Large Models as Compressors for Next Generation Communication Systems

Tianqi Ren

Networked Intelligence for Comprehensive Efficiency (NICE) Lab
College of Information Science and Electronic Engineering
Zhejiang University
http://nice.rongpeng.info/

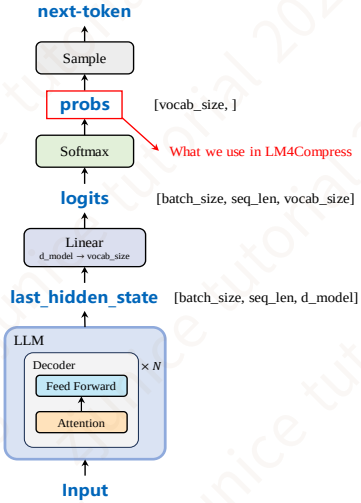
Aug. 25, 2025

Content

Large Models (LMs) as Compressors

Output Layers of LLM

- The most common form: **next-token**.
  - **The most intuitive output** when using LLMs.
  - Then mapped into characters via tokenizers.
- One level deeper: **probs** (probability distribution).
  - Where the **next-token** is **sampled** from (e.g. Top-k: sample from k tokens with highest probs).
  - **Key of LMs acting as compressors**.
- One level deeper: **logits**.
  - The **"scores" assigned to each token** in the vocabulary by the model's output.
  - `probs=Softmax(logits[0][-1], dim=-1)`
- One level deeper: **last_hidden_state**.
  - Output of the **last decoder layer**.
  - Hidden_states represent **high-dimensional contextual representations**.



Language Modeling is Compression[1]

- The key of lossless compression: **probabilistic modeling** of data.
- LMs, trained on massive data, can achieve **precise probability prediction**, able to accurately calculate $\rho(x_i|x_{<i})$
- Intuitively, **LMs** have the potential to be **compressors** of data.

Theoretically:

Based on **Shannon's Source Coding Theorem**

- **Entropy** of data:
  $H(x) = -\sum_x \rho(x) \log_2 \rho(x)$
- **Min average length** of lossless compression:
  $L^* = \mathbb{E}_{x_{1:n}\sim\rho}\left[\sum_{i=1}^n -\log_2 \rho\left(x_i|x_{<i}\right)\right]$
- Suboptimal length using an **estimated distribution** $\hat{\rho}$:
  $\sum_{i=1}^n -\log_2 \hat{\rho}\left(x_i|x_{<i}\right)$
- Expected length of lossless compression:
  $L^* = \mathbb{E}_{x_{1:n}\sim\rho}\left[\sum_{i=1}^n -\log_2 \hat{\rho}\left(x_i|x_{<i}\right)\right]$
- Binary Cross-Entropy Loss:
  $LogLoss = -\frac{1}{N}\sum_{i=1}^N \left[y_i \log\left(\hat{y}_i\right) + \left(1-y_i\right)\log\left(1-\hat{y}_i\right)\right]$

[1] G. Deletang et al., "Language modeling is compression," in *International Conference on Representation Learning*, vol. 2024, 2024, pp. 14 165–14 181.

Mechanism: LM-Powered Arithmetic Coding

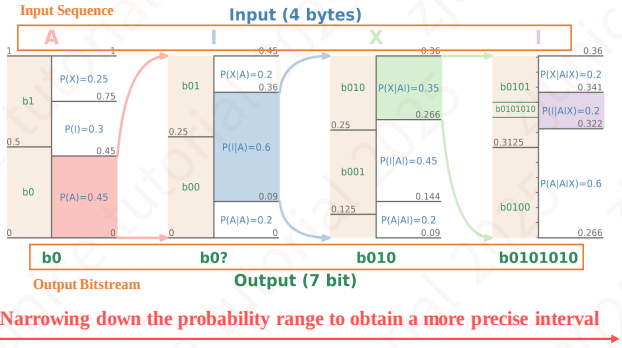
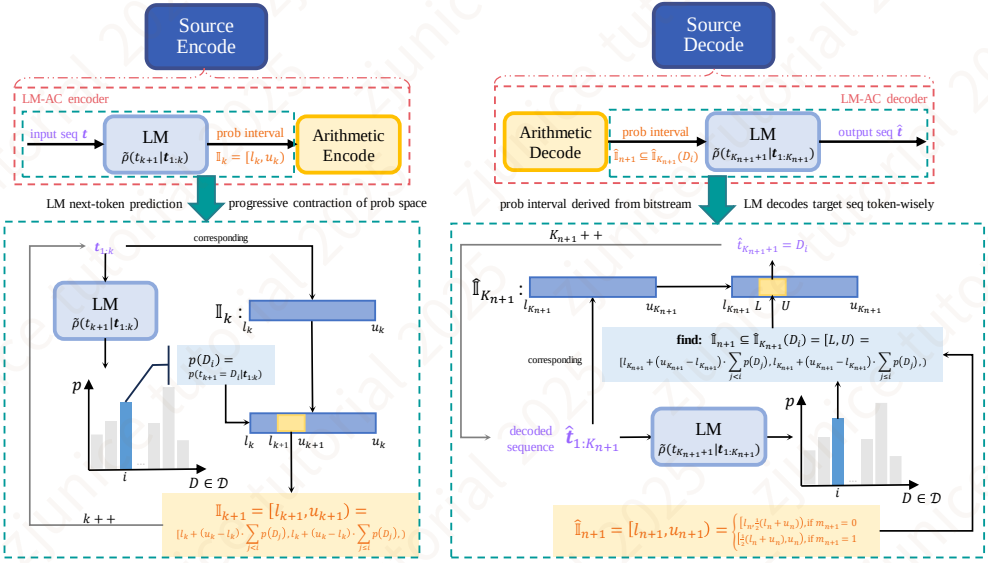


Figure: Example of Arithmetic Coding (AC)

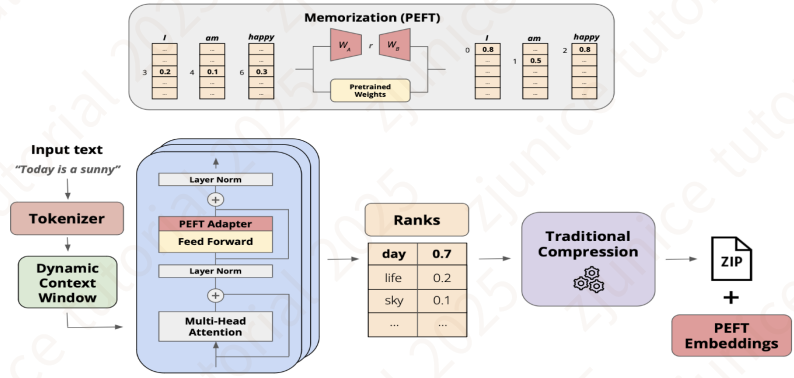
- **1. Initialization:**
  Start with the interval $[0, 1)$.
- **2. Get Probability:**
  Use LMs to get conditional probs $\hat{\rho}(x_i|x_{<i})$ of next symbol.
- **3. Narrow Interval:**
  AC encoder narrows current interval $[l, u)$ based on $\hat{\rho}(x_i|x_{<i})$.
- **4. Output Bitstream:**
  Output the shortest binary code in the final interval.

LM-based Arithmetic Codec



LM for Text Compression

- Typical works:
  - LLMZip(2023.6)[2], GPT-AC(2023.8)[3], AlphaZip(2024.9)[4], FineZip(2024.9)[5]



[2] C. S. K. Valmeekam et al., "Llmzip: Lossless text compression using large language models," *arXiv preprint arXiv:2306.04050*, 2023 .

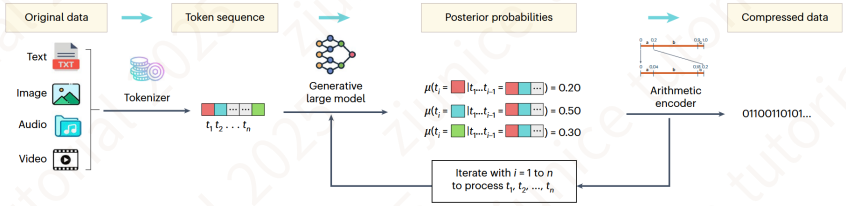
[3] C. Huang et al., "Approximating human-like few-shot learning with gpt-based compression," *arXiv preprint arXiv:2308.06942*, 2023 .

[4] S. S. Narashiman and N. Chandrachoodan, "Alphazip: Neural network-enhanced lossless text compression," *arXiv preprint arXiv:2409.15046*, 2024 .

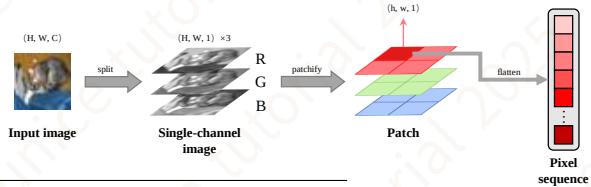
[5] F. Mittu et al., "Finezip: Pushing the limits of large language models for practical lossless text compression," *arXiv preprint arXiv:2409.17141*, 2024 .

LM for Image Compression

- Lossless Data Compression by Large Models[6] (using **iGPT**[7] for image compression)



- Image pre-processing:



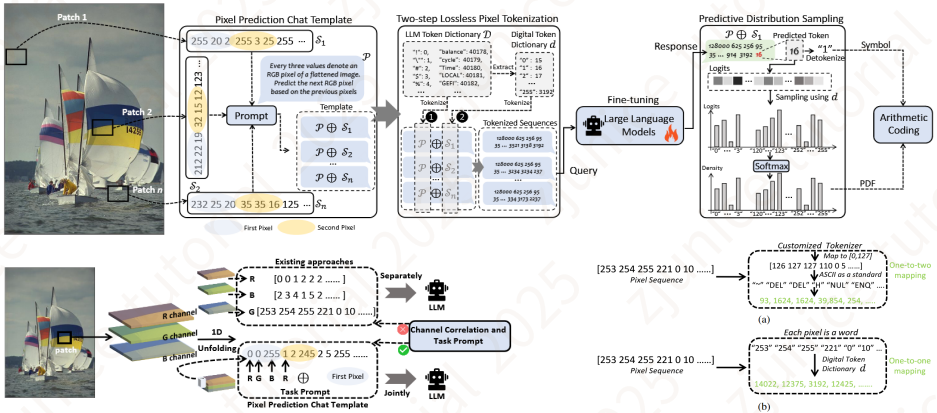
- 1. Channel splitting
- 2. Image patchification
- 3. Pixel flattening
- 4. Sequence concatenation

[6] Z. Li et al., "Lossless data compression by large models," *Nature Machine Intelligence*, pp. 1–6, 2025 .

[7] M. Chen et al., "Generative pretraining from pixels,", ser. ICML'20, JMLR.org, 2020.

LM for Image Compression

- Typical works:
  - Next-pixel prediction(2024.11)[8], Visual prompts(2025.2)[9]

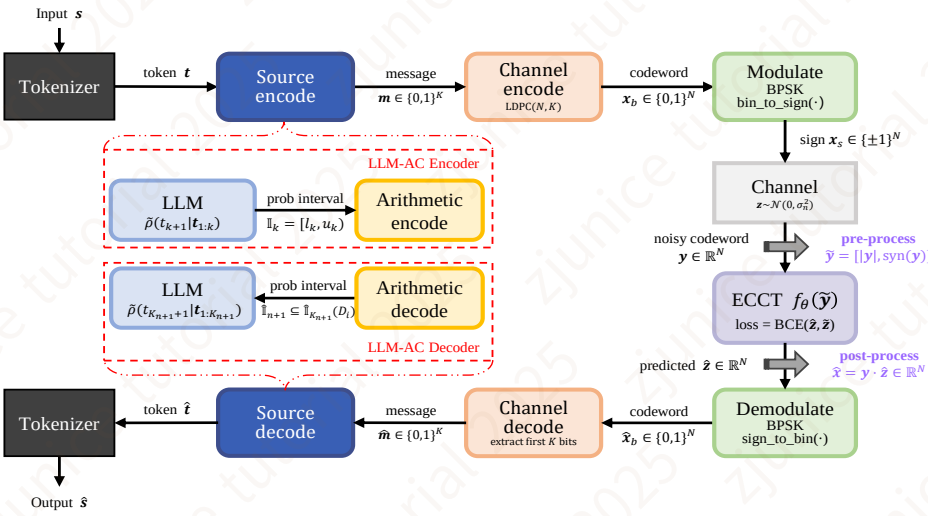


[8] K. Chen et al., "Large language models for lossless image compression: Next-pixel prediction in language space is all you need," *arXiv preprint arXiv:2411.12448*, 2024 .

[9] J. Du et al., "Large language model for lossless image compression with visual prompts," *arXiv preprint arXiv:2502.16163*, 2025 .

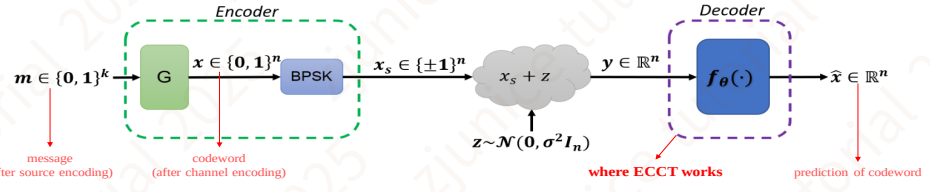
LM-based Separate Source Channel Coding (SSCC) Systems

System Framework of LM-SSCC[10]



[10] T. REN et al., "Separate source channel coding is still what you need what you need: An llm-based rethinking based rethinking," *ZTE COMMUNICATIONS*, vol. 23, no. 1, 2025 .

Error Correction Code Transformer (ECCT)[11]



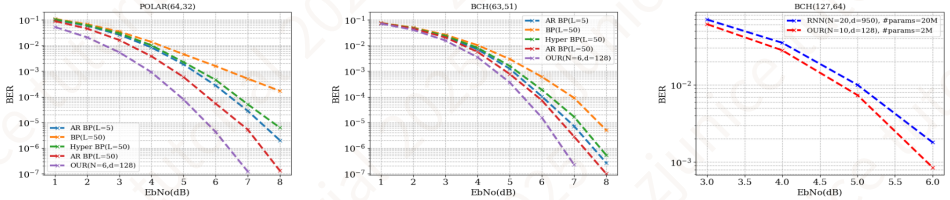
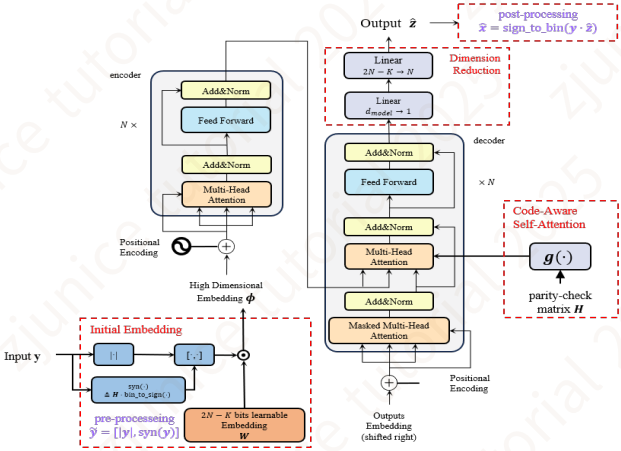


Figure: ECCT outperforms (a,b) traditional ECCs and (c) other NN-based approaches in terms of BER over Rayleigh channel.

[11] Y. Choukroun and L. Wolf, "Error correction code transformer," in *Advances in Neural Information Processing Systems*, vol. 35, Curran Associates, Inc., 2022, pp. 38 695–38 705.

Error Correction Code Transformer (ECCT)

- Transformer-based module for enhanced channel coding in SSCC Systems



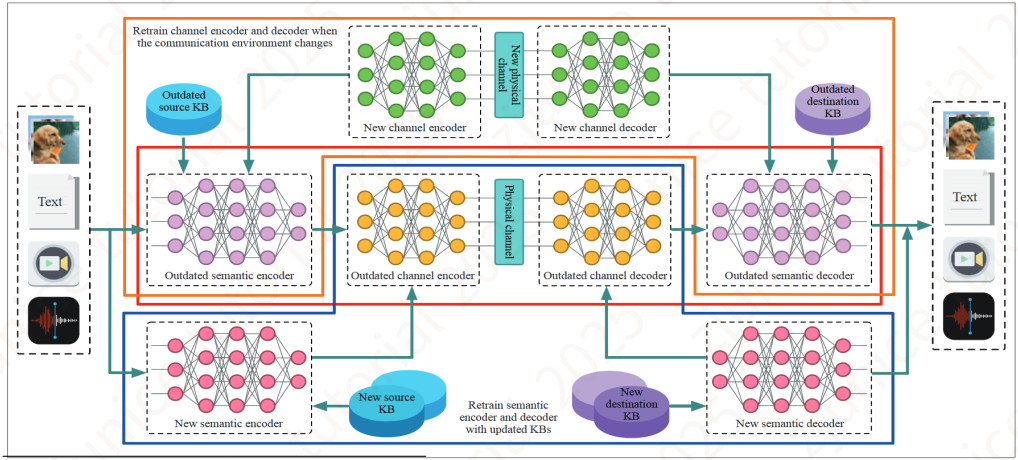
Additional elements:

- **Positional Reliability Coding**
  - Positional: Dimension-wise mapping into high-dimensional embeddings
  - Reliability: Attributable to the integrated embeddings of magnitude and syndrome
- **Code Aware Self-Attention**
  - Incorporate fundamental domain knowledge about relevant codes
  - Construct a symmetric mask according to the parity-check matrix

Comparisons with Joint Source Channel Coding (JSCC) Systems

Deep Learning-Based Communication Systems

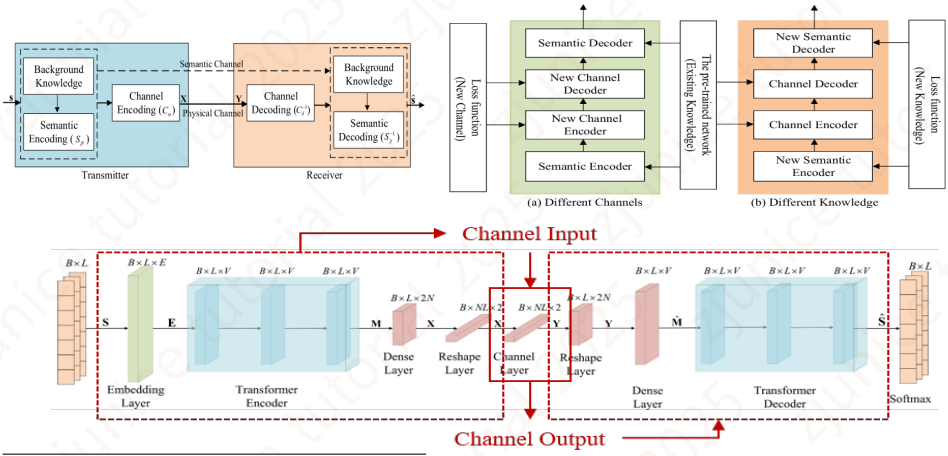
- Deep Learning-Based Semantic Communication Architecture[12]



[12] X. Luo et al., "Semantic communications: Overview, open issues, and future research directions," *IEEE Wireless Communications*, vol. 29, no. 1, pp. 210–219, 2022 .

JSCC System Frameworks

End-to-End Communication Systems for **Text** Modality (represented by **DeepSC**[13],**UT**[14])

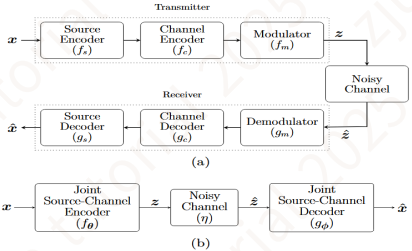


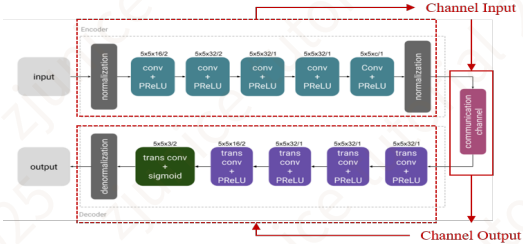
[13] H. Xie et al., "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021 .

[14] Q. Zhou et al., "Semantic communication with adaptive universal transformer," *IEEE Wireless Communications Letters*, vol. 11, no. 3, pp. 453–457, 2022 .

JSCC System Frameworks

End-to-End Communication Systems for **Image** Modality (represented by **DeepJSCC**[15])





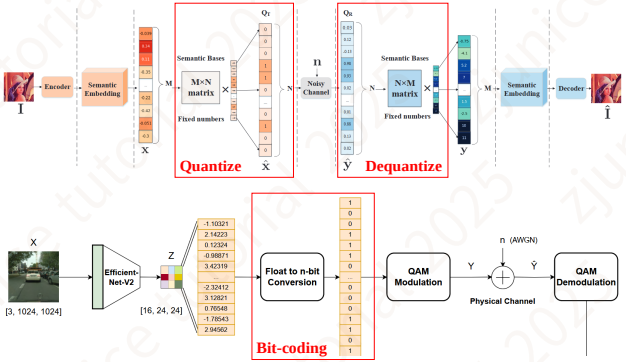
Two Typical JSCC Systems

- Most JSCC systems are modified based on these two systems (DeepSC and DeepJSCC), either updating the model (e.g. CNN to Transformer) or adding additional modules.

[15] E. Bourtsoulatze et al., "Deep joint source-channel coding for wireless image transmission," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 567–579, 2019 .

JSCC System Frameworks

- Some studies incorporate quantization[16]/bit-coding[17] modules



- **SparseSBC**: **Quantization** limits the bits to be transmitted, but also **limits the performance**.
- **D-SIC**: **Bit-coding** introduces **redundancy**, consuming more transmission energy.

[16] S. Tong et al., "Alternate learning-based snr-adaptive sparse semantic visual transmission," *IEEE Transactions on Wireless Communications*, vol. 24, no. 2, pp. 1737–1752, 2025 .

[17] B. Khalid et al., "D-sic: Energy-efficient digital semantic image communication via large generative models," *Authorea Preprints*, 2025 .

Limitations of JSCC

Despite the promising performance of JSCC, several **critical shortcomings** hinder its practical deployment:

- **Incompatibility with Digital Communication Systems**
  - JSCC systems are inherently **analog** in nature, transmitting **continuous-valued** symbols, incompatible with the **discrete, bit-based** architecture.
- **Performance Degradation from Digitization Efforts**
  - Introducing **quantization/bit-coding modules** to bridge the gap imposes **a new, often restrictive, upper bound** on achievable performance.
- **Lack of Adaptability and High Training Overhead**
  - JSCC models must be **fully trained end-to-end** for a specific **dataset/channel/SNR**, requiring a costly retraining process for every new deployment scenario.
- **Overly Optimistic SNR Calculation and Unfair Benchmarking**
  - For JSCC systems, performance is often evaluated based on the **average power per transmitted symbol** rather than the power **per information bit**.

Improvements of LM-SSCC[18]

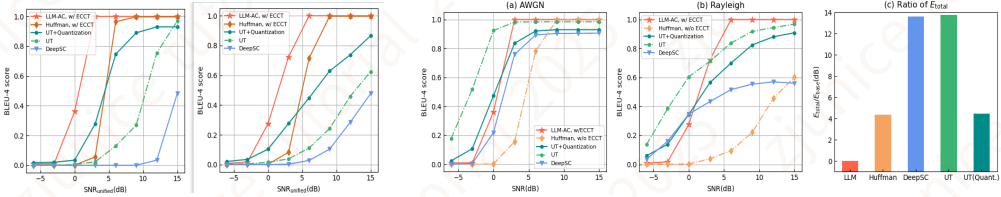
- **Superior System Compatibility**
  - Our scheme outputs a **standardized discrete bitstream**, seamlessly integrating with existing **digital** communication infrastructures.
- **Modular Design and High Flexibility**
  - LM-AC and ECCT can both be functioned as a **"plug-and-play" component**. Our work effectively **pushes the "cliff effect" of SSCC systems towards lower SNR**, making SSCC systems highly competitive even in harsh environments.
- **Strong Generalization Capability**
  - Leveraging foundation models pretrained on **massive, general-purpose datasets**, LM-SSCC exhibits powerful generalization capabilities across **diverse data domains** without exhaustive retraining.
- **Exceptional Performance Potential**
  - Attributed to the **vast prior knowledge** embedded within pretrained large models, their semantic prediction capability promises an unprecedented compression performance at a **super-low bitrate**.

[18] T. REN et al., "Separate source channel coding is still what you need: An llm-based rethinking based rethinking," *ZTE COMMUNICATIONS*, vol. 23, no. 1, 2025 .

Improvements of LM-SSCC

- Providing a **unified SNR benchmark** by **aligning the total transmitted energy**
  - For float32-based JSCC communication system, $R_m = \log_2(M) = 32$, consumes **additional** $10 \times \log_{10}(32) \approx 15.051$dB

$$\begin{aligned}\textbf{SNR} &= 10\log_{10}\left(\frac{E_{\text{total}}}{N_0 \cdot \text{Num}}\right) = 10\log_{10}\left(\frac{E_{\text{total}}}{N_0 \cdot \text{Num}_{\text{unified}}} \times \frac{\text{Num}_{\text{unified}}}{\text{Num}}\right) \\ &= \textbf{SNR}_{\textbf{unified}} + 10\log_{10}\left(\frac{\text{Num}_{\text{unified}}}{\text{Num}}\right).\end{aligned}$$



Conclusion & Future Prospect

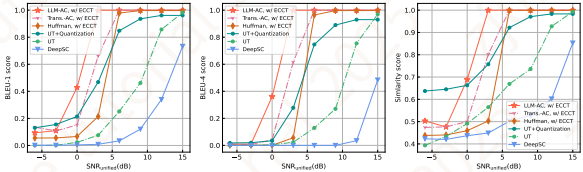
Conclusion

I have talked about

- **Language Modeling is Compression:**
  Minimizing log-loss is equivalent to minimizing compression rate, enabling efficient lossless compression via arithmetic coding.
- **Advantages of LM-SSCC:**
  Integrating LM-based compression with ECCT channel coding enhances system robustness and generalization while maintaining digital compatibility.
- **Superiority over JSCC:**
  LM-SSCC overcomes limitations of JSCC in digitization, generalization, and training overhead, offering a viable path for next-generation communication systems.

Future Prospect

- **Transformer-based** Communication Systems
  Transformer-based AC + ECCT
  Maybe key component of physical layer in next generation communication systems



Model	Params	FLOPs
LLM-SSCC	$124.51M$	$\approx 960G$
Trans.-SSCC	$0.33M$	$\approx 4.5G$
DeepSC	$10.58M$	$\approx 57G$
UT	$18.43M$	$\approx 12G$

- Fully leveraging the characteristic of image data
  Pixel-level priors (e.g., intra-pixel inter-channel correlation and local self-similarity)
  - Change the paradigm of pixel value sequence prediction.

  Fixed value range of the sequence (0-255)
  - Build up pixel-specific tokenizers and models.

Thank you

Tianqi Ren
Networked Intelligence for Comprehensive Efficiency (NICE) Lab
College of Information Science and Electronic Engineering
Zhejiang University
https://nice.rongpeng.info