

Dissecting BERT Layers: FFN Dual Role, Separability-Guided Layer Skip, and Interpretable Classification via Charge-Flow Learning

Yeonseong Cynn¹

¹River Lab

May 2026

Abstract

We present a layer-level analysis framework for BERT across five GLUE tasks. Using RX (River XAI), a charge-flow based interpretable learning framework, we replace BERT’s classifier with a 2–16 node interpretable network and identify removable layers through separability analysis. Our key contributions are: (1) a separability-guided layer skip method validated by actual BERT forward-pass experiments on all five tasks, (2) quantitative decomposition of FFN’s dual role — 92% structural (norm normalization) vs. 8% classification-relevant — explaining why FFN removal causes model collapse while individual layers appear “harmful” to classification, and (3) error analysis revealing that 60–93% of misclassifications are high-confidence errors (margin > 0.3), indicating BERT’s CLS representation itself is the bottleneck. RX is one application of a broader proprietary learning framework developed at River Lab; method specifics are subject to intellectual property protection.

1 Introduction

Transformer compression has been extensively studied through knowledge distillation [4], pruning [2], layer dropping [1, 3], and quantization. However, most approaches treat the model as a black box, optimizing for accuracy-efficiency tradeoffs without explaining *why* certain components can be removed.

Closer to our setting, Sajjad et al. [3] empirically tested various layer-dropping strategies on fine-tuned BERT, finding top layers more prunable. Probing classifiers [5] have shown that different layers encode different linguistic properties. Our work differs in three ways: (a) we provide quantitative per-layer role decomposition (structural vs. classification) rather than empirical strategy comparison, (b) we add an interpretable compensation classifier that achieves lossless or near-lossless accuracy on 3 of 5 GLUE tasks, and (c) we resolve the apparent paradox of “harmful but essential” FFN through structural/classification decomposition.

We analyze *what each layer and component does* for classification using RX (River XAI), a charge-flow based interpretable learning framework.¹ This perspective naturally leads to:

- **Separability analysis:** Quantifying each layer’s contribution to class separation

¹Charge-flow learning is a proprietary class of forward learning frameworks; architectural details are not disclosed in this paper.

- **FFN dual-role decomposition:** Separating structural (format-preserving) from classification-relevant changes
- **Interpretable classification:** Replacing 768-dimensional classifiers with 2–16 specialized nodes whose roles are directly readable

RX is one demonstration of a broader proprietary learning framework under active development. Method details (training procedure, optimization specifics) are not disclosed in this paper; we focus on *what the framework reveals* about BERT’s layer structure. The framework’s primary commercial target is large language model optimization (Llama, Qwen, GPT-class architectures), with BERT serving as a verification substrate.

2 Method

2.1 RX Classifier

We replace BERT’s linear classifier (768→2) with a compact RX network. RX uses a sign-preserving feature transformation as input preprocessing and a custom nonlinear activation that adaptively scales by fan-in. The classifier is trained efficiently on a single CPU. With 2–16 hidden nodes, each node’s class-conditional activation pattern reveals its functional role (e.g., “positive-sentiment specialist”).

2.2 Separability Analysis

For each layer l , we compute class separability of the CLS vector:

$$S_l = \frac{\|\mu_0 - \mu_1\|}{\sqrt{(\sigma_0^2 + \sigma_1^2)/2}} \quad (1)$$

where μ_c, σ_c^2 are class-conditional mean and variance. The separability delta $\Delta S_l = S_l - S_{l-1}$ indicates whether layer l improves ($\Delta > 0$) or degrades ($\Delta < 0$) class separation.

2.3 Layer Skip

Layers with low ΔS are candidates for removal. We validate by actually skipping the layer in BERT’s forward pass (feeding L_{k-1} ’s output directly to L_{k+1} ’s attention) and retraining a RX classifier on the modified output.

2.4 FFN Dual-Role Decomposition

For each layer, we decompose the FFN’s effect into structural and classification components:

$$\Delta_{\text{struct}} = \left\| \frac{\bar{d}_0 + \bar{d}_1}{2} \right\|, \quad \Delta_{\text{class}} = \|\bar{d}_1 - \bar{d}_0\| \quad (2)$$

where $\bar{d}_c = \mathbb{E}_{x \in c}[\text{FFN}(x) - x]$ is the mean change for class c . The ratio $\Delta_{\text{struct}}/\Delta_{\text{class}}$ quantifies how much of FFN’s computation is class-independent format transformation vs. classification-relevant modification.

3 Results

3.1 RX Classifier Performance

Fair comparison on identical test sets (Table 1). The RX classifier uses 2–16 hidden nodes, trained on BERT’s CLS output:

Table 1: BERT vs. RX classifier (fair comparison, same test set)

Task	BERT	RX	Δ	Verdict
SST-2	92.7%	92.4% \pm 0.3%	−0.3%	Equivalent
CoLA	80.5%	80.7% \pm 0.3%	+0.2%	Equivalent
MRPC	91.2%	90.2% \pm 0.3%	−1.0%	Slight loss
QNLI	92.2%	88.8% \pm 0.5%	−3.4%	Loss
RTE	68.3%	69.8% \pm 1.6%	+1.5%	Equivalent

RX achieves BERT-equivalent accuracy on 3/5 tasks with 2–16 nodes replacing 768-dimensional classification, enabling direct inspection of decision logic.

3.2 Layer Importance and Skip

Figure 1 shows the separability profile across all five tasks. A common pattern emerges: L1 causes dramatic separability collapse (normalization), L2–L7 show continued decline or stagnation, and L9–L12 recover sharply — with L12 consistently producing the largest increase.

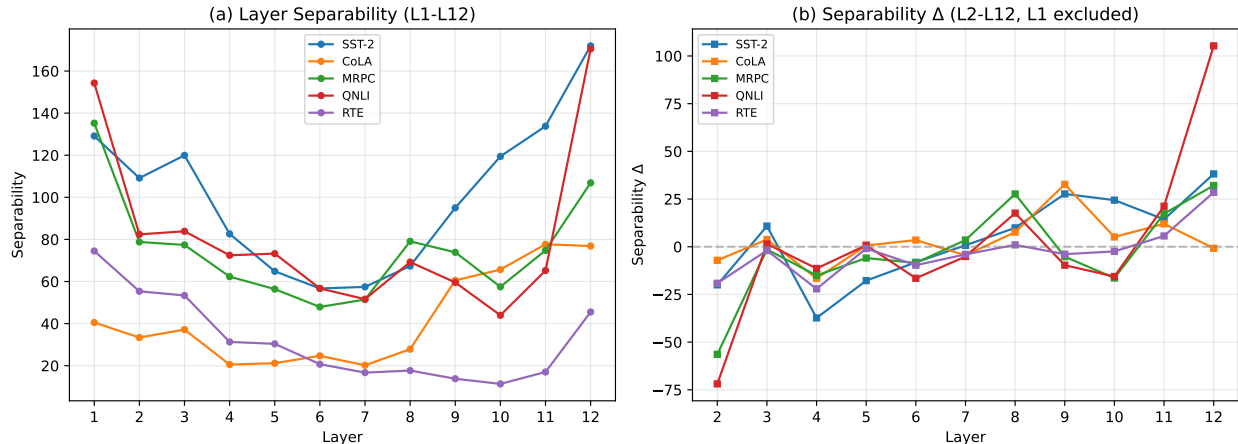


Figure 1: (a) Layer separability for five GLUE tasks. All tasks share a U-shaped pattern with minimum around L6–L7 and sharp recovery at L12. (b) Per-layer separability change (L1 excluded due to scale). Negative Δ layers are skip candidates.

Separability analysis identifies skip candidates ($\Delta S < 0$). Actual skip experiments validate the predictions (Table 2):

Separability correctly identifies skip candidates in 4/5 tasks (Figure 2). In QNLI, the predicted worst layer (L2, $\Delta = -72$) differs from the optimal skip (L10), suggesting separability is a useful heuristic but not definitive.

Table 2: 1-layer skip + RX classifier (6.5% parameter reduction)

Task	Skip	Accuracy	vs. BERT	Prediction
SST-2	L5 ($\Delta=-18$)	92.4%	-0.3%	Match
CoLA	L5 ($\Delta=-17$)	80.7%	+0.2%	Match
MRPC	L2 ($\Delta=-56$)	90.2%	-1.0%	Exact
QNLI	L10 ($\Delta=-16$)	88.8%	-3.4%	Mismatch
RTE	L6 ($\Delta=-10$)	69.8%	+1.5%	Match

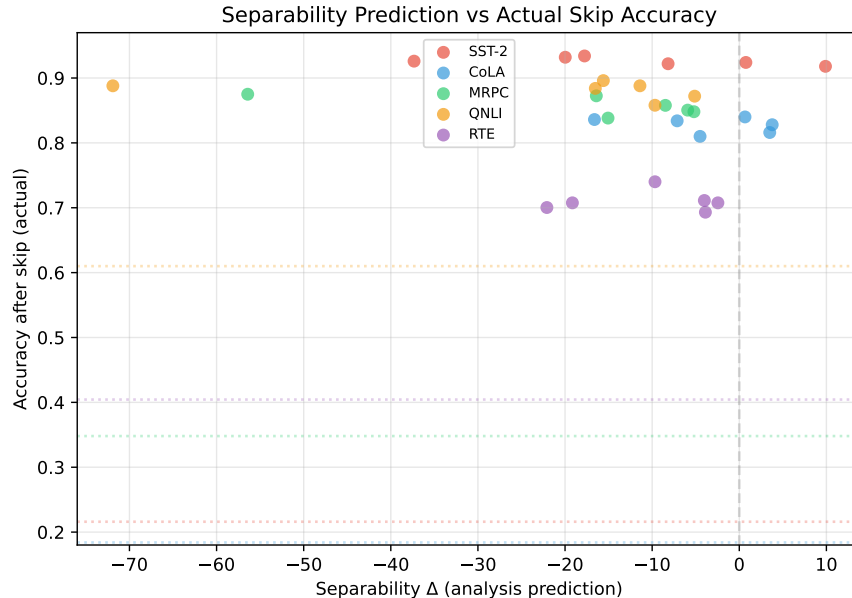


Figure 2: Separability Δ (x-axis) vs. actual accuracy after skip (y-axis). Points cluster in the upper-left (negative Δ , high accuracy), confirming that layers with negative separability change are safe to skip. Dotted lines show original BERT accuracy per task.

RTE shows an especially interesting result: removing L6 *improves* accuracy by 1.5%, confirming that this layer actively harms the classification signal.

Multi-layer skip results (Table 3) show graceful degradation:

Key observations:

- 1-layer skip (6.5%): All tasks within 1.5% of BERT; RTE *exceeds* original.
- 2-layer skip (12.9%): SST-2, CoLA, RTE remain within 2%.
- 3-layer skip (19.4%): SST-2 retains 90.6% (-2.7%); RTE 70.4% (-2.1%). MRPC, QNLI, CoLA show 6%+ drops.
- RTE is uniquely resilient — 2 layers removed with *no accuracy loss*, confirming multiple layers actively harm this task.

Table 3: Multi-layer skip: best configuration per task

Task	Skip layers	Accuracy	vs. BERT	Reduction
SST-2	L5	93.4%	−0.3%	6.5%
	L2+L5	92.0%	−1.3%	12.9%
	L2+L5+L6	90.6%	−2.7%	19.4%
CoLA	L5	84.0%	+1.4%	6.5%
	L2+L5	80.8%	−1.8%	12.9%
	L2+L4+L5	76.2%	−6.4%	19.4%
MRPC	L2	87.5%	−0.2%	6.5%
	L2+L10	85.8%	−1.9%	12.9%
	L2+L6+L10	81.1%	−6.6%	19.4%
QNLI	L10	89.6%	−0.8%	6.5%
	L6+L10	87.4%	−3.0%	12.9%
	L2+L6+L10	84.4%	−6.0%	19.4%
RTE	L6	74.0%	+1.5%	6.5%
	L4+L6	73.3%	+0.8%	12.9%
	L2+L4+L6	70.4%	−2.1%	19.4%

3.3 FFN Dual Role

We decompose FFN changes across all layers and tasks. Figure 3 visualizes the structural/classification ratio as a heatmap, revealing a clear gradient from early (purely structural) to late (classification-involved) layers.

Selected values are shown in Table 4:

Table 4: FFN structural/classification ratio by layer (selected tasks)

Layer	SST-2	CoLA	QNLI	RTE
L1	103×	234×	123×	213×
L4	27×	95×	24×	64×
L8	12×	28×	7×	15×
L12	0.8×	1.7×	2.0×	4.4×

Key findings:

- Early layers (L1–L4): FFN is 100–200× more structural than classification-relevant. Nearly pure format transformation (norm compression from 0.2–0.4×).
- Late layers (L9–L12): Ratio drops to 1–8×. FFN increasingly modifies classification-relevant dimensions.
- SST-2 L12: Ratio is 0.8× — classification change *exceeds* structural change. This is where FFN actively reshapes the classification signal.

Why FFN removal causes collapse: FFN compresses vector norms by 0.2–0.7× (e.g., SST-2 L8: $\|h\|$ from 34.8 to 18.5). This normalization is required by the next layer’s attention

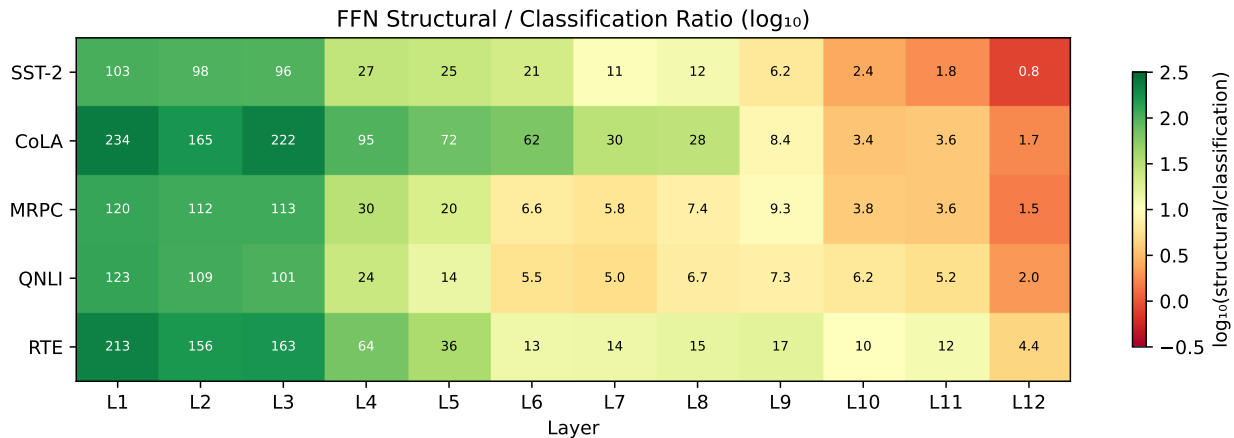


Figure 3: FFN structural/classification ratio (\log_{10} scale) across layers and tasks. Green = predominantly structural ($>10\times$), red = classification-dominant ($<1\times$). Early layers (L1–L3) are 100–200 \times structural across all tasks.

mechanism. Removing FFN disrupts input scale, causing catastrophic failure regardless of the small classification-relevant component.

SST-2 L8 detailed: The most “harmful” FFN (-17% accuracy when comparing attention-output vs. FFN-output classification). Decomposition shows 92% structural (norm compression) and 8% classification-relevant — but this 8% causes the entire 17% accuracy drop by slightly shifting 223/768 dimensions away from the class-separating direction.

3.4 Validation: FFN Neuron Pruning

To validate the dual-role analysis, we perform structural pruning on SST-2’s FFN neurons across all 12 layers. Neurons with activation frequency below a threshold (measured across all tokens in the full validation set, 872 samples) are masked to zero.

Table 5: All-layer FFN pruning on SST-2 (full validation, 872 samples)

Threshold	Removed	% of FFN	Accuracy	Δ
(original)	0	0%	92.4%	—
$<0.5\%$ active	3,092	8%	92.7%	+0.3%
$<1\%$ active	7,205	20%	91.6%	−0.8%
$<2\%$ active	13,933	38%	63.3%	collapse

At the 0.5% threshold, removing 8% of FFN neurons (3,092 / 36,864) *improves* accuracy by 0.3%, consistent with the dual-role finding: these rarely-active neurons contribute only noise to the classification signal. The sharp collapse at 2% threshold confirms that the remaining neurons perform essential structural transformation.

Per-layer sparsity follows the structural/classification gradient: early layers (L1–L3) have 82–84% inactive neurons ($<5\%$ activation), while L12 has only 34% — matching the finding that early FFN is predominantly structural and therefore more prunable.

3.5 Error Analysis

Table 6: Error characteristics across tasks

Task	Errors	High-conf (%)	Avg margin	Dominant direction
SST-2	30	60%	0.39	positive→negative (77%)
CoLA	75	93%	0.72	ungrammatical→grammatical (56%)
MRPC	44	87%	0.61	non-paraphrase→paraphrase (57%)
QNLI	46	73%	0.60	entailment→not-entailment (63%)
RTE	75	87%	0.58	not-entailment→entailment (63%)

Figure 4 shows the error direction distribution for each task. 60–93% of errors are high-confidence (margin > 0.3): the model is *confidently wrong*, not uncertain. This indicates the CLS representation itself points in the wrong direction for these samples, making classifier improvement insufficient — the upstream representation is the bottleneck.

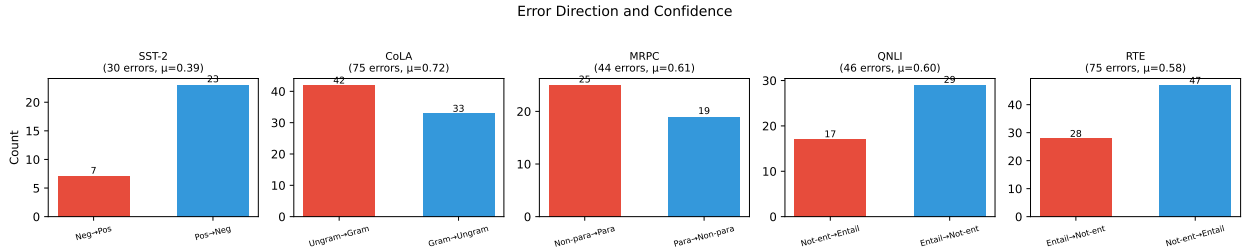


Figure 4: Error direction and average confidence margin (μ) per task. Each bar shows the count of misclassifications in that direction. SST-2 shows strong asymmetry: positive→negative errors dominate (77%).

Task-specific error patterns:

- **SST-2** (positive→negative, 77%): Negative lexical items (e.g., “cloying”, “preachy”) activate the negative-specialist node regardless of sentential context, causing negation-ignorant misclassification.
- **CoLA** (ungrammatical→grammatical, 56%): Subtle syntactic errors (e.g., “Did Calvin his homework?” — missing verb) are not detected because the surface structure appears well-formed.
- **MRPC** (non-paraphrase→paraphrase, 57%): Surface-similar sentences with different meanings are confused. The CLS representation captures lexical overlap but misses semantic divergence.
- **QNLI** (entailment→not-entailment, 63%): When question and answer lack keyword overlap, the entailment node fails to activate even when the answer logically follows.
- **RTE** (not-entailment→entailment, 63%): The model over-predicts entailment, possibly due to superficial similarity between premise and hypothesis.

Q=[0,0] pattern: In SST-2 and QNLI, some errors show $Q_{\text{out}} = [0, 0]$ — no RX node activates at all. These represent CLS vectors that fall entirely outside the learned activation patterns,

resulting in “judgment refusal.” This phenomenon suggests the existence of representation blind spots in BERT’s CLS space that are invisible to traditional accuracy metrics.

4 Discussion

4.1 Interpretable Classification

The RX classifier replaces BERT’s 768-dimensional linear head with 2–16 nodes. With 4 nodes for SST-2, we can state “H2 is the negative-sentiment specialist ($\Delta=2.1$)” — a level of interpretability impossible with standard classifiers. Each node’s class-conditional activation reveals its role in the decision process.

4.2 Separability as a Diagnostic Tool

Separability analysis is computationally trivial (mean/variance computation) yet provides actionable guidance for layer removal. Its 4/5 match rate with actual skip experiments suggests it captures meaningful information about layer function, though the QNLI mismatch warns against using it as the sole criterion.

4.3 FFN: Structurally Essential, Classification-Harmful

The dual-role decomposition resolves an apparent paradox: FFN layers can be simultaneously “harmful” to classification (accuracy drops when comparing attention-output to FFN-output) and “essential” to the model (removing FFN causes collapse). The resolution is that FFN’s dominant function (92%) is structural format transformation, while its minor classification-relevant changes (8%) happen to degrade the classification signal.

This has implications for FFN optimization: rather than removing FFN entirely, one could selectively suppress the classification-degrading components while preserving the structural transformation — a direction we leave for future work.

4.4 Method Disclosure

This paper documents what RX reveals about fine-tuned BERT layer structure and FFN function. RX itself, including its training procedure, optimization formulation, and the broader learning framework it is built upon, is proprietary. Korean patent applications cover the foundational method. Verification weights and analysis scripts are available for results reproduction; the training pipeline is not released.

4.5 Limitations

- QNLI shows 3.4% accuracy gap, suggesting RX’s capacity (16 nodes) may be insufficient for complex question-answering tasks
- RTE has high variance ($\pm 1.6\%$) due to small test set (277 samples)
- All analysis operates on CLS vectors, missing token-level interactions
- Separability is a heuristic, not a causal explanation of layer function

5 Conclusion

We demonstrate that layer-level decomposition provides complementary insights to existing compression methods. The key finding — FFN’s 92% structural / 8% classification split — explains a fundamental constraint on transformer compression and suggests new optimization directions. Combined with interpretable RX classification and separability-guided layer removal, this framework enables systematic per-layer analysis. Application to large language models is the primary commercial direction. Method details and commercial licensing inquiries: contact authors.

References

- [1] Angela Fan, Edouard Grave, and Armand Joulin. Reducing transformer depth on demand with structured dropout. *arXiv preprint arXiv:1909.11556*, 2019.
- [2] Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? *Advances in Neural Information Processing Systems*, 32, 2019.
- [3] Hassan Sajjad, Fahim Dalvi, Nadir Durrani, and Preslav Nakov. On the effect of dropping layers of pre-trained transformer models. *Computer Speech & Language*, 77:101429, 2023.
- [4] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.
- [5] Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.