PRIVACY AWARE QUANTUM–CLASSICAL IN-CONTEXT LEARNING FOR TABULAR DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Remote quantum serving for tabular in-context learning must simultaneously address privacy-sensitive context data, variable-sized task inputs, communication overhead, and tight quantum-resource constraints. We propose *QICL*, a privacy-aware split quantum–classical framework built around a compact structured interface between the client and a remote quantum server. On the client side, *QICL* maps row-level contextual representations into bounded class-wise summaries and combines them with class-prior information to form a compact interface with explicitly controlled record-level sensitivity. A block-wise Gaussian mechanism privatizes this interface before transmission, while the interface changes the dominant context communication from record-wise to class-wise scaling. On the server side, a candidate-aware relational predictor constructs query–candidate–rival interactions before compressing them into quantum-compatible coordinates for few-qubit prediction. We establish record-level differential privacy for the uploaded privatized interface and derive a condition linking privacy perturbation to prediction stability. Experiments on TALENT classification datasets evaluate *QICL* against multiple tabular ICL models, strong conventional tabular predictors, and classical prediction heads under the same structured interface, demonstrating a favorable privacy–utility–communication trade-off under constrained quantum resources.

1 INTRODUCTION

Quantum machine learning (QML) provides a potential route for using parameterized quantum circuits (PQCs) as compact nonlinear predictors under limited qubit budgets (Zaman et al., 2023; Benedetti et al., 2019). In practice, access to quantum hardware is typically provided through remote cloud services (Nguyen et al., 2024). This deployment pattern creates a natural opportunity for hybrid quantum-classical learning (Long et al., 2025) with a split client-server architecture, where most classical feature extraction is performed locally and the quantum backend is reserved for compact server-side inference. Such deployment, however, introduces two practical constraints. First, information transmitted to the remote server may reveal privacy-sensitive client data. Second, quantum inference remains constrained by qubit count, circuit depth, measurement shots, and serving cost. These constraints become particularly important for task-varying prediction, where repeatedly retraining a quantum model for each new task would be impractical.

In-context learning (ICL) offers an effective approach for fast adaptation from a small labelled context set, without task-specific fine-tuning (Dong et al., 2024). This is particularly attractive for tabular problems, where tasks are often small, heterogeneous, and frequently changing. However, existing tabular ICL models (Qu et al., 2025; Hollmann et al., 2022) are not directly suited to resource-constrained split quantum inference. Their rich contextual representations are typically row-level, high-dimensional, and dependent on the size of the context set. Transmitting such representations incurs communication costs that scale with the number of context records, while their dimensionality is poorly matched to few-qubit quantum circuits. A natural solution is to summarize the contextual representation before it is transmitted to the quantum server. However, compression alone does not determine whether the downstream quantum model will remain effective. The classical summarization stage also determines what task-relevant information remains available to the quantum predictor. Thus, the central challenge is not simply to reduce the **dimensionality** of the contextual representation, but to determine which **task-relevant structure** must be preserved in the quantum-classical interface so that a resource-constrained quantum predictor can still make effective decisions.

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.

Privacy further constrains this interface design. Because the summarized task representation is transmitted to a remote server, it may reveal information about individual context records. The uploaded interface therefore needs to support explicit record-level privacy protection in addition to being compact and compatible with limited quantum resources. In particular, its sensitivity to any single context record should be bounded so that Gaussian perturbation can be calibrated without introducing excessive distortion to the information used for prediction. Under these resource and privacy constraints, we focus on two forms of task-relevant structure. The first is `bounded class-level structure`, which preserves compact discriminative class information while enabling explicit sensitivity control. The second is `candidate-relative structure`, which captures comparative relations between each candidate and its strongest competitors before quantum inference. An effective design therefore preserves `bounded class-level structure` at the client–server interface while constructing `candidate-relative structure` on the server.

Motivated by these requirements, we propose *QICL*, a privacy-aware split quantum-classical framework for tabular in-context learning built around a structured quantum-classical interface. On the client side, *QICL* converts the row-level context representations into bounded class-level summaries while class-prior information is obtained directly from the labelled context. The summaries and class priors are jointly privatized before transmission, forming a compact record-level privatized interface at the client–server boundary. On the server side, *QICL* performs candidate-aware relational prediction: for each candidate class, the query, candidate representation, and strongest competing classes are modeled jointly together with similarity, prior, and concentration statistics before being compressed into quantum-compatible coordinates and processed by a shared few-qubit predictor. In this way, *QICL* preserves compact class-level structure through the privatized interface while constructing candidate-relative relational geometry on the server before quantum inference. Our contributions are:

- **A unified framework for quantum-classical tabular in-context learning.** We present the *QICL* framework that jointly addresses privacy protection, communication-efficient task summarization, and resource-constrained quantum inference for tabular ICL.
- **A bounded structured summarization module.** We design a structured summarization module that converts variable-sized labelled contexts into compact class-level summaries and class-prior information. Its construction enables explicit sensitivity control and block-wise Gaussian privatization. We establish record-level differential privacy for the released interface and characterize prediction stability under the resulting perturbation.
- **A candidate-aware relational quantum predictor.** We develop a candidate-aware prediction module that constructs candidate-specific comparative representations before quantum inference, allowing the limited quantum input budget to focus on discriminative inter-class structure rather than isolated class embeddings.
- **Empirical evaluation on tabular benchmarks.** We compare *QICL* with classical baselines under matched privacy and budget settings. The results show that *QICL* achieves a favorable privacy–utility–communication trade-off.

2 BACKGROUND AND RELATED WORK

QICL builds on tabular in-context learning, parameterized quantum circuits, and differential privacy. We give only the essential background, with further details provided in Appendix A and Appendix B.

Tabular In-Context Learning. In-context learning (ICL) refers to the ability of a model to adapt to a new task by conditioning on a labelled context set at inference time, without task-specific gradient-based fine-tuning (Dong et al., 2024). In the tabular setting considered in this paper, a task τ is represented by a context set $C_\tau = \{(x_i, y_i)\}_{i=1}^{N_{ctx}}$, where N_{ctx} denotes the number of context examples in task τ, x_i is a table row and y_i is the target. The goal is therefore to learn a predictor $p(y_q \mid x_q, C_\tau)$, where x_q is a query example and its label y_q needs to be predicted without retraining the model for each new task and to perform prediction in a single forward pass. In this work, we use *TabICL* as the classical representation backbone because of its superior performance and modular design. It separates row-level representation extraction from downstream dataset-level prediction. This allows the client to keep raw tabular records and representation extraction local, while replacing the downstream prediction stage with a compact quantum-compatible predictor.

Parameterized Quantum Circuits. A parameterized quantum circuit (PQC) maps a classical input to a quantum state through a sequence of data-dependent and trainable unitary operations, and then produces outputs through quantum measurement. *Data re-uploading* (Pérez-Salinas et al., 2020) interleaves trainable circuit blocks with data-encoding blocks so that the same classical input can be injected repeatedly across multiple circuit layers. Such repeated encoding is useful for a few-qubit system, where expressive prediction must be achieved under tight limits on qubit count and circuit depth. The resulting measurement outcomes are used as features for candidate-wise prediction.

Differential Privacy. Differential privacy (DP) bounds how much the output distribution of a mechanism changes when one input record is modified (Dwork & Roth, 2014). Gaussian mechanism calibrates noise according to the ℓ_2-sensitivity to provide protection. We use this mechanism to protect uploaded interface and rely on the post-processing property to ensure that subsequent quantum prediction does not weaken the upload-level guarantee.

Related Work. Recent progress in tabular learning has moved from task-specific deep models toward pretrained in-context learning frameworks. Early attention-based architectures such as FT-Transformer (Gorishniy et al., 2021) and SAINT (Somepalli et al., 2021) demonstrated strong representation learning for tabular data, but typically require dataset-specific training or fine-tuning. Prior-Data Fitted Networks (PFNs) (Müller et al., 2022) address this limitation by pretraining on synthetic tasks so that new tasks can be solved through in-context prediction without task-specific optimization, with *TabPFN* (Hollmann et al., 2022) providing a prominent tabular instantiation. *TabICL* (Qu et al., 2025) improves scalability to larger datasets by constructing fixed-dimensional row representations followed by an efficient dataset-level in-context learning module. Privacy concerns in this setting have motivated *DP-TabICL* (Carey et al., 2024), which applies DP to tabular demonstrations before serialization and prompting. In contrast, our work considers privacy-preserving split quantum-classical tabular ICL, where a bounded class-wise interface is privatized before transmission and subsequently used for few-qubit quantum prediction.

3 QICL Framework

3.1 Problem setup

We consider a split quantum-classical tabular ICL setting with split deployment between a privacy-sensitive client and a central server. Considering the ICL task introduced in Section 2, for a task τ, the client holds the labelled context $\mathcal{C}_\tau$ and query x_q, performs classical representation processing locally, and communicates with a remote server for prediction through a compact interface.

Threat model and privacy objectives. The tabular data associated with each client may contain privacy-sensitive information, including feature values and target labels. The context set $\mathcal{C}_\tau$ is treated as privacy-sensitive. We assume an honest-but-curious server that may attempt to infer sensitive data from any information it observes during serving. For each task, we assume a fixed candidate label set that is shared across neighboring contexts, and replacements do not introduce or remove candidate classes. The privacy guarantee in this work protects the labelled context records, while the query is treated as fixed under the neighboring relation.

Target. Our aim is to design a split quantum-classical tabular ICL framework that (i) supports fast task-adaptive prediction for new tasks under limited quantum resources, (ii) reduces client-server communication, and (iii) provides privacy protection for the uploaded information.

3.2 Overview of QICL

We present *QICL* framework as a joint privacy-communication-resource design, which implements the two structural principles: `bounded class-level structure` at the client–server interface and `candidate-relative structure` on the server. As shown in Figure 1, the framework consists of three components: `structured task summarization`, `interface perturbation`, and `candidate-aware quantum prediction`.

On the client side, the `structured task summarization` module maps the contextual representations through a bounded feature mapping and aggregates them within each class to obtain compact class-wise summaries. Class-prior information is obtained directly from the labelled context, while the query is mapped separately into the same latent space. The `interface perturbation` module then jointly privatizes the class summaries and class priors before transmission to the server,

providing record-level privacy protection at the client–server boundary. On the server side, the `candidate-aware quantum prediction` module first uses the privatized class-level information and the separately encoded query to construct candidate-relative relational representations. For each candidate, the server identifies its strongest rivals and combines latent vector relations with explicit similarity, prior, and concentration statistic before compressing them into quantum-compatible coordinates for shared few-qubit prediction.

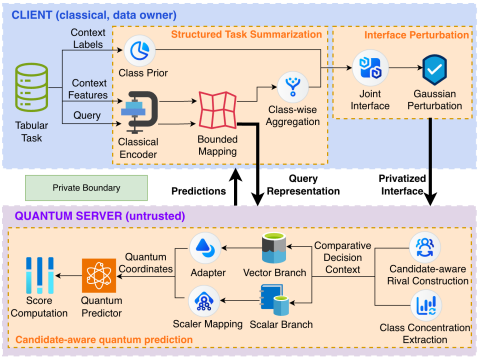


Figure 1: Overview of *QICL* framework

QICL follows an offline meta-training and online inference setting. During offline training, three modules are optimized over a distribution of tabular tasks, while the tabular representation backbone is kept fixed. During online inference, the client transmits the privatized interface with the separately encoded query representation. For a fixed context, the client samples and transmits privatized interface only once, and the server reuses the same interface for all associated queries. The query is outside the context-level neighboring relation and is therefore not perturbed. No task-specific gradient update is required for a new test task.

3.3 Task Summarization and Interface Perturbation

The first two modules define the client-side privatized interface of *QICL*. Starting from the row-level representations produced by the classical frozen Tabular ICL backbone, the client-side objective is to preserve compact class-level structure while ensuring that the contribution of each context record remains bounded. *QICL* achieves this through a bounded per-record mapping and class-wise aggregation, while class-prior information is obtained directly from the context labels. The resulting joint interface is privatized locally before transmission to the quantum server.

To ensure record-level sensitivity control, *QICL* uses the frozen *TabICL* front-end in a record-local mode, where each context row is encoded independently as h_i. Hence, replacing one context record affects only its own representation. The query is encoded independently in the same manner. We next map each row representation to a compact phase representation. Specifically, a learned projection maps h_i to a K-dimensional phase vector ϕ_i. To construct a bounded representation while retaining information at multiple periodic scales, we apply a multi-harmonic circular mapping. Let $\mathcal{H}$ denote the set of retained harmonic orders. For each phase coordinate and each $k \in \mathcal{H}$, we construct its sine and cosine responses,

$$\psi_{\mathrm{per}}(h_i) = \frac{1}{\sqrt{K|\mathcal{H}|}} \left[\{\cos(k\phi_i), \sin(k\phi_i)\}_{k\in\mathcal{H}} \right], \tag{1}$$

where the trigonometric functions are applied element-wise. The resulting periodic representation has $2K|\mathcal{H}|$ coordinates and satisfies $\|\psi_{\mathrm{per}}(h_i)\|_2 = 1$. This bounded construction directly controls the contribution of each context record and therefore facilitates record-level sensitivity calibration. At the same time, the multi-harmonic representation provides frequency-dependent information that is naturally compatible with data-encoded PQCs, whose outputs admit Fourier-type representations (Schuld et al., 2021). The bounded record-level representations are then aggregated within each class. Let $I_c = \{i : y_i = c\}$ denote the set of context indices belonging to class c, and let $n_c = |I_c|$. We construct the class-level summary as

$$S_c = \frac{1}{n_c} \sum_{i\in I_c} \psi_{\mathrm{per}}(h_i). \tag{2}$$

This class-wise aggregation converts the variable-sized context into a compact set of class representations whose size depends on the number of classes rather than on the number of context examples. The query is processed using the same phase transformation and periodic mapping, yielding $q = \psi_{\mathrm{per}}(h_q)$, so that the query and class summaries lie in the same representation space.

The class-prior information is obtained directly from the context labels. For each class c, we define the class prior as $\pi_c = \frac{n_c}{N_{\mathrm{ctx}}}$, where N_{ctx} denotes the total number of context examples. Collecting the class-wise summaries and priors gives $\mathbf{S} = \mathrm{vec}(S_1, \ldots, S_C)$ and $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_C]^\top$, and we define the joint interface as $\mathbf{F} = \begin{bmatrix} \mathbf{S} \\ \boldsymbol{\pi} \end{bmatrix}$. We privatize this interface using a block-wise Gaussian mechanism,

$$\widetilde{\mathbf{F}} = \mathbf{F} + \boldsymbol{\xi}, \qquad \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathrm{G}}), \tag{3}$$

where $\boldsymbol{\Sigma}_{\mathrm{G}} = \mathrm{diag}(\sigma_S^2 I_{D_S}, \sigma_\pi^2 I_{D_\pi})$. Here, σ_S and σ_π denote the Gaussian noise standard deviations for the summary and prior blocks, respectively, while $D_S = \dim(\mathbf{S})$ and $D_\pi = \dim(\boldsymbol{\pi})$ denote their corresponding dimensions. Accordingly, the privatized interface can be written as $\widetilde{\mathbf{F}} = [\widetilde{\mathbf{S}}; \widetilde{\boldsymbol{\pi}}]$, where $\widetilde{\mathbf{S}}$ and $\widetilde{\boldsymbol{\pi}}$ denote the Gaussian-perturbed class-summary and class-prior blocks, respectively. The two block-specific noise scales are jointly calibrated to a single overall privacy target according to their respective sensitivities and allocation coefficients, as detailed in Section 3.5.1.

3.4 Candidate-Aware Relational Quantum Prediction

Given the privatized class-level interface $\widetilde{\mathbf{F}} = [\widetilde{\mathbf{S}}; \widetilde{\boldsymbol{\pi}}]$ and the query representation q constructed on the client, the server constructs a candidate-relative representation before quantum inference. Rather than scoring each class only through its isolated relation to the query, ***QICL*** compares each candidate with its strongest competing classes (rivals). This allows discriminative inter-class relations to be formed before the representation is compressed to the limited input space of the quantum predictor.

As a deterministic post-processing step, the server derives a noisy concentration statistic from each privatized class summary, $\widetilde{\kappa}_c = \|\widetilde{\mathbf{S}}_c\|_2$. This operation incurs no additional privacy cost. Apart from the effect of Gaussian perturbation, a larger value indicates greater coherence among the mapped representations within the class, whereas a smaller value suggests greater dispersion.

For each candidate class c, ***QICL*** identifies its strongest rivals according to the query. Let $a_j = \mathrm{sim}(q, \widetilde{S}_j)$ denote the cosine similarity between the query and class j. ***QICL*** selects the J strongest rivals and orders them by decreasing similarity $(r_1(c), \ldots, r_J(c)) = \mathrm{TopJ}_{j \neq c}\, a_j$, where J denotes the number of retained rivals. We then represent the candidate-specific decision context through two complementary branches. The vector branch retains the representations required for joint relational modeling, $v_c = \left[q; \widetilde{S}_c; \widetilde{S}_{r_1(c)}; \ldots; \widetilde{S}_{r_J(c)}\right]$. In parallel, the scalar branch provides explicit comparative statistics. For candidate c, we construct

$$s_c = [a_c, \{a_c - a_{r_j(c)}\}_{j=1}^J, \widetilde{\pi}_c, \{\widetilde{\pi}_c - \widetilde{\pi}_{r_j(c)}\}_{j=1}^J, \widetilde{\kappa}_c, \{\widetilde{\kappa}_c - \widetilde{\kappa}_{r_j(c)}\}_{j=1}^J]. \tag{4}$$

The three groups provide explicit comparative cues based on query–class similarity a_c, privatized class prior $\widetilde{\pi}$, and noisy concentration statistic $\widetilde{\kappa}$. While the vector branch retains the representations used for joint relational modeling, the scalar branch explicitly exposes candidate–rival differences that may otherwise be difficult for the quantum model to recover.

In addition to the explicit scalar cues, the vector branch jointly models the query, candidate, and rival representations before quantum compression. Instead of independently projecting these representations, we concatenate them and apply a shared adapter, $h_c = g_\omega^{\mathrm{adp}}(v_c)$ followed by an angle projection, $\theta_c^{\mathrm{vec}} = g_\omega^{\mathrm{angle}}(h_c)$. This allows cross-representation interactions to be formed before the low-dimensional quantum encoding and provides a complementary source of relational information to the explicit scalar branch. The scalar statistics are mapped independently to a second set of quantum input coordinates, $\theta_c^{\mathrm{sca}} = g_{\omega_s}^{\mathrm{sca}}(s_c)$. The two branches are normalized separately using branch-wise RMS scaling and fused by element-wise addition. Each branch is mapped to tensors of shape $R \times n_q \times 2$, where R denotes the number of data-reuploading rounds and n_q the number of qubits. The fused quantum input θ_c is then encoded by the shared n_q-qubit PQC. Its measured observables are passed to a lightweight classical readout to obtain the candidate score ℓ_c. The same quantum predictor is shared across all candidate classes and the final prediction is $\hat{y} = \arg\max_c \ell_c$.

3.5 Privacy Analysis and Prediction Stability

We now establish record-level differential privacy for the joint interface and characterize the effect of interface perturbation on prediction stability. Proofs are provided in Appendix C.

3.5.1 Upload-Level Privacy Guarantee against the untrusted server

We consider upload-level privacy against an honest-but-curious server that observes the privatized interface, with the labelled context set treated as private. Two contexts $\mathcal{C}$ and $\mathcal{C}'$ are adjacent if they have the same size and differ in at most one record. Under the fixed candidate-label-set assumption introduced in Section 3.1, we further assume that every class affected by a replacement contains at least $n_{\min}$ records in both adjacent contexts, where $n_{\min}$ is a prescribed minimum threshold fixed independently of the private context, for example using public data or non-private domain knowledge.

Proposition 1 (Sensitivity of the interface) *Under record-level replace-one adjacency, the class-summary and class-prior blocks satisfy*

$$\Delta_S \le \frac{2\sqrt{2}}{n_{\min}}, \qquad \Delta_\pi \le \frac{\sqrt{2}}{N_{\mathrm{ctx}}}, \tag{5}$$

where $\Delta_S = \sup_{\mathcal{C}\sim\mathcal{C}'} \|\mathbf{S}(\mathcal{C}) - \mathbf{S}(\mathcal{C}')\|_2$ *and* $\Delta_\pi = \sup_{\mathcal{C}\sim\mathcal{C}'} \|\boldsymbol{\pi}(\mathcal{C}) - \boldsymbol{\pi}(\mathcal{C}')\|_2$. *If the replaced record retains its class label, the tighter bounds* $\Delta_S \le 2/n_{\min}$ *and* $\Delta_\pi = 0$ *hold.*

For the block covariance $\boldsymbol{\Sigma}_{\mathrm{G}}$ in Equation (3), define the covariance-normalized sensitivity as $\Delta_{\mathrm{G}} = \sup_{\mathcal{C}\sim\mathcal{C}'} \left\| \boldsymbol{\Sigma}_{\mathrm{G}}^{-1/2} \big(\mathbf{F}(\mathcal{C}) - \mathbf{F}(\mathcal{C}')\big) \right\|_2$. Using Equation (5), we have $\Delta_{\mathrm{G}}^2 \le \frac{\Delta_S^2}{\sigma_S^2} + \frac{\Delta_\pi^2}{\sigma_\pi^2}$. Therefore, calibrating the two block-specific noise scales so that the resulting sensitivity satisfies the Gaussian mechanism condition yields a privacy guarantee for the joint release.

Theorem 1 (DP guarantee for joint privatized interface) *Let* $\zeta_S, \zeta_\pi > 0$ *be allocation coefficients satisfying* $\zeta_S^2 + \zeta_\pi^2 = 1$. *Using the sensitivity bounds in Equation* (5)*, if the block-specific Gaussian noise scales satisfy*

$$\sigma_S \ge \frac{2\sqrt{2}}{n_{\min}\zeta_S} \frac{\sqrt{2\ln(1.25/\delta)}}{\epsilon}, \qquad \sigma_\pi \ge \frac{\sqrt{2}}{N_{\mathrm{ctx}}\zeta_\pi} \frac{\sqrt{2\ln(1.25/\delta)}}{\epsilon}, \tag{6}$$

then the joint release $\widetilde{\mathbf{F}}$ *satisfies* (ϵ, δ) *record-level differential privacy.*

3.5.2 Prediction Stability under privatized Interface Perturbation

We next characterize how the Gaussian perturbation introduced for upload-level privacy affects the final prediction. For a fixed query, let $\ell_c(\mathbf{F})$ denote the score assigned to candidate class c. Because rival selection is discrete, we restrict the analysis to a perturbation neighborhood in which the ordered rival tuple $(r_1(c), \ldots, r_J(c))$ remains unchanged for each candidate class. A sufficient rival-ranking margin condition for this assumption under Gaussian perturbation is provided in Appendix C.3. We assume that each candidate-score function is L-Lipschitz throughout this neighborhood, i.e., $|\ell_c(\mathbf{F}) - \ell_c(\mathbf{F}')| \le L\|\mathbf{F} - \mathbf{F}'\|_2$. Let $c^\star = \arg\max_c \ell_c(\mathbf{F})$ denote the clean prediction, and define its margin as $\gamma = \ell_{c^\star}(\mathbf{F}) - \max_{c \ne c^\star} \ell_c(\mathbf{F})$.

Proposition 2 (Margin-Based Prediction Stability) *Let* $\sigma_{\max} = \max\{\sigma_S, \sigma_\pi\}$. *For any* $\eta \in (0, 1)$*, the prediction obtained from the privatized interface* $\widetilde{\mathbf{F}}$ *remains identical to the clean prediction* $c^\star$ *with probability at least* $1 - \eta$ *if the clean prediction margin* γ *satisfies*

$$\gamma > 2L \left[\sqrt{D_S\sigma_S^2 + D_\pi\sigma_\pi^2} + \sigma_{\max}\sqrt{2\log(1/\eta)} \right]. \tag{7}$$

Proposition 2 makes the privacy–utility relation explicit: the clean prediction is preserved when the score gap between the predicted class and its strongest competitor is sufficiently large relative to the perturbation induced by the privatized interface. The latter depends on the local score sensitivity L,

the block dimensions D_S and D_π, and the perturbation scales σ_S and σ_π. Conversely, a stronger privacy requirement (smaller ϵ) increases the perturbation scales and therefore requires a larger clean score margin to preserve the prediction. Thus, compressing the context into a compact class-wise interface has a dual role: it reduces client–server communication and limits the aggregate perturbation affecting prediction. Together with Proposition 1, larger minimum class support further reduces the sensitivity and required noise of the summary block under the same privacy protection level.

4 EXPERIMENTS

Our experiments address three questions corresponding to the main components of *QICL*: (i) whether the compact class-wise interface can preserve predictive utility under differential privacy (Section 4.2); (ii) what task-relevant structure should be retained in the interface (Section 4.3); and (iii) what the candidate-aware quantum predictor contributes beyond the structured interface (Section 4.4).

4.1 DATASET AND SETTING

Datasets. We evaluate on 200 classification datasets from the TALENT benchmark (Ye et al., 2024), with the number of classes ranging from 2 to 100. Among them, 12 datasets contain more than 10 classes. Accuracy is computed per dataset and then averaged across datasets, with means and standard deviations reported over five random seeds.

Model configuration. Unless otherwise specified, *QICL* uses 8 qubits three-round data-reuploading PQC. The summary uses $K = 4$ latent phase coordinates and harmonic orders $\mathcal{H} = \{1, \ldots, 9\}$, resulting in a 72-dimensional record-level representation. Together with the class prior, each class contributes a 73-dimensional privatized interface. We retain $J = 2$ rivals for each candidate class. For binary tasks, the sole rival is duplicated to maintain a fixed-size candidate-aware input. The interface is privatized by Gaussian perturbation with privacy budget $(\epsilon = 1, \delta = 10^{-5})$ before transmission to the server. Full architectural and training details are provided in Appendix D.1.

Baselines. We compare *QICL* against three groups of baselines under matched privacy settings. (1) Uncompressed tabular ICL models. We include the uncompressed ICL models *TabICL* (Qu et al., 2025) and *TabPFN-v2* (Hollmann et al., 2025); (2) TabICL-based summarization variants. We construct three dimensionality-matched TabICL summarization baselines: *TabICL-Mean*, *TabICL-AttnPool*, *TabICL-SetTrans*, using mean pooling, attention pooling, and a Set Transformer, respectively. These variants use a lightweight classical TabICL-style predictor whose size is matched to the quantum predictor; (3) Conventional tabular predictors. *CatBoost* (Prokhorenkova et al., 2018) and *TabM* (Gorishniy et al., 2024) are included as representative high-performing tree-based and deep tabular predictors, respectively. Since these models are trained directly on the private task data rather than performing in-context inference through an uploaded context representation, DP is applied through perturbation of the input data prior to model fitting.

4.2 QICL PERFORMANCE

Table 1: Average accuracy under different privacy budgets.

Method	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 4$
QICL	**0.532** $\pm$ 0.006	**0.535** $\pm$ 0.006	**0.548** $\pm$ 0.005
TabICL	0.381 $\pm$ 0.022	0.403 $\pm$ 0.013	0.409 $\pm$ 0.022
TabPFNv2	0.479 $\pm$ 0.019	0.491 $\pm$ 0.011	0.511 $\pm$ 0.018
TabICL-Mean	0.441 $\pm$ 0.022	0.444 $\pm$ 0.018	0.447 $\pm$ 0.019
TabICL-AttnPool	0.432 $\pm$ 0.023	0.440 $\pm$ 0.019	0.442 $\pm$ 0.020
TabICL-SetTrans	0.424 $\pm$ 0.039	0.426 $\pm$ 0.035	0.433 $\pm$ 0.029
CatBoost	0.464 $\pm$ 0.008	0.495 $\pm$ 0.009	0.527 $\pm$ 0.003
TabM	0.461 $\pm$ 0.013	0.484 $\pm$ 0.011	0.515 $\pm$ 0.005

Table 2: Client-to-server communication cost under split in-context inference.

Method	Cost
QICL	$73C + 72N_q$
TabICL	$512(N_{\text{ctx}} + N_q) + N_{\text{ctx}}$
TabPFN-v2	$d(N_{\text{ctx}} + N_q) + N_{\text{ctx}}$

Table 1 reports the average classification accuracy under three upload-level privacy budgets. *QICL* achieves the highest mean accuracy across all three privacy settings, with performance generally improving as ϵ increases. This trend is consistent with Proposition 2, since stronger privacy requires larger interface perturbation relative to the clean prediction margin. Figure 2 further reports dataset-level ranks. Lower rank indicates better relative performance. It shows that *QICL* maintains

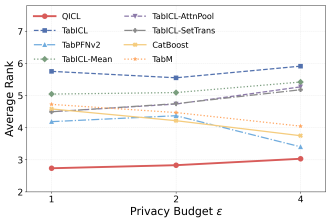


Figure 2: Average dataset-level rank of each method under different privacy budgets

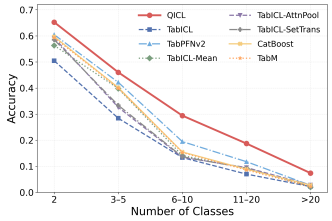


Figure 3: Accuracy across datasets grouped by the number of classes

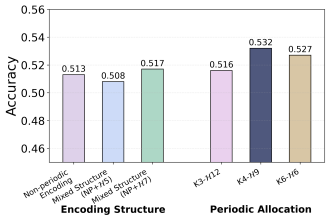


Figure 4: Accuracy under different summary encoding strategies

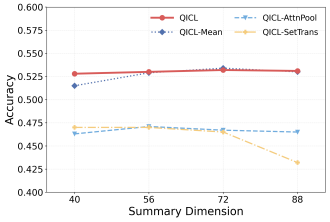


Figure 5: Effect of summary dimension on prediction accuracy

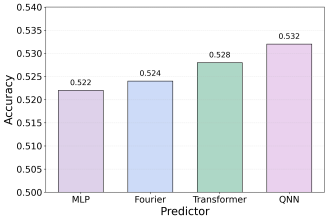


Figure 6: Predictor comparison with same interface

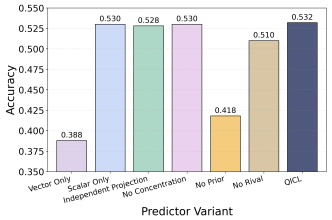


Figure 7: Ablation of the candidate-aware relational predictor

strong relative performance across the benchmark datasets rather than only in aggregate accuracy. Both Table 1 and Figure 2 further show that *QICL* maintains a larger relative advantage over the uncompressed and conventional baselines at smaller privacy budgets. This pattern suggests that class-wise aggregation is particularly beneficial under stronger privacy, since the private context is compressed before perturbation rather than being represented and protected at the record level. Figure 3 further groups the datasets according to their number of target classes. The degradation with increasing class count is attributable not only to the greater difficulty of the classification problem, but also to the larger privatized interface. This trend is consistent with Proposition 2, in which a larger interface increases the aggregate perturbation term. Nevertheless, *QICL* maintains stronger performance across the displayed class-count groups.

Table 2 compares the client-to-server communication required by the split in-context inference methods. The compressed TabICL variants are omitted because their dimensions are matched to those of *QICL*. *QICL* requires $73C+72N_q$, whereas *TabICL* and *TabPFN-v2* require $512(N_{\text{ctx}}+N_q)+N_{\text{ctx}}$ and $d(N_{\text{ctx}}+N_q)+N_{\text{ctx}}$, respectively. Thus, QICL changes the context-dependent communication from record-wise scaling in N_{ctx} to class-wise scaling in C.

Real quantum hardware validation. We additionally deploy the simulation-trained *QICL* model on `ibm_kingston` under $\epsilon = 1$ using 64 shots. Across the first nine TALENT datasets, hardware accuracy is 0.500, compared with 0.751 in simulation, with the gap consistent with device, transpilation, and finite-shot effects. These results demonstrate execution feasibility on real quantum hardware.

4.3 Analysis of the Structured Summary Interface

Effect of summary encoding. Figure 4 compares different summary encoding schemes under a fixed summary dimension. The non-periodic (NP) scheme uses only bounded non-periodic features, the periodic scheme uses multi-harmonic mapping, and the hybrid variants combine both components. K denotes the number of projected phase channels and $\mathcal{H}$ denotes the set of retained harmonic orders. *The periodic encoding achieves the strongest performance among the compared summary structures.* This result indicates that preserving more heterogeneous feature types does not necessarily improve downstream utility; the structure of the compressed interface is itself important. The result is also consistent with the Fourier structure of data-encoded PQCs, whose outputs naturally contain trigonometric components (Schuld et al., 2021). Thus, the multi-harmonic summary provides an interface whose representation structure is well aligned with the downstream quantum predictor.

Trade-off between summary dimensions and privacy. Figure 5 isolates the effect of the summary interface by keeping the downstream candidate-aware quantum predictor fixed across all variants.

Accordingly, *QICL-Mean*, *QICL-AttnPool*, and *QICL-SetTrans* differ from *QICL* only in the summary construction module; all subsequent candidate-aware processing and quantum prediction components are unchanged. First, increasing the summary dimension does not monotonically improve performance. Across the compared methods, the best accuracy is typically obtained at 72 dimensions, while the improvement over 56 dimensions is small. Although a larger interface can preserve more task information, it also increases D_S. According to Proposition 2, this enlarges the aggregate perturbation term, producing a direct trade-off between representational capacity and privacy-induced distortion. If communication cost is a stronger concern, the 56-dimensional interface provides a reasonable trade-off. Second, *QICL* achieves the strongest performance across most dimensions, and its advantage is more pronounced in the lower-dimensional settings. This suggests that the proposed structured summary is more *dimension-efficient*: it concentrates the available interface capacity on task-relevant class structure that remains useful after compression and privacy perturbation. By comparison, *QICL-Mean* becomes competitive only at larger dimensions, while *QICL-AttnPool* and *QICL-SetTrans* remain consistently weaker, suggesting that additional pooling flexibility does not automatically improve utility under privacy and compression constraints.

4.4 Analysis of the Candidate-Aware Quantum Predictor

Quantum versus classical predictor. Figure 6 compares different predictors under the same upstream *QICL* interface and evaluation setting. The quantum predictor shows a modest positive gap over all three classical heads. A likely contributor is the way the quantum predictor processes the resulting relational coordinates. While the Fourier predictor captures periodic structure through an explicit classical sinusoidal representation, the data-reuploading PQC repeatedly injects the same coordinates through trainable rotations and entangling layers, enabling a learnable composition of trigonometric responses and cross-coordinate interactions. This provides a richer nonlinear mechanism for exploiting the periodic and candidate-relative structure presented by the *QICL* interface. This result suggests that the benefit of the quantum predictor depends on presenting it with task-relevant relational structure before the few-qubit bottleneck.

Ablation of candidate-aware relational modeling. Figure 7 evaluates the main components of the candidate-aware predictor. *Vector Only* and *Scalar Only* retain only one of the two branches, *Independent Projection* removes the joint relational projection, *No Prior* and *No Concentration* remove the class-prior and concentration statistics, respectively, and *No Rival* removes the competing classes from candidate-wise prediction. It indicates *the scalar branch provides the dominant predictive signal*, suggesting that explicit comparative cues, particularly query–class similarity, privatized class-prior information, and rival-relative differences, remain highly useful after context compression. The relatively small gap between the full model and Independent Projection further indicates that joint vector projection provides a complementary rather than dominant contribution. In contrast, *removing rival information causes a clearer degradation*, supporting the candidate-aware design and the importance of explicit candidate-relative comparison. Modeling the strongest rivals provides decision-relevant contrast that cannot be obtained from the query–candidate relation alone, allowing the predictor to form more discriminative candidate-relative representations before quantum inference.

5 Conclusion

This paper presented *QICL*, a privacy-aware split quantum-classical framework for tabular in-context learning under communication, privacy, and quantum-resource constraints. *QICL* combines a bounded class-level interface with server-side candidate-relative modeling and few-qubit prediction. The proposed interface provides record-level DP while changing context communication from record-wise to class-wise scaling. Experiments show that class-wise aggregation is particularly beneficial under stronger privacy, while the proposed structured summary is especially effective under tighter interface budgets. Besides, candidate-relative cues and rival modeling further improve prediction. Real-hardware experiments further demonstrate the feasibility of the proposed split inference. A remaining limitation is that the current analysis does not explicitly account for hardware-induced quantum noise; future work will investigate its impact and develop hardware-aware training strategies for more reliable deployment on physical quantum devices.

AI USE STATEMENT

We did not use generative AI tools for any tasks requiring disclosure under the ICLR 2027 AI Policy. Generative AI tools were used to assist with language editing and polishing the manuscript, including improvements to grammar, clarity, and readability. All AI-assisted edits were reviewed by the authors, who take full responsibility for the final content of the paper.

ETHICS STATEMENT

This work studies privacy-preserving tabular in-context learning and does not involve human-subject experiments or the collection of new personal data. Our privacy analysis focuses on protecting record-level information exposed through the uploaded task summaries using differential privacy. This study does not pose additional ethical risk.

REPRODUCIBILITY STATEMENT

Details of the datasets and models are provided in Section 4.1 and Appendix D.1. The implementation code is provided as supplementary material to facilitate reproducibility.

REFERENCES

Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum science and technology*, 4(4):043001, 2019.

Alycia N Carey, Karuna Bhaila, Kennedy Edemacu, and Xintao Wu. Dp-tabicl: In-context learning with differentially private tabular data. In *2024 IEEE International Conference on Big Data (BigData)*, pp. 1552–1557. IEEE, 2024.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, et al. A survey on in-context learning. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 1107–1128, 2024.

Cynthia Dwork. Differential privacy. In *International Colloquium on Automata, Languages, and Programming*, pp. 1–12. Springer, 2006.

Cynthia Dwork. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pp. 1–19. Springer, 2008.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and trends® in theoretical computer science*, 9(3-4):211–407, 2014.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. *Advances in neural information processing systems*, 34:18932–18943, 2021.

Yury Gorishniy, Akim Kotelnikov, and Artem Babenko. Tabm: Advancing tabular deep learning with parameter-efficient ensembling. *arXiv preprint arXiv:2410.24210*, 2024.

Kai Helli, David Schnurr, Noah Hollmann, Samuel Müller, and Frank Hutter. Drift-resilient tabpfn: In-context learning temporal distribution shifts on tabular data. *Advances in Neural Information Processing Systems*, 37:98742–98781, 2024.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Changzhou Long, Meng Huang, Xiucai Ye, Yasunori Futamura, and Tetsuya Sakurai. Hybrid quantum-classical-quantum convolutional neural networks. *Scientific Reports*, 15(1):31780, 2025.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=KSugKcbNf9.

Jaehyun Nam, Woomin Song, Seong Hyeon Park, Jihoon Tack, Sukmin Yun, Jaehyung Kim, and Jinwoo Shin. Semi-supervised tabular classification via in-context learning of large language models. In *Workshop on Efficient Systems for Foundation Models@ ICML2023*, 2023.

Hoa T Nguyen, Prabhakar Krishnan, Dilip Krishnaswamy, Muhammad Usman, and Rajkumar Buyya. Quantum cloud computing: a review, open problems, and future directions. *arXiv preprint arXiv:2404.11420*, 2024.

Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José I Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, 2020.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *International Conference on Machine Learning*, 2025.

Günther Schindler, Maximilian Schambach, Michael Medek, and Sam Thelin. Tabgemma: Text-based tabular icl via llm using continued pretraining and retrieval. *arXiv preprint arXiv:2511.03570*, 2025.

Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Physical Review A*, 103(3):032430, 2021.

Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C Bayan Bruss, and Tom Goldstein. Saint: Improved neural networks for tabular data via row attention and contrastive pre-training. *arXiv preprint arXiv:2106.01342*, 2021.

Han-Jia Ye, Si-Yang Liu, Hao-Run Cai, Qi-Le Zhou, and De-Chuan Zhan. A closer look at deep learning methods on tabular datasets. *arXiv preprint arXiv:2407.00956*, 2024.

Kamila Zaman, Alberto Marchisio, Muhammad Abdullah Hanif, and Muhammad Shafique. A survey on quantum machine learning: Current trends, challenges, opportunities, and the road ahead. *arXiv preprint arXiv:2310.10315*, 2023.

A Detailed Background

A.1 Tabular In-Context Learning

In-context learning (ICL) refers to the ability of a model to adapt to a new task by conditioning on a labelled context set at inference time, without task-specific gradient-based fine-tuning (Dong et al., 2024). In the tabular setting considered in this paper, a task τ is represented by a context set $C_\tau = \{(x_i, y_i)\}_{i=1}^{N_{ctx}}$, where N_{ctx} denotes the number of context examples in task τ, x_i is a table row and y_i is the target. The goal is therefore to learn a predictor $p(y_q \mid x_q, C_\tau)$, where x_q is a query example and its label y_q needs to be predicted without retraining the model for each new task.

Among recent tabular ICL models (Qu et al., 2025; Schindler et al., 2025; Hollmann et al., 2022), *TabICL* provides a suitable classical backbone because of its superior performance and modular architecture. *TabICL* is a tabular foundation model pretrained on synthetic classification tasks and designed to perform prediction in a single forward pass. This separation is suitable for our split setting. On the one hand, it first converts raw tabular rows into dense row-level latent representations and then performs dataset-level in-context learning over these compact embeddings. The front-end representation encoder can remain on the client side, so raw tabular rows do not need to be uploaded. On the other hand, the back-end prediction module can be redesigned for resource-constrained quantum inference.

Architecturally, ***TabICL*** consists of a column-wise embedding transformer, a row-wise interaction transformer, and a dataset-wise ICL transformer followed by a final prediction MLP. For a task τ with context set C_τ and query row x_q, the column-wise transformer first processes each feature column across the rows in C_τ to produce contextualized feature embeddings. Its role is to capture feature-specific statistical patterns and to make the representation of each value depend not only on the value itself, but also on the distributional context of the corresponding column. The resulting feature embeddings are then fed into the row-wise transformer, which models interactions among different features within each row and aggregates them into row-level latent representations. After this stage, each context row x_i and the query row x_q are mapped into a shared latent space h_i,

$$h_i = f_b(x_i), \qquad h_{q,\tau} = f_b(x_q) \tag{8}$$

Stacking the context-row latents gives

$$H_{C,\tau} = [h_1, \ldots, h_{N_{ctx}}]^\top \in \mathbb{R}^{N_{ctx} \times d} \tag{9}$$

where $H_{C,\tau}$ collects the row-level latent representations of the context. These row-level latents are further processed by a dataset-wise ICL Transformer together with a final prediction MLP, which can be written as

$$\hat{p}(y_q \mid x_q, C_\tau) = g_{\mathrm{MLP}}\Big(TF_{\mathrm{icl}}(H_{C,\tau}, h_{q,\tau}, \{y_i\}_{i=1}^{N_{ctx}})\Big). \tag{10}$$

where TF_{icl} denotes the dataset-wise ICL module and g_{MLP} denotes the final prediction multilayer perceptron. Thus, the original classical ICL head in ***TabICL*** consists of the dataset-wise Transformer together with the final prediction MLP.

In this work, we retain the ***TabICL*** representation backbone f_b on the client side and re-organize the downstream prediction process for split quantum-classical deployment. This preserves the strong tabular inductive bias of ***TabICL*** and reserves quantum computation for expressive prediction. One thing to note is that the description above refers to the original ***TabICL*** inference. ***QICL*** retains the pretrained front-end parameters but executes the representation stage in a record-local mode, as described in Section 3.3, so that each context row representation depends only on its corresponding record.

A.2 Differential Privacy

Differential privacy (DP) provides a formal guarantee of privacy protection Dwork (2008). A mechanism is differentially private if its output distribution does not change much when a single sample in the input dataset is modified. DP is defined as follows:

Definition 1 (ϵ, δ)***-Differential Privacy*** *Dwork (2006). A random algorithm $\mathcal{M}$ provides (ϵ, δ)-differential privacy if for any two datasets D and D' that differ in at most a single record, and for all outputs: $Pr[\mathcal{M}(D) \in O] \leq e^\epsilon Pr[\mathcal{M}(D') \in O] + \delta$. Here, ϵ controls the privacy loss, with smaller values corresponding to stronger privacy, while δ is a small failure probability that allows a limited relaxation of pure privacy.*

To calibrate noise for vector-valued outputs, a standard quantity is the ℓ_2-sensitivity. For a function f defined on datasets, its ℓ_2-sensitivity is $\Delta_2(f) = \sup_{D \sim D'} \|f(D) - f(D')\|_2$. This measures the maximum change in the output of f caused by modifying one sample in the input dataset. The Gaussian mechanism Dwork & Roth (2014) $M_{Gau}(D) = f(D) + (X_1, X_2, ..., X_d)$ satisfies DP if X_i are i.i.d random variables drawn from the Gaussian distribution with $\mu = 0$ and $\sigma = \frac{\Delta_2(f)\sqrt{2\log(\frac{1.25}{\delta})}}{\epsilon}$. A fundamental property of differential privacy is *post processing*. If $\mathcal{M}$ is (ε, δ)-DP and T is any randomized mapping that does not depend on the private dataset except through $\mathcal{M}(D)$, then the composition $T \circ \mathcal{M}$ also satisfies (ε, δ)-DP.

B Detailed Related Work

Recent progress in tabular learning has shown that pretrained and Transformer-based models can provide strong transfer across heterogeneous tabular tasks. Early deep tabular architectures such as FT-Transformer (Gorishniy et al., 2021) and SAINT (Somepalli et al., 2021) demonstrated that attention-based models can learn competitive tabular representations by modeling feature interactions

and interactions across samples. However, these models require case-by-case training or fine-tuning when facing a new dataset or task, which limits their ability to amortize learning across heterogeneous tabular problems. This limitation motivated the Prior-Data Fitted Networks (PFNs) (Müller et al., 2022), which aim to learn the learning algorithm itself by pretraining on synthetic tasks sampled from a prior, so that prediction on a new task can be performed in a single forward pass via in-context learning rather than task-specific optimization. *TabPFN* (Hollmann et al., 2022) is the tabular instantiation of this idea. It is a tabular foundation model trained offline to approximate Bayesian inference on synthetic tabular tasks, and it achieves strong performance on small tabular classification problems without hyperparameter tuning or per-task retraining. Building on this line, *Drift-Resilient TabPFN* (Helli et al., 2024) was proposed to address a limitation of temporal distribution shift. Instead of assuming an i.i.d. tabular environment, it extends the PFN to settings where the underlying data-generating process evolves over time, allowing in-context prediction under drift.

TabICL (Qu et al., 2025) was introduced to make tabular in-context learning work better on larger datasets. TabPFN-style models perform strongly on small tables, but they become computationally expensive when the training set grows. In particular, repeatedly applying column-wise and row-wise attention over many samples leads to high memory and computation cost. To overcome this limitation, *TabICL* proposes an architecture that first builds fixed-dimensional row embeddings through column-then-row attention, and then applies an efficient dataset-wise ICL module on top of these row representations. *TabICL* is reported to deliver overall performance comparable to that of *TabPFN* while being systematically faster, and to offer particular advantages on larger tables. Some studies combine tabular in-context learning with Large Language Models (LLMs) (Carey et al., 2024; Nam et al., 2023). The main idea is to serialize tabular rows into text and let an LLM perform prediction from text-form demonstrations. The advantage is that besides tabular prediction, LLM-based TabICL can potentially work together with broader LLM abilities such as conversational interaction, coding, and reasoning. However, because tabular instances must be converted into text, they consume many tokens, which makes existing LLM-based TabICL methods mostly limited to few-shot settings. Consequently, they still generally underperform the strongest specialized tabular models in overall predictive accuracy. Tabular in-context learning with LLM also raises a privacy concern. Once tabular records are serialized into prompts, sensitive information can be exposed through the input demonstrations themselves. *DP-TabICL* (Carey et al., 2024) studies how DP can be applied to protect tabular data before serialization and prompting, and proposes both local-DP and global-DP variants for private tabular ICL.

Quantum machine learning offers a new direction for this setting. In particular, parameterized quantum circuits can provide expressive nonlinear prediction Benedetti et al. (2019) with only a small number of trainable parameters and limited qubit budgets, making them attractive for resource-constrained inference. However, quantum in tabular in-context learning remains largely unexplored. A key obstacle is that existing tabular ICL models typically rely on high-dimensional contextual representations that are costly to transmit, difficult to protect, and poorly matched to few-qubit quantum circuits, especially in remote or split deployment where task information must cross the client-server boundary.

C Proofs Related to Privacy

C.1 Proof of Proposition 1

By Equation (1), each mapped context record satisfies $\|\psi_{\mathrm{per}}(h_i)\|_2 = 1$. By construction, the representation encoder used for the private interface is record-local: for fixed preprocessing, h_i depends only on the i-th record. Therefore, under replace-one adjacency, replacing one context record leaves the representations of all unchanged records identical and changes only the mapped representation associated with the replaced record. Let $\mathcal{A}$ denote the set of classes whose summaries differ between the neighbouring contexts $\mathcal{C}$ and $\mathcal{C}'$. Under replace-one adjacency, $|\mathcal{A}| \leq 2$. For any affected class $c \in \mathcal{A}$, the replacement either substitutes, removes, or adds one unit-norm record in the class-wise average defined in Equation (2). In all cases,

$$\|S_c(\mathcal{C}) - S_c(\mathcal{C}')\|_2 \leq \frac{2}{n_{\min}}, \tag{11}$$

because both an individual mapped record and any average of such unit-norm records have ℓ_2-norm at most one.

Since the affected class summaries occupy disjoint blocks in $\mathbf{S}(\mathcal{C})$,

$$\|\mathbf{S}(\mathcal{C}) - \mathbf{S}(\mathcal{C}')\|_2^2 = \sum_{c\in\mathcal{A}} \|S_c(\mathcal{C}) - S_c(\mathcal{C}')\|_2^2 \le |\mathcal{A}| \left(\frac{2}{n_{\min}}\right)^2 \le \frac{8}{n_{\min}^2}. \tag{12}$$

Hence, $\Delta_S \le \frac{2\sqrt{2}}{n_{\min}}$. For the class-prior vector $\boldsymbol{\pi} = [n_1/N_{\mathrm{ctx}}, \dots, n_C/N_{\mathrm{ctx}}]^\top$, a replacement that changes the label from class a to class b changes exactly two coordinates by $1/N_{\mathrm{ctx}}$ in magnitude. Therefore,

$$\|\boldsymbol{\pi}(\mathcal{C}) - \boldsymbol{\pi}(\mathcal{C}')\|_2^2 \le \frac{2}{N_{\mathrm{ctx}}^2}, \tag{13}$$

which gives $\Delta_\pi \le \frac{\sqrt{2}}{N_{\mathrm{ctx}}}$. If the replaced record retains its class label, the prior vector is unchanged and only one summary block can change, yielding $\Delta_S \le 2/n_{\min}$ and $\Delta_\pi = 0$.

C.2 Proof of Theorem 1

For the block-wise Gaussian mechanism in Equation (3), we can get

$$\mathbf{Y} = \boldsymbol{\Sigma}_{\mathrm{G}}^{-1/2}\widetilde{\mathbf{F}} = \boldsymbol{\Sigma}_{\mathrm{G}}^{-1/2}\mathbf{F}(\mathcal{C}) + \mathbf{Z} \qquad \mathbf{Z} \sim \mathcal{N}(\mathbf{0}, I) \tag{14}$$

Its ℓ_2-sensitivity is

$$\Delta_{\mathrm{G}} = \sup_{\mathcal{C}\sim\mathcal{C}'} \left\| \boldsymbol{\Sigma}_{\mathrm{G}}^{-1/2} \left(\mathbf{F}(\mathcal{C}) - \mathbf{F}(\mathcal{C}')\right) \right\|_2. \tag{15}$$

Since $\boldsymbol{\Sigma}_{\mathrm{G}}^{-1/2} = \mathrm{diag}\left(\sigma_S^{-1} I_{D_S}, \sigma_\pi^{-1} I_{D_\pi}\right)$, for any neighboring contexts $\mathcal{C} \sim \mathcal{C}'$ we have

$$\left\| \boldsymbol{\Sigma}_{\mathrm{G}}^{-1/2} \left(\mathbf{F}(\mathcal{C}) - \mathbf{F}(\mathcal{C}')\right) \right\|_2^2 = \frac{\|\mathbf{S}(\mathcal{C}) - \mathbf{S}(\mathcal{C}')\|_2^2}{\sigma_S^2} + \frac{\|\boldsymbol{\pi}(\mathcal{C}) - \boldsymbol{\pi}(\mathcal{C}')\|_2^2}{\sigma_\pi^2}.$$

Using the block sensitivity bounds in Proposition 1 gives

$$\Delta_{\mathrm{G}}^2 \le \frac{\Delta_S^2}{\sigma_S^2} + \frac{\Delta_\pi^2}{\sigma_\pi^2}. \tag{16}$$

Let $\zeta_S, \zeta_\pi > 0$ denote the allocation coefficients for the summary and prior blocks, respectively, satisfying $\zeta_S^2 + \zeta_\pi^2 = 1$. By adopting the Gaussian mechanism, we set

$$\sigma_S \ge \frac{\Delta_S}{\zeta_S} \frac{\sqrt{2\ln(1.25/\delta)}}{\epsilon}, \qquad \sigma_\pi \ge \frac{\Delta_\pi}{\zeta_\pi} \frac{\sqrt{2\ln(1.25/\delta)}}{\epsilon}. \tag{17}$$

The coefficients ζ_S and ζ_π control the relative allocation of the normalized sensitivity across the two blocks, while the condition $\zeta_S^2 + \zeta_\pi^2 = 1$ ensures that the joint covariance-normalized sensitivity satisfies a single (ϵ, δ) Gaussian privacy condition. Therefore,

$$\Delta_{\mathrm{G}}^2 \le \left(\zeta_S^2 + \zeta_\pi^2\right) \frac{\epsilon^2}{2\ln(1.25/\delta)} = \frac{\epsilon^2}{2\ln(1.25/\delta)}.$$

Hence, $\Delta_{\mathrm{G}} \le \frac{\epsilon}{\sqrt{2\ln(1.25/\delta)}}$. Since $\mathbf{Y}$ has unit Gaussian noise and satisfies (ϵ, δ)-DP, $\widetilde{\mathbf{F}}$ also satisfies (ϵ, δ)-DP.

C.3 Proof of Proposition 2

Recall that the privatized interface as in Equation (3). Let $\boldsymbol{\xi} = \boldsymbol{\Sigma}_{\mathrm{G}}^{1/2}\mathbf{Z}, \mathbf{Z} \sim \mathcal{N}(\mathbf{0}, I)$ and define $\sigma_{\max} = \sqrt{\lambda_{\max}(\boldsymbol{\Sigma}_{\mathrm{G}})} = \max\{\sigma_S, \sigma_\pi\}$. Considering the function $f(\mathbf{Z}) = \left\|\boldsymbol{\Sigma}_{\mathrm{G}}^{1/2}\mathbf{Z}\right\|_2$, for any $\mathbf{Z}$ and $\mathbf{Z}'$,

$$|f(\mathbf{Z}) - f(\mathbf{Z}')| \le \left\|\boldsymbol{\Sigma}_{\mathrm{G}}^{1/2}(\mathbf{Z} - \mathbf{Z}')\right\|_2 \le \sigma_{\max}\|\mathbf{Z} - \mathbf{Z}'\|_2, \tag{18}$$

so f is $\sigma_{\max}$-Lipschitz. Moreover,

$$\mathbb{E}\|\boldsymbol{\xi}\|_2 \le \sqrt{\mathbb{E}\|\boldsymbol{\xi}\|_2^2} = \sqrt{\mathrm{tr}(\boldsymbol{\Sigma}_{\mathrm{G}})} = \sqrt{D_S\sigma_S^2 + D_\pi\sigma_\pi^2}. \tag{19}$$

By Gaussian concentration, for any $\eta \in (0,1)$,

$$\Pr\left(\|\boldsymbol{\xi}\|_2 \le \sqrt{D_S\sigma_S^2 + D_\pi\sigma_\pi^2} + \sigma_{\max}\sqrt{2\log(1/\eta)}\right) \ge 1-\eta. \tag{20}$$

We first establish a sufficient condition under which the ordered rival tuple remains unchanged. For candidate class c, let $r_1(c),\dots,r_J(c)$ denote the competing classes ordered by their clean similarity scores, and let $r_{J+1}(c)$ denote the highest-ranked class outside the retained rival set. Define the rival-ranking margin as

$$m_c^{\mathrm{riv}} = \min_{1\le j\le J}\left[a_{r_j(c)}(\mathbf{F}) - a_{r_{j+1}(c)}(\mathbf{F})\right]. \tag{21}$$

Assume that each similarity score is locally L_{sim}-Lipschitz,

$$|a_j(\mathbf{F}) - a_j(\mathbf{F}')| \le L_{\mathrm{sim}}\|\mathbf{F}-\mathbf{F}'\|_2. \tag{22}$$

Then, on the event in Equation (20), the ordered rival tuple is preserved whenever

$$m_c^{\mathrm{riv}} > 2L_{\mathrm{sim}}\left[\sqrt{D_S\sigma_S^2 + D_\pi\sigma_\pi^2} + \sigma_{\max}\sqrt{2\log(1/\eta)}\right]. \tag{23}$$

Under the local L-Lipschitz condition, for every candidate class c,

$$\left|\ell_c(\widetilde{\mathbf{F}}) - \ell_c(\mathbf{F})\right| \le L\|\boldsymbol{\xi}\|_2. \tag{24}$$

Let $c^\star$ denote the clean prediction. For any $c \ne c^\star$, we have

$$\begin{aligned}\ell_{c^\star}(\widetilde{\mathbf{F}}) - \ell_c(\widetilde{\mathbf{F}}) &= \ell_{c^\star}(\mathbf{F}) - \ell_c(\mathbf{F}) + \left[\ell_{c^\star}(\widetilde{\mathbf{F}}) - \ell_{c^\star}(\mathbf{F})\right] - \left[\ell_c(\widetilde{\mathbf{F}}) - \ell_c(\mathbf{F})\right] \\ &\ge \ell_{c^\star}(\mathbf{F}) - \ell_c(\mathbf{F}) - 2L\|\boldsymbol{\xi}\|_2 \ge \gamma - 2L\|\boldsymbol{\xi}\|_2.\end{aligned} \tag{25}$$

Therefore, with probability at least $1-\eta$,

$$\ell_{c^\star}(\widetilde{\mathbf{F}}) - \ell_c(\widetilde{\mathbf{F}}) \ge \gamma - 2L\left[\sqrt{D_S\sigma_S^2 + D_\pi\sigma_\pi^2} + \sigma_{\max}\sqrt{2\log(1/\eta)}\right]. \tag{26}$$

Hence, if $\gamma > 2L\left[\sqrt{D_S\sigma_S^2 + D_\pi\sigma_\pi^2} + \sigma_{\max}\sqrt{2\log(1/\eta)}\right]$, then $\ell_{c^\star}(\widetilde{\mathbf{F}}) > \ell_c(\widetilde{\mathbf{F}})$ for all $c \ne c^\star$. Thus, $\arg\max_c \ell_c(\widetilde{\mathbf{F}}) = c^\star$ with probability at least $1-\eta$, which proves Proposition 2.

D QICL Implementation

D.1 Implementation Details of QICL

All *QICL* variants use the pretrained *TabICL* classifier checkpoint tabic v1.1 as the representation backbone. To ensure the record-level sensitivity assumption used in our privacy analysis, *QICL* operates the *TabICL* front-end in a record-local mode: each context row is encoded independently, without the cross-row contextualization used by the original *TabICL* inference pipeline. Thus, replacing one context record affects only its corresponding row representation. The query is processed independently in the same manner. For each candidate class, the server retains the two highest-scoring rival classes. For binary tasks, the sole rival is duplicated to maintain a fixed-size candidate-aware input. For tasks containing more than ten classes, we follow the hierarchical class-extension strategy of *TabICL*, where each prediction node considers at most ten candidate classes and larger label spaces are handled recursively through class grouping.

The training tasks are generated synthetically using the publicly released *TabICL* prior-data generation pipeline, which combines MLP-based and tree-based structural causal models. Following the configuration used in our training pipeline, each synthetic task contains between 2 and 50 features and up to 10 classes. *QICL* is trained in two stages. Stage 1 performs clean training for 2,000 optimization steps without privacy perturbation. Stage 2 initializes from the Stage 1 checkpoint and performs an additional 500 steps of privacy-aware adaptation with the block-wise Gaussian perturbation enabled. For the block-wise Gaussian mechanism, we set $\zeta_\pi = 0.5$, which controls the relative allocation of the covariance-normalized sensitivity between the summary and prior blocks and does not represent a

direct split of ϵ. All reported results are evaluated over five random seeds, $\{41, 42, 43, 44, 45\}$. The simulator-based experiments were run on a single GPU server with one Intel Xeon Gold 6126 CPU, 192 GB ECC DDR4 RAM, and one NVIDIA Tesla V100 GPU.

Quantum predictor configuration. *QICL* uses an 8-qubit parameterized quantum circuit with three data-reuploading rounds. Each round applies data-dependent and trainable R_Y and R_Z rotations followed by a ring of CNOT entangling gates. The circuit is read out using eight single-qubit Z observables and eight nearest-neighbor ZZ observables. The candidate-aware vector branch uses a shared adapter with hidden width 16. Before fusion, the vector and scalar quantum coordinates are normalized separately using branch-wise RMS scaling with a target RMS of 0.5, and the resulting coordinates are combined by element-wise addition before being encoded into the quantum circuit.

D.2 Pairwise Statistical Significance Tests

To assess whether the performance differences between *QICL* and the competing methods are statistically significant, we conduct two-sided paired Wilcoxon signed-rank tests across datasets. Holm correction is applied separately within each privacy budget to account for multiple comparisons. Table 3 reports the resulting Holm-adjusted p-values. The results show that *QICL* achieves statistically significant performance differences from all competing methods under $\epsilon \in \{1, 2, 4\}$, with higher average accuracy in every comparison.

Table 3: Holm-adjusted p-values from two-sided paired Wilcoxon signed-rank tests comparing *QICL* with baselines

Baseline	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 4$
TabICL	1.20×10^{-25}	4.30×10^{-20}	2.49×10^{-23}
TabPFNv2	4.70×10^{-13}	3.55×10^{-11}	1.30×10^{-4}
TabICL-Mean	2.63×10^{-22}	1.00×10^{-16}	1.25×10^{-20}
TabICL-AttnPool	1.95×10^{-12}	3.55×10^{-11}	3.00×10^{-14}
TabICL-SetTrans	3.14×10^{-13}	2.03×10^{-12}	8.86×10^{-16}
CatBoost	1.53×10^{-12}	2.59×10^{-8}	2.67×10^{-4}
TabM	1.95×10^{-12}	6.51×10^{-9}	1.30×10^{-4}

D.3 Detailed Real-hardware Validation

We additionally deploy the simulation-trained model on the IBM `ibm_kingston` backend under $\epsilon = 1$ using 64 shots. Due to the limited available hardware runtime, we first excluded datasets with more than $30{,}000$ context samples and then processed the first 9 TALENT datasets in alphabetical order. The resulting average hardware accuracy was 0.500, compared with 0.751 for the corresponding simulation evaluations. The performance gap is mainly attributable to the simulation-to-hardware mismatch, including device-specific gate and decoherence errors, readout noise, and transpilation-induced circuit distortions that were not accounted for during simulation-based training. In addition, finite-shot estimation may introduce further statistical variability. Nevertheless, these results demonstrate the execution feasibility of *QICL* on real quantum hardware.

D.4 Architectural Sensitivity

D.4.1 The effect of Rivals

We examine the sensitivity of QICL to the number of retained rivals. We vary $J \in \{0, 1, 2, 3\}$ while keeping the remaining architecture and training settings fixed, where $J = 2$ corresponds to the default configuration. When $J > C - 1$, we retain all available rivals and duplicate them in descending similarity order until all J rival positions are filled. The case $J = 0$ removes rival information entirely and therefore reduces prediction to query–candidate modeling. Figure 8 shows that retaining rivals improves performance over the $J = 0$ case, confirming the benefit of candidate-aware comparison. Performance increases from $J = 0$ to $J = 2$, where the best result is obtained, and then decreases slightly at $J = 3$. This suggests that a small number of strongest rivals provides useful discriminative

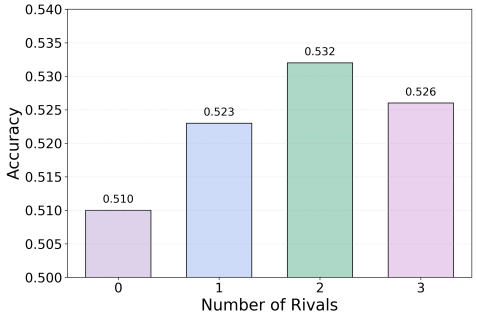


Figure 8: Effect of the number of retained rivals on QICL performance

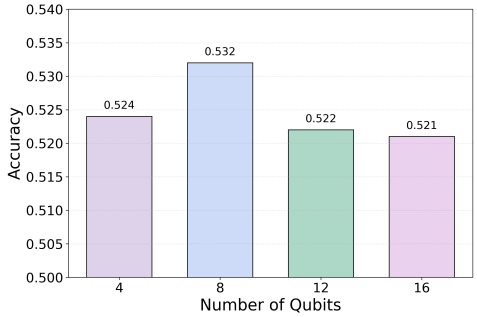


Figure 9: Effect of qubits on QICL performance

context, whereas adding more rivals introduces weaker and less informative comparisons under the same fixed prediction budget. Therefore, $J = 2$ provides the best trade-off.

D.4.2 The effect of qubit count

We vary the number of qubits as $n_q \in \{4, 8, 12, 16\}$ while keeping the structured interface and candidate-aware construction fixed. As shown in Figure 9, performance is non-monotonic with respect to the qubit count. Increasing the quantum predictor from 4 to 8 qubits improves accuracy, whereas further increasing the circuit to 12 or 16 qubits leads to lower performance. This result indicates that a larger quantum model does not necessarily provide greater predictive utility under a fixed structured interface. Once the task-relevant relational information has been compressed into a compact representation, additional qubits increase model capacity without introducing additional task information and may increase optimization difficulty.