{L}(\mathbf{w})\|. \tag{9}$$

A.4 Statistical Analysis of Theoretical Solution and Real-User Data

We employ statistical methods to validate the consistency between the theoretical solutions and real-user data. For the i-th participant, we denote his/her real investment decision as $\{\tilde{P}_i(t)\}_{t \in \mathcal{T}}$,



(a) `GLM-4` (b) `Qwen-2` (c) `Llama-3.1`

Figure 3: Comparison of the gradient norms between using theoretical solution and real-user data.

and the theoretical solution with the same investment attribute as $\{\tilde{P}_i(t)\}_{t\in\mathcal{T}}$, respectively. We first calculate the difference and correlation coefficient between $\{\tilde{P}_i(t)\}_{t\in\mathcal{T}}$ and $\{\hat{P}_i(t)\}_{t\in\mathcal{T}}$, which are defined as

$$d(\tilde{P}_i, \hat{P}_i) = \textstyle\sum_{t\in\mathcal{T}}[\tilde{P}_i(t) - \hat{P}_i(t)] \text{ and } \rho(\tilde{P}_i, \hat{P}_i) = \frac{\sum_{t\in\mathcal{T}}[\tilde{P}_i(t) - \bar{\tilde{P}}_i][\hat{P}_i(t) - \bar{\hat{P}}_i]}{\sqrt{\sum_{t\in\mathcal{T}}[\tilde{P}_i(t) - \bar{\tilde{P}}_i]^2 \sum_{t\in\mathcal{T}}[\hat{P}_i(t) - \bar{\hat{P}}_i]^2}}, \quad (10)$$

respectively, where $\bar{\tilde{P}}_i = \frac{1}{T}\sum_{t\in\mathcal{T}}\tilde{P}_i(t)$ and $\bar{\hat{P}}_i = \frac{1}{T}\sum_{t\in\mathcal{T}}\hat{P}_i(t)$ are the averages of the i-th participant's investment decisions and the theoretical solution at different time steps, respectively. Next, we conduct t-tests on the means of the differences $\{d(\tilde{P}_i, \hat{P}_i)\}_{i\in\mathcal{I}}$ and the correlation coefficients $\{\rho(\tilde{P}_i, \hat{P}_i)\}_{i\in\mathcal{I}}$ [20], respectively. For the differences $\{d(\tilde{P}_i, \hat{P}_i)\}_{i\in\mathcal{I}}$, the results show that their mean does not significantly deviate from 0 at the 1% significance level, with a t-statistic $= -1.075$. For the correlation coefficients $\{\rho(\tilde{P}_i, \hat{P}_i)\}_{i\in\mathcal{I}}$, the results show that their mean does not significantly deviate from 0.85 at the 1% significance level, with a t-statistic $= -0.843$. Since a mean difference close to 0 indicates minimal discrepancy and a correlation coefficient close to 0.85 reflects a strong positive relationship, we show that there exists significant consistency between the theoretical solution and real-user data.

A.5 Evaluation on the Parameter Convergence Rate in Fune-Tuning

We conduct an experiment to validate our above analysis on open-source models including `GLM-4-9B-CHAT`, `Qwen2-7B-Instruct`, and `Llama-3.1-8B-Instruct`. We construct the SFT training datasets using both the theoretical solution and real-user data, and fine-tune the LLMs with these training datasets using low-rank adaptation (LoRA) in [21]. We set the LoRA rank, alpha, and dropout rate as 4, 32, and 0.1, respectively, and keep the training parameters, such as the learning rate and batch size, etc., unchanged. The experimental results of the gradient norm $\|\nabla L(\mathbf{w})\|$ are in Figure 3. From Figure 3, the gradient norm when using the training dataset constructed from theoretical solution is significantly higher than when using real-user data across different LLMs, validating that fine-tuning LLMs on the training datasets constructed from theoretical solution leads to faster parameter convergence compared to using real-user data.

A.6 Questionnaire and Prompts

Questionnaire for real-user data in *P2*

1. Task Description

Starting from next year, you plan to use a portion of your savings (10 million dollars) to invest in a stock and a deposit as part of your personal retirement fund. You will establish a dedicated account to manage this retirement fund. This means you will make a one-time deposit of 10 million dollars into this account and will not deposit any additional funds or withdraw any funds from this account afterward.

The annualized return of the stock is 7%, with a volatility of 17%. An annualized return of 7% means that if you invest $100 in this stock, you can expect to have $107 after one year on average (the original $100 plus $7 in return). A volatility of 17% indicates that:

With a 68% probability, the price will be between $100 $\pm$ $17 (i.e., $83 to $117) after one year.

With a 95% probability, the price will be between $100 $\pm$ 2 $\times$ $17 (i.e., $66 to $134) after one year.

With a 99.7% probability, the price will be between $100 $\pm$ 3 $\times$ $17 (i.e., $49 to $151) after one year.

The annualized return of the deposit is 4%. If you invest $100 in the deposit, you will receive $104 after one year (the original $100 plus $4 in return).

Over the next 10 years, you will make investment and savings decisions once per year, for a total of {T} decisions. These 10 decision points are labeled 1, 2, ..., 10. At the beginning of year t ($1 \leq t \leq 10$), let the funds in your dedicated account be X(t). Your decision is to allocate part of these funds to invest in the stock, denoted as P(t); the remaining funds will be allocated to savings, which will be X(t) - P(t). **You will determine the proportion of funds to allocate to the stock, i.e., P(t) / X(t).**

During the decision-making process, we will provide you with a **investment assistant** developed by **Omitted for Anonymity**. The investment assistant will provide you with auxiliary information at each decision point. You can refer to the investment assistant's recommendations to some extent, but note that these recommendations may not be optimal. You should also use your own investment insights to avoid blindly following the investment assistant.

Your goal is to maximize the total amount of funds after 10 years and minimize the risk.

2. Investment Decisions

Now, you have 10 million dollars for investment and savings, and the investment assistant recommends the following investment proportions for the stock over the 10 years: [36.21%, 35.59%, 34.96%, 34.35%, 33.73%, 33.13%, 32.53%, 31.93%, 31.34%, 30.75%]. Considering the investment assistant's recommendations, based on your own investment insights, what is your decided investment proportion sequence for the stock over these 10 years? You need to give a list containing 10 percentages, with each percentage ranging from 0% to 100% and precise to two decimal places, representing the investment proportion for each year t. For example, [34.79%, 38.58%, 35.75%, 32.17%, 31.61%, 30.52%, 34.01%, 32.48%, 34.20%, 31.70%]. You need to replace this percentage list with your actual investment proportion sequence. [________]

3. Your Investment Characteristics

(1) At what probability (denoted by p) are the following two choices indifferent to you? A. A probability p of receiving $20, and a probability 1 - p of receiving nothing. B. Receiving $6. [________]

(2) When making a decision, how much do you rely on the investment assistant? Please directly give an integer between 0 and 10. 10 means you rely heavily on the investment assistant, and 0 means you rely little on him/her. [________]

Figure 4: Questionnaire for real-user data in ***P2***.

Prompt for pre-SFT LLMs and InvestAgents in *P2*.

Task Description
Background
Assume you are an investment expert. Starting from next year, you plan to use a portion of your savings (10 million dollars) to invest in (1) a stock (hereinafter referred to as **Investment**) and (2) a deposit (hereinafter referred to as **Savings**) as part of your personal retirement fund. You will establish a dedicated account to manage this retirement fund. This means you will make a one-time deposit of 10 million dollars into this account and will not deposit any additional funds or withdraw any funds from this account afterward. Please remember that you need to provide the proportion of funds allocated to the stock each year over the 10 years in the form of a percentage list, rather than providing decision-making recommendations or writing code.
Financial Market
Information on the stock: **The annualized return of the stock is 7%, with a volatility of 17%.** An annualized return of 7% means that if you invest \$100 in this stock, you can expect to have \$107 after one year on average (the original \$100 plus \$7 in return). A volatility of 17% indicates that:
With a 68% probability: The asset price will be between \$100 $\pm$ \$17 (i.e., \$83 to \$117) after one year.
With a 95% probability: The asset price will be between \$100 $\pm$ 2 $\times$ \$17 (i.e., \$66 to \$134) after one year.
With a 99.7% probability: The asset price will be between \$100 $\pm$ 3 $\times$ \$17 (i.e., \$49 to \$151) after one year.
Information on the deposit: **The annualized return of the deposit is 4%.** If you invest \$100 in the deposit, you will receive \$104 after one year (the original \$100 plus \$4 in return).
Investment Period and Assistant
Over the next 10 years, you will make investment and savings decisions once per year, for a total of 10 decisions. These 10 decision points are labeled 1, 2, ..., 10. At the beginning of year t (1 $\leq$ t $\leq$ 10), let the funds in your dedicated account be X(t). Your decision is to allocate part of these funds to invest in the stock, denoted as P(t); the remaining funds will be allocated to savings, which will be X(t) - P(t). **You will determine the proportion of funds to allocate to the stock.**
During the decision-making process, we will provide you with a **investment assistant** developed by **Omitted for Anonymity**. The investment assistant will provide you with auxiliary information at each decision point. You can refer to the investment assistant's recommendations to some extent, but note that these recommendations may not be optimal. You should also use your own investment insights to avoid blindly following the investment assistant.
Task Objective
Your goal is to maximize the total amount of funds after 10 years (while earning returns and mitigating risks; note: the annualized return of the deposit is 4%, and the annualized return of the stock is 7% with a volatility of 17%).

Your Investment Characteristics
As an investment expert, you have the following characteristics:
Your risk aversion coefficient is {alpha}, which means you consider the following two choices to be indifferent when the probability (i.e., p) is {p}: A. With probability p, you can obtain \$20, and with probability 1 - p, you can obtain \$0; B. With 100% probability, you obtain \$6. Note that as an investor, you have a certain level of optimism about "winning" and are willing to take on some risk, so you consider the two options equivalent at probability p = {p}, which is higher than the 30.00% in a completely rational scenario.
Your influence coefficient is {theta}, which means in decision-making, your level of dependence on the investment assistant is: {k} points. A score of 10 indicates a high level of dependence on the investment assistant, while a score of 0 indicates a low level of dependence.

(The next part of this Figure 5 will be continued on the next page.)

Prompt for pre-SFT LLMs and InvestAgents in *P2* (continued)

 # Output Format Requirements

Please output your decision in JSON format, including two parts: (1) Decision Explanation: Explain the reasons behind your investment proportion decisions. (2) Investment Proportion Sequence: The percentage sequence of funds allocated to the stock each year over the 10 years. You need to output a list containing 10 percentages, with each percentage ranging from 0% to 100% and precise to two decimal places, representing the investment proportion for each year t. For example:
{"Decision Explanation": "Briefly explain the reasons behind your investment proportion decisions.", "Investment Proportion Sequence": ["34.79%", "38.58%", "35.75%", "32.17%", "31.61%", "30.52%", "34.01%", "32.48%", "34.20%", "31.70%"]}
Here, ["34.79%", "38.58%", "35.75%", "32.17%", "31.61%", "30.52%", "34.01%", "32.48%", "34.20%", "31.70%"] is just an example. You need to replace this percentage list with your actual investment proportion sequence. Providing the investment proportion sequence is the most important; do not just focus on the explanation and forget to provide the investment proportion sequence!!!

 # Question

Now, you have 10 million dollars for investment and savings, and the investment assistant recommends the following investment proportions for the stock over the 10 years: {refer_ratios}. Considering historical investment situations and the investment assistant's recommendations, based on your own investment insights, what is your decided investment proportion sequence for the stock over these 10 years? (Please follow the previously provided JSON format requirements, and provide a list of 10 specific percentages indicating your investment proportion sequence for these 10 years, rather than giving investment recommendations or writing code.)
Answer:

Figure 5: Prompt for pre-SFT LLMs and **InvestAgent**s in ***P2***.

Prompt for SFT

(The beginning part of is the same as Prompt for pre-SFT LLMs and InvestAgent in *P2*.)

 # Output

According to optimal investment theory, in the above scenario, the optimal amount for investing in the stock, $\hat{P}(t)$, equals the product of the smart investment advisor's investment amount (i.e., the advisor's decision proportion multiplied by the current budget) and a hyperbolic tangent function. The specific calculation is as follows:

$$\hat{P}(t) = \frac{\eta\alpha_2\sigma^2 \mathrm{e}^{2r(T-t)}+\theta}{\eta\alpha_1\sigma^2 \mathrm{e}^{2r(T-t)}+\theta} \cdot \frac{v}{\alpha_2\sigma^2}\mathrm{e}^{r(t-T)},\ t \in \{1, 2, ..., 10\}, \tag{11}$$

where:
r is the interest rate, which is 4%.
σ is the volatility of the stock, which is 17%.
v is the excess return of the stock, which is 3%.
α_1 is my risk aversion coefficient: α_1 = {alpha}.
α_2 represents the risk aversion coefficient of the smart investment advisor: α_2 = 0.2.
θ is my convergence coefficient: θ = {theta}.
The integral constant η depends on θ. In the current settings, η = {eta}.
Substituting the specific numbers, the proportion sequence of funds allocated to the stock is: {optimal_ratios}.
Note that I also need to output the investment proportion sequence in JSON format:
{"Decision Explanation": "Based on the optimal investment theory and substituting specific numbers, the investment proportion sequence for the stock is calculated.", "Investment Proportion Sequence": {optimal_ratios}}

Figure 6: Prompt for SFT.

Prompt for pre-SFT LLMs and InvestAgents in *P1*.

(The beginning part of is the same as Prompt for pre-SFT LLMs and InvestAgent in *P2*.)

 # Output Format Requirements
Please output your decision in JSON format, including two parts: (1) Decision Explanation: Explain the reasoning behind your investment proportion decisions. (2) Investment Proportion Change Sequence: The sequence of **changes** in the percentage of funds allocated to the stock each year over the 10 years. You need to output a list containing 9 percentages, where each percentage represents the change in the investment proportion from year t - 1 to year t, ranging from -100% to 100%. Positive values indicate an increase in investment, while negative values indicate a decrease. For example:
{"Decision Explanation": "Briefly explain the reasons behind your investment proportion decisions.", "Investment Proportion Sequence": ["3.88%", "0.01%", "-4.13%", "1.37%", "1.37%", "-2.79%", "-2.56%", "2.02%", "-0.06%"]}
Here, ["3.88%", "0.01%", "-4.13%", "1.37%", "1.37%", "-2.79%", "-2.56%", "2.02%", "-0.06%"] is just an example. You need to replace this percentage list with your actual investment proportion change sequence. Providing the investment proportion change sequence is crucial; do not just focus on the explanation and forget to include the investment proportion change sequence!!!
 # Initial Investment Situation
In the first year, the proportion of funds allocated to the stock was: {initial_decision}.
 # Question
Now, you have 10 million dollars for investment and savings, and the investment assistant recommends the following investment proportions for the stock over the 10 years: {refer_ratios}. Considering the initial investment situation and the advisor's recommendations, based on your own investment insights, what is your decided annual change sequence for the investment proportion in the stock over these 10 years? (Please follow the previously provided JSON format requirements, and provide a list of 9 specific percentages indicating the changes in your investment proportion over these 10 years, rather than giving investment recommendations or writing code.)
Answer:

Figure 7: Prompt for pre-SFT LLMs and **InvestAgent**s in ***P1***.