White-box Error Correction Code Transformer

Ziyan Zheng [1,2] Chin Wa (Ken) Lau [2,3] Nian Guo [2] Xiang Shi [1,2] Shao-Lun Huang [1]

[1]Tsinghua University [2]Huawei Technologies Co., Ltd. [3]The Chinese University of Hong Kong

Abstract

Error correcting codes (ECCs) play a crucial role in modern communication systems by ensuring reliable data transmission over noisy channels. While traditional algorithms based on belief propagation suffer from limited decoding performance, transformer-based approaches have emerged as powerful solutions for ECC decoding. However, the internal mechanisms of transformer-based approaches remain largely unexplained, making it challenging to understand and improve their performance.

In this work, we propose a White-box Error Correction Code Transformer (WECCT) that provides theoretical insights into transformer-based decoding. By formulating the decoding problem from a sparse rate reduction perspective and introducing a novel Multi-head Tanner-subspaces Self Attention mechanism, our approach provides a parameter-efficient and theoretically principled framework for understanding transformer-based decoding. Extensive experiments across various code families demonstrate that this interpretable design achieves competitive performance compared to state-of-the-art decoders.

Introduction

Error Correction Code Decoding

Error correcting codes (ECCs) are fundamental building blocks in modern communication systems, enabling reliable data transmission across noisy channels by adding redundant information to the transmitted messages.

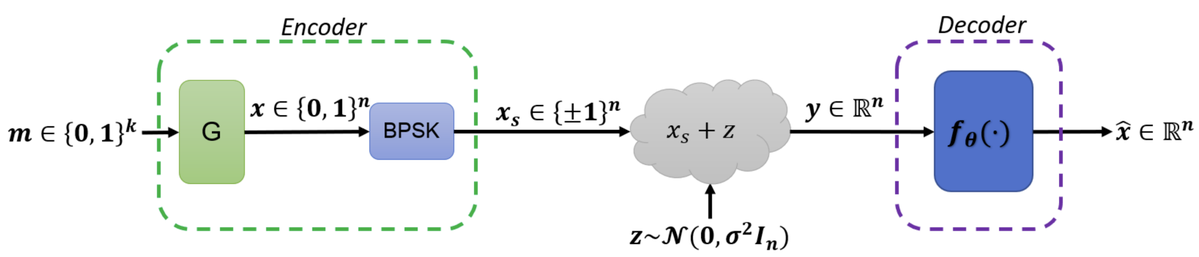

Figure 1. Overview of ECC encoding, transmission, and decoding process [1].

- **Codeword Bits**: The potentially corrupted transmitted symbols $\boldsymbol{y} \in \mathbb{R}^n$
- **Syndromes**: $\boldsymbol{s}(\boldsymbol{y}) = H\boldsymbol{y}_b \in \{0,1\}^{n-k}$ where $HG = 0$ over $\mathbb{F}_2$ and $\boldsymbol{y}_b$ represents hard decisions on $\boldsymbol{y}$
- **Tanner graph**: A bipartite graph with n variable nodes (codeword bits) and $n-k$ check nodes (parity check equations/syndromes), where connections are determined by the parity check matrix H.

Related Work

Several major approaches have been developed for ECC decoding:

- **Classical Decoders**: Belief Propagation (BP) iteratively exchanges probabilistic messages between bit and check nodes in the Tanner graph; Model-based neural decoders enhance BP by parameterizing message-passing operations with neural networks; Model-free neural decoders employ generic neural architectures without explicit reliance on traditional decoding algorithms.
- **Error Correction Code Transformer (ECCT)** [1]: Pioneered transformer architectures for ECC decoding by processing concatenated magnitude and syndrome vectors through masked self-attention modules.
- **Cross-attention Message-Passing Transformer (CrossMPT)** [2]: Processes magnitude and syndrome vectors separately through cross-attention blocks, better reflecting their distinct roles in error correction.

We propose a White-box Error Correction Code Transformer (WECCT) framework that builds upon theoretical foundation of **CRATE**[3] while specifically targeting the challenges of ECC decoding. Our WECCT represents the first attempt to introduce an interpretable, white-box transformer architecture for decoding tasks. The theoretically-motivated design provides clear mathematical objectives and bridges the gap between transformer architectures and coding theory.

WECCT Framework

From ML Decoding to Sparse Rate Reduction

We reformulate ML decoding as a sparse rate reduction objective:

$$\max_f \mathbb{E}_{\boldsymbol{Z}}\left[R(\boldsymbol{Z}) - R^c(\boldsymbol{Z} \mid \boldsymbol{U}_{[K]}) - \lambda\|\boldsymbol{Z}\|_1\right] \tag{1}$$

- $\boldsymbol{Z} = [\boldsymbol{Z}_b, \boldsymbol{Z}_s]$: Combined token embeddings of both bits ($\boldsymbol{Z}_b$) and syndromes ($\boldsymbol{Z}_s$)
- $R(\boldsymbol{Z}) = \frac{1}{2}\log\det(\boldsymbol{I} + \alpha\boldsymbol{Z}^*\boldsymbol{Z})$: Coding rate measuring overall information content
- $R^c(\boldsymbol{Z} \mid \boldsymbol{U}_{[K]}) = \frac{1}{2}\sum_{k=1}^K \log\det(\boldsymbol{I} + \beta(\boldsymbol{U}_k^*\boldsymbol{Z})^*(\boldsymbol{U}_k^*\boldsymbol{Z}))$: Rate when tokens are encoded by K subspaces
- $\lambda\|\boldsymbol{Z}\|_1$: Promotes sparsity in representations

Tanner Subspace

For token embeddings $\boldsymbol{Z} = [\boldsymbol{Z}_b, \boldsymbol{Z}_s] \in \mathbb{R}^{d\times(2n-k)}$ where d is embedding dimension, we define the Tanner subspace for each node i:

$$\mathcal{T}_i \triangleq \mathrm{Span}\{\boldsymbol{z}_j \mid \mathcal{M}(H)_{ji} = 1\} \subset \mathbb{R}^d \tag{2}$$

where $\mathcal{M}(H)$ extends the connectivity matrix to include bit-syndrome connections.

Compression via Multi-head Tanner-subspaces Self Attention (MTSA) $\rightarrow \min_f R^c(\boldsymbol{Z} \mid \boldsymbol{U}_{[K]})$

MTSA implements gradient descent on coding rate while preserving the Tanner graph structure:

$$\mathrm{TSA}(\boldsymbol{Z} \mid \boldsymbol{U}_k) = (\boldsymbol{U}_k^*\boldsymbol{Z})\,\mathrm{softmax}((\boldsymbol{U}_k^*\boldsymbol{Z})^*(\boldsymbol{U}_k^*\boldsymbol{Z}) + \phi(\mathcal{M}(H))) \tag{3}$$

$$\mathrm{MTSA}(\boldsymbol{Z} \mid \boldsymbol{U}_{[K]}) = \beta[\boldsymbol{U}_1,\ldots,\boldsymbol{U}_K][\mathrm{TSA}(\boldsymbol{Z} \mid \boldsymbol{U}_1),\ldots,\mathrm{TSA}(\boldsymbol{Z} \mid \boldsymbol{U}_K)]^T \tag{4}$$

where $\phi(\mathcal{M}(H))$ serves as a masking function ensuring attention only flows between connected nodes.

Sparsification via Iterative Shrinkage-Thresholding Algorithm (ISTA) $\rightarrow \min_f \lambda\|\boldsymbol{Z}\|_1 - R(\boldsymbol{Z})$

ISTA optimizes the sparsification term:

$$\boldsymbol{Z}^{l+1} \approx \underset{\boldsymbol{Z}}{\mathrm{argmin}}\left\{\lambda\|\boldsymbol{Z}\|_1 + \frac{1}{2}\|\boldsymbol{Z}^{l+1/2} - \boldsymbol{D}^l\boldsymbol{Z}\|_F^2\right\} \tag{5}$$

Implemented as:

$$\boldsymbol{Z}^{l+1} = \mathrm{ReLU}(\boldsymbol{Z}^{l+1/2} - \eta(\boldsymbol{D}^l)^*(\boldsymbol{D}^l\boldsymbol{Z}^{l+1/2} - \boldsymbol{Z}^{l+1/2}) - \eta\lambda\boldsymbol{1}) \tag{6}$$

where $\boldsymbol{Z}^{l+1/2}$ is the intermediate output after MTSA, $\boldsymbol{D}^l$ is a learnable dictionary, and η is the step size.

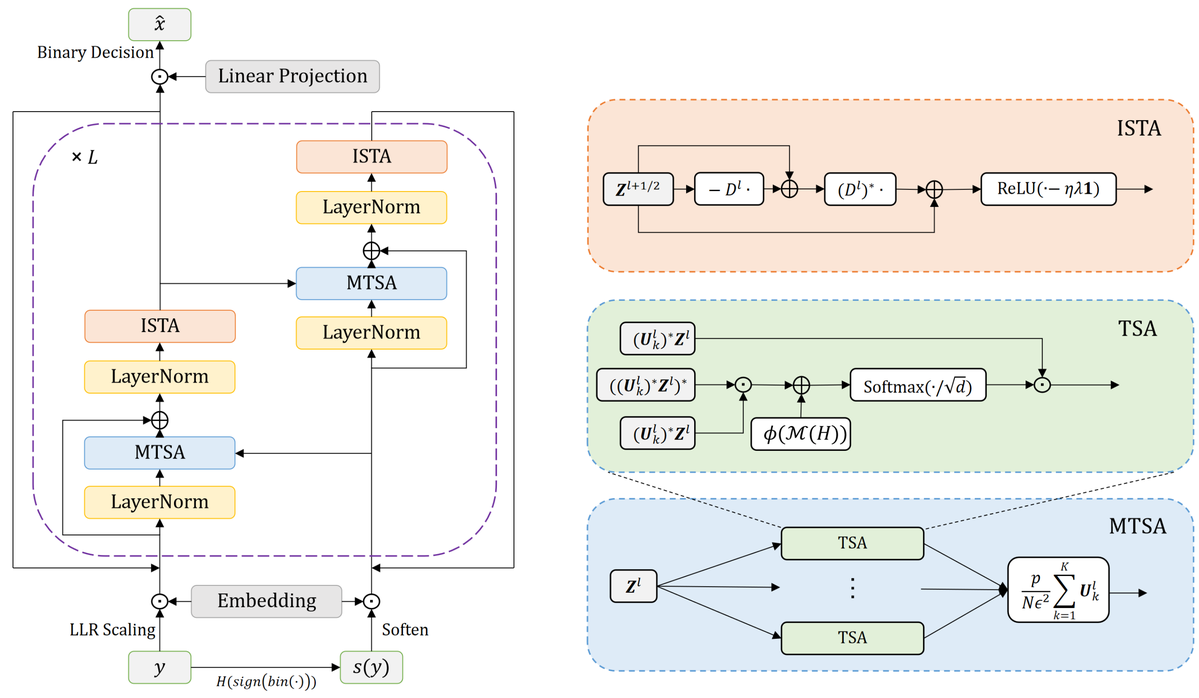

Figure 2. Overview of the WECCT architecture. The model architecture flows from bottom to top on the left, consisting of inputs embedding, decoder layers and outputs prediction, with the structure of decoder layers expanded on the right.

Results

Table 1. **Competitive Decoding Performance**. Comparison at three different SNR values (4 dB, 5 dB, 6 dB) for different decoders, measured by the negative natural logarithm of BER (higher is better). For each specific code in the WECCT column, the first row shows results with 6 decoder layers ($L = 6$), while the second row shows results with 12 decoder layers ($L = 12$). Best results are shown in **bold** and second best results are underlined.

Model		BP			AR BP			ECCT			CrossMPT			WECCT		
Codes	Parameter	4	5	6	4	5	6	4	5	6	4	5	6	4	5	6
BCH	(31,16)	4.63	5.88	7.60	5.48	7.37	9.60	6.39	8.29	10.66	**6.98**	**9.25**	12.48	6.31	8.52	11.39
														6.51	8.73	**12.65**
BCH	(63,36)	4.03	5.42	7.26	4.57	6.39	8.92	4.86	6.65	9.10	**5.03**	**6.91**	**9.37**	4.81	6.53	9.01
														4.91	6.70	9.24
BCH	(63,45)	4.36	5.55	7.26	4.97	6.90	9.41	5.60	7.79	10.93	**5.90**	8.20	**11.62**	5.55	7.80	10.90
														5.87	**8.27**	11.25
BCH	(63,51)	4.50	5.82	7.42	5.17	7.16	9.53	5.66	7.89	11.01	**5.78**	**8.08**	**11.41**	5.54	7.76	10.86
														5.62	7.89	11.04
Polar	(64,32)	4.26	5.38	6.50	5.57	7.43	9.82	6.99	9.44	12.32	**7.50**	**9.97**	**13.31**	6.42	8.69	11.34
														6.71	9.03	12.54
Polar	(64,48)	4.74	5.94	7.42	5.41	7.19	9.30	6.36	8.46	11.09	**6.51**	**8.70**	11.31	6.08	8.19	11.13
														6.32	8.48	**11.36**
Polar	(128,64)	4.10	5.11	6.15	4.84	6.78	9.30	5.92	8.64	12.18	**7.52**	**11.21**	**14.76**	5.43	7.86	11.20
														6.11	8.94	12.32
Polar	(128,86)	4.49	5.65	6.97	5.39	7.37	10.13	6.31	9.01	12.45	**7.51**	**10.83**	**15.24**	6.11	8.83	12.60
														6.97	10.22	14.29
Polar	(128,96)	4.61	5.79	7.08	5.27	7.44	10.20	6.31	9.12	12.47	**7.15**	**10.15**	13.13	6.09	8.84	11.96
														6.48	9.35	**13.48**
LDPC	(49,24)	6.23	8.19	11.72	6.58	9.39	12.39	6.13	8.71	12.10	6.68	9.52	13.19	6.36	9.08	12.92
														6.70	**9.63**	**14.02**
LDPC	(121,60)	4.82	7.21	10.87	5.22	8.31	13.07	5.17	8.31	13.30	5.74	9.26	14.78	5.63	8.97	13.91
														6.05	**9.77**	**14.92**
LDPC	(121,70)	5.88	8.76	13.04	6.45	10.01	14.77	6.40	10.21	16.11	7.06	11.39	**17.52**	6.97	11.17	14.70
														7.42	**12.20**	14.92
MacKay	(96,48)	6.84	9.40	12.57	7.43	10.65	14.65	7.38	10.72	14.83	7.97	**11.77**	15.52	7.50	10.97	14.29
														8.43	11.66	**16.08**
CCSDS	(128,64)	6.55	9.65	13.78	7.25	10.99	16.36	6.88	10.90	15.90	7.68	11.88	**17.50**	7.40	11.70	14.76
														8.24	**12.36**	15.67

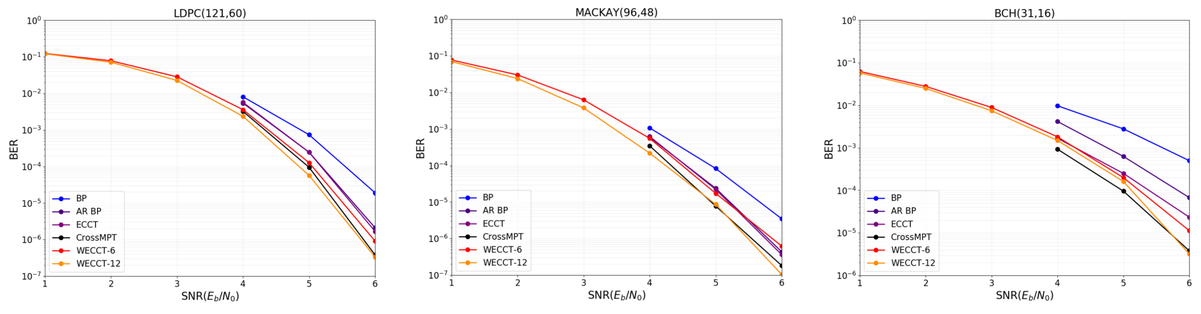

Figure 3. The BER performance of various decoders (BP, AR BP, ECCT, CrossMPT) and WECCT.

Table 2. **Parameter Efficiency**. Comparison of parameters and FLOPs for different decoders.

Code	Model	Parameters (M)	FLOPs (M)
LDPC(121,70)	ECCT	1.23	37.7
	CrossMPT	1.23	28.8
	WECCT-6	0.46	17.2
	WECCT-12	0.85	33.8
BCH(63,45)	ECCT	1.19	14.0
	CrossMPT	1.19	11.8
	WECCT-6	0.43	8.4
	WECCT-12	0.82	16.2

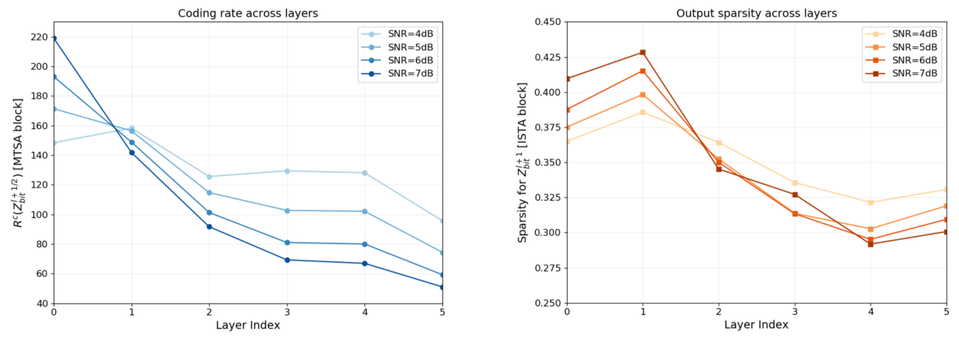

Figure 4. **Interpretability**. Left: Coding rate across layers for different SNR values. Right: Output sparsification of ISTA blocks across layers.

References

[1] Yoni Choukroun and Lior Wolf. Error correction code transformer. *Advances in Neural Information Processing Systems*, 35:38695–38705, 2022.

[2] Seong-Joon Park, Hee-Youl Kwak, Sang-Hyo Kim, Yongjune Kim, and Jong-Seon No. Crossmpt: Cross-attention message-passing transformer for error correcting codes. *arXiv preprint arXiv:2405.01033*, 2024.

[3] Yaodong Yu, Sam Buchanan, Druv Pai, Tianzhe Chu, Ziyang Wu, Shengbang Tong, Benjamin Haeffele, and Yi Ma. White-box transformers via sparse rate reduction. *Advances in Neural Information Processing Systems*, 36:9422–9457, 2023.

zhengzy19@mails.tsinghua.edu.cn

shaolun.huang@sz.tsinghua.edu.cn