Aligning LLMs using Reinforcement Learning from Market Feedback (RLMF) for Regime Adaptation

Raeid Saqur
Department of Computer Science
University of Toronto
Vector Institute for AI
raeidsaqur@cs.toronto.edu

Frank Rudzicz
Faculty of Computer Science
Dalhousie University
Vector Institute for AI
frank@dal.ca

Abstract

We propose a regime adaptive execution methodology in the financial market domain to tackle the *regime switching* problem. Dynamic regime switching, or underlying correlation and covariance shifts in true (hidden) market variables, diminishes the robustness of expert/specialist models on downstream tasks like forecasting or market movement prediction from unseen, online data. Our method uses natural, intrinsic market rewards for adaptive RL alignment (RLMF) of expert LLMs; and a teacher-student, repeating dual-phase (train, execute) pipeline that consistently outperforms SOTA trillion parameter models like GPT-4o. Our approach does not rely on the strength of underlying expert models – any contemporary off-the-shelf foundational LLM model is compatible with our (plug-and-play) algorithm. We use the Llama-2 7B parameter class of base model to show the efficacy of our method that outperforms both generalist and specialist class of expert models and attain strong empirical results including 15% increase in predictive accuracy on concurrent stock-movement prediction benchmarks (detailed in §B).

1 Introduction

In this work, we juxtapose and explore the efficacy of techniques that allow robots to adapt and generalize locomotion in unseen terrains [33, 24, 22] in a vastly different and more complex domain: the financial market.

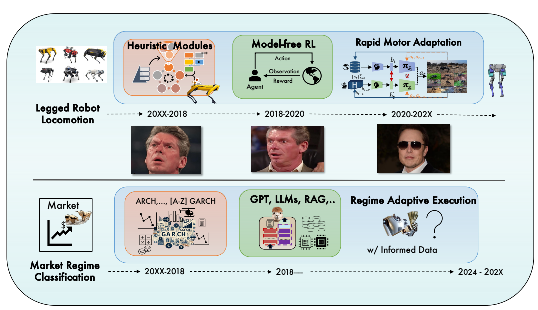


Figure 1: Juxtaposing recent successes of adaptive methods from robot locomotion that supplants decades-old heuristic architectures with our proposed approach to the *regime adaptation* problem.

The true, plausibly large number of variables and mechanics that move the market are hidden or unobservable – making financial market forecasting an extremely hard problem. Thus, reliable market simulation, thereby generating randomized market value trajectories to train agents in simulation is not yet effective, making market prediction in essence a *one-shot* learning task with only one true trajectory or available environment history. Any mapping of input observations ($o_t \in \mathcal{O}$) to output price movement (i.e., market/environment reaction) learned via traditional ML techniques does not generalize well to out-of-domain (or, regime-shifted) distributions due to the hidden, underlying correlation and covariate shifts in a dynamic market regime [1, 13]. Basically, even if we are able to train a model that fits perfectly to past market trajectories (i.e., success in backtesting), it does not guarantee future accuracy.

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

Our solution to this dynamic market regime adaptation problem is motivated and ideated (Fig. 1) by recent, remarkable successes of RL-based adaptive quadruped locomotion techniques in the robotics domain that use two-stage training of *teacher-student* policies [24, 22]. We adopt a similar 2-stage training, then adaptive execution (detailed in §3), using pre-trained LLMs as base policies that we align using an automatic, natural market feedback signal as auxiliary reward. Our preliminary experiments and empirical results show that LLMs, with their imbued generic world knowledge, can support regime adaptation with continual adaptation using RL from intrinsic, natural market rewards – dubbed as the *Reinforcement Learning from Market Feedback* (**RLMF**) loss for LLM alignment.

2 Preliminaries and Background

Regime-Switching in Finance In empirical finance literature, regime switching processes are modeled as *Markovian Switching Models*, introduced by the seminal work of Hamliton [14], in the 1990s. The canonical regime switching problem can be presented by letting o_t be an outcome variable for a market process, which recurrently depends on its own past history, y_{t-1}, ε_t representing random shocks and (for ML/RL community, a conveniently termed) $s_t \in \{0, 1, ..., k\}$ a discrete random variable modeling some underlying *regime process* at time, t. Then regimes affect the intercept(mean), μ_{s_t}, auto-correlation, ϕ_{s_t}, and volatility, σ_{s_t}, of the process [16]:

$$o_t = \mu_{s_t} + \phi_{s_t} o_{t-1} + \sigma_{s_t} \varepsilon_t, \quad \varepsilon_t \sim \text{iid}(0, 1). \tag{1}$$

Enthusiastic readers are encouraged to read [12, 15, 16] for a detailed overview of Markovian switching models, and works on modern heuristic solutions to detecting, classifying, or adopting to such canonical regime switching models. For a comprehensive appreciation and answer to '*why regime adaptation is important*?', we highly encourage reading [1, 13].

Modern deep learning based techniques essentially subsume and skip the problem of regime classification as an intermediary step to some means (like market prediction), and allow the distributional latent embeddings to encapsulate the true regime state from some input data (as a belief b encoding from POMDP formulation). In essense, we too, are adhering to this paradigm, however, unlike other the other methods (relying on deep learning or RL based solutions), we dynamically adapt and update the learned policy using our proposed methodology.

Reward based alignment of Language Models Tuning pretrained LMs using reward feedback and RL enables remarkable capabilities of current chat-bots and assistants to follow instructions. The RLHF pipeline [58, 43, 32] is a well-formulated approach in the NLP domain. While variants to RLHF have been proposed [35], we discuss only the popular RLHF pipeline for our purposes here. At a high-level, the RLHF pipeline starts with fine-tuning a pre-trained LM in supervised manner (typically with the same LM objective, but on new, high-quality domain-specific data) to obtain π^{SFT}, then training a reward model f_θ^{RM} that, once trained, is able to evaluate (usually pairs of) LM generated prompt (x_p) completions: $(\hat{x}_r^1, \hat{x}_r^2) \sim \pi^{SFT}(x_p)$ and provide scalar reward $f_\theta^{RM}(\hat{x}_r) \rightarrow r \in \mathbb{R}$. A human labelled preferences dataset is typically (we deviate from in our presented approach) used to for the reward model training using MLE objective. In the final step, the domain fine-tuned LM, and the trained reward model is used to fine-tune an aligned policy using RL (e.g. PPO [38]) where π^{SFT} acts as the reference based policy: π^{ref}. PPO uses the base, reference model to impose a KL-divergence penalty during RL fine-tuning using reward feedback to ensure the fine-tuned model does not deviate or diverge too far away from the base policy and preventing unwanted scenarios like mode-collapse to high-reward answers.

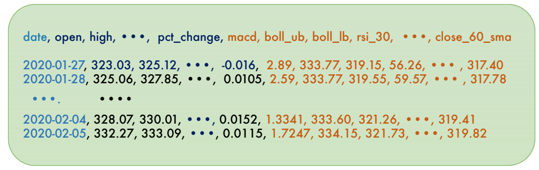
Going forward, observation at time t, o_t, will be referred to as a LM query, x_q comprised of a prompt x_{p_t} and action prediction label from previous time step: $\hat{x}_{r_{t-1}}$ (Fig. 2).

3 Alignment using RLMF

There are two distinct phases in our proposed approach. In the *training* phase, we train a fine-tuned, and aligned language model as our *teacher policy* $\pi_\phi^{teacher}$, and a *reward model* f_θ^{RM}, following the well-formulated RLHF pipeline [58, 43, 32], and using samples from the NIFTY datasets [34] (see details of the datasets NIFTY-LM ($\mathcal{D}_{LM}$) and NIFTY-RL ($\mathcal{D}_{RL}$) in Appendix §C.1.

Each sample (JSON-object line) of the $\mathcal{D}_{LM}$ contain high-quality, processed (one-turn) conversational query, where a query x_q comprises of a prompt x_p and a response x_r, i.e., $x_q = (x_p; x_r)$. Thus, this dataset samples can be used for supervised fine-tuning (SFT) of a pretrained LM policy using the language modeling objective. Similarly, the `NIFTY-RL` dataset compiles a preferences dataset for rejection sampling and RL fine-tuning availing samples of chosen and rejected labels: $\mathcal{D}_{RL} = \left\{ \left(x_p^{(i)}, x_{r_w}^{(i)}, x_{r_l}^{(i)} \right) \right\}_{i=1}^{N}$ where $(x_{r_w} \succ x_{r_l} | x_p)$.

(a) Instruction component of a π_{LM} policy query x_q.



(b) The market's **history** is provided as the past t days of numerical statistics like the (OHLCV) price (in blue) and common technical indicators (in orange) data.

Figure 2: Instruction or prompt prefix, x_p, decomposition into components 5a and 5b.

Supervised Fine-tuning Teacher Policy The loss on a sequence $\mathbf{x}$ (comprised of tokens $x_1, ..., x_T$) from a vocabulary of size V is the autoregressive cross-entropy loss (presuming a decoder-only transformer model akin to the GPT series [6]:

$$\mathcal{L}(x, \boldsymbol{\theta}) = -\sum_{t=1}^{T} \log P_{\hat{y}|x} \left(x_t \mid x_{1:t-1}; \boldsymbol{\theta} \right) \tag{2}$$

where $P_{\hat{y}|x}$ is the output distribution of a model parameterized by θ.

Training a Reward Model We train a reward model f_θ^{RM}, initialized with a SFT language model (using Eq. 2), sampling from $\mathcal{D}_{RM}$ in a MLE fashion formulating the preferences labels as a binary classification problem and optimizing for the negative log-likelihood loss:

$$\mathcal{L}_{RM}(\theta) = -\mathbb{E}_{\substack{(x_p, x_{r_w}, x_{r_l}) \\ \sim \mathcal{D}_{RL}}} \left[\log \left(\sigma \left(r_\theta(x, x_{r_w}) - r_\theta(x, x_{r_l}) \right) \right) \right] \tag{3}$$

where $r_\theta(x_p, x_r)$ is a scalar reward for prompt x_p and response x_r with parameters θ, x_{r_w} is the preferred or chosen response out of the pair (x_{r_w}, x_{r_l}) sampled from $\mathcal{D}_{RL}$ (see §C.1).

3.1 Deriving the RLMF objective

Intuition Our formulation of *Reinforcement Learning from Market Feedback* or **RLMF** can be explained in a simple, intuitive manner conceptually. We all can formalize the market movement tomorrow based on our own beliefs (formed from our unique life-experiences or, history) about the market's state and the new information we learned today from any possible sources (news, social media, chatting with a friendly neighbour etc.). The most natural feedback or correction to our beliefs come from the true, observed movement the next day morning. However big the correction is, this feedback will (and should) not be so radical that we forget everything we have internalized from experience up until yesterday – we are likely to attribute the mismatch to the current (likely deviated) market condition (like the inflation, war, interest rate changes etc.).

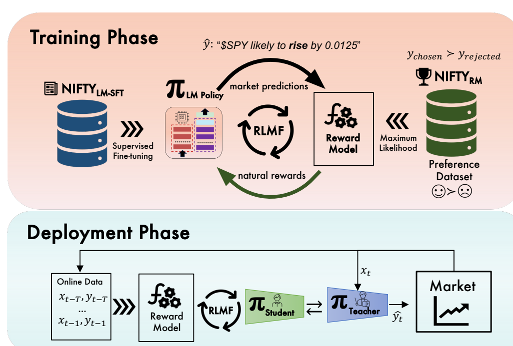


Figure 3: **Regime adaptive execution** uses the NIFTY dataset to train a reward model (RM) and align a pretrained LLM during the training phase. In the deployment phase, streaming online market data is used to continually update the RM, subsequently a student policy that swaps place with an executor teacher policy after windowed intervals.

Technical details : Let π_ϕ^{LM}, be a policy we want to train, that is parameterized by ϕ. We define a policy query as: $x_q = (x_p; x_r)$. Let D_{MF} be a dataset of size T containing tuples of $(x_p, \hat{x}_r, x_r)$, where $\hat{x}_r$ is a generative completion or, response by the policy π_ϕ^{LM}. Let f_θ^{RM} be a trained reward model (using MLE (Eq. 3)), parameterized by θ. And π^{ref} is a frozen (teacher or,) reference policy .

In our setup, the response is an action label of market movement prediction $s.t.\ \hat{x}_r \in$ {rise, fall, neutral}. Note that for each $\hat{x}_r$, we can collect a corresponding truth label from the market's reaction, that we denote by x_r. Having such a rollout dataset, D_{MF} allows us to define a simple MLE based loss objective that we define as our **RLMF**) loss:

$$\begin{aligned}\mathcal{L}_{MF}(\phi) &= \min_\phi \frac{1}{T}\sum_{t=1}^{T} \|\hat{x}_r^t - x_r^t\|^2 \\ &= \min_\phi \mathbb{E}_{(x_p,x_r,\hat{x}_r)\sim\mathcal{D}_{MF}} \left[\|\hat{x}_r - x_r\|^2\right]\end{aligned} \tag{4}$$

The regular RL fine-tuning loss [43] is defined as:

$$\mathcal{L}_{RL}(\phi) = \mathbb{E}_{(x_p,x_r,\hat{x}_r)\sim\mathcal{D}_{MF}}\left[r_\theta(x_p,\hat{x}_r) - \beta \log\left(\frac{\pi_\phi^{LM}(\hat{x}_r|x_p)}{\pi^{ref}(\hat{x}_r|x_p)}\right)\right]$$

where the KL reward coefficient β controls the strength of the KL penalty.

$$\mathcal{L}_{RL}(\phi) = \max_\phi \mathbb{E}_{(x_p,x_r,\hat{x}_r)\sim\mathcal{D}_{MF}}\left[r_\theta(x_p,\hat{x}_r) - \beta\mathbb{D}_{KL}\left[\pi_\phi^{LM}(\hat{x}_r|x_p)\,\|\,\pi^{ref}(\hat{x}_r|x_p)\right]\right] \tag{5}$$

Using the equations 4, 5, we can maximize the following combined objective function using RL for updating policy π_ϕ^{LM}:

$$\mathcal{L}_{RLMF}(\phi) = \min_\phi -\mathcal{L}_{RL}(\phi) + \gamma\mathcal{L}_{MF}(\phi) \tag{6}$$

where the MF reward coefficient γ controls the strength of market feedback reward.

Algorithm 1 Training Phase

Step 1: Fine-tune teacher policy $\pi_\phi^{teacher}$
Init $\pi_\phi^{teacher} \leftarrow \pi^{LM}$ assistant (e.g., Llama2-chat-7b)
Input: D_{tr} (train split of NIFTY-LM), size m, batch B
for $b = 1$ **to** $\lfloor m/B \rfloor$ **do**
  Init batch queries $S_B = \{\}$
  for $i = 1$ **to** B **do**
    Sample $(x_{p_i}; x_{r_i})$
    Append to S_B
  end for
  Update $\pi_\phi^{teacher}$ with SFT {using Eq. 2}
end for
Step 2: Train reward model f_θ^{RM} {using Eq. 3}
Step 3: RL fine-tune $\pi_\phi^{teacher}$ using PPO [38] and $\mathcal{D}_{RL}$, NIFTY-RL preferences dataset.

Algorithm 2 Deployment Phase

Student Policy Adaptation
$t \leftarrow 0, T \leftarrow$ freq
Init $\pi^{student}$ from $\pi^{teacher}$
Repeat every T steps:
  Collect $\mathcal{D}_{MF} = \{(x_p, \hat{x}_\phi^r, x_r^{MF})\}_{t=1}^T$
  Step 1: Update f_θ^{RM} {with Eq. 3}
  Step 2: Update $\pi^{student}$ using $f_{\theta_{\text{upd}}}^{RM}$ and Eq. 6
  Step 3: Set $\pi^{teacher} \leftarrow \pi^{student}$, execute for T

The *The Adaptation algorithms* provide high-level pseudocode for the training[1] and deployment [2] phases of our approach respectively as depicted in Fig. 3.

We present the preliminary results of our approach in the appendix §B.

Limitations and Future Directions Firstly, we note that the goal of our work was to show the feasibility and efficacy of doing financial forecasting and regime adaptation in a fundamentally different, novel way in the current era of LLMs and AI. Thus, our adopted choices of LLMs – like using Llama-2-7b [47] instead of larger or, newer models [18, 48], or RL based alignment techniques instead of RL-free techniques are perhaps best left for future works as variants sweeping for **best performance** was **not our main goal**, but showing **feasibility**/efficacy of **a new direction** was. Future research could see optimization and exploring our method with larger model. Secondly, we want to point out the (deliberate) omission of any downstream financial tasks in this work. The proposed approach can be used for downstream financial tasks, including the use of UnREAL models performing stock trading or portfolio allocation [26].

Acknowledgements Raeid Saqur is supported by Canada NSERC CGS-D Doctoral Grant. The authors acknowledge that resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute https://vectorinstitute.ai/partnerships/current-partners/.

References

[1] Andrew Ang and Allan Timmermann. Regime changes and financial markets. *Annu. Rev. Financ. Econ.*, 4(1):313–337, 2012.

[2] Werner Antweiler and Murray Z Frank. Is all that talk just noise? the information content of internet stock message boards. *The Journal of finance*, 59(3):1259–1294, 2004.

[3] Dogu Araci. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*, 2019.

[4] Gregory W Brown and Michael T Cliff. Investor sentiment and the near-term stock market. *Journal of empirical finance*, 11(1):1–27, 2004.

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

[6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[7] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

[8] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[9] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.

[10] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[11] Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation. September 2021.

[12] Massimo Guidolin. Markov switching models in empirical finance. In *Missing data methods: Time-series methods and applications*, pages 1–86. Emerald Group Publishing Limited, 2011.

[13] Massimo Guidolin and Allan Timmermann. Size and value anomalies under regime shifts. *Journal of Financial Econometrics*, 6(1):1–48, 2008.

[14] James D Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica: Journal of the econometric society*, pages 357–384, 1989.

[15] James D Hamilton. Analysis of time series subject to changes in regime. *Journal of econometrics*, 45(1-2):39–70, 1990.

[16] James D Hamilton. Regime switching models. In *Macroeconometrics and time series analysis*, pages 202–209. Springer, 2010.

[17] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Zhao, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2021.

[18] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[19] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts, 2024.

[20] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

[21] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

[22] Ashish Kumar, Zhongyu Li, Jun Zeng, Deepak Pathak, Koushil Sreenath, and Jitendra Malik. Adapting rapid motor adaptation for bipedal robots. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1161–1168. IEEE, 2022.

[23] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

[24] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47):eabc5986, 2020.

[25] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *ArXiv preprint*, abs/2211.09110, 2022.

[26] Xiao-Yang Liu, Ziyi Xia, Jingyang Rui, Jiechao Gao, Hongyang Yang, Ming Zhu, Christina Wang, Zhaoran Wang, and Jian Guo. Finrl-meta: Market environments and benchmarks for data-driven financial reinforcement learning. *Advances in Neural Information Processing Systems*, 35:1835–1849, 2022.

[27] Dakuan Lu, Jiaqing Liang, Yipei Xu, Qianyu He, Yipeng Geng, Mengkun Han, Yingsi Xin, Hengkui Wu, and Yanghua Xiao. Bbt-fin: Comprehensive construction of chinese financial domain pre-trained language model, corpus and benchmark. *arXiv preprint arXiv:2302.09432*, 2023.

[28] Pekka Malo, Ankush Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796, 2014.

[29] MosaicAI. Introducing dbrx: A new state-of-the-art open llm. `https://www.databricks.com/blog/introducing-dbrx-new-state-art-open-llm`, 2024. Accessed: 2024-05-21.

[30] National Institute of Standards and Technology (NIST). Reuters dataset at trec. `https://trec.nist.gov/data/reuters/reuters.html`, 2024. Accessed: 2024-02-01.

[31] OpenAI. Gpt-4 technical report, 2023.

[32] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback, 2022. *URL https://arxiv. org/abs/2203.02155*, 13, 2022.

[33] Xue Bin Peng, Erwin Coumans, Tingnan Zhang, Tsang-Wei Lee, Jie Tan, and Sergey Levine. Learning agile robotic locomotion skills by imitating animals. *arXiv preprint arXiv:2004.00784*, 2020.

[34] S. Raeid, R. Frank, K. Kato, and N. Vinden. Nifty financial news headlines dataset, 2024. Manuscript under review.

[35] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

[36] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[37] Claude Sammut and Geoffrey I. Webb, editors. *TF–IDF*, pages 986–987. Springer US, Boston, MA, 2010.

[38] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[39] Raj Sanjay Shah, Kunal Chawla, Dheeraj Eidnani, Agam Shah, Wendi Du, Sudheer Chava, Natraj Raman, Charese Smiley, Jiaao Chen, and Diyi Yang. When flue meets flang: Benchmarks and large pre-trained language model for financial domain. *arXiv preprint arXiv:2211.00083*, 2022.

[40] ShareGPT. Sharegpt. `https://sharegpt.com`, 2024. Accessed: 2024-02-01.

[41] Yueqi Song, Catherine Cui, Simran Khanuja, Pengfei Liu, Fahim Faisal, Alissa Ostapenko, Genta Indra Winata, Alham Fikri Aji, Samuel Cahyawijaya, Yulia Tsvetkov, et al. Global-Bench: A benchmark for global progress in natural language processing. *ArXiv preprint*, abs/2305.14716, 2023.

[42] Yejun Soun, Jaemin Yoo, Minyong Cho, Jihyeong Jeon, and U Kang. Accurate stock movement prediction with self-supervised learning from sparse noisy tweets. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1691–1700. IEEE, 2022.

[43] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

[44] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

[45] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[46] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *ArXiv preprint*, abs/2302.13971, 2023.

[47] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[48] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023.

[49] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

[50] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions, 2023.

[51] Huizhe Wu, Wei Zhang, Weiwei Shen, and Jun Wang. Hybrid deep sequential modeling for social text-driven stock prediction. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 1627–1630, 2018.

[52] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.

[53] Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. Pixiu: A large language model, instruction data and evaluation benchmark for finance. *arXiv preprint arXiv:2306.05443*, 2023.

[54] Frank Z Xing, Erik Cambria, and Roy E Welsch. Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1):49–73, 2018.

[55] Yumo Xu and Shay B Cohen. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1979, 2018.

[56] Yi Yang, Mark Christopher Siy Uy, and Allen Huang. Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:2006.08097*, 2020.

[57] Zhuosheng Zhang, Hanqing Zhang, Keming Chen, Yuhang Guo, Jingyun Hua, Yulong Wang, and Ming Zhou. Mengzi: Towards lightweight yet ingenious pre-trained models for chinese. *arXiv preprint arXiv:2110.06696*, 2021.

[58] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Appendices

Appendix Contents

A Definitions and Terminology

Markov Decision Process (MDP) An MDP is defined by a tuple $(S, A, T, R, \gamma, p_0)$ where S is a set of states (state space), A is a set of actions, $T : S \times A \to \Pi(S)$ is the transition function, $R : S \to \mathbb{R}$ is the reward function, $\gamma \in [0, 1]$ is the discount factor, and $p_0 : S \to [0, 1]$ is the distribution over initial states. A policy over an MDP is a function $\pi : S \to \Pi(A)$, and is optimal if it maximizes the expected discounted sum of rewards.

$$\mathcal{L} = \mathbb{E}_{\pi,T}\left(\sum_{s_i \in \tau} \gamma^i R(s_i)\right), \tag{7}$$

where $\tau = (s_0, a_0, \ldots, s_T)$ is a trajectory.

Partially Observable Markov Decision Process (POMDP) A POMDP is a generalisation of an MDP defined by the tuple $(S, A, T, O, \omega, R, \gamma, p_0)$ where O is a set of observations and $\omega : S \to \Pi(O)$ is the *observation function*. An agent in a POMDP thus only receives an observation (i.e., partial information about the state) rather than the actual state of the environment. Therefore, policies on POMDPs act based on the history of observations received and actions taken at timestep t.

Belief MDPs Since using the complete history is impractical, many algorithms instead use *belief states* $b : O \to \Pi(S)$, which is a probability distribution over possible states updated at each timestep, given history h_t comprising of previous observations. Intuitively, it can be thought of an agent maintaining a 'belief' – a probability distribution over what it thinks the true state of the environment might be.

The belief update after taking the action $a \in A$ and receiving observation $o \in O$ is done through the following equation:

$$\begin{aligned} b_o^a(s') &= P(s' \mid b, a, o) \\ &= \frac{\omega(s', o) \sum_s T(s, a, s')\, b(s)}{P(o \mid b, a)} \quad \forall s' \in S, \end{aligned} \tag{8}$$

where $P(o \mid b, a) = \sum_{s'} \omega(s', o) \sum_s T(s, a, s')\, b(s)$.

We can formulate any POMDP problem as an MDP over belief states **(author?)** [20]. Thus, an agent's belief state at time t, b_t can be seen as a sufficient statistic of the history h_t towards deciding optimal actions.

B Preliminary Experiments and Results

B.1 Experimental Setup

We demonstrate the efficacy of our proposed adaptive algorithm/framework using various SoTA class large language models on the *financial market movement* prediction task.

Task The Financial Market Movement (FMM) prediction task for experts' evaluation can be defined as a ternary or binary market movement direction *classification task* among the labels' set C: { *'Fall'*, *'Neutral'*, *'Rise'* } conditioned on a history (or, expert memory) of window size H (i.e., $P_{w_{t+1}|w_{t-H:t}}$) – similar to the auto-regressive or causal generative language model (causal LM) training objective.

Experts We set up a diverse list of SoTA general purpose instruction-tuned LLMs as experts for the experiments on our proposed adaptive algorithms [1, 2]. For single LLM experts, we use Meta's open-weights models: Llama-2 (7B, 70B), Llama-3(8B, 70B) [45]. For mixture of experts (MoE) architecture models, we pick two of the current SoTA open-weights models: Mixtral (8x7B) [19] – which is a mixture of 8 Mistral (7B) [18] models – and DBRX-Instruct [29] introduced by DataBricks with 132B total parameters and a mixture of 16 (fine-grained, smaller, 65x more combinations of) experts. For evaluation, we deployed these open-weights models as vLLM [23] OpenAI compatible API endpoints and ran the dataset queries against them. We use API/model configurations like *guided-choice*, *max-tokens* to format class label converged expert responses alongside specific prompt instructions. Addtionally, we use the closed-source, latest variant of the GPT-4 [31] class of models: GPT4o, using the OpenAI API. These collection of experts are leading foundation models on current performance benchmarks on language understanding (MMLU [17]), programming (HumanEval [7]), math (GSM8K [10]) tasks and other relevant concurrent LLM benchmarks [41, 25].

Datasets For real-world experiments on the defined FMM task, we use the US equities market movement (NYSE ticker: $SPY) dataset NIFTY ($\mathcal{D}_{LM}$) [34]. Its test split statistics are tabulated in Table 1.

Each sample of the $\mathcal{D}_{LM}$ contains high-quality, processed (one-turn) conversational queries for an expert instruction fine-tuned LLM, where a query, x_q^t, comprises a prompt x_p^t and a response x_r^t, i.e., $x_q^t = (x_p^t; x_r^t)$ corresponding to a day (or time-step) t.

Table 1: Statistics of NIFTY test split

Category	Statistics
Number of days ($\boldsymbol{T}$) / increment (Δt)	**317** / 1
Label support (Fall / Neutral / Rise)	73 / 143 / 101
Date range (start to end)	2019-02-13 to 2020-09-21

For evaluation, at each time step t, an expert LLM is prompted (x_p^t) to predict the market movement the following day (i.e., $t+1$), based on the market's current contextual information (relevant financial news headlines and the market's financial numerics (like the standard OHLCV and common technical indicators) from past few days capturing trends). Fig. 4 depicts a snapshot of an expert prompt x_p^t for elucidation. Please see Fig. 5 in Appendix §C.1 for details.

B.2 UNReaL Results

We name our LLM policy trained using the RLMF alignment loss as **UNReAL**: *Underpinning News Reward Augmented Learning in Large Language Models*. Table 2 shows our results on the NIFTY (*test split*), in comparison to other SOTA language expert models. In Table 3 we compare SM classification accuracies on base LLaMA models, our model finetuned on the NIFTY dataset, and models finetuned on similar SM datasets from the FLARE Benchmark (Described in §C.3).

Estée Lauder Cuts Profit Goals as Coronavirus Slows Travel Sales | Russia Blocks OPEC Response to Coronavirus | Yum China Shows Coronavirus Outbreak Curbs China's Consumption | Hedge-Fund Billionaire's Deal for Mets Collapses | Fed's Quarles Calls Current Stance on Interest Rates Appropriate |Pinterest's Revenue Topped $1 Billion in 2019 |NYSE Owner Abandons Potential eBay Deal | T-Mobile Projects More Customer Gains in 2020 | Aurora Cannabis Chief Executive To Depart Amid Layoffs | Meredith Shares Rally as Publishing Giant Digests Time Inc. | CBD Producer GenCanna Files for Bankruptcy | Risky Corporate Debt to Take Center Stage in 2020 Stress Tests | Tyson Feels Weight of Lower Poultry Prices | China Tariff Relief Boosts Stock Market | Shale Gas Swamps Asia, Pushing LNG Prices to Record Lows | FAA Flags Warning-Light Problem with 737 MAX | Juul Raises $700 Million From Investors | Shares of NYSE Owner Slide on Fresh eBay Deal Jitters | Deutsche Bank Shares Rally on Capital Group Stake | Kellogg Lowers Expectations for 2020 | New York Times Posts Strong Subscription Growth | Mnuchin Says U.S. 2020 Growth to Be Less Than 3% Due to Boeing | ArcelorMittal Posts Earnings Beat Despite Tough Times for Steelmakers | Canadian Antitrust Officials Probe Farm Giants | Zantac Recall Weighs on Sanofi's Earnings |News Corp Posts Lower Profit, Revenue |

Figure 4: A snapshot of the 'news' key value on date: 2020-02-06, at the upstart of the global coronavirus epidemic. Our π_{LM} policy's prompt is composed of task instruction as query prefix, market context, and this news value concatenated: $s.t.\ x_p \leftarrow (x_{instruction}; x_{context}; x_{news})$. The semantic text colors red, and green conveys negative and positive sentiments. The day's market relevant news was dominated by mostly negative sentiments.

Table 2: Performance of our model **UnREAL** using the RLMF adaptive pipeline compared with a collection of SOTA models on the NIFTY (*test split*).

	LLM Experts							Adaptive Execution
Metrics ↑	Llama-2 7b-chat	Llama-2 70b-chat	Llama-3 8B-Instruct	Llama-3 70B-Instruct	Mixtral-8x7B Instruct-v0.1	DBRX Instruct	OpenAI GPT-4o	UnREAL (ours)
Acc	0.27	0.37	0.39	0.30	0.33	0.34	0.37	**0.72**
F1	0.22	0.33	0.35	0.20	0.34	0.34	0.34	**0.71**

Table 3: Performance of Llama-2-7b-chat and Llama-3-8B-Instruct base models with (SFT LoRA adapter) variants on the NIFTY Stock Price Movement Prediction Task (*test* split).

	Llama-2-7b-chat					**Llama-3-8B-Instruct**				
Metrics ↑	Base	+nifty	+acl18	+bigdata22	+cikm18	Base	+nifty	+acl18	+bigdata22	+cikm18
F1 Score	0.22	0.28	0.20	**0.29**	0.27	0.34	**0.36**	0.19	0.23	0.24
Accuracy	0.27	**0.45**	0.25	0.29	0.27	0.39	**0.41**	0.26	0.26	0.28

Discussions The results presented in Table 2 demonstrate the superior performance of UnREAL, using the RLMF adaptive pipeline when comparing results to other SOTA language models on the NIFTY test set. UnREAL achieves a substantial increase in both accuracy (0.72) and F1 score (0.71), outperforming every model including OpenAI's newest model, GPT-4o. This overwhelming improvement highlights the effectiveness of the RLMF loss in enhancing the model's capability to predict stock price movements accurately. Furthermore, in Table 3 we observe LLaMA models finetuned on NIFTY and evaluated on NIFTY-test in general outperform base and FLARE models trained and evaluated on their corresponding datasets. This leads credence to the hypothesis that the NIFTY dataset is more rich in pertinent information for stock market movement tasks.

C Datasets and Benchmarks

C.1 NIFTY Dataset

The **N**ews-**I**nformed **F**inancial **T**rend **Y**ield (NIFTY) dataset [34] is a processed and curated daily news headlines dataset for the stock (US Equities) market price movement prediction task. NIFTY is comprised of two related datasets, NIFTY-LM and NIFTY-RL. In this section we outline the composition of the two datasets, and comment on additional details.

Dataset statistics Table 4 and Table 5 present pertinent statistics related to the dataset.

C.1.1 NIFTY-LM: SFT Fine-tuning Dataset

The NIFTY-LM prompt dataset was created to finetune and evaluate LLMs on predicting future stock movement given previous market data and news headlines. The dataset was assembled by aggregating information from three distinct sources from January 6, 2010, to September 21, 2020.

Table 4: Statistics and breakdown of splits sizes

Category	Statistics
Number of data points	2111
Number of Rise/Fall/Neutral label	558 / 433 / 1122
Train/Test/Evaluation split	1477 / 317 / 317

Table 5: Date Ranges of news headlines in splits

Split	Num. Samples	Date range
Train	1477	2010-01-06 to 2017-06-27
Valid	317	2017-06-28 to 2019-02-12
Test	317	2019-02-13 to 2020-09-21

Anticipate the direction of the $SPY by analyzing market data and news from 2020-02-06.

(a) Instruction component of a π_{LM} policy query x_q.

date, open, high, • • •, pct_change, macd, boll_ub, boll_lb, rsi_30, • • •, close_60_sma

2020-01-27, 323.03, 325.12, • • •, -0.016, 2.89, 333.77, 319.15, 56.26, • • • , 317.40
2020-01-28, 325.06, 327.85, • • •, 0.0105, 2.59, 333.77, 319.55, 59.57, • • • , 317.78
• • •. • • • •
2020-02-04, 328.07, 330.01, • • •, 0.0152, 1.3341, 333.60, 321.26, • • •, 319.41
2020-02-05, 332.27, 333.09, • • •, 0.0115, 1.7247, 334.15, 321.73, • • •, 319.82

(b) The market's **history** is provided as the past t days of numerical statistics like the (OHLCV) price (in blue) and common technical indicators (in orange) (e.g. moving averages) data.

Figure 5: Breaking down the instruction or prompt prefix, and market context components of a prompt, x_p.

The compilation includes headlines from The **Wall Street Journal** and **Reuters News**, as well as market data of the $SPY index from **Yahoo Finance**. The NIFTY-LM dataset consists of:

- **Meta data**: Dates and data ID.
- **Prompt** (x_p): LLM question ($x_{question}$), market data from previous days ($x_{context}$), and news headlines (x_{news}).
- **Response**: Qualitative movement label (x_r) $\in \{Rise, Fall, Neutral\}$, and percentage change of the closing price of the $SPY index.

To generate LLM questions, ($\boldsymbol{x_{question}}$), the authors used the self-instruct [50] framework and OpenAI GPT4 to create 20 synthetic variations of the instruction below:

> Create 20 variations of the instruction below.
> Examine the given market information and news headlines data on DATE to forecast whether the $SPY index will rise, fall, or remain unchanged. If you think the movement will be less than 0.5%, then return 'Neutral'. Respond with Rise, Fall, or Neutral and your reasoning in a new paragraph.

Where `DATE` would be substituted later, during the training phase with a corresponding date.

Context The key 'context' ($\boldsymbol{x_{context}}$) was constructed to have newline delimited market metrics over the past T ($\approx$ 10) days (N.B. Not all market data for the past days for were available and therefore prompts might have less than 10 days of market metrics.).

Table 6 show the details of financial context provided in each day's sample.

News Headlines ($\boldsymbol{x_{news}}$): Final list of filtered headlines from the aggregation pipeline. The non-finance related headlines were filtered out by performing a similarity search with SBERT model,

Table 6: Summary of the dataset columns with their respective descriptions.

Column Name	Description
Date	Date of the trading session
Opening Price	Stock's opening market price
Daily High	Highest trading price of the day
Daily Low	Lowest trading price of the day
Closing Price	Stock's closing market price
Adjusted Closing Price	Closing price adjusted for splits and dividends
Volume	Total shares traded during the day
Percentage Change	Day-over-day percentage change in closing price
MACD	Momentum indicator showing the relationship between two moving averages
Bollinger Upper Band	Upper boundary of the Bollinger Bands, set at two standard deviations above the average
Bollinger Lower Band	Lower boundary, set at two standard deviations below the average
30-Day RSI	Momentum oscillator measuring speed and change of price movements
30-Day CCI	Indicator identifying cyclical trends over 30 days
30-Day DX	Indicates the strength of price trends over 30 days
30-Day SMA	Average closing price over the past 30 days
60-Day SMA	Average closing price over the past 60 days

"all-MiniLM-L6-v2" [36]. Each headline was compared to a set of artificially generated financial headlines generated by GPT-4, with the prompt *"Generate 20 financial news headlines"*. Headlines with a similarity score below 0.2, were excluded from the dataset. To respect the prompting 'context length' of LLMs, in instances where the prompt exceeded a length of 3000 words, a further refinement process was employed. This process involved the elimination of words with a tf-idf [37] score below 0.2 and truncating the prompt to a maximum of 3000 words.

It is also important to note that the dataset does not encompass all calendar dates within the specified time range. This limitation emanates from the trading calendar days, and absence of relevant financial news headlines for certain dates.

Label $(\boldsymbol{x_r})$: The label is determined by the percentage change in closing prices from one day to the next, as defined in equation 9. This percentage change is categorized into three labels: {Rise, Fall, Neutral}, based on the thresholds specified in equation 10.

$$PCT_{\text{change}} = \left(\frac{\text{Closing Price}_t - \text{Closing Price}_{t-1}}{\text{Closing Price}_{t-1}} \right) \times 100\% \tag{9}$$

$$x_r = \begin{cases} \text{Fall} & \text{if } PCT_{\text{change}} < -0.5\% \\ \text{Neutral} & \text{if } -0.5\% \leq PCT_{\text{change}} \leq 0.5\% \\ \text{Rise} & \text{if } PCT_{\text{change}} > 0.5\% \end{cases} \tag{10}$$

C.2 NIFTY-RL: Preferences Dataset

The preference dataset is a variation of the fine-tuning dataset and it is designed for alignment training of LLMs using reward model. In NIFTY-RL, labels are omitted and replaced with chosen and rejected results. The chosen result is a label corresponding to a rise, a fall or neutral movement in the stock market and is equivalent to the response in NIFTY-LM. The rejected result is a random label not equal to the chosen label.

- **Metadata**: Includes dates and data identifiers.
- **Prompt** (x_p): Includes an LLM instruction ($x_{question}$), preceding market data ($x_{context}$), and relevant news headlines (x_{news}).
- **Chosen Result**: A qualitative movement label (x_r) from $\{Rise, Fall, Neutral\}$ indicating the predicted market trend.
- **Rejected Result**: A label ($\overline{x}_r$) randomly selected from $\{Rise, Fall, Neutral, Surrender\} \setminus \{x_r\}$, representing an incorrect market prediction.

C.3 FLARE Benchmark Datasets

Stock Movement Prediction Datasets and Tasks: Flare-SM tasks **FLARE** proposed by [53], extends to include one financial prediction task – the **CIKM** dataset [51] as an evaluation task among (four) other general financial NLP tasks. Under the hood, this benchmark is a fork of the '*lm-eval*' harness [11] with addendums. Other stock price movement prediction from social dataset include what is referred to as *ACL18* (or, 'acl18') in this paper is essentially the **StockNet** [55] dataset which comprises of stock tweets of 88 stock tickers from 9 financial market industries from Twitter over two years (from 2014-2015) aligned with their corresponding historical price data. **BigData22** [42] is another more recent tweets dataset comprising of tweets about 50 stock tickers during the period 2019-07-05 to 2020-06-30.

Table 7: Summary of Flare stock price movement datasets. The 'Stocks' column indicates the total number of different stock tickers referenced. The 'Tweets' and 'Days' columns represent the number of tweets and days respectively in each dataset.

Data	Stocks	Tweets	Days	Start Date	End Date
ACL18	87	106,271	696	2014-01-02	2015-12-30
BigData22	50	272,762	362	2019-07-05	2020-06-30
CIKM18	38	955,788	352	2017-01-03	2017-12-28

D Additional Related Work

In this section we enclose works encompassing ML/AI/RL based techniques for financial market downstream tasks, specifically tasks pertaining to market forecasting (that can be movement prediction, or, regression tasks of price forecasting).

D.1 History of using PLMs, then LLMs in the Financial domain

Many PLMs for the financial domain have been proposed by continual pre-training PLMs with large-scale financial texts. [3] proposed the first financial PLM called FinBERT that pre-trained BERT [21] with open released financial corpus such as TRC2financial [30] and Financial Phrase Bank [28]. FinBERT outperforms neural network methods such as LSTM in financial sentiment classification tasks. [56] further proposed FinBERT by pre-training BERT with a 4.9 billion tokens financial communication corpus, which outperforms BERT on three financial sentiment classification datasets. [39] proposed FLANG, a financial PLM with BERT and ELECTRA [9] as the backbone. Besides English, financial PLMs in other languages, such as Chinese, were also proposed, such as Mengzi-fin [57] and BBT-FinT5 [27].

Financial LLM Evolution Latest, [52] proposed BloombergGPT, the first financial large language model with 50 billion parameters, that is pre-trained with mixed datasets from the general and financial domain. However, neither the model nor pre-trained domain datasets are released. The model is also not instruction-following like other LLMs such as ChatGPT and GPT-4. Meta AI's LLaMA [45] was the first open-source LLM with parameters ranging from 7B and 13B to 65B that gained widespread traction in the research and open-source community. LLaMA-13B has comparable and even better performance than GPT-3 [5] with 175B parameters on common sense reasoning tasks. Following efforts have been proposed to improve LLaMA for instruction following like ChatGPT, by instruction tuning. Such as the Alpaca [44] model by fine-tuning LLaMA-7B with 52K instruction-following samples generated with the self-instruct method [49]. [8] proposed Vicuna-13B by fine-tuning LLaMA-13B with 70K conversation data from ShareGPT [40]. It can generate better answers to user's questions compared with Alpaca. However, there are no open-sourced LLMs and instruction-tuning data entirely focused on the financial domain. FinMA [53] series of model along with the recently release Flare benchmark aims to fill this void, however, these models uses (Llama 1 [46]) as the base model that were not tuned to be instruction following assistants.

Natural language based financial forecasting We direct interested readers to survey papers like [54] that details recent related works. We note that while financial news has long been used for financial forecasting, however, majority of such works first does (variants of) sentiment classification, i.e. attaching an (human opinionated) label of '*goodness*' of the news prior to feeding that (opinionated) label for downstream forecasting, or prediction pipeline. We think such approaches are ineffective if not naive. The **sentiment** of this sentence (as we perceive it): "*The new Apple iPhones got horrendous reviews*" is **irrelevant**; labelling (if any) should come from the market. In this case, the sentiment is positive if Apple's stock price goes up. [4]'s related work show that sentiment has little predictive power for near-term future stock returns. Further, evidence did not support the conventional wisdom that sentiment primarily affects individual investors and small stocks. [2] explores whether Internet stock message boards can move markets.