FEDPTR: FEDERATED SOFT-PROMPT ROUTING UNDER CORRUPTED EVALUATION FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) are deployed with libraries of reusable prompts, creating an online routing problem in which each query must be assigned to an appropriate prompt. In collaborative deployments, federated sharing improves routing by pooling cross-client evidence, but also creates a new vulnerability. Manipulated feedback received by honest clients can propagate through protocol-compliant updates and affect clients with clean evaluation channels. We call this *cross-client evaluation contamination*. This creates a tension between preserving useful sharing and limiting contamination, while standard uncertainty-based influence control cannot distinguish confidence derived locally from that gained through federation. To address this tension, we introduce FEDPTR, a corruption-robust framework for federated online soft-prompt routing that controls feedback influence before evaluation scores are revealed. FedPTR contrasts local and federated uncertainty to quantify dependence on cross-client evidence and augments standard uncertainty clipping with federation-aware attenuation. We distinguish manipulated evaluator scores from trusted response utility and establish affected-to-unaffected prediction influence bounds and trusted-utility regret guarantees for both the full federation and unaffected clients. Extensive experiments show that FedPTR limits cross-client contamination while preserving the benefits of federated routing.

1 INTRODUCTION

Large language models (LLMs) (Achiam et al., 2023; Bai et al., 2023; Liu et al., 2024) are sensitive to how they are prompted, while soft-prompt tuning provides a parameter-efficient way to adapt a frozen LLM (Lester et al., 2021; Vu et al., 2022; Han et al., 2024). As deployments accumulate multiple prompt assets, they face a natural decision problem: which prompt should be used for each query? Exhaustively evaluating every candidate is costly, motivating query-level prompt routing that learns from feedback on previously served responses (Ong et al., 2025; Chen et al., 2024; Wang et al., 2025). In collaborative deployments, federated sharing can further transfer routing evidence across clients with related query streams or prompt libraries (Lu et al., 2025; Gong et al., 2026). We study this setting with a fixed LLM and soft-prompt library, where clients continually learn which prompt to invoke from online evaluation feedback.

Automatic evaluators can exhibit systematic biases (Zheng et al., 2023; Wang et al., 2024; Koo et al., 2024), fail to reflect response-quality degradation (Doddapaneni et al., 2024), and be deliberately manipulated through adversarial inputs or evaluator backdoors (Raina et al., 2024; Shi et al., 2024; Tong et al., 2025). In federated routing, however, an adversary need not compromise a client or directly manipulate its updates. Corrupted feedback received by otherwise honest clients can enter the shared router through protocol-compliant updates and affect clients whose own evaluation channels remain clean. We call this phenomenon ***cross-client evaluation contamination***: manipulation confined to a subset of evaluation channels can propagate beyond its source through federated evidence sharing.

The online nature of routing (Jia et al., 2025; Raj et al., 2026) makes such contamination particularly difficult to contain. A manipulated evaluation changes the shared routing state, which can alter future prompt selections, responses, and subsequent feedback: $\underbrace{\text{manipulated evaluation}}_{\text{local}} \Longrightarrow \underbrace{\text{shared-state change}}_{\text{federated}} \Longrightarrow \underbrace{\text{future routing change}}_{\text{sequential}} \Longrightarrow \underbrace{\text{different future observations}}_{\text{feedback loop}}$. This sequential

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

dependence limits post-hoc defenses, since later correction cannot recover the counterfactual observations lost once corrupted feedback has altered the routing trajectory. Unlike conventional federated poisoning, which targets local data or transmitted updates (Xia et al., 2023; Wan et al., 2024), the malicious signal here enters through feedback received by honest clients. Although this feedback can be viewed as corrupted rewards (He et al., 2022; Bukharin et al., 2024), federation repeatedly reuses it through a shared online decision state, allowing its effects to persist in future interactions.

This creates a fundamental tension: cross-client sharing improves routing under reliable feedback, yet the same shared state provides a pathway for corrupted feedback to propagate. Isolating clients would remove this pathway but also forfeit the benefits of collaboration. Moreover, shared uncertainty reveals how well a query–prompt direction is supported overall, but not how much of that support comes from local versus cross-client evidence. We therefore contrast local and federated uncertainty and use their gap to measure the uncertainty reduction induced by federation. A larger gap indicates that confidence in the selected direction depends more strongly on cross-client evidence, providing a natural signal for regulating how new feedback enters the shared state. Importantly, $\Delta_{t,i}^{\mathrm{fed}}$ is not a corruption detector; it measures how strongly confidence in the selected direction depends on cross-client evidence. When $\Delta_{t,i}^{\mathrm{fed}}$ is small, FedPTR reduces toward standard uncertainty clipping, whereas larger values induce stronger attenuation to control federation-specific influence.

Motivated by this observation, we develop FEDPTR, a corruption-robust framework for federated online soft-prompt routing over a frozen query–prompt representation. For each selected query–prompt pair, FedPTR computes its federated and local uncertainties, and commits to the pre-feedback weight before observing the evaluation score. The numerator bounds the normalized influence of an individual observation, while the denominator makes this control federation-aware by accounting for the uncertainty reduction induced by cross-client evidence. FedPTR therefore regulates both individual feedback influence and how strongly new feedback re-enters directions whose confidence has already been shaped by federation. It requires neither attack detection nor knowledge of the affected clients or realized corruption budget.

We distinguish potentially manipulated evaluator scores from trusted response utility and characterize how corruption propagates through the shared estimator. If $\mathcal{A}$ denotes the affected clients, then for any query–prompt direction x, including those at unaffected clients, we bound the prediction influence as $\left|x^\top e_t^{\mathcal{A}}\right| \le \alpha C_{\mathcal{A},<t}\|x\|_{A_t^{-1}}$, where $C_{\mathcal{A},<t}$ measures cumulative corruption. We further establish a uniform high-probability confidence guarantee and trusted-utility regret bounds for both the full federation and unaffected clients, separating the ordinary cost of online learning from the corruption-dependent cost of cross-client contamination. Empirically, we evaluate FedPTR under both controlled score corruption and LLM-evaluator manipulation. Matched local/federated and weighted/unweighted comparisons separate the benefits of collaboration from the cost of contamination.

Our contributions are threefold: (1) We identify and empirically characterize *cross-client evaluation contamination*, whereby manipulated evaluator feedback enters through honest, protocol-compliant clients and propagates across clients and future online interactions. (2) We introduce federation-aware pre-feedback influence control and instantiate it in FEDPTR, which contrasts local and federated uncertainty to regulate how strongly new feedback enters the shared state, without attack detection or prior knowledge of the corruption budget. (3) We provide trusted-utility guarantees that connect local evaluation corruption to its federated consequences, including an explicit affected-to-unaffected prediction influence bound and regret guarantees for both the full federation and unaffected clients, complemented by experiments under controlled corruption and LLM-based evaluator attacks.

2 RELATED WORK

Prompt learning and routing. Parameter-efficient prompt learning adapts pretrained models by optimizing continuous prompts while keeping most parameters frozen (Li & Liang, 2021; Liu et al., 2022; Vu et al., 2022). Beyond learning a single prompt, recent work explores multiple prompt components and input-dependent selection for heterogeneous inputs or tasks (Wang et al., 2022; Wu et al., 2026), including mixture-of-prompts methods that route inputs among learned prompts (Du et al., 2025). Prompt learning and optimization have also been studied in distributed and federated settings (Lu et al., 2025; Gong et al., 2026). In contrast, we fix both the LLM and soft-prompt library and learn online which existing prompt to invoke for each query, focusing on the reliability of the evaluation feedback that drives routing rather than on learning the prompts themselves.

Evaluation manipulation and federated poisoning. LLM-based evaluators can exhibit position, preference, and other systematic biases (Wang et al., 2024; Stureborg et al., 2024; Panickssery et al., 2024), and can be actively manipulated through adversarial or prompt-injection attacks (Shi et al., 2024; Zheng et al., 2025; Akter et al., 2026) or evaluator compromise (Tong et al., 2025). While these works primarily study attacks on the evaluator or its robustness, we study what happens after unreliable evaluations become online learning signals shared by a federated router. Our threat also differs from conventional federated poisoning, where adversaries manipulate local data, training, gradients, or model updates (Bagdasaryan et al., 2020; Li et al., 2021). Byzantine-robust federated learning accordingly filters, clips, validates, or robustly aggregates suspicious client updates (Blanchard et al., 2017; Yin et al., 2018; Li et al., 2021). Here, clients can remain honest and their transmitted statistics protocol-compliant; corruption instead enters through an external evaluator and can propagate through federation to clients whose own evaluation channels remain unaffected.

Online learning under corrupted feedback. Robust stochastic and contextual bandits study adversarial reward corruption and provide robust estimators and regret guarantees (Lykouris et al., 2018; Gupta et al., 2019; Bogunovic et al., 2021; Wei et al., 2022). Confidence-weighted methods limit the influence of corrupted observations according to uncertainty within a single learner (He et al., 2022; Bukharin et al., 2024). Offline robust learning instead seeks to recover reliable predictors or policies from fixed logged data containing corrupted observations (Zhang et al., 2022; Mandal et al., 2024). In our online setting, corrupted feedback can change the shared routing state and thereby alter which observations are collected in subsequent rounds, making post-hoc correction unable to recover counterfactual observations that were never collected. Federated neural bandits exploit observations across clients to accelerate online exploration (Dai et al., 2022), but typically assume stochastic rather than adversarially corrupted feedback. FedPTR connects these directions by augmenting uncertainty-based influence clipping with the local–federated uncertainty gap, enabling pre-feedback control and analysis of corruption that propagates through cross-client evidence sharing.

3 Threat Model and Evaluation-Corrupted Prompt Routing

Federated Prompt-Routing System. Consider N clients that collaboratively route queries over a shared library of pretrained soft prompts $\mathcal{P} = \{P_1, \dots, P_K\}$ using a frozen language model F. Each prompt $P_k \in \mathbb{R}^{m_p \times d_e}$ remains fixed during deployment, and client i has access to a nonempty subset $\mathcal{P}_i \subseteq \mathcal{P}$. At round t, client i receives a query $q_{t,i}$, selects a prompt $P_{t,i} \in \mathcal{P}_i$, and serves the response $y_{t,i} = F(q_{t,i}; P_{t,i})$. Neither F nor the prompt library is updated; online learning determines only which existing prompt to invoke for each query.

We also consider T synchronous rounds, yielding $Q = NT$ interactions. All clients make their round-t decisions from the same immutable server snapshot. After the selected responses are evaluated, clients contribute routing statistics that are aggregated into a shared state. Consequently, feedback collected in round t can first affect routing decisions in round $t + 1$.

Evaluation-Channel Threat Model. For a query–prompt pair $(q_{t,i}, P)$, let $\mu(q_{t,i}, P) \in [0, 1]$ denote its expected trusted utility, i.e., the response quality that the deployment ultimately seeks to maximize, such as ground-truth correctness, human preference, or an independent trusted evaluation. This utility is unavailable to the online router. After selecting $P_{t,i}$, the router instead observes

$$\widetilde{r}_{t,i} = \underbrace{\mu(q_{t,i}, P_{t,i})}_{\text{trusted utility}} + \underbrace{\epsilon_{t,i}}_{\text{evaluation noise}} + \underbrace{c_{t,i}(P_{t,i})}_{\text{systematic distortion}}, \tag{1}$$

where $\widetilde{r}_{t,i} \in [0, 1]$, $\epsilon_{t,i}$ denotes centered stochastic evaluation noise, and $c_{t,i}(P)$ captures systematic, potentially persistent distortion between the observed evaluation and trusted utility. This formulation captures the downstream effect of unreliable or manipulated evaluators (Zheng et al., 2023; Wang et al., 2024; Koo et al., 2024; Shi et al., 2024; Tong et al., 2025).

The adversary may affect a subset of clients, rounds, or prompts. FedPTR knows neither which evaluations are corrupted nor the realized corruption budget. Clients otherwise follow the routing protocol, and the server is trusted. Byzantine clients, malicious servers, prompt-library poisoning, modification of the frozen LLM, and attacks that directly alter served responses are outside our threat model. We also make no formal privacy guarantee for the shared routing statistics. Figure 1 illustrates the resulting propagation pathway. Manipulated evaluation feedback can enter the shared routing state through an otherwise honest client and subsequently affect clients whose own evaluation

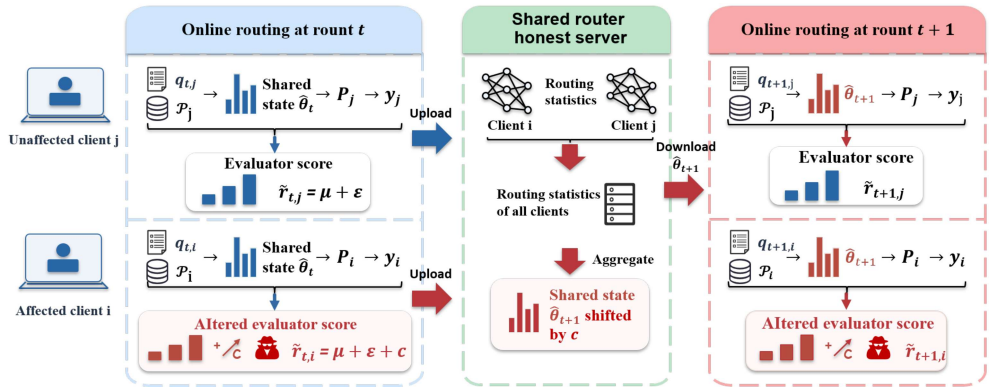


Figure 1: **Evaluation-channel manipulation in federated prompt routing.** An affected client receives a potentially manipulated evaluation $\widetilde{r}_{t,i}$ and contributes the resulting statistics to the shared router. Although another client j has an unaffected evaluation channel, its subsequent prompt choices can change through the contaminated shared state.

channels remain clean. Let $\mathcal{A} \subseteq [N]$ denote the clients whose evaluation channels may be corrupted, and let $\mathcal{U} = [N] \setminus \mathcal{A}$ denote the unaffected clients. For every $j \in \mathcal{U}$, we have $c_{t,j}(P) = 0$ for all t and $P \in \mathcal{P}_j$. Nevertheless, corruption originating from a client $i \in \mathcal{A}$ can modify the shared state and thereby alter future routing decisions at $j \in \mathcal{U}$:

$$\underbrace{c_{t,i} \neq 0}_{\text{local manipulation}} \implies \underbrace{\Delta \widehat{w}_{t+1}}_{\text{shared-state change}} \implies \underbrace{\Delta P_{\tau,j}}_{\text{routing change at } j \in \mathcal{U},\ \tau > t}$$

We refer to this downstream propagation as *cross-client evaluation contamination*. FedPTR does not attempt to identify or repair the upstream evaluator; instead, it controls how strongly potentially corrupted feedback can influence the shared routing state.

Trusted-Utility Objective. Because $\widetilde{r}_{t,i}$ may be corrupted, we evaluate routing using trusted response utility rather than the observed evaluator score. For the analysis, the query sequence, available prompt sets, and candidate-wise corruption sequence are exogenous to the router's private decision randomness. We define the total evaluation corruption as

$$C_{\text{eval}} \triangleq \sum_{t=1}^{T} \sum_{i=1}^{N} \max_{P \in \mathcal{P}_i} |c_{t,i}(P)|. \tag{2}$$

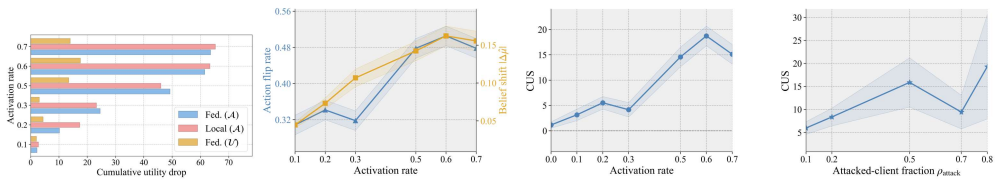
The action-uniform maximum makes C_{eval} independent of the prompts selected by the router. This quantity is used only for analysis and is unknown. Let $P^*_{t,i} \in \arg\max_{P \in \mathcal{P}_i} \mu(q_{t,i}, P)$ denote the trusted-optimal available prompt. Our primary objective is the cumulative trusted-utility regret

$$R_Q^{\text{true}} = \mathbb{E}\left[\sum_{t=1}^{T} \sum_{i=1}^{N} \left(\mu(q_{t,i}, P^*_{t,i}) - \mu(q_{t,i}, P_{t,i}) \right) \right]. \tag{3}$$

Clients may have heterogeneous query distributions and prompt sets $\mathcal{P}_i$, while sharing the same underlying utility function $\mu(q, P)$. Our objective is therefore to retain the benefits of federated evidence sharing while limiting the downstream influence of corrupted evaluation feedback. Section 5 develops FEDPTR toward this objective.

4 MOTIVATING OBSERVATION

The threat model in Section 3 raises a central question: Can locally manipulated evaluation feedback affect unaffected clients through federation? To examine this question, we partition clients into an affected set $\mathcal{A}$ and an unaffected set $\mathcal{U}$, and compare federated routing with matched local routers under identical query streams, prompt pools, and corruption schedules. Only clients in $\mathcal{A}$ receive manipulated feedback. Each local router observes the same queries and evaluations as its federated counterpart but does not share evidence, thereby isolating the effect introduced by federation.



(a) (b) (c) (d)

Figure 2: **Evaluation manipulation does not remain local under federated routing.** (a) Trusted-utility degradation on affected and unaffected clients. (b) Shared-estimate and routing changes on unaffected clients as manipulation becomes active. (c) CUS under increasing corruption. (d) CUS as the fraction of affected clients increases at fixed corruption budget $C_{\text{eval}} = 0.3Q^{1/2}$.

Figure 2a first reveals a clear asymmetry. On affected clients, both local and federated routers lose trusted utility because both receive corrupted evaluations. On unaffected clients, however, the matched local routers remain largely stable, whereas the federated router degrades despite these clients never receiving manipulated feedback. Since the two conditions share the same local queries and evaluations, this additional degradation emerges only when evidence is shared across clients, providing direct evidence of *cross-client evaluation contamination*. Figure 2b then traces how this nonlocal effect emerges. As manipulation becomes active, the shared estimate associated with the targeted prompt shifts, followed by changes in prompt selection at unaffected clients: corrupted local feedback $\rightarrow$ shared-state change $\rightarrow$ routing change at unaffected clients. Once corrupted feedback enters the shared routing state, it can therefore affect not only future decisions but also which observations are subsequently collected. This sequential effect motivates controlling feedback influence online, before it alters the routing trajectory, rather than relying solely on post-hoc correction.

This nonlocal effect becomes more pronounced as the corruption carried by the shared state increases. To quantify it, we define the cross-client utility spillover (CUS) $\mathrm{CUS} := U^{\mathrm{local}}_{\mathrm{unaff}} - U^{\mathrm{fed}}_{\mathrm{unaff}}$, which measures the additional utility loss on unaffected clients. Figure 2c shows that CUS is near zero without manipulation and becomes positive under corruption, with larger spillover as corruption increases. Figure 2d further shows that CUS increases overall as a larger fraction of clients becomes affected, despite some non-monotonicity at intermediate points. Federation can therefore expand the footprint of evaluation manipulation beyond the clients at which it originates.

These observations expose the central tension of federated prompt routing. Cross-client evidence improves routing under reliable feedback, yet the same shared state provides a pathway for corrupted evaluations to propagate. Eliminating federation would prevent this propagation but also discard its collaborative benefit. The goal is therefore to preserve evidence sharing while controlling the influence of new feedback *before* it modifies the shared routing state. This suggests the key design principle behind FedPTR: distinguish confidence supported by a client's own history from confidence gained through cross-client evidence, and regulate new feedback accordingly. Section 5 realizes this principle through local–federated uncertainty and pre-feedback influence control.

5 FedPTR: Pre-Feedback Influence Control

Section 4 shows that federated evidence sharing can propagate locally corrupted evaluations to unaffected clients. FedPTR addresses this tension by retaining federated routing while controlling the influence of each evaluation before its feedback is revealed. Figure 3 illustrates the framework, which combines federated query–prompt routing, local–federated pre-feedback influence control, and exact aggregation of weighted sufficient statistics.

5.1 Federated Prompt Routing

For each client i and query $q_{t,i}$, FedPTR treats the available soft prompts as contextual actions and represents each query–prompt pair using a frozen neural feature map $\phi(q, P) \in \mathbb{R}^{d_\phi}$. The server maintains a shared weighted ridge estimator $\widehat{w}_t = A_t^{-1} b_t$ from observations contributed by all clients, where A_t summarizes the accumulated query–prompt geometry and b_t the corresponding evaluator

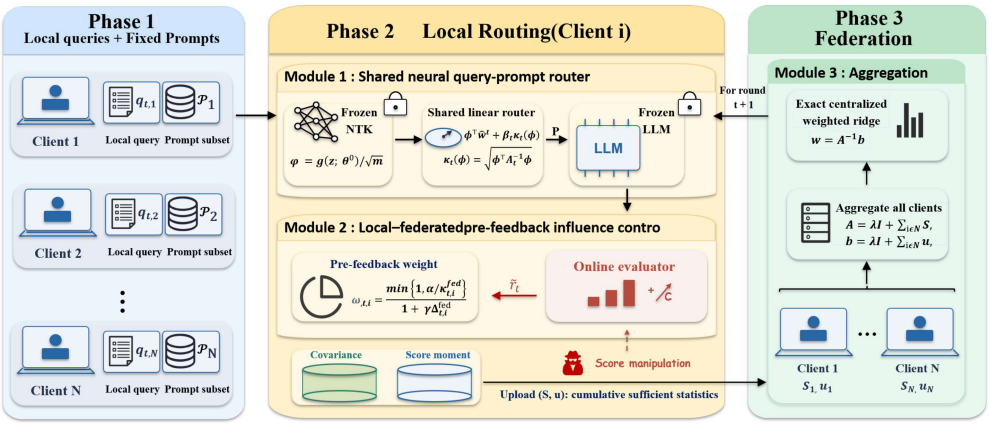


Figure 3: FedPTR has three components: local queries and frozen prompt assets, query-dependent routing with evaluation feedback, and exact statistical federation. Clients select one prompt and weight its update before uploading versioned statistics; the server broadcasts the next-round state. Score manipulation enters through the evaluator.

feedback. For each candidate $P_k \in \mathcal{P}_i$, let $\phi_{t,i,k} = \phi(q_{t,i}, P_k)$ and define its federated uncertainty as $\kappa^{\text{fed}}_{t,i,k} = \|\phi_{t,i,k}\|_{A_t^{-1}}$. FedPTR routes the query according to

$$P_{t,i} \in \arg\max_{P_k \in \mathcal{P}_i} \left[\phi_{t,i,k}^\top \widehat{w}_t + \beta_t \kappa^{\text{fed}}_{t,i,k}\right]. \tag{4}$$

Eq. (4) enables two complementary forms of generalization. The frozen representation transfers routing experience across related query–prompt pairs, while the shared estimator transfers evidence across clients. Consequently, an observation collected at one client can reduce uncertainty for related routing decisions at another, allowing clients to benefit from one another's routing experience.

This cross-client sharing, however, is also the propagation channel in Section 4: corrupted evaluation feedback contributed by one client can modify the shared estimator and subsequently affect routing at clients whose own evaluation channels remain clean. FedPTR therefore retains federated exploration while controlling how strongly each new evaluation can influence the shared state. After routing, we abbreviate the selected feature and its federated uncertainty by $\phi_{t,i}$ and $\kappa^{\text{fed}}_{t,i}$, respectively.

5.2 Local–Federated Pre-Feedback Influence Control

Shared uncertainty measures how well a selected direction is supported overall, but not how much of that support comes from local versus cross-client evidence. To expose this distinction, FedPTR evaluates the selected feature under both the shared and local design matrices:

$$\kappa^{\text{fed}}_{t,i} = \|\phi_{t,i}\|_{A_t^{-1}}, \qquad \kappa^{\text{loc}}_{t,i} = \|\phi_{t,i}\|_{(A^{\text{loc}}_{t,i})^{-1}}.$$

Because the shared matrix contains the local history together with evidence from other clients, $\kappa^{\text{fed}}_{t,i} \le \kappa^{\text{loc}}_{t,i}$. We therefore define the *federation-induced uncertainty reduction*

$$\Delta^{\text{fed}}_{t,i} := \kappa^{\text{loc}}_{t,i} - \kappa^{\text{fed}}_{t,i} \ge 0. \tag{5}$$

A larger $\Delta^{\text{fed}}_{t,i}$ indicates that confidence in the selected direction depends more strongly on cross-client evidence; it does not serve as a corruption detector. It characterizes the contribution of federation to the current confidence geometry. FedPTR assigns the selected observation the pre-feedback weight

$$\omega_{t,i} = \frac{\min\left\{1,\ \alpha/\kappa^{\text{fed}}_{t,i}\right\}}{1 + \gamma \Delta^{\text{fed}}_{t,i}}. \tag{6}$$

The numerator bounds the normalized leverage of an individual evaluation, while the denominator makes this control federation-aware by accounting for cross-client uncertainty reduction. Crucially, $\omega_{t,i}$ is fixed before observing $\widetilde{r}_{t,i}$ and therefore does not depend on whether the realized feedback

appears suspicious. This pre-feedback control is important because once an evaluation modifies the shared state, it may already affect subsequent routing decisions and observations. The two factors are complementary: setting $\gamma = 0$ recovers standard uncertainty-based clipping, $\omega_{t,i} = \min\{1, \alpha/\kappa_{t,i}^{\mathrm{fed}}\}$, while $\gamma > 0$ additionally accounts for uncertainty reduction induced by federation. In either case, $\omega_{t,i}\kappa_{t,i}^{\mathrm{fed}} \le \alpha$, yielding the influence-control property used in our analysis. Thus, the local–federated term augments rather than replaces standard uncertainty clipping.

Only after $\omega_{t,i}$ is fixed does client i obtain $\widetilde{r}_{t,i}$ and incorporate the weighted observation into its cumulative sufficient statistics. The server aggregates the latest client statistics to reconstruct the shared weighted ridge estimator for the next round; Algorithms 1-2 give the complete procedures. This ordering preserves cross-client evidence sharing while regulating each new observation before it enters the shared state. Evidence from one client can therefore improve routing at others without allowing newly observed feedback unrestricted influence over the shared estimator.

5.3 Theoretical Guarantees

Our analysis formalizes how pre-feedback influence control limits both estimation error and cross-client contamination. Let $\mathcal{A}$ denote clients whose evaluation channels may be corrupted and $\mathcal{U}$ clients whose own evaluation channels remain clean. These sets are used only for analysis and are unknown to FedPTR. Define the federation-aware realized corruption

$$C_{\mathcal{A},<t}^{\mathrm{obs},(\gamma)} := \sum_{s<t}\sum_{i\in\mathcal{A}} \frac{|c_{s,i}|}{1+\gamma\Delta_{s,i}^{\mathrm{fed}}}.$$

For any candidate feature x at an unaffected client, Appendix B shows that the prediction shift attributable to corrupted evaluations originating from $\mathcal{A}$ satisfies

$$\left|x^\top e_t^{\mathcal{A}}\right| \le \alpha C_{\mathcal{A},<t}^{\mathrm{obs},(\gamma)} \|x\|_{A_t^{-1}} \le \alpha C_{\mathcal{A},<t}^{\mathrm{obs}} \|x\|_{A_t^{-1}}.$$

Thus, standard uncertainty clipping ($\gamma = 0$) recovers the coarser realized corruption budget, whereas the federation-aware factor discounts each corrupted observation in proportion to the uncertainty reduction induced by cross-client evidence along the realized trajectory. We further prove a high-probability trusted-utility regret guarantee. With $\alpha = Q^{-1/2}$ and $C_{\mathrm{eval}} \le \sqrt{Q}$, Theorem 1 gives

$$R_Q^{\mathrm{true}} \le \min\left\{Q,\, 2\beta V_Q^{(\gamma)} + Q\delta\right\},$$

where $V_Q^{(\gamma)}$ captures the cumulative uncertainty under the local–federated weighting rule. For fixed representation dimension and regularization parameters, the resulting bound remains sublinear in Q up to logarithmic and synchronous-batch terms. Finally, we obtain a separate guarantee for the unaffected clients. The corruption-dependent part of their regret scales as $\alpha C_{\mathcal{A}}^{\mathrm{obs},(\gamma)} V_{\mathcal{U}}^{(\gamma)}$, making explicit the cost of cross-client contamination after federation-aware attenuation rather than absorbing it into a federation-wide corruption term. The bound also clarifies the scope of this attenuation: corruption at small Δ^{fed} receives little additional discount beyond standard uncertainty clipping, whereas corruption along directions more strongly shaped by cross-client evidence is attenuated more. These results show that FedPTR does not eliminate the benefit of federated evidence sharing to obtain robustness; instead, it bounds the additional influence that corrupted evaluation feedback can acquire through that sharing process.

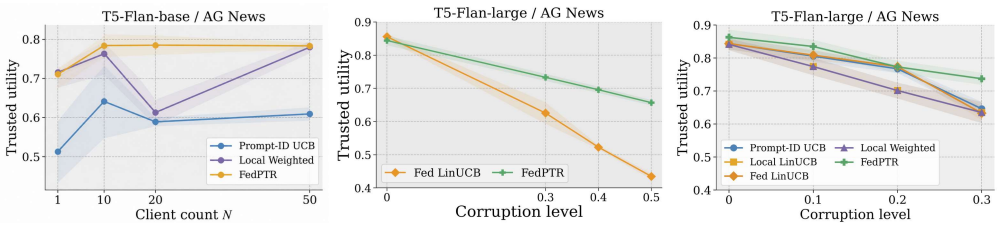
6 Experiments

Experimental Setup. We evaluate FedPTR across both language and vision tasks. The language suite includes AG News (Zhang et al., 2015), TREC (Li & Roth, 2002), XSum (Narayan et al., 2018), and E2E (Novikova et al., 2017). We use T5-Flan-base (Chung et al., 2024) across all four datasets, and additionally evaluate AG News and TREC with T5-Flan-large and Qwen2.5-7B (Hui et al., 2024). The vision suite includes EuroSAT (Helber et al., 2019) and CIFAR-10 (Krizhevsky et al., 2009), using ViT-B/16 and ViT-L/14 (Radford et al., 2021). Across these settings, the backbone model and prompt library remain frozen throughout online routing.

Unless otherwise specified, each controlled experiment uses $N = 20$ clients, $T = 500$ rounds, $K = 16$ frozen prompts, and a $d_\phi = 128$-dimensional query–prompt representation. We report

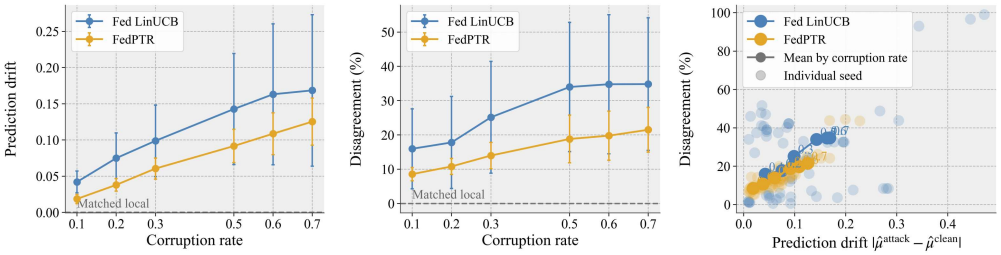
Table 1: Trusted utility at the primary corruption budget.

	T5-Flan-base				T5-Flan-large		Qwen2.5-7B		ViT-B/16		ViT-L/14	
Method	AG News	TREC	XSum	E2E	AG News	TREC	AG News	TREC	EuroSAT	CIFAR-10	EuroSAT	CIFAR-10
Prompt-ID UCB	0.589 ± 0.01	0.580 ± 0.02	0.105 ± 0.01	0.360 ± 0.01	0.440 ± 0.02	0.473 ± 0.01	0.538 ± 0.064	0.226 ± 0.02	0.439 ± 0.01	0.514 ± 0.01	0.470 ± 0.02	0.494 ± 0.01
Local LinUCB	0.630 ± 0.02	0.623 ± 0.03	0.116 ± 0.01	0.341 ± 0.03	0.503 ± 0.04	0.542 ± 0.02	0.546 ± 0.027	0.253 ± 0.03	0.591 ± 0.03	0.627 ± 0.05	0.591 ± 0.06	0.611 ± 0.07
Fed LinUCB	0.591 ± 0.01	0.582 ± 0.02	0.109 ± 0.01	0.342 ± 0.01	0.435 ± 0.01	0.470 ± 0.01	0.541 ± 0.030	0.238 ± 0.02	0.515 ± 0.05	0.668 ± 0.03	0.502 ± 0.03	0.758 ± 0.08
Local Weighted	0.613 ± 0.03	0.639 ± 0.04	0.118 ± 0.02	0.338 ± 0.01	0.503 ± 0.04	0.542 ± 0.02	0.531 ± 0.073	0.261 ± 0.04	0.671 ± 0.05	0.663 ± 0.07	0.591 ± 0.06	0.628 ± 0.09
FedPTR	$\mathbf{0.785 \pm 0.02}$	$\mathbf{0.778 \pm 0.01}$	$\mathbf{0.122 \pm 0.01}$	$\mathbf{0.427 \pm 0.02}$	$\mathbf{0.657 \pm 0.01}$	$\mathbf{0.730 \pm 0.01}$	$\mathbf{0.566 \pm 0.028}$	$\mathbf{0.282 \pm 0.02}$	$\mathbf{0.709 \pm 0.02}$	$\mathbf{0.895 \pm 0.05}$	$\mathbf{0.714 \pm 0.02}$	$\mathbf{0.889 \pm 0.02}$



(a) Client-count scaling (b) Controlled corruption (c) Real evaluator attack

Figure 4: **Robustness stress tests.** (a) Client-count scaling. (b) Controlled corruption. (c) Real LLM-evaluator attack.



(a) Prediction drift (b) Routing disagreement (c) Drift vs. disagreement

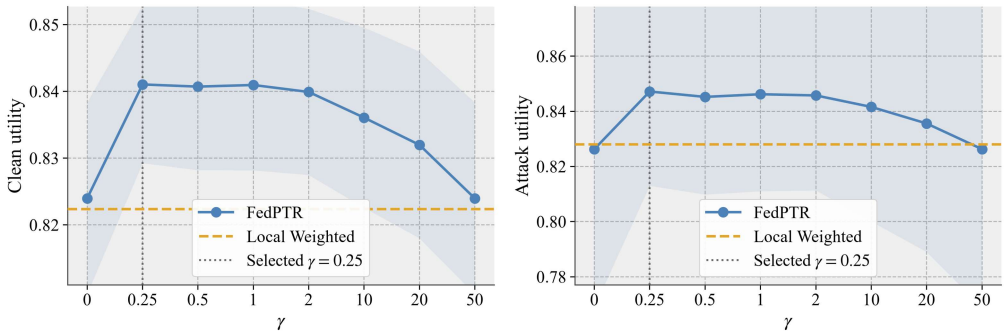
Figure 5: Shared-state and routing changes on unaffected clients under corruption.

results over paired random seeds $\{71, 89, 107\}$, while all configuration choices are made using a separate seed 313 on a disjoint development split. For T5-Flan-base, we consider corruption budgets $C_{\text{eval}} \in \{0.1, 0.2, 0.3\}Q^{1/2}$; for the other controlled backbones, we use $C_{\text{eval}} \in \{0.3, 0.4, 0.5\}Q^{1/2}$. Beyond the default $N = 20$ setting, we evaluate client-count scaling over $N \in \{1, 10, 20, 50\}$. We also consider a real LLM-evaluator manipulation setting on T5-Flan-large with AG News, sweeping attack levels $\{0, 0.1, 0.2, 0.3\}$ under the same paired report seeds.

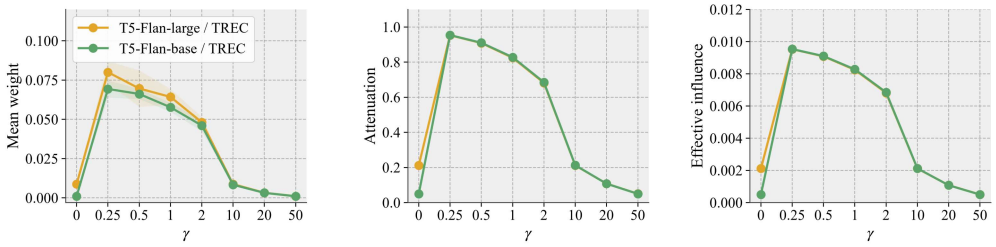
We compare FedPTR against Prompt-ID UCB (Auer et al., 2002), Local LinUCB, Fed LinUCB (Li et al., 2010), and Local Weighted (He et al., 2022). In particular, the Local/Federated and Unweighted/Weighted variants form the main 2×2 comparison, separating the effects of cross-client sharing from those of robust weighting. We report trusted response utility as the primary evaluation endpoint; potentially corrupted evaluator scores are used only as online learning feedback and are never treated as ground-truth performance. Full implementation details are provided in Appendix C.

Robustness under controlled corruption. Table 1 reports all-client trusted utility at the primary corruption budget. FedPTR outperforms Local Weighted across all settings and matches or exceeds both unweighted baselines. Comparing with Fed LinUCB isolates the benefit of influence control under federation, while Local Weighted shows that local weighting alone does not explain the gain. Fed LinUCB is not consistently better than Local LinUCB, showing that uncontrolled sharing can lose its advantage when contaminated feedback propagates through the shared state.

Robustness stress tests. Figure 4 evaluates larger federations, stronger controlled corruption, and real LLM-evaluator manipulation. FedPTR remains robust as N increases, degrades more slowly than others as corruption grows, and retains its advantage as real-attack levels increase. These results show that its robustness extends beyond a particular federation size or synthetic corruption regime.



(a) Utility trade-off.



(b) TREC update-weight and influence diagnostics.

Figure 6: **Federation-aware attenuation.** Utility trade-off and TREC mechanism across γ.

Propagation mechanism. Figure 5 examines unaffected clients. As corruption increases, federated prediction drift and routing disagreement both increase, while the matched local reference remains at zero; larger drift is also associated with more routing changes. This is consistent with the propagation pathway in Section 4, where corrupted feedback perturbs the shared estimator and subsequently affects clean clients.

Role of federation-aware attenuation. We isolate the denominator of FedPTR's pre-feedback weight $\omega_{t,i}$ in (6). Setting $\gamma = 0$ removes local–federated attenuation while retaining standard uncertainty clipping. Figure 6 shows a robustness–sharing trade-off: moderate γ improves trusted utility over $\gamma = 0$, whereas larger γ increasingly suppresses both effective influence and useful cross-client evidence. Thus, the local–federated uncertainty gap regulates, rather than eliminates, cross-client sharing.

Overall, FedPTR combines federated evidence sharing with pre-feedback influence control to limit contamination while retaining useful cross-client evidence. Additional sweeps, post-hoc diagnostics, and recovery tests are provided in Appendix C.

7 Conclusion

We study federated online soft-prompt routing under unreliable evaluation feedback and identify *cross-client evaluation contamination*, where corrupted feedback received by honest clients propagates through the shared routing state to clients with unaffected evaluation channels. We develop FedPTR, which combines uncertainty-based influence clipping with local–federated uncertainty to regulate feedback before its evaluation score is revealed. This enables cross-client evidence sharing while controlling the influence of potentially corrupted feedback on the shared estimator. Our analysis establishes affected-to-unaffected prediction influence and trusted-utility regret guarantees, while experiments under controlled corruption and LLM-evaluator manipulation demonstrate the robustness–collaboration tradeoff. More broadly, our results show that protocol-compliant updates need not provide trustworthy evidence when the feedback channel itself is unreliable. Extending this framework to misspecified utility models, adaptive corruption, privacy-aware aggregation, and evolving prompt libraries offers several directions for future work.

AI USE STATEMENT

AI tools assisted manuscript organization, mathematical checking, and editing. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

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.

Sanjeda Akter, Ibne Farabi Shihab, and Anuj Sharma. Detecting proxy gaming in rl and llm alignment via evaluator stress tests. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 10554–10583, 2026.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47:235–256, 2002.

Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pp. 2938–2948. PMLR, 2020.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017.

Ilija Bogunovic, Arpan Losalka, Andreas Krause, and Jonathan Scarlett. Stochastic linear bandits robust to adversarial attacks. In *International Conference on Artificial Intelligence and Statistics*, pp. 991–999. PMLR, 2021.

Alexander Bukharin, Ilgee Hong, Haoming Jiang, Zichong Li, Qingru Zhang, Zixuan Zhang, and Tuo Zhao. Robust reinforcement learning from corrupted human feedback. *Advances in Neural Information Processing Systems*, 37:124093–124113, 2024.

Shuhao Chen, Weisen Jiang, Baijiong Lin, James Kwok, and Yu Zhang. Routerdc: Query-based router by dual contrastive learning for assembling large language models. *Advances in Neural Information Processing Systems*, 37:66305–66328, 2024.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of machine learning research*, 25(70):1–53, 2024.

Zhongxiang Dai, Yao Shu, Arun Verma, Flint Xiaofeng Fan, Bryan Kian Hsiang Low, and Patrick Jaillet. Federated neural bandits. *arXiv preprint arXiv:2205.14309*, 2022.

Sumanth Doddapaneni, Mohammed Safi Ur Rahman Khan, Sshubam Verma, and Mitesh M Khapra. Finding blind spots in evaluator llms with interpretable checklists. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 16279–16309, 2024.

Yu Du, Tong Niu, and Rong Zhao. Mixture of prompts learning for vision-language models. *Frontiers in Artificial Intelligence*, 8:1580973, 2025.

Shuai Gong, Chaoran Cui, Xiaolin Dong, Xiushan Nie, Lei Zhu, and Xiaojun Chang. Token-level prompt mixture with parameter-free routing for federated domain generalization. *IEEE Transactions on Image Processing*, 2026.

Anupam Gupta, Tomer Koren, and Kunal Talwar. Better algorithms for stochastic bandits with adversarial corruptions. In *Conference on Learning Theory*, pp. 1562–1578. PMLR, 2019.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.

Jiafan He, Dongruo Zhou, Tong Zhang, and Quanquan Gu. Nearly optimal algorithms for linear contextual bandits with adversarial corruptions. *Advances in neural information processing systems*, 35:34614–34625, 2022.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Yongju Jia, Jiarui Ma, Xiangxian Li, Baiqiao Zhang, Xianhui Cao, Juan Liu, and Yulong Bian. Llm-empowered dynamic prompt routing for vision-language models tuning under long-tailed distributions. In *EMNLP (Findings)*, pp. 14809–14822, 2025.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. Benchmarking cognitive biases in large language models as evaluators. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 517–545, 2024.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 3045–3059, 2021.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 661–670, 2010.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers)*, pp. 4582–4597, 2021.

Xin Li and Dan Roth. Learning question classifiers. In *Coling 2002: The 19th international conference on computational linguistics*, 2002.

Xingyu Li, Zhe Qu, Shangqing Zhao, Bo Tang, Zhuo Lu, and Yao Liu. Lomar: A local defense against poisoning attack on federated learning. *IEEE Transactions on Dependable and Secure Computing*, 20(1):437–450, 2021.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 61–68, 2022.

Pingchen Lu, Zhi Hong, Zhiwei Shang, Zhiyong Wang, Yikun Ban, Yao Shu, Min Zhang, Shuang Qiu, and Zhongxiang Dai. Fedpob: Sample-efficient federated prompt optimization via bandits. *arXiv preprint arXiv:2509.24701*, 2025.

Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 114–122, 2018.

Debmalya Mandal, Andi Nika, Parameswaran Kamalaruban, Adish Singla, and Goran Radanović. Corruption robust offline reinforcement learning with human feedback. *arXiv preprint arXiv:2402.06734*, 2024.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 1797–1807, 2018.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. The e2e dataset: New challenges for end-to-end generation. In *Proceedings of the 18th annual SIGdial meeting on discourse and dialogue*, pp. 201–206, 2017.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, Mohammed Kadous, and Ion Stoica. Routellm: Learning to route llms from preference data. In *International Conference on Learning Representations*, volume 2025, pp. 34433–34448, 2025.

Arjun Panickssery, Samuel R Bowman, and Shi Feng. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37:68772–68802, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Vyas Raina, Adian Liusie, and Mark Gales. Is llm-as-a-judge robust? investigating universal adversarial attacks on zero-shot llm assessment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7499–7517, 2024.

Ritik Raj, Souvik Kundu, Sarbartha Banerjee, Dheemanth Joshi, Ishita Vohra, and Tushar Krishna. Trace-router: Task-consistent and adaptive online routing for agentic ai. *arXiv preprint arXiv:2607.22465*, 2026.

Jiawen Shi, Zenghui Yuan, Yinuo Liu, Yue Huang, Pan Zhou, Lichao Sun, and Neil Zhenqiang Gong. Optimization-based prompt injection attack to llm-as-a-judge. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pp. 660–674, 2024.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. Large language models are inconsistent and biased evaluators. *arXiv preprint arXiv:2405.01724*, 2024.

Terry Tong, Fei Wang, Zhe Zhao, and Muhao Chen. Badjudge: Backdoor vulnerabilities of llm-as-a-judge. In *International Conference on Learning Representations*, volume 2025, pp. 18300–18325, 2025.

Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Cer. Spot: Better frozen model adaptation through soft prompt transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5039–5059, 2022.

Yichen Wan, Youyang Qu, Wei Ni, Yong Xiang, Longxiang Gao, and Ekram Hossain. Data and model poisoning backdoor attacks on wireless federated learning, and the defense mechanisms: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 26(3):1861–1897, 2024.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, et al. Large language models are not fair evaluators. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 9440–9450, 2024.

Xinyuan Wang, Yanchi Liu, Wei Cheng, Xujiang Zhao, Zhengzhang Chen, Wenchao Yu, Yanjie Fu, and Haifeng Chen. Mixllm: Dynamic routing in mixed large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 10912–10922, 2025.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 139–149. IEEE, 2022.

Chen-Yu Wei, Christoph Dann, and Julian Zimmert. A model selection approach for corruption robust reinforcement learning. In *International Conference on Algorithmic Learning Theory*, pp. 1043–1096. PMLR, 2022.

Wei-Chi Wu, Sheng-Lun Wei, Hen-Hsen Huang, and Hsin-Hsi Chen. No one fits all: From fixed prompting to learned routing in multilingual llms. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 37404–37423, 2026.

Geming Xia, Jian Chen, Chaodong Yu, and Jun Ma. Poisoning attacks in federated learning: A survey. *Ieee Access*, 11:10708–10722, 2023.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, pp. 5650–5659. Pmlr, 2018.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

Xuezhou Zhang, Yiding Chen, Xiaojin Zhu, and Wen Sun. Corruption-robust offline reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 5757–5773. PMLR, 2022.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Xiaosen Zheng, Tianyu Pang, Chao Du, Qian Liu, Jing Jiang, and Min Lin. Cheating automatic llm benchmarks: Null models achieve high win rates. In *International Conference on Learning Representations*, volume 2025, pp. 61994–62027, 2025.