Information-theoretic principles for LLM collaboration

Ishan Khare
Stanford University
iskhare@stanford.edu

Faculty Mentor: Prof. Christopher Ré

Abstract

Frontier cloud large language models (LLMs) like GPT-4O excel at complex reasoning but incur high inference costs and pose privacy risks when processing long inputs. Meanwhile, lightweight open-weight LLMs such as QWEN2.5 and LLAMA3 can run locally but lack the capacity for high-fidelity decision-making. This paper studies a hybrid edge–cloud setup where a local encoder compresses a long document context X into a shorter summary Z, which is then sent to a remote decoder to predict the final answer Y. We cast this $X \to Z \to Y$ protocol as a noisy information channel and introduce a mutual information (MI) estimator for quantifying how much task-relevant information is preserved. Using this estimator, we empirically analyze two QA benchmarks—LONGHEALTH and FINANCEBENCH—and observe that higher MI correlates with higher answer accuracy. We also report bit efficiency and empirical rate–distortion curves to compare local model families. Our results show that QWEN models produce more efficient summaries than LLAMA models, especially at smaller scales. Together, these findings offer a principled foundation for designing efficient and private edge–cloud LLM systems.

1 Introduction

Why split a question between *two* language models?

Frontier cloud models such as GPT-4O [23] or CLAUDE 4 [3] reliably perform code-base refactoring, multi-document legal analysis, and long-horizon clinical reasoning. Yet even a single pass over a one-million token repository costs $15 on the OpenAI `o1` endpoint. Meanwhile, compact (1-8B parameters) open-weight LLMs are now performant enough to run on local devices via projects such as LLAMA.CPP [18] and Ollama [21]. These lightweight model families like QWEN2.5 [24] and LLAMA3 [9] excel at many basic tasks like text manipulation but still fall short on the data-intensive reasoning demanded by modern workflows.

We therefore study a hybrid, *local–remote* architecture in which an on-device *local* LLM transforms the raw context $\mathbf{x}$ into a compressed message $\mathbf{z}$ that is sent over the network to a powerful *remote* LLM, which then returns the final answer $\mathbf{y}$. This pipeline is attractive for three intertwined reasons:

1. **Cost.** Once the hardware ships, local FLOPs are free whereas cloud inference is *token-metered*. Shrinking a 10 k-token context to a 300-token summary can reduce end-to-end cost by more than an order of magnitude.
2. **Privacy and regulation.** Sensitive inputs like electronic health records (HIPAA) [11], merger term sheets (SEC Rule 10b-5) [25], or chat transcripts (GDPR) [8] may be prohibited from leaving the device in raw form.

Stanford CS 191W Senior Project (Winter 2025).

3. **Latency, bandwidth, and energy.** Transmitting the entire raw context to the cloud can overwhelm uplink bandwidth and drain battery, especially on mobile devices; a local preprocessor reduces both latency and energy by minimizing data transfer.

Yet overly aggressive compression risks discarding the very bits required for high-fidelity answers. This tension raises a fundamental question:

> *How much information must the local model preserve for a specified task to ensure the remote model can still perform accurate reasoning?*

We cast the protocol $\mathbf{x} \to \mathbf{z} \to \mathbf{y}$ as a noisy information channel and analyze its efficiency through the lens of *mutual information (MI)* and *rate–distortion theory*. Under this view, the edge LLM is a learned source coder, and the cloud LLM is a stochastic decoder whose distortion is measured by downstream task error.

Key Contributions of this work.

- **Information–theoretic framework for edge–cloud LLMs.** We cast the local $\to$ remote summary channel as a rate–distortion problem providing a novel empirical analysis.
- **Practical mutual-information estimator.** We propose an in-context sampling estimator that needs $\mathcal{O}(n^2 m)$ remote log-probability calls—where n is the number of contexts/documents and m is the number of candidate summaries per context—and demonstrate that higher estimated MI *predicts* higher answer accuracy on the LongHealth [1] and FinanceBench [13] datasets.
- **Empirical distortion curves.** By evaluating a spectrum of local LLM model families and sizes under a fixed summarization prompt, we obtain multiple operating points in the $(\text{rate}, \text{distortion})$ plane—where mutual information $\mathcal{I}(X; Z)$ serves as a proxy for communication rate.

Together, these results supply principled guidelines for deciding *how much* an edge model should say to minimize cost without compromising correctness.

2 Preliminaries

2.1 Problem Setup and Notation

We study a two–stage LLM communication protocol illustrated in Figure 1. Let $X \in \mathcal{X}$ denote the **document context**, $Z \in \mathcal{Z}$ the **compressed summary**, and $Y \in \mathcal{Y}$ the **final answer**.

Here, $\mathcal{X}$ is the space of all possible document contexts—e.g., concatenations of user queries and source materials such as webpages, financial filings, or legal contracts. Each instance $x \in \mathcal{X}$ may contain tens of thousands of tokens. The encoder transforms x into a shorter summary $z \in \mathcal{Z}$ designed to capture task-relevant information.

The output space $\mathcal{Y}$ represents the set of all task outputs. For multiple-choice QA tasks, $\mathcal{Y}$ is a discrete set of answer options (e.g., {A, B, C, D, E}); for financial reasoning benchmarks like FinanceBench, $\mathcal{Y}$ may instead be a structured output space—e.g., numeric tuples like revenue and net income values extracted from 10-K filings. In both cases, the remote decoder consumes z and stochastically produces a prediction $\hat{y} \in \mathcal{Y}$ approximating the true label y.

The **local encoder** is modeled as a conditional distribution $p(Z \mid X)$ parameterized by a lightweight on–device LLM. It transforms a potentially long context X into a shorter discrete string Z (e.g., a task–specific summary). The **remote decoder** is another LLM realizing $p(Y \mid Z)$ that consumes the summary and stochastically generates the final answer. We treat $X \to Z \to Y$ as a Markov chain governed by the joint $p(x, z, y) = p(x)\, p(z \mid x)\, p(y \mid z)$.

2.2 Mutual Information and Rate–Distortion Perspective

The *mutual information* between the context and summary is

$$\mathcal{I}(X; Z) \;=\; \mathrm{D}_{\mathrm{KL}}\big(p(x, z) \big\| p(x)\, p(z)\big) \;=\; \mathbb{E}_{p(x,z)}\Big[\log \tfrac{p(z|x)}{p(z)}\Big]. \tag{1}$$

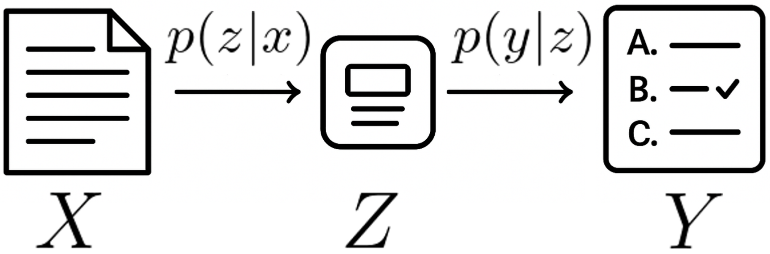

Figure 1: **The Simplified Local–Remote Collaboration Setting.** We consider a two-stage language model (LM) protocol where an on-device *local* encoder compresses the document context X into a question-specific summary Z via $p(z \mid x)$, which is then sent to a *remote* decoder that produces the final answer Y via $p(y \mid z)$. This setup enables efficient and private inference by offloading only essential information to the remote model.

High $\mathcal{I}$ implies that Z preserves many bits about X; low $\mathcal{I}$ indicates aggressive compression.

For a supervised task with loss $\ell(y, \hat{y})$, we quantify **distortion** as the expected task error $D = \mathbb{E}[\ell(Y, \hat{Y}(Z))]$, where $\hat{Y}$ denotes the stochastic prediction of the cloud LLM. Rate–distortion theory studies the Pareto frontier $D(R)$ achievable for compression budgets R measured in bits or, here, in units of mutual information.

2.3 Related Work

Edge–cloud LLM inference. A growing systems literature studies how to *split* a large language-model's computation between an on-device "edge" component and a cloud back-end to slash latency, cost, or privacy risk. Recent advances include EDGELLM's speculative-decoding engine for 7–13B paramater models on smartphones [28], model-based RL for adaptive layer splitting over wireless links [6], token-level routing that selectively consults a remote LLM only on "hard" tokens [26], resource-aware offloading frameworks for edge servers [10], and a recent survey distilling the design space of pruning, quantization, and split-inference techniques [27].

Summarize-then-answer pipelines. Prior to the advent of very large context windows (100k+ tokens), QA systems often adopted a "summarize, then reason" strategy. In multi-hop reading comprehension, the SUMMARIZE-THEN-ANSWER framework demonstrated that an abstractive explainer can *raise* end accuracy while shrinking context [12]. Follow-up work introduced candidate-conditioned summarization (SURE) for open-domain QA [14] and reward-model–guided context filtering that deletes irrelevant tokens before prompting a reader [15]. These studies, however, treat the compression length as a heuristic and do not quantify how much retained *information* suffices for correctness.

Information-theoretic analyses of summarization. Drawing on past theory, a recent work formalizes a summarizer rate–distortion function and empirically links higher mutual information $\mathcal{I}(X; Z)$ to better human preference scores [4]. Parallel work argues that mutual information offers a task-agnostic evaluation signal for summarization quality [7], and sliced-MI bounds have been proposed for classification from compressed views [20]. Unlike these text-only studies, we model the *two-step* channel $X \to Z \to Y$ and measure distortion as downstream task error.

Neural mutual information estimation. Estimating the mutual information in high dimension is notoriously challenging. The Mutual Information Neural Estimator (MINE) [5] and the InfoNCE bound [2] of Contrastive Predictive Coding [22] supply tractable lower bounds widely used in representation learning. Additionally, a NEURIPS-24 benchmark shows their bias–variance trade-offs on real text corpora [17]. We, however, use a different estimator for an *in-context* regime that needs only black-box log-probability access to the remote LLM.

Positioning of this work. Our study unifies the above threads: from edge–cloud split inference we inherit the system motivation, from summarize-then-answer pipelines we inherit the two-stage $X \to Z \to Y$ design, and from information-theoretic summarization we inherit MI-based evaluation.

Bridging the three, we present the first empirical rate–distortion curves for LLM Question-Answering datasets.

3 Mutual Information Estimator

3.1 Theoretical Derivation

As introduced earlier, we consider a two–stage language model (LM) communication protocol

$$X \xrightarrow{p(z|x)} Z \xrightarrow{p(y|z)} Y,$$

where a context X is first compressed by an *encoder* $p(z \mid x)$ into a summary Z, and then decoded by $p(y \mid z)$ to produce labels Y. By the Data Processing Inequality (DPI) [19], $I(X;Z) \geq I(Y;Z)$, so Z cannot introduce information about X beyond what X already contains.

Mutual information (MI). We define[1] the MI between X and Z as the Kullback–Leibler (KL) divergence [16] between the joint distribution $p(x,z)$ and the product of marginals $p(x)p(z)$:

$$\begin{aligned} \mathcal{I}(X;Z) &= D_{\mathrm{KL}}\big(p(x,z) \,\|\, p(x)p(z)\big) \\ &= \mathbb{E}_{x,z\sim p(x,z)}\Big[\log \tfrac{p(z|x)}{p(z)}\Big]. \end{aligned} \tag{2}$$

Rewriting the intractable term. Directly computing the marginal from the denominator $p(z) = \mathbb{E}_x[p(z \mid x)]$ is intractable. Because we can both sample $x \sim p(x)$ and evaluate the encoder $p(z \mid x)$, we rewrite Equation (2) as

$$\mathcal{I}(X;Z) = \mathbb{E}_{x,z\sim p(x,z)}\Big[\log \tfrac{p(z|x)}{\mathbb{E}_{x'}[p(z|x')]}\Big]. \tag{3}$$

Monte-Carlo estimator. Let $\{x_i\}_{i=1}^{n}$ be IID samples from $p(x)$ and $\{z_{ij}\}_{j=1}^{m} \sim p(z \mid x_i)$ be IID encoder draws for each x_i. A simple Monte-Carlo (MC) estimator of Equation (3) is

$$\hat{\mathcal{I}}(X;Z) \approx \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\Big[\log p(z_{ij} \mid x_i) - \log\Big(\tfrac{1}{n}\sum_{l=1}^{n} p(z_{ij} \mid x_l)\Big)\Big]. \tag{4}$$

3.2 Practical Estimation Pipeline

1. **Sample candidate summaries.** For each of the n document–query pairs, a local open-weight LLM produces m sampled summaries. The resulting $n \times m$ draft messages are cached for downstream analysis.
2. **Measure cross-likelihoods.** A powerful remote LLM is then asked to score the ground-truth answer under every *mismatched* context–summary pair, yielding the full grid of n^2m log-probabilities $\{\log p_\theta(z_{kj} \mid x_i)\}_{i,k\in[n],j\in[m]}$. Batched requests keep the entire sweep within the memory envelope of a single A100 GPU.
3. **Estimate mutual information.** Finally, the cached log-probabilities are plugged directly into the Monte-Carlo approximation from Equation (4) to calculate the final answer.

4 Experiments and Results

We evaluate our proposed mutual information (MI) estimator and the edge–cloud LLM collaboration protocol on two distinct benchmarks: LONGHEALTH, a medical QA dataset with long document contexts, and FINANCEBENCH, a structured financial reasoning task based on SEC filings. Across both, we explore the interplay between local model compression and remote model accuracy.

[1] All logarithms are natural.

4.1 Setup

For each dataset, we fix a downstream task (QA or extraction), a set of n document–query pairs, and m summaries per context produced by a local LLM. Each summary is sent to a cloud-based supervisor model, which predicts the final answer. We evaluate the accuracy of this answer and compute MI using the pipeline described in Section 3.2.

We vary the local encoder by choosing from several open-weight models (e.g., QWEN 1.5B/3B/7B, LLaMA 1B/3B/8B), while testing against several remote supervisors including GPT-4O and LLAMA3.1 405B Instruct.

4.2 Mutual Information and Token Counts

Tables 1 and 2 report MI estimates and average token counts across model families.

Table 1: LONGHEALTH: Mutual Information (MI) and Average Token Count

Model	MI ($\mathcal{I}(X;Z)$)	Avg. Token Count
QWEN 1.5B	2.40	247.3
QWEN 3B	2.76	190.3
QWEN 7B	2.99	97.8
LLAMA 1B	-4.18	555.3
LLAMA 3B	-0.69	461.1
LLAMA 8B	1.10	464.2

Table 2: FINANCEBENCH: Mutual Information (MI) and Average Token Count

Model	MI ($\mathcal{I}(X;Z)$)	Avg. Token Count
QWEN 1.5B	2.70	456.84
QWEN 3B	3.00	278.72
QWEN 7B	3.00	226.60
LLAMA 1B	2.37	427.84
LLAMA 3B	2.78	449.50
LLAMA 8B	2.82	493.20

Larger models such as QWEN 7B and LLAMA 8B typically achieve higher MI—indicating more task-relevant information is preserved during summarization. For instance, QWEN 7B yields MI ≈ 3.00 on FINANCEBENCH, while LLaMA 1B trails at ≈ 2.37.

These mutual information scaling curves versus model size are also displayed in Figures 2 and 3.

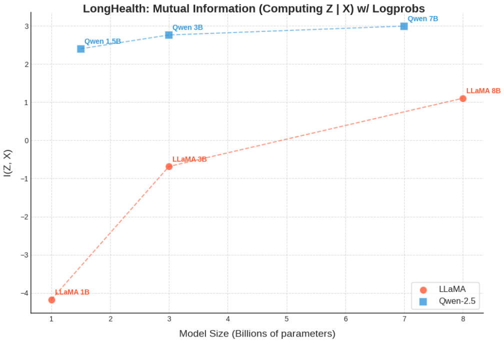

Figure 2: Mutual Information vs. Model Size on LONGHEALTH dataset.

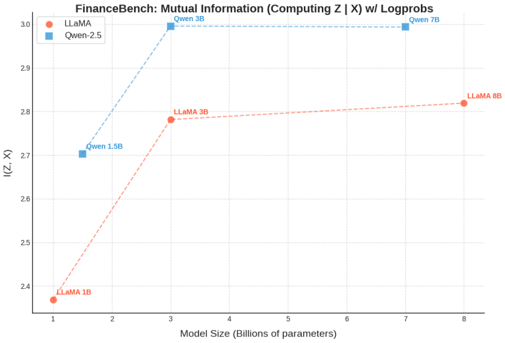

Figure 3: Mutual Information vs. Model Size on FINANCEBENCH dataset.

Moreover, we observe that the larger models generally tend to have a smaller average token count except for the exception of the LLAMA family on the FINANCEBENCH dataset.

These results are also displayed in Figures 4 and 5.

4.3 Bit Efficiency

To understand not just the absolute performance but the *efficiency* of local models, we evaluate **bit efficiency**, defined as the mutual information $\mathcal{I}(X;Z)$ normalized by the number of remote tokens consumed to produce the final answer.

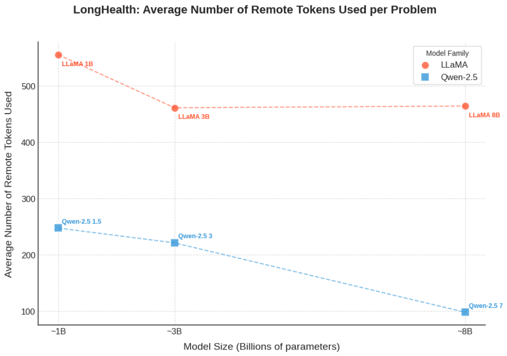

Figure 4: Average Token Count vs. Model Size on LONGHEALTH dataset.

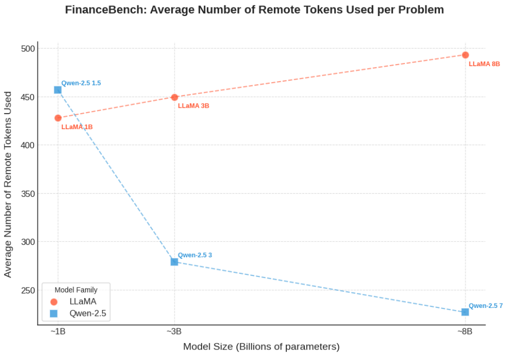

Figure 5: Average Token Count vs. Model Size on FINANCEBENCH dataset.

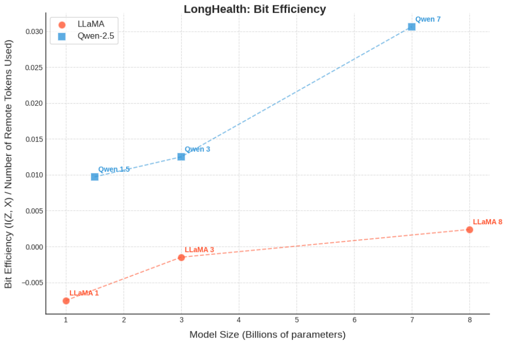

Figure 6: Bit efficiency on LONGHEALTH dataset.

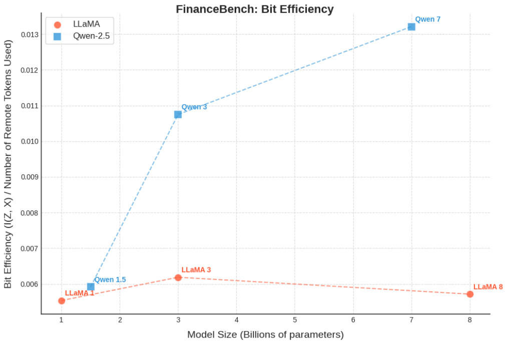

Figure 7: Bit efficiency on FINANCEBENCH dataset.

Figures 6 and 7 show the bit efficiency curves for both LONGHEALTH and FINANCEBENCH. In both datasets, the QWEN family consistently achieves higher MI per remote token compared to the LLAMA family, especially at smaller model sizes.

For instance, on LONGHEALTH, QWEN 1.5B already surpasses LLAMA 8B in bit efficiency, and QWEN 7B delivers more than 3× the efficiency of LLAMA 8B. Similarly, on FINANCEBENCH, QWEN 7B reaches a bit efficiency of over 0.013, compared to LLAMA 8B's peak of just under 0.006. These results reinforce the earlier conclusion: QWEN models not only retain more task-relevant information but do so with far fewer remote resources. In other words, the QWEN models are better at generating succinct and compressed summaries as compared to the LLAMA ones.

4.4 Accuracies by Local Model Family

We analyze the performance of each local model family—QWEN and LLAMA—in terms of the final answer accuracy achieved by the remote supervisor.

Qwen. Across both datasets, QWEN models exhibit strong performance and favorable scaling trends as presented in Table 3. On LONGHEALTH, QWEN 1.5B achieves accuracies in the 43–53% range depending on the supervisor, while QWEN 3B improves to 61–66%, and QWEN 7B reaches as high as 68% with GPT-4o. Similar behavior is observed on FINANCEBENCH, where QWEN 1.5B peaks at 50% accuracy with GPT-4o, while QWEN 3B and 7B both consistently reach 60% across several supervisors.

Llama. LLAMA models generally underperform Qwen models of comparable size. The LLAMA results are displayed in Table 4. On LONGHEALTH, LLAMA 1B struggles, with accuracies in the 21–33% range. LLAMA 3B improves to 54–60%, and LLAMA 8B reaches 63–70%, indicating better retention of task-relevant content as size increases. On FINANCEBENCH, LLAMA 1B remains

Table 3: Accuracy comparison of Qwen2.5 worker models on LongHealth and FinanceBench

Worker Model	Supervisor Model	LongHealth Accuracy	FinanceBench Accuracy
Qwen 1.5B	GPT-4o	0.53	0.50
	Llama3.1 405B Instruct	0.49	0.40
	Llama3.3 70B Instruct	0.46	0.36
	Llama3.1 8B Instruct	0.43	0.32
	Llama4 Maverick Instruct	0.51	0.44
Qwen 3B	GPT-4o	0.66	0.60
	Llama3.1 405B Instruct	0.63	0.60
	Llama3.3 70B Instruct	0.63	0.60
	Llama3.1 8B Instruct	0.61	0.52
	Llama4 Maverick Instruct	0.65	0.60
Qwen 7B	GPT-4o	0.68	0.60
	Llama3.1 405B Instruct	0.65	0.56
	Llama3.3 70B Instruct	0.65	0.56
	Llama3.1 8B Instruct	0.63	0.56
	Llama4 Maverick Instruct	0.66	0.52

Table 4: Accuracy comparison of Llama3 worker models on LongHealth and FinanceBench

Worker Model	Supervisor Model	LongHealth Accuracy	FinanceBench Accuracy
Llama 1B	GPT-4o	0.33	0.30
	Llama3.1 405B Instruct	0.28	0.25
	Llama3.3 70B Instruct	0.21	0.25
	Llama3.1 8B Instruct	0.23	0.20
	Llama4 Maverick Instruct	0.28	0.275
Llama 3B	GPT-4o	0.60	0.575
	Llama3.1 405B Instruct	0.58	0.50
	Llama3.3 70B Instruct	0.54	0.50
	Llama3.1 8B Instruct	0.54	0.575
	Llama4 Maverick Instruct	0.60	0.625
Llama 8B	GPT-4o	0.70	0.70
	Llama3.1 405B Instruct	0.68	0.70
	Llama3.3 70B Instruct	0.65	0.675
	Llama3.1 8B Instruct	0.63	0.625
	Llama4 Maverick Instruct	0.65	0.75

under 30%, while Llama 3B achieves moderate accuracy (50–63%) and Llama 8B attains strong performance (up to 75%) when paired with Llama4 Maverick.

Takeaway. These results confirm that larger local models yield more accurate remote predictions, consistent with their higher mutual information. While both families benefit from scaling, Qwen models are more efficient at retaining relevant content at smaller sizes, making them more suitable for edge deployment under tighter computational constraints.

4.5 Distortion Curves

To visualize the tradeoff between compression and task performance, we plot empirical **rate–distortion curves** where distortion is defined as $1 -$ accuracy and rate is measured via mutual information $\mathcal{I}(X; Z)$. These curves reflect how much information the local model must retain to enable the remote model to perform accurate reasoning.

Distortion curves for LongHealth are given in Figures 8 and 9 whereas the ones for FinanceBench are presented in Figures 10 and 11.

Across both datasets, we observe that Llama-based local models produce cleaner, more convex rate–distortion curves. In contrast, Qwen-based workers show irregular distortion patterns.

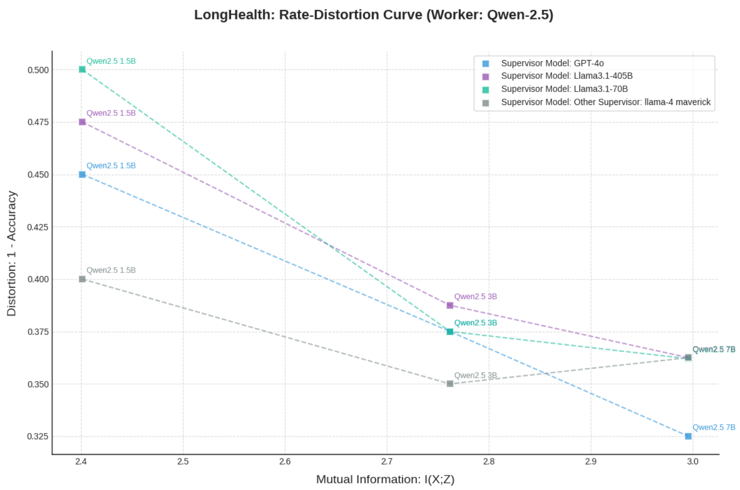

Figure 8: Rate-distortion curves on LONGHEALTH with QWEN workers.

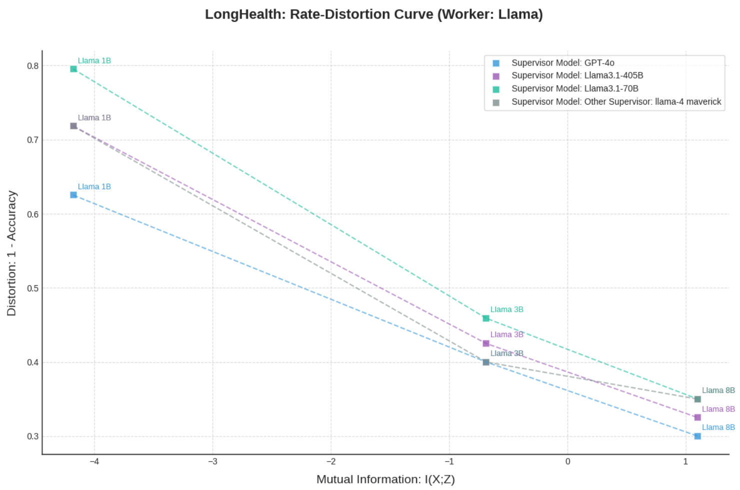

Figure 9: Rate-distortion curves on LONGHEALTH with LLAMA workers.

Across both datasets, we observe that LLAMA-based local models yield smoother and more convex rate–distortion curves, especially in the mid-to-large model sizes. This suggests that LLAMA summaries degrade gracefully under compression. In contrast, QWEN-based workers often exhibit irregular or jagged curves, indicating sensitivity to architectural or optimization differences across scales. Nevertheless, QWEN models still outperform LLAMA in terms of absolute distortion at a given MI level, underscoring their overall efficiency despite the noisier scaling behavior.

4.6 Summary

Our experimental analysis reveals several key insights about the behavior of local–remote LLM systems under information-theoretic constraints:

- **Mutual information tracks performance.** Higher mutual information $\mathcal{I}(X;Z)$ is consistently predictive of better downstream accuracy. Larger local models preserve more task-relevant content, as reflected in both MI estimates and accuracy metrics.
- **Qwen models are more efficient.** Across both LONGHEALTH and FINANCEBENCH, QWEN models consistently achieve higher accuracy and bit efficiency than their LLAMA counterparts. They are especially performant at smaller model sizes, making them attractive for edge deployment.

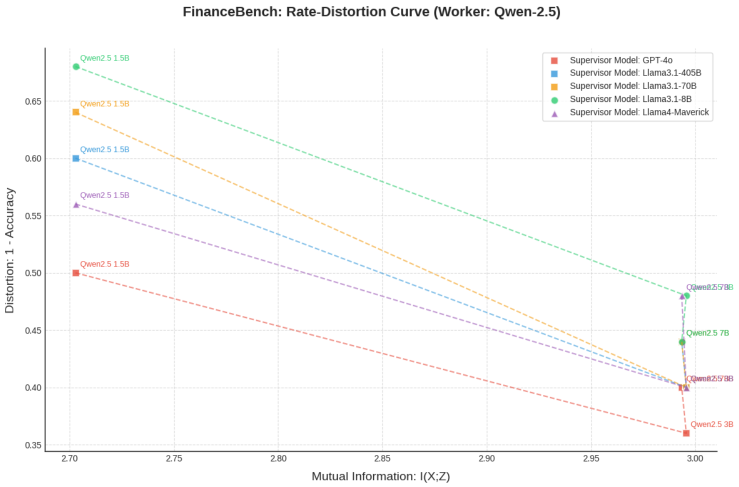

Figure 10: Rate-distortion curves on FINANCEBENCH with QWEN workers.

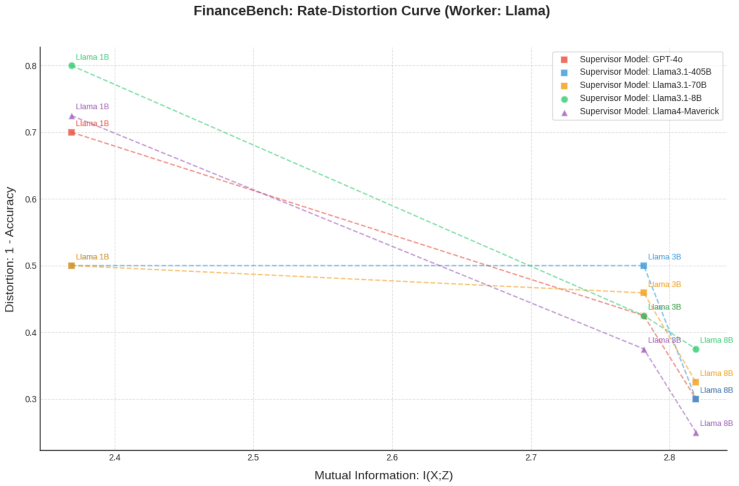

Figure 11: Rate-distortion curves on FINANCEBENCH with LLAMA workers.

- **Bit efficiency distinguishes model quality.** By normalizing MI by the number of remote tokens used, we expose a clearer measure of summarization efficiency. QWEN 7B achieves over 3× the bit efficiency of LLAMA 8B, suggesting that it produces more compact yet informative summaries.
- **Rate–distortion curves characterize tradeoffs.** While LLAMA models exhibit cleaner and more convex distortion curves—indicative of stable performance degradation—QWEN models offer better absolute accuracy–compression tradeoffs, even if their scaling behavior is noisier.
- **Practical guidance for system designers.** These findings supply principled tools—mutual information estimation, bit efficiency, and distortion curves—for choosing local LLMs that balance cost, compression, and answer fidelity in edge–cloud architectures.

5 Conclusion and Future Work

This work proposes an information-theoretic framework for studying edge–cloud collaboration in language models, where a local encoder compresses a long input into a summary that a powerful remote model uses to generate a final answer. We introduce a practical estimator for mutual information $\mathcal{I}(X;Z)$ that requires only black-box access to remote log-likelihoods, and shows empirically that higher MI correlates with lower task distortion. Through extensive experiments on LongHealth and FinanceBench, we demonstrated that MI, bit efficiency, and distortion curves serve as effective tools for evaluating and selecting local models.

Limitations. Our current MI estimator requires $O(n^2m)$ remote log-probability calls for n document contexts and m summaries per context. While this is tractable for small-scale evaluations, it becomes costly at scale.

Next Steps. To reduce runtime and generalize across datasets, we propose training a small neural network to approximate mutual information directly. Specifically:

- **Estimator:** Learn a contrastive objective (e.g., InfoNCE [22]) that distinguishes true document–summary pairs from mismatched ones.
- **Efficiency:** This learned estimator offers a fast lower-bound proxy for MI, avoiding the need for expensive n^2m evaluations.
- **Evaluation:** We will also plot *perplexity vs. accuracy* curves for each local–remote model pair to better understand the tradeoffs between confidence and correctness.

Overall, our framework supplies a principled foundation for designing summarization-aware systems that balance compression and fidelity. We believe future work combining neural MI estimation with large-scale deployment will further bridge the gap between efficient inference and high-quality reasoning in edge–cloud LLM architectures.

Acknowledgments

I would like to express my deepest gratitude to Prof. Chris Ré for his guidance and mentorship throughout this project. I am especially thankful to Avanika Narayan and Dan Biderman for their invaluable feedback, technical insights, and support during the research process. I also thank Prof. Scott Linderman, Thomas Breydo, and Shizhe He for their helpful conversations and constructive discussions, which shaped many of the ideas explored in this work.

Finally, I am grateful to the entire Hazy Research group for fostering a stimulating and collaborative research environment.

References

[1] Lisa Adams, Felix Busch, Tianyu Han, Jean-Baptiste Excoffier, Matthieu Ortala, Alexander Löser, Hugo JWL Aerts, Jakob Nikolas Kather, Daniel Truhn, and Keno Bressem. Longhealth: A question answering benchmark with long clinical documents. *arXiv preprint arXiv:2401.14490*, 2024.

[2] Laurence Aitchison and Stoil Ganev. Infonce is variational inference in a recognition parameterised model. *arXiv preprint arXiv:2107.02495*, 2021.

[3] Anthropic. System card: Claude opus 4 & claude sonnet 4. Technical report, Anthropic, May 2025. Accessed: 2025-05-29.

[4] Enes Arda and Aylin Yener. A rate-distortion framework for summarization. *arXiv preprint arXiv:2501.13100*, 2025.

[5] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

[6] Yuxuan Chen, Rongpeng Li, Xiaoxue Yu, Zhifeng Zhao, and Honggang Zhang. Adaptive layer splitting for wireless llm inference in edge computing: A model-based reinforcement learning approach. *arXiv preprint arXiv:2406.02616*, 2024.

[7] Maxime Darrin, Philippe Formont, Jackie Cheung, and Pablo Piantanida. COSMIC: Mutual information for task-agnostic summarization evaluation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12696–12717, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[8] General data protection regulation (gdpr), regulation (eu) 2016/679. `https://eur-lex.europa.eu/eli/reg/2016/679/oj`, 2016. Official Journal of the European Union, L119, 4 May 2016.

[9] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, et al. The llama 3 herd of models, 2024.

[10] Ying He, Jingcheng Fang, F Richard Yu, and Victor C Leung. Large language models (llms) inference offloading and resource allocation in cloud-edge computing: An active inference approach. *IEEE Transactions on Mobile Computing*, 2024.

[11] Health insurance portability and accountability act of 1996 (hipaa). `https://www.hhs.gov/hipaa/for-professionals/privacy/index.html`, 1996. U.S. Department of Health and Human Services.

[12] Naoya Inoue, Harsh Trivedi, Steven Sinha, Niranjan Balasubramanian, and Kentaro Inui. Summarize-then-answer: Generating concise explanations for multi-hop reading comprehension. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6064–6080, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[13] Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. Financebench: A new benchmark for financial question answering. *arXiv preprint arXiv:2311.11944*, 2023.

[14] Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha, and Jinwoo Shin. Sure: Summarizing retrievals using answer candidates for open-domain qa of llms. *arXiv preprint arXiv:2404.13081*, 2024.

[15] Sangryul Kim and James Thorne. Context filtering with reward modeling in question answering. *arXiv preprint arXiv:2412.11707*, 2024.

[16] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[17] Kyungeun Lee and Wonjong Rhee. A benchmark suite for evaluating neural mutual information estimators on unstructured datasets. *arXiv preprint arXiv:2410.10924*, 2024.

[18] llama.cpp contributors. llama.cpp: Efficient inference of llama and other llms on cpus and gpus. `https://github.com/ggml-org/llama.cpp`, 2023. Accessed: 2025-05-29.

[19] Kevin P Murphy. *Probabilistic machine learning: Advanced topics*. MIT press, 2023.

[20] Kimia Nadjahi, Kristjan Greenewald, Rickard Brüel Gabrielsson, and Justin Solomon. Slicing mutual information generalization bounds for neural networks. *arXiv preprint arXiv:2406.04047*, 2024.

[21] Ollama Team. Ollama: Run and build open language models locally. `https://ollama.com`, 2023. Accessed: 2025-05-29.

[22] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[23] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, et al. Gpt-4o system card, 2024.

[24] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, et al. Qwen2.5 technical report, 2025.

[25] Securities and exchange commission rule 10b-5. `https://www.law.cornell.edu/cfr/text/17/240.10b-5`, 1942. Adopted under the Securities Exchange Act of 1934.

[26] Jianshu She, Wenhao Zheng, Zhengzhong Liu, Hongyi Wang, Eric Xing, Huaxiu Yao, and Qirong Ho. Token level routing inference system for edge devices. *arXiv preprint arXiv:2504.07878*, 2025.

[27] Rui Wang, Zhiyong Gao, Liuyang Zhang, Shuaibing Yue, and Ziyi Gao. Empowering large language models to edge intelligence: A survey of edge efficient llms and techniques. *Computer Science Review*, 57:100755, 2025.

[28] Daliang Xu, Wangsong Yin, Hao Zhang, Xin Jin, Ying Zhang, Shiyun Wei, Mengwei Xu, and Xuanzhe Liu. Edgellm: Fast on-device llm inference with speculative decoding. *IEEE Transactions on Mobile Computing*, 2024.