

000 QUANTIZED CACHE-TO-CACHE: COMMUNICATION- 001 BUDGETED KV TRANSFER FOR HETEROGENEOUS 002 LLMS 003

004 **Anonymous authors**
 005 Paper under double-blind review
 006

010 ABSTRACT 011

012 We study communication-efficient transfer between heterogeneous large language
 013 models (LLMs) by quantizing Cache-to-Cache (C2C) KV-cache transfer. Our goal
 014 is to reduce bandwidth and memory while preserving accuracy. We present post-
 015 training quantization (INT8/INT4), cache-length reduction, and accuracy-versus-
 016 bytes curves for a heterogeneous model pair. Empirically, quantization is nearly
 017 lossless, while cache-length pruning reveals a strong front/back asymmetry that is
 018 critical for budgeted transfer. We release a reproducible evaluation pipeline and
 019 analysis scripts, and we outline a main-conference path toward sparse, projector-
 020 aware token selection and mixed precision.
 021

023 1 INTRODUCTION 024

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

029 Contributions.

- 030 • We introduce a precision-aware C2C evaluation pipeline and quantify INT8/INT4 PTQ effects on
 031 C2C accuracy.
- 032 • We study cache-length reduction as a second budget axis and show that back-pruning consistently
 033 outperforms front-pruning.
- 034 • We report accuracy vs. communication-budget curves that jointly compare precision and cache
 035 length.
- 036 • We provide a reproducible benchmarking setup and analysis scripts to support extensions to QAT,
 037 mixed precision, heterogeneity, and selective transfer.

041 2 BACKGROUND AND MOTIVATION 042

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

049 3 RELATED WORK 050

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

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

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

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

4 METHOD

4.1 C2C RECAP

Let the sharer model produce KV caches (K_ℓ^S, V_ℓ^S) and the receiver produce (K_ℓ^R, V_ℓ^R) at layer ℓ . C2C projects sharer KV into receiver space via Π_ℓ^K, Π_ℓ^V and fuses them through a learnable gate:

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

This avoids intermediate text and transfers richer internal semantics.

4.2 POST-TRAINING QUANTIZATION (PTQ)

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

4.3 CACHE-LENGTH REDUCTION

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

4.4 SELECTIVE AND COMPRESSED CACHE TRANSFER (SPARSEC2C)

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

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

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

4.5 COMMUNICATION-BUDGET CURVES

We report accuracy as a function of transmitted bytes, enabling fair comparison under equal communication constraints. For a sequence of length T , the approximate bytes are

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

where p is the retained cache proportion, L the number of layers, H_{kv} KV heads, d_h head dim, and b bits per element. We use this accounting for consistent budget curves.

108

5 EXPERIMENTS

109

5.1 SETUP

112 We evaluate on OpenBookQA and ARC-C with a Qwen3-0.6B receiver and Qwen2.5-0.5B sharer.
 113 We follow the C2C eval protocol: temperature 0, max_new_tokens 64, no CoT, unified chat template.
 114 All models are frozen; only the projector is trained when QAT is enabled. The OpenBookQA test
 115 split has 500 samples and ARC-C has 1150 samples.

116

5.2 MAIN RESULTS

117 All results below are full runs. PTQ is effectively lossless relative to FP16, and cache pruning shows
 118 a strong front/back asymmetry.

122 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

130 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

137 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

143

5.3 COMMUNICATION-BUDGET CURVE

145 Figure 1 and Figure 2 report accuracy versus effective transmitted bytes. Each point is annotated
 146 with the retained cache proportion. These curves provide a single, comparable view across precision
 147 (FP16/INT8/INT4) and cache-length reduction.

148

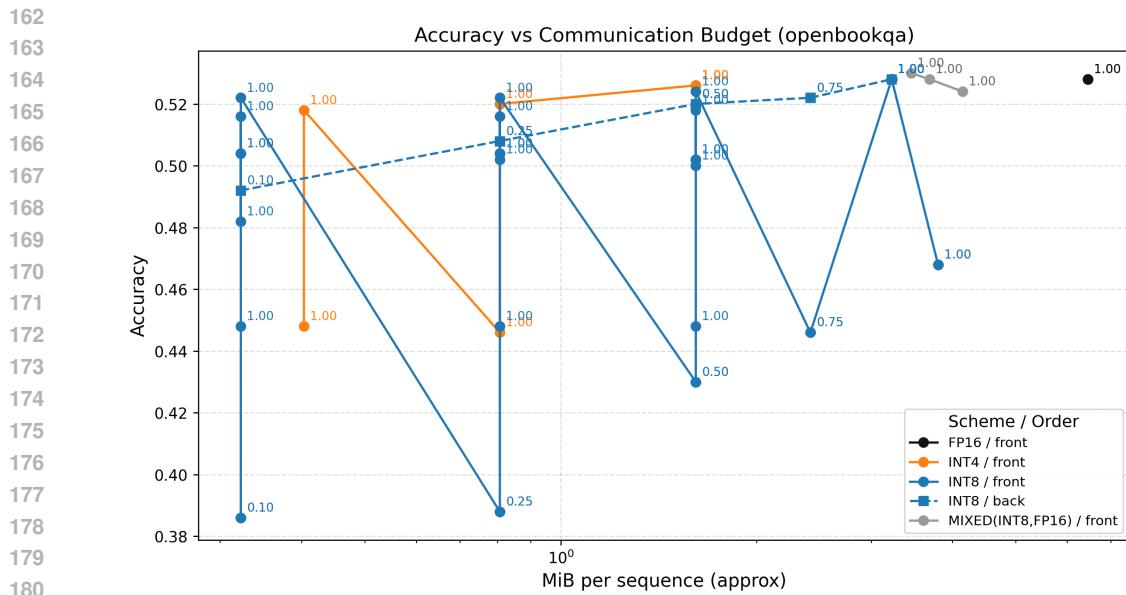
5.4 ORDER-MODE ABLATION

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

155

5.5 MAIN-CONFERENCE EXTENSIONS (PRELIMINARY)

157 We report early results for main-conference extensions. Mixed precision (INT8 with FP16 in the last
 158 layers) remains near baseline across last-2/last-4/last-8 schedules. Projector-only QAT (INT8) cur-
 159 rently degrades accuracy (39.6/40.2), indicating that longer training or recipe tuning is needed. An
 160 alignment-only ablation (same model pair, alignment enabled) reduces accuracy, suggesting align-
 161 ment should be reserved for heterogeneous pairs. For a heterogeneous pair (Qwen3→Llama3.2),
 alignment-on yields 44.2/47.8; alignment-off was unstable and is omitted. For SparseC2C (token se-



216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

Table 4: Preliminary 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

Table 5: SparseC2C token selection at p=0.5 (prompt-only, sparse fuse).

Setting	OpenBookQA	ARC-C
INT8 front	44.8	47.6
INT8 random	50.0	53.0
INT8 proj_vnorm_topk	50.2	54.1
INT8 knorm_topk	51.8	56.3
INT8 vnorm_topk	52.4	56.2
INT4 front	44.6	47.8
INT4 vnorm_topk	52.0	56.3

A main-conference path includes QAT recovery, mixed-precision schedules, heterogeneous model pairs, and SparseC2C token selection.

7 LIMITATIONS

Our results currently focus on a single model pair and two datasets. We do not yet report end-to-end latency or FLOP measurements for the fuser, and SparseC2C remains an ongoing extension. These limitations will be addressed in the main-conference track.

8 BROADER IMPACT

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.

9 CONCLUSION

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.

ACKNOWLEDGMENTS

Placeholder.

REFERENCES

- Anonymous. Attention score is not all you need for token importance indicator in kv cache reduction: Value also matters. 2024a.
- Anonymous. Zipcache: Accurate and efficient kv cache compression for llm inference. 2024b.

- 270 Anonymous. Tokenselect: Efficient long-context inference with token-level kv cache selection.
271 2025.
- 272 Lucio M. Dery, Zohar Yahav, Henry Prior, Qixuan Feng, Jiajun Shen, and Arthur Szlam. Latent
273 space communication via k-v cache alignment. 2026.
- 274 Tianyu Fu, Zihan Min, Hanling Zhang, Jichao Yan, Guohao Dai, Wanli Ouyang, and Yu Wang.
275 Cache-to-cache: Direct semantic communication between large language models. 2025.
- 276 Boris Kriuk and Logic Ng. Q-kvcomm: Efficient multi-agent communication via adaptive kv cache
277 compression. 2025.
- 278 Hancheng Ye, Zhengqi Gao, Mingyuan Ma, Qinsi Wang, Yuzhe Fu, Ming-Yu Chung, Yueqian Lin,
279 Zhijian Liu, Jianyi Zhang, Danyang Zhuo, and Yiran Chen. Kvcomm: Online cross-context kv-
280 cache communication for efficient llm-based multi-agent systems. 2025.
- 281 Jiaru Zou, Xiyuan Yang, Ruizhong Qiu, Gaotang Li, Katherine Tieu, Pan Lu, Ke Shen, Hanghang
282 Tong, Yejin Choi, Jingrui He, James Zou, Mengdi Wang, and Ling Yang. Latent collaboration in
283 multi-agent systems. 2025.
- 284
- 285
- 286
- 287
- 288
- 289
- 290
- 291
- 292
- 293
- 294
- 295
- 296
- 297
- 298
- 299
- 300
- 301
- 302
- 303
- 304
- 305
- 306
- 307
- 308
- 309
- 310
- 311
- 312
- 313
- 314
- 315
- 316
- 317
- 318
- 319
- 320
- 321
- 322
- 323