UNICOM: UNIFYING EXPLICIT AND IMPLICIT COMMUNICATION FOR LLM AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Effective collaboration among large language model (LLM) agents hinges on the fidelity of their communication protocols, which currently diverge into *explicit communication* in *textual space* and *implicit communication* in *latent space*. This dichotomy inherently imposes a fundamental tension between *interpretability* and *expressiveness*, wherein textual communication sacrifices high-dimensional reasoning nuances for human readability, whereas latent communication preserves rich semantics at the cost of rendering the underlying reasoning logic opaque and unauditable. To reconcile this tension, we present **UniCom**, a unified agentic communication protocol that integrates the complementary strengths of textual and latent modalities within a single coherent paradigm. At its core, UniCom features a lightweight dual-stream adapter, in which a shared encoder first maps the hidden states of heterogeneous senders into a unified communication space, and two parallel decoders then disentangle the encoded representations into discrete textual tokens and continuous latent semantics. In this way, UniCom naturally embodies the *communication duality*, delivering *an interpretable preview* for human inspection alongside *an expressive payload* for receiver inference. Experiments on five standard benchmarks under two heterogeneous agent scales show that UniCom consistently outperforms both text-only and latent-only communication baselines, revealing its potential to unify agent-to-human and agent-to-agent communication.[1]

1 INTRODUCTION

The rapid advancement of Large Language Models (LLMs) (Brown et al., 2020; Chowdhery et al., 2023; Dubey et al., 2024; Yang et al., 2025; Zhao et al., 2023) has ushered in a new era of agentic collaboration (Wang et al., 2023; Xi et al., 2023), wherein networks of autonomous agents interact to resolve complex knowledge reasoning, software engineering, and decision-making tasks (Li et al., 2025; Hong et al., 2024; Yao et al., 2023b). Within such agentic frameworks, swarm intelligence is intrinsically bounded by the fidelity of inter-agent communication (Fu et al., 2025). Prevailing works gravitate towards either *explicit communication* in *textual space* (Du et al., 2024; Zhang et al., 2025) or *implicit communication* in *latent space* (Fu et al., 2025; Zheng et al., 2025), yet this dichotomy inherently enforces a fundamental trade-off between *interpretability* and *expressiveness*.

Explicit textual communication serves as the de facto standard for the vast majority of multi-agent frameworks. Anchored in discrete text sequences, it possesses intrinsic interpretability and cross-model generalizability, offering a transparent view of the underlying interaction logic. Yet, natural language fundamentally entails a lossy discretization process (Hockett & Hockett, 1960; Pinker, 2003). When LLM agents collapse high-dimensional hidden states into discrete tokens, a wealth of critical implicit information, such as confidence distributions over candidate decisions (Lin et al., 2022; Li et al., 2026), pruned logical branches (Yao et al., 2023a), and vital contextual nuances (Pham et al., 2023), is irreversibly discarded. Such a low-bandwidth bottleneck not only impedes the transmission of multi-faceted reasoning but also frequently precipitates semantic misalignment at the receiver end (Fu et al., 2025). To circumvent this bottleneck, recent works resort to implicit communication (Fu et al., 2025; Zheng et al., 2025; Zou et al., 2025), which transmits latent semantics in a continuous space without discretization. Despite its potential expressiveness, such a black-box paradigm inherently compromises interpretability due to the absence of human-readable textual

[1]The code is available at https://anonymous.4open.science/r/UniCom-DA58/.

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.

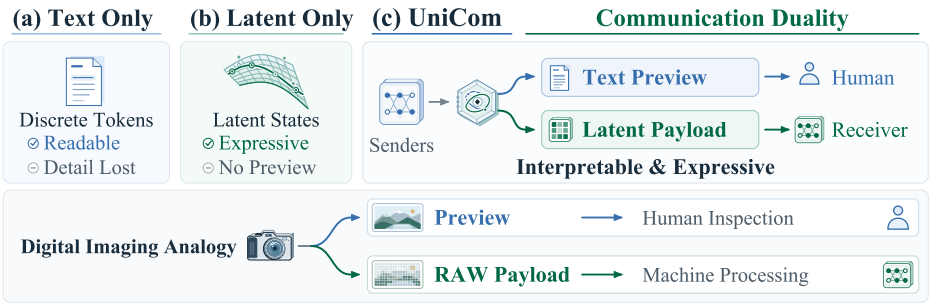

Figure 1: Illustration of the communication duality.

previews, and this opacity further hinders receiver agents from deciphering the latent signals, thus undermining the practicality of implicit protocols in complex multi-agent scenarios (Yan et al., 2025).

This paper moves beyond the explicit-implicit dichotomy and posits that an ideal communication paradigm should *take the best of both worlds* instead of forcing a choice between interpretability and expressiveness. As shown in Figure 1, we envision a unified agentic communication protocol that (*i*) facilitates expressive collaboration by leveraging continuous latent semantics and (*ii*) anchors core reasoning logic within discrete textual space for transparent interaction. To conceptualize this vision, we draw an analogy from digital imaging (Chen et al., 2018; Brooks et al., 2019; Duan et al., 2020), which encodes image data into two synergistic streams, namely a high-fidelity `Raw` payload for machine-to-machine processing and a compressed `Preview` format for human inspection. Such a *communication duality* naturally serves as the conceptual blueprint for our protocol design.

To this end, we introduce **UniCom**, which harmonizes the expressiveness of latent semantics and the interpretability of textual tokens via *a dual-stream adapter*. During inter-agent communication, UniCom first projects the senders' local hidden states onto a shared communication substrate (Huth et al., 2016) for semantic fusion, and then decodes these dense signals into two parallel receiving pathways, wherein one is directed towards the general textual space to establish semantic anchors, and the other is mapped into the receiver-specific latent space under semantic consistency regularization. Notably, to foster high-fidelity utilization of the latent signals without receiver-side retraining, UniCom not only casts the embedded semantics as contextual priors for *instant prefix adaptation*, but also promotes deeper assimilation of the implicit payloads through *dynamic parameter adaptation*. By virtue of such synergistic designs, UniCom dismantles the inherent barrier between explicit and implicit communication and delivers superior collaborative performance across five standard agent benchmarks. The main contributions of this paper are summarized as follows:

- **Blueprint:** Transcending the inherent tension between the interpretability of explicit texts and the expressiveness of latent semantics, we envision an ideal *communication duality* that could simultaneously afford a transparent preview for human inspection and a high-fidelity payload for agentic inference, akin to the dual-stream formats in modern digital imaging.
- **Prototype:** We develop a prototype of Unified Agentic Communication (UniCom) to harmonize the complementary strengths of explicit and implicit communication paradigms. UniCom models discrete textual previews and continuous latent semantics with *a lightweight dual-stream adapter*, accompanied by two decipher mechanisms (i.e., *instant prefix adaptation* and *dynamic parameter adaptation*) for more effective utilization of the latent signals.
- **Evaluation:** We evaluate UniCom across five standard agent benchmarks, covering diverse LLM backbones and task scenarios. These experiments show that our design consistently strengthens agentic collaboration and outperforms both text-only and latent-only baselines, revealing its potential for reconciling agent-to-human and agent-to-agent communication.

2 Preliminaries

Generalized Agentic Communication. Given a complex task $\mathcal{T}$, a multi-agent collaborative system orchestrates the communication network of LLM agents to derive a solution. For clarity of exposition, we consider an atomic n-to-1 communication topology without loss of generality, which comprises n upstream senders $\mathcal{S} = \{A_1, A_2, \ldots, A_n\}$ and a single downstream receiver A_{rcv}. In light of practical agent heterogeneity, we assume that each $A_i \in \mathcal{S} \cup \{A_{\text{rcv}}\}$ is parameterized by specific

foundational weights Θ_i (Vaswani et al., 2017). Each LLM parameterization comprises an embedding module $h^{(i)}_{\text{emb}}$, stacked Transformer blocks $h^{(i)}_{\text{trans}}$, and a language-modeling head $h^{(i)}_{\text{lmh}}$, operating over a potentially distinct vocabulary $\mathcal{V}_i$ and a corresponding continuous hidden space $\mathbb{R}^{d_i}$. During interaction, each sender A_i processes its local context C_i to produce a message M_i, and the receiver A_{rcv} then aggregates the set of incoming messages $\mathcal{M} = \{M_1, \ldots, M_n\}$ alongside its own context C_{rcv} for reasoning. Formally, we can denote the basic n-to-1 communication process as:

$$y = \Phi_{\text{rcv}}(\mathcal{M}, C_{\text{rcv}}; \Theta_{\text{rcv}}), \qquad \mathcal{M} = \{\Phi_{\text{snd}}(C_i; \Theta_i) \mid A_i \in \mathcal{S}\}, \tag{1}$$

where Φ_{snd} and Φ_{rcv} denote the sending and receiving functions that yield the output y at the receiver, such that the communication efficacy hinges on the instantiation of $\Phi = (\Phi_{\text{snd}}, \Phi_{\text{rcv}})$.

Explicit Communication with Information Loss. The prevailing instantiation of Φ relies on explicit textual communication, whose sending function $\Phi^{\text{exp}}_{\text{snd}}$ mandates an autoregressive decoding process, wherein each sender A_i extracts continuous hidden states via $h^{(i)}_{\text{trans}}$ and forces them through $h^{(i)}_{\text{lmh}}$ to sample discrete tokens, thereby collapsing the high-dimensional reasoning states into a discrete textual sequence $M_i = (w_1, \ldots, w_{L_i}) \in \mathcal{V}_i^{L_i}$. Subsequently, the receiving function $\Phi^{\text{exp}}_{\text{rcv}}$ aggregates these discrete messages, typically by concatenating them into a flattened textual prompt $\mathcal{M}_{\text{exp}} = [M_1 \oplus \cdots \oplus M_n]$, and maps $\mathcal{M}_{\text{exp}}$ back into hidden space via its own embedding module $h^{(\text{rcv})}_{\text{emb}}$ to condition the generation. While natural language serves as a universally compatible interface, this process inherently suffers from a severe *discretization bottleneck*, since the non-differentiable sampling from the continuous space $\mathbb{R}^{d_i}$ to the discrete vocabulary $\mathcal{V}_i$ irreversibly truncates the probabilistic reasoning states of each sender, discarding its latent confidence information.

Implicit Communication with Latent Signals. A straightforward approach to breaking this bandwidth bottleneck is to bypass the lossy discretization process via implicit communication, where the sending function $\Phi^{\text{imp}}_{\text{snd}}$ directly outputs continuous reasoning states rather than discrete texts. Such internal reasoning states can manifest in various forms, ranging from accumulated key-value caches (Fu et al., 2025) to pre-decoding contextual representations (Zheng et al., 2025; Du et al., 2025). For instance, following continuous autoregressive decoding (Hao et al., 2024), one can formulate $\Phi^{\text{imp}}_{\text{snd}}$ such that each sender A_i executes k consecutive forward passes via $h^{(i)}_{\text{trans}}$, which bypasses the language-head projection $h^{(i)}_{\text{lmh}}$ and produces a fixed-length sequence of latent messages $\mathbf{M}_i = [\mathbf{h}^{(i)}_1, \ldots, \mathbf{h}^{(i)}_k] \in \mathbb{R}^{k \times d_i}$. Correspondingly, the receiving function $\Phi^{\text{imp}}_{\text{rcv}}$ is expected to assimilate these latent payloads directly into the receiver agent A_{rcv} for downstream inference.

Challenges. While implicit communication potentially affords high bandwidth, its deployment is constrained by three fundamental challenges, namely (*i*) *Black-box Opacity*, where the total circumvention of discrete vocabularies renders inter-agent reasoning logic opaque and unauditable, (*ii*) *Semantic Instability*, where the absence of structural regularization from discrete textual anchors exposes continuous representations to semantic drift that hinders stable generation, and (*iii*) *Cross-Agent Heterogeneity*, where disparate backbone architectures and coordinate spaces preclude direct interoperability across models without extensive retraining. These limitations necessitate a unified protocol that can harness the bandwidth of latent signals while retaining the interpretability and stability afforded by explicit textual previews, which motivates the design of UniCom.

3 Unified Agentic Communication

In this section, we introduce UniCom, a unified agentic communication protocol designed to reconcile the dichotomy between textual interpretability and latent expressiveness. We first present the overall communication pipeline of UniCom (Section 3.1). Then, we further detail the architectural instantiation of our generalized protocol $\Phi^{\text{uni}} = (\Phi^{\text{uni}}_{\text{snd}}, \Phi^{\text{uni}}_{\text{rcv}})$, delineating its message transmission and reception mechanisms (Section 3.2). Finally, we formulate the end-to-end training objective (Section 3.3) to simultaneously foster structural interpretability and semantic expressiveness.

3.1 Overview

The overall communication pipeline of UniCom is illustrated in Figure 2. To unify the respective strengths of explicit textual and implicit latent paradigms, our framework encapsulates the inter-agent

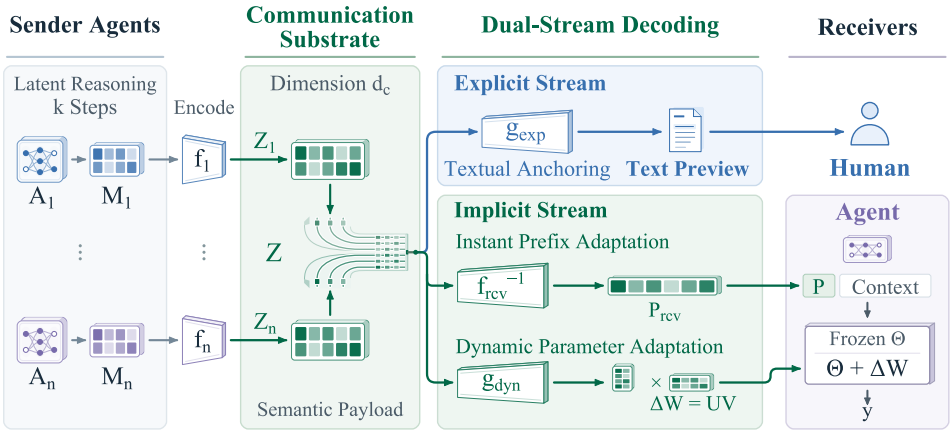

Figure 2: Overview of our UniCom prototype implementation.

communication into a coherent *encode-then-decode* workflow. Specifically, UniCom consists of three functional modules, namely a sender-specific autoencoder set that projects heterogeneous reasoning states onto a shared communication substrate, a global explicit decoder designated for transparent textual anchoring, and a receiver-specific implicit decoder tailored for dynamic semantic adaptation.

Upon receiving a collaborative task $\mathcal{T}$, UniCom initiates the communication workflow by prompting each sender $A_i \in \mathcal{S}$ to execute k consecutive forward passes without language-head projection. This standard latent reasoning procedure (Hao et al., 2024) unveils a continuous sequence of internal hidden states $\mathbf{M}_i = [\mathbf{h}_1^{(i)}, \ldots, \mathbf{h}_k^{(i)}] \in \mathbb{R}^{k \times d_i}$. Rather than transmitting these heterogeneous signals directly, A_i leverages its dedicated autoencoder f_i to project $\mathbf{M}_i$ into $\mathbf{Z}_i \in \mathbb{R}^{k \times d_c}$ within a shared communication space of dimension d_c. The encoded representations from all upstream senders are concatenated into the aggregated latent semantic payload $\mathbf{Z} = [\mathbf{Z}_1, \ldots, \mathbf{Z}_n]$, which is subsequently disentangled through two parallel decoding pathways. Along the explicit decoding stream, an explicit textual decoder g_{exp} converts $\mathbf{Z}$ into a human-readable natural language preview, thereby regularizing the continuous space with discrete semantic anchors. Concurrently, within the implicit decoding stream, a receiver-specific latent decoder, comprising f_{rcv}^{-1} and $g_{\text{dyn}}^{(\text{rcv})}$, transforms $\mathbf{Z}$ into the receiver's coordinate space to facilitate downstream inference with two synergistic decipher mechanisms.

Such a cohesive dual-stream adapter harmonizes textual interpretability with latent expressiveness to establish the communication duality. For the challenges identified in Section 2, the explicit decoding stream not only provides a transparent language preview that penetrates black-box opacity, but also reinforces the semantic stability of the latent space through textual anchoring. Concurrently, projecting disparate agentic reasoning states onto a shared communication substrate circumvents the quadratic cross-adaptation bottleneck (Fu et al., 2025) typical of heterogeneous multi-agent networks. Furthermore, the *encode-then-decode* pipeline preserves end-to-end differentiability throughout the forward pass, thereby enabling efficient optimization of the communication adapters while sparing the prohibitive computational overhead associated with retraining the foundational LLM backbones.

3.2 Dual-Stream Communication Workflow

To operationalize the UniCom protocol, we formally delineate the step-by-step mechanics of the dual-stream communication workflow, detailing how the generalized functions $\Phi_{\text{snd}}^{\text{uni}}$ and $\Phi_{\text{rcv}}^{\text{uni}}$ are instantiated. We elucidate how heterogeneous agentic reasoning states are first elicited, harmonized into a shared substrate, and finally disentangled for textual interpretability and latent expressiveness.

3.2.1 Message Transmission

Sender-Side Latent Reasoning. The agentic communication protocol commences with the extraction of high-fidelity cognitive representations from each sender. Unlike traditional paradigms that collapse internal states into discrete tokens via the language-modeling head (Vaswani et al., 2017), UniCom operates entirely within the continuous space to preserve subtle reasoning nuances. Specifically, for each sender $A_i \in \mathcal{S}$, its local context C_i alongside the task query $\mathcal{T}$ is initially projected by the input embedding layer into a continuous token sequence $\mathbf{C}_0^{(i)} \in \mathbb{R}^{l_0 \times d_i}$, where l_0 is the initial

context length, and the agent then executes k consecutive steps of implicit reasoning. At each step $t \in \{1, \ldots, k\}$, the sender A_i processes the current context through its foundational Transformer blocks, denoted as $h^{(i)}_{\text{trans}}(\cdot; \Theta_i)$, to generate the subsequent hidden representation $\mathbf{h}^{(i)}_t$. Instead of discrete sampling, these dense states are concatenated to the ongoing context along the sequence axis, serving as input for subsequent reasoning steps:

$$\mathbf{h}^{(i)}_t = h^{(i)}_{\text{trans}}\big(\mathbf{C}^{(i)}_{t-1}; \Theta_i\big) \in \mathbb{R}^{d_i}, \qquad \mathbf{C}^{(i)}_t = \mathbf{C}^{(i)}_{t-1} \oplus \mathbf{h}^{(i)}_t, \tag{2}$$

where $\oplus$ denotes concatenation along the sequence axis. This latent reasoning procedure thus yields a dense semantic payload $\mathbf{M}_i = [\mathbf{h}^{(i)}_1, \ldots, \mathbf{h}^{(i)}_k] \in \mathbb{R}^{k \times d_i}$, which encapsulates the contextual understanding, logical reasoning paths, and implicit confidence distributions of each sender A_i.

Shared Communication Substrate. Despite the rich semantic density of $\mathbf{M}_i$, direct transmission of such internal payloads is hindered by cross-agent heterogeneity, as disparate hidden dimensions and misaligned coordinate spaces across different LLM backbones inherently impede semantic integration at the receiver end. To circumvent this barrier, we complete the formulation of the sending function $\Phi^{\text{uni}}_{\text{snd}}$ by cascading the latent reasoning process with an underlying projection onto a shared communication substrate. Specifically, each sender agent A_i is equipped with a lightweight autoencoder (f_i, f_i^{-1}), wherein a projection function $f_i : \mathbb{R}^{k \times d_i} \to \mathbb{R}^{k \times d_c}$ faithfully standardizes the agent-specific latents $\mathbf{M}_i \in \mathbb{R}^{k \times d_i}$ into a unified space of dimension d_c:

$$\mathbf{Z}_i = f_i(\mathbf{M}_i) \in \mathbb{R}^{k \times d_c}. \tag{3}$$

An ideal communication projection should constitute a high-fidelity semantic compression rather than a destructive dimensionality reduction. To enforce this property, we employ a reconstruction criterion (Hyvärinen et al., 2024) for each f_i during optimization, presupposing the existence of an inverse decoding function $f_i^{-1} : \mathbb{R}^{k \times d_c} \to \mathbb{R}^{k \times d_i}$ that can recover the raw latent reasoning payloads (i.e., $f_i^{-1}(\mathbf{Z}_i) \approx \mathbf{M}_i$). With such communication autoencoders, UniCom encourages the shared substrate to retain essential topological characteristics and semantic richness of the reasoning manifolds.

Sending Procedure of UniCom. Consequently, the overarching sending function $\Phi^{\text{uni}}_{\text{snd}}$ of UniCom is formally instantiated as the composition of the latent reasoning and the communication projection:

$$\mathbf{Z}_i = \Phi^{\text{uni}}_{\text{snd}}(C_i; \Theta_i) = f_i\Big(\big[h^{(i)}_{\text{trans}}(\mathbf{C}^{(i)}_{t-1}; \Theta_i)\big]^k_{t=1}\Big). \tag{4}$$

This process naturally bypasses the quadratic cross-adaptation issue in heterogeneous agentic systems, delivering standardized messages $\mathcal{M} = \{\mathbf{Z}_i\}^n_{i=1}$ that are compatible with downstream reception.

3.2.2 Message Reception

Upon gathering the standardized sending messages $\mathcal{M} = \{\mathbf{Z}_i\}^n_{i=1}$ within the shared communication substrate, UniCom aggregates them along the sequence dimension to synthesize a fused latent payload $\mathbf{Z} = [\mathbf{Z}_1, \ldots, \mathbf{Z}_n] \in \mathbb{R}^{(n \times k) \times d_c}$. The reception phase disentangles this payload into two parallel streams, namely an explicit decoding stream for human-readable textual preview, and an implicit decoding stream that enables the receiver to assimilate these signals for downstream inference.

Transparent Textual Anchoring. To resolve the opacity of latent messages, UniCom employs an explicit decoding stream that projects continuous representations into human-readable text. The fused payload $\mathbf{Z}$ is processed by a shared textual translator g_{exp}, parameterized by θ_{exp}, to generate a natural language preview $\mathbf{Y}_{\text{prev}} \in \mathcal{V}^L_{\text{exp}}$:

$$\mathbf{Y}_{\text{prev}} = g_{\text{exp}}(\mathbf{Z}; \theta_{\text{exp}}). \tag{5}$$

Notably, this textual stream not only renders a transparent visualization interface for human overseers, but also operates as a structural regularizer that intrinsically grounds the latent signals in discrete semantics. By compelling the shared communication space to retain the capacity for coherent natural language generation, this decoding pathway stabilizes the latent payload $\mathbf{Z}$ and prevents it from drifting into nonsensical regions during the joint optimization of communication adapters.

Instant Prefix Adaptation. Beyond furnishing an interpretable agent-to-human preview, the fundamental imperative of UniCom resides in effectively assimilating the latent semantics into the receiver agent for agent-to-agent collaboration. To this end, we introduce *Instant Prefix Adaptation* as a concise mechanism for implicit semantic integration. Specifically, to bridge coordinate misalignments

between the shared communication substrate and the receiver's reasoning manifold, the receiver A_{rcv} utilizes its inverse projection function f_{rcv}^{-1} to map the payload $\mathbf{Z}$ into its intrinsic space:

$$\mathbf{P}_{\text{rcv}} = f_{\text{rcv}}^{-1}(\mathbf{Z}) \in \mathbb{R}^{(n\times k)\times d_{\text{rcv}}}. \tag{6}$$

These projected representations $\mathbf{P}_{\text{rcv}}$ function as continuous soft prompts (Li & Liang, 2021) that encapsulate the collaborative consensus of all upstream senders. By prepending $\mathbf{P}_{\text{rcv}}$ to the context C_{rcv}, the inference process of A_{rcv} is denoted as:

$$y = h_{\text{lmh}}^{(\text{rcv})}\Big(h_{\text{trans}}^{(\text{rcv})}\big(\mathbf{P}_{\text{rcv}} \oplus h_{\text{emb}}^{(\text{rcv})}(C_{\text{rcv}}); \Theta_{\text{rcv}}\big)\Big), \tag{7}$$

where $h_{\text{emb}}^{(\text{rcv})}$ represents the receiver's embedding module, and $\oplus$ is the concatenation operation. This decipher strategy leverages the self-attention blocks to reference collaborative intents from the initial layer, augmenting downstream generation without requiring modifications to the receiver parameters.

Dynamic Parameter Adaptation. While the instant prefix adaptation effectively injects collaborative priors at the input level of A_{rcv}, its influence may attenuate across cascaded Transformer blocks due to semantic dilution (Lester et al., 2021; Li & Liang, 2021; Dong et al., 2021). To foster deeper structural assimilation of the latent payloads, we further introduce *Dynamic Parameter Adaptation*, which modulates the receiver's backbone weights to directly rewire its internal reasoning pathways. Since fully updating the LLM parameters for each communicative interaction is computationally intractable, we resort to Low-Rank Adaptation (LoRA) (Hu et al., 2022) as an efficient vehicle for this transient intervention. Unlike conventional methods that optimize static LoRA weights, UniCom employs a dedicated communication hypernetwork $g_{\text{dyn}}^{(\text{rcv})}$ for A_{rcv} to adaptively synthesize LoRA weights conditioned on the incoming payload $\mathbf{Z}$. Specifically, for a given target module m (e.g., the query or value projections) at layer l of the receiver backbone, $g_{\text{dyn}}^{(\text{rcv})}$ ingests $\mathbf{Z}$ alongside the target embeddings to compute LoRA matrices:

$$(U_{\text{rcv}}^{(l,m)}, V_{\text{rcv}}^{(l,m)}) = g_{\text{dyn}}^{(\text{rcv})}\big(\mathbf{Z}, \mathbf{e}_l, \mathbf{e}_m; \theta_{\text{dyn}}\big), \tag{8}$$

where $U_{\text{rcv}}^{(l,m)} \in \mathbb{R}^{d_{\text{rcv}}\times r}$ and $V_{\text{rcv}}^{(l,m)} \in \mathbb{R}^{r\times d_{\text{rcv}}}$ denote the LoRA matrices with rank r. The context-aware parameter offsets are then computed as $\Delta W_{\text{rcv}}^{(l,m)} = U_{\text{rcv}}^{(l,m)} V_{\text{rcv}}^{(l,m)}$. By injecting these transient updates into the backbone weights, A_{rcv} dynamically reconfigures its internal reasoning pathways to align with the upstream messages. Notably, the *prefix* and *parameter* adaptation constitute *synergistic reception mechanisms* rather than isolated strategies. While the former directly injects collaborative priors at the input level via continuous soft prompts, the latter tailors the backbone weights of every Transformer layer to modulate the reasoning dynamics at the parametric level.

Receiving Procedure of UniCom. UniCom employs a dual-stream receiving procedure to realize the targeted communication duality, which renders transparent textual previews for interpretable semantic anchoring while assimilating high-fidelity latent signals for more expressive inference. Formally, with θ_{rcv} denoting the receiver-side adapter parameters, the receiving function $\Phi_{\text{rcv}}^{\text{uni}}$ is defined as:

$$\Phi_{\text{rcv}}^{\text{uni}}(\mathbf{Z}, C_{\text{rcv}}; \Theta_{\text{rcv}}, \theta_{\text{rcv}}) = \Big(\underbrace{g_{\text{exp}}(\mathbf{Z}; \theta_{\text{exp}})}_{\text{Explicit Stream}}, \underbrace{h_{\text{lmh}}^{(\text{rcv})}\big(h_{\text{trans}}^{(\text{rcv})}(\mathbf{P}_{\text{rcv}} \oplus h_{\text{emb}}^{(\text{rcv})}(C_{\text{rcv}}); \Theta_{\text{rcv}} + \Delta W_{\text{rcv}})\big)}_{\text{Implicit Stream}}\Big). \tag{9}$$

Through this reception duality, UniCom enables the receiver to transcend passive contextual observation, internally reconfiguring its reasoning pathways to align with upstream messages.

3.3 Optimization

Training Objectives. The training objective comprises three terms, wherein an autoregressive language modeling loss $\mathcal{L}_{\text{task}}$ ensures task performance on the receiver output, a cross-entropy loss $\mathcal{L}_{\text{exp}}$ supervises explicit rationale reconstruction via g_{exp}, and a latent reconstruction loss $\mathcal{L}_{\text{rec}}$ enforces representational fidelity by minimizing the mean squared error between each $\mathbf{M}_i$ and $f_i^{-1}(\mathbf{Z}_i)$. The unified objective is formulated as:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{exp}}\mathcal{L}_{\text{exp}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}}, \tag{10}$$

where λ_{exp} and λ_{rec} are balancing hyperparameters that modulate the relative weights of rationale transparency and latent fidelity against the primary task objective $\mathcal{L}_{\text{task}}$ of the receiver.

Table 1: Overall accuracy comparison of UniCom against representative communication baselines on five benchmarks under two heterogeneous multi-agent scales. The best results are in **bold** and the second-best are underlined, with colored subscripts denoting the absolute change relative to NoCom.

Method	MMLU		C-Eval		GSM8K		MATH		HumanEval	
	Small	Large	Small	Large	Small	Large	Small	Large	Small	Large
NoCom	36.0	68.5	28.9	70.3	59.2	89.0	43.4	70.2	16.5	62.8
TextCom	36.6$_{\uparrow 0.6}$	70.8$_{\uparrow 2.3}$	33.5$_{\uparrow 4.6}$	69.9$_{\downarrow 0.4}$	63.0$_{\uparrow 3.8}$	91.2$_{\uparrow 2.2}$	49.0$_{\uparrow 5.6}$	71.0$_{\uparrow 0.8}$	21.3$_{\uparrow 4.8}$	65.2$_{\uparrow 2.4}$
Cache-to-Cache	36.9$_{\uparrow 0.9}$	69.9$_{\uparrow 1.4}$	36.8$_{\uparrow 7.9}$	72.0$_{\uparrow 1.7}$	65.6$_{\uparrow 6.4}$	92.0$_{\uparrow 3.0}$	42.2$_{\downarrow 1.2}$	67.4$_{\downarrow 2.8}$	18.9$_{\uparrow 2.4}$	62.2$_{\downarrow 0.6}$
ThoughtComm	35.1$_{\downarrow 0.9}$	70.5$_{\uparrow 2.0}$	27.2$_{\downarrow 1.7}$	71.1$_{\uparrow 0.8}$	64.0$_{\uparrow 4.8}$	86.8$_{\downarrow 2.2}$	47.8$_{\uparrow 4.4}$	69.0$_{\downarrow 1.2}$	22.6$_{\uparrow 6.1}$	64.0$_{\uparrow 1.2}$
Interlat	37.5$_{\uparrow 1.5}$	70.8$_{\uparrow 2.3}$	35.6$_{\uparrow 6.7}$	72.4$_{\uparrow 2.1}$	66.2$_{\uparrow 7.0}$	92.2$_{\uparrow 3.2}$	48.0$_{\uparrow 4.6}$	70.2$_{\uparrow 0.0}$	24.4$_{\uparrow 7.9}$	65.9$_{\uparrow 3.1}$
UniCom	**38.4**$_{\uparrow 2.4}$	**72.0**$_{\uparrow 3.5}$	**38.5**$_{\uparrow 9.6}$	**74.1**$_{\uparrow 3.8}$	**68.4**$_{\uparrow 9.2}$	**93.6**$_{\uparrow 4.6}$	**49.8**$_{\uparrow 6.4}$	**73.0**$_{\uparrow 2.8}$	**26.2**$_{\uparrow 9.7}$	**68.3**$_{\uparrow 5.5}$

Module Instantiation. The communication projections f_i and f_i^{-1} are implemented as multi-layer perceptrons with GELU activations (Hendrycks & Gimpel, 2016), and the explicit textual translator g_{exp} is parameterized as a small language model that autoregressively decodes the fused latent payload $\mathbf{Z}$ into the vocabulary space. For dynamic parameter adaptation, the hypernetwork g_{dyn} adopts a bottleneck linear architecture (Charakorn et al., 2025) to generate LoRA matrices. With the sender agents initialized via standard implicit reasoning training (Hao et al., 2024; Zhang et al., 2026), the communication adapter parameters (i.e., θ_{exp}, $\{f_i, f_i^{-1}\}$, and g_{dyn}) are jointly optimized, while the receiver backbone remains frozen throughout training (details in Appendix E).

Complexity Analysis. UniCom circumvents the computational bottlenecks inherent in traditional multi-agent fine-tuning, since the trainable parameters within the communication adapters constitute a negligible fraction (typically less than 2%) of the foundational LLM parameters. At inference, the explicit stream can be bypassed for pure latent interaction to eliminate the text generation latency, while the overhead of the implicit stream remains marginal as it is confined to lightweight projections and LoRA computations. Furthermore, by transmitting dense embeddings rather than verbose prompt strings or full key-value caches, UniCom enhances context efficiency and reduces the communication overhead of inter-agent collaboration, as empirically verified in Section 4.3.

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate UniCom on five widely adopted benchmarks covering diverse agentic scenarios, including MMLU (Hendrycks et al., 2020) and C-Eval (Huang et al., 2023) for multi-disciplinary knowledge, GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021) for mathematical reasoning, and HumanEval (Chen et al., 2021) for program synthesis, with the evaluation splits, scoring metrics, and detailed descriptions of these datasets provided in Appendix C.

Baselines. We compare UniCom with a series of representative baselines categorized by their communication patterns, namely (*i*) *No Communication* (NoCom), where the receiver solves each task in isolation, (*ii*) *Textual Communication* (TextCom), where the senders' textual rationales are appended to the receiver context for inference, and (*iii*) three recent *Implicit Communication* methods, including Cache-to-Cache (Fu et al., 2025), ThoughtComm (Zheng et al., 2025), and Interlat (Du et al., 2025), which differ in the continuous signals they transmit and are further detailed in Appendix D.

Implementation Details. Our main experiments adopt two multi-agent configurations of distinct scales, each composed of three divergent backbones to instantiate the cross-agent heterogeneity formalized in Section 2. The *small*-scale setup employs Qwen2.5-1.5B-Instruct and Llama-3.2-1B-Instruct as the senders and Qwen3-0.6B as the receiver, whereas the *large*-scale setup scales the senders to Qwen2.5-7B-Instruct and Llama-3.1-8B-Instruct with Qwen3-4B-Instruct-2507 as the receiver. By default, each sender emits $k = 32$ latent tokens, and the receiver adopts rank-8 dynamic LoRA. During training, the receiver backbones remain frozen while only the communication adapters are trained with AdamW (Loshchilov & Hutter, 2017) under a cosine learning rate schedule, with further configurations and training details given in Section 4.3 and Appendix E.

4.2 Main Results

Table 1 reports the overall accuracy on five benchmarks under two heterogeneous agent scales, where **UniCom attains the best accuracy in every (task, scale) cell**. **(i)** The gains are robust across knowledge tasks (MMLU, C-Eval), mathematical reasoning (GSM8K, MATH), and program synthesis (HumanEval), with absolute improvements over the no-communication reference ranging from $+2.4$ to $+9.7$ points, which indicates that the dual-stream design generalizes across task families. **(ii)** The advantage over TextCom is generally larger under the small-scale configuration, where the limited capacity of the receiver makes external context most valuable, and UniCom exceeds TextCom by $+5.0$ on C-Eval and $+5.4$ on GSM8K, suggesting that latent payloads carry information lost to tokenization. **(iii)** Interlat is the strongest implicit baseline and often shares second place with TextCom, yet UniCom still surpasses the best implicit baseline by $+0.9$ to $+2.8$ points in every cell. We attribute this advantage to the explicit textual stream, which anchors the latent payload to discrete semantic structure and curbs the representational drift typical of purely latent protocols.

4.3 More Analyses

Diverse Multi-Agent Configurations. We further evaluate UniCom at four points of the sender-receiver capability spectrum, covering the homogeneous **S→S** and **L→L** settings reused from the main experiments as well as two mixed regimes, namely **L→S**, in which capable senders orchestrate a constrained receiver, and **S→L**, in which weak senders inform a stronger receiver. Backbone assignments and the MMLU counterpart are deferred to Appendix F. As shown in Table 2, UniCom remains the strongest protocol in every regime, and its margin is widest in **L→S** ($+2.2$ over the best baseline), where the receiver cannot solve the task alone and accuracy hinges on whether the sender signal survives the channel. In **S→L**, weak senders supply less novel information and all protocols converge towards the receiver's standalone ceiling, yet UniCom still performs best, indicating its robustness to noisy upstream signals.

Table 2: Accuracy on MATH under four heterogeneous agent configurations.

Method	S→S	L→S	S→L	L→L
TextCom	49.0	60.2	69.6	71.0
Cache-to-Cache	42.2	58.2	64.0	67.4
ThoughtComm	47.8	56.8	68.2	69.0
Interlat	48.0	59.4	67.6	70.2
UniCom	**49.8**	**62.4**	**70.8**	**73.0**

Communication Efficiency. Figure 3 plots accuracy against single-GPU sequential latency on MMLU and MATH, where UniCom occupies the lower-right Pareto frontier in both panels by attaining the highest accuracy at substantially lower wall-clock cost than TextCom while remaining competitive in latency with the implicit baselines. These savings stem from replacing the autoregressive decoding of each sender, which generates hundreds of tokens per sample, with a short sequence of k latent forward passes that emits the compact latent payload. UniCom thus runs $2.4\times$ and $1.7\times$ faster than TextCom on MMLU and MATH while improving accuracy by $+1.2$ and $+2.0$ points, respectively.

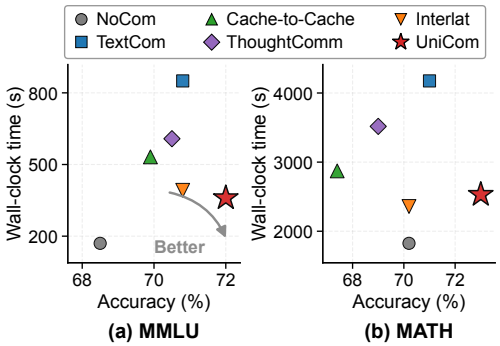

Figure 3: Efficiency-performance trade-off.

Ablation Variants. Table 3 evaluates the components of UniCom under the L→L configuration, where removing any single component degrades performance on all three benchmarks. Regarding training objectives, removing the explicit stream loss $\mathcal{L}_{\text{exp}}$ causes a significant drop of up to -4.9 points on HumanEval, indicating its role as a structural anchor against semantic drift, while omitting the latent reconstruction loss $\mathcal{L}_{\text{rec}}$ leads to a moderate yet consistent decline, which demonstrates that encoder invertibility is essential for semantic recovery. Regarding reception, removing instant prefix adaptation incurs the largest drops on MMLU and MATH (-2.7 and -2.6 points),

Table 3: Component ablation under L→L.

Variant	MMLU	MATH	HumanEval
UniCom (Full)	**72.0**	**73.0**	**68.3**
w/o $\mathcal{L}_{\text{exp}}$	70.5	70.8	63.4
w/o $\mathcal{L}_{\text{rec}}$	71.7	71.2	64.6
w/o Prefix Adaptation	69.3	70.4	64.0
w/o Param. Adaptation	71.1	72.4	66.5

confirming its necessity for surfacing latent payloads, whereas removing dynamic parameter adaptation results in a smaller yet consistent loss of at most 1.8 points on every benchmark. These results indicate that the two reception mechanisms are complementary, wherein instant prefix adaptation injects collaborative priors at the input embedding level and dynamic parameter adaptation sustains their influence against semantic dilution across the deeper Transformer blocks of the frozen receiver.

Hyperparameter Sensitivity. We next study two structural hyperparameters, namely the per-sender latent length k, which controls the bandwidth of each implicit message, and the number of communicating senders n, which determines the size of the agent pool. Figure 4 reports the mean and standard deviation over three runs, with the TextCom reference overlaid as gray dashed lines in panel (b). As k increases, accuracy rises sharply from $k = 4$ to $k = 16$ and then saturates, with our default $k = 32$ lying at the knee of the curve, since overly short payloads cannot carry nuanced reasoning while longer ones yield diminishing returns and even a slight decline on MATH. As n increases, UniCom improves overall and peaks at $n = 5$ on MATH and at $n = 8$ on MMLU, whereas TextCom plateaus near $n = 3$ and *regresses* thereafter as accumulated rationales exhaust the receiver's context. UniCom avoids this degradation because each sender emits only $k = 32$ latent tokens rather than hundreds of textual ones.

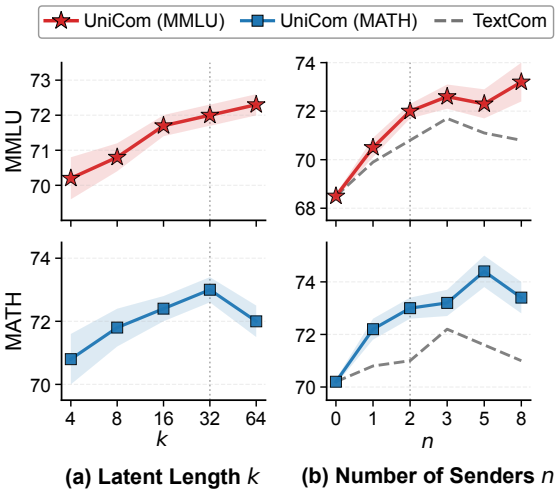

Figure 4: Hyperparameter sensitivity of UniCom.

5 Related Work

Multi-Agent Collaboration. The multi-agent paradigm has emerged as a powerful approach for decomposing complex tasks spanning reasoning and software engineering (Yan et al., 2025), wherein LLM agents specialize in distinct sub-skills and exchange intermediate insights to converge on solutions (Li et al., 2025; Hong et al., 2024; Yao et al., 2023b). Recent literature further indicates that the efficacy of agentic collaboration is dictated by the fidelity of inter-agent information exchange rather than by individual backbone capabilities alone (Fu et al., 2025; Yan et al., 2025). Consequently, research has shifted from optimizing individual agents to refining communication protocols, with increasing focus on the content and representational form of inter-agent information flow.

From Textual to Latent Communication. Most multi-agent frameworks rely on natural language for communication (Du et al., 2024; Zhang et al., 2025) owing to its readability and flexibility. However, the requisite tokenization compresses continuous hidden states into discrete sequences and inevitably discards nuanced signals such as calibration and internal reasoning dynamics (Lin et al., 2022; Yao et al., 2023a). To address this limitation, recent studies transmit internal reasoning states directly via KV-cache injection or shared latent thoughts (Fu et al., 2025; Zheng et al., 2025; Zou et al., 2025; Du et al., 2025), echoing the broader shift towards continuous-space reasoning (Hao et al., 2024). Although these latent methods improve expressivity, they remain inherently susceptible to black-box opacity and ungrounded semantic drift. UniCom reconciles this tension by leveraging textual generation as a structural regularizer that anchors the latent payloads, thereby delivering both interpretability and expressiveness for collaborative reasoning across heterogeneous agents.

6 Conclusion

This paper introduces **UniCom**, a unified framework that reconciles textual interpretability and latent expressiveness for inter-agent communication. With a lightweight dual-stream adapter, UniCom emits a transparent textual preview alongside high-fidelity latent payloads, integrated via instant prefix and dynamic parameter adaptation to facilitate message reception without backbone retraining. Extensive experiments show that UniCom outperforms both explicit and implicit baselines while maintaining inference efficiency and semantic transparency. Limitations are discussed in Appendix A.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish the writing of the manuscript, such as improving grammar, wording, and readability. All AI-assisted content has been carefully reviewed and revised by the authors to ensure that it is accurate and faithfully reflects our work. We take responsibility for the final content of this paper, including any text produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we release the anonymized source code at `https://anonymous.4open.science/r/UniCom-DA58/`, which includes the evaluation data splits and the scripts for training and evaluation. The formulation of each UniCom component and the training objective are detailed in Sections 3.2 and 3.3, respectively. Appendix C describes the benchmark splits and evaluation metrics, Appendix D the baseline settings, and Appendix E the architecture, optimization, and evaluation details, while Table 6 summarizes the backbone assignments of all agent configurations.

REFERENCES

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11036–11045, 2019.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Rujikorn Charakorn, Edoardo Cetin, Yujin Tang, and Robert Tjarko Lange. Text-to-lora: Instant transformer adaption. *arXiv preprint arXiv:2506.06105*, 2025.

Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3291–3300, 2018.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24:240:1–240:113, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International conference on machine learning*, pp. 2793–2803, 2021.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*, 2024.

Zhuoyun Du, Runze Wang, Huiyu Bai, Zouying Cao, Xiaoyong Zhu, Yu Cheng, Bo Zheng, Wei Chen, and Haochao Ying. Enabling agents to communicate entirely in latent space. *arXiv preprint arXiv:2511.09149*, 2025.

Lingyu Duan, Jiaying Liu, Wenhan Yang, Tiejun Huang, and Wen Gao. Video coding for machines: A paradigm of collaborative compression and intelligent analytics. *IEEE Transactions on Image Processing*, 29:8680–8695, 2020.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

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. *arXiv preprint arXiv:2510.03215*, 2025.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Charles F Hockett and Charles D Hockett. The origin of speech. *Scientific American*, 203(3):88–97, 1960.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations*, 2024.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Yao Fu, et al. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. *Advances in neural information processing systems*, 36: 62991–63010, 2023.

Alexander G Huth, Wendy A De Heer, Thomas L Griffiths, Frédéric E Theunissen, and Jack L Gallant. Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532 (7600):453–458, 2016.

Aapo Hyvärinen, Ilyes Khemakhem, and Ricardo Monti. Identifiability of latent-variable and structural-equation models: from linear to nonlinear. *Annals of the Institute of Statistical Mathematics*, 76(1):1–33, 2024.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059, 2021.

Rui Li, Quanyu Dai, Zeyu Zhang, Xu Chen, Zhenhua Dong, and Ji-Rong Wen. Knowtrace: Bootstrapping iterative retrieval-augmented generation with structured knowledge tracing. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 1470–1480, 2025.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, 2021.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Chau Pham, Boyi Liu, Yingxiang Yang, Zhengyu Chen, Tianyi Liu, Jianbo Yuan, Bryan A Plummer, Zhaoran Wang, and Hongxia Yang. Let models speak ciphers: Multiagent debate through embeddings. *arXiv preprint arXiv:2310.06272*, 2023.

Steven Pinker. *The language instinct: How the mind creates language*. Penguin uK, 2003.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *arXiv preprint arXiv:2308.11432*, 2023.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, et al. The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*, 2023.

Bingyu Yan, Zhibo Zhou, Litian Zhang, Lian Zhang, Ziyi Zhou, Dezhuang Miao, Zhoujun Li, Chaozhuo Li, and Xiaoming Zhang. Beyond self-talk: A communication-centric survey of llm-based multi-agent systems. *arXiv preprint arXiv:2502.14321*, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822, 2023a.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023b.

Guibin Zhang, Yanwei Yue, Zhixun Li, Sukwon Yun, Guancheng Wan, Kun Wang, Dawei Cheng, Jeffrey Xu Yu, and Tianlong Chen. Cut the crap: An economical communication pipeline for LLM-based multi-agent systems. In *The Thirteenth International Conference on Learning Representations*, 2025.

Zeyu Zhang, Rui Li, Xiaoyan Zhao, Yang Zhang, Wenjie Wang, Xu Chen, and Tat-Seng Chua. Nextmem: Towards latent factual memory for llm-based agents. *arXiv preprint arXiv:2603.15634*, 2026.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

Yujia Zheng, Zhuokai Zhao, Zijian Li, Yaqi Xie, Mingze Gao, Lizhu Zhang, and Kun Zhang. Thought communication in multiagent collaboration. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jiaru Zou, Ruizhong Qiu, Gaotang Li, Xiyuan Yang, Katherine Tieu, Pan Lu, Ke Shen, Hanghang Tong, Yejin Choi, Jingrui He, et al. Latent collaboration in multi-agent systems. *arXiv preprint arXiv:2511.20639*, 2025.

A LIMITATIONS

Despite the consistent improvements that UniCom achieves across our experiments, two limitations remain. The first concerns the scope of generalization, as the communication adapters are currently trained on a per-task basis, with each downstream benchmark requiring its own fitted instance. An idealized version of UniCom would instead deliver a generalizable adapter that encodes task-invariant semantics and transfers zero-shot across heterogeneous benchmarks, thereby turning the protocol into a general-purpose communication primitive for multi-agent systems, and we view such cross-task transferability as a natural next step of this work. The second concerns the model scale, since our experiments are confined to lightweight and mid-scale agent pools and leave the behavior of UniCom at frontier scales empirically untested. Whether the reported gains persist or amplify when both the senders and the receiver approach frontier capability remains an open question whose answer requires compute beyond our current budget, and we leave a systematic scaling study to future work.

B GLOSSARY OF SYMBOLS

For ease of reference, Table 4 consolidates the principal notation used throughout the paper.

Table 4: Notation used in this paper.

Symbol	Description
$\mathcal{T}$	Collaborative task instance
$\mathcal{S} = \{A_1, \ldots, A_n\}$	Set of n upstream sender agents
A_{rcv}	Downstream receiver agent
Θ_i	Frozen backbone parameters of agent A_i
d_i	Hidden dimension of agent A_i
$\Phi_{\text{snd}}, \Phi_{\text{rcv}}$	Message sending and receiving functions
l_0	Initial context length of a sender
k	Latent length emitted per sender
d_c	Dimension of the shared communication space
$\mathbf{M}_i \in \mathbb{R}^{k \times d_i}$	Raw latent message of sender A_i
$\mathbf{Z}_i \in \mathbb{R}^{k \times d_c}$	Encoded message of A_i in the shared space
$\mathbf{Z} \in \mathbb{R}^{(n \times k) \times d_c}$	Aggregated payload across senders
f_i, f_i^{-1}	Encoder and decoder of agent A_i
$g_{\text{exp}}, \theta_{\text{exp}}$	Shared explicit textual decoder and its parameters
$\mathbf{Y}_{\text{prev}}$	Human-readable textual preview
$\mathbf{P}_{\text{rcv}} \in \mathbb{R}^{(n \times k) \times d_{\text{rcv}}}$	Soft-prompt prefix on the receiver
$g_{\text{dyn}}^{(\text{rcv})}, \theta_{\text{dyn}}$	Receiver-side hypernetwork for LoRA synthesis and its parameters
$U_{\text{rcv}}^{(l,m)}, V_{\text{rcv}}^{(l,m)}$	Dynamic LoRA factors at layer l and module m
r	Rank of the dynamic LoRA
$\Delta W_{\text{rcv}}^{(l,m)}$	Synthesized parameter offset
θ_{rcv}	Receiver-side adapter parameters
$\mathcal{L}_{\text{task}}$	Receiver-side language modeling loss
$\mathcal{L}_{\text{exp}}$	Cross-entropy loss for the explicit stream
$\mathcal{L}_{\text{rec}}$	Latent reconstruction loss
$\lambda_{\text{exp}}, \lambda_{\text{rec}}$	Loss-balancing coefficients

C DATASET STATISTICS

We evaluate UniCom on five public benchmarks spanning multi-disciplinary knowledge, mathematical reasoning, and program synthesis, whose evaluation splits and scoring metrics are summarized in Table 5, with the details of each benchmark and its evaluation protocol described below.

Table 5: Evaluation splits and scoring metrics for the five benchmarks. MCQ denotes multiple-choice answer selection, and Open denotes free-form answers such as numbers or expressions.

Benchmark	#Eval	Format	Metric
MMLU	336	MCQ	Accuracy
C-Eval	239	MCQ	Accuracy
GSM8K	500	Open	Exact match
MATH-500	500	Open	Exact match
HumanEval	164	Code	pass@1

MMLU (Hendrycks et al., 2020) aggregates 57 academic subjects ranging from the humanities to STEM. We evaluate on 336 held-out questions that span all 57 subjects, and score each item as a four-way multiple-choice classification over the options A, B, C, and D.

C-Eval (Huang et al., 2023) is the Chinese counterpart of MMLU, covering 52 subjects with culturally grounded knowledge, and we evaluate on 239 held-out questions spanning all of these subjects.

GSM8K (Cobbe et al., 2021) is a corpus of grade-school math word problems that require multi-step arithmetic reasoning, and we evaluate on 500 problems randomly sampled from its official test split.

MATH (Hendrycks et al., 2021) contains competition-level mathematics problems with structured solutions. We use the canonical MATH-500 subset for evaluation and extract the final answer from the boxed expression in each output, which is scored by exact match against the reference answer.

HumanEval (Chen et al., 2021) comprises 164 hand-written Python programming problems evaluated by unit tests, with correctness defined as passing every test case. Given the limited evaluation size, we train the communication adapters on MBPP (Austin et al., 2021), a similar Python program synthesis dataset, rather than carving out a training split from HumanEval itself.

D Baselines

We benchmark UniCom against representative protocols spanning the three families introduced in Section 2, namely no, explicit textual, and implicit latent communication.

NoCom. NoCom denotes the no-communication reference, where the receiver solves each task in isolation without any input from the upstream senders. It anchors the standalone task-solving capacity of the receiver and exposes the marginal value of any subsequent communication protocol.

TextCom (Du et al., 2024; Zhang et al., 2025). TextCom captures the de facto explicit paradigm, wherein each sender greedily decodes a textual rationale and the receiver consumes the concatenated rationales as additional context. To remove decoding stochasticity across runs, these rationales are pre-generated once and cached, and the same rationales are also provided to UniCom.

Cache-to-Cache (Fu et al., 2025). Cache-to-Cache moves the exchange into latent space by transmitting the KV cache of each sender to the receiver, where a learnable projector maps the cache into the representation space of the receiver and fuses it with the receiver's own cache.

ThoughtComm (Zheng et al., 2025). ThoughtComm formulates communication as a latent-variable problem. Agents are assumed to share a space of identifiable thoughts, which encoder-decoder modules recover from frozen hidden states and inject into the receiver through a learnable prefix.

Interlat (Du et al., 2025). Interlat transmits the last-layer hidden states of each sender as latent messages, which a lightweight adapter maps into the input embedding space of the receiver before they are inserted into its input sequence, bypassing explicit textual decoding.

E Implementation Details

Architecture. Each sender emits $k = 32$ latent tokens per message, which are mapped into the shared communication space by a lightweight MLP encoder f_i, while the receiver-side projection f_{rcv}^{-1} maps the fused payload back into the hidden space of the receiver. The explicit decoder g_{exp} is

instantiated as a small language model that decodes the fused latent payload autoregressively, and the hypernetwork $g_{\text{dyn}}^{(\text{rcv})}$ adopts a bottleneck linear architecture to synthesize rank-8 LoRA factors for the attention projections of the receiver at every Transformer layer.

Training. All LLM backbones remain frozen during adapter training, and only the communication adapters, namely $\{f_i, f_i^{-1}\}$, g_{exp}, and $g_{\text{dyn}}^{(\text{rcv})}$, are jointly optimized with AdamW (Loshchilov & Hutter, 2017) under a cosine learning rate schedule. The textual rationales of the senders are pre-generated and cached, which amortizes their generation cost across training epochs and ensures that TextCom and UniCom condition on identical rationales in both training and evaluation.

Evaluation. At inference, the receiver consumes the latent payload through the dual-stream reception mechanism, while the explicit decoder is activated only when textual previews are needed for inspection. We evaluate on the splits described in Appendix C and align the maximum generation length of the receiver with the typical solution length of each task to avoid truncating answers.

F Diverse Multi-Agent Configurations

This appendix supplements the diverse-configuration analysis in Section 4.3, with the backbone assignments of the four sender-receiver configurations listed in Table 6. The two homogeneous regimes (**S→S** and **L→L**) reuse the small- and large-scale setups of the main experiments, while the two mixed regimes probe asymmetric coordination, wherein **L→S** pairs capable senders with a constrained receiver to simulate powerful experts feeding a lightweight coordinator, and **S→L** pairs weak senders with a stronger receiver to simulate limited peripheral agents informing a capable hub.

Table 6: Backbone assignments for the four sender-receiver configurations.

Config.	Sender A	Sender B	Receiver
S→S	Qwen2.5-1.5B-Instruct	Llama-3.2-1B-Instruct	Qwen3-0.6B
L→S	Qwen2.5-7B-Instruct	Llama-3.1-8B-Instruct	Qwen3-0.6B
S→L	Qwen2.5-1.5B-Instruct	Llama-3.2-1B-Instruct	Qwen3-4B-Instruct-2507
L→L	Qwen2.5-7B-Instruct	Llama-3.1-8B-Instruct	Qwen3-4B-Instruct-2507

Table 7 reports the MMLU results under the same four configurations, complementing the MATH results in Table 2 of the main text. The trend is consistent across benchmarks, as UniCom leads in every configuration and the margin is most pronounced in **L→S** (+2.1 over the best baseline).

Table 7: Accuracy on MMLU under the four sender-receiver configurations in Section 4.3.

Method	S→S	L→S	S→L	L→L
TextCom	36.6	53.0	69.6	70.8
Cache-to-Cache	36.9	50.3	68.8	69.9
ThoughtComm	35.1	51.5	69.3	70.5
Interlat	37.5	52.4	69.6	70.8
UniCom	**38.4**	**55.1**	**70.8**	**72.0**

G Algorithm

We summarize the procedure of UniCom in algorithmic form, with the notation following Section 3.2. Algorithm 1 details a single communication round at inference, covering sender-side latent reasoning, projection onto the shared substrate, and the dual-stream reception at the receiver, while Algorithm 2 outlines the joint optimization of the communication adapters with frozen backbones.

Algorithm 1 Communication round of UniCom at inference

Input: Task $\mathcal{T}$, sender contexts $\{C_i\}_{i=1}^n$, receiver context C_{rcv}, frozen backbones $\{\Theta_i\}_{i=1}^n$ and Θ_{rcv}, adapters $\{f_i\}_{i=1}^n$, f_{rcv}^{-1}, g_{exp}, and $g_{\text{dyn}}^{(\text{rcv})}$, latent length k
Output: Receiver output y and optional textual preview $\mathbf{Y}_{\text{prev}}$

for each sender $A_i \in \mathcal{S}$ **do** ▷ message transmission
  $\mathbf{C}_0^{(i)} \leftarrow h_{\text{emb}}^{(i)}(C_i, \mathcal{T})$
  for $t = 1, \ldots, k$ **do**
    $\mathbf{h}_t^{(i)} \leftarrow h_{\text{trans}}^{(i)}\big(\mathbf{C}_{t-1}^{(i)}; \Theta_i\big)$ ▷ latent reasoning step
    $\mathbf{C}_t^{(i)} \leftarrow \mathbf{C}_{t-1}^{(i)} \oplus \mathbf{h}_t^{(i)}$
  end for
  $\mathbf{M}_i \leftarrow [\mathbf{h}_1^{(i)}, \ldots, \mathbf{h}_k^{(i)}]$
  $\mathbf{Z}_i \leftarrow f_i(\mathbf{M}_i)$ ▷ projection onto shared substrate
end for
$\mathbf{Z} \leftarrow [\mathbf{Z}_1, \ldots, \mathbf{Z}_n]$ ▷ message reception
if a textual preview is requested **then**
  $\mathbf{Y}_{\text{prev}} \leftarrow g_{\text{exp}}(\mathbf{Z}; \theta_{\text{exp}})$ ▷ explicit stream
end if
$\mathbf{P}_{\text{rcv}} \leftarrow f_{\text{rcv}}^{-1}(\mathbf{Z})$ ▷ instant prefix adaptation
for each layer l and target module m of the receiver **do**
  $\big(U_{\text{rcv}}^{(l,m)}, V_{\text{rcv}}^{(l,m)}\big) \leftarrow g_{\text{dyn}}^{(\text{rcv})}\big(\mathbf{Z}, \mathbf{e}_l, \mathbf{e}_m; \theta_{\text{dyn}}\big)$
  $\Delta W_{\text{rcv}}^{(l,m)} \leftarrow U_{\text{rcv}}^{(l,m)} V_{\text{rcv}}^{(l,m)}$ ▷ dynamic parameter adaptation
end for
$y \leftarrow h_{\text{lmh}}^{(\text{rcv})}\Big(h_{\text{trans}}^{(\text{rcv})}\big(\mathbf{P}_{\text{rcv}} \oplus h_{\text{emb}}^{(\text{rcv})}(C_{\text{rcv}}); \Theta_{\text{rcv}} + \Delta W_{\text{rcv}}\big)\Big)$ ▷ implicit stream
return y and $\mathbf{Y}_{\text{prev}}$

Algorithm 2 Training of UniCom

Input: Training set $\mathcal{D}$ of tasks with reference answers $y^\star$ and sender rationales R, senders initialized via implicit reasoning training, balancing coefficients λ_{exp} and λ_{rec}
Output: Trained communication adapters θ_{exp}, $\{f_i, f_i^{-1}\}_{i=1}^n$, f_{rcv}^{-1}, and θ_{dyn}

Freeze all backbone parameters $\{\Theta_i\}_{i=1}^n$ and Θ_{rcv}
repeat
  Sample $\big(\mathcal{T}, \{C_i\}_{i=1}^n, C_{\text{rcv}}, y^\star, R\big)$ from $\mathcal{D}$
  Compute $\{\mathbf{M}_i\}$, $\{\mathbf{Z}_i\}$, $\mathbf{Z}$, $\mathbf{P}_{\text{rcv}}$, and ΔW_{rcv} following Algorithm 1
  $\mathcal{L}_{\text{task}} \leftarrow -\log p\big(y^\star \mid \mathbf{P}_{\text{rcv}}, C_{\text{rcv}}; \Theta_{\text{rcv}} + \Delta W_{\text{rcv}}\big)$ ▷ task loss
  $\mathcal{L}_{\text{exp}} \leftarrow -\log p\big(R \mid \mathbf{Z}; \theta_{\text{exp}}\big)$ ▷ explicit stream loss
  $\mathcal{L}_{\text{rec}} \leftarrow \sum_{i=1}^n \text{MSE}\big(f_i^{-1}(\mathbf{Z}_i), \mathbf{M}_i\big)$ ▷ latent reconstruction loss
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{task}} + \lambda_{\text{exp}}\mathcal{L}_{\text{exp}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}}$
  Update θ_{exp}, $\{f_i, f_i^{-1}\}_{i=1}^n$, f_{rcv}^{-1}, and θ_{dyn} with AdamW on $\mathcal{L}$
until convergence

H Case Study

To complement the aggregate metrics with a qualitative view, we contrast TextCom, which uses the explicit pathway in isolation, with UniCom on two real exchanges drawn from the L→L evaluation. Both cases involve the same sender pool, and the textual rationales of the senders are identical for the two protocols, so that the only difference lies in whether the receiver is also given the latent payload. The first case exhibits a subtle task constraint that the textual rationale strips away, whereas the second exhibits a structural specification that the text under-specifies.

Case 1 A constraint that text strips away (MATH-500, Counting & Probability)

Query. In how many ways can 8 people be seated around a square table with 2 people on a side? (Two configurations are equivalent if one is a rotation of another.)

Senders' rationale. *"Fix one person in position to break the rotational symmetry, then permute the remaining 7."*

TextCom ✗	UniCom ✓
Answer: 5040 (7!, treating the table as round)	Answer: **10080** (8!/4, the gold answer)

Analysis. The textual hint of the senders reduces to the standard round-table trick of fixing one seat and dividing by the number of seats, which yields $7! = 5040$ and is exactly what TextCom outputs. The geometry of a *square* table, however, admits only 4 rotational symmetries rather than 8, a constraint that neither sender verbalizes. Conditioned on the latent payload, the UniCom receiver recovers this constraint and writes "a square has 4 rotational symmetries, so the count is $8!/4 = 10080$", indicating that the constraint survives in the latent channel even though both textual rationales point to the wrong rule.

Case 2 A specification that text under-specifies (HumanEval, problem 58)

Query. Return the sorted unique common elements of two lists `l1` and `l2` in the function `common(l1, l2)`.

Senders' rationale. *"Use set intersection on* `l1` *and* `l2`*, then sort the result before returning."*

TextCom ✗	UniCom ✓
`set(l1) & set(l2)` (bare expression without `return`, sorting, or indentation, failing the unit tests)	`return sorted(list(set(l1) & set(l2)))` (passing all unit tests)

Analysis. Both senders state in plain text that the result should be sorted and returned. TextCom nonetheless outputs a bare set expression that drops the sorting step, the `return` statement, and the four-space indentation that places the body inside the function. With the same textual context plus the latent payload, UniCom recovers all three structural commitments simultaneously, suggesting that the latent channel restores format and specification details that the textual rationales of the senders leave implicit.

Together, the two cases trace the qualitative advantage of UniCom to a consistent mechanism. The hidden states of the senders encode information that the senders *possess* but their textual rationales do not fully *verbalize*, whether it is a subtle structural property of the problem (Case 1) or the surface-level specification of the expected output (Case 2). The latent stream preserves this information and the dual-stream reception delivers it to the receiver, while the textual preview $\mathbf{Y}_{\text{prev}}$ from g_{exp} remains available as an auditable summary of what the latent channel transports.

I Prompt Templates

We present the prompt templates used in our experiments. Within each task, both senders share the same sender prompt, which elicits a short textual rationale under strict rules that forbid revealing the final answer, so that the senders provide guidance rather than solutions. The receiver prompt is shared by TextCom and UniCom, and the two protocols differ only in whether the latent slot is filled with the payload $\mathbf{P}_{\text{rcv}}$. For C-Eval, we adopt the Chinese counterparts of the MMLU templates.

Receiver input format (shared across tasks)

```
[System]
<task-specific receiver system prompt>

[User]
Latent messages from the advisors: <latent payload>    # UniCom only

<task text>

Insight from advisor A:
```

```
<textual rationale of sender A>

Insight from advisor B:
<textual rationale of sender B>
```

MMLU

```
[Sender System]
You are a knowledgeable advisor for multiple-choice exam questions. Given a question
   with four choices, produce a SHORT high-level insight to help another model pick
   the right answer. You MUST NOT state the answer.

[Sender User]
Question:
{question}

Provide a brief insight (1-2 sentences). STRICT RULES:
1. Do NOT name the choice letter or repeat a choice text as your answer.
2. Describe ONLY the relevant concept, fact, or distinction.
3. Keep it under 80 tokens.

Insight:

[Receiver System]
You are a careful exam assistant. You may see brief insights from advisor models.
   The insights describe relevant concepts but do not state the answer. Given a
   multiple-choice question, output ONLY the single letter (A, B, C, or D) of the
   correct answer. Do not explain, do not reason out loud, do not add any other
   text.
```

GSM8K

```
[Sender System]
You are a math reasoning advisor for grade-school word problems. Given a problem,
   you produce a SHORT high-level insight to help another model solve it. You MUST
   NOT do any arithmetic yourself.

[Sender User]
Problem:
{problem}

Provide a short approach hint (1-3 sentences). STRICT RULES:
1. Do NOT compute the final number and do NOT execute any arithmetic step.
2. Describe ONLY which quantities to identify and the order to combine them.
3. Keep it under 80 tokens.

Approach hint:

[Receiver System]
You are a careful mathematician. You may see high-level approach hints from advisor
   models. The hints describe an approach but do not contain the answer. Solve the
   problem step by step and end with the final numeric answer inside \boxed{}.
```

MATH

```
[Sender System]
You are a mathematics reasoning advisor. You will be shown a math problem. Provide a
   SHORT, HIGH-LEVEL insight that helps another model solve it, WITHOUT performing
   the calculation yourself.

[Sender User]
Problem:
{problem}
```

```
Provide a short approach hint (1-3 sentences). STRICT RULES:
1. Do NOT give the final answer and do NOT execute any arithmetic or symbolic
   computation.
2. Describe ONLY the key idea, applicable theorem, or recommended setup.
3. Keep it under 80 tokens.

Approach hint:

[Receiver System]
You are a careful mathematician. You will be shown a math problem and may also see
   high-level approach hints from advisor models. The hints describe an approach but
   do not contain the answer, so verify every step and compute the answer yourself.
   Solve the problem step by step and end with the final answer inside \boxed{}.
```

HumanEval

```
[Sender System]
You are an expert Python programming advisor. Given a function signature and
   docstring, provide a SHORT high-level outline of the implementation approach
   (algorithm or data-structure choice). You MUST NOT write code.

[Sender User]
Function to implement:
```python
{prompt}```

Outline the approach in 1-2 sentences. STRICT RULES:
1. Do NOT write any code, snippet, or pseudocode.
2. Describe ONLY the algorithmic idea.
3. Keep it under 80 tokens.

Approach outline:

[Receiver System]
You are an expert Python programmer. You may see high-level approach hints from
   advisor models. The hints describe an algorithmic idea but do not contain code.
   Given a programming task, write the complete solution as a single Python code
   block. Include any imports you need and the full function definition.
```
```