

Special Topic: 6G Networks: Large AI Model Technologies and Agentic AI Communication

In-context source and channel coding

Ziqiong WANG¹, Tianqi REN¹, Rongpeng LI^{1*}, Zhifeng ZHAO^{1, 2} & Honggang ZHANG³¹College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China²Zhejiang Lab, Hangzhou 311121, China³Faculty of Innovation Engineering, Macau University of Science and Technology, Macao 999078, China

Received 14 January 2026/Revised 20 May 2026/Accepted 4 August 2026/Published online 3 September 2026

Abstract Separate source-channel coding (SSCC) remains attractive for text transmission due to its modularity and compatibility with mature entropy coders and powerful channel codes. However, SSCC often suffers from a pronounced cliff effect in low signal-to-noise ratio (SNR) regimes, where residual bit errors after channel decoding can catastrophically break lossless source decoding, especially for arithmetic coding (AC) driven by large language models (LLMs). This paper proposes a receiver-side in-context decoding (ICD) framework that enhances SSCC robustness without modifying the transmitter. ICD leverages an error correction code transformer (ECCT) to obtain bit-wise reliability for the decoded information bits. Based on the context-consistent bitstream, ICD constructs a confidence-ranked candidate pool via reliability-guided bit flipping, samples a compact yet diverse subset of candidates, and applies an LLM-based arithmetic decoder to obtain both reconstructions and sequence-level log-likelihoods. A reliability-likelihood fusion rule then selects the final output. We further provide theoretical guarantees on the stability and convergence of the proposed sampling procedure. Extensive experiments over additive white Gaussian noise (AWGN) and Rayleigh fading channels demonstrate consistent gains compared with conventional SSCC baselines and representative joint source-channel coding (JSCC) schemes.

Keywords SSCC, LLM-based arithmetic coding, ECCT, context-consistent decoding, candidate processing pipeline

Citation Wang Z Q, Ren T Q, Li R P, et al. In-context source and channel coding. *Sci China Inf Sci*, 2026, 69(10): 200310, <https://doi.org/10.1007/s11432-026-5067-6>

1 Introduction

In recent years, semantic communications (SemCom) have attracted growing attention [1–3], driven by the need to deliver task-relevant meaning rather than faithfully transporting raw symbols, and further accelerated by advances in deep learning and generative models [4, 5]. While deep learning-based joint source-channel coding (JSCC) enables end-to-end semantic-aware transceivers with robustness under finite blocklength and time-varying channels [6, 7], separate source-channel coding (SSCC) remains attractive due to its modularity and standard compliance [8]. By decoupling source coding and channel coding, SSCC can directly reuse mature compressors [9] and capacity-approaching channel codes [10], and in principle achieves near-error-free, bit-exact reconstruction at sufficiently high signal-to-noise ratio (SNR). Meanwhile, motivated by the prediction-compression duality [11], modern neural predictors, ranging from recurrent and hybrid compressors [12] to large language model (LLM)-based schemes [13], can benefit the source encoders such as Huffman coding [14] and arithmetic coding (AC) [15], yielding improved transmission efficiency. However, though LLM-based SSCC could bring superior performance than JSCC [8], it suffers from the severe “cliff effect”, whereby the performance degrades sharply in low-SNR regimes. This degradation primarily arises because even a small number of residual bit errors after channel decoding may catastrophically break subsequent source decoding.

There is no doubt that to improve robustness in low-SNR regimes, SSCC can resort to enhancing the channel-coding side [16, 17]. However, under finite blocklength and stringent latency constraints, closing the gap to optimal decoding remains challenging. Learning-aided decoders have been explored, including model-based designs [18] and model-free approaches [19]. Notably, the error correction code transformer (ECCT) [20] has incorporated code-aware inductive biases via parity-check-guided masked self-attention and has achieved error-rate improvements in low-SNR conditions. Nevertheless, such decoders remain constrained by channel observations alone [21], without introducing explicit external information.

* Corresponding author (email: lirongpeng@zju.edu.cn)

On the other hand, leveraging contextual information [22] at the receiver offers a backward-compatible means to enhance robustness without violating existing SSCC architectures. However, directly applying in-context decoding (ICD) to corrupted bitstreams is often insufficient, as early decoding errors can dominate the inference process and lead to unstable or suboptimal reconstructions. To address this limitation, a natural approach is to maintain multiple candidates during decoding [23], perform backtracking [24], or introduce diversity through stochastic sampling [25]. However, exhaustively exploring the candidate space is computationally prohibitive when expensive LLM-based decoding [8] is involved, while unconstrained sampling may generate redundant or low-quality candidates. In principle, both in-context information and channel decoding reliability can contribute to maintaining a compact yet diverse set of plausible candidates for improved decoding performance.

In this paper, we propose in-context source and channel coding for SSCC-based transmission, aiming to mitigate the cliff effect. Particularly, we calibrate a receiver-side ICD framework that augments a standard SSCC pipeline by coupling ECCT-assisted channel decoding with LLM-driven arithmetic source decoding in a lightweight manner. Here, “in-context” refers to the same fundamental conditioning principle as natural language processing (NLP)-style in-context learning, where context changes the model’s conditional output distribution and restricts the plausible output space. Specifically, ECCT provides a bit-wise reliability vector that identifies probabilistically corrupted positions, while in-context information, delivered through previous transmissions and/or other reliable channels, offers a strong before reduce ambiguity in the recovered message. Based on the channel-decoded bitstream, its reliability vector, and the contextual bits, we leverage the in-context candidate generator (CCG) to construct a confidence-ranked candidate pool through reliability-guided bit flipping. Afterward, to balance the computational budget and decoding accuracy, we adopt the in-context candidate sampler (CCS), which selects a sufficiently compact yet highly plausible subset of candidates and their sequence-level log-likelihoods for LLM-based source decoding. Finally, we adopt in-context likelihood ranking (CLR) to select the most plausible reconstruction by jointly accounting for ECCT-derived reliability and linguistic plausibility. Compared to directly decoding a single channel-decoded bitstream as in a conventional ECCT-assisted LLM-AC cascade [8], ICD treats the receiver-side bitstream estimate as an uncertain observation and performs candidate-based decoding by jointly exploiting channel-side reliability, contextual side information, and source-level likelihood. Besides the key distinctions with representative prior studies in Table 1, we summarize our contributions as follows.

- We introduce a practical in-context source and channel coding mechanism that incorporates contextual information into the SSCC receiver and explicitly maintains context consistency via an overwrite-based step, thereby constraining the feasible message space and improving robustness without modifying the transmitter.
- We develop a three-stage candidate processing pipeline by leveraging the ECCT-provided bit-wise reliability. Specifically, we employ CCG to construct a confidence-ranked set of candidate bitstreams via reliability-guided bit flipping. To achieve a favorable accuracy-complexity trade-off, we further apply CCS to select a compact yet diverse subset. Finally, we use CLR to integrate the ECCT-derived reliability with the LLM decoding log-likelihood to determine the final reconstruction. Moreover, we provide theoretical guarantees on the stability and convergence of the proposed CCS module.
- We conduct extensive evaluations and demonstrate the superiority of ICD over conventional SSCC baselines, represented by Huffman-SSCC and ECCT-aided scheme [8], as well as over representative JSCC schemes, including DeepSC [26], universal transformer (UT) [27], and UT with quantization [28].

The remainder of the paper is organized as follows. Section 2 succinctly reviews the key components of our SSCC system. Section 3 briefly introduces the system model and formulates the problem. Section 4 presents an overview of our proposed ICD framework. In Section 5, we elaborate on the experimental results and discussions. Finally, Section 6 concludes the paper. For convenience, we list the major notations of this paper in Table 2.

2 Preliminaries

2.1 LLM-based source coding

We consider LLM-based AC [15] for source coding [8]. For a tokenizer vocabulary $\mathcal{D} = \{D_1, \dots, D_\tau\}$, a pre-trained LLM can map a source sequence $\mathbf{s}$ to a token sequence $\mathbf{t}_{1:N_t}$ with $t_k \in \mathcal{D}$, while providing token-level conditional probabilities $\tilde{p}(t_k | \mathbf{t}_{1:k-1})$ for generation. Using these conditionals, AC maps $\mathbf{t}_{1:N_t}$ to a binary message $\mathbf{m} \in \{0, 1\}^K$. By progressively updating a probability interval using the autoregressive structure of the LLM, the source can eventually be encoded with a minimal number of bits¹⁾.

1) The standard AC interval updates and the corresponding LLM-AC decoding steps can be found in [8].

Table 1 Summary and comparison of related papers.

References	Strengths	Limitations	Relation to ICD
JSCC-based SemCom [26–28]	Enables robust semantic reconstruction and graceful degradation under noisy channels	Requires end-to-end transceiver redesign, limiting SSCC compatibility and bit-exact recovery	ICD improves low-SNR robustness in a modular manner
Source-side coding enhancement for SSCC [12, 14, 15]	Improves transmission efficiency through stronger probabilistic source modeling	Still vulnerable to residual-error-induced cliff effects	ICD complements efficient source coding by stabilizing LLM-AC under channel errors
Channel-side reliability enhancement for SSCC [8, 16–20]	Improves bit-level recovery while preserving the modular SSCC pipeline	Mainly relies on channel observations, leaving LLM-AC sensitive to residual bit errors	ICD extends ECCT outputs from single-stream decoding to reliability-guided candidate exploration
Context- and candidate-aided decoding [22–25]	Exploits receiver-side contextual information and multiple decoding candidates	May suffer from early-error propagation, high search cost, or redundant candidates	ICD enables reliability-guided, in-context candidate processing
This paper	Jointly exploits contextual information, ECCT-derived bit-wise reliability, and LLM likelihood under a limited decoding budget	Requires contextual side information and multiple LLM decoding attempts	Provides a practical robustness enhancement for LLM-based SSCC

Table 2 Major notations used in this paper.

Notation	Definition
$\mathbf{s}_{1:N_s}, N_s$	Input text sequence and its length
$\mathbf{t}_{1:N_t}, N_t$	Token sequence produced by the LLM tokenizer and its length
$\mathbf{m}, \hat{\mathbf{m}}$	Source-coded message and its channel-decoded version
$\mathbf{x}_b, \mathbf{x}_s$	Channel-coded binary codeword and its BPSK-modulated symbol sequence
N, K	Codeword length and message length of channel code
$\mathcal{D} = \{D_1, \dots, D_\tau\}$	Token vocabulary defined by the tokenizer and τ is the vocabulary size
$\tilde{p}(\cdot)$	LLM-induced conditional probability used as the distribution for AC
$\mathbf{G}, \mathbf{H}$	Generator matrix and parity check matrix
h	Channel fading coefficient
$\mathbf{z}, \hat{\mathbf{z}}$	Additive Gaussian noise and ECCT output estimating the multiplicative disturbance
$\sigma_n^2, \mathbf{I}$	Noise variance and identity matrix in the Gaussian noise model
$\mathbf{y}$	Received real-valued channel output vector
$\tilde{\mathbf{y}}$	ECCT input feature vector (i.e., concatenation of magnitude and syndrome features of $\mathbf{y}$)
f_k	Bit-flip indicator when constructing the candidate $\hat{\mathbf{m}}$
$\boldsymbol{\rho}$	Normalized bit-wise reliability vector obtained by applying sigmoid to $\hat{\mathbf{z}}$
$\boldsymbol{\rho}_m$	Bit-wise reliability vector for the information bits
$\tilde{\boldsymbol{\rho}}_m$	Candidate-specific reliability associated with $\hat{\mathbf{m}}$
ℓ	Sequence-level log-likelihood of the LLM-decoded candidate
$\hat{\mathbf{x}}_s, \hat{\mathbf{x}}_b$	ECCT-processed soft symbol estimate and hard-decision demodulated estimate
$\mathbf{m}_{\text{pre}}$	Contextual information bit sequence
$\tilde{\mathbf{m}}$	Candidate bitstream generated by applying a specific bit-flip pattern to $\hat{\mathbf{m}}$
$\tilde{\mathbf{s}}$	Reconstructed text sequence by the LLM-based source decoder from candidate bitstream $\tilde{\mathbf{m}}$
$\tilde{\mathbf{s}}^*$	Final reconstructed text sequence
$\mathcal{M}_{L_c}, L_c$	Candidate set retained by CCG and its size
$\mathcal{M}_{L_s}, L_s$	Subset sampled by CCS and its size
Ω	State space consisting of all subsets of $\mathcal{M}_{L_c}$ with fixed cardinality L_s
$\mathcal{S}^{(t)}$	State of the Markov chain at iteration t , represented as a subset of $\mathcal{M}_{L_c}$
$E(\cdot)$	Energy function evaluating the quality of a subset in terms of reliability and diversity
$\pi(\cdot)$	Target probability distribution over the state space Ω

At the receiver, given a channel-decoded bitstream estimate $\hat{\mathbf{m}}$ and the same LLM, an arithmetic decoder gradually locates the most appropriate interval corresponding to a token sequence $\tilde{\mathbf{t}}$ and deterministically recovers the reconstructed text $\tilde{\mathbf{s}}$. Beyond producing the reconstruction, the LLM also yields a sequence-level plausibility for the decoded sequence by accumulating token log-probabilities along the decoding path

$$\ell(\tilde{\mathbf{t}}) \triangleq \sum_{k=1}^{|\tilde{\mathbf{t}}|} \log \tilde{p}(\tilde{t}_k | \tilde{\mathbf{t}}_{1:k-1}). \quad (1)$$

For convenience, we define the transmitter-side LLM-driven arithmetic encoding operation by $\mathcal{F}_{\text{LLM}_s}(\cdot)$ and the receiver-side arithmetic decoding and likelihood evaluation operation by $\mathcal{F}_{\text{LLM}_r}(\cdot)$.

2.2 Error correction code transformer

We adopt the ECCT²⁾ [20] as a receiver-side reliability estimation module to quantify bit-wise confidence under severe noise. Without loss of generality, assume that for a binary phase-shift keying (BPSK)-coded sequence $\mathbf{x}_s \in \{\pm 1\}^N$, the corresponding channel output can be written as

$$\mathbf{y} = h\mathbf{x}_s + \mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I}), \quad (2)$$

where h is the additive white Gaussian noise (AWGN) or Rayleigh fading coefficient, and $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I})$ represents additive Gaussian noise. Following [20], the channel observation can also be expressed in a multiplicative form $\mathbf{y} = \mathbf{x}_s \odot \tilde{\mathbf{z}}$, where $\tilde{\mathbf{z}} \in \mathbb{R}^N$ denotes an equivalent multiplicative disturbance.

We consider a binary linear block code of length N and dimension K characterized by a parity-check matrix $\mathbf{H} \in \{0, 1\}^{(N-K) \times N}$ (e.g., a low-density parity-check (LDPC) code). Then, given the real-valued channel output $\mathbf{y} \in \mathbb{R}^N$, ECCT exploits the code constraints $\mathbf{H}$ via the binary syndrome computed from a hard-decision mapping. Specifically, define the syndrome vector as

$$\text{syn}(\mathbf{y}) \triangleq \mathbf{H} \text{sign_to_bin}(\mathbf{y}) \pmod{2} \in \{0, 1\}^{N-K}, \quad (3)$$

where $\text{sign_to_bin}(\mathbf{y}) \triangleq \frac{1}{2}(\mathbf{1} - \text{sign}(\mathbf{y})) \in \{0, 1\}^N$ while $\text{syn}(\mathbf{y}) = \mathbf{0}$ indicates that the hard-decision codeword satisfies all parity checks. To jointly incorporate soft channel observations and code-structure information, ECCT forms an augmented input feature vector

$$\tilde{\mathbf{y}} \triangleq [\mathbf{y}, \text{syn}(\mathbf{y})] \in \mathbb{R}^{2N-K}. \quad (4)$$

ECCT processes $\tilde{\mathbf{y}}$ and outputs an estimate of the disturbance

$$\hat{\mathbf{z}} = \mathcal{F}_{\text{ECCT}}(\tilde{\mathbf{y}}), \quad (5)$$

which implicitly captures position-dependent decoding uncertainty while respecting parity-check consistency. On this basis, ECCT computes a recovered BPSK-coded sequence by

$$\hat{\mathbf{x}}_s = \text{sign_to_bin}(\mathbf{y} \odot \hat{\mathbf{z}}) \in \{0, 1\}^N. \quad (6)$$

In this work, we further convert $\hat{\mathbf{z}}$ into a bit-wise reliability vector

$$\boldsymbol{\rho} = \sigma(\hat{\mathbf{z}}) \in [0, 1]^N, \quad (7)$$

where a larger ρ_d indicates higher confidence on the accuracy of the d -th decoded bit. The reliability $\boldsymbol{\rho}$ is used as the common input to the subsequent ICD modules to guide candidate construction and selection.

3 System model and problem formulation

3.1 System model

We consider an SSCC-based text transmission framework in which the receiver is augmented with ECCT-assisted channel decoding and an LLM-based source decoder equipped with ICD, which comprises CCG, CCS, and CLR, in addition to conventional source and channel coding, modulation, and demodulation. Figure 1 shows the corresponding system block diagram.

For the input text sequence $\mathbf{s}_{1:N_s}$, a lossless source encoder compresses it into a binary message

$$\mathbf{m} = \mathcal{F}_{\text{LLM}_s}(\mathbf{s}_{1:N_s}) \in \{0, 1\}^K, \quad (8)$$

where as mentioned in Subsection 2.1, $\mathcal{F}_{\text{LLM}_s}(\cdot)$ denotes a LLM-compatible source encoder that maps text into a length- K bitstream through tokenization and probability-driven AC. The same pipeline can also be applied to other discrete sources (e.g., images) after appropriate lossless formatting or compression.

2) Complete architectural and training details of ECCT can be found in [20].

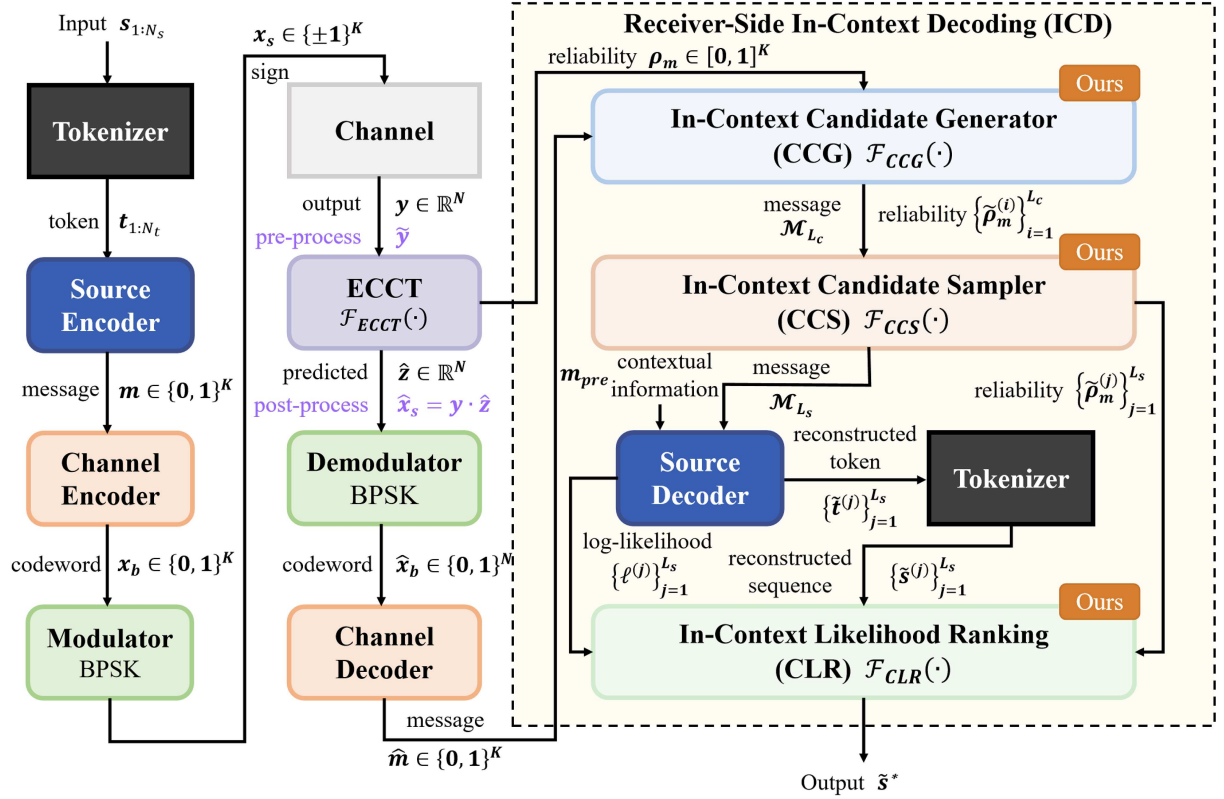


Figure 1 (Color online) Framework of the proposed ICD-aided SSCC system.

The message $\mathbf{m}$ is then protected by an (N, K) LDPC channel code as

$$\mathbf{x}_b = \mathbf{m}\mathbf{G} \pmod{2} \in \{0, 1\}^N, \quad (9)$$

where $\mathbf{G} \in \{0, 1\}^{K \times N}$ is the generator matrix. The corresponding parity-check matrix $\mathbf{H} \in \{0, 1\}^{(N-K) \times N}$ satisfies

$$\mathbf{G}\mathbf{H}^\top = \mathbf{0}. \quad (10)$$

Notably, other error correction codes such as Polar codes [17] can be applied as well [20]. The binary codeword $\mathbf{x}_b$ is modulated via BPSK into $\mathbf{x}_s = 1 - 2\mathbf{x}_b \in \{\pm 1\}^N$ and transmitted over the channel, as defined in (2).

At the receiver, ECCT takes the real-valued channel output $\mathbf{y}$ and the code constraints as input, and produces a recovered BPSK-coded sequence $\hat{\mathbf{x}}_s$, an estimate of disturbance $\hat{\mathbf{z}}$, and a bit-wise reliability vector $\boldsymbol{\rho} \in [0, 1]^N$ as described in Subsection 2.2. Hence, hard-decision BPSK demodulation yields the binary codeword estimate

$$\hat{\mathbf{x}}_b = \text{sign_to_bin}(\hat{\mathbf{x}}_s) \in \{0, 1\}^N. \quad (11)$$

In this work, we adopt a systematic channel encoder, such that the information bits occupy the first K positions of the codeword. Since ECCT operates as a channel decoder producing an estimated codeword, recovering the information bitstream reduces to extracting the first K bits

$$\hat{\mathbf{m}} = \hat{\mathbf{x}}_{b,1:K} \in \{0, 1\}^K. \quad (12)$$

Similarly, since the ECCT-derived reliability $\boldsymbol{\rho} \in [0, 1]^N$ is defined at the codeword-bit level, we take its subvector aligned with the information positions

$$\boldsymbol{\rho}_m = \boldsymbol{\rho}_{1:K} \in [0, 1]^K. \quad (13)$$

Given the channel decoding result $\hat{\mathbf{m}}$, as mentioned in Subsection 2.1, an LLM-based source decoder $\mathcal{F}_{LLM_r}(\cdot)$ can eventually produce a reconstructed sequence $\tilde{\mathbf{s}}$, along with its linguistic log-likelihood ℓ :

$$(\tilde{\mathbf{s}}, \ell) = \mathcal{F}_{LLM_r}(\hat{\mathbf{m}}). \quad (14)$$

3.2 Problem formulation

Conventional SSCC systems often suffer from a *cliff effect* in low-SNR scenarios, where crossing below a specific SNR threshold triggers an abrupt rise in the bit error rate and a severe drop in reconstruction fidelity. The primary reason lies in the fact that an incorrect early bit can shift the LLM-based decoder to a wrong sub-interval for arithmetic decoding, which then perturbs subsequent token boundary decisions and triggers error propagation across many tokens. Conceptually, LLM-driven AC performs a strictly sequential interval-localization process. Starting from the initial interval $[0, 1)$ and the beginning-of-sequence (BOS) condition, the LLM predicts the distribution of the first token and partitions the current interval into token-specific sub-intervals. The received bitstream selects a sub-interval, and the associated token is appended to the prefix for subsequent LLM prediction. Since later interval partitions depend on the decoded prefix, a residual bit error may shift the binary value across a sub-interval boundary and induce a wrong token decision. This erroneous prefix then causes the decoder trajectory to deviate from the encoder-side partitioning process. Consequently, even a small number of residual bit errors after channel decoding may catastrophically break lossless source decoding.

To improve reliability under such unfavorable conditions, we consider receiver-side enhancements that utilize additional contextual information $\mathbf{m}_{\text{pre}}$, which has been obtained through previous transmissions and/or other reliable channels. In practical wireless deployments, such contextual information can be obtained through temporal caching or heterogeneous transmission. For continuous transmission sessions, $\mathbf{m}_{\text{pre}}$ may consist of previously transmitted segments that have already been verified by the receiver's cyclic redundancy check (CRC) and stored in a local cache. Alternatively, in mission-critical scenarios, critical metadata or contextual information can be transmitted via an ultra-reliable low-latency communication (URLLC)-style reliable channel [29–31] to ensure highly reliable delivery. To quantify the overhead, we define the raw-text-level context ratio as

$$\eta_{\text{raw}} = \frac{N_{\text{pre}}}{N_s}, \quad (15)$$

where N_{pre} denotes the number of words used as contextual information and N_s is the length of the original text sequence. More clearly, we argue that the in-context information $\mathbf{m}_{\text{pre}}$ can provide a strong probabilistic prior that helps combat the severe noise. In other words, if we can construct multiple candidate bitstreams $\tilde{\mathbf{m}}$ from the channel decoding result $\hat{\mathbf{m}}$, it will benefit the receiver by evaluating the contextual plausibility (i.e., log-likelihood) ℓ of one bitstream and determining the superior one. To support such an intuition, a confidence-ranked candidate set shall be computed from the available contextual information

$$\left(\mathcal{M}_{L_c}, \{\tilde{\rho}_{\mathbf{m}}^{(i)}\}_{i=1}^{L_c}\right) = \mathcal{F}_{\text{CCG}}(\hat{\mathbf{m}}, \rho_{\mathbf{m}}; L_c), \quad (16)$$

where $\mathcal{F}_{\text{CCG}}(\cdot)$ denotes a CCG function, while $\mathcal{M}_{L_c} = \{\tilde{\mathbf{m}}^{(i)}\}_{i=1}^{L_c}$ denotes the top- L_c candidate bitstreams retained according to the ECCT-derived bit-wise reliability.

However, if all candidates go through the LLM, it will incur substantial computational cost. To better balance cost and diversity, it becomes appealing to design a sampling module $\mathcal{F}_{\text{CCS}}(\cdot)$ that selects L_s candidates from $\mathcal{M}_{L_c}$:

$$\left(\mathcal{M}_{L_s}, \{\tilde{\rho}_{\mathbf{m}}^{(j)}\}_{j=1}^{L_s}\right) = \mathcal{F}_{\text{CCS}}\left(\mathcal{M}_{L_c}, \{\tilde{\rho}_{\mathbf{m}}^{(i)}\}_{i=1}^{L_c}; L_s\right), \quad L_s \leq L_c - 2 \quad (17)$$

with $\mathcal{M}_{L_s} = \{\tilde{\mathbf{m}}^{(j)}\}_{j=1}^{L_s}$. Each selected candidate is then fed into the LLM-based source decoder $\mathcal{F}_{\text{LLM}_r}(\cdot)$, producing a reconstructed sequence and its log-likelihood

$$\left(\tilde{\mathbf{s}}^{(j)}, \ell^{(j)}\right) = \mathcal{F}_{\text{LLM}_r}\left(\tilde{\mathbf{m}}^{(j)}; \mathbf{m}_{\text{pre}}\right), \quad j = 1, \dots, L_s. \quad (18)$$

Finally, the ECCT-derived bit-wise reliability and linguistic log-likelihood can undergo a CLR $\mathcal{F}_{\text{CLR}}$ to output the final reconstruction

$$\tilde{\mathbf{s}}^* = \mathcal{F}_{\text{CLR}}\left(\{\tilde{\mathbf{s}}^{(j)}\}_{j=1}^{L_s}, \{\ell^{(j)}\}_{j=1}^{L_s}, \{\tilde{\rho}_{\mathbf{m}}^{(j)}\}_{j=1}^{L_s}\right), \quad (19)$$

where $\mathcal{F}_{\text{CLR}}(\cdot)$ implements a generic confidence-likelihood fusion rule and returns the optimal candidate $\tilde{\mathbf{s}}^*$ as the decoding result.

For notational convenience, we denote the decoder by $\Phi(\cdot) = \mathcal{F}_{\text{CLR}} \circ \mathcal{F}_{\text{LLM}_r} \circ \mathcal{F}_{\text{CCS}} \circ \mathcal{F}_{\text{CCG}}(\cdot)$. Our goal is to design a computationally efficient source and channel decoder $\Phi(\cdot)$ that maps the ECCT-assisted decoding output $\rho_{\mathbf{m}}$, the channel decoding result $\hat{\mathbf{m}}$, and the contextual side information $\mathbf{m}_{\text{pre}}$ to the final reconstruction $\hat{\mathbf{s}}^*$:

$$\min \mathbb{E}[d(\mathbf{s}_{1:N_s}, \tilde{\mathbf{s}}^*)] \quad \text{s.t.} \quad \text{Cost}(\mathcal{F}_{\text{CCS}} \circ \mathcal{F}_{\text{LLM}_r}) \leq \mathcal{B}, \quad \tilde{\mathbf{s}}^* = \Phi(\hat{\mathbf{m}}, \rho_{\mathbf{m}}, \mathbf{m}_{\text{pre}}), \quad (20)$$

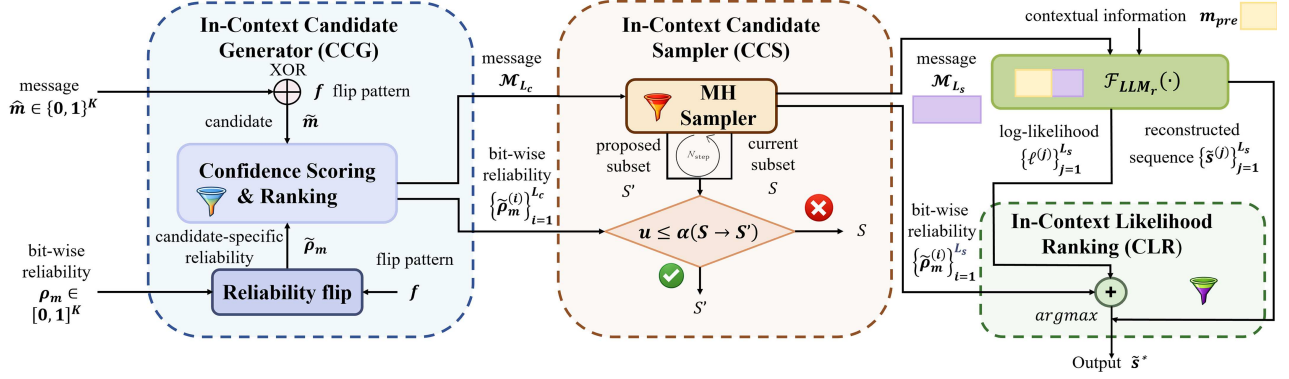


Figure 2 (Color online) Framework of ICD.

where $d(\cdot, \cdot)$ measures the reconstruction mismatch, and $\text{Cost}(F_{\text{CCS}} \circ F_{\text{LLM}_r})$ denotes the computational cost of candidate sampling followed by LLM-based arithmetic source decoding. Specifically, we measure it as

$$\text{Cost}(F_{\text{CCS}} \circ F_{\text{LLM}_r}) = C_{\text{CCS}} + L_s C_{\text{LLM}_r}, \quad (21)$$

where C_{CCS} denotes the cost of the CCS module, L_s is the number of selected candidate bitstreams, and C_{LLM_r} is the average cost of decoding one candidate and computing its sequence-level log-likelihood. Since C_{LLM_r} dominates the lightweight candidate sampling overhead, the budget constraint in (20) can be equivalently interpreted as limiting the number of LLM-based decoding attempts under a given computational budget $\mathcal{B}$. We leave the design of the modules in ICD in Section 4.

4 The framework of ICD

Figure 2 illustrates the overall framework of the proposed ICD.

4.1 In-context candidate generator

The role of CCG is to produce a confidence-ranked candidate set for the information-bit sequence. Given the channel decoding output $\hat{\mathbf{m}} \in \{0,1\}^K$, we construct a candidate set $\mathcal{M} \triangleq \{\tilde{\mathbf{m}}^{(d)}\}_{d=1}^{2^K}$ by enumerating all possible bit-flip patterns. Specifically, let $\mathbf{f}^{(d)} = [f_1^{(d)}, \dots, f_K^{(d)}] \in \{0,1\}^K$ denote the d -th flip pattern, where $f_p^{(d)} = 1$ indicates flipping the p -th bit and $f_p^{(d)} = 0$ otherwise, with $d = 1, \dots, 2^K$. Given $\mathbf{f}^{(d)}$, the corresponding candidate bitstream $\tilde{\mathbf{m}}^{(d)} \in \mathcal{M}$ is constructed entry-wise as

$$\tilde{m}_p^{(d)} = \hat{m}_p \oplus f_p^{(d)}, \quad p = 1, \dots, K, \quad (22)$$

where $\oplus$ denotes the XOR (exclusive OR) operation.

Given the ECCT-derived bit-wise reliability vector $\rho_m = [\rho_{m,1}, \dots, \rho_{m,K}] \in [0,1]^K$, we define the candidate-specific reliability as $\tilde{\rho}_m^{(d)} = [\tilde{\rho}_{m,1}^{(d)}, \dots, \tilde{\rho}_{m,K}^{(d)}]$, whose p -th entry under candidate $\tilde{\mathbf{m}}^{(d)}$ can be computed as

$$\tilde{\rho}_{m,p}^{(d)} = (1 - f_p^{(d)}) \rho_{m,p} + f_p^{(d)} (1 - \rho_{m,p}), \quad p = 1, \dots, K. \quad (23)$$

Accordingly, the aggregate confidence score of candidate $\tilde{\mathbf{m}}^{(d)}$ is defined as

$$\mathcal{F}_{\text{Conf}}(\tilde{\mathbf{m}}^{(d)}) = \sum_{p=1}^K \tilde{\rho}_{m,p}^{(d)}, \quad d = 1, \dots, 2^K. \quad (24)$$

Finally, CCG retains the top- L_c candidates with the largest confidence scores, yielding a confidence-ranked candidate set

$$\mathcal{M}_{L_c} = \left\{ \tilde{\mathbf{m}}^{(i)} \mid i \in \arg \top_{L_c, 1 \leq d \leq 2^K} \mathcal{F}_{\text{Conf}}(\tilde{\mathbf{m}}^{(d)}) \right\}, \quad i = 1, \dots, L_c. \quad (25)$$

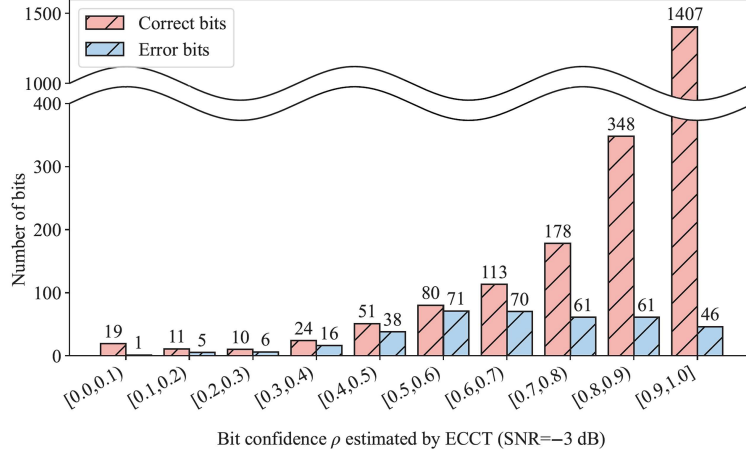


Figure 3 (Color online) Distribution of correct and erroneous bits across confidence levels.

4.2 In-context candidate sampler

CCG leverages the ECCT-derived bit reliability to score candidate bitstreams and efficiently prune an exponentially large flip space, resulting in a confidence-ranked set of top- L_c candidates. While this confidence-guided pruning is essential for computational tractability, the resulting ranking is inevitably affected by reliability miscalibration, as evidenced by Figure 3, which shows that a non-negligible number of erroneous bits persist even in high-confidence regimes. Moreover, candidates with high confidence scores are often generated by flipping similar subsets of low-reliability bits, leading to strong correlations among top-ranked candidates and consequently small pairwise Hamming distances.

As a consequence, directly truncating the ranked list to the top- L_s candidates would produce a highly concentrated and redundant subset, severely limiting the effective exploration space available to the downstream LLM-based decoder. To address this issue, CCS performs diversity-preserving subset selection by explicitly balancing aggregate confidence and inter-candidate Hamming diversity, so that with a limited number of LLM decoding attempts L_s , the selected candidates span multiple plausible error patterns and substantially increase the probability of including a decodable or correct bitstream.

This subset-level trade-off naturally leads to a large search space. For typical values of L_c and L_s , the number of feasible subsets scales as $\binom{L_c}{L_s}$, making exhaustive evaluation costly in practice. This motivates the use of approximate methods that can efficiently explore the subset space without incurring prohibitive computational overhead. To this end, CCS adopts a sampling-based approach inspired by Metropolis-Hastings (MH) [32, 33], which enables efficient exploration of the subset space with a controllable computational budget while naturally introducing randomness to promote diversity.

Specifically, CCS formulates the subset selection problem as sampling from a structured probability distribution defined over subsets of $\mathcal{M}_{L_c}$. To this end, we construct a Markov chain $\{\mathcal{S}^{(t)}\}_{t \geq 0}$ on the state space $\Omega \triangleq \{\mathcal{S} \subseteq \mathcal{M}_{L_c} \mid |\mathcal{S}| = L_s\}$, where each state $\mathcal{S}^{(t)} \in \Omega$ is a candidate subset of fixed cardinality L_s to be forwarded to the subsequent LLM-based decoding stage. Subsequently, we define the target distribution over Ω as

$$\pi(\mathcal{S}) = \exp(-\beta E(\mathcal{S})), \quad \mathcal{S} \in \Omega, \quad (26)$$

where $\beta > 0$ controls the sharpness and the energy $E(\mathcal{S})$ jointly captures ECCT-derived reliability and inter-candidate diversity, given by

$$E(\mathcal{S}) = - \sum_{\tilde{\mathbf{m}} \in \mathcal{S}} \mathcal{F}_{\text{Conf}}(\tilde{\mathbf{m}}) - \lambda \sum_{\substack{\tilde{\mathbf{m}}, \tilde{\mathbf{m}}' \in \mathcal{S} \\ \tilde{\mathbf{m}} \neq \tilde{\mathbf{m}}'}} \mathcal{F}_{\text{Hamming}}(\tilde{\mathbf{m}}, \tilde{\mathbf{m}}'), \quad (27)$$

where $\mathcal{F}_{\text{Conf}}(\cdot)$ denotes the ECCT-derived aggregate confidence while $\mathcal{F}_{\text{Hamming}}(\cdot)$ measures pairwise Hamming distance, and $\lambda \geq 0$ balances reliability preservation and diversity encouragement. This energy function is explicitly tailored to the residual-error characteristics observed after ECCT decoding. While the confidence term $\mathcal{F}_{\text{Conf}}(\cdot)$ ensures alignment with the ECCT-derived bit-wise reliability, the Hamming-diversity term $\mathcal{F}_{\text{Hamming}}(\cdot)$ discourages selecting highly similar flip patterns. As a result, the induced target distribution favors reliable yet diverse candidate subsets, allowing CCS to allocate the limited LLM decoding budget across multiple plausible residual-error patterns.

Given the current state $\mathcal{S} \in \Omega$, we generate a proposal state $\mathcal{S}' \in \Omega$ via a single replacement move $\mathcal{S}' = (\mathcal{S} \setminus \{\tilde{\mathbf{m}}\}) \cup \{\tilde{\mathbf{m}}''\}$, by changing a uniformly sampled $\tilde{\mathbf{m}} \in \mathcal{S}$ to $\tilde{\mathbf{m}}'' \in (\mathcal{M}_{L_c} \setminus \mathcal{S})$, where $\setminus$ denotes the set difference operator. This defines the proposal distribution

$$q(\mathcal{S}'|\mathcal{S}) = \frac{1}{L_s(L_c - L_s)}, \quad (28)$$

and $q(\mathcal{S}'|\mathcal{S}) = q(\mathcal{S}|\mathcal{S}')$ holds due to the symmetry of the replacement move.

The Markov chain transition kernel P is then induced by the MH accept-reject rule

$$P(\mathcal{S}'|\mathcal{S}) = q(\mathcal{S}'|\mathcal{S}) \alpha(\mathcal{S} \rightarrow \mathcal{S}'), \quad \mathcal{S}' \neq \mathcal{S} \quad (29)$$

with acceptance probability

$$\alpha(\mathcal{S} \rightarrow \mathcal{S}') = \min \left\{ 1, \frac{\pi(\mathcal{S}')q(\mathcal{S}|\mathcal{S}')}{\pi(\mathcal{S})q(\mathcal{S}'|\mathcal{S})} \right\} = \min \left\{ 1, \frac{\pi(\mathcal{S}')}{\pi(\mathcal{S})} \right\} = \min \{ 1, \exp(-\beta[E(\mathcal{S}') - E(\mathcal{S})]) \}. \quad (30)$$

A uniform random number $u \sim \mathcal{U}(0, 1)$ is then drawn, and the proposal is accepted if $u \leq \alpha(\mathcal{S} \rightarrow \mathcal{S}')$. Otherwise, the chain remains at $\mathcal{S}$. For completeness, the self-transition probability is

$$P(\mathcal{S}|\mathcal{S}) = 1 - \sum_{\mathcal{S}' \in \Omega, \mathcal{S}' \neq \mathcal{S}} P(\mathcal{S}'|\mathcal{S}). \quad (31)$$

After a fixed number of MH iterations N_{step} , CCS outputs a sampled subset $\mathcal{M}_{L_s}$ of size L_s , which exhibits both high aggregate confidence and sufficient diversity across candidate bitstreams. The computational overhead of CCS is mainly determined by the sampled subset size L_s . In each sampling update, CCS replaces one candidate in the current subset and evaluates the corresponding change in the subset-level energy. Accordingly, the effective CCS overhead scales approximately as $\mathcal{O}(L_s K)$ per update. This overhead is lightweight compared with the LLM-based arithmetic source decoding applied to the L_s sampled candidates. This sampling-enhanced selection avoids excessive concentration on highly similar candidates that arise from correlated low-confidence bit flips, while maintaining tractable computational complexity.

Definition 1 (Stationarity under detailed balance [34]). Let $\{\mathcal{S}^{(t)}\}_{t \geq 0}$ be a Markov chain on a finite state space Ω with transition kernel P . If there exists a distribution π such that the detailed balance condition holds

$$\pi(\mathcal{S}) P(\mathcal{S}'|\mathcal{S}) = \pi(\mathcal{S}') P(\mathcal{S}|\mathcal{S}'), \quad \forall \mathcal{S}, \mathcal{S}' \in \Omega, \quad (32)$$

then π is a stationary distribution of the Markov chain.

Theorem 1 (Properties of CCS). Consider the Markov chain $\{\mathcal{S}^{(t)}\}_{t \geq 0}$ induced by the CCS sampler on the state space Ω , it satisfies the following properties. (1) The state space Ω is finite. (2) The transition kernel P meets the detailed balance condition with respect to the target distribution π . (3) The Markov chain is irreducible and aperiodic on Ω .

The proof follows the standard finite-state Markov-chain argument, and is omitted here.

Lemma 1 (Ergodicity and convergence [35]). Let $\{\mathcal{S}^{(t)}\}_{t \geq 0}$ be a Markov chain on a finite state space with stationary distribution π . If the chain is irreducible and aperiodic, then π is unique and the distribution of $\mathcal{S}^{(t)}$ converges to π in total variation distance, regardless of the initial state.

Theorem 2 (Stationarity and convergence of CCS). Consider the Markov chain $\{\mathcal{S}^{(t)}\}_{t \geq 0}$ induced by the CCS sampler. π is the unique stationary distribution, and for any initial state $\mathcal{S}^{(0)}$, the distribution of $\mathcal{S}^{(t)}$ converges to π in total variation distance, that is

$$\lim_{t \rightarrow \infty} \|P^t(\mathcal{S}^{(0)}, \cdot) - \pi(\cdot)\|_{\text{TV}} = 0. \quad (33)$$

Proof. By Theorem 1, the CCS chain operates on a finite state space and satisfies detailed balance with respect to π . Hence, by Definition 1, π is a stationary distribution of the chain. Moreover, Theorem 1 also establishes that the chain is irreducible and aperiodic. Therefore, by Lemma 1, the stationary distribution π is unique and the distribution of $\mathcal{S}^{(t)}$ converges to π in total variation distance for any initial state $\mathcal{S}^{(0)}$.

Remark 1. Given this CCS-specific target distribution in (26), the theoretical derivations above validate the effectiveness of the proposed sampler in the present decoding setting. Theorem 1 verifies the standard finite-state MH conditions for the CCS-induced chain, while Theorem 2, together with Lemma 1, establishes convergence to the unique stationary distribution induced by the ECCT-derived reliability-diversity energy.

4.3 In-context likelihood ranking

For each retained bitstream candidate $\tilde{\mathbf{m}}^{(j)} \in \mathcal{M}_{L_s}$, CLR integrates ECCT-derived reliability with source-level linguistic plausibility to select the final reconstruction. Formally, CLR computes a fused score for each candidate by combining the ECCT-derived reliability profile $\tilde{\rho}_m^{(j)}$ and the LLM log-likelihood $\ell^{(j)}$, and outputs

$$\tilde{\mathbf{s}}^* = \tilde{\mathbf{s}}^{(j^*)}, \quad j^* = \arg \max_{1 \leq j \leq L_s} \left\{ \mathcal{F}_{\text{Conf}}(\tilde{\mathbf{m}}^{(j)}) + \alpha \ell^{(j)} \right\}. \quad (34)$$

In summary, the pseudocode of the proposed receiver-side ICD procedure is provided in Algorithm 1.

Algorithm 1 The procedure of ICD.

Require: Channel-decoded bitstream $\tilde{\mathbf{m}}$, ECCT-derived reliability ρ_m , contextual information $\mathbf{m}_{\text{pre}}$, candidate pool size L_c , sampled subset size L_s , MH steps N_{step} , temperature β , diversity weight λ , fusion weight α ;

Ensure: Final reconstruction $\tilde{\mathbf{s}}^*$;

1: **Stage I: CCG**

2: $\mathcal{M} \leftarrow$ constructs candidates $\tilde{\mathbf{m}}$ by (22);

3: $F_{\text{Conf}}(\tilde{\mathbf{m}}) \leftarrow$ computes confidence scores by (23)–(24);

4: $\mathcal{M}_{L_c} \leftarrow$ retains top- L_c candidates by (25);

5: **Stage II: CCS**

6: $\mathcal{S}^{(0)} \leftarrow$ initializes a subset in $\Omega = \{\mathcal{S}^{(0)} \subseteq \mathcal{M}_{L_c} \mid |\mathcal{S}^{(0)}| = L_s\}$;

7: **for** $t = 0$ to $N_{\text{step}} - 1$ **do**

8: $\mathcal{S}^{(t)'} \leftarrow$ generates a proposal state by a single-element replacement in $\mathcal{S}^{(t)}$;

9: $\alpha(\mathcal{S}^{(t)} \rightarrow \mathcal{S}^{(t)'}) \leftarrow$ computes the MH acceptance probability by (30);

10: $\mathcal{S}^{(t+1)} \leftarrow$ accepts/rejects $\mathcal{S}^{(t)'}$ according to $u \sim \mathcal{U}(0, 1)$ and $\alpha(\mathcal{S}^{(t)} \rightarrow \mathcal{S}^{(t)'})$;

11: **end for**

12: $\mathcal{M}_{L_s} \leftarrow$ outputs the selected subset;

13: **Stage III: CLR**

14: $(\tilde{\mathbf{s}}^{(j)}, \ell^{(j)}) \leftarrow$ decodes and computes likelihood with contextual information by (18);

15: $\tilde{\mathbf{s}}^* \leftarrow$ selects the final reconstruction by (34).

5 Simulation settings and results

5.1 Simulation settings

We evaluate the proposed receiver-side decoding framework ICD against conventional SSCC pipelines [8] and representative JSCC-based semantic communication schemes [26–28] over both AWGN and Rayleigh fading channels. To ensure a fair and reproducible comparison, all methods are tested on the same text source and are assessed using widely adopted NLP-oriented quality metrics [36, 37].

Dataset and source coding. The adopted dataset is the proceedings of the European Parliament [38], which consists of around 2.0 million sentences. The dataset is pre-processed to have sentence lengths ranging from 4 to 30 words for training and testing. Unless otherwise stated, we adopt GPT-2 base (124.44 M parameters) as the default source model. The arithmetic coder is configured with a precision of 31 bits, which provides a practical trade-off between numerical stability and coding accuracy. For each corpus, some leading words, regarded as the contextual information $\mathbf{m}_{\text{pre}}$, have been correctly decoded. In the current experiments, $\mathbf{m}_{\text{pre}}$ corresponds to approximately the leading half of the source text, i.e., $\eta_{\text{raw}} \approx 0.5$.

Channel coding and ECCT configuration. For the SSCC baselines, we employ an LDPC(49, 24) code, corresponding to a code rate of approximately 0.5. The ECCT architecture and training hyperparameters, as well as those of the JSCC baselines, are summarized in Table 3.

Baselines. The baselines are selected to cover representative text transmission paradigms, including conventional entropy-coding-based SSCC, ECCT-assisted LLM-based SSCC, and JSCC-based semantic communication.

- Huffman with ECCT, where the LLM-driven arithmetic source coding/decoding is replaced by Huffman coding, while keeping the rest of the SSCC pipeline unchanged.

- LLM-AC with ECCT, which combines ECCT-assisted channel decoding with LLM-based lossless source decoding under the SSCC pipeline.

- DeepSC [26] and UT [27], which serve as representative JSCC baselines for comparison.

- BART-JSCC, which employs a pretrained BART [39] encoder-decoder backbone and trainable channel mapping modules for end-to-end semantic transmission.

- UT with quantization [28], where the continuous latent representation is mapped to a fixed-length bitstream (30 bits) for transmission.

Table 3 Main hyperparameter settings used in the experiments.

Model	Hyperparameter	Value
ECCT	Learning rate	10^{-4}
	Batch size	128
	Number of decoder layers	6
	Embedding dimension	32
	Number of attention heads	8
DeepSC	Learning rate	10^{-4}
	Batch size	64
	Number of encoder/decoder layers	4
	Embedding dimension	128
	FFN hidden dimension	512
	Number of attention heads	8
UT	Channel dimension	16
	Learning rate	10^{-4}
	Batch size	64
	Number of encoder/decoder layers	3
	Embedding dimension	128
	FFN hidden dimension ³⁾	1024
BART-JSCC	Number of attention heads	8
	Channel dimension	16
	Learning rate	10^{-4}
	Batch size	8
	Number of encoder/decoder layers	6
	Embedding dimension	768
	FFN hidden dimension	3072
	Number of attention heads	12
	Channel dimension	16

Evaluation metrics. Reconstruction quality is quantified by bilingual evaluation understudy (BLEU) [36] and a bidirectional encoder representations from transformers (BERT)-based semantic similarity metric [37], which are widely used in text reconstruction tasks to assess lexical fidelity and semantic preservation, respectively.

On the other hand, when comparing heterogeneous pipelines (e.g., SSCC vs. JSCC), different schemes generally transmit different amounts of physical-layer payload N for the same source content. Therefore, to maintain consistent energy accounting, consistent with [8], we normalize SNR using a fixed total transmission energy budget. Let E_{total} denote the total energy consumed by sending N_{um} payload units through the channel in an LLM-based SSCC reference system [8], and define the unified SNR according to

$$\text{SNR} = 10 \log \left(\frac{E_{\text{total}}}{N_0 N} \right) = 10 \log \left(\frac{E_{\text{total}}}{N_0 N_{\text{um}}} \right) + 10 \log \left(\frac{N_{\text{um}}}{N} \right) = \text{SNR}_{\text{unified}} + 10 \log \left(\frac{N_{\text{um}}}{N} \right). \quad (35)$$

Here, $\text{SNR}_{\text{unified}}$ serves as the independent variable that aligns E_{total} across different methodologies, while the ratio N_{um}/N compensates for the differing physical-layer payload lengths. Meanwhile, for JSCC methods such as DeepSC [26] and UT [27], the transmitter commonly outputs continuous latent vectors, and the channel input is represented by floating-point values rather than binary bits. To maintain consistent energy accounting, we treat one transmitted float as consuming the energy of multiple bits. With float16, one float corresponds to 16 bits, resulting in an additional offset of $10 \log(16) \approx 12.041$ dB. Accordingly, the unified evaluation metric is given by

$$\text{SNR} = \begin{cases} \text{SNR}_{\text{unified}} + 10 \log \left(\frac{N_{\text{um}}}{N} \right) + 12.041, & \text{float-based JSCC (float16),} \\ \text{SNR}_{\text{unified}} + 10 \log \left(\frac{N_{\text{um}}}{N} \right), & \text{bit-oriented transmission.} \end{cases} \quad (36)$$

5.2 Overall system-level performance comparison

Figures 4 and 5 present the system-level overall performance comparison between the proposed method and multiple baselines over AWGN and Rayleigh channels, respectively, evaluated by BLEU-1, BLEU-4, and semantic similarity. The results demonstrate that the proposed receiver-side ICD consistently achieves the best performance in both

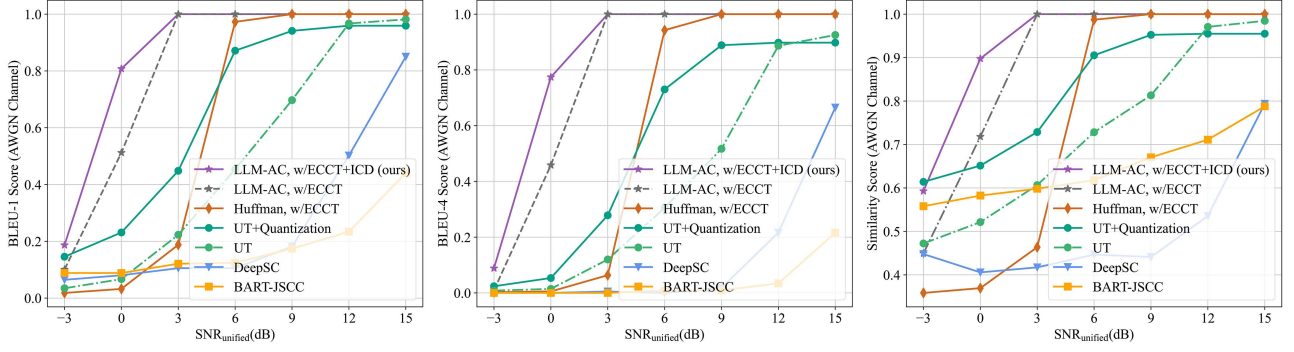


Figure 4 (Color online) Overall system-level performance under AWGN channels.

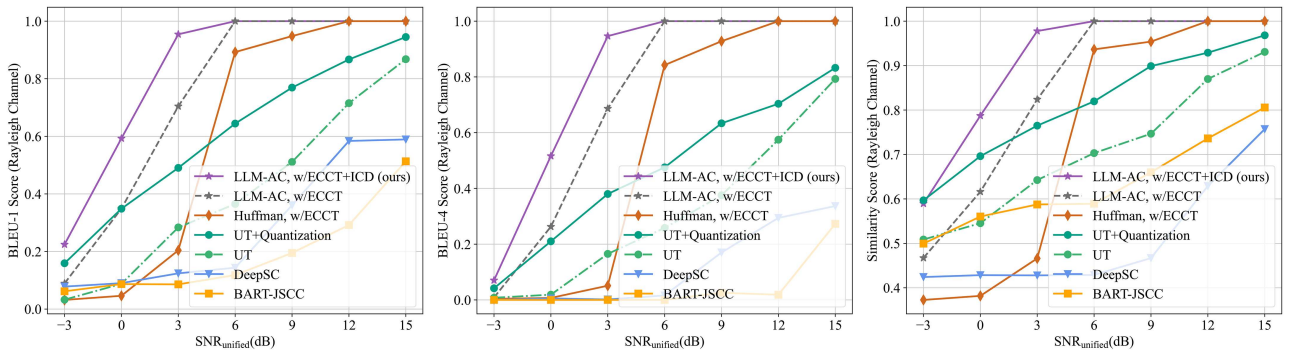


Figure 5 (Color online) Overall system-level performance under Rayleigh channels.

channel environments, outperforming all baselines. The performance improvement is most evident in the low-SNR regime, where residual bit errors after channel decoding can trigger interval misselection in LLM-driven arithmetic decoding, resulting in a pronounced performance cliff. In this regard, ICD leverages the contextual information to stabilize the initial interval localization and prevent catastrophic early deviation, then constructs multiple bit-stream candidates via reliability-guided bit flipping, while a sampling step further enhances candidate diversity and controls decoding overhead. Notably, although BART-JSCC benefits from pretrained semantic modeling, its relatively long continuous latent representation and aggressive compression from the 768-dimensional hidden space to a 16-dimensional channel representation introduce additional energy and representation bottlenecks, limiting its performance under the unified-energy setting. We further investigate the impact of context length in Figure 6. The results show that ICD remains effective even without contextual information, provided that the other proposed components (CCG, CCS, and CLR) are retained. Furthermore, increasing the context ratio further improves reconstruction performance, with the best result among the evaluated settings achieved at $\eta_{\text{raw}} = 0.5$. This is because a longer context provides a stronger conditional prior for AC interval localization and shortens the error-sensitive suffix, thereby reducing the propagation of residual decoding errors. We also report the computational complexity and inference latency of different schemes in Table 4 to assess the practical feasibility of ICD. The results show that ICD introduces additional receiver-side computation compared with ECCT-only SSCC and end-to-end JSCC baselines. This cost increases with the sampled subset size L_s . Therefore, L_s explicitly controls the computational cost of ICD by determining the number of LLM-based decoding and likelihood-evaluation passes. This provides a tunable robustness-complexity trade-off: smaller L_s can be used under stricter latency constraints, while larger L_s can be adopted when stronger decoding robustness is required. To evaluate the robustness against inaccurate context, we sweep the context corruption probability $p_{\text{ctx}} \in \{10^{-1}, 10^{-2}, 10^{-3}\}$, with a substantially more adverse setting than typical URLLC-level reliability requirements [40]. Figure 7 illustrates the normalized performance retention rate relative to the error-free context case. The results show that noticeable degradation occurs only under the extreme setting of $p_{\text{ctx}} = 10^{-1}$ at $\text{SNR}_{\text{unified}} = -3$ dB, while all metrics rapidly recover to near-saturation levels when $p_{\text{ctx}} \leq 10^{-2}$, indicating the robustness of ICD under deliberately severe context-corruption conditions.

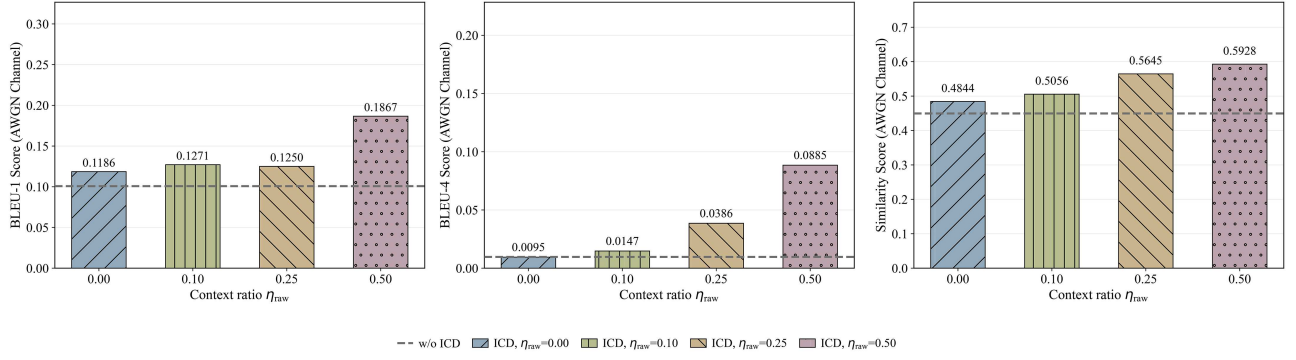


Figure 6 (Color online) Performance of ICD under different context ratios η_{raw} at $\text{SNR}_{\text{unified}} = -3$ dB.

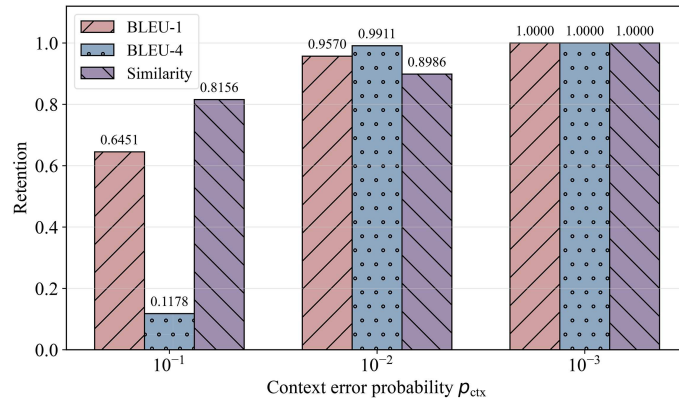


Figure 7 (Color online) Robustness of ICD under different context corruption probabilities at $\text{SNR}_{\text{unified}} = -3$ dB.

Table 4 Computational complexity and inference latency comparison.

Metric	LLM-AC, w/ECCT+ICD			LLM-AC, w/ECCT	Huffman, w/ECCT	UT+ quantization	UT	DeepSC	BART-JSCC
	$L_s = 5$	$L_s = 10$	$L_s = 15$						
FLOPs (G)	2695.31	4957.62	7177.6049	942.69	3.50	12.03	12.03	57.04	1287.69
Total latency (s)	144.06	263.22	400.77	45.00	45.00	16.26	17.54	4.39	21.07

5.3 Impact of confidence-ranked and sampled candidates hyperparameters

Figures 8 and 9 show the BLEU-4 performance of ICD under two coupled hyperparameters, namely, the size of the confidence-ranked candidate pool and the number of sampled candidates, which determines how many candidate bitstreams are ultimately decoded. A consistent trend is that using a moderate confidence-ranked pool together with a moderate number of sampled candidates yields the most reliable gains across SNR points. Further increasing the number of sampled candidates does not reliably yield better performance, suggesting that ICD is driven primarily by forming high-quality, reliability-consistent candidates, rather than simply decoding as many candidates as possible. Conversely, using too few sampled candidates under-explores the candidate space and makes ICD more prone to missing the correct interval updating path at low SNR. A similar non-monotonic effect is observed for the confidence-ranked pool size. When the ranked pool is reduced, ICD may discard the true bitstream prematurely, biasing the sampled set toward a narrow subset. By contrast, a larger ranked pool may include additional low-confidence candidates, potentially reducing sampling focus and weakening the reliability-likelihood fusion through less effective exploration. These results also reflect the performance-complexity trade-off of ICD. A larger L_c increases the subset search and energy-evaluation overhead, while a larger L_s almost linearly increases the number of LLM-based decoding attempts. Since moderate (L_c, L_s) settings achieve the best BLEU-4 performance in Figures 8 and 9, ICD does not require exhaustive candidate decoding and can be configured to balance robustness and latency in real-time applications.

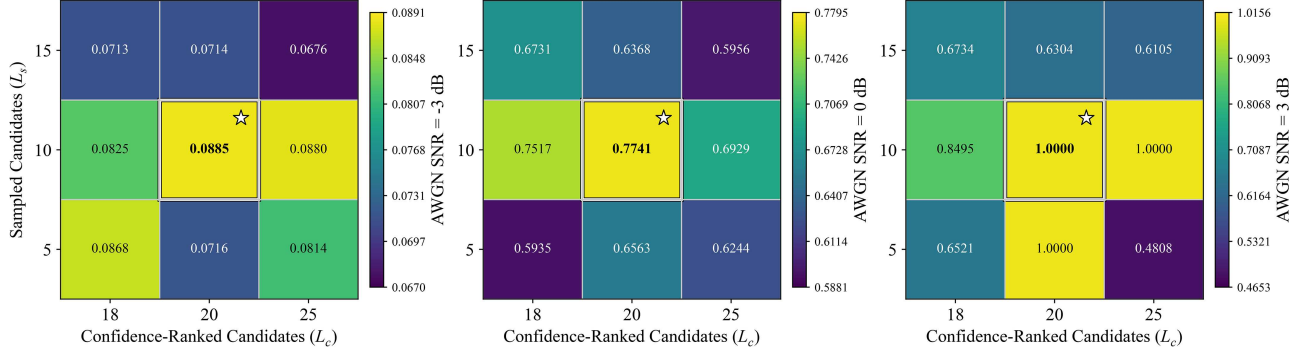


Figure 8 (Color online) BLEU-4 heatmaps of ICD performance under AWGN channels.

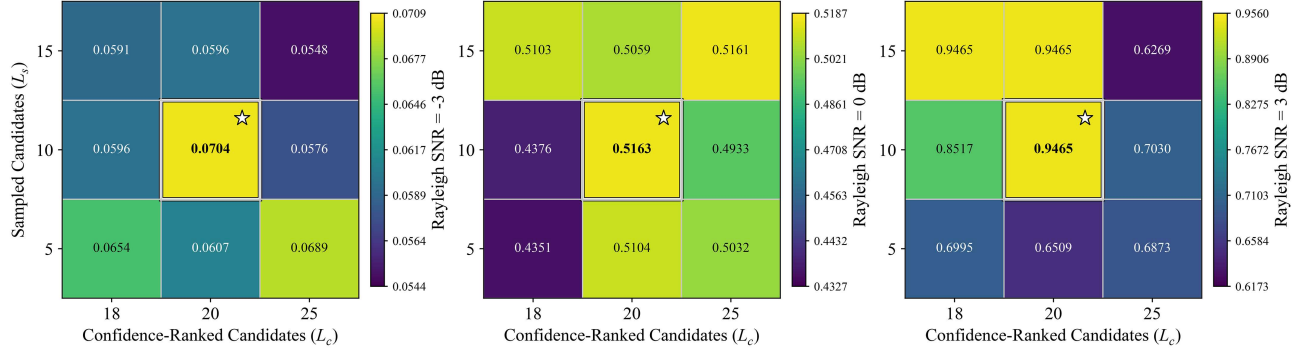


Figure 9 (Color online) BLEU-4 heatmaps of ICD performance under Rayleigh channels.

Table 5 Model sizes of the pretrained language-model backbones.

Model	Parameters (M)
GPT-Base	124.44
GPT-Medium	379.99
GPT-Large	811.78
GPT-XL	1607.94
BART	139.42

5.4 Scalability of the LLM backbone

We examine how the LLM backbone affects the performance of the proposed ICD framework. Specifically, we evaluate ICD with progressively larger LLM sizes while keeping all other system components unchanged, and the parameter counts of the evaluated backbones are summarized in Table 5. As shown in Figure 10, consistent but moderate gains are observed as the backbone size increases. This trend indicates that ICD can directly benefit from improved language modeling quality, as larger models provide more accurate conditional probability estimates for candidate generation and refinement. At the same time, the performance gap between different model scales remains bounded, suggesting that ICD is largely model-agnostic and already achieves strong performance with the base model. These results confirm that the proposed framework scales favorably with model capacity while maintaining robustness across different backbone choices.

5.5 Ablation experiments

Table 6 presents a comprehensive ablation study under AWGN channels across different code rates and SNRs. Overall, LLM-AC with ECCT+ICD consistently delivers the strongest performance, especially in the low-SNR regimes, where reliable recovery is most challenging. Compared with the ECCT-only baseline, incorporating ICD yields substantial gains across all three metrics, confirming the necessity of structured enhancement beyond ECCT-based decoding. As the code rate increases, performance tends to decline at the same SNR, since higher-rate coding provides less redundancy for error correction and thus leaves more residual errors after decoding. Under such conditions, the baselines degrade more noticeably, whereas ICD more consistently preserves its advantage.

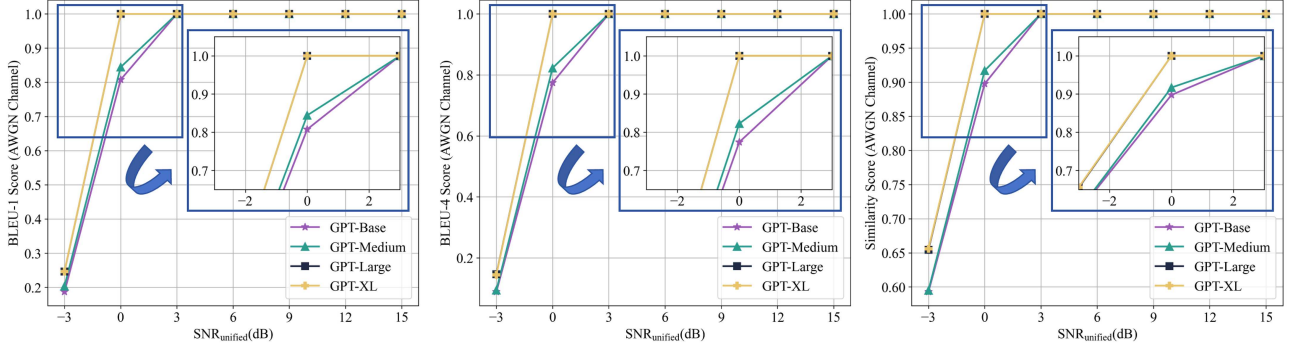


Figure 10 (Color online) Scalability evaluation with varying LLM backbone sizes under AWGN channels.

Table 6 Results under AWGN channels with different code rates. The best results are highlighted in bold.

Code rate (N, K)	SNR (dB)	Method	Metrics		
			BLEU1	BLEU4	Similarity score
(49, 24)	-3	LLM-AC, w/ECCT+ICD	0.1867	0.0885	0.5928
		LLM-AC, w/ECCT+Context	0.1832	0.0714	0.5921
		LLM-AC, w/ECCT	0.1007	0.0097	0.4496
	0	LLM-AC, w/ECCT+ICD	0.8076	0.7741	0.8976
		LLM-AC, w/ECCT+Context	0.6647	0.6135	0.8398
		LLM-AC, w/ECCT	0.5120	0.4584	0.7181
	3	LLM-AC, w/ECCT+ICD	1.0000	1.0000	1.0000
		LLM-AC, w/ECCT+Context	1.0000	1.0000	1.0000
		LLM-AC, w/ECCT	1.0000	1.0000	1.0000
(49, 30)	-3	LLM-AC, w/ECCT+ICD	0.1778	0.0708	0.5462
		LLM-AC, w/ECCT+Context	0.1398	0.0530	0.5275
		LLM-AC, w/ECCT	0.0937	0.0093	0.4623
	0	LLM-AC, w/ECCT+ICD	0.5457	0.4833	0.7532
		LLM-AC, w/ECCT+Context	0.5125	0.4429	0.7376
		LLM-AC, w/ECCT	0.3152	0.2322	0.6313
	3	LLM-AC, w/ECCT+ICD	0.9697	0.9602	0.9852
		LLM-AC, w/ECCT+Context	0.9697	0.9602	0.9852
		LLM-AC, w/ECCT	0.9444	0.9266	0.9734
(49, 36)	-3	LLM-AC, w/ECCT+ICD	0.1761	0.0480	0.5746
		LLM-AC, w/ECCT+Context	0.1828	0.0485	0.5738
		LLM-AC, w/ECCT	0.0970	0.0090	0.4668
	0	LLM-AC, w/ECCT+ICD	0.4433	0.3602	0.6973
		LLM-AC, w/ECCT+Context	0.2922	0.1719	0.5936
		LLM-AC, w/ECCT	0.1416	0.0478	0.5133
	3	LLM-AC, w/ECCT+ICD	0.9762	0.9674	0.9901
		LLM-AC, w/ECCT+Context	0.8779	0.8455	0.9402
		LLM-AC, w/ECCT	0.6786	0.6093	0.8073

When the CCS module is removed and only context-assisted decoding is retained (denoted as w/ECCT+Context), performance degrades but remains clearly superior to the ECCT-only baseline. This indicates that context modeling alone provides meaningful improvements, yet is insufficient to fully exploit the candidate space without sampling. The full ICD configuration consistently achieves the best or near-best results, highlighting the complementary roles of context modeling and candidate sampling. We also present a comparison of the runtime test on NVIDIA GeForce RTX 4090 and find that the CCS module yields a $1.6457\times$ speedup. Overall, it validates the effectiveness of the CCS module for a more computationally efficient solution.

6 Conclusion

In this work, we presented ICD, a receiver-side ICD framework for improving the robustness of SSCC-based transmission in low-SNR regimes. By leveraging ECCT-derived bit-wise reliability and contextual information, ICD uses a three-stage processing chain composed of CCG, CCS, and CLR. Specifically, CCG produces a confidence-ranked candidate pool via reliability-guided bit flipping, while CCS performs diversity-preserving sampling to achieve a favorable accuracy-complexity trade-off, and CLR fuses ECCT-derived reliability with source-level log-likelihood to determine the final reconstruction. We further provide theoretical guarantees on the stability and convergence of this CCS module. Extensive evaluations over AWGN and Rayleigh fading channels demonstrate consistent improvements compared with conventional SSCC pipelines and representative JSCC schemes, and achieve a favorable balance between decoding reliability and computational efficiency. Beyond the text transmission scenario considered in this work, ICD can be extended to other modalities by adapting the source model and contextual information. For image or video transmission, a large vision model (LVM)-based SSCC pipeline can represent visual blocks as pixel-token sequences and use an ImageGPT [41]-like autoregressive visual model to provide next-pixel likelihoods for AC-based decoding and candidate ranking. Thus, the linguistic likelihood in CLR can be replaced by visual-token likelihood, while CCG and CCS remain applicable at the bitstream level. In such visual scenarios, contextual information can be instantiated as visual blocks, regions, or frames obtained from previous transmissions and/or reliable auxiliary channels. In future work, we will extend ICD by incorporating stronger model-free decoding beyond ECCT and further investigate modality-specific SSCC designs for reliable visual content transmission.

Acknowledgements This work was supported in part by National Science and Technology Major Project (Grant No. 2025ZD1302200) and Huawei Cooperation Project (Grant No. TC20240829036).

References

- 1 Cui Q M, You X H, Wei N, et al. Overview of AI and communication for 6G network: fundamentals, challenges, and future research opportunities. *Sci China Inform Sci*, 2025, 68: 171301
- 2 Lu K, Zhou Q Y, Li R P, et al. Rethinking modern communication from semantic coding to semantic communication. *IEEE Wireless Commun*, 2023, 30: 158–164
- 3 Lu Z L, Li R P, Lu K, et al. Semantics-empowered communications: a tutorial-cum-survey. *IEEE Commun Surv Tutor*, 2024, 26: 41–79
- 4 Vaswani A. Attention is all you need. In: *Proceedings of Advances in Neural Information Processing Systems*, 2017. 30: 6000–6010
- 5 Cao X Y, Nan G S, Guo H C, et al. Exploring LLM-based multi-agent situation awareness for zero-trust space-air-ground integrated network. *IEEE J Sel Areas Commun*, 2025, 43: 2230–2247
- 6 Tong S Y, Yu X X, Li R P, et al. Alternate learning-based SNR-adaptive sparse semantic visual transmission. *IEEE Trans Wirel Commun*, 2025, 24: 1737–1752
- 7 Lu Z L, Li R P, Lei M, et al. Self-critical alternate learning-based semantic broadcast communication. *IEEE Trans Commun*, 2025, 73: 3347–3363
- 8 Ren T Q, Li R P, Zhao M m, et al. Separate source channel coding is still what you need: an LLM-based rethinking. *ArXiv:2501.04285*
- 9 Chiarot G, Silvestri C. Time series compression survey. *ACM Comput Surv*, 2023, 55: 1–32
- 10 Miao S S, Kestel C, Johannsen L, et al. Trends in channel coding for 6G. *Proc IEEE*, 2024, 112: 653–675
- 11 Delétang G, Ruoss A, Duquenne P A, et al. Language modeling is compression. *ArXiv:2309.10668*
- 12 Goyal M, Tatwawadi K, Chandak S, et al. DZip: improved general-purpose lossless compression based on novel neural network modeling. In: *Proceedings of Data Compression Conference*, 2021. 153–162
- 13 Mittu F, Bu Y, Gupta A, et al. Finezip: pushing the limits of large language models for practical lossless text compression. *ArXiv:2409.17141*
- 14 Huffman D A. A method for the construction of minimum-redundancy codes. *Proc IRE*, 1952, 40: 1098–1101
- 15 Witten I H, Neal R M, Cleary J G. Arithmetic coding for data compression. *Commun ACM*, 1987, 30: 520–540
- 16 Gallager R. Low-density parity-check codes. *IRE Trans Inf Theory*, 1962, 8: 21–28
- 17 Arikan E. Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Trans Inf Theory*, 2009, 55: 3051–3073
- 18 Liu D, Bober M, Kittler J. Neural belief propagation for scene graph generation. *IEEE Trans Pattern Anal Mach Intell*, 2023, 45: 10161–10172
- 19 Karg1 E U, Duman T M. A deep learning based decoder for concatenated coding over deletion channels. In: *Proceedings of the IEEE International Conference on Communications*, 2024. 2797–2802
- 20 Choukroun Y, Wolf L. Error correction code transformer. In: *Proceedings of Advances in Neural Information Processing Systems*, 2022. 38695–38705
- 21 Park S J, Kwak H Y, Kim S H, et al. Multiple-masks error correction code transformer for short block codes. *IEEE J Sel Areas Commun*, 2025, 43: 2518–2529
- 22 Liu Y, Jiang S, Zhang Y, et al. Extended context-based semantic communication system for text transmission. *Digit Commun Netw*, 2024, 10: 568–576
- 23 Ren Y Q, Kristensen A T, Shen Y F, et al. A sequence repetition node-based successive cancellation list decoder for 5G polar codes: algorithm and implementation. *IEEE Trans Signal Process*, 2022, 70: 5592–5607
- 24 Sagitov I, Pillet C, Balatsoukas-Stimming A, et al. Generalized restart mechanism for successive-cancellation flip decoding of polar codes. *J Signal Process Syst*, 2025, 97: 11–29
- 25 Meister C, Pimentel T, Wiher G, et al. Locally typical sampling. *Trans Assoc Comput Linguist*, 2023, 11: 102–121
- 26 Xie H Q, Qin Z J, Li G Y, et al. Deep learning enabled semantic communication systems. *IEEE Trans Signal Process*, 2021, 69: 2663–2675
- 27 Zhou Q Y, Li R P, Zhao Z F, et al. Semantic communication with adaptive universal transformer. *IEEE Wirel Commun Lett*, 2022, 11: 453–457
- 28 Zhou Q Y, Li R P, Zhao Z F, et al. Adaptive bit rate control in semantic communication with incremental knowledge-based HARQ. *IEEE Open J Commun Soc*, 2022, 3: 1076–1089
- 29 Zhang L T Y, She C Y, Ying K, et al. Unsupervised learning for ultra-reliable and low-latency communications with practical channel estimation. *IEEE Trans Wirel Commun*, 2024, 23: 3633–3647
- 30 Haque M E, Tariq F, Khandaker M R A, et al. A comprehensive survey of 5G URLLC and challenges in the 6G era. *ArXiv:2508.20205*

- 31 Xue J Z, Yu K, Zhang T Q, et al. Cooperative deep reinforcement learning enabled power allocation for packet duplication URLLC in multi-connectivity vehicular networks. *IEEE Trans Mob Comput*, 2024, 23: 8143–8157
- 32 Faria G R A, Agrawal S, Farinhas A, et al. QUEST: quality-aware Metropolis-Hastings sampling for machine translation. In: *Proceedings of Advances in Neural Information Processing Systems*, 2024. 89042–89068
- 33 Karan A, Du Y L. Reasoning with sampling: your base model is smarter than you think. *ArXiv:2510.14901*
- 34 Levin D A, Peres Y, Wilmer E L. *Markov Chains and Mixing Times*. Providence: American Mathematical Society, 2017
- 35 Norris J R. *Markov Chains*. Cambridge: Cambridge University Press, 1998
- 36 Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation. In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2002. 311–318
- 37 Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2019. 4171–4186
- 38 Koehn P. Europarl: a parallel corpus for statistical machine translation. In: *Proceedings of the Machine Translation Summit*, 2005. 79–86
- 39 Lewis M, Liu Y, Goyal N, et al. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2020. 7871–7880
- 40 Sultan A. Ultra reliable and low latency communications. 3GPP, 2023. <https://www.3gpp.org/technologies/urllc-2022>
- 41 Chen M, Radford A, Child R, et al. Generative pretraining from pixels. In: *Proceedings of the International Conference on Machine Learning*, 2020. 1691–1703