INTERVENTION-AWARE COMPRESSION OF NEURAL NETWORKS VIA CAUSAL ABSTRACTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Model distillation and mechanistic interpretability share a common objective: replacing a pre-trained model (the teacher) with a more compact algorithm (the student) that preserves relevant aspects of its behavior. Distillation typically targets input-output agreement, while mechanistic interpretability instead focuses on creating compact representations that are causally faithful to a model's internal computation, often on a restricted set of inputs. Prior work suggests that constraining a student to reproduce aspects of a teacher's internal computation can improve distillation. We pursue this direction and build on recent advances in causal abstraction that provide differentiable objectives for measuring whether a smaller model constitutes a causally faithful abstraction of a larger one. We use these objectives to develop Causal Abstraction Distillation (CAD) and its practical variant CAB, which trains student Transformers to match not only the teacher's input-output behavior, but also its responses to interventions and perturbations applied to inputs and intermediate activations. This interventional training encourages the student to preserve the teacher's computational structure rather than merely imitate its predictions. Across encoder and decoder-only models, we find that CAB improves intervention-response robustness in several settings while maintaining competitive task performance. CAB can also improve generalization to related datasets, with benefits extending to 14B-to-3B Qwen2.5 distillation.

1 INTRODUCTION

Modern language models achieve strong performance at the cost of high computational and memory requirements, making post-hoc compression an important avenue for efficient deployment (Hoffmann et al., 2022; Zhu et al., 2024). Model compression covers a broad set of techniques such as pruning, quantization, and knowledge distillation. In particular, knowledge distillation trains a smaller *student* to mimic the behavior of a larger *teacher*, typically by matching their predictive distributions over a broad set of inputs (Hinton et al., 2015).

A seemingly different objective arises in mechanistic interpretability: constructing simpler algorithmic descriptions of how a neural network computes its outputs (Rai et al., 2025). Rather than merely reproducing input-output behavior, a mechanistic explanation is a simple algorithm that should preserve the relevant internal causal structure of the computation. Causal abstraction provides a general framework for this requirement: a simpler high-level causal model is a valid abstraction of a detailed low-level model when their computations agree under corresponding interventions (Beckers & Halpern, 2019; Geiger et al., 2025).

From this perspective, *compression and mechanistic interpretation are two instances of the same computation-compression problem*. Standard distillation seeks a compact model that behaviorally mimics a teacher, usually within a similar model class (e.g., Transformers) for both student and teacher, and over many inputs. Mechanistic interpretability instead seeks a much simpler, often qualitatively different model that reproduces the relevant causal computation, typically for a restricted behavior or set of inputs. The distinction is a matter of perspective about what matters when compressing the computation.

This connection suggests using causal abstraction also as a more general foundation for distillation, an idea also suggested previously by Wu et al. (2022). Discovering an abstraction of a fixed neural network is difficult in the general interpretability setting: the space of possible high-level variables, mechanisms, and mappings is enormous, and valid mechanistic explanations need not be identifiable

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(Méloux et al., 2025; Li et al., 2025; Sutter et al., 2025). Distillation provides a more flexible goal. The teacher and student have related computational structures, their layer correspondence can be user-specified, and we are free to optimize both the student and the maps relating its representations to those of the teacher, i.e., we are free to force the abstraction to our chosen level of abstraction. Instead of *discovering* a causal abstraction inside a fixed model, we can therefore *train* a smaller model to become one.

We develop and formalize this idea as *Causal Abstraction Distillation*. When teacher and student networks are seen as causal models, we can learn abstraction maps to align internal representations. During training, we generate causally informative interventions in the teacher, transport them to the student through the learned maps, and require the resulting computations to remain consistent at downstream abstraction boundaries. This yields a general distillation objective in which the intervention and the location at which causal consistency is evaluated can both vary. Standard output distillation (Hinton et al., 2015), intermediate-representation distillation (Romero et al., 2015; Sun et al., 2019; Jiao et al., 2020), distillation under input perturbations (Beyer et al., 2022; Goldblum et al., 2020), and output matching under internal interventions (Wu et al., 2022) can consequently be viewed as particular cases of the same abstraction objective. We show that it improves intervention-response robustness in several settings while maintaining competitive task performance, and can improve generalization to related datasets.

Contributions. Our contributions are: (i) **Distillation as causal abstraction.** We formalize model distillation as learning a high-level causal abstraction of a teacher and derive a unified objective that encompasses several existing distillation paradigms as special cases. (ii) **Causal Abstraction Distillation.** We introduce a training procedure that learns abstraction maps between teacher and student representations and uses teacher-induced internal interventions to enforce causal consistency across multiple downstream layers. (iii) **Interventional evaluation of compressed models.** We evaluate compressed students not only by task performance but also by their responses to held-out interventions, using both map-independent intervention-response robustness and map-dependent causal abstraction diagnostics.

Our results further show that predictive accuracy alone can conceal large differences in computational faithfulness: models with similar task performance can exhibit substantially different intervention-response robustness. Causal abstraction therefore provides both a training principle and an evaluation criterion for compression methods that aim to preserve not only what a teacher computes, but also how its computation responds to intervention. It opens a space for importing algorithmic progress from mechanistic interpretability more directly into model compression.

2 Related Work

Knowledge Distillation. Large neural networks are costly to store and deploy, motivating a broad literature on model compression, including pruning, quantization, and knowledge distillation (Zhu et al., 2024). While pruning and quantization directly reduce the parameter count or numerical precision of a pretrained model, knowledge distillation trains a smaller model to reproduce properties of a larger teacher. Our work belongs to the latter family, but targets preservation of the teacher's *causal computation*, rather than only its predictive behavior. Original knowledge distillation trains a student to match the teacher's output distribution (Hinton et al., 2015). Subsequent work extends this supervision to internal representations. FitNets aligns intermediate teacher and student features (Romero et al., 2015), while Patient KD and TinyBERT transfer information across multiple Transformer layers (Sun et al., 2019; Jiao et al., 2020). TED further learns task-specific transformations of teacher representations before matching them to the student (Liang et al., 2023). These approaches constrain representations observed during standard forward passes. In contrast, we learn abstraction maps between teacher and student states and require their computations to remain consistent under interventions.

Intervention-Aware Distillation. Several works augment distillation with perturbations or causal objectives. At the input level, applying data augmentations to teacher and student can substantially improve distillation (Beyer et al., 2022), while adversarially robust distillation transfers teacher behavior on adversarially perturbed inputs (Goldblum et al., 2020). Causally motivated approaches have also been proposed: Deng & Zhang (2021) uses causal adjustment to limit the transfer of context-induced teacher biases, whereas DIITO directly matches teacher and student outputs under corresponding interchange interventions on aligned internal representations (Wu et al., 2022). Re-

lately, Causal Mechanism Reduction studies pruning through internal mechanism replacement and evaluates reduced networks using interchange interventions (Asiaee, 2026).

Our approach generalizes intervention-based distillation in two ways. Rather than restricting interventions to values obtained from another forward pass, we search directly for causally informative interventions in the teacher and transport them to the student through the learned abstraction maps. Moreover, we enforce consistency at multiple downstream internal states, not only at the model output, thereby directly optimizing the student as an approximate causal abstraction of the teacher. Our contribution builds on recent developments in causal abstraction, therefore we dedicate the next background section to further describe it.

3 Background: Causal Abstraction

Structural causal models and interventions. A Structural Causal Model (SCM) (Pearl, 2009) is a tuple $\mathcal{M} = \langle \mathcal{U}, \mathcal{X}, \mathcal{R}, \mathcal{F}, P_{\mathcal{U}} \rangle$, where $\mathcal{U}$ and $\mathcal{X}$ are exogenous and endogenous variables, $\mathcal{R}$ specifies their domains, and $\mathcal{F} = \{f_X\}_{X \in \mathcal{X}}$ defines the structural mechanisms $X := f_X(\mathbf{PA}_X, \mathbf{U}_X)$. An intervention $\mathrm{do}(X \leftarrow x)$ replaces the mechanism for X by the assignment $X := x$. An *interchange intervention* (Geiger et al., 2021; 2022; 2025) instead replaces an internal variable in a base computation by the value it takes for a source input x', e.g., $\mathrm{do}(X \leftarrow X(x'))$.

Causal abstraction. Causal abstraction relates a detailed low-level model $\mathcal{M}_L$ to a simpler high-level model $\mathcal{M}_H$ through a state abstraction $\tau : \mathcal{R}_{\mathcal{X}_L} \to \mathcal{R}_{\mathcal{X}_H}$ and a corresponding map ω between admissible interventions (Rubenstein et al., 2017; Beckers & Halpern, 2019; Beckers et al., 2019). Exact abstraction requires the following diagram to commute:

$$\tau(\mathcal{M}_L(\mathcal{X}_L \mid u, \mathrm{do}(\mu))) = \mathcal{M}_H(\mathcal{X}_H \mid \tau_u(u), \mathrm{do}(\omega(\mu))) \,. \tag{1}$$

It means that *intervening, executing in $\mathcal{M}_L$, and abstracting* should result in the same state as *intervening, abstracting, and executing in $\mathcal{M}_H$*. Many variants of the consistency requirements have been proposed, we guide the reader to Zennaro (2022) for an overview.

It is also possible to quantify approximate commutativity (Beckers et al., 2019; Zennaro et al., 2023a;b). In this work, we follow *Causal Abstraction Error* (CAE) (Méloux et al., 2026) which showed the best empirical abilities when stress-tested on complex systems. For compatible low- and high-level interventions μ and $\omega(\mu)$,

$$\mathrm{CAE} = \mathbb{E}_{u,\mu}\left[D(\tau(\mathcal{M}_L(\cdot \mid u, \mathrm{do}(\mu))), \mathcal{M}_H(\cdot \mid \tau_u(u), \mathrm{do}(\omega(\mu))))\right], \tag{2}$$

for a suitable discrepancy D and evaluation distribution over contexts and admissible interventions.

Causal distillation. DIITO (Wu et al., 2022) frames distillation through causal abstraction using interchange interventions, independently patching aligned teacher and student representations with their respective source activations. We build on this perspective by learning value-level abstraction maps that transport teacher interventions into the student and enforce consistency across multiple downstream abstraction boundaries. DIITO is recovered as a special case of our general objective, as described next.

4 Intervention-Aware Compression

We view a feed-forward neural network as an SCM whose endogenous variables are layer representations and whose mechanisms are layer transformations (Geiger et al., 2021; 2022). The teacher $\mathcal{M}_L$ contains representations $\mathcal{X}_L = \{a_0, \ldots, a_K\}$ with $a_i = g_i(a_{i-1})$, while the smaller student $\mathcal{M}_H$ contains $\mathcal{X}_H = \{Z_0, \ldots, Z_{K'}\}$, $K' \leq K$, with $Z_j = f_j(Z_{j-1})$. Both receive the same input, $a_0 = Z_0 = x$, and produce distributions over the same output space.

Alignment and abstraction maps. We partition the teacher layers into K' contiguous blocks with boundaries $0 = b_0 < b_1 < \cdots < b_{K'} = K$ and align teacher representation a_{b_j} with student representation Z_j. We learn layer-wise abstraction maps

$$\tau_j : \mathbb{R}^{d_{b_j}} \to \mathbb{R}^{d'_j}, \qquad Z_j \approx \tau_j(a_{b_j}), \tag{3}$$

with $\tau_0 = \mathrm{id}$. Let $G_j = g_{b_j} \circ \cdots \circ g_{b_{j-1}+1}$ denote the computation performed by teacher block j. Causal abstraction requires the corresponding computations to approximately commute,

$$\tau_j \circ G_j \approx f_j \circ \tau_{j-1}, \tag{4}$$

including when aligned internal representations are intervened upon.

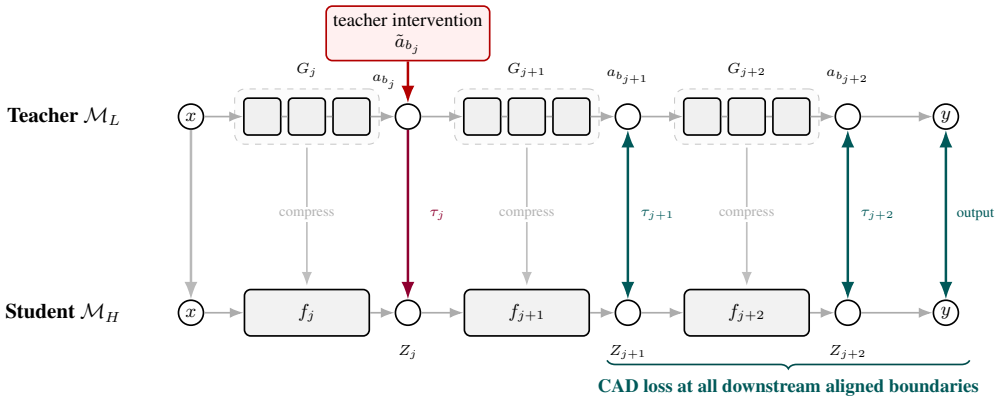

Figure 1: Illustration of the original neural network (teacher, top) and the student network (bottom)

4.1 Learning the Abstraction Maps

We anchor each abstraction map to naturally occurring teacher-student representations using

$$\mathcal{L}_{\text{map}}^{(j)} = \mathbb{E}_{x\sim\mathcal{D}}\left[\left\|\tau_j(a_{b_j}(x)) - Z_j(x)\right\|_2^2\right]. \tag{5}$$

Optimization alternates between (i) fitting the maps τ_j with teacher and student representations fixed and (ii) optimizing the student under interventional constraints with the maps fixed.

4.2 Interventional Causal Distillation

Our formulation is not restricted to activation swaps between pairs of inputs: interventions are defined directly in the teacher. For a base input x and an aligned site j, any admissible teacher state $\tilde{a}_{b_j}$ can serve as an intervention. For example, one could search by gradient ascent for a state that produces a substantial change in the teacher's downstream behavior, e.g., maximizing

$$D_{\text{out}}\big(p_L(\cdot \mid x), p_L(\cdot \mid x; \text{do}[a_{b_j} \leftarrow \tilde{a}_{b_j}])\big).$$

The corresponding student intervention is induced by the learned abstraction (Figure 1):

$$\text{do}[a_{b_j} \leftarrow \tilde{a}_{b_j}] \quad \mapsto \quad \text{do}[Z_j \leftarrow \tau_j(\tilde{a}_{b_j})]. \tag{6}$$

Interchange interventions are recovered when $\tilde{a}_{b_j} = a_{b_j}(x')$ for some source input x', while interventions at $j = 0$ correspond to perturbing the model input. In our experiments, we use interchange interventions with source inputs x' sampled uniformly from the training data, and leave optimized interventions to future work.

Causal Abstraction Distillation loss. Let μ denote a teacher intervention and $\omega(\mu)$ its induced student intervention. For an intervention at boundary j, we enforce commutativity at downstream aligned boundaries through

$$\mathcal{L}_{\text{CAD}} = \mathbb{E}_{x,\mu}\left[\sum_{k\in\mathcal{K}(\mu)} w_k\, D_k\big(\tau_k(a_{b_k}(x;\mu)),\, Z_k(x;\omega(\mu))\big)\right], \tag{7}$$

where $\mathcal{K}(\mu)$ contains the downstream abstraction boundaries at which consistency is evaluated, and w_k are scalar weights controlling the importance of each abstraction anchor.[1] This objective can be understood as optimizing CAE over deliberately selected, causally informative teacher interventions.

A unified view of distillation. Different choices of μ and $\mathcal{K}(\mu)$ recover several existing distillation objectives as special cases. With the null intervention and evaluation only at the output, Eq. 7 reduces to standard knowledge distillation (Hinton et al., 2015). Evaluating the null intervention at intermediate representations instead recovers feature-based distillation objectives such as FitNets (Romero

[1] In practice, we do not tune these and fix them to 1.

et al., 2015), Patient KD (Sun et al., 2019), and TinyBERT (Jiao et al., 2020). Intervening on the input and evaluating only at the output corresponds to distillation over augmented or adversarially perturbed inputs (Beyer et al., 2022; Goldblum et al., 2020). Finally, choosing internal interchange interventions but evaluating only their output effect recovers the setting of DIITO-style causal distillation (Wu et al., 2022). Our method considers the more general case of arbitrary teacher-induced internal interventions and evaluates their consistency across multiple downstream abstraction boundaries.

CAB, a practical variant of CAD. We use a variant of the CAD objective, denoted CAB, with two complementary terms. The first matches teacher and student outputs under paired native block interventions using temperature-scaled KL divergence. The second enforces consistency under mapped interventions at both hidden boundaries and the output:

$$\mathcal{L}_{\mathrm{CAB}} = \lambda_{\mathrm{B}}\mathcal{L}_{\mathrm{B}} + \lambda_{\mathrm{C}}\mathcal{L}_{\mathrm{CAD}}. \tag{8}$$

Here, $\mathcal{L}_{\mathrm{B}}$ (the *bridge* term) is a DIITO-style objective, where teacher and student are intervened independently with their own source activations at aligned block boundaries, without the abstraction maps, and their output distributions are matched. For $\mathcal{L}_{\mathrm{CAD}}$, we use Eq. 7 with $\mathcal{K}(\mu)$ containing all aligned hidden boundaries except the directly intervened boundary, together with the output. We use normalized mean squared error for hidden states and base-two Jensen–Shannon divergence at the output, averaging equally over the evaluated boundaries.

The complete training objective combines CAB with supervised learning and ordinary knowledge distillation:

$$\mathcal{L} = \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{KD}}\mathcal{L}_{\mathrm{KD}} + \mathcal{L}_{\mathrm{CAB}}. \tag{9}$$

5 Experiments

We evaluate our approach on task-fine-tuned encoder models across several natural language understanding tasks, and on decoder-only models from the Qwen2.5 family.

5.1 Experimental Setup

Our experiments assess whether CAB preserves teacher behavior while maintaining task performance.

Data. Our experiments are conducted on the General Language Understanding Evaluation (GLUE) benchmark (Wang et al., 2018), which comprises a diverse collection of natural language understanding tasks. Our main experiments focus on three representative tasks spanning different task categories: SST-2 for single-sentence sentiment classification, QQP for paraphrase detection, and MNLI for natural language inference. Results for the remaining GLUE tasks are reported in Appendix D.

To evaluate our approach under a different task formulation, we additionally consider the Stanford Question Answering Dataset (SQuAD 2.0) (Rajpurkar et al., 2018), which contains both answerable and unanswerable questions in an extractive question-answering setting. We further evaluate on Yelp Review Full (Zhang et al., 2016), a review-classification dataset with labels on a five-point rating scale.

Teacher and Student Models. Our primary teacher, $\mathcal{M}_L$, is a task-fine-tuned BERT-base model (Devlin et al., 2019) with 12 Transformer layers and hidden dimension 768. The primary student, $\mathcal{M}_H$, keeps the same hidden dimension but uses 6 layers, isolating depth compression. We additionally evaluate 3- and 9-layer students to study the effect of compression depth. We also tested a different compression architecture, by pairing a task-fine-tuned DeBERTaV3-base teacher (He et al., 2023) with a DeBERTaV3-xsmall student, which keeps the 12 layers but halves the hidden dimension to 384, isolating width compression. Results are reported in Appendix D.2.

For the decoder-only experiments, we use models from the Qwen2.5-Instruct family (Qwen Team et al., 2025). Specifically, we distill Qwen2.5-14B-Instruct into Qwen2.5-3B-Instruct on Yelp review classification. We perform further diagnostics on the same teacher with 0.5B, 1.5B, 3B, and 7B Qwen2.5-Instruct students to study different compression ratios. Model architectures and approximate parameter counts are summarized in Appendix Table 5.

Alignments and Abstraction Maps. To define a candidate causal abstraction and enable causal evaluation, we learn an abstraction map τ_j from each aligned teacher representation to the corresponding student state space. We adapt the FULL, MIDDLE, and LATE alignment schemes of Wu et al. (2022). FULL aligns teacher blocks with student layers throughout the network, while MIDDLE and LATE select intermediate and later teacher-student mapping. Since no ground-truth correspondence is available, each alignment and its learned maps define a candidate causal abstraction. Exact layer correspondences and examples are provided in Appendix B.

Baselines. We compare CAB against increasingly structured distillation baselines: knowledge distillation (KD) (Hinton et al., 2015), layer-wise distillation (LWD) (Romero et al., 2015), and task-aware layer-wise distillation (TED) (Liang et al., 2023). These enforce teacher-student agreement at the output, intermediate, and task-relevant representation levels.

We also include a task-specific adaptation of DIITO (Wu et al., 2022), which matches predictions after swapping activations at aligned layers. DIITO is the closest baseline to CAB, while CAB additionally optimizes the causal-abstraction objective from Section 4.

Training Procedure. To ensure a fair comparison, encoder methods differ only in their training objective. For each task, architecture, and seed, we use the same initial student, teacher, data split, and evaluation setup. Loss coefficients are tuned on small task-specific grids (Appendix C).

The pipeline has three stages. **Stage 1: Map fitting.** We freeze teacher and student and fit abstraction maps for 10 epochs at a learning rate of 10^{-3}. **Stage 2: Student training.** We fix the teacher and maps and train each student with its corresponding objective at 4×10^{-5}. **Stage 3: Evaluation-map refitting.** For each selected student, we independently refit the abstraction maps using the same procedure and partitions, with up to 2,048 fitting and 512 held-out examples.

All methods are evaluated under the same layer alignment and interventions. Seeds and coefficient grids are given in Appendix C and the Qwen protocol in Appendix I.2.

Metrics. We report task score, teacher-student agreement, $\mathrm{CAD}_{\mathrm{IO}}$, and CAD. Task score measures performance against ground-truth labels, while agreement measures how often teacher and student predictions match on unmodified inputs.

$\mathrm{CAD}_{\mathrm{IO}}$ measures robustness under controlled input perturbations by comparing teacher and student output distributions after applying the same intervention to their inputs. It therefore requires no hidden-state correspondence or abstraction map. In principle, the intervention may be any controlled input modification and in our encoder experiments, we use masking. Lower $\mathrm{CAD}_{\mathrm{IO}}$ indicates that the student more closely preserves the teacher's response to the perturbation, making it a map-independent measure of intervention-response robustness.

In contrast, CAD in Equation 7 depends on an abstraction map τ between the teacher and student representations. For CAB, this map is learned as part of the method. For KD, LWD, TED, and DIITO, no such map is learned during training, so we must fit one afterwards to compute CAD. Their CAD scores therefore depend both on the student and on the quality of this post-hoc map, and should be viewed mainly as diagnostic results. We therefore visually de-emphasize these baseline values.

5.2 Results on Encoder Models

Table 1 shows that CAB variants achieve the lowest $\mathrm{CAD}_{\mathrm{IO}}$ on MNLI-m, QQP, and SQuAD v2 while maintaining competitive task performance. On MNLI-m, $\mathrm{CAB}_{\mathrm{FULL}}$ reduces $\mathrm{CAD}_{\mathrm{IO}}$ by 9.9% relative to the best baseline on this metric (TED), with nearly unchanged accuracy. It also obtains the lowest map-dependent CAD, with the caveats on baseline CAD values discussed above. On SQuAD v2, it improves F1 by 1.07 percentage points over DIITO, while $\mathrm{CAB}_{\mathrm{LATE}}$ achieves the lowest $\mathrm{CAD}_{\mathrm{IO}}$. SST-2 is an exception: TED achieves the lowest CAD, while DIITO achieves the lowest $\mathrm{CAD}_{\mathrm{IO}}$ and highest accuracy.

Further results in the appendix support this pattern across additional tasks and architectures. On QNLI, $\mathrm{CAB}_{\mathrm{FULL}}$ reduces $\mathrm{CAD}_{\mathrm{IO}}$ relative to DIITO by 15.6% for BERT and 11.5% for DeBERTa, alongside small accuracy gains (Tables 11 and 20). Overall, CAB improves intervention-response robustness in several settings while maintaining competitive task performance, with benefits depending on the task and alignment.

Table 1: 12-layer BERT-base teacher and 6-layer students ($n = 5$). †Teacher compared with itself.

Task	Model	Task score ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
SST-2	BERT-base (teacher)	92.32	100†	0†	0†
	KD	90.80 ± 0.36	95.32 ± 0.27	0.0240 ± 0.0018	0.3396 ± 0.0033
	LWD	90.89 ± 0.35	95.32 ± 0.52	0.0252 ± 0.0018	0.1957 ± 0.0015
	TED	90.83 ± 0.51	95.21 ± 0.34	0.0246 ± 0.0020	$\mathbf{0.1777 \pm 0.0016}$
	DIITO	$\mathbf{91.03 \pm 0.40}$	$\mathbf{95.64 \pm 0.16}$	$\mathbf{0.0218 \pm 0.0016}$	0.3245 ± 0.0025
	$\mathrm{CAB}_{\mathrm{FULL}}$ (ours)	90.80 ± 0.55	95.60 ± 0.50	0.0221 ± 0.0014	0.2264 ± 0.0017
	$\mathrm{CAB}_{\mathrm{MIDDLE}}$ (ours)	90.92 ± 0.52	95.44 ± 0.55	0.0243 ± 0.0020	0.2616 ± 0.0117
	$\mathrm{CAB}_{\mathrm{LATE}}$ (ours)	90.64 ± 0.39	95.11 ± 0.31	0.0256 ± 0.0017	0.2636 ± 0.0046
MNLI-m	BERT-base (teacher)	84.84	100†	0†	0†
	KD	82.49 ± 0.30	89.04 ± 0.11	0.0576 ± 0.0027	0.3363 ± 0.0072
	LWD	82.77 ± 0.20	89.93 ± 0.11	0.0513 ± 0.0022	0.2955 ± 0.0034
	TED	83.12 ± 0.14	$\mathbf{90.36 \pm 0.24}$	0.0496 ± 0.0017	0.2311 ± 0.0029
	DIITO	82.77 ± 0.19	89.65 ± 0.18	0.0508 ± 0.0036	0.2952 ± 0.0048
	$\mathrm{CAB}_{\mathrm{FULL}}$ (ours)	83.16 ± 0.21	90.34 ± 0.15	$\mathbf{0.0447 \pm 0.0016}$	$\mathbf{0.1796 \pm 0.0086}$
	$\mathrm{CAB}_{\mathrm{MIDDLE}}$ (ours)	$\mathbf{83.18 \pm 0.15}$	90.15 ± 0.09	0.0474 ± 0.0021	0.2021 ± 0.0071
	$\mathrm{CAB}_{\mathrm{LATE}}$ (ours)	82.99 ± 0.15	89.74 ± 0.06	0.0514 ± 0.0022	0.2139 ± 0.0075
QQP	BERT-base (teacher)	91.31	100†	0†	0†
	KD	90.98 ± 0.07	94.73 ± 0.06	0.0305 ± 0.0011	0.3205 ± 0.0032
	LWD	91.02 ± 0.05	95.22 ± 0.08	0.0261 ± 0.0032	0.2529 ± 0.0020
	TED	91.03 ± 0.02	95.06 ± 0.06	0.0276 ± 0.0033	0.2312 ± 0.0016
	DIITO	91.08 ± 0.09	95.00 ± 0.09	0.0283 ± 0.0019	0.2780 ± 0.0020
	$\mathrm{CAB}_{\mathrm{FULL}}$ (ours)	91.09 ± 0.06	95.15 ± 0.02	$\mathbf{0.0240 \pm 0.0012}$	$\mathbf{0.2037 \pm 0.0010}$
	$\mathrm{CAB}_{\mathrm{MIDDLE}}$ (ours)	91.16 ± 0.06	$\mathbf{95.31 \pm 0.02}$	0.0248 ± 0.0024	0.2223 ± 0.0025
	$\mathrm{CAB}_{\mathrm{LATE}}$ (ours)	$\mathbf{91.28 \pm 0.04}$	95.14 ± 0.13	0.0252 ± 0.0019	0.2392 ± 0.0040
SQuAD v2	BERT-base (teacher)	–	100†	0†	0†
	KD	73.46 ± 0.35	74.47 ± 0.26	0.1411 ± 0.0029	0.2175 ± 0.0005
	LWD	73.44 ± 0.35	74.46 ± 0.27	0.1410 ± 0.0028	0.2177 ± 0.0005
	TED	73.46 ± 0.35	74.48 ± 0.25	0.1411 ± 0.0029	0.2169 ± 0.0005
	DIITO	73.74 ± 0.23	75.20 ± 0.15	0.1294 ± 0.0029	0.2060 ± 0.0006
	$\mathrm{CAB}_{\mathrm{FULL}}$ (ours)	$\mathbf{74.81 \pm 0.10}$	74.97 ± 0.34	0.1316 ± 0.0025	$\mathbf{0.1884 \pm 0.0011}$
	$\mathrm{CAB}_{\mathrm{MIDDLE}}$ (ours)	74.55 ± 0.24	$\mathbf{75.26 \pm 0.28}$	0.1292 ± 0.0025	0.2002 ± 0.0011
	$\mathrm{CAB}_{\mathrm{LATE}}$ (ours)	73.84 ± 0.24	75.01 ± 0.29	$\mathbf{0.1281 \pm 0.0025}$	0.2131 ± 0.0006

Layerwise intervention diagnostic. We test whether matching the teacher's internal response to interventions also leads to similar output responses. For each aligned layer pair, we compare how an intervention changes the teacher and student relative to their normal computation. Hidden-response error measures differences in downstream hidden-state changes after mapping them into the same representation space, while output-response error measures differences in class-probability changes. Lower error means the teacher and student react more similarly to the intervention.

On SST-2, we compare $\mathrm{CAB}_{\mathrm{FULL}}$ with DIITO using the same intervention pairs at five aligned intervention sites (Figure 2). Hidden responses are compared through abstraction maps refit after training for both students (Stage 3). Because CAB is trained to be consistent under such maps, its advantage in hidden-response error is partly expected by construction. The interesting question is whether it also reaches the output. CAB has lower mean hidden-response error at all five sites, indicating that its downstream hidden states respond more similarly to the teacher under the learned maps. However, this internal advantage does not necessarily translate into closer output responses. The distinction is relevant to the main SST-2 results: DIITO achieves slightly lower $\mathrm{CAD}_{\mathrm{IO}}$ than $\mathrm{CAB}_{\mathrm{FULL}}$, despite CAB's closer internal responses in this diagnostic.

Possible explanations for this gap are the choice of interventions, the layer alignment, and limitations of the abstraction maps. An objective ablation in Appendix H further isolates the contribution of hidden-boundary supervision to the improved internal intervention-response consistency. Reconstruction diagnostics in Appendix E complement this analysis by assessing hidden-state correspondence.

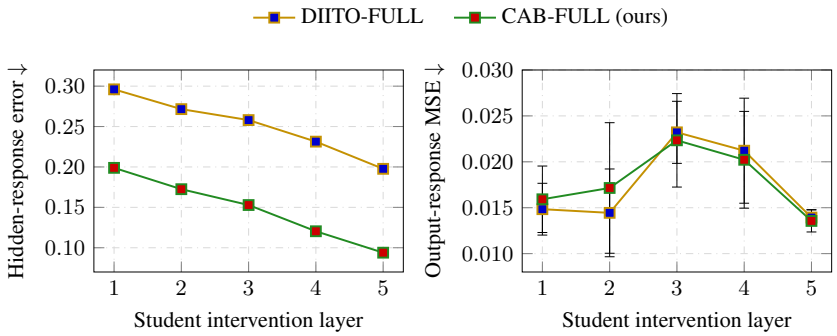

Figure 2: Layerwise intervention diagnostics on SST-2 for CAB_{FULL} and DIITO.

Table 2: Results for 6-layer BERT students (mean ± SD over seeds 42, 43, 46; %).

	Trained on QQP				Trained on MNLI			
Method	QQP Acc	QQP F1	MRPC Acc	MRPC F1	MNLI-m	MNLI-mm	SNLI	QNLI
DIITO	91.08 ± 0.05	87.92 ± 0.05	64.54 ± 2.84	69.86 ± 3.05	82.77 ± 0.26	83.12 ± 0.12	78.42 ± 0.14	55.82 ± 0.69
CAB_{FULL} (ours)	91.08 ± 0.07	87.78 ± 0.13	65.77 ± 0.14	71.90 ± 0.29	83.05 ± 0.21	83.44 ± 0.16	78.60 ± 0.21	55.48 ± 0.37
$\text{CAB}_{\text{MIDDLE}}$ (ours)	91.12 ± 0.02	88.03 ± 0.03	66.67 ± 0.42	73.01 ± 0.27	**83.20** ± 0.20	**83.47** ± 0.05	**78.77** ± 0.14	**57.61** ± 0.75
CAB_{LATE} (ours)	**91.31** ± 0.04	**88.28** ± 0.06	**68.38** ± 1.61	**74.60** ± 1.71	82.93 ± 0.16	83.34 ± 0.12	78.30 ± 0.20	57.44 ± 1.28

Compression trade-off. We distill a fixed 12-layer teacher into 9-, 6-, and 3-layer students under matched training budgets (Appendix F, Figure 4). The 6- and 9-layer CAB students achieve the same mean accuracy (62.45%), but the 6-layer student has higher CAD_{IO} (0.00989 vs. 0.00407). Further compression to three layers reduces accuracy and increases CAD_{IO} for both CAB and DIITO. These results indicate a practical limit to depth compression: intervention-response robustness can deteriorate before task accuracy declines, while more aggressive compression compromises both.

Generalization. To assess whether CAB improves cross-dataset generalization relative to DIITO, we evaluate models trained on one dataset on related datasets within the same task category. For paraphrase identification, we transfer QQP-trained models, trained on question pairs, to MRPC, which contains sentence pairs from news articles. For natural language inference, we evaluate MNLI-trained models on SNLI, which contains premise–hypothesis pairs derived from image captions, and QNLI, a question–sentence entailment task. For QNLI, we map both *neutral* and *contradiction* predictions to *not entailment*.

Table 2 shows that CAB can improve transfer performance over DIITO, with the gains depending on layer alignment. CAB_{LATE} achieves the strongest MRPC results, improving accuracy by 3.84 percentage points and F1 by 4.74 points. $\text{CAB}_{\text{MIDDLE}}$ improves both SNLI and QNLI accuracy by 0.35 and 1.79 points, respectively. These results show that CAB can improve generalization to related datasets, although the best alignment depends on the target dataset. The full comparison with additional baselines is reported in Appendix G (Table 26).

5.3 CAB AT LLM SCALE

We next explore whether CAB extends to larger models, using Qwen2.5-Instruct on Yelp five-class review rating. We first compare frozen models of different sizes and a randomly initialized control to examine the relationship between task performance, teacher agreement, and intervention-response robustness. We then evaluate whether CAB improves teacher-student consistency while preserving task performance when distilling a 14B teacher into a 3B student.

Pretrained and Random-Weight Students. Table 3 compares frozen pretrained models and a random-weight control on a shared panel of 500 Yelp reviews. We report rating RMSE, accuracy, and clean teacher agreement, together with CAD_{IO} over 1,119 synonym, deletion, and antonym edits.

Pretrained models outperform the random-weight control on all reported measures. Among pretrained students, clean teacher agreement increases with model size, whereas CAD_{IO} does not decrease monotonically: the 3B model has higher clean agreement than the 1.5B model but greater

Table 3: Frozen Qwen2.5-Instruct models and a random-weight 3B control on Yelp.

Model	**RMSE**↓	**Acc. (%)**↑	**Agr. (%)**↑	$\mathrm{CAD_{IO}}$ ↓
Mean-rating baseline	1.4142	–	–	–
14B (teacher)	0.5964	65.2	–	–
0.5B	0.8318	48.4	51.8	0.3688
1.5B	1.1903	41.4	52.8	0.3619
3B	0.8185	51.6	58.2	0.3843
7B	0.6560	61.2	66.8	0.2754
Random-weight 3B	1.5522	20.0	20.2	0.5993

divergence on edited inputs. The 7B model achieves the best student task performance and lowest $\mathrm{CAD_{IO}}$. These results further motivate evaluating teacher-student consistency on intervened inputs alongside clean task performance. We therefore next test whether CAB improves this consistency during distillation.

Distillation at LLM Scale. To evaluate CAB at larger model scales, we distill the same Qwen2.5-14B-Instruct teacher into a Qwen2.5-3B-Instruct student on five-class Yelp review rating using FULL layer alignment. Both KD and CAB initialize the student from the same pretrained, instruction-tuned Qwen2.5-3B-Instruct checkpoint. Distillation therefore further fine-tunes an existing smaller model. We compare standard KD with CAB, training both methods on the same 2,000 reviews. Training settings, checkpoint selection, and layer alignment are detailed in Appendix I.2.

As shown in Table 4, CAB improves average accuracy by 0.40 percentage points, clean teacher agreement by 0.60 points, and edited agreement by 2.00 points. It also reduces $\mathrm{CAD_{IO}}$ by approximately 11% relative to KD, indicating closer teacher-student output distributions on intervened inputs.

Table 4: Yelp results, reported as mean $\pm$ sample standard deviation across three training seeds ($n = 3$).

Method	Accuracy (%) ↑	Agreement (%) ↑	Edited agreement (%) ↑	$\mathrm{CAD_{IO}}$ ↓
KD	64.07 ± 0.12	83.60 ± 3.12	80.10 ± 1.81	0.1349 ± 0.0115
CAB	64.47 ± 1.70	84.20 ± 2.09	82.10 ± 1.34	0.1200 ± 0.0026

CAB thus applies to 14B-to-3B distillation, improving consistency on edited inputs ($\mathrm{CAD_{IO}}$ -11%, edited agreement $+2.0$ points) at comparable task accuracy.

6 Discussion

We formulate robust model distillation through causal abstraction and propose a general loss that recovers several existing distillation objectives as special cases. Across our experiments, this approach can improve intervention-response robustness while maintaining competitive task performance, although the gains vary across tasks and metrics. Our diagnostics further show that internal and output-level intervention-response consistency need not improve together: closer hidden-state responses under the learned abstraction maps do not necessarily translate into closer output responses. Interventional evaluation therefore provides information about student behavior that task performance alone does not capture. These findings remain conditional on the learned maps, evaluated tasks, and sampled interventions, motivating broader evaluation and further study of when internal intervention-response consistency transfers to output behavior.

While this paper uses causal abstraction to guide distillation, future work could explore the reverse direction: using compression to search for causally faithful explanations. Optimizing variants of causal abstraction error within more interpretable model classes could offer a route toward discrete, semantically grounded representations and explanatory structures such as attribution graphs (Ameisen et al., 2025).

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on experimental design, assist with implementing methods and summarize and interpret experimental results. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, assist with translation, or clean and reformat datasets. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Emmanuel Ameisen, Jack Lindsey, Adam Pearce, Wes Gurnee, Nicholas L. Turner, Brian Chen, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Chris Olah, and Joshua Batson. Circuit tracing: Revealing computational graphs in language models. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

Amir Asiaee. Causal mechanism reduction: Mechanism replacement for neural network pruning and abstraction, 2026. URL https://arxiv.org/abs/2602.24266.

Sander Beckers and Joseph Y. Halpern. Abstracting causal models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 2678–2685, 2019. doi: 10.1609/aaai.v33i01.33012678.

Sander Beckers, Frederick Eberhardt, and Joseph Y. Halpern. Approximate causal abstraction, 2019. URL https://arxiv.org/abs/1906.11583.

Lucas Beyer, Xiaohua Zhai, Amelie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10915–10924, Los Alamitos, CA, USA, June 2022. IEEE Computer Society. doi: 10.1109/CVPR52688.2022.01065. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.01065.

Xiang Deng and Zhongfei Zhang. Comprehensive knowledge distillation with causal intervention. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 22158–22170. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/b9f35816f460ab999cbc168c4da26ff3-Paper.pdf.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019. doi: 10.18653/v1/N19-1423.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 9574–9586, 2021. URL https://arxiv.org/abs/2106.02997.

Atticus Geiger, Zhengxuan Wu, Hanson Lu, Josh Rozner, Elisa Kreiss, Thomas Icard, Noah Goodman, and Christopher Potts. Inducing causal structure for interpretable neural networks. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pp. 7324–7338, 2022. URL https://proceedings.mlr.press/v162/geiger22a.html.

Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, and Thomas Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability. *Journal of Machine Learning Research*, 26(83):1–64, 2025. URL http://jmlr.org/papers/v26/23-0058.html.

Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. Adversarially robust distillation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3996–4003, April 2020. ISSN 2159-5399. doi: 10.1609/aaai.v34i04.5816. URL http://dx.doi.org/10.1609/aaai.v34i04.5816.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2111.09543.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, Jack Rae, Oriol Vinyals, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30016–30030, 2022. URL https://papers.nips.cc/paper_files/paper/2022/hash/c1e2faff6f588870935f114ebe04a3e5-Abstract-Conference.html.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4163–4174, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.372. URL https://aclanthology.org/2020.findings-emnlp.372.

Xiusi Li, Sékou-Oumar Kaba, and Siamak Ravanbakhsh. On the identifiability of causal abstractions. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 258 of *Proceedings of Machine Learning Research*, pp. 3241–3249, 2025. URL https://arxiv.org/abs/2503.10834.

Chen Liang, Simiao Zuo, Qingru Zhang, Pengcheng He, Weizhu Chen, and Tuo Zhao. Less is more: Task-aware layer-wise distillation for language model compression. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 20852–20867, 2023. URL https://proceedings.mlr.press/v202/liang23j.html.

Maxime Méloux, Silviu Maniu, François Portet, and Maxime Peyrard. Everything, everywhere, all at once: Is mechanistic interpretability identifiable? In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2502.20914.

Maxime Méloux, Tiago Pimentel, François Portet, and Maxime Peyrard. Validating causal abstraction metrics on simulated complex systems, 2026. URL https://arxiv.org/abs/2607.00267. ICML 2026 Workshop on Mechanistic Interpretability.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, Cambridge, UK, 2 edition, 2009. ISBN 9780521895606.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. A practical review of mechanistic interpretability for transformer-based language models, 2025. URL https://arxiv.org/abs/2407.02646.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, 2018. doi: 10.18653/v1/P18-2124.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for thin deep nets. In *3rd International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1412.6550.

Paul K. Rubenstein, Sebastian Weichwald, Stephan Bongers, Joris M. Mooij, Dominik Janzing, Moritz Grosse-Wentrup, and Bernhard Schölkopf. Causal consistency of structural equation models, 2017. URL https://arxiv.org/abs/1707.00819.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for BERT model compression. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4323–4332, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1441. URL https://aclanthology.org/D19-1441.

Denis Sutter, Julian Minder, Thomas Hofmann, and Tiago Pimentel. The non-linear representation dilemma: Is causal abstraction enough for mechanistic interpretability? In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2507.08802.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446.

Zhengxuan Wu, Atticus Geiger, Joshua Rozner, Elisa Kreiss, Hanson Lu, Thomas Icard, Christopher Potts, and Noah Goodman. Causal distillation for language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 4288–4295, 2022. doi: 10.18653/v1/2022.naacl-main.318.

Fabio Massimo Zennaro. Abstraction between structural causal models: A review of definitions and properties. *arXiv preprint arXiv:2207.08603*, 2022.

Fabio Massimo Zennaro, Máté Drávucz, Geanina Apachitei, W Dhammika Widanage, and Theodoros Damoulas. Jointly learning consistent causal abstractions over multiple interventional distributions. In *Conference on Causal Learning and Reasoning*, pp. 88–121. PMLR, 2023a.

Fabio Massimo Zennaro, Paolo Turrini, and Theodoros Damoulas. Quantifying Consistency and Information Loss for Causal Abstraction Learning. In *Electronic proceedings of IJCAI 2023*, volume 5, pp. 5750–5757, August 2023b. doi: 10.24963/ijcai.2023/638. URL https://www.ijcai.org/proceedings/2023/638. ISSN: 1045-0823.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification, 2016. URL https://arxiv.org/abs/1509.01626.

Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. A survey on model compression for large language models. *Transactions of the Association for Computational Linguistics*, 12: 1556–1577, 2024. URL https://doi.org/10.1162/tacl_a_00704.

APPENDIX

A MODEL ARCHITECTURES

We evaluate our method across several model families and scales. Table 5 summarizes the architectures used in our experiments.

Model	Layers	Hidden dimension	Parameters
BERT-base teacher	12	768	109.5M
BERT student	9	768	88.2M
BERT student	6	768	67.0M
BERT student	3	768	45.7M
DeBERTaV3-base	12	768	183M
DeBERTaV3-xsmall	12	384	70M
Qwen2.5-0.5B-Instruct	24	896	0.49B
Qwen2.5-1.5B-Instruct	28	1536	1.54B
Qwen2.5-3B-Instruct	36	2048	3.09B
Qwen2.5-7B-Instruct	28	3584	7.62B
Qwen2.5-14B-Instruct	48	5120	14.7B

Table 5: Model architectures used in our experiments. Parameter counts are approximate and include embeddings. Task-specific prediction heads introduce minor variations in the encoder-model totals. M and B denote millions and billions.

B Layer Alignments and Abstraction Maps

Full alignment. For a teacher with L layers and a student with H layers, where $L \geq H$, we partition the teacher into contiguous blocks

$$B_j = \left\{ \left\lfloor \frac{(j-1)L}{H} \right\rfloor + 1, \ldots, \left\lfloor \frac{jL}{H} \right\rfloor \right\}, \qquad j = 1, \ldots, H.$$

Student layer j corresponds to teacher block B_j. Its abstraction map acts on the final hidden state of that block, at teacher layer $\max B_j$. This construction also supports depth ratios that are not integers.

Middle and late alignments. Table 6 lists the selected correspondences for the 12-layer teacher and 6-layer student. MIDDLE and LATE replace the teacher blocks at the selected student locations with the listed teacher layers and restrict training interventions to those locations. Other block correspondences retain their FULL definitions.

Alignment	Teacher layers	Student layers
FULL	$\{2j-1, 2j\}$	$j = 1, \ldots, 6$
MIDDLE	7	4
LATE	10; 11	4; 5

Table 6: Structural alignments for a 12-layer teacher and 6-layer student. Semicolons separate paired correspondences. For FULL, maps use teacher block endpoints $2, 4, 6, 8, 10, 12$, while training interventions select student layers $1, \ldots, 5$.

C Training and Evaluation Protocol

For each baseline, we select the loss coefficient from the task-specific grids in Table 7; all reported results use the same training and evaluation protocol across methods. BERT GLUE and SQuAD results average seeds 42–46 ($n = 5$), except MRPC (seeds 42–44, $n = 3$); DeBERTa results average three seeds (42–44).

Task	KD	LWD / TED	DIITO
SST-2, RTE, WNLI	$\{5, 20, 80\}$	$\{5, 20, 80\}$	$\{5, 20, 80\}$
QNLI, MRPC	$\{5, 20, 80\}$	$\{25, 100, 400\}$	$\{5, 20, 80\}$
QQP	$\{5, 20, 80\}$	$\{50, 200, 600\}$	$\{5, 20, 80\}$
MNLI	$\{1.25, 5, 20\}$	$\{50, 200, 600\}$	$\{5, 20, 80\}$
CoLA	$\{0, 5, 20\}$	$\{2.5, 10, 40\}$	$\{5, 20, 80\}$
SQuAD 2.0	$\{5, 20, 80\}$	$\{5, 20, 80\}$	$\{5, 20, 80\}$

Table 7: Task-specific coefficient grids used for baseline tuning.

C.1 Coefficient Selection

We select loss coefficients using task-specific pilot searches on training-derived tuning partitions. The baseline search grids are listed in Table 8. These grids are implementation-specific search ranges rather than exact reproductions of the original papers' settings.

Task	KD	LWD / TED	DIITO
SST-2	$\{5, 20, 80\}$	$\{5, 20, 80\}$	$\{5, 20, 80\}$
QNLI	$\{5, 20, 80\}$	$\{25, 100, 400\}$	$\{5, 20, 80\}$
QQP	$\{5, 20, 80\}$	$\{50, 200, 600\}$	$\{5, 20, 80\}$
MNLI	$\{1.25, 5, 20\}$	$\{50, 200, 600\}$	$\{5, 20, 80\}$
MRPC	$\{5, 20, 80\}$	$\{25, 100, 400\}$	$\{5, 20, 80\}$
CoLA	$\{0, 5, 20\}$	$\{2.5, 10, 40\}$	$\{5, 20, 80\}$
SQuAD 2.0	$\{5, 20, 80\}$	$\{5, 20, 80\}$	$\{5, 20, 80\}$

Table 8: Baseline coefficient search grids. LWD and TED are tuned separately despite sharing candidate values.

In the causal-strength pilot, CAB uses FULL alignment and fixes $\lambda_{\mathrm{B}} = 20$ while searching $\lambda_{\mathrm{C}} \in \{1, 4, 10\}$. The corresponding pilot recipe uses seed 99, a 5% tuning holdout, and 10% of the remaining training examples, with two epochs of method-specific student training.

D GLUE Results

Reporting convention. Across the appendix, $\mathrm{CAD}_{\mathrm{IO}}$ is the primary intervention-robustness metric. It compares teacher and student responses to the same input perturbations and does not require a learned hidden-state map. The map-dependent CAD diagnostic is reported in the final column. For CAB, this quantity uses the abstraction maps learned as part of the method; for the baselines, the maps are fitted post hoc for evaluation, so their values (in gray) should be interpreted as secondary diagnostics rather than directly method-intrinsic quantities. In all tables, the best value per column is in **bold** and the second best is underlined. CAD is directly defined by the learned abstraction maps for CAB. For baselines, shown in gray, an auxiliary map is fitted post hoc for evaluation.

D.1 BERT

Method	Acc. (%) ↑	Agree. (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
Matched validation				
BERT-base (teacher)	84.84	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	82.49 ± 0.30	89.04 ± 0.11	0.0576 ± 0.0027	0.3363 ± 0.0072
LWD	82.77 ± 0.20	89.93 ± 0.11	0.0513 ± 0.0022	0.2955 ± 0.0034
TED	83.12 ± 0.14	$\mathbf{90.36} \pm 0.24$	0.0496 ± 0.0017	0.2311 ± 0.0029
DIITO	82.77 ± 0.19	89.65 ± 0.18	0.0508 ± 0.0036	0.2952 ± 0.0048
$\mathrm{CAB}_{\mathrm{FULL}}$	$\underline{83.16} \pm 0.21$	$\underline{90.34} \pm 0.15$	$\mathbf{0.0447} \pm 0.0016$	$\mathbf{0.1796} \pm 0.0086$
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	$\mathbf{83.18} \pm 0.15$	90.15 ± 0.09	$\underline{0.0474} \pm 0.0021$	$\underline{0.2021} \pm 0.0071$
$\mathrm{CAB}_{\mathrm{LATE}}$	82.99 ± 0.15	89.74 ± 0.06	0.0514 ± 0.0022	0.2139 ± 0.0075
Mismatched validation				
BERT-base (teacher)	84.56	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	82.90 ± 0.14	88.46 ± 0.17	0.0689 ± 0.0033	0.3307 ± 0.0071
LWD	83.29 ± 0.20	89.31 ± 0.36	0.0628 ± 0.0030	0.2941 ± 0.0033
TED	$\mathbf{83.52} \pm 0.06$	$\mathbf{89.91} \pm 0.20$	0.0574 ± 0.0034	0.2290 ± 0.0027
DIITO	83.14 ± 0.11	89.19 ± 0.13	0.0613 ± 0.0031	0.2924 ± 0.0053
$\mathrm{CAB}_{\mathrm{FULL}}$	$\underline{83.50} \pm 0.17$	$\underline{89.89} \pm 0.20$	$\mathbf{0.0502} \pm 0.0028$	$\mathbf{0.1796} \pm 0.0087$
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	83.49 ± 0.08	89.88 ± 0.13	$\underline{0.0529} \pm 0.0021$	$\underline{0.2016} \pm 0.0069$
$\mathrm{CAB}_{\mathrm{LATE}}$	83.34 ± 0.08	89.23 ± 0.20	0.0576 ± 0.0020	0.2125 ± 0.0074

Table 9: BERT students on MNLI validation: mean ± sample SD over $n = 5$ seeds.

MNLI Across both MNLI validation splits, CAB improves intervention robustness while preserving task performance. On matched validation, CAB_{FULL} obtains the lowest CAD_{IO} (0.0447), a 9.9% reduction relative to the best baseline, TED (0.0496), while its accuracy remains unchanged (83.16% vs. 83.12%). On mismatched validation, CAB_{FULL} again gives the lowest CAD_{IO} (0.0502), a 12.6% reduction relative to TED (0.0574), while its accuracy is within 0.02 percentage points of the best (TED, 83.52%). All students remain about 1.1–1.7 points below the teacher (84.84% matched, 84.56% mismatched).

Method	Accuracy (%) ↑	F1 (%) ↑	Agreement (%) ↑	CAD_{IO} ↓	CAD ↓
BERT-base	91.13	88.10	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	90.98 ± 0.07	87.80 ± 0.10	94.73 ± 0.06	0.0305 ± 0.0011	0.3205 ± 0.0032
LWD	91.02 ± 0.05	87.84 ± 0.05	$\underline{95.22} \pm 0.08$	0.0261 ± 0.0032	0.2529 ± 0.0020
TED	91.03 ± 0.02	87.85 ± 0.04	95.06 ± 0.06	0.0276 ± 0.0033	0.2312 ± 0.0016
DIITO	91.08 ± 0.09	87.92 ± 0.09	95.00 ± 0.09	0.0283 ± 0.0019	0.2780 ± 0.0020
CAB_{FULL}	91.09 ± 0.06	87.81 ± 0.13	95.15 ± 0.02	$\mathbf{0.0240} \pm 0.0012$	$\mathbf{0.2037} \pm 0.0010$
$\text{CAB}_{\text{MIDDLE}}$	$\underline{91.16} \pm 0.06$	$\underline{88.08} \pm 0.07$	$\mathbf{95.31} \pm 0.02$	$\underline{0.0248} \pm 0.0024$	$\underline{0.2223} \pm 0.0025$
CAB_{LATE}	$\mathbf{91.28} \pm 0.04$	$\mathbf{88.25} \pm 0.06$	95.14 ± 0.13	0.0252 ± 0.0019	0.2392 ± 0.0040

Table 10: BERT students on QQP validation: mean ± sample SD over $n = 5$ seeds.

QQP On QQP, CAB improves both task performance and intervention robustness. CAB_{FULL} achieves the lowest CAD_{IO} (0.0240), compared with 0.0261 for the strongest baseline on this metric, while CAB_{LATE} obtains the highest accuracy (91.28%) and F1 (88.25%). This indicates that the robustness gain does not require sacrificing clean-task performance.

Method	Accuracy (%) ↑	Agreement (%) ↑	CAD_{IO} ↓	CAD ↓
BERT-base	91.49	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	88.59 ± 0.32	91.21 ± 0.30	0.0582 ± 0.0037	0.3872 ± 0.0085
LWD	88.61 ± 0.20	91.11 ± 0.20	0.0607 ± 0.0040	0.3635 ± 0.0042
TED	88.74 ± 0.16	91.35 ± 0.21	0.0599 ± 0.0044	0.3528 ± 0.0046
DIITO	89.30 ± 0.20	91.92 ± 0.22	0.0510 ± 0.0017	0.3702 ± 0.0044
CAB_{FULL}	$\mathbf{89.54} \pm 0.18$	$\mathbf{92.48} \pm 0.12$	$\mathbf{0.0430} \pm 0.0022$	$\mathbf{0.2520} \pm 0.0094$
$\text{CAB}_{\text{MIDDLE}}$	$\underline{89.40} \pm 0.16$	$\underline{92.19} \pm 0.14$	$\underline{0.0448} \pm 0.0019$	$\underline{0.2766} \pm 0.0034$
CAB_{LATE}	89.15 ± 0.23	91.96 ± 0.20	0.0482 ± 0.0031	0.2791 ± 0.0056

Table 11: BERT students on QNLI validation: mean ± sample SD over $n = 5$ seeds.

QNLI The QNLI results show one of the clearest robustness gains. CAB_{FULL} reduces CAD_{IO} from 0.0510 for DIITO to 0.0430, a 15.6% reduction, while also improving accuracy from 89.30% to 89.54% and agreement from 91.92% to 92.48%. $\text{CAB}_{\text{MIDDLE}}$ is second best on all three measures.

Method	Accuracy (%) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
BERT-base	92.78	100†	0†	0†
KD	90.80 ± 0.36	95.32 ± 0.27	0.0240 ± 0.0018	0.3396 ± 0.0033
LWD	90.89 ± 0.35	95.32 ± 0.52	0.0252 ± 0.0018	0.1957 ± 0.0015
TED	90.83 ± 0.51	95.21 ± 0.34	0.0246 ± 0.0020	**0.1777** ± 0.0016
DIITO	**91.03** ± 0.40	**95.64** ± 0.16	**0.0218** ± 0.0016	0.3245 ± 0.0025
$\mathrm{CAB}_{\mathrm{FULL}}$	90.80 ± 0.55	95.60 ± 0.50	0.0221 ± 0.0014	0.2264 ± 0.0017
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	90.92 ± 0.52	95.44 ± 0.55	0.0243 ± 0.0020	0.2616 ± 0.0117
$\mathrm{CAB}_{\mathrm{LATE}}$	90.64 ± 0.39	95.11 ± 0.31	0.0256 ± 0.0017	0.2636 ± 0.0046

Table 12: BERT students on SST-2 validation: mean ± sample SD over $n = 5$ seeds.

SST-2 SST-2 is an exception to the broader pattern. DIITO obtains the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0218), with $\mathrm{CAB}_{\mathrm{FULL}}$ close behind at 0.0221, a difference well within one standard deviation; DIITO also has the highest accuracy (91.03%) and agreement (95.64%). All students remain about 1.8–2.1 points below the teacher (92.78%). On CAD, the representation-matching baselines TED and LWD score lowest, and $\mathrm{CAB}_{\mathrm{FULL}}$ is the best CAB variant (0.2264).

Method	MCC ($\times 100$) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
BERT-base	60.39	100†	0†	0†
KD	35.52 ± 2.50	82.07 ± 0.42	0.0874 ± 0.0223	0.3865 ± 0.0214
LWD	36.02 ± 1.65	81.50 ± 1.05	0.0900 ± 0.0286	**0.2008** ± 0.0097
TED	37.38 ± 1.63	82.07 ± 0.80	0.0978 ± 0.0097	0.2080 ± 0.0083
DIITO	39.17 ± 2.19	**84.47** ± 0.85	0.0446 ± 0.0039	0.3322 ± 0.0088
$\mathrm{CAB}_{\mathrm{FULL}}$	**40.08** ± 2.92	84.39 ± 1.21	**0.0375** ± 0.0025	0.2956 ± 0.0104
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	39.57 ± 1.55	84.07 ± 1.22	0.0405 ± 0.0040	0.3219 ± 0.0103
$\mathrm{CAB}_{\mathrm{LATE}}$	39.57 ± 0.95	84.03 ± 0.65	0.0430 ± 0.0035	0.3423 ± 0.0118

Table 13: BERT students on CoLA validation: mean ± sample SD over $n = 5$ seeds.

CoLA On CoLA, $\mathrm{CAB}_{\mathrm{FULL}}$ achieves both the best MCC (40.08) and the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0375). Relative to DIITO, this corresponds to a 16.0% reduction in $\mathrm{CAD}_{\mathrm{IO}}$, while $\mathrm{CAB}_{\mathrm{MIDDLE}}$ and $\mathrm{CAB}_{\mathrm{LATE}}$ also remain below all non-DIITO baselines on the robustness metric.

Method	F1 (%) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
BERT-base	89.63	100†	0†	0†
KD	86.77 ± 0.22	86.11 ± 1.63	0.0633 ± 0.0048	0.3531 ± 0.0286
LWD	87.59 ± 0.83	87.17 ± 1.84	0.0651 ± 0.0067	**0.2377** ± 0.0234
TED	87.31 ± 0.71	87.01 ± 1.47	0.0616 ± 0.0048	0.2537 ± 0.0276
DIITO	87.17 ± 0.70	86.52 ± 3.24	0.0552 ± 0.0149	0.3373 ± 0.0243
$\mathrm{CAB}_{\mathrm{FULL}}$	87.67 ± 0.42	87.75 ± 1.47	**0.0539** ± 0.0073	0.2917 ± 0.0191
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	**87.80** ± 0.50	**88.07** ± 1.13	0.0569 ± 0.0072	0.3104 ± 0.0180
$\mathrm{CAB}_{\mathrm{LATE}}$	86.89 ± 0.44	86.93 ± 2.21	0.0619 ± 0.0103	0.3262 ± 0.0271

Table 14: BERT students on MRPC validation: mean ± sample SD over $n = 3$ seeds.

MRPC On MRPC, $\mathrm{CAB}_{\mathrm{MIDDLE}}$ obtains the highest F1 (87.80%) and agreement (88.07%), and $\mathrm{CAB}_{\mathrm{FULL}}$ the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0539), slightly below DIITO (0.0552). These differences are small

relative to the seed variation (DIITO's $\mathrm{CAD}_{\mathrm{IO}}$ SD is 0.0149), so on MRPC CAB matches the strongest baselines rather than clearly surpassing them.

Method	Spearman ($\times 100$) ↑	Teacher MSE ↓	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
BERT-base	87.56	$0^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	86.75 ± 0.18	0.1741 ± 0.0137	0.0063 ± 0.0007	0.2732 ± 0.0200
LWD	85.29 ± 0.30	0.3073 ± 0.0365	0.0117 ± 0.0014	$\mathbf{0.0704} \pm 0.0020$
TED	$\underline{86.86} \pm 0.16$	0.1718 ± 0.0116	0.0064 ± 0.0007	$\underline{0.2143} \pm 0.0202$
DIITO	86.70 ± 0.23	0.1618 ± 0.0118	0.0058 ± 0.0006	0.2599 ± 0.0172
$\mathrm{CAB}_{\mathrm{FULL}}$	$\mathbf{86.94} \pm 0.09$	$\mathbf{0.1247} \pm 0.0068$	$\mathbf{0.0052} \pm 0.0004$	0.2531 ± 0.0155
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	86.78 ± 0.20	0.1401 ± 0.0073	0.0058 ± 0.0005	0.2623 ± 0.0157
$\mathrm{CAB}_{\mathrm{LATE}}$	86.70 ± 0.12	$\underline{0.1379} \pm 0.0087$	$\underline{0.0057} \pm 0.0003$	0.2565 ± 0.0161

Table 15: BERT students on STS-B validation: mean $\pm$ sample SD over $n = 5$ seeds. STS-B is a regression task, so teacher MSE (squared difference from the teacher's raw score) replaces class agreement.

STS-B For STS-B, $\mathrm{CAB}_{\mathrm{FULL}}$ performs best on both correlation with the ground truth and consistency with the teacher: it obtains the highest Spearman correlation (86.94), the lowest teacher MSE (0.1247, 23% below the best baseline, DIITO), and the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0052, a 10.2% reduction relative to DIITO). It is also the most stable method across seeds (Spearman SD 0.09). LWD obtains the lowest CAD (0.0704) but the worst score on every other metric, again showing that low map-dependent CAD alone does not indicate a better student.

Method	Accuracy (%) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
BERT-base (teacher)	61.37	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	59.42 ± 1.59	81.01 ± 1.32	0.0140 ± 0.0016	0.4040 ± 0.0054
LWD	53.00 ± 0.65	65.78 ± 3.87	0.0179 ± 0.0025	$\mathbf{0.0358} \pm 0.0096$
TED	50.76 ± 2.70	56.46 ± 15.15	0.0184 ± 0.0025	$\underline{0.0395} \pm 0.0020$
DIITO	59.35 ± 0.83	$\underline{81.52} \pm 0.78$	$\mathbf{0.0090} \pm 0.0012$	0.3905 ± 0.0048
$\mathrm{CAB}_{\mathrm{FULL}}$	$\underline{60.72} \pm 1.15$	80.87 ± 1.69	0.0109 ± 0.0024	0.3786 ± 0.0042
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	$\mathbf{60.94} \pm 1.68$	$\underline{81.52} \pm 1.79$	$\underline{0.0107} \pm 0.0015$	0.3831 ± 0.0036
$\mathrm{CAB}_{\mathrm{LATE}}$	59.86 ± 1.41	$\mathbf{81.88} \pm 1.10$	0.0109 ± 0.0017	0.3861 ± 0.0019

Table 16: BERT students on RTE validation: mean $\pm$ sample SD over $n = 5$ seeds.

RTE RTE highlights why the gray baseline values of the map-dependent CAD should be interpreted cautiously. LWD and TED obtain very low internal CAD values, yet their accuracies fall to 53.00% and 50.76%. By contrast, $\mathrm{CAB}_{\mathrm{MIDDLE}}$ achieves the highest accuracy (60.94%), although DIITO retains the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0090). Thus, low map-dependent CAD alone is not evidence of a useful distilled student.

Method	Exact match (%) ↑	F1 (%) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
BERT-base (teacher)	72.42	75.87	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	70.03 ± 0.31	73.46 ± 0.35	74.47 ± 0.26	0.1411 ± 0.0029	0.2175 ± 0.0005
LWD	70.02 ± 0.30	73.44 ± 0.35	74.46 ± 0.27	0.1410 ± 0.0028	0.2177 ± 0.0005
TED	70.03 ± 0.28	73.46 ± 0.35	74.48 ± 0.25	0.1411 ± 0.0029	0.2169 ± 0.0005
DIITO	70.30 ± 0.22	73.74 ± 0.23	$\underline{75.20} \pm 0.15$	0.1294 ± 0.0029	0.2060 ± 0.0006
$\mathrm{CAB}_{\mathrm{FULL}}$	$\mathbf{71.84} \pm 0.10$	$\mathbf{74.81} \pm 0.10$	74.97 ± 0.34	0.1316 ± 0.0025	$\mathbf{0.1884} \pm 0.0011$
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	$\underline{71.38} \pm 0.29$	$\underline{74.55} \pm 0.24$	$\mathbf{75.26} \pm 0.28$	$\underline{0.1292} \pm 0.0025$	$\underline{0.2002} \pm 0.0011$
$\mathrm{CAB}_{\mathrm{LATE}}$	70.53 ± 0.25	73.84 ± 0.24	75.01 ± 0.29	$\mathbf{0.1281} \pm 0.0025$	0.2131 ± 0.0006

Table 17: BERT students on SQuAD v2 validation: mean $\pm$ sample SD over $n = 5$ seeds

SQuAD v2 On SQuAD v2, CAB_{FULL} yields the strongest task performance, improving exact match to 71.84% and F1 to 74.81%, about 1.5 and 1.1 points above the best baseline (DIITO), well beyond the seed variation. This brings the student within 0.6 EM and 1.1 F1 points of the teacher (72.42% and 75.87%), compared with a gap of about 2.1–2.4 points for the baselines. CAB_{FULL} also obtains the lowest CAD (0.1884). The lowest CAD_{IO} is obtained by CAB_{LATE} (0.1281), slightly below DIITO (0.1294), a difference within one standard deviation. The different CAB alignments therefore expose a trade-off between maximizing extractive QA performance and minimizing intervention-response divergence.

D.2 DeBERTa

The same reporting convention is used for DeBERTa: CAD_{IO} is the primary map-independent robustness metric, while CAD is a secondary map-dependent diagnostic, with baseline values in gray.

Method	Acc. (%) ↑	Agree. (%) ↑	CAD_{IO} ↓	CAD ↓
Matched validation				
DeBERTa-v3-base	90.48	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	88.48 ± 0.12	92.08 ± 0.13	0.0727 ± 0.0046	0.3803 ± 0.0025
LWD	88.63 ± 0.13	92.52 ± 0.06	0.0661 ± 0.0038	$\mathbf{0.2906} \pm 0.0025$
TED	88.51 ± 0.16	92.21 ± 0.20	0.0693 ± 0.0041	0.3363 ± 0.0036
DIITO	88.70 ± 0.10	92.37 ± 0.12	0.0653 ± 0.0054	0.3537 ± 0.0069
CAB_{FULL}	$\underline{88.82} \pm 0.17$	$\underline{92.58} \pm 0.07$	$\mathbf{0.0586} \pm 0.0037$	$\underline{0.3153} \pm 0.0016$
$\text{CAB}_{\text{MIDDLE}}$	$\mathbf{88.91} \pm 0.23$	$\mathbf{92.61} \pm 0.15$	$\underline{0.0600} \pm 0.0025$	0.3266 ± 0.0027
CAB_{LATE}	88.59 ± 0.13	92.33 ± 0.22	0.0652 ± 0.0071	0.3326 ± 0.0049
Mismatched validation				
DeBERTa-v3-base	90.96	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	88.62 ± 0.17	92.43 ± 0.15	0.0753 ± 0.0072	0.3854 ± 0.0031
LWD	88.72 ± 0.10	92.65 ± 0.16	0.0722 ± 0.0065	$\mathbf{0.2961} \pm 0.0027$
TED	88.57 ± 0.20	92.53 ± 0.18	0.0739 ± 0.0085	0.3412 ± 0.0040
DIITO	$\underline{88.86} \pm 0.16$	92.75 ± 0.03	0.0701 ± 0.0049	0.3593 ± 0.0072
CAB_{FULL}	$\mathbf{89.02} \pm 0.20$	$\underline{92.88} \pm 0.17$	$\mathbf{0.0641} \pm 0.0016$	$\underline{0.3207} \pm 0.0023$
$\text{CAB}_{\text{MIDDLE}}$	$\mathbf{89.02} \pm 0.03$	$\mathbf{92.94} \pm 0.13$	0.0688 ± 0.0033	0.3320 ± 0.0033
CAB_{LATE}	88.83 ± 0.32	92.61 ± 0.46	$\underline{0.0665} \pm 0.0011$	0.3382 ± 0.0044

Table 18: DeBERTa students on MNLI validation: mean ± sample SD over $n = 3$ seeds.

MNLI On MNLI, CAB_{FULL} gives the lowest CAD_{IO} on both splits: 0.0586 on matched validation, a 10.1% reduction relative to the best baseline, DIITO (0.0653), and 0.0641 on mismatched validation, an 8.6% reduction relative to DIITO (0.0701). The CAB variants also obtain the highest accuracy and agreement on both splits (up to 88.91% matched and 89.02% mismatched), though these gains are within about one standard deviation.

Method	Accuracy (%) ↑	F1 (%) ↑	Agreement (%) ↑	$\text{CAD}_{\text{IO}}{}^{*}$ ↓	CAD ↓
DeBERTa-v3-base	92.40	89.79	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	91.58 ± 0.15	88.70 ± 0.19	94.30 ± 0.13	0.1057 ± 0.0002	0.4033 ± 0.0033
LWD	91.67 ± 0.17	88.83 ± 0.15	94.36 ± 0.22	0.1107 ± 0.0073	$\mathbf{0.3379} \pm 0.0070$
TED	91.67 ± 0.08	88.80 ± 0.08	94.35 ± 0.09	0.1123 ± 0.0028	0.3768 ± 0.0061
DIITO	91.72 ± 0.12	88.91 ± 0.03	94.56 ± 0.15	$\underline{0.0783} \pm 0.0020$	0.3757 ± 0.0050
CAB_{FULL}	$\mathbf{91.99} \pm 0.08$	$\mathbf{89.28} \pm 0.14$	$\mathbf{94.83} \pm 0.01$	$\mathbf{0.0729} \pm 0.0046$	$\underline{0.3498} \pm 0.0022$
$\text{CAB}_{\text{MIDDLE}}$	91.87 ± 0.13	89.18 ± 0.12	94.71 ± 0.16	0.0893 ± 0.0030	0.3605 ± 0.0031
CAB_{LATE}	$\underline{91.96} \pm 0.11$	$\underline{89.20} \pm 0.14$	$\underline{94.75} \pm 0.13$	0.0997 ± 0.0054	0.3711 ± 0.0020

Table 19: DeBERTa students on QQP validation: mean ± sample SD over $n = 3$ seeds.

QQP For DeBERTa on QQP, $\mathrm{CAB}_{\mathrm{FULL}}$ is strongest across the main measures: it reaches 91.99% accuracy, 89.28% F1, 94.83% agreement, and the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0729). Relative to DIITO, $\mathrm{CAD}_{\mathrm{IO}}$ is reduced by approximately 6.9%.

Method	Accuracy (%) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
DeBERTa-v3-base	94.16	100†	0†	0†
KD	92.78 ± 0.16	94.63 ± 0.21	0.0283 ± 0.0023	0.3813 ± 0.0008
LWD	92.86 ± 0.17	94.73 ± 0.26	0.0287 ± 0.0046	0.3371 ± 0.0014
TED	92.79 ± 0.27	94.74 ± 0.18	0.0278 ± 0.0032	0.3348 ± 0.0004
DIITO	92.84 ± 0.08	94.73 ± 0.08	0.0260 ± 0.0025	0.3784 ± 0.0019
$\mathrm{CAB}_{\mathrm{FULL}}$	**93.03** ± 0.10	94.74 ± 0.20	**0.0230** ± 0.0014	**0.2840** ± 0.0034
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	92.92 ± 0.09	**94.82** ± 0.10	0.0235 ± 0.0027	0.3016 ± 0.0006
$\mathrm{CAB}_{\mathrm{LATE}}$	92.91 ± 0.25	94.38 ± 0.10	0.0262 ± 0.0019	0.2881 ± 0.0041

Table 20: DeBERTa students on QNLI validation: mean ± sample SD over $n = 3$ seeds.

QNLI On QNLI, $\mathrm{CAB}_{\mathrm{FULL}}$ achieves the highest accuracy (93.03%) and the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.0230), an 11.5% reduction relative to the best baseline, DIITO (0.0260). It also attains the lowest CAD (0.2840), 15.2% below the best baseline value (TED, 0.3348). This makes QNLI the strongest DeBERTa result, with $\mathrm{CAB}_{\mathrm{FULL}}$ ranking first on both the input–output and the internal intervention diagnostics.

Method	Accuracy (%) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
DeBERTa-v3-base	96.22	100†	0†	0†
KD	**93.43** ± 0.96	95.53 ± 0.90	0.0385 ± 0.0022	0.3631 ± 0.0041
LWD	93.27 ± 0.54	95.68 ± 0.54	0.0376 ± 0.0023	0.3057 ± 0.0009
TED	93.23 ± 0.50	95.57 ± 0.46	0.0383 ± 0.0039	0.2518 ± 0.0027
DIITO	93.23 ± 0.11	**95.87** ± 0.34	0.0358 ± 0.0007	0.3380 ± 0.0023
$\mathrm{CAB}_{\mathrm{FULL}}$	92.97 ± 0.35	95.53 ± 0.40	0.0367 ± 0.0024	0.2100 ± 0.0036
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	93.16 ± 0.40	95.64 ± 0.41	**0.0351** ± 0.0027	0.2386 ± 0.0046
$\mathrm{CAB}_{\mathrm{LATE}}$	93.08 ± 0.18	95.64 ± 0.30	0.0383 ± 0.0037	**0.2079** ± 0.0048

Table 21: DeBERTa students on SST-2 validation: mean ± sample SD over $n = 3$ seeds.

SST-2 On SST-2, the differences in $\mathrm{CAD}_{\mathrm{IO}}$ are small. $\mathrm{CAB}_{\mathrm{MIDDLE}}$ gives the lowest value (0.0351), only slightly below DIITO (0.0358), whereas KD obtains the highest clean accuracy. This mirrors the BERT SST-2 result: intervention robustness improves only marginally and does not translate into a clean-accuracy advantage.

Method	MCC (×100) ↑	Agreement (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓
DeBERTa-v3-base	68.95	100†	0†	0†
KD‡	61.68 ± 0.92	91.75 ± 0.69	0.1550 ± 0.0028	0.2797 ± 0.0045
LWD	61.83 ± 1.24	91.88 ± 0.77	0.1649 ± 0.0042	**0.2228** ± 0.0018
TED	62.07 ± 0.78	91.85 ± 0.77	0.1607 ± 0.0015	0.2293 ± 0.0023
DIITO	60.58 ± 0.87	91.85 ± 0.50	0.1408 ± 0.0077	0.2642 ± 0.0012
$\mathrm{CAB}_{\mathrm{FULL}}$	61.54 ± 1.18	91.75 ± 0.60	**0.1279** ± 0.0032	0.2606 ± 0.0012
$\mathrm{CAB}_{\mathrm{MIDDLE}}$	61.65 ± 1.43	**92.43** ± 0.91	0.1325 ± 0.0053	0.2692 ± 0.0043
$\mathrm{CAB}_{\mathrm{LATE}}$	**62.58** ± 1.66	92.17 ± 0.28	0.1349 ± 0.0059	0.2723 ± 0.0040

Table 22: DeBERTa students on CoLA validation: mean ± sample SD over $n = 3$ seeds.

CoLA On CoLA, CAB_{FULL} reduces CAD_{IO} to 0.1279 from 0.1408 for DIITO, while CAB_{LATE} achieves the highest MCC (62.58). $\text{CAB}_{\text{MIDDLE}}$ obtains the highest teacher agreement (92.43%), again showing that the preferred alignment can depend on whether task performance or intervention robustness is prioritized.

Method	F1 (%) ↑	Agreement (%) ↑	CAD_{IO} ↓	CAD ↓
DeBERTa-v3-base	92.64	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	91.23 ± 0.18	92.65 ± 0.49	0.0334 ± 0.0008	0.3180 ± 0.0057
LWD	91.04 ± 0.38	93.38 ± 0.25	0.0349 ± 0.0037	$\mathbf{0.2503} \pm 0.0191$
TED	91.40 ± 0.25	92.73 ± 0.62	0.0328 ± 0.0018	0.2603 ± 0.0183
DIITO	91.74 ± 0.65	93.30 ± 1.63	0.0304 ± 0.0026	0.3124 ± 0.0083
CAB_{FULL}	$\mathbf{92.03} \pm 0.08$	93.46 ± 0.75	$\mathbf{0.0294} \pm 0.0009$	0.3109 ± 0.0097
$\text{CAB}_{\text{MIDDLE}}$	91.77 ± 0.35	$\mathbf{94.04} \pm 0.62$	0.0305 ± 0.0015	0.3083 ± 0.0072
CAB_{LATE}	91.36 ± 0.65	92.89 ± 0.65	0.0326 ± 0.0028	0.3158 ± 0.0091

Table 23: DeBERTa students on MRPC validation: mean ± sample SD over $n = 3$ seeds.

MRPC On MRPC, CAB_{FULL} achieves the best F1 (92.03%) and the lowest CAD_{IO} (0.0294), while $\text{CAB}_{\text{MIDDLE}}$ gives the highest teacher agreement (94.04%). The robustness improvement over DIITO is modest but is accompanied by improved task performance.

Method	Accuracy (%) ↑	Agreement (%) ↑	CAD_{IO} ↓	CAD ↓
DeBERTa-v3-base	83.03	$100^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	73.29 ± 1.65	81.83 ± 1.63	0.0776 ± 0.0033	0.3825 ± 0.0095
LWD	75.09 ± 1.25	83.87 ± 0.83	0.0780 ± 0.0066	$\mathbf{0.3673} \pm 0.0170$
TED	75.33 ± 0.75	84.36 ± 0.21	0.0786 ± 0.0091	0.3699 ± 0.0126
DIITO	75.09 ± 1.08	84.84 ± 1.08	0.0707 ± 0.0011	0.3791 ± 0.0091
CAB_{FULL}	75.33 ± 1.71	83.87 ± 1.16	0.0706 ± 0.0071	0.3759 ± 0.0070
$\text{CAB}_{\text{MIDDLE}}$	74.61 ± 2.73	84.36 ± 1.50	$\mathbf{0.0650} \pm 0.0030$	0.3754 ± 0.0061
CAB_{LATE}	$\mathbf{75.45} \pm 2.25$	$\mathbf{85.44} \pm 2.66$	0.0707 ± 0.0159	0.3792 ± 0.0100

Table 24: DeBERTa students on RTE validation: mean ± sample SD over $n = 3$ seeds.

RTE For RTE, $\text{CAB}_{\text{MIDDLE}}$ obtains the lowest CAD_{IO} (0.0650), compared with 0.0707 for DIITO. CAB_{LATE}, however, gives the highest accuracy (75.45%) and agreement (85.44%), showing that the alignment giving the strongest robustness is not necessarily the one giving the strongest clean-task result.

Method	Spearman ($\times 100$) ↑	Teacher MSE ↓	CAD_{IO} ↓	CAD ↓
DeBERTa-v3-base	89.56	$0^{\dagger}$	$0^{\dagger}$	$0^{\dagger}$
KD	89.43 ± 0.06	0.1847 ± 0.0078	0.00635 ± 0.00059	0.2907 ± 0.0087
LWD	89.39 ± 0.03	0.1946 ± 0.0023	0.00667 ± 0.00044	0.2607 ± 0.0040
TED	89.16 ± 0.03	0.2153 ± 0.0077	0.00690 ± 0.00049	$\mathbf{0.2263} \pm 0.0032$
DIITO	89.23 ± 0.11	0.1597 ± 0.0209	0.00620 ± 0.00024	0.2744 ± 0.0083
CAB_{FULL}	89.34 ± 0.04	0.1006 ± 0.0021	0.00571 ± 0.00023	0.2604 ± 0.0076
$\text{CAB}_{\text{MIDDLE}}$	89.40 ± 0.07	0.1025 ± 0.0021	$\mathbf{0.00536} \pm 0.00029$	0.2712 ± 0.0093
CAB_{LATE}	$\mathbf{89.45} \pm 0.02$	$\mathbf{0.1004} \pm 0.0031$	0.00544 ± 0.00020	0.2648 ± 0.0069

Table 25: DeBERTa students on STS-B validation: mean ± sample SD over $n = 3$ seeds.

STS-B On STS-B, $\mathrm{CAB}_{\mathrm{MIDDLE}}$ achieves the lowest $\mathrm{CAD}_{\mathrm{IO}}$ (0.00536), a reduction of approximately 13.5% relative to DIITO. $\mathrm{CAB}_{\mathrm{LATE}}$ obtains the highest Spearman correlation (89.45) and lowest teacher MSE (0.1004), while $\mathrm{CAB}_{\mathrm{FULL}}$ is close on both measures.

E LAYERWISE ABSTRACTION MAP

We report held-out reconstruction MSE for each layerwise abstraction map on SST-2 under $\mathrm{CAB}_{\mathrm{FULL}}$ (Figure 3). Error is lowest at S1, peaks at S4–S5, and decreases at S6. These values quantify reconstruction error in student representation space.

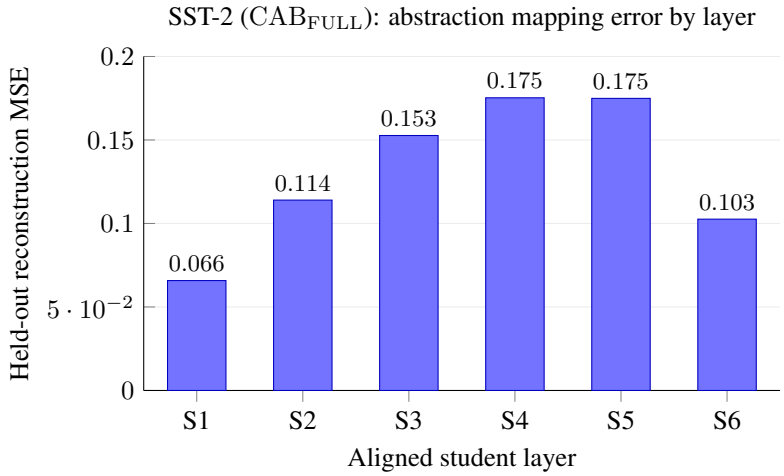

Figure 3: Layer-wise reconstruction error on SST-2 under $\mathrm{CAB}_{\mathrm{FULL}}$.

F DEPTH COMPRESSION

We evaluate depth compression on RTE by distilling a fixed 12-layer BERT teacher into 9-, 6-, and 3-layer students with the same hidden dimension. We compare $\mathrm{CAB}_{\mathrm{FULL}}$ with DIITO using the same FULL alignment and training budget.

As shown in Figure 4, the 6- and 9-layer CAB students achieve the same mean accuracy (62.45%), while $\mathrm{CAD}_{\mathrm{IO}}$ increases from 0.00407 at nine layers to 0.00989 at six layers. At six layers, CAB exceeds DIITO's mean accuracy by 3.24 percentage points (62.45% vs. 59.21%), although its $\mathrm{CAD}_{\mathrm{IO}}$ is slightly higher. At nine layers, accuracy is similar (62.45% vs. 62.21%), while CAB achieves lower $\mathrm{CAD}_{\mathrm{IO}}$ (0.00407 vs. 0.00591). Compression to three layers reduces accuracy and increases $\mathrm{CAD}_{\mathrm{IO}}$ for both methods. These results indicate a practical limit to depth compression under the evaluated training setup: preserving task accuracy does not ensure preserved interventional fidelity, and more aggressive compression compromises both.

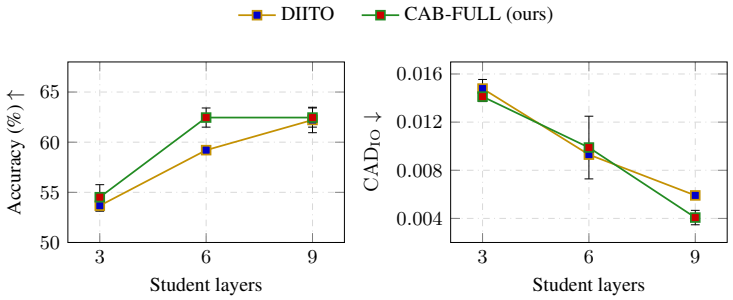

Figure 4: RTE performance under depth compression using a 12-layer BERT teacher and 3-, 6-, and 9-layer BERT students. Points show means and error bars indicate sample standard deviations across student seeds 42–44.

G GENERALIZATION

Table 26 evaluates whether the learned behavior transfers beyond the training distribution. When trained on QQP, CAB_{LATE} achieves the strongest transfer to MRPC while also improving in-domain QQP performance, suggesting that the learned abstraction can improve generalization rather than only fit the training task. For MNLI training, the differences are smaller and more mixed, although $\text{CAB}_{\text{MIDDLE}}$ obtains the strongest QNLI transfer result.

	Trained on QQP				**Trained on MNLI**			
	In-domain		Transfer		In-domain		Transfer	
Method	QQP Acc	QQP F1	MRPC Acc	MRPC F1	MNLI-m	MNLI-mm	SNLI	QNLI
KD	90.95 ± 0.06	87.74 ± 0.09	66.01 ± 1.59	71.98 ± 1.81	82.36 ± 0.34	82.87 ± 0.18	78.06 ± 0.11	56.93 ± 1.43
LWD	91.02 ± 0.06	87.86 ± 0.05	67.16 ± 0.65	73.82 ± 0.73	82.68 ± 0.22	83.26 ± 0.23	78.45 ± 0.07	56.73 ± 1.10
TED	91.02 ± 0.02	87.83 ± 0.05	66.91 ± 2.18	73.46 ± 2.51	83.05 ± 0.14	**83.53** ± 0.08	**78.81** ± 0.25	56.37 ± 1.16
DIITO	91.08 ± 0.05	87.92 ± 0.05	64.54 ± 2.84	69.86 ± 3.05	82.77 ± 0.26	83.12 ± 0.12	78.42 ± 0.14	55.82 ± 0.69
CAB_{FULL}	91.08 ± 0.07	87.78 ± 0.13	65.77 ± 0.14	71.90 ± 0.29	83.05 ± 0.21	83.44 ± 0.16	78.60 ± 0.21	55.48 ± 0.37
$\text{CAB}_{\text{MIDDLE}}$	91.12 ± 0.02	88.03 ± 0.03	66.67 ± 0.42	73.01 ± 0.27	**83.20** ± 0.20	83.47 ± 0.05	78.77 ± 0.14	**57.61** ± 0.75
CAB_{LATE}	**91.31** ± 0.04	**88.28** ± 0.06	**68.38** ± 1.61	**74.60** ± 1.71	82.93 ± 0.16	83.34 ± 0.12	78.30 ± 0.20	57.44 ± 1.28

Table 26: In-domain and transfer performance for 6-layer BERT students (mean $\pm$ SD, $n = 3$; %). Models trained on QQP are evaluated on MRPC, while models trained on MNLI are evaluated on SNLI and QNLI.

H ABLATION OF THE CAB OBJECTIVE

We investigate the contribution of hidden-boundary consistency to CAB by separating the hidden and output components of the CAD loss:

$$\mathcal{L}_{\text{CAD}} = \mathcal{L}_{\text{CAD}}^{\text{hid}} + \mathcal{L}_{\text{CAD}}^{\text{out}}. \tag{10}$$

Here, $\mathcal{L}_{\text{CAD}}^{\text{hid}}$ contains the normalized mean squared error at hidden boundaries, while $\mathcal{L}_{\text{CAD}}^{\text{out}}$ contains the base-two Jensen–Shannon divergence at the output. We keep the same weighting as in the full objective when either component is removed; in particular, the output term is not renormalized.

We compare three training objectives:

$$\mathcal{L}_{\text{bridge}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{KD}}\mathcal{L}_{\text{KD}} + \lambda_{\text{B}}\mathcal{L}_{\text{B}}, \tag{11}$$

$$\mathcal{L}_{\text{output}} = \mathcal{L}_{\text{bridge}} + \lambda_{\text{C}}\mathcal{L}_{\text{CAD}}^{\text{out}}, \tag{12}$$

$$\mathcal{L}_{\text{full}} = \mathcal{L}_{\text{bridge}} + \lambda_{\text{C}}\left(\mathcal{L}_{\text{CAD}}^{\text{hid}} + \mathcal{L}_{\text{CAD}}^{\text{out}}\right). \tag{13}$$

The bridge-only condition uses the bridge loss $\mathcal{L}_{\text{B}}$. The output-only condition additionally includes the output CAD term, while full CAB also includes the hidden CAD term. Comparing output-only with full CAB therefore isolates the contribution of hidden-boundary consistency.

Setup. We distill a 12-layer BERT teacher into a six-layer student on SST-2 and MRPC using FULL alignment and three seeds ($n = 3$). Within each seed, all conditions use the same teacher, student initialization, data splits, training budget, and shared loss coefficients. Evaluation maps are independently refitted for each trained student using the same procedure as in the main experiments. All conditions are evaluated using the same intervention diagnostics. The output-only condition was added after inspecting the initial bridge-only and full-CAB results, so we treat this analysis as exploratory.

Results. Adding the hidden CAD term reduces mean hidden-response error relative to the output-only condition by 8.26% on SST-2 and 1.47% on MRPC, with lower error for full CAB in every paired seed. This indicates that the hidden CAD term contributes to the improved internal intervention-response consistency of full CAB.

These improvements occur with nearly unchanged task performance. All three conditions have the same mean SST-2 accuracy, while mean MRPC F1 differs by less than 0.05 percentage points

Method	Task score (%) ↑	$\mathrm{CAD}_{\mathrm{IO}}$ ↓	CAD ↓	Hidden error ↓	Output JSD ↓
SST-2					
Bridge only	90.37 ± 0.23	0.0267 ± 0.0017	0.3333 ± 0.0047	0.3932 ± 0.0056	0.0335 ± 0.0011
CAB output-only	90.37 ± 0.23	0.0267 ± 0.0017	0.3333 ± 0.0047	0.3932 ± 0.0056	0.0335 ± 0.0011
CAB full	90.37 ± 0.34	0.0268 ± 0.0018	0.3062 ± 0.0048	0.3608 ± 0.0059	0.0333 ± 0.0010
MRPC					
Bridge only	87.30 ± 0.38	0.0535 ± 0.0111	0.3291 ± 0.0212	0.3861 ± 0.0239	0.0439 ± 0.0078
CAB output-only	87.30 ± 0.52	0.0534 ± 0.0102	0.3289 ± 0.0212	0.3859 ± 0.0239	0.0435 ± 0.0077
CAB full	87.26 ± 0.33	0.0537 ± 0.0114	0.3241 ± 0.0204	0.3803 ± 0.0229	0.0434 ± 0.0078

Table 27: CAB objective ablation with FULL alignment. Values are validation means ± sample standard deviations over three seeds ($n = 3$).

between output-only and full CAB. Differences at the output are also small, and full CAB does not reduce mean $\mathrm{CAD}_{\mathrm{IO}}$ on either task.

Overall, the ablation shows that the hidden CAD term improves internal intervention-response consistency under the learned abstraction maps, particularly on SST-2, without corresponding improvements in task performance or $\mathrm{CAD}_{\mathrm{IO}}$. Together with the layerwise diagnostic in Figure 2, this suggests that internal and output intervention consistency capture related but distinct properties.

I QWEN DISTILLATION

I.1 YELP PROMPT AND RATING READOUT

We use the Qwen chat template with the following user message and an assistant prefill. The placeholder `<review text>` is replaced by the review being evaluated.

> **User:**
> Predict the rating of this Yelp review from 1 (worst) to 5 (best). Give only the number.
> `Review: <review text>`
>
> **Assistant prefill:**
> `Rating:`

Prompts are limited to 512 tokens, including chat-template tokens. Reviews exceeding the available budget are truncated by retaining their initial tokens. We score the next-token logits for the five single-token ratings $\{1, 2, 3, 4, 5\}$ and normalize over these tokens:

$$p_M(r \mid x) = \frac{\exp z_M(r; x)}{\sum_{k=1}^{5} \exp z_M(k; x)}.$$

Accuracy and teacher agreement use the highest-probability rating. For rating RMSE, we use the expected rating $\hat{r}_M(x) = \sum_{r=1}^{5} r\, p_M(r \mid x)$.

I.2 QWEN DISTILLATION SETUP

Layer alignment and abstraction maps. We use FULL proportional alignment between the teacher's 48 transformer layers and the student's 36 layers, grouping consecutive teacher layers into blocks associated with student layers. Hidden states are taken at transformer block outputs before the model's final normalization.

Optimization. We update all student parameters for 250 optimizer steps, using a microbatch size of one and gradient accumulation over eight examples. AdamW uses a peak learning rate of 2×10^{-6}, weight decay 0.01, and a 20-update linear warmup. Student parameters and saved checkpoints use FP32; the frozen teacher uses BF16. Both methods are run with training seeds 42, 43, and 44.