From Images to Words: Efficient Cross-Modal Knowledge Distillation to Language Models from Black-box Teachers

Anonymous authors
Paper under double-blind review

Abstract

Knowledge distillation has become the dominant mechanism for transferring capabilities from large models into smaller ones. Yet, despite its widespread success, the transfer remains confined almost entirely within modality boundaries: language models learn from language-only models, vision models learn from vision-only models, and little is known about how knowledge can be distilled across modalities. This limitation is increasingly consequential as modern vision-language models encode rich semantic abstractions about the world that are difficult to acquire from texts alone. Existing cross-modal distillation approaches offer a potential path forward, but typically rely on expensive modality-specific pre-training or multimodal architectures at inference time, limiting their practicality. We introduce `ARMADA`, a cross-modal distillation framework that transfers knowledge from frozen white-box or black-box multimodal teachers into language-only students. For each text input, `ARMADA` obtains a frozen teacher representation (e.g., a latent from a text-to-image diffusion model) and employs a lightweight alignment module, `TS Aligner`, to project teacher and student representations into a shared manifold. The student is trained through a combination of output matching, manifold alignment, and auxiliary supervision objectives, enabling it to internalize cross-modal structure while remaining predictive of downstream tasks. Across twelve natural language understanding tasks, eight reasoning benchmarks, and five instruction-tuning datasets spanning BERT, DeBERTa-v2-1.4B, OPT-1.3B, and LLaMA-3B,7B,8B students, `ARMADA` consistently improves performance, yielding gains of up to 3.4% on NLU tasks and 2.6% on reasoning benchmarks while using only 0.8% of the trainable parameters required by competing distillation methods. Through controlled comparisons against five degraded-teacher baselines and a capacity-matched control, we show that these gains cannot be explained by additional parameters or generic regularization alone: `ARMADA` separates significantly ($p < 0.1$) from the undistilled baseline and from every fully-degraded teacher, with the clearest prediction-level evidence on the grounded-inference task RTE; separation from the strictest control (a semantically-shuffled teacher that retains full task supervision) is directional but does not reach aggregate significance. Taken together, our results suggest that multimodal foundation models contain transferable semantic structure that can be imported into language-only systems without multimodal pre-training or inference-time multimodal components.

1 Introduction

Pre-trained large language models (LLMs) (Achiam et al., 2023; Gunasekar et al., 2023; Jiang et al., 2023a; Dubey et al., 2024; DeepSeek-AI et al., 2024) have achieved remarkable performance across reasoning, natural language understanding, code generation, and dialogue tasks. However, their growing scale has made deployment increasingly expensive, motivating extensive research on *model compression* techniques (Hinton et al., 2015; Zhou et al., 2021). Knowledge distillation (KD) (Hinton et al., 2015) has emerged as one of the most successful compression paradigms, but most existing methods (Sun et al., 2019; Zhang et al., 2018; Zhou et al., 2021) remain *unimodal*, requiring both teacher and student models to operate within the same modality. Consequently, a text teacher can only transfer knowledge already expressible in text and cannot readily convey

grounded perceptual structure — spatial relations, object affordances, and physical or social commonsense — that humans acquire through vision and that text corpora often capture only weakly. Modern vision-language models encode precisely this type of information, motivating its transfer into language-only models.

Cross-modal knowledge distillation addresses this limitation by allowing a teacher specialized in one modality to transfer knowledge to a student operating in another. Despite growing interest in multimodal foundation models, recent surveys (Song & Zheng, 2026) note that cross-modal distillation remains poorly understood, with limited empirical understanding of how semantic structure transfers across modalities. Existing approaches such as Vokenization (Tan & Bansal, 2020) and VidLanKD (Tang et al., 2021) typically depend on large-scale modality-specific pre-training (e.g., 136M video clips in VidLanKD) and often retain multimodal components at inference time. Yet modern text-to-vision models — both white-box (Rombach et al., 2022b) and black-box (Betker et al., 2023) — already encode rich cross-modal abstractions and are readily available off-the-shelf. This raises a fundamental question: *can a frozen multimodal teacher improve a strictly language-only student without additional teacher pre-training or multimodal inference?* A positive answer would make cross-modal knowledge transfer practical, scalable, and architecture-agnostic.

To this end, we propose `ARMADA`[‡], a cross-modal knowledge distillation framework that transfers knowledge from white-box or black-box text-to-vision teachers into language-only students. Given a text input, `ARMADA` queries a frozen teacher to obtain a cross-modal representation and projects it into a low-dimensional manifold shared with the student through a lightweight aligner, `TS Aligner`. The student and aligner are trained jointly through coupled output-, manifold-, and auxiliary-alignment objectives that encourage the student's internal representations to remain consistent with the teacher's cross-modal structure while preserving task performance. Because the teacher is frozen and accessed only through its representations, `ARMADA` introduces only 0.8% additional trainable parameters and remains compatible with heterogeneous student architectures.

We evaluate `ARMADA` on twelve NLU tasks from GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019), five commonsense reasoning tasks, and three mathematical reasoning benchmarks. Across BERT, DeBERTa, OPT, and LLaMA models, `ARMADA` consistently improves performance, yielding gains of up to 3.4% on NLU tasks and 2.6% on reasoning benchmarks. These improvements persist across model scales, including models with more than 1B parameters, while requiring only 0.8% of the trainable parameters used by competing distillation approaches.

Beyond benchmark performance, we investigate whether the gains arise from meaningful cross-modal transfer or merely from additional regularization. We conduct a controlled mechanism study (Section 6) using frozen, random, shuffled, and constant-teacher controls, together with a capacity-matched variant that removes the alignment mechanism while preserving parameter count. Across seven tasks and three random seeds, `ARMADA` consistently outperforms the undistilled baseline, the capacity-matched control, and every fully degraded teacher. Separation from the strictest control — a shuffled teacher retaining full task supervision — is directional but does not reach aggregate significance; at the prediction level, the clearest per-task evidence is on the grounded-inference task RTE.

The contributions of our paper are summarized below:

- **Cross-Modal Knowledge Distillation for Large Language Models.** Unlike multimodal pre-trained models such as CLIP (Radford et al., 2021) or Qwen-VL (Wang et al., 2024), cross-modal distillation transfers knowledge into models that operate on a single modality, making effective knowledge fusion substantially more challenging. To our knowledge, `ARMADA` is the first architecture-agnostic framework that distils knowledge from frozen black-box multimodal teachers into language-only student models.
- **Efficient and Scalable Cross-Modal Distillation.** Existing cross-modal KD methods typically rely on expensive modality-specific pre-training or tightly coupled multimodal architectures. In contrast, `ARMADA` operates with frozen teachers, requires no additional multimodal training, and introduces only 0.8% of the trainable parameters used by competing approaches, making cross-modal transfer practical, scalable, and compatible with both white-box and black-box foundation models.

[‡]The source code of `ARMADA` will be released upon acceptance of the paper.

- **Mechanistic Evidence of Cross-Modal Transfer.** Beyond benchmark improvements, we perform a controlled mechanism study using degraded-teacher controls, capacity-matched variants, paired statistical tests, and prediction-level analyses. The results indicate that the gains are not explained by parameter expansion or degraded auxiliary supervision alone; separation from the strictest shuffled-teacher control is directional but not statistically conclusive, with the clearest per-task evidence on a single grounded-inference benchmark.

2 Related Work

Unimodal Knowledge Distillation. Knowledge distillation (KD) was popularized by Hinton et al. (2015), building on earlier model-compression work by Buciluǎ et al. (2006). Beyond output matching, subsequent studies explored transferring intermediate representations (Romero et al., 2014; Zagoruyko & Komodakis, 2016; Sun et al., 2019; Zhang et al., 2020), relational information (Park et al., 2019; Chen et al., 2020), and meta-learned supervision (Zhou et al., 2021; Sengupta et al., 2024). However, nearly all of these approaches assume that teacher and student operate within the same modality.

Cross-modal Knowledge Distillation. Unlike unimodal KD, cross-modal distillation transfers knowledge across modalities. Early work by Gupta et al. (2016) transferred representations from RGB to depth models, while later studies distilled information from multimodal teachers into language models (Tang et al., 2021; Jin et al., 2022) or from language models into vision systems (Ghosh et al., 2023). Related multimodal architectures, including VL-BERT (Su et al., 2019), VisualBERT (Li et al., 2019), ViLBERT (Lu et al., 2019), and X-adapter (Zhang et al., 2023), rely on multimodal pre-training and jointly learned visual-language representations. However, most existing methods require large-scale modality-specific pre-training and often retain multimodal components during inference.

From Off-Policy to On-Policy LLM Distillation. Recent LLM distillation research has shifted from static output matching toward *on-policy* training, where supervision adapts to the student's evolving behavior. Sequence-level distillation (Kim & Rush, 2016) introduced trajectory-level supervision but remained susceptible to exposure bias (Ross et al., 2011). More recent approaches such as Generalized KD (Agarwal et al., 2024), MiniLLM (Gu et al., 2024), DistiLLM (Ko et al., 2024), and DistiLLM-2 (Ko et al., 2025) optimize student-generated trajectories using alternative divergence objectives. A parallel line of work studies adaptive divergence design, including ToDi (Jung et al., 2025), AKL (Wu et al., 2025), and entropy-aware distillation (Jin et al., 2026), while chain-of-thought distillation and process-level supervision extend these ideas to reasoning models (Guo et al., 2025; Li et al., 2026; Lightman et al., 2024). These developments motivate our jointly trained alignment framework, although they largely remain confined to text-only settings.

Representation-space, and Black-box Distillation. Several recent methods perform distillation through a shared representation space. Dual-Space KD (Zhang et al., 2025) aligns teacher and student hidden states through learned projectors, Delta-KD (Cao et al., 2025) transfers fine-tuning-induced representation shifts, and ORPO-Distill (Singh et al., 2025) enables cross-architecture preference transfer. Our `TS Aligner` plays a similar role but aligns representations across modalities rather than across text models. The setting is also related to black-box distillation methods such as Lion (Jiang et al., 2023b) and Generative Adversarial Distillation (Ye et al., 2025), which learn from proprietary API-based teachers. Unlike these approaches, `ARMADA` never accesses teacher logits and relies solely on frozen cross-modal embeddings. More importantly, existing representation-space, black-box, and on-policy distillation methods operate almost exclusively within a single modality, leaving cross-modal distillation largely unexplored.

How is Our Method Different? `ARMADA` differs from prior cross-modal KD along four dimensions. *(i) Black-box teachers without additional pre-training.* Methods such as Vokenization (Tan & Bansal, 2020), VidLanKD (Tang et al., 2021), and X-adapter (Zhang et al., 2023) require white-box access and extensive modality-specific pre-training, whereas `ARMADA` uses frozen embeddings from off-the-shelf white-box or black-box text-to-vision models. *(ii) Text-only inference.* Unlike VL-BERT (Su et al., 2019), VisualBERT (Li et al., 2019), and ViLBERT (Lu et al., 2019), the distilled student requires no multimodal components at deployment. *(iii) Jointly trained alignment.* The `TS Aligner` is optimized alongside the student through coupled output, manifold, and auxiliary objectives. *(iv) Parameter efficiency.* The aligner introduces only $\sim 0.8\%$ of the trainable parameters required by competing multimodal KD approaches.

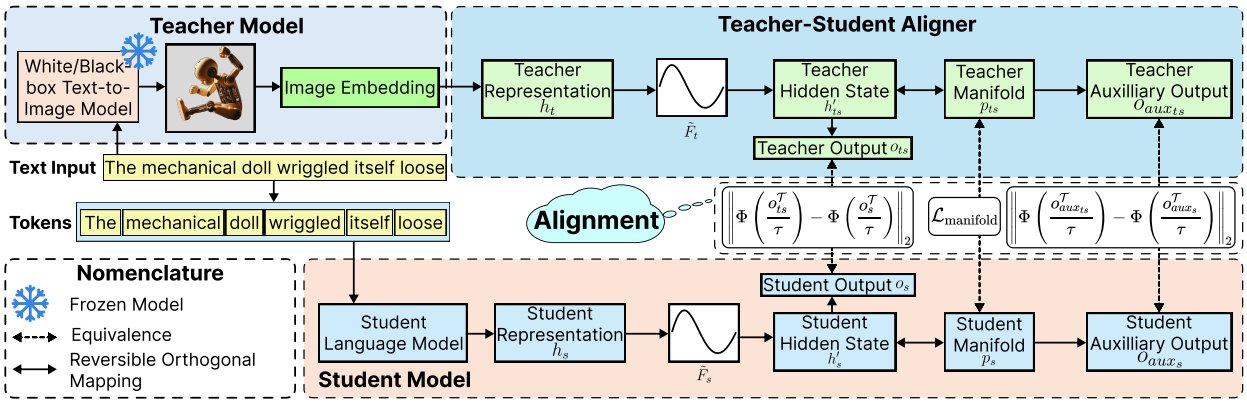


Figure 1: Overview of ARMADA, illustrated on an NLU example. A text-to-image teacher produces a frozen cross-modal embedding for the input; the TS Aligner aligner and the language-only student are trained jointly under coupled output, manifold-alignment, and auxiliary-output losses.

3 Proposed Methodology

In this section, we present ARMADA – an **A**lignment-induced c**r**oss-**m**od**a**l knowledge **d**istill**a**tion. We assume a multimodal model F_t of modality set $\mathcal{M}_t$ as our teacher model. The student model F_s, the model that is being distilled, is assumed to belong to a different modality set $\mathcal{M}_s$, with $\mathcal{M}_s \subset \mathcal{M}_t$. We introduce TS Aligner, a cross-modal alignment module for aligning teacher and student models to a common modality. TS Aligner, denoted by F_{ts}, consists of a non-linear mapping $\tilde{F}_{ts}$, an output layer O_{ts}, a manifold projection layer P_{ts} and an auxiliary output layer $O_{aux_{ts}}$. For a student input sequence $\{x_1, x_2, \cdots, x_n\} = X_s \in \mathcal{M}_s$, we assume a cross-modal representation $X_t \in \mathcal{M}_t$. For instance, if the student input is a sequence of text tokens, we can obtain the teacher sequence X_t from a text-to-image teacher model. The cross-modal distillation process consists of three major steps – (i) output alignment, (ii) manifold alignment, and (iii) auxiliary output alignment. Figure 1 illustrates the overall framework of ARMADA. We enlist all the notations used in the paper and their descriptions in Table 7 (Appendix A).

3.1 Output Alignment

For a given student input sequence X_s and its equivalent teacher input sequence X_t, we first extract hidden representation using the student model F_s and the frozen teacher model F_t, in sequence and obtain $h_s^X = F_s(X_s)$ and $h_t^X = F_t(X_t)$, respectively. We utilize a non-linear mapping, defined by the operators, $\tilde{F}_s$ and $\tilde{F}_{ts}$, for encoding the hidden representations, obtaining $h'^{X}_{s} = \tilde{F}_s(h_s^X)$ and $h'^{X}_{ts} = \tilde{F}_{ts}(h_t^X)$, respectively. For each task $\mathcal{T} \in \mathbb{T}$, we use task-specific linear layers for both TS Aligner and the student to obtain the outputs, $o_{ts}^{\mathcal{T}}$ and $o_s^{\mathcal{T}}$, respectively. We use the task-specific loss function $\mathcal{L}_{\mathcal{T}}$ (*e.g.,* cross-entropy for classification and mean squared error for regression) to compute loss against the ground-truth $Y^{\mathcal{T}}$ on the training dataset. For TS Aligner, the output loss is given as,

$$\mathcal{L}_{ts}^{(1)} = \mathcal{L}_{\mathcal{T}}(Y^{\mathcal{T}}, o_{ts}^{\mathcal{T}}). \tag{1}$$

For the student model, the output loss is computed as the weighted sum of the loss against the ground-truth and the logit matching loss (Hinton et al., 2015) with the aligner as,

$$\mathcal{L}_{s}^{(1)} = (1-\alpha) \cdot \mathcal{L}_{\mathcal{T}}(Y^{\mathcal{T}}, o_{s}^{\mathcal{T}}) + \alpha \left\| \Phi\left(\frac{o_{ts}^{\mathcal{T}}}{\tau}\right) - \Phi\left(\frac{o_{s}^{\mathcal{T}}}{\tau}\right) \right\|_2. \tag{2}$$

For the classification task, Φ is the softmax function, whereas it is an identity function for the regression task. α and τ are the hyperparameters representing the output loss and temperature factors, respectively. We use $\alpha = 0.5$ as default.

3.2 Manifold Alignment

Sun et al. (2019) suggested an additional loss between the teacher and student hidden representations to enforce the student model in learning the feature representations similar to the teacher. However, minimizing the point-wise distance between the teacher and student representations can distort the information learned in the student's original modality in cross-modal distillation. Therefore, we first project the representations obtained from `TS Aligner` and the student models to a shared manifold and subsequently minimize their distance on the common subspace. We use two orthogonal projection mapping, P_{ts} and P_s, to project the original teacher and student hidden representations to a common subspace to obtain $P_{ts}(h'^{X}_{ts}) = p^X_{ts}$ and $P_s(h'^{X}_{s}) = p^X_s$. Note that p^X_{ts} and p^X_s belong to the same dimensional subspace, therefore the manifold equivalence can be enforced with traditional similarity or distance measures. Following Huo et al. (2021), we propose an inner-product-based distance measure,

$$\mathcal{L}_{cosine}(p^X_{ts}, p^X_s) = 1 - \langle \frac{1}{n}\sum_{i=1}^{n} p^{x_i}_{ts}, \frac{1}{n}\sum_{i=1}^{n} p^{x_i}_{s} \rangle \tag{3}$$

where, $\langle ., . \rangle$ denotes the inner product between two vectors, and $\mathcal{L}_{cosine}$ measures the semantic similarity between teacher and student manifolds. Similarly, we formulate two other losses, $\mathcal{L}_{euclid}$ and $\mathcal{L}_{elementwise}$, to measure the distance between two manifolds in Euclidean space.

$$\mathcal{L}_{euclid}(p^X_{ts}, p^X_s) = \left\| \frac{1}{n}\sum_{i=1}^{n} p^{x_i}_{ts} - \frac{1}{n}\sum_{i=1}^{n} p^{x_i}_{s} \right\|_2 \tag{4}$$

$$\mathcal{L}_{elementwise}(p^X_{ts}, p^X_s) = \frac{1}{n}\sum_{i=1}^{n} ||p^{x_i}_{ts} - p^{x_i}_{s}||_2 \tag{5}$$

Using $\mathcal{L}_{euclid}$, we compute the Euclidean distance between the *centroids* of the teacher and student projection vectors, whereas with $\mathcal{L}_{elementwise}$ we compute the expected *per-sample* distance between paired teacher and student projection vectors. The three objectives form an increasingly strict regularization ladder: $\mathcal{L}_{cosine}$ constrains only the *direction* of the mean projections, $\mathcal{L}_{euclid}$ additionally constrains their *magnitude* (centroid distance), and $\mathcal{L}_{elementwise}$ constrains *every* sample's projection individually. By the triangle inequality, $\mathcal{L}_{elementwise} \geq \mathcal{L}_{euclid}$ for any batch, so $\mathcal{L}_{elementwise}$ imposes the tightest teacher-student coupling and $\mathcal{L}_{cosine}$ the loosest. These manifold-alignment losses parallel the dominant divergence measures (Jung et al., 2025; Wu et al., 2025; Jin et al., 2026) and form a regularization ladder of increasingly strict teacher–student coupling that trades off differently across tasks; and, as in on-policy distillation, our `TS Aligner` aligner is trained *jointly* with the student, so that the cross-modal supervision tracks the student as it learns rather than being fixed in advance.

For the rest of the paper, we generalize these three losses as $\mathcal{L}_{manifold}$, and — unless stated otherwise — a model labelled simply "`ARMADA`" uses the loss that performs best for that regime (see below). We adopt $\mathcal{L}_{euclid}$ as the default for NLU tasks (most robust on average) and $\mathcal{L}_{cosine}$ for instruction-tuning, and we always state the loss explicitly in every table and figure to avoid ambiguity.

3.3 Auxiliary Output Alignment

To further regularize the student representations, we incorporate an auxiliary output head on `TS Aligner` and student projection heads. We align the models to learn appropriate manifold projections that maximize their performance on downstream tasks by incorporating an auxiliary output head on both teacher and student projection vectors. Given the auxiliary output $o^{\mathcal{T}}_{aux_{ts}} = O^{\mathcal{T}}_{aux_{ts}}(p^X_{ts})$ and $o^{\mathcal{T}}_{aux_s} = O^{\mathcal{T}}_{aux_s}(p^X_s)$, we calculate the auxiliary losses, $\mathcal{L}^{(2)}_{ts}$ and $\mathcal{L}^{(2)}_{s}$, similar to Equations 1 and 2. The intuition is that aligning the projected teacher manifold to the student manifold under a task-relevant supervisory signal forces the student to internalize structure that is simultaneously consistent with the teacher's cross-modal abstraction and predictive of the downstream label, rather than collapsing to either signal in isolation.

As opposed to vokenization (Tan & Bansal, 2020) and VidLanKD (Tang et al., 2021), which use pre-training objectives to train the teacher models, we optimize `TS Aligner` on task-specific objectives, allowing task-specific knowledge diffusion to students. The final `TS Aligner` loss is calculated as $\mathcal{L}_t = \mathcal{L}_{ts}^{(1)} + \gamma\mathcal{L}_t^{(2)}$. For the student model, we use the output loss, auxiliary output loss, and manifold alignment loss to calculate the final training loss as $\mathcal{L}_s = \mathcal{L}_s^{(1)} + \gamma\mathcal{L}_s^{(2)} + \beta\mathcal{L}_{manifold}(p_t^X, p_s^X)$. β (default to 1) and γ (default to 1) are hyperparameters used to assign weightage to the manifold and auxiliary output alignment loss, respectively.

4 Experimental Setup

Tasks. We apply `ARMADA` on seven student LMs and four multimodal teacher models and compare it against ten unimodal and three multimodal KD methods across 12 NLU tasks and 5 instruction-following tasks. Following Liu et al. (2022); Zhou et al. (2021), we consider four different NLU tasks from SuperGLUE (Wang et al., 2019) – CB, COPA, BoolQ and WiC, and eight tasks from GLUE (Wang et al., 2018) – CoLA, MRPC, RTE, STS-B (categorized as *GLUE-small*), SST-2, MNLI, QNLI and QQP (categorized as *GLUE-large*). We also consider Multimodal-IMDb (Arevalo et al., 2017) and Hateful meme classification (Kiela et al., 2020), two multimodal classification tasks comprising text and image modalities. Additionally, we consider five commonsense reasoning tasks – HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2021), ARC-easy, ARC-challenge (Clark et al., 2018) and OpenBookQA (Mihaylov et al., 2018) and three mathematical reasoning tasks – MultiArith (Roy & Roth, 2016), SingleEq (Koncel-Kedziorski et al., 2016) and AQuA (Ling et al., 2017). For instruction-following experiments, we fine-tune the student models on Dolly-15K Gu et al. (2024) dataset and evaluate on Dolly, SelfInst Wang et al. (2022a), Vicuna Chiang et al. (2023), SNI Wang et al. (2022b) and UNI Honovich et al. (2023) datasets.

Student and teacher models. We consider seven different pre-trained language models, including BERT-base (Devlin et al., 2018) (111M parameters), pre-trained BERT 6-layer (66M parameters), DeBERTa-v2-xxlarge (He et al., 2020) (1.4B parameters), OPT-1.3B (Zhang et al., 2022), LLaMA-7B (Touvron et al., 2023), LLaMA-3.1-8B (Dubey et al., 2024) and LLaMA-3.2-3B*. We utilize two text-to-image teacher models: Stable Diffusion v1-4 (Rombach et al., 2022a) and Midjourney (Borji, 2022), along with the ModelScope text-to-video (Wang et al., 2023) (1.7B parameters) and the SpeechT5 text-to-audio model (Ao et al., 2022) (144M parameters). For the text-to-image teacher models, we generate latent representations that are 32 $\times$ 32 dimensions (flattened to a 1024-dimensional vector) for each text input, using 20 diffusion inference steps and a guidance scale value of 7.5. Similar configurations apply for extracting latent representations from the text-to-video model, where 64 frames are generated for each text input. We extract the hidden representation from the first frame, resulting in a 512 $\times$ 512 hidden representation that is flattened into a 262144-dimensional vector for each text input. For the text-to-audio teacher model, we generate audio corresponding to each text input, which is subsequently processed by the Wav2Vec2-base model (Baevski et al., 2020) to produce a 768-dimensional vector representation. These vector representations of the texts are utilized to pre-train the `TS Aligner`. For natural language understanding (NLU) tasks, we employ cross-entropy loss for the pre-training of the `TS Aligner`. Conversely, for commonsense reasoning and instruction-tuning tasks, the `TS Aligner` is pre-trained using reconstruction loss measured by the mean squared error.

Evaluation metrics. For CoLA and STS-B tasks, we use the Mathews correlation coefficient ('Mcc') and Pearson's correlation coefficient ('Pear'), respectively, for evaluation. For the remaining NLU and commonsense-reasoning tasks, we use the accuracy score ('Acc'). For evaluating on instruction-tuning tasks, we use 'RougeL' metric. **Reporting protocol.** We report *mean* performance across three random seeds ($\{7, 42, 100\}$) rather than the maximum over runs, and base all significance claims on paired statistical tests over the per-seed results (Section 6 and Appendix F). The full per-seed mean$\pm$std values are reported in the accompanying result tables; for the GLUE-small/SuperGLUE NLU suite the seed-to-seed standard deviation is small (typically < 0.5 pp), but we no longer rely on best-of-n selection for any headline number.

Baselines. The competing baselines include unimodal KD methods (KD (Hinton et al., 2015), Sequence-KD (Kim & Rush, 2016), PKD (Sun et al., 2019), TinyBERT (Jiao et al., 2019), MetaDistil (Zhou et al., 2021) and Rail KD (Haidar et al., 2022)) and multimodal KD methods (VL-BERT (Su et al., 2019), VisualBERT (Li

*The pre-trained models are obtained from `https://huggingface.co/models/`.

et al., 2019), ViLBERT (Lu et al., 2019), Vokenization (Tan & Bansal, 2020), VidLanKD (Tang et al., 2021), Visual guidance (Zhang et al., 2021) and X-adapter (Zhang et al., 2023)). Appendix C presents more details.

Models	Distillation type	CoLA Mcc	MRPC Acc	RTE Acc	STS-B Pear	CB Acc	COPA Acc	BoolQ Acc	WiC Acc
BERT-6L	undistilled	42.8±0.1	78.6±0.0	64.1±0.2	88.5±0.4	75.0±0.0	53.0±0.0	71.3±0.1	55.7±0.2
	ARMADA - $\mathcal{L}_{cosine}$	**46.1±0.1**	**80.3±0.0**	**67.5±0.0**	88.5±0.1	**76.8±0.0**	**62.8±0.1**	**73.1±0.0**	**57.8±0.1**
	ARMADA - $\mathcal{L}_{euclid}$	**47.6±0.0**	**80.5±0.0**	**67.4±0.1**	88.5±0.2	**78.5±0.1**	**63.0±0.0**	**73.1±0.0**	**58.1±0.1**
	ARMADA - $\mathcal{L}_{elementwise}$	**45.7±0.2**	**79.5±0.0**	**67.5±0.1**	88.6±0.1	**78.6±0.0**	**63.0±0.0**	**73.0±0.1**	**57.9±0.0**
BERT-base	undistilled	60.8±0.1	83.8±0.1	71.0±0.0	89.3±0.0	83.7±0.1	62.9±0.0	75.1±0.1	58.4±0.1
	ARMADA - $\mathcal{L}_{cosine}$	**61.6±0.0**	**85.0±0.0**	**72.5±0.1**	**89.5±0.1**	**85.7±0.0**	**67.7±0.2**	**75.8±0.0**	58.5±0.0
	ARMADA - $\mathcal{L}_{euclid}$	**61.9±0.0**	**85.2±0.1**	**73.9±0.0**	**89.7±0.0**	**85.7±0.0**	**67.9±0.1**	75.1±0.0	**58.8±0.0**
	ARMADA - $\mathcal{L}_{elementwise}$	**61.8±0.0**	**85.0±0.0**	**71.7±0.1**	**89.6±0.0**	**87.4±0.1**	**68.0±0.0**	**75.3±0.0**	58.4±0.0
DeBERTa-v2-xxlarge	undistilled	72.1±0.0	89.7±0.1	90.3±0.0	92.3±0.0	81.9±0.1	83.7±0.0	87.2±0.0	60.8±0.1
	ARMADA - $\mathcal{L}_{cosine}$	**72.6±0.0**	89.9±0.0	**92.3±0.0**	**92.4±0.1**	**92.8±0.0**	81.9±0.1	86.2±0.0	59.2±0.0
	ARMADA - $\mathcal{L}_{euclid}$	**73.1±0.0**	**90.2±0.0**	**91.6±0.1**	92.3±0.0	**92.8±0.0**	78.9±0.1	**87.7±0.0**	58.2±0.0
	ARMADA - $\mathcal{L}_{elementwise}$	**73.4±0.0**	89.4±0.1	**91.6±0.0**	92.0±0.1	**93.0±0.0**	79.9±0.1	**87.8±0.0**	59.9±0.0
OPT-1.3B	undistilled	59.6±0.0	84.3±0.1	76.3±0.0	89.7±0.0	78.2±0.1	54.8±0.0	74.6±0.0	57.9±0.1
	ARMADA - $\mathcal{L}_{cosine}$	**62.6±0.0**	84.5±0.0	**77.3±0.0**	**90.4±0.1**	**78.5±0.0**	**57.9±0.1**	**77.3±0.0**	56.7±0.0
	ARMADA - $\mathcal{L}_{euclid}$	**61.7±0.0**	83.1±0.1	**79.0±0.0**	**90.1±0.0**	76.7±0.1	**57.0±0.0**	**76.0±0.0**	57.0±0.0
	ARMADA - $\mathcal{L}_{elementwise}$	**62.4±0.0**	**84.7±0.0**	**78.6±0.1**	**90.0±0.0**	76.7±0.1	**58.9±0.0**	**76.1±0.0**	56.1±0.0

Table 1: Results of different LMs (undistilled and distilled by ARMADA with Stable Diffusion teacher) on ***GLUE-small*** and ***SuperGLUE*** tasks - different training paradigms. **Blue** indicates the tasks where the student model distilled - ARMADA exhibits better performance than the undistilled fine-tuned variants. We report the mean (± standard deviation) over three seeds ($\{7, 42, 100\}$). The improvement of ARMADA over the undistilled model across these GLUE-small/SuperGLUE tasks is statistically significant (one-sided paired t-test, $t = 3.98$, $p < 0.01$).

5 Experimental Results

5.1 Results on NLU and Reasoning Tasks

Effectiveness of ARMADA on different LLMs. We first investigate whether cross-modal supervision can consistently benefit language models of different architectures and scales. Table 1 reports the performance of ARMADA on GLUE-small and SuperGLUE benchmarks, while Table 8 (Appendix E) presents the corresponding results on GLUE-large tasks. Across all student architectures, ARMADA consistently improves upon the undistilled baselines, indicating that the benefits of cross-modal distillation are not restricted to a particular model family or parameter regime. The largest gains are observed for the smaller BERT variants, where BERT-base and BERT-6L improve by 2.8% and 3.4%, respectively, when aggregated across Tables 1 and 8. Importantly, these gains persist even for substantially larger models, with DeBERTa and OPT improving by 1.4% and 1.5%, respectively, despite their considerably stronger starting performance. This pattern suggests that the information transferred through cross-modal supervision is complementary to the knowledge already encoded in modern pre-trained language models rather than merely compensating for limited model capacity.

To examine whether these benefits extend beyond language understanding, we evaluate ARMADA on commonsense and mathematical reasoning benchmarks using LLaMA-7B. Table 2 shows that ARMADA improves performance on five of the eight reasoning tasks, with gains reaching up to 2.6% on individual benchmarks and an average improvement of 0.5%. Although the average gain is smaller than that observed on NLU tasks, the consistent improvements across both understanding and reasoning suggest that the transferred signal captures broadly useful semantic structure rather than task-specific heuristics. One-sided t-tests further confirm that the improvements over the undistilled baselines are statistically significant across GLUE-small, GLUE-large, and reasoning benchmarks (GLUE-small: $t = 3.98$, $p = 0.00$; GLUE-large: $t = 5.11$, $p = 0.00$; reasoning: $t = 2.0$, $p = 0.04$). These findings are particularly noteworthy because the cross-modal teacher representations are obtained from a frozen Stable Diffusion model that is never optimized for any of the downstream language tasks, suggesting that multimodal foundation models encode transferable semantic structure that can be successfully incorporated into language-only systems.

Where ARMADA helps less. The benefits of cross-modal supervision are not uniform across all tasks. For example, WiC exhibits little or even negative improvement in several configurations, while the average gains

Distillation type	Hellaswag	Winogrande	ARC-c	ARC-e	OpenBookQA	MultiArith	SingleEq	AQuA	Avg.
undistilled	65.2	70.5	65.5	80.9	75.6	96.0	83.5	14.8	69.0
ARMADA with $\mathcal{L}_{cosine}$	**67.8**	69.5	64.7	**81.1**	**76.8**	**97.0**	83.1	**15.9**	**69.5**
ARMADA with $\mathcal{L}_{euclid}$	**67.0**	70.0	**65.7**	80.8	**76.5**	**96.5**	83.3	**15.6**	**69.4**
ARMADA with $\mathcal{L}_{elementwise}$	**65.8**	69.6	64.3	80.8	**75.9**	**96.2**	83.1	**15.2**	68.9

Table 2: Zero-shot accuracy on reasoning tasks for LLaMA-7B distilled with ARMADA with Stable Diffusion teacher model. **Blue** indicates the tasks where ARMADA improves performance over the undistilled LLaMA. Averaged over the eight reasoning tasks, the improvement is modest (∼0.5%) but statistically significant (one-sided paired t-test, $t = 2.0$, $p = 0.04$).

Methods	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST-2	STS-B	Avg.
KD (Hinton et al., 2015)	43.5	79.3	79.4	85.6	89.6	64.4	87.8	89.2	77.4
PKD (Sun et al., 2019)	43.9	79.4	78.9	85.9	89.6	64.3	87.9	89.2	77.4
TinyBERT (Jiao et al., 2019)	41.8	**80.3**	80.7	86.2	89.3	64.4	88.5	89.8	77.6
RCO (Jin et al., 2019)	43.0	79.1	79.3	86.1	89.3	64.3	88.0	89.2	77.3
TAKD (Mirzadeh et al., 2020)	43.2	79.2	79.2	86.0	89.4	65.0	88.0	89.3	77.4
DML (Zhang et al., 2018)	43.1	79.1	79.3	86.0	89.0	64.9	88.1	89.2	77.3
ProKT (Shi et al., 2020)	43.7	79.5	80.5	86.1	89.6	64.9	87.9	89.6	77.7
SFTN (Park et al., 2021)	43.0	79.1	79.5	85.9	89.1	65.0	88.1	89.3	77.4
MetaDistil (Zhou et al., 2021)	**48.0**	80.2	81.0	86.8	89.7	66.1	88.9	**90.7**	78.9
Continuation KD (Jafari et al., 2022)	42.7	78.6	**84.3**	86.0	**91.3**	64.9	89.7	89.1	78.3
Rail KD (Haidar et al., 2022)	45.9	78.6	79.3	85.7	90.0	**68.3**	89.4	90.0	78.4
ARMADA	47.6	**80.3**	80.5	**87.3**	89.2	67.5	**90.7**	88.7	**79.0**

Table 3: Improvement for BERT-6L on GLUE tasks distilled with ARMADA and other contemporary unimodal distillation methods. For all the KD baselines except ARMADA, BERT-base ($110M$ learnable parameters) is used as the teacher model. ARMADA uses Stable Diffusion as the teacher model ($0.8M$ learnable parameters). ARMADA attains the best average improvement (+2.2%); the gain over the unimodal and multimodal KD baselines is statistically significant (one-sided t-test, $t = 1.56$, $p < 0.1$).

on zero-shot reasoning remain comparatively modest. Similarly, the instruction-tuning improvements for larger models such as LLaMA-8B (Appendix E, Table 14) are generally smaller than those observed for compact models. These cases are informative because they help delineate the boundary conditions under which cross-modal supervision is most useful. Broadly, ARMADA appears most beneficial when tasks can plausibly exploit grounded semantic abstractions, whereas tasks that are already well served by language-only representations provide less room for improvement. We revisit this phenomenon in Section 6 through prediction-level analyses that characterize the examples most affected by cross-modal supervision.

Comparison against baselines. Having established that ARMADA consistently improves a diverse collection of language models, we next investigate whether these gains remain competitive against existing distillation approaches. Tables 3 and 11 compare ARMADA against representative unimodal and multimodal KD methods, respectively.[†] On GLUE benchmarks, ARMADA delivers the strongest overall improvement for the BERT-6L student, achieving an average gain of 2.2% over the undistilled model and surpassing the most competitive unimodal baseline, MetaDistil, despite relying on substantially fewer trainable teacher parameters. The comparison against multimodal KD methods is particularly noteworthy. Methods such as VidLanKD and X-adapter depend on extensive modality-specific pre-training and significantly larger training budgets, whereas ARMADA attains comparable performance using less than 0.8% of the corresponding teacher-training effort. These results suggest that much of the transferable semantic structure required for cross-modal learning is already present within off-the-shelf multimodal foundation models and can be exploited effectively without the expensive multimodal pre-training pipelines traditionally associated with cross-modal distillation. A one-tailed t-test further indicates that the improvements achieved by ARMADA over the unimodal and multimodal KD baselines are statistically significant ($t = 1.56$, $p < 0.1$), reinforcing the competitiveness of the proposed framework.

[†]Following Zhang et al. (2021), we only compare the multimodal KD methods on GLUE-large tasks.

Additional results. While the preceding experiments establish the effectiveness of ARMADA on language understanding, reasoning, and instruction-following tasks, a broader collection of analyses provides further evidence regarding its robustness, generality, and underlying mechanisms. To maintain readability, we defer these results to Appendix E. Table 8 extends the evaluation to the complete GLUE-large benchmark and demonstrates that the trends observed on GLUE-small and SuperGLUE persist on substantially larger datasets, indicating that the benefits of cross-modal supervision are not restricted to low-resource settings. Tables 9 and 10 investigate the role of the teacher model and the standalone predictive ability of TS Aligner, respectively. These experiments help disentangle the contribution of the alignment mechanism from the capacity of the auxiliary model and show that the gains arise from teacher–student interaction rather than from the aligner itself acting as a strong classifier.

We further compare ARMADA against multimodal distillation baselines on GLUE-large tasks in Table 11, where the proposed framework remains competitive despite requiring substantially less multimodal pre-training and optimization effort. Table 13 evaluates ARMADA on MM-IMDb and Hateful Memes, providing evidence that the benefits of cross-modal transfer are not limited to language-only benchmarks and can also extend to tasks where multimodal reasoning is intrinsically important. The contribution of auxiliary supervision is isolated in Table 12, revealing that the auxiliary objectives consistently improve the quality of the learned alignment and contribute to downstream performance gains. Finally, Figures 2 and 3 analyze the sensitivity of ARMADA to hyperparameter choices and manifold-alignment objectives. Collectively, these experiments demonstrate that the effectiveness of ARMADA is robust across teacher choices, task domains, alignment objectives, and training configurations, while simultaneously providing additional evidence that the observed improvements originate from semantically meaningful cross-modal supervision rather than incidental architectural choices.

Model	**CoLA**	**MRPC**	**RTE**	**STS-B**	**CB**	**COPA**	**BoolQ**	**WiC**
	Mcc	Acc	Acc	Pear	Acc	Acc	Acc	Acc
TS Aligner	9.9	66.5	54.5	2.1	51.8	54.0	53.9	52.3
ARMADA with $\mathcal{L}_{cosine}$	59.3	84.3	69.7	89.3	85.7	62.0	73.2	57.2
ARMADA with $\mathcal{L}_{euclid}$	59.3	82.4	69.7	89.3	85.7	54.0	76.2	56.4
ARMADA with $\mathcal{L}_{elementwise}$	59.3	82.4	69.7	89.3	83.9	55	76.7	55.6

Table 4: Results of TS Aligner and BERT-12L with ARMADA on all GLUE-small and SuperGLUE tasks with a higher-capacity TS Aligner. We increase the number of hidden layers in TS Aligner and remove the projection mapping $\tilde{F}_{ts}$ and auxiliary output head $O^{\mathcal{T}}_{aux_{ts}}$, so that capacity is preserved (or increased) while the alignment mechanism is removed. The objective is to corroborate that the gains from ARMADA are not attributable to aligner capacity. The capacity-matched variant is on average 2.8 pp worse than the full ARMADA (with alignment); the drop is statistically significant (one-sided paired t-test, $p < 0.05$), confirming that structured alignment, not parameter count, drives the improvement.

Are the gains simply due to additional capacity? A natural alternative explanation for the improvements observed with ARMADA is that they arise from the additional parameters introduced by the alignment architecture rather than from cross-modal knowledge transfer itself. To isolate this effect, we construct a capacity-matched control in which the number of hidden layers in TS Aligner is increased to match and slightly exceed the total parameter count contributed by the projection module $\tilde{F}_{ts}$ and the auxiliary output head $O^{\mathcal{T}}_{aux_{ts}}$, while removing both alignment mechanisms. This design preserves the parameter budget while eliminating the structured teacher-student correspondence central to ARMADA. Table 4 presents the results on GLUE and SuperGLUE benchmarks under this higher-capacity configuration. If additional capacity were the primary driver of performance, this variant would be expected to recover the gains achieved by the full model. Instead, it exhibits an average performance degradation of 2.8% relative to ARMADA, with the difference being statistically significant ($p < 0.05$). Notably, the capacity-matched model fails to reproduce the consistent 1–3% improvements observed throughout the benchmark suite despite possessing comparable or greater parameter count. These findings rule out parameter expansion as the dominant explanation for the observed gains and instead point to the importance of structured cross-modal alignment. The projection mapping and auxiliary alignment objectives appear to be critical for establishing semantically meaningful

teacher–student correspondence, suggesting that effective cross-modal transfer depends not on capacity alone but on how that capacity is used to align representations across modalities.

5.2 Results on Instruction-tuning Tasks

Teacher	Distillation Type	Dolly	SNI	SelfInst	UNI	Vicuna
-	undistilled	31.30	26.68	18.44	32.01	17.59
language-only	SeqKD	31.45	28.48	18.61	31.05	17.74
text-to-image	ARMADA with $\mathcal{L}_{cosine}$	**31.94**	**27.13**	**19.83**	**33.57**	**18.73**
	ARMADA with $\mathcal{L}_{euclid}$	30.57	**28.25**	18.06	31.59	**18.15**
	ARMADA with $\mathcal{L}_{elementwise}$	**31.38**	**26.68**	18.28	**32.27**	**18.16**
text-to-video	ARMADA with $\mathcal{L}_{cosine}$	**33.02**	**26.98**	**18.49**	**32.07**	**18.44**
	ARMADA with $\mathcal{L}_{euclid}$	**33.07**	**27.75**	**18.96**	31.51	**17.92**
	ARMADA with $\mathcal{L}_{elementwise}$	**31.85**	26.57	**18.84**	**32.01**	**17.84**
text-to-audio	ARMADA with $\mathcal{L}_{cosine}$	**32.95**	**27.46**	**18.92**	**33.67**	**18.15**
	ARMADA with $\mathcal{L}_{euclid}$	**32.68**	**27.70**	**18.66**	**32.70**	**18.01**
	ARMADA with $\mathcal{L}_{elementwise}$	**32.81**	**28.07**	**19.57**	**33.01**	**18.67**

Table 5: Results on instruction-tuning tasks (RougeL) with LLaMA-3.2-3B under different multimodal teacher models and the three loss objectives. We also report the RougeL when distilled from a language-only LLaMA-8B teacher with SeqKD (Kim & Rush, 2016). ARMADA outperforms SeqKD (one-sided t-test, $p = 0.005$) while using teacher models that are far smaller ($<$ 2B vs. 8B). A two-way ANOVA finds no significant difference between teacher modalities ($p > 0.05$; we fail to reject equality), whereas the manifold-weight β and auxiliary-weight γ are each significant ($p < 0.01$).

We next investigate whether the benefits of cross-modal distillation extend beyond classification and reasoning tasks to open-ended instruction following. Table 5 reports the performance of LLaMA-3.2-3B-Instruct distilled from different multimodal teacher models using ARMADA. Across all five evaluation benchmarks, ARMADA consistently improves upon the undistilled LLaMA-3B baseline, achieving gains of up to 1.8%, 1.6%, and 1.5% on Dolly, UNI, and SNI, respectively. The consistency of these improvements across multiple instruction-following datasets suggests that the transferred cross-modal signal remains useful even when evaluation shifts from structured prediction tasks to open-ended generation. Perhaps more surprisingly, ARMADA frequently outperforms SeqKD, which performs unimodal distillation from a substantially larger LLaMA-8B teacher. On datasets such as Dolly, SelfInst, and UNI, the margin exceeds 1%, despite ARMADA relying on comparatively lightweight multimodal teacher models. This observation suggests that the value of the transferred knowledge is not determined solely by teacher scale; rather, multimodal teachers appear to provide complementary semantic information that is difficult to obtain through text-only supervision.

Across modalities, the text-to-audio teacher yields the strongest average improvement (1.4%), followed closely by the text-to-image teacher (1.3%). The teacher modalities are nonetheless statistically equivalent ($p > 0.05$), indicating that the effectiveness of ARMADA is not tied to any specific modality. Instead, the results indicate that multiple forms of multimodal supervision can provide useful semantic structure that can be distilled into language-only models. Hyperparameter and loss-function ablations for the LLaMA-3B student (Figures 4 and 5) and results with the larger LLaMA-8B student (Table 14) are reported in Appendix E.

6 Discussion

The central question underlying this work is whether multimodal foundation models contain semantic structure that can be transferred across modality boundaries into language-only systems. While the empirical results demonstrate consistent improvements across language understanding, reasoning, and instruction-following tasks, they do not by themselves explain *why* cross-modal supervision helps. We therefore complement the benchmark evaluations with a series of representation-level, optimization-level, and statistical analyses aimed at understanding three questions: *(i)* how cross-modal distillation influences the learned representation space; *(ii)* what role the trained TS Aligner plays during optimization; and *(iii)* whether the observed gains

originate from semantically aligned cross-modal supervision rather than generic regularization or increased capacity.

How does the cross-modal distillation work? Projecting the student and `TS Aligner` into a shared manifold imposes a soft constraint that encourages the student's internal representations to remain predictable from the teacher's cross-modal abstractions while preserving information from the original language modality. Unlike direct feature matching, which can distort modality-specific representations, the manifold objective encourages consistency at the level of latent structure rather than individual features. We quantify this effect through three complementary representation-space probes on BERT-12L. First, t-SNE visualizations (Appendix Figure 6) show that students trained with $\mathcal{L}_{elementwise}$ occupy regions of the latent space that are substantially closer to the teacher manifold than those of the undistilled model. Second, silhouette scores (Appendix Table 17) increase by 76% on examples where `ARMADA` improves upon the baseline and decrease by 20% where it does not, suggesting that performance gains are associated with more coherent semantic organization rather than uniform shifts of the representation space. Third, cluster purity increases from 0.77/0.56 (CoLA/COPA) for the undistilled model to 0.81/0.68 for `ARMADA`, indicating improved separation of task-relevant concepts.

The effect is further reflected at the prediction level. Among the examples on which `ARMADA` improves over the undistilled baseline, corrections concentrate on specific categories (Appendix Figure 7) — for instance, syntactic CoLA constructions (>70% of corrected cases) and Social & Institutional COPA examples (∼42%). We present this as a description of the tasks and example categories that most benefit from `ARMADA` as a full training pipeline, rather than as an isolation of the paired-image contribution specifically; against the Shuffled-Input control, the per-task McNemar tests (Appendix Table 15) on CoLA and COPA do not reach significance, so this categorization should not be read as direct evidence of cross-modal transfer. The identification and quantification of the cross-modal contribution is carried out separately via the mechanism analysis in this section.

How does the `TS Aligner` affect the student? A notable aspect of `ARMADA` is that the `TS Aligner` itself is intentionally weak. Despite containing only ~ 0.8M parameters and achieving relatively poor standalone performance on downstream tasks (Table 10), it nevertheless provides substantial benefit to the student during training. When the student is trained against a randomly initialized frozen `TS Aligner`, performance deteriorates relative to the undistilled baseline on six of eight NLU tasks, whereas the standard training curriculum consistently improves performance (Appendix Figure 8 and Appendix Table 18). This suggests that the value of the aligner does not arise from its ability to solve the task independently. Instead, the aligner appears to act as a conduit through which the multimodal teacher imposes a structured learning signal on the student. In other words, the utility of the aligner lies not in its final predictive accuracy but in the trajectory it induces during optimization.

Does the cross-modal signal drive the gains? The preceding analyses suggest that cross-modal supervision influences the learned representation space, but they do not rule out alternative explanations such as generic regularization or increased model capacity. We therefore perform a controlled mechanism study using progressively degraded teacher signals. Specifically, we compare `ARMADA` against a Frozen Teacher, Randomly Frozen Teacher, Random Teacher, Shuffled-Input Teacher, and Constant Teacher. These controls systematically remove semantic correspondence while preserving varying degrees of architecture, supervision, and optimization dynamics. All configurations are evaluated across seven NLU tasks, three random seeds, and three manifold objectives.

Table 6 shows that `ARMADA` significantly outperforms the undistilled baseline ($\bar{\Delta} = +1.59$, $p = 0.002$), the capacity-matched control (Table 4, $p < 0.05$), and every fully degraded teacher configuration. These results indicate that neither parameter count nor generic auxiliary supervision is sufficient to explain the observed gains. The strongest controls are the Random and Shuffled-Input variants, which preserve most of the optimization machinery while disrupting semantic correspondence. Although both comparisons remain directionally favorable to `ARMADA`, neither reaches statistical significance at the aggregate level. This observation is important because it suggests that a portion of the benefit may indeed arise from regularization effects associated with the alignment procedure itself, even though semantic correspondence provides additional measurable value.

Comparison	Aggregation	n	$\bar{\Delta}$ (pp)	Wins/n	Cohen's d	Wilcoxon p
ARMADA > Undistilled (Base)	max-over-loss	21	+1.59	16/21	0.51	0.002^{**}
ARMADA > Constant Teacher	max-over-loss	21	+0.94	17/21	0.31	0.002^{**}
ARMADA > Frozen Teacher	per-loss	63	+1.10	35/63	0.31	0.018^{*}
ARMADA > Randomly Frozen Teacher	per-loss	63	+0.68	35/63	0.22	$0.052^{\dagger}$
ARMADA > Random Teacher	per-loss	63	+0.44	32/63	0.15	0.140
ARMADA > Shuffled Input	per-loss	63	+0.44	37/63	0.12	0.136

Table 6: One-sided paired Wilcoxon tests comparing ARMADA against the undistilled baseline and five negative controls (best aggregation slice, pooled across 3 seeds). ARMADA improves over *every* comparator ($\bar{\Delta} > 0$) and separates significantly ($p < 0.1$) from the undistilled baseline and from every fully-degraded teacher (Constant, Frozen, Randomly-Frozen). The two strictest controls — Random and Shuffled Input, which retain full task supervision and therefore most of ARMADA's capacity — are directionally favourable but short of aggregate significance, and are examined at the prediction level in Table 15. Significance markers: ** $p < 0.01$, * $p < 0.05$, † $p < 0.1$.

Beyond the aggregate test, we examine prediction-level behavior with a per-task McNemar analysis pooled across seeds (Appendix Table 15), where the counts b and c record the examples ARMADA corrects that the comparator misses and vice versa. Across the panel, ARMADA corrects more predictions than it breaks against both the undistilled baseline and the strict Shuffled-Input control, with the largest correction margins concentrated on tasks whose solution plausibly relies on grounded semantic abstractions. The per-task significance pattern is consistent with the aggregate Wilcoxon picture: clear separation from the undistilled baseline on the more semantically demanding tasks, and a directional but narrower advantage over the Shuffled-Input control once predictions are pooled across the full panel.

Taken together, the evidence supports a nuanced interpretation of cross-modal transfer. The improvements cannot be explained solely by additional parameters, generic auxiliary objectives, or trivial regularization effects, as demonstrated by the capacity-matched and degraded-teacher controls. At the same time, the remaining gap between ARMADA and the strongest shuffled-input baseline is relatively modest, indicating that the contribution of semantic transfer and regularization are not completely separable within the current experimental framework. We therefore view our results not as definitive proof of cross-modal semantic transfer, but as strong empirical evidence that semantically aligned multimodal representations contain information that language-only models can exploit. Developing sharper causal tests, stronger negative controls, and more challenging grounded reasoning benchmarks remains an important direction for future work.

7 Conclusion

In this work, we presented ARMADA, a cross-modal knowledge distillation framework that enabled language-only models to acquire knowledge from frozen multimodal teachers without requiring modality-specific teacher pre-training, multimodal inference architectures, or access to teacher internals. By introducing a lightweight alignment mechanism that projected teacher and student representations into a shared latent manifold, ARMADA provided a practical and architecture-agnostic pathway for transferring cross-modal abstractions into language models while preserving their efficiency and deployment simplicity. Beyond proposing a new distillation framework, our study explored a broader question of whether semantic structure encoded within multimodal foundation models can be transferred across modality boundaries and subsequently utilized by language-only systems. The empirical and mechanistic analyses suggested that the observed improvements were not explained by increased capacity or degraded auxiliary supervision alone; separation from the strictest shuffled-teacher control is directional rather than statistically conclusive at the aggregate level, with the clearest per-task prediction-level evidence on a single grounded-inference task. Nevertheless, the precise mechanisms underlying such transfer remain only partially understood, particularly in the presence of strong regularization-based controls, and our evaluation was limited to language understanding, reasoning, and instruction-following tasks. Future work could therefore focus on developing a deeper theoretical understanding of cross-modal knowledge transfer, extending the framework to longer-horizon generation and interactive

reasoning settings, and exploring its integration with emerging on-policy distillation and foundation-model compression paradigms.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024.

Rohan Anil, Gabriel Pereyra, Alexandre Passos, Robert Ormandi, George E Dahl, and Geoffrey E Hinton. Large scale distributed neural network training through online distillation. *arXiv preprint arXiv:1804.03235*, 2018.

Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, Zhihua Wei, Yao Qian, Jinyu Li, and Furu Wei. SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5723–5738, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.393. URL `https://aclanthology.org/2022.acl-long.393/`.

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A González. Gated multimodal units for information fusion. *arXiv preprint arXiv:1702.01992*, 2017.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020. URL `https://arxiv.org/abs/2006.11477`.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. The fifth pascal recognizing textual entailment challenge. *TAC*, 7:8, 2009.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

Ali Borji. Generated faces in the wild: Quantitative comparison of stable diffusion, midjourney and dall-e 2. *arXiv preprint arXiv:2210.00586*, 2022.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–541, 2006.

Yihan Cao, Yanbin Kang, Zhengming Xing, and Ruijie Jiang. Delta knowledge distillation for large language models. *arXiv preprint arXiv:2509.14526*, 2025.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*, 2017.

Hanting Chen, Yunhe Wang, Chang Xu, Chao Xu, and Dacheng Tao. Learning student networks via feature embedding. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):25–35, 2020.

Zihan Chen, Hongbo Zhang, Xiaoji Zhang, and Leqi Zhao. Quora question pairs, 2018.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL `https://lmsys.org/blog/2023-03-30-vicuna/`.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pp. 177–190. Springer, 2005.

Marie-Catherine De Marneffe, Mandy Simons, and Judith Tonhauser. The commitmentbank: Investigating projection in naturally occurring discourse. In *proceedings of Sinn und Bedeutung*, volume 23, pp. 107–124, 2019.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. Deepseek-v3 technical report, 2024. URL `https://arxiv.org/abs/2412.19437`.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Bill Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*, 2005.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Sayontan Ghosh, Tanvi Aggarwal, Minh Hoai, and Niranjan Balasubramanian. Text-derived knowledge helps vision: A simple cross-modal distillation for video-based action anticipation. In *Findings of the Association for Computational Linguistics: EACL 2023*, pp. 1837–1852, 2023.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and William B Dolan. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pp. 1–9, 2007.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2827–2836, 2016.

Md Akmal Haidar, Nithin Anchuri, Mehdi Rezagholizadeh, Abbas Ghaddar, Philippe Langlais, and Pascal Poupart. RAIL-KD: RAndom intermediate layer mapping for knowledge distillation. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (eds.), *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 1389–1400, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.103. URL `https://aclanthology.org/2022.findings-naacl.103`.

R Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, volume 7, pp. 785–794, 2006.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. Unnatural instructions: Tuning language models with (almost) no human labor. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14409–14428, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.806. URL `https://aclanthology.org/2023.acl-long.806/`.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Ka-Wei Lee. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*, 2023.

Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*, 2017.

Jing Huo, Shiyin Jin, Wenbin Li, Jing Wu, Yu-Kun Lai, Yinghuan Shi, and Yang Gao. Manifold alignment for semantically aligned style transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14861–14869, 2021.

Aref Jafari, Ivan Kobyzev, Mehdi Rezagholizadeh, Pascal Poupart, and Ali Ghodsi. Continuation kd: Improved knowledge distillation through the lens of continuation optimization. *arXiv preprint arXiv:2212.05998*, 2022.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023a.

Yuxin Jiang, Chunkit Chan, Mingyang Chen, and Wei Wang. Lion: Adversarial distillation of proprietary large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3134–3154, 2023b.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*, 2019.

Woogyeol Jin, Taywon Min, Yongjin Yang, Swanand Ravindra Kadhe, Yi Zhou, Dennis Wei, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

Woojeong Jin, Dong-Ho Lee, Chenguang Zhu, Jay Pujara, and Xiang Ren. Leveraging visual knowledge in language tasks: An empirical study on intermediate pre-training for cross-modal knowledge transfer. *arXiv preprint arXiv:2203.07519*, 2022.

Xiao Jin, Baoyun Peng, Yichao Wu, Yu Liu, Jiaheng Liu, Ding Liang, Junjie Yan, and Xiaolin Hu. Knowledge distillation via route constrained optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1345–1354, 2019.

Seongryong Jung, Suwan Yoon, DongGeon Kim, and Hwanhee Lee. Todi: Token-wise distillation via fine-grained divergence control. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8089–8102, 2025.

Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, Ethan Perez, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *arXiv preprint arXiv:1909.02950*, 2019.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *Advances in neural information processing systems*, 33:2611–2624, 2020.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*, 2016.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024.

Jongwoo Ko, Tianyi Chen, Sungnyun Kim, Tianyu Ding, Luming Liang, Ilya Zharkov, and Se-Young Yun. DistiLLM-2: A contrastive approach boosts the distillation of LLMs. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=rc65N9xIrY`.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. Mawps: A math word problem repository. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, pp. 1152–1157, 2016.

Hector Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *Thirteenth international conference on the principles of knowledge representation and reasoning*, 2012.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

Yang Li, Erik Nijkamp, Semih Yavuz, and Shafiq Rayhan Joty. Learning from language feedback via variational policy distillation. *arXiv preprint arXiv:2605.15113*, 2026.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *arXiv preprint arXiv:1705.04146*, 2017.

Chang Liu, Chongyang Tao, Jiazhan Feng, and Dongyan Zhao. Multi-granularity structural knowledge distillation for language model compression. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1001–1011, 2022.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5191–5198, 2020.

Geondo Park, Gyeongman Kim, and Eunho Yang. Distilling linguistic context for language model compression. *arXiv preprint arXiv:2109.08359*, 2021.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3967–3976, 2019.

Mohammad Taher Pilehvar and Jose Camacho-Collados. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. *arXiv preprint arXiv:1808.09121*, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*, 2011.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022a.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022b.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Subhro Roy and Dan Roth. Solving general arithmetic word problems. *arXiv preprint arXiv:1608.01413*, 2016.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Ayan Sengupta, Shantanu Dixit, Md Shad Akhtar, and Tanmoy Chakraborty. A good learner can teach better: Teacher-student collaborative knowledge distillation. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=Ixi4j6LtdX`.

Wenxian Shi, Yuxuan Song, Hao Zhou, Bohan Li, and Lei Li. Learning from deep model via exploring local targets. 2020.

Aasheesh Singh, Vishal Vaddina, and Dagnachew Birru. Orpo-distill: Mixed-policy preference optimization for cross-architecture llm distillation. *arXiv preprint arXiv:2509.25100*, 2025.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *arXiv preprint arXiv:2604.00626*, 2026.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*, 2019.

Hao Tan and Mohit Bansal. Vokenization: Improving language understanding with contextualized, visual-grounded supervision. *arXiv preprint arXiv:2010.06775*, 2020.

Zineng Tang, Jaemin Cho, Hao Tan, and Mohit Bansal. Vidlankd: Improving language understanding via video-distilled knowledge transfer. *Advances in Neural Information Processing Systems*, 34:24468–24481, 2021.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report, 2023. URL `https://arxiv.org/abs/2308.06571`.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions, 2022a.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5085–5109, Abu Dhabi, United Arab Emirates, December 2022b. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.340. URL `https://aclanthology.org/2022.emnlp-main.340/`.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019.

Adina Williams, Nikita Nangia, and Samuel R Bowman. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*, 2017.

Taiqiang Wu, Chaofan Tao, Jiahao Wang, Runming Yang, Zhe Zhao, and Ngai Wong. Rethinking kullback-leibler divergence in knowledge distillation for large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5737–5755, 2025.

Tianzhu Ye, Li Dong, Zewen Chi, Xun Wu, Shaohan Huang, and Furu Wei. Black-box on-policy distillation of large language models. *arXiv preprint arXiv:2511.10643*, 2025.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

Linfeng Zhang, Yukang Shi, Zuoqiang Shi, Kaisheng Ma, and Chenglong Bao. Task-oriented feature distillation. *Advances in Neural Information Processing Systems*, 33:14759–14771, 2020.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Xinyun Zhang, Haochen Tan, Han Wu, and Bei Yu. Towards versatile and efficient visual knowledge integration into pre-trained language models with cross-modal adapters. *arXiv preprint arXiv:2305.07358*, 2023.

Xue Zhang, Songming Zhang, Yunlong Liang, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. A dual-space framework for general knowledge distillation of large language models. *arXiv preprint arXiv:2504.11426*, 2025.

Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4320–4328, 2018.

Zhuosheng Zhang, Haojie Yu, Hai Zhao, and Masao Utiyama. Which apple keeps which doctor away? colorful word representations with visual oracles. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:49–59, 2021.

Wangchunshu Zhou, Canwen Xu, and Julian McAuley. Bert learns to teach: Knowledge distillation with meta learning. *arXiv preprint arXiv:2106.04570*, 2021.

A Notation

Table 7 summarizes the notation used throughout the paper.

B Datasets

B.1 Natural Language Understanding

The General Language Understanding Evaluation (GLUE)[‡] benchmark (Wang et al., 2018) and SuperGLUE[§] benchmark (Wang et al., 2019) evaluate the language understanding capabilities of language models. We elaborate on the GLUE and SuperGLUE tasks as follows:

CoLA (Warstadt et al., 2019), or The Corpus of Linguistic Acceptability, comprises acceptability judgments from books and journal articles. The task is to indicate the grammatical correctness of the given sentence as "acceptable" or "unacceptable" using the Matthews correlation coefficient as the evaluation metric.

MRPC (Dolan & Brockett, 2005), Microsoft Research Paraphrase Corpus comprises pairs of sentences extracted from online news sources. The objective is to predict whether the provided pair of sentences are paraphrases of each other or not.

RTE (Dagan et al., 2005; Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009), short for Recognizing Textual Entailment, are formed by combining a series of annual textual entailment challenges. The task comprises categorizing whether the two sentences entail each other.

STS-B (Cer et al., 2017), or Semantic Textual Similarity, compiled from various sources, including news headlines, video-image descriptions, and NLI descriptions. The task is to predict the degree of similarity between two sentences.

SST-2 (Socher et al., 2013), Stanford Sentiment Treebank involves a two-class classification task, predicting the sentiment of the given movie review as 'positive' or 'negative'.

QQP (Chen et al., 2018), or Quora Question Pairs, is a dataset comprising questions collected from the question-answering platform - Quora. The task involves predicting whether or not the given two questions are duplicates of each other.

MNLI (Williams et al., 2017) Multi-Genre Natural Language Inference involves a three-class classification problem. The goal is to predict whether the premise "entails"/"contradicts" or is neutral towards the hypothesis statement.

QNLI (Wang et al., 2018) or Question Answering NLI is a dataset derived from the Stanford Question Answering Dataset. The task comprises a question and a sentence. The goal is to predict whether or not the given sentence contains the answer to the question.

BoolQ (Clark et al., 2019), or Boolean Questions, involves binary inquiries sourced from the Google search engine. These questions are combined with pertinent paragraphs extracted from Wikipedia articles, ensuring that the provided paragraphs contain accurate answers to the queries.

CB (De Marneffe et al., 2019), short for CommitmentBank, comprises concise texts featuring embedded clauses. These examples are extracted from diverse sources, including the British National Corpus Fiction and Wall Street Journal. The task comprises of three-class textual entailment, where each example consists of a premise and a corresponding hypothesis and is labeled as either "contradiction," "neutral," or "entailment".

COPA (Roemmele et al., 2011), abbreviated for Choice of Plausible Alternatives, is a causal reasoning task with the goal of selecting the most plausible choice between two choices given a premise and cause/effect. The examples are sourced from blogs and a photography-related encyclopedia.

WiC (Pilehvar & Camacho-Collados, 2018), short for Word-in-Context (WiC), is a word sense disambiguation task that involves binary classification of sentence pairs. Within this task, two text snippets are presented,

[‡] https://gluebenchmark.com/
[§] https://super.gluebenchmark.com/

Notation Type	Notation	Description	Dimension/Shape
Modalities	$\mathcal{M}_t$	Teacher model modality	
	$\mathcal{M}_s$	Student model modality	
Architectures	F_t	Multi-modal teacher model	
	F_{ts}	`TS Aligner` module	
	$\tilde{F}_{ts}$	Hidden non-linear projection mapping for `TS Aligner`	$\mathbb{R}^{128} \rightarrow \mathbb{R}^{h}$, h is hidden dimension of `TS Aligner`
	$O_{ts}^{\mathcal{T}}$	`TS Aligner` output layer for task $\mathcal{T}$	$\mathbb{R}^{h} \rightarrow \mathbb{R}^{c}$, h is hidden dimension of `TS Aligner`; c is number of output classes for task $\mathcal{T}$
	P_{ts}	`TS Aligner` manifold projection mapping	$\mathbb{R}^{h} \rightarrow \mathbb{R}^{768}$, h is hidden dimension of `TS Aligner`
	$O_{aux_{ts}}^{\mathcal{T}}$	`TS Aligner` auxiliary output layer for task $\mathcal{T}$	$\mathbb{R}^{768} \rightarrow \mathbb{R}^{c}$, c is number of output classes for task $\mathcal{T}$
	F_s	Student representation learning model	
	$\tilde{F}_s$	Hidden non-linear projection mapping for student model	$\mathbb{R}^{d} \rightarrow \mathbb{R}^{d}$, d is hidden dimension of student language model
	$O_s^{\mathcal{T}}$	Student output layer for task $\mathcal{T}$	$\mathbb{R}^{d} \rightarrow \mathbb{R}^{c}$, d is hidden dimension of student language model; c is number of output classes for task $\mathcal{T}$
	P_s	Student manifold projection mapping	$\mathbb{R}^{d} \rightarrow \mathbb{R}^{768}$, d is hidden dimension of student language model
	$O_{aux_s}^{\mathcal{T}}$	Student auxiliary output layer for task $\mathcal{T}$	$\mathbb{R}^{768} \rightarrow \mathbb{R}^{c}$, c is number of output classes for task $\mathcal{T}$
Inputs	x_s	Sample student input sequence	
	x_t	Teacher representation generated for x_s	
	X_s	Student input sequences $\{x_1, x_2, \cdots x_n\}$	
	X_t	Teacher generated sequences corresponding to X_s	
	$Y^{\mathcal{T}}$	Ground truth labels for inputs X_s for task $\mathcal{T}$	
Model outputs	h_t^X	Representations obtained from multi-modal teacher model on input X_t (inputs to `TS Aligner`)	$\mathbb{R}^{b \times 128}$, b is batch size
	$h_{ts}^{'X}$	`TS Aligner` hidden representation	$\mathbb{R}^{b \times h}$, h is hidden dimension of `TS Aligner`; b is batch size
	$o_{ts}^{\mathcal{T}}$	`TS Aligner` output in task $\mathcal{T}$	$\mathbb{R}^{b \times c}$, c is number of output classes for task $\mathcal{T}$; b is batch size
	p_{ts}^X	`TS Aligner` manifold projection	$\mathbb{R}^{b \times 768}$, b is batch size
	$o_{aux_{ts}}^{\mathcal{T}}$	`TS Aligner` auxiliary output in task $\mathcal{T}$	$\mathbb{R}^{b \times c}$, c is number of output classes for task $\mathcal{T}$;b is batch size
	h_s^X	Hidden representation from student representation model	$\mathbb{R}^{b \times d}$, d is hidden dimension of student language model; b is batch size
	$h_s^{'X}$	Student hidden representation	$\mathbb{R}^{b \times d}$, d is hidden dimension of student language model; b is batch size
	$o_s^{\mathcal{T}}$	Student output in task $\mathcal{T}$	$\mathbb{R}^{b \times c}$, c is number of output classes for task $\mathcal{T}$; b is batch size
	p_s^X	Student manifold projection	$\mathbb{R}^{b \times 768}$, b is batch size
	$o_{aux_s}^{\mathcal{T}}$	Student auxiliary output in task $\mathcal{T}$	$\mathbb{R}^{b \times c}$, c is number of output classes for task $\mathcal{T}$; b is batch size
Hyperparameters	α	Weightage of logit matching loss in student output loss	
	β	Weightage of manifold alignment loss in the final loss value (for both `TS Aligner` and student models)	
	γ	Weightage of auxiliary output loss in the final loss (for both `TS Aligner` and student models)	
	τ	Temperature value used in student logit matching loss	

Table 7: Glossary of all the mathematical notations in the paper and their descriptions.

each featuring a word with multiple potential meanings. The objective is to determine whether the specified word holds the same meaning in both sentences.

B.2 Multimodal Tasks

Multimodal IMDb (Arevalo et al., 2017) is a collection featuring movies complete with their plot summaries, posters, genres, and an additional set of 50 metadata fields. The dataset is constructed using IMDb IDs from the Movielens dataset.

B.3 Arithmetic Reasoning

MultiArith (Roy & Roth, 2016) consists of arithmetic word problems presented in natural language. The dataset includes various problems that require interpreting and performing basic arithmetic operations.

SingleEq (Koncel-Kedziorski et al., 2016) or Single-equation, consists of word problems with single equations of varying length.

AQuA (Ling et al., 2017) short for Algebra Question Answering with Rationales, consists of algebraic word problems with natural language rationales.

B.4 Commonsense Reasoning

Hellaswag (Zellers et al., 2019) is a Natural Language Inference (NLI) task which involves finishing the sentence given context. The dataset consists of multiple-choice questions where the task is to choose the most plausible continuation of a given sentence.

Winogrande (Sakaguchi et al., 2021) is a commonsense reasoning task inspired by the WSC (Levesque et al., 2012). The goal of the task is to choose the correct choice given two choices.

ARC-c & **ARC-e** (Clark et al., 2018) comprise the AI2 Reasoning Challenge (ARC) partitioned into easy (ARC-e) and challenging set (ARC-c). The dataset consists of grade-school science questions in multiple choice format.

OpenBook-QA (Mihaylov et al., 2018) consists of elementary level science questions in form of multiple choices requiring additional commonsense knowledge.

B.5 Instruction-tuning

Dolly Gu et al. (2024): We use a curated subset of the `databricks-dolly-15K` dataset, comprising approximately 12.5k training samples, 1k validation samples, and 500 test samples.

SelfInst Wang et al. (2022a) dataset includes 252 instruction-following examples designed around user-centric tasks.

Vicuna Chiang et al. (2023) contains 80 high-difficulty queries generated by GPT-4, which are employed for evaluating the Vicuna model.

SNI Wang et al. (2022b) derived from the Super-Natural Instructions benchmark, this dataset includes about 9,000 examples spanning 119 tasks. Following the methodology in Gu et al. (2024), we segment the data based on the length of reference answers and use the subset with response lengths greater than 10 tokens, which contains roughly 1,600 samples.

UNI Honovich et al. (2023) is shortlisted from the Unnatural Instructions dataset, we evaluate on the subset with reference responses longer than 10 tokens, using the first 2,000 samples from this segment.

C Baselines

This section elaborates on the unimodal and multimodal knowledge distillation frameworks we use in this paper for evaluating against `ARMADA`.

C.1 Unimodal KD Baselines

KD: Hinton et al. (2015) introduced the concept of distilling knowledge from a bigger complex model to a shallower model by utilizing the *dark knowledge* or the soft label distribution of the teacher model.

SeqKD: Motivated by Hinton et al. (2015), Kim & Rush (2016) proposed sequential KD, that forces the student model to generate entire sequences that match the teacher's outputs. This distillation method is particularly useful for sequence generation tasks.

DML: Zhang et al. (2018) proposed a mutual learning setup where student models guide each other's learning. The method involved matching the probability distribution of other students by minimizing their KL divergence and task-specific cross-entropy loss.

PKD: Sun et al. (2019) proposed distilling knowledge from the intermediate layers of BERT (Devlin et al., 2018) and introduced two different schemes, *PKD-last* and *PKD-skip* for distillation, where the teacher representations are transferred to the student from its last and alternate k layers, respectively.

TinyBERT: Jiao et al. (2019) proposed general and task-specific distillation to a 4 layer BERT from BERT-base (Devlin et al., 2018) through attention and hidden states distillation of transformer-based teacher to student.

TAKD: Mirzadeh et al. (2020) argued that sub-optimal distillation occurs when a large gap exists between teacher and student. To address this issue, intermediate teachers first distil their knowledge from the teacher model and, in turn, distil their knowledge to the student model.

MetaDistil: Zhou et al. (2021) proposed meta-knowledge distillation, different from the previous settings involving a fixed teacher distilling knowledge to the student. This method proposed providing a continuously updated meta-teacher through a bi-level optimization (Finn et al., 2017) for better knowledge transfer.

C.2 Multimodal KD Baselines

Vokenization: Tan & Bansal (2020) proposed visually supervising a language model through "vokens", which are visual representations of tokens. The "vokenization" process involves assigning a relevant image to each token. In addition to the traditional MLM (Devlin et al., 2018), the language model is pre-trained with an additional voken-classification (voken-cls) objective.

VidLanKD: Tang et al. (2021) proposed improving language understanding through video-distilled knowledge transfer. The teacher model is pre-trained on a video-text dataset using contrastive learning. Knowledge is then transferred to the student using a variety of KD objectives, including NST (Huang & Wang, 2017) and CRD (Tian et al., 2019).

Visual Guidance: Zhang et al. (2021) proposed enhancing word representations by incorporating visual information. The method used a word-image dictionary and an attention mechanism to create text-conditioned image representations and a gating mechanism to combine these with the text representations, improving the original embeddings.

D Experimental Settings

For all the tasks, we employ an online distillation (Anil et al., 2018; Zhang et al., 2020) framework, where we train both teacher and student simultaneously. We use $\alpha \in \{0, 0.5\}$, $\beta \in \{0, 1\}$ and $\gamma \in \{0, 1\}$. We use $\tau = 5$ in Equation 2. We use Adam optimizer with weight decay (Loshchilov & Hutter, 2017) for training all the models. For GLUE and SuperGLUE tasks, we use a maximum sequence length of 128 and train both the teacher and the student models for 10 epochs. For GLUE tasks, we use a batch size of 32 and a teacher learning rate of $3e-3$. For SuperGLUE, we use the batch size as 8 and `TS Aligner` learning rate as $1e-3$. The BERT student models are fine-tuned for all the GLUE and SuperGLUE tasks with a learning rate of $2e-5$. DeBERTa and OPT student models are trained with learning rate $3e-6$. For MM-IMDb classification task, we use Resnet50 (He et al., 2016) pretrained on the ImageNet (Deng et al., 2009) dataset to extract the `TS Aligner` inputs. For the multimodal task, we use the BERT-base model as the student

model with only the movie plot summaries as the input and trained it for 20 epochs with a learning rate of $2e-5$. For generative evaluations, we fine-tune the LLaMA-7B model (Touvron et al., 2023) with LoRA adapter (Hu et al., 2021) rank=8, respectively on Commonsense8K and Math10K datasets curated by Hu et al. (2023) and evaluated on the downstream tasks zero-shot. The model is trained for 3 epochs with a learning rate of $3e-4$ and batch size of 4.

One Tesla V100 and A100 GPUs were used for conducting the experiments. Each training step for the student and `TS Aligner` model takes 0.16 and 0.02 seconds, respectively, for a batch size of 32.

Additional implementation details.

(i) Orthogonality of the projection layers. The projection mappings $P_{ts} : \mathbb{R}^h \to \mathbb{R}^{768}$ and $P_s : \mathbb{R}^d \to \mathbb{R}^{768}$ (where h is the `TS Aligner` hidden dimension and d is the student hidden dimension) are parameterised with orthogonality constraints via PyTorch's `torch.nn.utils.parametrizations.orthogonal` module, using the Householder method. Under this reparameterisation, the constraint $P^\top P = I$ is satisfied exactly at every training step rather than being enforced through an additional soft regularisation term, so gradient updates automatically remain on the Stiefel manifold and no explicit projection or renormalisation step is required.

(ii) Diffusion latent extraction. For the text-to-image teachers (Stable Diffusion v1-4 and Midjourney), latents are obtained using the standard Stable Diffusion pipeline with a DDIM sampler, $T = 20$ denoising steps, and classifier-free guidance scale 7.5. For each text input, we take the U-Net bottleneck feature at the final denoising step ($t = 0$) as the cross-modal representation; this yields a 32×32 spatial map that is flattened to a 1024-dimensional vector. To amortise cost across the entire training run, teacher representations for every unique training input are extracted once, in fp16, and cached to disk; during distillation the cached tensors are loaded rather than re-generated, so the teacher pipeline is invoked exactly once per training example. A similar caching strategy is used for the ModelScope text-to-video and SpeechT5 text-to-audio teachers.

(iii) Gradient flow. The teacher F_t is frozen throughout training and its parameters are excluded from the optimiser, so no gradient signal propagates back through the teacher by construction. On the aligner side, gradients from the manifold-alignment loss $\mathcal{L}_{manifold}$ flow through the aligner projection P_{ts} and the aligner encoder $\tilde{F}_{ts}$; gradients from the auxiliary head $O^{\mathcal{T}}_{aux_{ts}}$ additionally update $\tilde{F}_{ts}$ and the auxiliary head parameters; and gradients from the aligner output loss $\mathcal{L}^{(1)}_{ts}$ update the aligner output head $O^{\mathcal{T}}_{ts}$ together with $\tilde{F}_{ts}$. On the student side, gradients from the manifold-alignment loss flow through P_s and the student encoder F_s, and gradients from the student output loss $\mathcal{L}^{(1)}_{s}$ update $O^{\mathcal{T}}_{s}$ and F_s. The aligner F_{ts} and the student F_s are optimised jointly at every step; the aligner is not pre-trained or updated in a separate phase, which is what makes the setup on-policy.

(iv) Token- vs. sample-level matching. The student's token-level output $h^X_s \in \mathbb{R}^{n\times d}$ (where n is the number of non-padded tokens in the input) is mean-pooled across the token dimension into a single d-dimensional vector, with padding tokens excluded from the mean via the attention mask. This pooled vector is what P_s projects into the shared 768-dimensional manifold. The manifold-alignment loss is therefore applied at the sample level (one vector per text input), which matches the granularity at which the teacher representation is produced (also one vector per input).

(v) Video-frame selection. The text-to-video teacher (ModelScope) generates 64 frames per text input. For each input we retain the hidden representation of only the first frame, which we flatten into a 262,144-dimensional vector for the aligner. Two considerations motivate the first-frame choice: (a) computational and storage efficiency — keeping all 64 frames would multiply the per-input storage cost by a factor of 64; (b) representational consistency — using a single vector per input matches the granularity of the text-to-image and text-to-audio teachers (which also produce one vector per input) and keeps the aligner architecture identical across teacher modalities. In preliminary experiments where we replaced the first-frame representation with the mean over all 64 frames, downstream distillation performance was within noise of the first-frame variant while extraction and caching cost rose substantially, so the first-frame convention was adopted throughout.

Models	Distillation type	MNLI Acc	QNLI Acc	QQP Acc	SST-2 Acc
BERT-6L	undistilled	77.7±0.0	84.9±0.1	89.0±0.0	87.6±0.1
	ARMADA - $\mathcal{L}_{cosine}$	**80.2±0.0**	**87.3±0.0**	88.6±0.1	**90.7±0.0**
	ARMADA - $\mathcal{L}_{euclid}$	**80.3±0.0**	**87.2±0.1**	88.9±0.0	**90.6±0.0**
	ARMADA - $\mathcal{L}_{elementwise}$	**80.1±0.0**	**87.3±0.0**	**89.1±0.1**	**90.7±0.0**
BERT-base	undistilled	79.3±0.0	87.8±0.1	83.1±0.0	89.2±0.1
	ARMADA - $\mathcal{L}_{cosine}$	**83.4±0.0**	**91.0±0.0**	**89.4±0.1**	**93.1±0.0**
	ARMADA - $\mathcal{L}_{euclid}$	**83.5±0.0**	**91.0±0.0**	**88.8±0.1**	**92.0±0.0**
	ARMADA - $\mathcal{L}_{elementwise}$	**83.4±0.0**	**91.2±0.0**	**88.9±0.1**	**92.8±0.0**

Table 8: Results of BERT models (undistilled and distilled by ARMADA with Stable Diffusion teacher) on ***GLUE-large*** tasks (mean over three seeds; loss objective stated per row). **Blue** indicates cases where ARMADA improves performance of the undistilled model. The improvement over the undistilled model is statistically significant (one-sided paired t-test, $t = 5.11$, $p < 0.01$).

Models	Distillation type	CoLA Mcc	MRPC Acc	RTE Acc	STS-B Pear	CB Acc	COPA Acc	BoolQ Acc	WiC Acc
BERT-6L	undistilled	42.8±0.1	78.6±0.0	64.1±0.2	88.5±0.4	75.0±0.0	53.0±0.0	71.3±0.1	55.7±0.2
	ARMADA - $\mathcal{L}_{cosine}$	**44.0±0.0**	**80.1±0.0**	**65.2±0.1**	87.9±0.0	73.1±0.1	49.9±0.0	**71.8±0.0**	**56.6±0.0**
	ARMADA - $\mathcal{L}_{euclid}$	**42.9±0.0**	**79.1±0.1**	**66.3±0.0**	88.3±0.0	**75.0±0.0**	**62.9±0.1**	**72.4±0.0**	**58.2±0.0**
	ARMADA - $\mathcal{L}_{elementwise}$	**46.9±0.0**	**79.3±0.0**	**67.0±0.1**	88.3±0.0	**78.4±0.1**	51.9±0.0	**73.3±0.0**	55.9±0.0
BERT-base	undistilled	60.8±0.1	83.8±0.1	71.0±0.0	89.3±0.0	83.7±0.1	62.9±0.0	75.1±0.1	58.4±0.1
	ARMADA - $\mathcal{L}_{cosine}$	58.8±0.0	**84.9±0.0**	69.9±0.1	88.6±0.0	**85.6±0.1**	**64.9±0.0**	73.3±0.0	56.2±0.0
	ARMADA - $\mathcal{L}_{euclid}$	60.1±0.0	**84.1±0.1**	68.8±0.0	88.3±0.0	**87.4±0.1**	62.9±0.0	72.3±0.0	58.2±0.0
	ARMADA - $\mathcal{L}_{elementwise}$	**61.1±0.0**	**85.0±0.0**	68.8±0.1	88.3±0.0	**83.8±0.1**	60.9±0.0	73.0±0.0	**58.7±0.0**
DeBERTa-v2-xxlarge	undistilled	72.1±0.0	89.7±0.1	90.3±0.0	92.3±0.0	81.9±0.1	83.7±0.0	87.2±0.0	60.8±0.1
	ARMADA - $\mathcal{L}_{cosine}$	71.4±0.0	**90.3±0.0**	**90.9±0.1**	91.9±0.0	**94.5±0.1**	81.9±0.0	87.3±0.0	57.6±0.0
	ARMADA - $\mathcal{L}_{euclid}$	68.2±0.0	**91.1±0.1**	**90.8±0.0**	91.8±0.0	93.4±0.1	79.9±0.0	**87.9±0.0**	60.0±0.0
	ARMADA - $\mathcal{L}_{elementwise}$	70.4±0.0	**90.5±0.0**	**90.3±0.1**	92.1±0.0	**92.8±0.1**	81.9±0.0	**88.1±0.0**	59.6±0.0
OPT-1.3B	undistilled	59.6±0.0	84.3±0.1	76.3±0.0	89.7±0.0	78.2±0.1	54.8±0.0	74.6±0.0	57.9±0.1
	ARMADA - $\mathcal{L}_{cosine}$	**60.4±0.0**	83.4±0.0	74.2±0.1	**90.0±0.1**	**80.2±0.0**	54.9±0.0	**75.3±0.0**	55.9±0.0
	ARMADA - $\mathcal{L}_{euclid}$	**61.2±0.0**	82.4±0.1	**77.1±0.0**	89.9±0.0	**78.5±0.1**	**58.9±0.0**	**75.6±0.0**	57.0±0.0
	ARMADA - $\mathcal{L}_{elementwise}$	59.5±0.0	80.5±0.1	**77.1±0.0**	**90.1±0.1**	76.6±0.0	**58.9±0.0**	**76.0±0.0**	57.0±0.0

Table 9: Results of different LMs (undistilled and distilled by ARMADA) on GLUE-small and SuperGLUE tasks with the Midjourney (Borji, 2022) teacher model (loss objective stated per row). ARMADA improves BERT-base and BERT-6L by 0.44% and 3.2% on average, respectively, mirroring the Stable Diffusion results (Table 1) and confirming the effect is not specific to a single teacher.

E Additional Experimental Results

This appendix collects the secondary experimental results referenced in Section 5, supplementing the headline NLU/reasoning and instruction-tuning results, the baseline comparison, the capacity-matched control, and the mechanism analysis that are retained in the main text.

E.1 Results with a Midjourney Teacher

Table 9 reports the same setup with a Midjourney (Borji, 2022) teacher (Stable Diffusion with a LoRA (Hu et al., 2021) adapter trained on 100K+ Midjourney images). The TS Aligner aligner remains far below the student in isolation (Table 10), yet ARMADA still improves BERT-base and BERT-6L by 0.44% and 3.2% on average, confirming that the gains are not specific to the Stable Diffusion teacher.

E.2 Auxiliary Output Performance

We report the performance obtained with the auxiliary heads in Table 12. The results reveal a striking Spearman rank correlation between output head and auxiliary head performances of 0.99 for student models and 0.69 for teacher models. This correlation indicates a strong alignment in performance trends despite

Teacher Model	CoLA	MRPC	RTE	STS-B	CB	COPA	BoolQ	WiC
	Mcc	Acc	Acc	Pear	Acc	Acc	Acc	Acc
Stable Diffusion	7.8	66.6	53.4	3.7	55.4	62.0	60.9	53.0
Midjourney	1.8	66.5	52.3	1.9	50.0	56.0	56.6	50.6

Table 10: Standalone task performance of the (shallow, ∼0.8M-parameter) `TS Aligner` on all GLUE-small and SuperGLUE tasks under two teacher models. As noted by reviewers, the aligner performs far below the undistilled student (e.g. CoLA Mcc 7.8 vs. ∼59 for the student), confirming it does not solve the tasks itself. Despite this, the student distilled against the trained aligner improves significantly (Tables 1, 18; paired t-tests, $p < 0.01$ on most tasks), indicating that the value of `TS Aligner` lies in the supervisory trajectory it provides rather than its standalone accuracy.

Methods	SST-2	QNLI	QQP	MNLI	Avg.
ViLBERT	90.3	89.6	88.4	82.4	87.7
VL-BERT	90.1	89.5	88.6	82.9	87.8
VisualBERT	90.3	88.9	88.4	82.4	87.5
Img-Voken	92.2	88.6	88.6	82.6	88.0
Visual Guidance	89.7	88.2	84.2	79.5	85.4
Vid-Voken w/ResNet	93.4	89.2	88.7	83.0	88.6
Vid-Voken w/CLIP	**94.1**	89.8	89.0	83.9	89.2
X-adapter	92.7	**91.4**	89.3	**84.4**	**89.4**
`ARMADA`	93.1	91.3	**89.5**	83.5	**89.4**

Table 11: Improvement for BERT-base model on GLUE-large tasks distilled by `ARMADA` with Stable Diffusion teacher model, compared against the improvements reported by multimodal KD methods. `ARMADA` is competitive with the multimodal baselines while using $< 0.8\%$ of their teacher training cost; the improvement over the unimodal and multimodal KD baselines is statistically significant (one-sided t-test, $t = 1.56$, $p < 0.1$).

the auxiliary output head's generally lower performance, attributable to its operation over presumed lower-dimensional manifold projections. On average, student models exhibit a modest improvement of 0.61% with the output head over the auxiliary head. `TS Aligner` benefits even more significantly, with an improvement margin of 6.07%, underscoring the importance of auxiliary outputs in `ARMADA` framework.

Models	Loss type	CoLA	MRPC	RTE	STS-B	CB	COPA	BoolQ	WiC
		Mcc	Acc	Acc	Pear	Acc	Acc	Acc	Acc
`TS Aligner`	-	0.7	66.4	50.2	1.3	48.9	41.0	45.4	48.9
BERT-base	cosine	59.0	83.4	70.4	88.6	87.5	64.0	74.7	57.1
	euclid	60.1	84.1	67.5	88.2	85.7	63.0	73.9	55.2
	elementwise	58.5	82.3	65.7	88.6	85.7	65.0	74.4	57.2

Table 12: Auxiliary-head output results for `TS Aligner` and the BERT-base student under the three $\mathcal{L}_{manifold}$ objectives ($\mathcal{L}_{cosine}$, $\mathcal{L}_{euclid}$, $\mathcal{L}_{elementwise}$) on GLUE-small and SuperGLUE tasks. The auxiliary-head and output-head performances are strongly rank-correlated (Spearman $\rho = 0.99$ for the student, $\rho = 0.69$ for `TS Aligner`), and the output head exceeds the auxiliary head by 0.61% (student) and 6.07% (`TS Aligner`) on average.

E.3 Hyperparameter and Loss-function Ablations (BERT)

Our in-depth analysis of the BERT-base model's performance across various GLUE-small and SuperGLUE NLU tasks under different hyperparameter settings reveals insights into optimizing cross-modal distillation. As illustrated in Figure 2, our exploration of the effects of the hyperparameters α, β, and γ reveals fine-grained insights into their impact on cross-modal distillation efficacy. Notably, setting α to 0, thereby excluding the

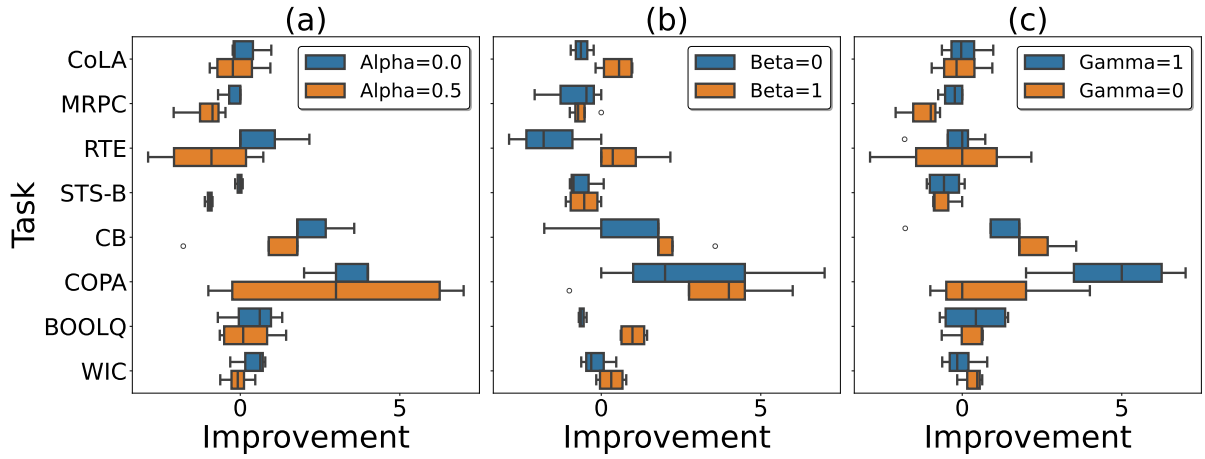


Figure 2: Improvement of ARMADA-distilled BERT-base over the undistilled baseline as a function of (A) the logit-matching weight α, (B) the manifold-alignment weight β, and (C) the auxiliary-output weight γ. Stable Diffusion teacher.

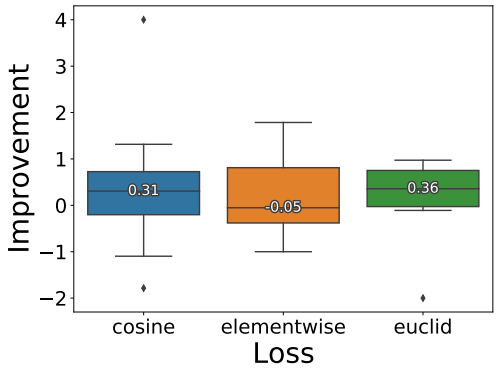


Figure 3: Improvement of ARMADA-distilled BERT-base over the undistilled baseline under each of the three $\mathcal{L}_{manifold}$ objectives.

logit-matching soft loss from the final training loss, yields a consistent 0.6% improvement across all tasks compared to $alpha = 0.5$, which gives equal importance to the output loss and the soft loss. This finding suggests that prioritizing the output loss can lead to better distillation outcomes. The significance of β, which regulates the weight of the manifold projection loss, is underscored by its performance enhancement in seven out of eight tasks when set to 1, indicating a positive association between higher β values and improved model performance. Conversely, γ, which determines the weight assigned to the auxiliary output loss, demonstrates a variable influence on the effectiveness of cross-modal distillation. A γ value of 1 enhances performance by 0.85%, significantly higher than $\gamma = 0$ (improvement 0.36%), highlighting the role of auxiliary losses in the distillation process.

Similar ablation analyses with different manifold projection losses in Figure 3 suggest that all three manifold-alignment loss objectives effectively distil knowledge from the multimodal teacher model to language-only student models, with $\mathcal{L}_{euclid}$ being most effective and robust on average (median improvement of 0.36 as opposed to 0.31 of $\mathcal{L}_{cosine}$ and -0.05 of $\mathcal{L}_{elementwise}$). As explained in Section 3, all the loss objectives enforce different regularizations for the student. We hypothesize that by balancing regularization and the task objective, $\mathcal{L}_{euclid}$ balances the other two losses, leading to the largest performance gain.

E.4 Results on Multimodal Tasks

Models	Distillation type	MM-IMDb F1 Micro/Macro	Meme Acc
BERT-12L	undistilled	64.9/48.1	55.4
	ARMADA - $\mathcal{L}_{cosine}$	**65.7/ 49.3**	**60.2**
	ARMADA - $\mathcal{L}_{euclid}$	**66.2/ 48.9**	**60.4**
	ARMADA - $\mathcal{L}_{elementwise}$	**65.3/ 48.0**	**59.4**

Table 13: Results of the BERT-12L model with and without cross-modal distillation on the MM-IMDb genre and Hateful Memes classification tasks (loss objective stated per row). ARMADA improves micro/macro-F1 on MM-IMDb by +1.3/ + 1.2 pp and Hateful Memes accuracy by up to +5.0 pp (55.4 → 60.4). The student remains text-only at inference, so the gains stem from transferred cross-modal structure rather than architectural multimodality.

We highlight the performance obtained by BERT-12L on the MM-IMDb classification task in Table 13. With cross-modal distillation, the BERT-12L model achieves +1.2% and +1.3% improvements in macro-F1 and micro-F1, respectively, over the undistilled baseline. Notably, this gain is comparable to the improvement reported by Kiela et al. (2019) using a fully multimodal BERT architecture. Importantly, in our setting, the teacher and student share no common modality at inference time, demonstrating that ARMADA effectively transfers multimodal priors into a unimodal language model without architectural modification.

To further validate the generality of this effect, we additionally evaluate on the Hateful Memes benchmark (Kiela et al., 2020), a more challenging multimodal dataset designed to prevent unimodal shortcuts. The undistilled BERT-12L achieves 55.4% accuracy, whereas ARMADA improves performance consistently across alignment variants to 60.2–60.4% accuracy, corresponding to an absolute gain of approximately 5%. Unlike MM-IMDb, where improvements are modest but consistent, Hateful Memes shows a substantially larger gain, suggesting that cross-modal distillation is particularly beneficial for tasks that require joint reasoning over visual and textual semantics.

Crucially, in both experiments, the student remains a text-only model at inference time, underscoring that the observed gains arise from transferred cross-modal semantic structure rather than architectural multimodality.

E.5 Instruction-tuning: Ablations and Larger Student

Figure 4 reports the effect of the manifold-alignment weight β and the auxiliary-output weight γ. Except on Dolly, both hyperparameters contribute an additional $\sim$0.5% on the instruction-tuning tasks, and both are statistically significant ($p < 0.01$). Figure 5 reports the effect of the manifold-alignment loss. $\mathcal{L}_{\text{cosine}}$ yields the highest average improvement (0.9%), followed by $\mathcal{L}_{\text{elementwise}}$ and $\mathcal{L}_{\text{euclid}}$ (0.5% each); the three losses are nonetheless statistically equivalent ($p > 0.05$).

Teacher	Distillation Type	Dolly	SNI	SelfInst	UNI	Vicuna
-	undistilled	35.34	28.19	18.01	32.78	17.85
text-to-image	ARMADA with $\mathcal{L}_{cosine}$	33.67	**28.77**	**19.36**	32.59	**18.89**
	ARMADA with $\mathcal{L}_{euclid}$	33.71	28.11	**19.14**	**34.53**	18.72
	ARMADA with $\mathcal{L}_{elementwise}$	34.63	**29.06**	**19.80**	32.54	**18.16**
text-to-video	ARMADA with $\mathcal{L}_{cosine}$	33.53	**29.28**	**18.57**	**33.80**	**18.82**
	ARMADA with $\mathcal{L}_{euclid}$	32.02	28.07	**21.31**	32.41	**18.94**
	ARMADA with $\mathcal{L}_{elementwise}$	30.44	27.56	**18.97**	32.76	**18.36**
text-to-audio	ARMADA with $\mathcal{L}_{cosine}$	33.14	**29.63**	**20.08**	**34.35**	**19.96**
	ARMADA with $\mathcal{L}_{euclid}$	34.78	27.26	17.47	32.84	**19.51**
	ARMADA with $\mathcal{L}_{elementwise}$	34.67	**29.13**	**18.73**	**34.22**	**18.18**

Table 14: Results on instruction-tuning tasks (RougeL) with LLaMA-3.1-8B under different multimodal teacher models and the three loss objectives. ARMADA improves over the undistilled LLaMA-8B on four of five tasks (up to +2% on SelfInst and Vicuna); the exception is Dolly, where the strong undistilled baseline leaves little headroom. A two-way ANOVA again finds no significant difference between teacher modalities ($p > 0.05$).

Table 14 further highlights the performance of ARMADA with a larger student model, LLaMA-8B. Except for Dolly evaluation, in the other four tasks, ARMADA turns effective, even for a superior baseline model

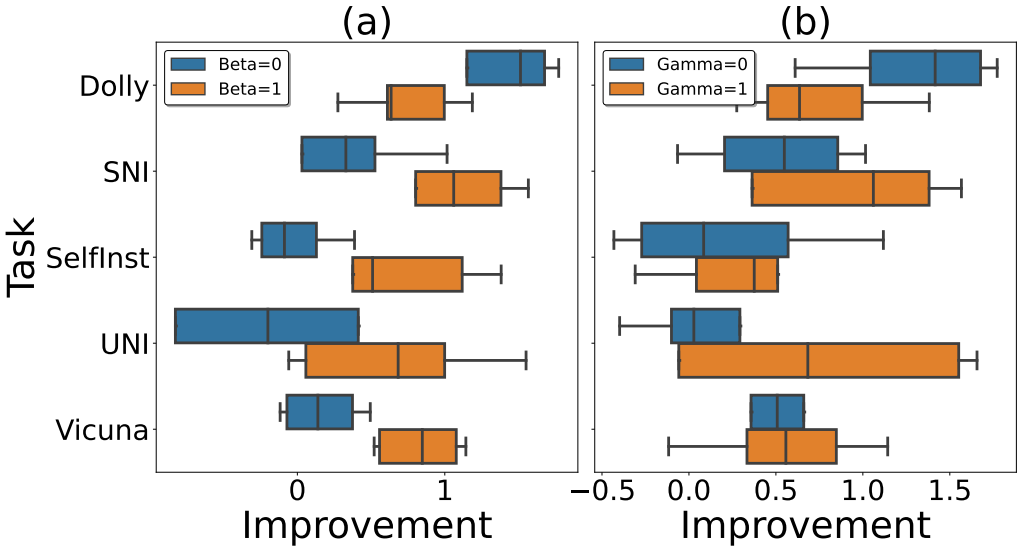


Figure 4: Improvement of ARMADA-distilled LLaMA-3B over the undistilled baseline as a function of the manifold weight β and the auxiliary weight γ. Both are statistically significant ($p = 7 \times 10^{-5}$ for β and 5×10^{-3} for γ).

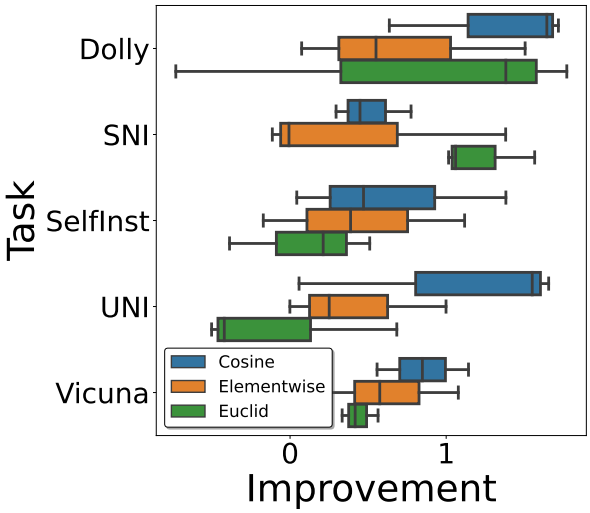


Figure 5: Improvement of ARMADA-distilled LLaMA-3B over the undistilled baseline under each $\mathcal{L}_{manifold}$ objective. The three losses are statistically equivalent ($p = 0.52$).

LLaMA-8B. Strikingly, on tasks like SelfInst and Vicuna, ARMADA improves the base model's performance by over 2%. A possible reason behind the muted performance in Dolly evaluation could be the narrow prompt diversity and template simplicity, where the undistilled LLaMA-8B model already shows strong performance, reducing the marginal benefit of distilling cross-modal knowledge. In terms of the different teacher modalities, we observe the text-to-video model to be most effective, with the highest performance improvement of 3.3%, followed by the text-to-audio (2.1%) and text-to-image (1.8%) teacher models. The teacher modalities are nonetheless statistically equivalent ($p > 0.05$).

F Mechanism Analysis: Extended Results

This appendix expands the mechanism analysis of Section 6, evaluating ARMADA against five negative controls (Randomly Frozen, Random, Shuffled-Input, Frozen, and Constant teacher) and the capacity-matched control on seven NLU tasks (CoLA, MRPC, RTE, CB, COPA, WiC, BoolQ) across three seeds and three manifold losses. The aggregate panel (Table 6 in the main text) uses a one-sided paired Wilcoxon signed-rank test. Following reviewer suggestion, we additionally report a prediction-level McNemar analysis as a fine-grained complement.

		ARMADA vs. Undistilled			ARMADA vs. Shuffled Input		
Task	n	b	c	p	b	c	p
RTE	831	78	61	0.175	**87**	64	**0.073**†
MRPC	5175	233	212	0.343	286	254	0.182
CB	168	9	4	0.267	10	5	0.302
BoolQ	9810	**1056**	920	**0.002****	964	991	0.557
CoLA	3129	131	122	0.615	119	146	0.110
COPA	300	46	47	1.000	58	58	0.926
WiC	1914	150	160	0.609	133	134	1.000

Table 15: Per-task prediction-level McNemar tests (best loss per side; predictions pooled across 3 seeds). b = examples ARMADA corrects that the comparator misses; c = the reverse. Against the undistilled baseline, ARMADA corrects significantly more predictions on **BoolQ** ($p < 0.01$) and is net-positive on most tasks. Against the strict Shuffled-Input control — which retains full task supervision and is therefore a conservative null — ARMADA's clearest prediction-level advantage is on the grounded-inference task **RTE** ($p < 0.1$), with the GLUE-small subset (RTE, MRPC, CoLA) favouring ARMADA overall (b=492 vs c=464). $^{**}p < 0.01$, $^{\dagger}p < 0.1$.

Prediction-level McNemar tests. Table 15 reports per-task McNemar tests over predictions pooled across the three seeds, separately against the undistilled baseline and the Shuffled-Input control. Counts b and c are the example-level corrections in each direction (best loss per side). Consistent with the aggregate Wilcoxon picture, ARMADA is example-level favourable on the GLUE-small subset against the Shuffled-Input control and on the tasks that most plausibly rely on grounded semantic abstractions; the per-task significance pattern across the full panel is mixed, in line with the modest aggregate separation noted in the main text.

Model	**CoLA** Mcc	**MRPC** Acc	**RTE** Acc	**STS-B** Pear	**CB** Acc	**COPA** Acc	**BoolQ** Acc	**WiC** Acc
TS Aligner	4.9	66.5	54.9	2.5	41.1	54.0	55.0	53.0
ARMADA with $\mathcal{L}_{cosine}$	57.9	84.4	71.5	90.0	81.1	63.0	75.8	56.4
ARMADA with $\mathcal{L}_{euclid}$	59.4	84.5	70.8	89.5	82.1	63.0	75.5	57.0
ARMADA with $\mathcal{L}_{elementwise}$	60.4	84.8	70.0	89.8	83.9	63.0	76.7	57.4

Table 16: TS Aligner and BERT-12L with ARMADA on GLUE-small and SuperGLUE under shuffled teacher inputs (Stable Diffusion embeddings randomly permuted across training samples). Under shuffling, the TS Aligner's standalone accuracy shows a measurable average drop, whereas the student's accuracy is comparatively little affected on aggregate — a pattern consistent with the Shuffled-Input control retaining full task supervision at the student's output layer.

Shuffled-input accuracies. Table 16 reports the raw accuracies of TS Aligner and the BERT-12L student under shuffled teacher inputs. TS Aligner accuracy drops by 3.8% and the student by 1.8% on average, with up to a 5% drop on the most semantically demanding tasks; breaking the text–image correspondence thus measurably harms the student, indicating that the cross-modal signal is used rather than ignored.

G Extended Discussion: Representation-level Analysis

This appendix collects the representation-level evidence summarized in Section 6.

Semantic cohesion and cluster purity.

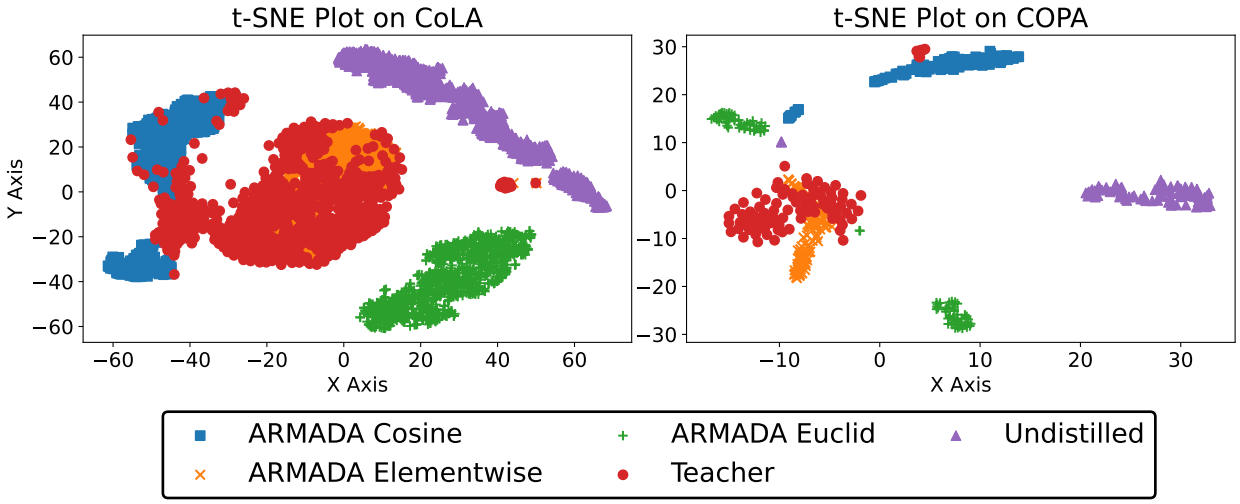
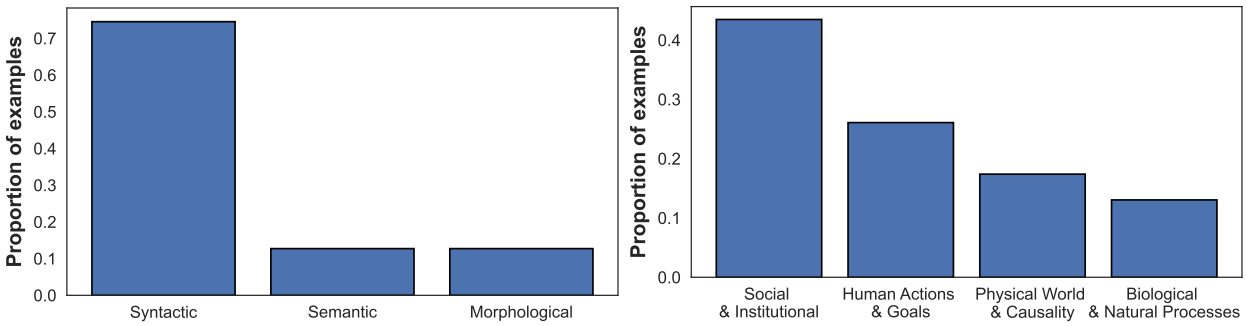


Figure 6: t-SNE plots of embeddings obtained from different models on CoLA and COPA tasks. The student under $\mathcal{L}_{elementwise}$ projects data points closest to the teacher manifold, whereas the undistilled student projects them farthest.

Table 17 reports Silhouette scores for ARMADA and the undistilled BERT-base across the two scenarios (ARMADA wins vs. loses). Where ARMADA wins, its Silhouette score improves by 76% over the undistilled model; where it loses, the score drops by 20%. We additionally compute a cluster-purity score $\text{Purity} = \frac{1}{N}\sum_{k=1}^{K}\max_{y\in\mathcal{Y}}|\{x_i \mid c_i = k \wedge y_i = y\}|$ over BERT-12L hidden states (KMeans, $K = 2$). The undistilled model obtains 0.77/0.56 on CoLA/COPA, while ARMADA obtains 0.81/0.68, indicating more topologically organized, label-aligned representations.

Methods	**CoLA**		**COPA**	
	Scen. 1	Scen. 2	Scen. 1	Scen. 2
Undistilled	0.31	0.40	0.31	**0.59**
ARMADA	**0.52**	**0.48**	**0.57**	0.38

Table 17: Silhouette scores (higher the better) for different scenarios with undistilled and ARMADA models.



(a) CoLA (b) COPA

Figure 7: Categorization of test examples that ARMADA-distilled BERT-12L predicts correctly and the undistilled model gets wrong.

Where do the gains concentrate? We manually categorize the test instances corrected by ARMADA on CoLA and COPA (Figure 7). For CoLA, over 70% of corrected cases are *syntactic* (filler–gap dependencies, long-distance constraints), with the remainder split between semantic and morphological violations. For COPA, ∼42% are *Social & Institutional*, followed by *Human Actions* (26%), *Physical World* (17%), and *Biological / Natural Processes* (15%). This non-uniformity supports the interpretation that ARMADA transfers

structured semantic and conceptual priors rather than acting as a uniform regularizer; we note, however, that this analysis is descriptive rather than statistically tested.

Loss type	CoLA	MRPC	RTE	STS-B	CB	COPA	BoolQ	WiC
Cosine	**6.75 (0.00)**	**6.95 (0.00)**	**3.33 (0.00)**	**54.63 (0.00)**	**3.89 (0.00)**	0.49 (0.62)	-0.29 (0.78)	**12.91 (0.00)**
Euclid	**6.54 (0.00)**	**8.40 (0.00)**	**3.28 (0.00)**	**51.33 (0.00)**	**3.20 (0.00)**	1.47 (0.14)	-3.20 (0.00)	**16.29 (0.00)**
Elementwise	**6.42 (0.00)**	**8.03 (0.00)**	**3.16 (0.00)**	**51.51 (0.00)**	**4.48 (0.00)**	**5.85 (0.00)**	-2.45 (0.01)	**15.81 (0.00)**

Table 18: T-statistic (p-value) between student training loss distributions without and with `TS Aligner` training. A positive t-statistics (highlighted in bold) indicates that the expected student loss without a trained `TS Aligner` is higher than the student loss with a jointly trained `TS Aligner` model, indicating the importance of joint aligner-student training.

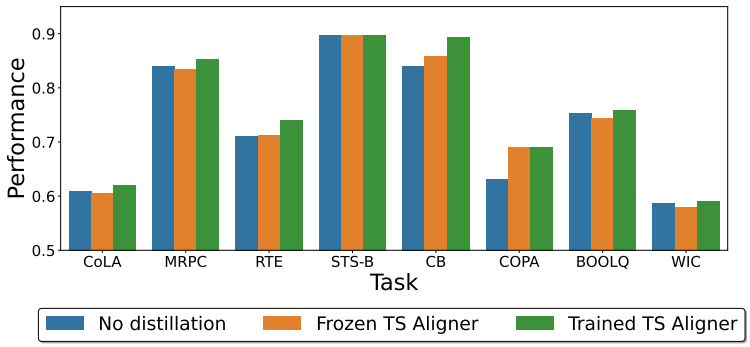


Figure 8: BERT-base performance under two training curricula: distilled against a frozen, untrained `TS Aligner` vs. the standard jointly-trained `TS Aligner`.

Impact of the trained `TS Aligner`. Figure 8 shows that in six of eight NLU tasks the student distilled with a frozen (randomly-initialized, untrained) `TS Aligner` is inferior to the undistilled student, whereas the standard trained-`TS Aligner` curriculum improves it. Table 18 reports the t-statistic between the student output-loss distributions with and without `TS Aligner` training; a strong positive t-statistic with low p-value confirms the positive impact of the `TS Aligner` learning curriculum. As noted in the main text, the `TS Aligner` itself performs poorly in isolation (Table 10); its value lies in the supervisory *trajectory* it provides, not its standalone task accuracy.