Collaborative Editable Model

Kaiwen Tang [*1] **Aitong Wu** [*2] **Guangda Sun** [1 2]

Abstract

Vertical-domain large language models (LLMs) play a crucial role in specialized scenarios such as finance, healthcare, and law; however, their training often relies on large-scale annotated data and substantial computational resources, impeding rapid development and continuous iteration. To address these challenges, we introduce the Collaborative Editable Model (CoEM), which constructs a candidate knowledge pool from user-contributed domain snippets, leverages interactive user–model dialogues combined with user ratings and attribution analysis to pinpoint high-value knowledge fragments, and injects these fragments via in-context prompts for lightweight domain adaptation. With high-value knowledge, the LLM can generate more accurate and domain-specific content. In a financial information scenario, we collect 15k feedback from about 120 users and validate CoEM with user ratings to assess the quality of generated insights, demonstrating significant improvements in domain-specific generation while avoiding the time and compute overhead of traditional fine-tuning workflows.

1. Introduction

Large Language Models (LLMs) have demonstrated exceptional performance in general-purpose applications and have achieved notable success in *vertical domains*, such as finance, healthcare, law, and scientific research, by providing precise domain knowledge and specialized text generation (Li et al., 2023; Ren et al., 2025; Lin et al., 2025; Li et al., 2024). However, the development of such vertical-domain models typically depends on large-scale annotated datasets and substantial computational resources (Wu et al., 2023; Gururangan et al., 2020). This dependency results in prolonged development cycles and slow iteration rates, making it challenging to meet the rapidly evolving demands of real-world applications.

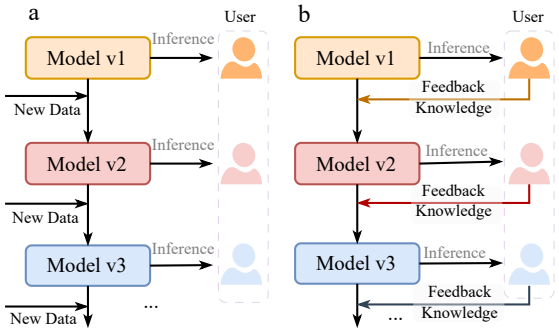


Figure 1. Comparison of Traditional Model Updating vs. Collaborative User-driven Updating. (a) Traditional model updates rely on externally provided training data; (b) Collaborative updates leverage user contributions to refine the model.

Traditional domain adaptation methods have significant drawbacks. Supervised fine-tuning requires expensive annotation efforts and large labeled corpora, while retrieval-augmented generation (RAG) (Siriwardhana et al., 2023) leverages external knowledge sources but depends on comprehensive, up-to-date knowledge bases (Zhang et al., 2024; Zheng et al., 2024; Susnjak et al., 2025). These approaches rely heavily on external, annotated inputs – data acquisition and labeling thus remain time-consuming and costly, and the limited volume of annotated domain examples constrains the benefits of scaling laws as model capacity grows.

Given these challenges, we explore the possibility of developing high-quality vertical-domain models without relying on manually annotated external data. Collaborative knowledge platforms, such as wiki projects, continuously evolve through user contributions, revealing the potential of crowd-sourced expertise as an ongoing learning signal (Ebersbach et al., 2008). This suggests that user–model conversations can serve as the universal interface to both utilize and contribute to the model, transforming them into a valuable internal source of domain knowledge.The concept difference is shown in Figure 1. Although a single user session may combine seeking assistance and contributing, and lack the structure and quality of traditional annotations, these

[*]Equal contribution [1]Department of Computer Science, National University of Singapore, Singapore, Singapore [2]Vak Labs. Correspondence to: Guangda Sun <sung@comp.nus.edu.sg>.

Accepted at the ICML 2025 Workshop on Collaborative and Federated Agentic Workflows (CFAgentic@ICML'25), Vancouver, Canada. July 19, 2025.

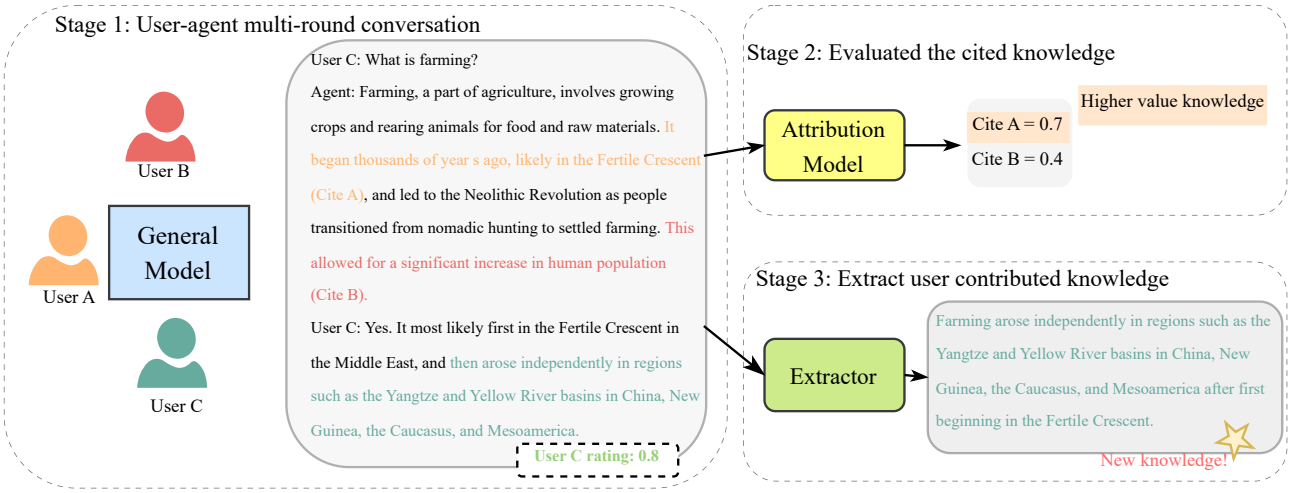


Figure 2. Overview of the Collaborative Editable Model's workflow, illustrating how user contributions drive model adaptation through multi-turn interactions and an attribution mechanism. User identities remain confidential to protect privacy during citation in generated text. The general model will be continuously updated with high-value knowledge.

conversations inherently encode valuable insights shared during interactions (Chen et al., 2024). These insights can be extracted, evaluated, and leveraged to improve the model. However, current LLMs treat each conversation independently, without a mechanism for reintegrating conversational data into subsequent learning.

To address this gap, we present the *Collaborative Editable Model (CoEM)*, a framework designed to enable the continuous improvement of vertical-domain models through user–model interactions. CoEM aggregates domain-relevant information fragments from user-contributed snippets and conversation exchanges, creating a continuously evolving knowledge pool. Users then engage in iterative dialogue sessions with the LLM, providing ratings on each generated response or insight. As the user ratings are for the whole responses, we apply attribution analysis to measure each input fragment's contribution to overall performance, automatically identifying high-value knowledge. The fragments with high value after several iterations will be dynamically updated into the model, allowing for rapid domain adaptation without the need for extensive fine-tuning or external data. At the same time, the knowledge pool is still updating via extracting new information from user-model dialogues. The overview of the CoEM workflow is shown in Figure 2.

CoEM offers several key advantages. First, it is **cost-efficient**, eliminating the need for costly, manually annotated data by relying on user-generated content and interactions, which significantly reduces data acquisition costs. Second, CoEM is **user-driven and responsive** to feedback. End users can identify issues or gaps in the model's performance and provide direct feedback, enabling real-time adjustments and ensuring that the system remains relevant and up-to-date. Finally, CoEM is **parallelized and scalable**. As more users engage with the system, the model benefits from an increasing volume of valuable contributions, supporting rapid updates and continuous improvement.

In this paper, we apply CoEM to the financial domain for initial validation. We collect a set of financial news articles to construct the initial knowledge pool. A general-purpose LLM is used to generate summaries with insights for several relevant news pieces. These summaries are then presented to users for feedback. By applying attribution analysis to the user ratings, we assess the value of each knowledge fragment based on its contribution to the generated insights. These value scores are then compared to those obtained from a state-of-the-art (SOTA) financial LLM (Liu et al., 2023) to validate the reliability of our data and the effectiveness of our method. Through this process, we demonstrate CoEM's ability to incorporate valuable user-generated knowledge into the model's context, facilitating rapid domain-specific refinement.

Our contributions are summarized as follows:

- We introduce the CoEM, a framework that enables the continuous improvement of vertical-domain models through user–model interactions, eliminating the need for large-scale annotated data.
- Through the innovative contribution attribution mechanism, we address the technical challenges of extracting valuable knowledge from unstructured user dialogues, enabling rapid domain adaptation for the model.
- The effectiveness of the CoEM framework is initially

validated through experiments in the financial news domain, with future work focusing on enhancing the model's ability to learn from user contributions through model editing or reinforcement learning.

2. Related Work

2.1. Domain Adaptation for LLMs

Early efforts in vertical-domain adaptation have predominantly relied on large-scale supervised fine-tuning. Researchers typically continue pre-training or fine-tuning a general-purpose LLM on domain-specific corpora to achieve specialized performance (Gu et al., 2021; Que et al., 2024); however, this approach still demands substantial annotated data and computational resources. To reduce these costs, recent work has introduced lightweight techniques, such as low-rank adaptation (LoRA) (Hu et al., 2022; Mao et al., 2025), adapter modules (Li et al., 2022), and prompt tuning (Ge et al., 2023; Bai et al., 2024; Fahes et al., 2023), which update only a small set of additional parameters or employ plug-and-play adaptation layers, leaving most pre-trained weights untouched, and enabling rapid convergence in new domains. Retrieval-augmented generation frameworks (Guu et al., 2020) further enhance domain performance by integrating external knowledge retrieval into the generation process, dynamically incorporating up-to-date documents for tasks like question answering and summarization, though they still face challenges in retriever maintenance and latency.

2.2. Interactive Editable Models

As human–machine collaboration paradigms evolve, editable and updatable language models have emerged as a key research direction. One strand of work employs reinforcement learning from human feedback (RLHF), using explicit user ratings or click behaviors to refine the model's generative policies and improve alignment. (Bai et al., 2022; Kaufmann et al., 2023) Another focuses on fragment-level knowledge injection (Czekalski & Watson, 2024; Song et al., 2025). These methods identify "high-value" text fragments that most strongly influence outputs and prioritize their reuse. At the same time, crowdsourced snippets provide a flexible and rich source of domain knowledge, and by borrowing ideas from collaborative filtering, preferences can propagate across multiple agents to enable multi-directional preference diffusion. Although existing editable-model research has demonstrated local weight modifications and fact updates, it has yet to integrate explicit user feedback, fragment selection, and multi-agent collaborative learning into a unified framework. (Casper et al., 2023) In contrast, CoEM orchestrates multi-turn user–model dialogues with rating feedback, enabling lightweight, sustainable domain iteration and real-time knowledge updating.

3. Method

3.1. Problem Definition

We begin with two models: a pretrained model $\mathcal{M}_p$, which is a general-purpose model without domain-specific adaptation, and a vertical domain model $\mathcal{M}_v$, which represents a model tailored to a specific vertical domain. $\mathcal{M}_p$ predicts the distribution P_p of tokens based on a given sequence $\mathbf{s}$:

$$\mathcal{M}_p(\mathbf{s}) = P_p(t|\mathbf{s}) \tag{1}$$

While $\mathcal{M}_v$ predicts the domain-specific distribution P_v based on the same sequence:

$$\mathcal{M}_v(\mathbf{s}) = P_v(t|\mathbf{s}) \tag{2}$$

As we want to get a domain-specific model efficiently from the general model, our objective is to minimize the difference between the predictions of these two models by aligning $\mathcal{M}_p$ with $\mathcal{M}_v$:

$$\mathcal{L} = \min_{\mathbf{s}} |P_p(t|\mathbf{s}) - P_v(t|\mathbf{s})| \tag{3}$$

In traditional fine-tuning approaches, P_v is explicit, requiring annotated training data to guide the model. However, in our approach, P_v is implicit, as we do not rely on manually annotated data or pre-built knowledge bases. Instead, we rely on the vast amount of user feedback and contributions to guide the model's learning direction.

3.2. User Contribution Modeling

In our framework, user contributions are categorized into two complementary forms: user feedback and user knowledge. Specifically, for model-user dialogue session $\mathbf{d}$, **User feedback contribution** quantifies the user's evaluation of the model's output quality and domain relevance, typically expressed as a scalar score $r \in [0, 1]$. This rating indicates the relevance and usefulness of the generated content with respect to the target domain. This feedback score serves as an indicator of the effectiveness of the supporting knowledge. **User knowledge contribution** represents the domain-relevant information fragments extracted from the user's input during the session, each associated with a value score $v_i \in [0, 1]$.

Assumption 3.1. We assume that the user feedback score $r \in [0, 1]$ serves as a soft reinforcement learning reward signal for a model output $\mathbf{d}_m$, reflecting the similarity between the predicted token distribution $P_p(t|\mathbf{s})$ and the target vertical domain distribution $P_v(t|\mathbf{s})$. Specifically,

$$r = \mathcal{R}(\mathbf{d}_m) \quad \text{with} \quad r = 1 \iff P_p(t|\mathbf{s}) = P_v(t|\mathbf{s}),$$

Moreover, for any $r_1 > r_2$, it holds that

$$\rho\big(P_p^{(r_1)}(t|\mathbf{s}), P_v(t|\mathbf{s})\big) < \rho\big(P_p^{(r_2)}(t|\mathbf{s}), P_v(t|\mathbf{s})\big),$$

where $\mathcal{R}(\cdot)$ denotes the reward function induced by user feedback, and $\rho(\cdot,\cdot)$ denotes a distance metric between distributions. Thus, higher feedback corresponds to model outputs closer to the target domain distribution.

The overall objective is to guide model adaptation by aligning the weighted sum of user knowledge contributions with the vertical domain distribution:

$$P_v(t|\mathbf{s}) = \sum_i v_i P_i(t|\mathbf{s}) \tag{4}$$

This formalization enables the framework to utilize user-generated content and real-time feedback as continuous learning signals. Over successive interactions, the model incrementally evolves to better capture domain-specific knowledge, eliminating the need for extensive manually annotated data and facilitating rapid adaptation within vertical domains.

3.3. CoEM Knowledge Pool

Consider a multi-round dialogue session $\mathbf{d}$ between the model and the user. During this session, the model generated output $\mathbf{d}_m$ is supported by a collection of domain-specific knowledge fragments $\{k_1, k_2, \dots\}$ retrieved from the current knowledge pool $\mathbf{K}$. Each knowledge fragment k_i is associated with a value score v_i, representing its estimated relevance and utility within the vertical domain.

After receiving the model's response, the user provides the feedback score $r \in [0,1]$ as introduced in Section 3.2. To quantify the contribution of individual knowledge fragments to the overall feedback, an attributor $\mathcal{A}$ is employed to assign a contribution weight p_i to each fragment k_i. The adjusted contribution value for fragment k_i in the current session is then computed as

$$v_i' = p_i \cdot r. \tag{5}$$

Treating each dialogue session as an iteration, the value scores of the knowledge fragments are updated according to an exponential moving average scheme (Lucas & Saccucci, 1990):

$$v_i \leftarrow (1-\alpha)v_i + \alpha v_i', \tag{6}$$

Where $\alpha \in (0,1)$ denotes the learning rate that controls the magnitude of the update. Knowledge fragments whose value scores fall below a predefined threshold θ after n iterations are pruned from the knowledge pool, as they are considered insufficiently relevant and useful to the domain.

Concurrently, the user's input during the dialogue session, denoted $\mathbf{d}_u$, is processed by a knowledge extractor $\mathcal{E}$ to identify potential new domain-relevant knowledge fragments $\{u_1, u_2, \dots\}$. These newly extracted fragments are incorporated into the knowledge pool $\mathbf{K}$, thereby expanding the pool.

We initialize newly added knowledge fragments with a value score of 1, reflecting an *optimistic initialization* strategy (Lobel et al., 2022). This approach encourages the model to treat new knowledge as valuable initially, while the exponential moving average update bounds the scores below 1 and allows subsequent feedback to adjust them dynamically. Optimistic initialization is commonly used in reinforcement and online learning to promote exploration and avoid premature dismissal of novel information. The whole process of updating the CoEM knowledge pool is shown in Algorithm 1.

Algorithm 1 CoEM Knowledge Pool Update

Input: General model $\mathcal{M}_p$, Knowledge pool $\mathbf{K} = \{(k_i, v_i)\}$, dialogue session $\mathbf{d} = (\mathbf{d}_m, \mathbf{d}_u)$, learning rate α, value threshold θ, attribution method $\mathcal{A}$, knowledge extractor $\mathcal{E}$.
Output: Updated knowledge pool $\mathbf{K}$
$\mathbf{d}_m \leftarrow \mathcal{M}_p(\hat{K}), \hat{K} \subseteq \mathbf{K}$
$r \leftarrow \mathcal{R}(\mathbf{d}_m), r \in [0,1]$ {Get user feedback on $\mathbf{d}_m$}
Attributor:
$p_i = \mathcal{A}(\mathbf{d}_m, k_i), k_i \in \hat{K}$
for $k_i \in \hat{K}$ **do**
  $v_i \leftarrow (1-\alpha)\cdot v_i + \alpha \cdot p_i \cdot r$
end for
Extractor:
$u_j = \mathcal{E}(\mathbf{d}_u)$
$v_{u_j} \leftarrow 1 \quad \forall u_j$ {Initialize value scores.}
$\mathbf{K} \leftarrow \mathbf{K} \cup \{(u_j, v_{u_j})\}$
$\mathbf{K} \leftarrow \mathbf{K} \setminus \{k_i \mid v_i < \theta\}$
Return updated knowledge pool $\mathbf{K}$

Through this iterative process of feedback-driven value updating and continuous knowledge extraction, the knowledge pool is dynamically refined and enlarged. This mechanism enables the model to progressively enhance its domain expertise and improve response quality, all achieved through ongoing user interaction without reliance on extensive manual annotation.

3.4. Attribution Mechanism

In this section, we briefly introduce the design of our attribution mechanism. To quantitatively assess the value of knowledge k_i from the model generated text $\mathbf{d}_m$ and user feedback r, CoEM employs an attribution function $\mathcal{A}$ defined as

$$\mathcal{A}(\mathbf{d}_m, k_i) \in [0,1] \tag{7}$$

which represents the strength of the causal influence of knowledge k_i on the generated response $\mathbf{d}_m$. Intuitively, this value measures how much the distribution of $\mathbf{d}_m$ would differ if the model had not incorporated knowledge from k_i.

Figure 3. Attribution score distribution weighted by user feedback.

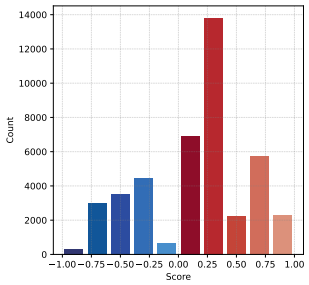


Figure 4. Distribution of knowledge fragment value scores after iterative updates.

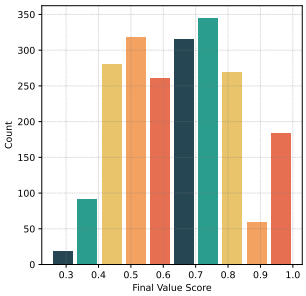


Figure 5. Distribution of knowledge fragment value scores from baseline model.

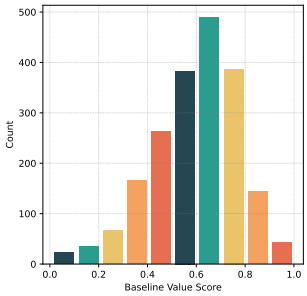
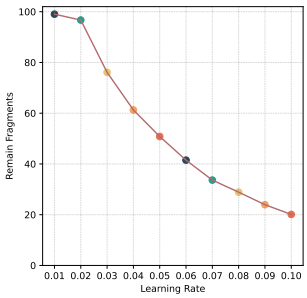


Figure 6. Proportion of high-value knowledge fragments under varying learning rates.



The attribution model satisfies natural boundary conditions:

$$\mathcal{A}(k_i, k_i) = 1 \tag{8}$$

Notably, $\mathcal{A}$ operates without requiring domain-specific knowledge distribution P_v, ensuring broad applicability.

Considering a set of model outputs $\{\mathbf{d}_m^i\}$ generated during interactions and corresponding user feedback scores $\{r_i\} \in [0, 1]$, the value score v_j for each input knowledge u_j is computed as the expected weighted attribution:

$$v_j = \mathbb{E}_{\mathbf{d}_m^i} \left[\mathcal{A}(\mathbf{d}_m^i, u_j) \cdot r_i \right] \tag{9}$$

This expectation aggregates the causal effect of u_j across multiple model responses, modulated by user feedback, yielding a statistically meaningful measure of the input's overall contribution.

4. Experiment

In this work, we performed experiments in the financial domain as a starting point. We used the Gemini-2.0-flash-lite API to access Gemini (Team et al., 2023) as the general model. For the vertical domain model, we employed FinGPT (Liu et al., 2023) from Huggingface[1] built based on LLaMa-3 (Grattafiori et al., 2024).

4.1. Data Construction

We first collected 2,578 knowledge fragments related to finance and cryptocurrency news from a variety of news websites and forums, including Yahoo Finance, MarketWatch.com, Cointelegraph.com, Crypto.news, and so on, in both English and Chinese. These fragments served as the initial knowledge pool for our model.

We then prompted the general model to generate summaries with viewpoints for 3 related fragments in the pool. These summaries were presented to users, who provided feedback in the form of "like" or "dislike" evaluations, which serve as user feedback contributions. The users involved in the feedback process all had relevant knowledge and backgrounds in the finance and cryptocurrency domains. We received a total of 15,040 feedback responses from approximately 120 users, with a "like" to "dislike" ratio of 10,696: 4,344. Importantly, all user data was anonymized, and no personal information was recorded, ensuring the privacy of the contributors.

Then we use another general model as the attributor. Given the model-generated summary with viewpoints and user feedback, the attributor model evaluates the contribution of each news article to the generated summary. The feedback score provided by the user is used to compute the contribution weight for each news article with Equation (6). As we only collect the user rating as likes or dislikes, we assign $r = 1$ to what the user likes and -1 to what the user dislikes. We choose the learning rate α as 0.03.

4.2. Results

High-value knowledge The key step for CoEM is to distinguish which contributions are useful to the model. After the iterative updating of the value score, we regard the knowledge with high scores as valuable knowledge for the model to learning to become a vertical domain model. In the current dataset, after calculating with Equation (5), we got the value score for each iteration. The distribution is shown as Figure 3.

With the value score v' for each iteration, we update the value score for each knowledge fragment in the knowledge pool; the distribution of the total value score is shown in Figure 4. Only 0.14% of the fragments didn't get feedback from the users, with a learning rate of 0.03 and a threshold of 0.5, 76.11% of the fragments remain in the knowledge pool and are marked as high-value knowledge.

[1] https://huggingface.co/FinGPT/fingpt-mt_llama3-8b_lora

Comparison with Vertical Domain Model To illustrate the effectiveness of our user contribution strategy, we also used the vertical domain model to evaluate our knowledge and validate our Assumption 3.1. Specifically, we ask the vertical domain model to rate the relevance and usefulness of each knowledge fragment and compare the results with the value scores in our knowledge pool. The distribution is shown in Figure 5. From FinGPT, 73.62% of the fragments are marked as high-value knowledge. Among all the high-value knowledge in the knowledge pool, 76.11% of them are also recognized by the baseline model, indicating the effectiveness of our methods.

Learning Rate of CoEM We also performed an ablation study on the learning rate to illustrate the reason for picking 0.03 in the experiment. As shown in Figure 6, when the learning rate increases, the proportion of high-value knowledge decreases, so 0.03 could serve as a value to keep enough knowledge in the pool. This may also show that our attributor tends to give lower marks, indicating that the attributor and the threshold require further exploration.

5. Discussion

User Feedback and Attribution Mechanism CoEM heavily depends on integrating user feedback with the attribution model. However, accurately attributing individual user contributions to the model's output remains challenging as we discussed in Section 4.2, especially in multi-turn dialogues where prior context and existing knowledge interact in complex ways. One possible approach to address this challenge is using Shapley value-based attribution, which offers a principled and fair method for quantifying the contribution of each user input, enhancing transparency. Additionally, as shown in Figure 6, the current optimistic initialization strategy may lead to a decrease in attribution scores over time. To mitigate this, future versions of CoEM may incorporate a decay mechanism within the value score update process to better balance score evolution.

Scalability and Further Experiment As CoEM relies on continuous user interaction, scalability is a key concern. With an increasing number of users, the computational overhead for real-time updates and knowledge pool management could become a bottleneck. One possible solution could be to implement incremental learning techniques that allow for more efficient adaptation with minimal computational cost. For instance, leveraging sparse updates rather than full model retraining can drastically reduce resource consumption while preserving model performance. Additionally, the use of distributed learning frameworks can help balance the load across different computing nodes, ensuring scalability without compromising real-time performance. Optimizing these algorithms will be crucial for large-scale deployment.

User Privacy and Data Security A major advantage of CoEM is its inherent respect for user privacy. Unlike traditional models, CoEM eliminates the need for centralized data collection, directly addressing privacy concerns arising from large-scale data usage. However, to ensure the protection of the user's privacy, we need to design the decentralized learning process to keep the user's data local. A possible solution could be integrating differential privacy techniques (Yan et al., 2024), which add noise to the data, ensuring that individual user contributions cannot be traced back to a specific user while still allowing the model to learn effectively from the aggregated knowledge. Additionally, secure multi-party computation (SMPC) (Knott et al., 2021) could be utilized to allow the model to learn from user data without directly accessing sensitive information. These techniques would further reinforce CoEM's privacy-preserving framework, ensuring robust protection for users while maintaining high model performance.

6. Conclusion

We propose the Collaborative Editable Model (CoEM), a novel user-driven framework for efficiently adapting large language models to vertical domains. CoEM maintains a dynamic knowledge pool updated through user contributions and multi-turn dialogue feedback. A core component is the attribution mechanism, which quantifies the impact of each knowledge fragment on model outputs, enabling precise updates to their value scores without relying on costly annotated datasets or extensive fine-tuning.

Experimental results on financial news demonstrate CoEM's effectiveness in identifying and reinforcing high-value domain knowledge, with over 76% agreement with a state-of-the-art vertical domain model. We also study the influence of learning rate on knowledge retention, revealing important trade-offs for practical deployment. Overall, CoEM offers a scalable, privacy-aware, and interactive approach for continuous vertical domain adaptation driven by collaborative user feedback.

Acknowledgment

We would like to thank Liangjun Deng and Wei Zhang for assisting our evaluation. We also thank the anonymous reviewers from CFAgentic @ ICML'25 and our workmates at Vak Labs for their valuable comments.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Bai, S., Zhang, M., Zhou, W., Huang, S., Luan, Z., Wang, D., and Chen, B. Prompt-based distribution alignment for unsupervised domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 729–737, 2024.

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Casper, S., Davies, X., Shi, C., Gilbert, T. K., Scheurer, J., Rando, J., Freedman, R., Korbak, T., Lindner, D., Freire, P., et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

Chen, J., Liu, Z., Huang, X., Wu, C., Liu, Q., Jiang, G., Pu, Y., Lei, Y., Chen, X., Wang, X., et al. When large language models meet personalization: Perspectives of challenges and opportunities. *World Wide Web*, 27(4):42, 2024.

Czekalski, E. and Watson, D. Efficiently updating domain knowledge in large language models: Techniques for knowledge injection without comprehensive retraining. 2024.

Ebersbach, A., Glaser, M., Heigl, R., and Warta, A. *Wiki: web collaboration*. springer science & business media, 2008.

Fahes, M., Vu, T.-H., Bursuc, A., Pérez, P., and De Charette, R. Poda: Prompt-driven zero-shot domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18623–18633, 2023.

Ge, C., Huang, R., Xie, M., Lai, Z., Song, S., Li, S., and Huang, G. Domain adaptation via prompt learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23, 2021.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8342–8360, 2020.

Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M. Retrieval augmented language model pre-training. In *International conference on machine learning*, pp. 3929–3938. PMLR, 2020.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Kaufmann, T., Weng, P., Bengs, V., and Hüllermeier, E. A survey of reinforcement learning from human feedback. *arXiv preprint arXiv:2312.14925*, 10, 2023.

Knott, B., Venkataraman, S., Hannun, A., Sengupta, S., Ibrahim, M., and van der Maaten, L. Crypten: Secure multi-party computation meets machine learning. *Advances in Neural Information Processing Systems*, 34: 4961–4973, 2021.

Li, H., Chen, J., Yang, J., Ai, Q., Jia, W., Liu, Y., Lin, K., Wu, Y., Yuan, G., Hu, Y., et al. Legalagentbench: Evaluating llm agents in legal domain. *arXiv preprint arXiv:2412.17259*, 2024.

Li, Y., Wang, S., Ding, H., and Chen, H. Large language models in finance: A survey. In *Proceedings of the fourth ACM international conference on AI in finance*, pp. 374–382, 2023.

Li, Z., Ren, K., Jiang, X., Li, B., Zhang, H., and Li, D. Domain generalization using pretrained models without fine-tuning. *arXiv preprint arXiv:2203.04600*, 2022.

Lin, T., Zhang, W., Li, S., Yuan, Y., Yu, B., Li, H., He, W., Jiang, H., Li, M., Song, X., et al. Healthgpt: A medical large vision-language model for unifying comprehension and generation via heterogeneous knowledge adaptation. *arXiv preprint arXiv:2502.09838*, 2025.

Liu, X.-Y., Wang, G., Yang, H., and Zha, D. Fingpt: Democratizing internet-scale data for financial large language models. *arXiv preprint arXiv:2307.10485*, 2023.

Lobel, S., Gottesman, O., Allen, C., Bagaria, A., and Konidaris, G. Optimistic initialization for exploration in continuous control. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7612–7619, 2022.

Lucas, J. M. and Saccucci, M. S. Exponentially weighted moving average control schemes: properties and enhancements. *Technometrics*, 32(1):1–12, 1990.

Mao, Y., Ge, Y., Fan, Y., Xu, W., Mi, Y., Hu, Z., and Gao, Y. A survey on lora of large language models. *Frontiers of Computer Science*, 19(7):197605, 2025.

Que, H., Liu, J., Zhang, G., Zhang, C., Qu, X., Ma, Y., Duan, F., Bai, Z., Wang, J., Zhang, Y., et al. D-cpt law: Domain-specific continual pre-training scaling law for large language models. *Advances in Neural Information Processing Systems*, 37:90318–90354, 2024.

Ren, S., Jian, P., Ren, Z., Leng, C., Xie, C., and Zhang, J. Towards scientific intelligence: A survey of llm-based scientific agents. *arXiv preprint arXiv:2503.24047*, 2025.

Siriwardhana, S., Weerasekera, R., Wen, E., Kaluarachchi, T., Rana, R., and Nanayakkara, S. Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, 11:1–17, 2023.

Song, Z., Yan, B., Liu, Y., Fang, M., Li, M., Yan, R., and Chen, X. Injecting domain-specific knowledge into large language models: a comprehensive survey. *arXiv preprint arXiv:2502.10708*, 2025.

Susnjak, T., Hwang, P., Reyes, N., Barczak, A. L., McIntosh, T., and Ranathunga, S. Automating research synthesis with domain-specific large language model fine-tuning. *ACM Transactions on Knowledge Discovery from Data*, 19(3):1–39, 2025.

Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Wu, Y., Zhou, S., Liu, Y., Lu, W., Liu, X., Zhang, Y., Sun, C., Wu, F., and Kuang, K. Precedent-enhanced legal judgment prediction with llm and domain-model collaboration. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12060–12075, 2023.

Yan, B., Li, K., Xu, M., Dong, Y., Zhang, Y., Ren, Z., and Cheng, X. On protecting the data privacy of large language models (llms): A survey. *arXiv preprint arXiv:2403.05156*, 2024.

Zhang, B., Liu, Z., Cherry, C., and Firat, O. When scaling meets llm finetuning: The effect of data, model and finetuning method. *arXiv preprint arXiv:2402.17193*, 2024.

Zheng, J., Hong, H., Liu, F., Wang, X., Su, J., Liang, Y., and Wu, S. Fine-tuning large language models for domain-specific machine translation. *arXiv preprint arXiv:2402.15061*, 2024.