

# 000 QUANTIZED CACHE-TO-CACHE: COMMUNICATION- 001 BUDGETED KV TRANSFER FOR HETEROGENEOUS 002 LLMS

006 **Anonymous authors**

007 Paper under double-blind review

## 011 ABSTRACT

013 We study communication-efficient transfer between heterogeneous large language  
014 models (LLMs) by quantizing Cache-to-Cache (C2C) KV-cache transfer. Our  
015 goal is to reduce bandwidth and memory while preserving accuracy. We present  
016 post-training quantization (INT8/INT4), cache-length reduction, and accuracy-  
017 versus-bytes curves for C2C, and introduce receiver-aware token selection and  
018 rate-distortion mixed-precision scheduling for sparse transfer. Empirically, quan-  
019 tization is nearly lossless, cache-length pruning reveals a strong front/back asym-  
020 metry, and our delta-selection improves accuracy at fixed token budgets while RD  
021 scheduling matches or improves accuracy at fixed byte budgets. We release a  
022 reproducible evaluation pipeline and analysis scripts.

## 024 1 INTRODUCTION

027 Large language models (LLMs) often communicate via text, which is slow and lossy. Cache-to-  
028 Cache (C2C) communicates via KV-cache projection and fusion, but does not address precision or  
029 bandwidth constraints. We ask: *How low can KV precision go before accuracy collapses, and can*  
030 *we recover performance under tight communication budgets?*

### 032 Contributions.

- 034 • We introduce a precision-aware C2C evaluation pipeline and quantify INT8/INT4 PTQ effects on  
035 C2C accuracy.
- 037 • We study cache-length reduction as a second budget axis and show that back-pruning consistently  
038 outperforms front-pruning.
- 039 • We report accuracy vs. communication-budget curves that jointly compare precision and cache  
040 length.
- 041 • We propose receiver-aware delta token selection and rate-distortion token×precision scheduling  
042 for sparse, budgeted C2C transfer.
- 044 • We provide a reproducible benchmarking setup and analysis scripts to support extensions to QAT,  
045 mixed precision, heterogeneity, and selective transfer.

## 047 2 BACKGROUND AND MOTIVATION

050 C2C projects sharer KV caches into receiver space and fuses them with learned gates, preserving rich  
051 semantics compared to text relay. However, KV caches are large: they scale with sequence length,  
052 KV heads, and head dimension. Quantization and cache-length reduction can shrink the communi-  
053 cation footprint while retaining accuracy. This work reframes C2C through a communication-budget  
lens.

054 **3 RELATED WORK**

056 **C2C.** Cache-to-Cache (C2C) enables direct semantic communication by projecting and fusing a  
 057 sharer model’s KV cache into a receiver’s KV cache with learnable gates, avoiding intermediate text  
 058 generation (Fu et al., 2025).

059 **KV communication across agents.** KVComm aligns KV caches across diverging prefixes us-  
 060 ing training-free offset correction with online anchors (Ye et al., 2025). Q-KVComm adds  
 061 adaptive layer-wise quantization, hybrid information extraction, and heterogeneous calibration  
 062 for compressed KV transfer (Kriuk & Ng, 2025). These works focus on multi-agent cache  
 063 reuse/compression; our work studies quantization and cache-length pruning within the C2C pro-  
 064 jector+fuser pipeline.

065 **Latent collaboration and cache alignment.** KV cache alignment learns a shared latent space with  
 066 adapters to align KV caches across models (Dery et al., 2026). LatentMAS enables latent-space  
 067 collaboration with shared working memory without extra training (Zou et al., 2025). Our approach  
 068 stays within C2C’s KV fusion but emphasizes communication budgets and precision/length trade-  
 069 offs.

070 **Token selection and KV compression.** Token-level KV selection and value-norm importance  
 071 improve long-context inference for a single model (ZipCache, TokenSelect, VATP) (Anonymous,  
 072 2024b; 2025; 2024a). We adopt the budget perspective for C2C rather than single-model KV com-  
 073 pression.

075 **4 METHOD**

076 **4.1 C2C RECAP**

077 Let the sharer model produce KV caches  $(K_\ell^S, V_\ell^S)$  and the receiver produce  $(K_\ell^R, V_\ell^R)$  at layer  $\ell$ .  
 078 C2C projects sharer KV into receiver space via  $\Pi_\ell^K, \Pi_\ell^V$  and fuses them through a learnable gate:

$$(K_\ell^{R'}, V_\ell^{R'}) = \mathcal{F}_\ell(K_\ell^R, V_\ell^R, \Pi_\ell^K(K_\ell^S), \Pi_\ell^V(V_\ell^S)).$$

079 This avoids intermediate text and transfers richer internal semantics.

080 **4.2 POST-TRAINING QUANTIZATION (PTQ)**

081 We quantize the KV caches using INT8 or INT4/NF4 with per-head scaling. We evaluate accuracy  
 082 and latency under fixed precision budgets. Our current implementation uses fake-quant (quantize  
 083 then dequantize) to model quantization noise without bit-packing.

084 **4.3 CACHE-LENGTH REDUCTION**

085 We prune KV tokens using a fixed ratio (e.g., 50%, 25%, 10%), reducing transmitted bytes further.  
 086 We evaluate front-pruning and back-pruning to diagnose which instruction tokens are most valuable  
 087 for cross-model transfer.

088 **4.4 SELECTIVE AND COMPRESSED CACHE TRANSFER (SPARSEC2C)**

089 As a main-conference extension, we select a sparse subset of token positions to transfer and fuse.  
 090 Let  $I \subset \{1, \dots, T\}$  be selected tokens and  $S_I$  the gather operator. We fuse only selected tokens and  
 091 scatter updates back:

$$\begin{aligned} (\tilde{K}_\ell^R, \tilde{V}_\ell^R) &= S_I^\top(K_\ell^R, V_\ell^R), \quad (\tilde{K}_\ell^S, \tilde{V}_\ell^S) = S_I^\top(K_\ell^S, V_\ell^S) \\ (\tilde{K}_\ell^{R'}, \tilde{V}_\ell^{R'}) &= \mathcal{F}_\ell(\tilde{K}_\ell^R, \tilde{V}_\ell^R, \Pi_\ell^K(\tilde{K}_\ell^S), \Pi_\ell^V(\tilde{V}_\ell^S)). \end{aligned}$$

092 We then scatter the update to the full cache. We use projector-aware token scoring by computing  
 093 value norms in receiver space (`proj_vnorm_topk`), tying selection to the cross-model mapping.

108 4.4.1 RECEIVER-AWARE DELTA SELECTION (M9)  
109110 For sparse transfer, the receiver already has a cache baseline, so sending large-but-redundant tokens  
111 wastes bandwidth. We score each token by its *marginal update* in receiver space:

112 
$$\Delta V_t^\ell = \widehat{V}_{:,:,t,:}^\ell - V_{:,:,t,:}^{R,\ell}, \quad u^\ell(t) = \mathbb{E}_{b,h} [\|\Delta V_{b,h,t}^\ell\|_2].$$
  
113

114 We select the top- $k$  tokens by  $u^\ell(t)$  under a token budget  $|I_\ell| \leq \lfloor pT \rfloor$ :

115 
$$I_\ell = \text{TopK}(u^\ell(t); \lfloor pT \rfloor).$$

116 This `delta_proj_vnorm_topk` score is projector-aware and redundancy-aware by construction.  
117118 4.4.2 RATE-DISTORTION TOKEN×PRECISION SCHEDULING (M10)  
119120 Under a fixed communication budget, we jointly select tokens and precisions. Each token  $t$  chooses  
121 an action  $a_t \in \{\text{drop, int4, int8}\}$  with rate  $r(a_t)$  (bits/element). We minimize distortion under a byte  
122 budget:

123 
$$\min_{a_t} \sum_t D_t(a_t) \quad \text{s.t.} \quad \sum_t r(a_t) \leq R_{\text{budget}},$$
  
124

125 with  $D_t(\text{drop}) = \|\widehat{V}_t - V_t^R\|_2^2$  and  $D_t(\text{intb}) = \|\widehat{V}_t - \widehat{V}_t^{(\text{intb})}\|_2^2$ . We implement a deterministic greedy  
126 allocator (RD-Greedy) that assigns INT8 to highest-utility tokens, then INT4, then drop, until the  
127 byte budget is met.128 4.5 COMMUNICATION-BUDGET CURVES  
129130 We report accuracy as a function of transmitted bytes, enabling fair comparison under equal com-  
131 munication constraints. For a sequence of length  $T$ , the approximate bytes are  
132

133 
$$\text{bytes} \approx T \cdot p \cdot 2 \cdot L \cdot H_{kv} \cdot d_h \cdot b/8,$$

134 where  $p$  is the retained cache proportion,  $L$  the number of layers,  $H_{kv}$  KV heads,  $d_h$  head dim, and  
135  $b$  bits per element. We use this accounting for consistent budget curves.  
136137 5 EXPERIMENTS  
138139 5.1 SETUP  
140141 We evaluate on OpenBookQA and ARC-C with a Qwen3-0.6B receiver and Qwen2.5-0.5B sharer.  
142 We follow the C2C eval protocol: temperature 0, max\_new\_tokens 64, no CoT, unified chat template.  
143 All models are frozen; only the projector is trained when QAT is enabled. The OpenBookQA test  
144 split has 500 samples and ARC-C has 1150 samples.145 5.2 MAIN RESULTS  
146147 All results below are full runs. PTQ is effectively lossless relative to FP16, and cache pruning shows  
148 a strong front/back asymmetry.  
149150 151 **Table 1: Baseline vs. PTQ (full-cache, %).**

Setting	OpenBookQA	ARC-C
FP16 baseline	52.8	55.1
INT8 PTQ	52.8	55.0
INT4 PTQ	52.6	55.4

152 5.3 COMMUNICATION-BUDGET CURVE  
153154 Figure 1 and Figure 2 report accuracy versus effective transmitted bytes. Each point is annotated  
155 with the retained cache proportion. These curves provide a single, comparable view across precision  
156 (FP16/INT8/INT4) and cache-length reduction.  
157

162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215

Table 2: OpenBookQA accuracy (%), 500 samples) for cache-length pruning (INT8).

Order mode	75%	50%	25%	10%
Front	44.6	43.0	38.8	38.6
Back	52.2	52.0	50.8	49.2

Table 3: ARC-C accuracy (%), 1150 samples) for cache-length pruning (INT8).

Order mode	75%	50%	25%	10%
Front	40.2	46.3	38.3	40.7
Back	55.7	57.2	56.2	53.7

#### 5.4 ORDER-MODE ABLATION

Across all cache lengths, **back-pruning** (keeping later instruction tokens) consistently outperforms **front-pruning**. At 50% cache length, for example, back-pruning retains near-baseline accuracy while front-pruning degrades sharply. This suggests late instruction tokens carry higher utility for cross-model KV fusion, a useful design signal for future selective transfer methods.

#### 5.5 RECEIVER-AWARE SELECTION AND RD SCHEDULING

We compare M9 delta selection against value-norm baselines at a fixed token budget ( $p = 0.25$ , INT8, prompt-only). Delta selection improves accuracy over both vnorm and proj\_vnorm heuristics on both datasets (Table 4).

Table 4: M9 selection at  $p = 0.25$  (INT8, base pair).

Token selector	OpenBookQA	ARC-C
vnorm.topk	47.0	49.6
proj_vnorm.topk	46.2	52.6
delta_proj_vnorm.topk	<b>49.8</b>	<b>54.8</b>

RD scheduling provides an additional budget axis. At 1/16 and 1/8 byte budgets, mixed-precision RD (drop+int4+int8) matches or slightly improves over drop+int8 (Table 5), indicating that token-level precision allocation is competitive under tight budgets.

#### 5.6 HETEROGENEOUS PAIR SPOT CHECK

On a heterogeneous pair (Qwen3→Llama3.2, alignment on), M9 delta selection and M10 RD scheduling remain viable (Table 6).

#### 5.7 ADDITIONAL EXTENSIONS

Mixed precision (INT8 with FP16 in the last layers) remains near baseline across last-2/last-4/last-8 schedules. Projector-only QAT (INT8) currently degrades accuracy (39.6/40.2), indicating that longer training or recipe tuning is needed. An alignment-only ablation (same model pair, alignment enabled) reduces accuracy, suggesting alignment should be reserved for heterogeneous pairs.

## 6 DISCUSSION

Quantized C2C provides large bandwidth reductions with limited accuracy drop. Receiver-aware delta selection consistently improves low-budget accuracy, and RD-C2C achieves near-baseline performance at moderate byte budgets. These results suggest that redundancy-aware token selection is a key lever for cross-model cache transfer. A main-conference path includes QAT recovery, mixed-precision schedules, heterogeneous model pairs, and system-level latency measurements.

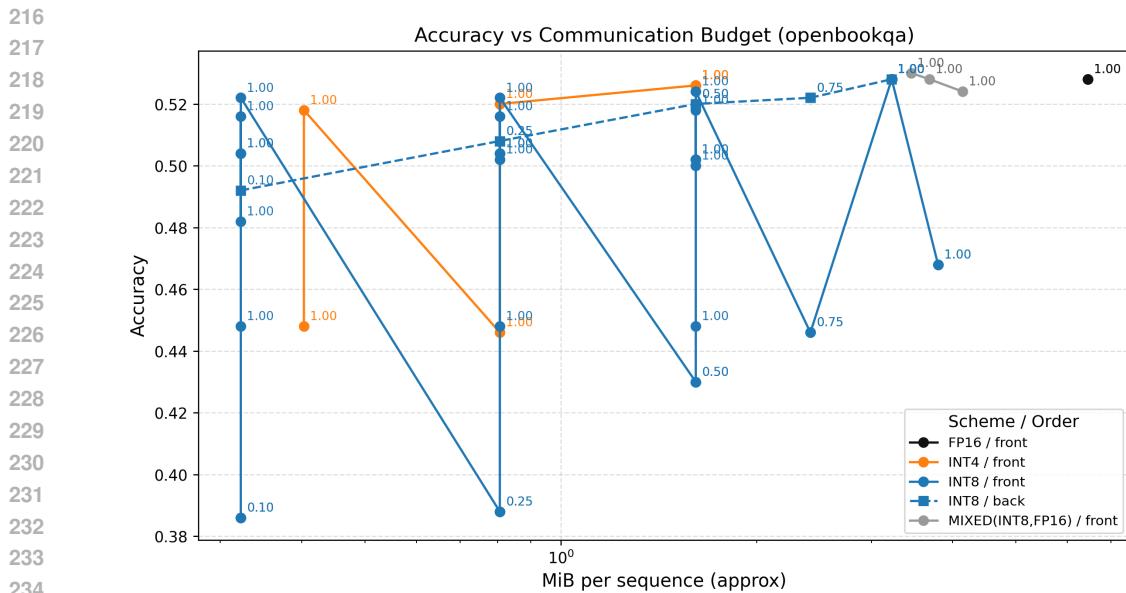


Figure 1: Accuracy vs. communication budget (OpenBookQA).

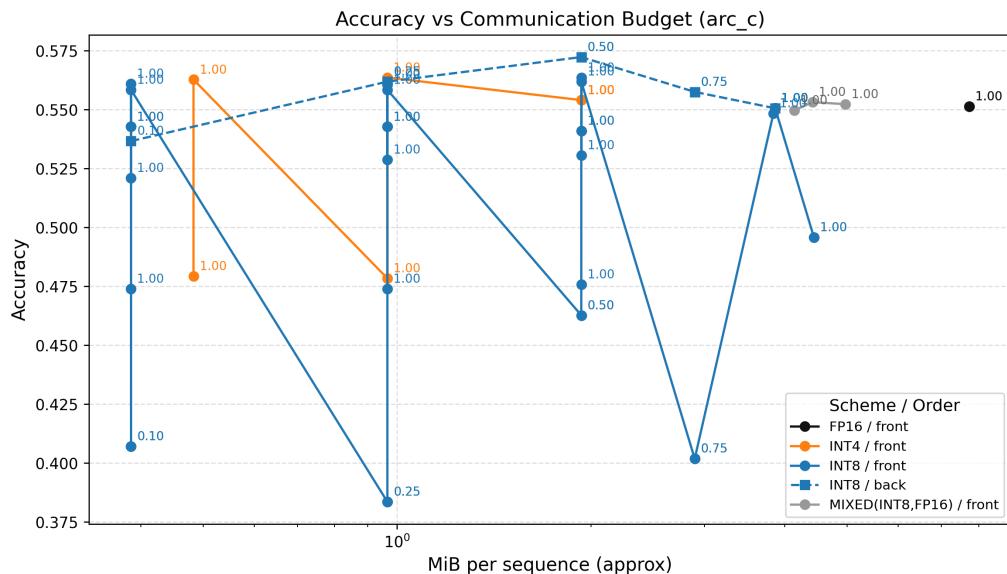


Figure 2: Accuracy vs. communication budget (ARC-C).

## 7 LIMITATIONS

Our results focus on a single base pair and two primary datasets; heterogeneity is evaluated via a single spot check. We do not yet report end-to-end latency or FLOP measurements for the fuser (timing-sync runs are pending). GSM8K accuracy is low without chain-of-thought prompting. These limitations will be addressed in the final main-conference revision.

270

271

Table 5: M10 RD ablation (base pair).

272

273

Setting	OpenBookQA	ARC-C
RD 1/16 (drop+int8)	52.6	57.2
RD 1/16 (drop+int4+int8)	<b>53.4</b>	57.0
RD 1/8 (drop+int8)	52.4	54.9
RD 1/8 (drop+int4+int8)	52.4	54.9

274

275

276

277

278

279

Table 6: Hetero spot check (alignment on).

280

281

282

283

Setting	OpenBookQA	ARC-C
M9 delta ( $p=0.25$ )	40.0	44.6
M10 RD (1/8)	40.4	46.0

284

285

## 8 BROADER IMPACT

286

287

Communication-efficient multi-LLM systems can reduce compute and latency, but they may also enable higher-throughput deployment of models. We emphasize reproducible evaluation, careful reporting of accuracy/latency tradeoffs, and responsible deployment in sensitive domains.

288

289

290

291

292

## 9 CONCLUSION

293

294

We introduce precision-aware C2C and report accuracy vs. bytes curves. This establishes a communication-budget perspective for cross-model KV transfer and opens the door to low-latency, low-bandwidth agent collaboration.

295

296

297

298

## ACKNOWLEDGMENTS

299

300

Placeholder.

301

302

## REFERENCES

303

Anonymous. Attention score is not all you need for token importance indicator in kv cache reduction: Value also matters. 2024a.

304

Anonymous. Zipcache: Accurate and efficient kv cache compression for llm inference. 2024b.

305

Anonymous. Tokenselect: Efficient long-context inference with token-level kv cache selection. 2025.

306

Lucio M. Dery, Zohar Yahav, Henry Prior, Qixuan Feng, Jiajun Shen, and Arthur Szlam. Latent space communication via k-v cache alignment. 2026.

307

Tianyu Fu, Zihan Min, Hanling Zhang, Jichao Yan, Guohao Dai, Wanli Ouyang, and Yu Wang. Cache-to-cache: Direct semantic communication between large language models. 2025.

308

Boris Kriuk and Logic Ng. Q-kvcomm: Efficient multi-agent communication via adaptive kv cache compression. 2025.

309

Hancheng Ye, Zhengqi Gao, Mingyuan Ma, Qinsi Wang, Yuzhe Fu, Ming-Yu Chung, Yueqian Lin, Zhijian Liu, Jianyi Zhang, Danyang Zhuo, and Yiran Chen. Kvcomm: Online cross-context kv-cache communication for efficient llm-based multi-agent systems. 2025.

310

311

312

Jiaru Zou, Xiyuan Yang, Ruizhong Qiu, Gaotang Li, Katherine Tieu, Pan Lu, Ke Shen, Hanghang Tong, Yejin Choi, Jingrui He, James Zou, Mengdi Wang, and Ling Yang. Latent collaboration in multi-agent systems. 2025.

324  
 325  
 326  
 327  
 328  
 329  
 330  
 331  
 332  
 333  
 334  
 335  
 336  
 337  
 338  
 339  
 340  
 341  
 342  
 343  
 344  
 345  
 346  
 347

Table 7: Additional extension results (% accuracy).

Setting	OpenBookQA	ARC-C
Mixed precision (INT8 + last-2 FP16)	53.0	55.0
Mixed precision (INT8 + last-4 FP16)	52.8	55.3
Mixed precision (INT8 + last-8 FP16)	52.4	55.2
QAT (projector-only, INT8)	39.6	40.2
Alignment ablation (same pair)	46.8	49.6
Hetero pair (Qwen3→Llama3.2, align on)	44.2	47.8

356  
 357  
 358  
 359  
 360  
 361  
 362  
 363  
 364  
 365  
 366  
 367  
 368  
 369  
 370  
 371  
 372  
 373  
 374  
 375  
 376  
 377