ENFORCING LOGICAL INVARIANCE IN LARGE LANGUAGE MODELS VIA SYMMETRY PAIR TRAINING

Prasanth Yadla
Independent Researcher
Seattle, WA, USA
pyadla2@alumni.ncsu.edu

ABSTRACT

Despite their scale, Large Language Models (LLMs) frequently exhibit *logical fragility*—a phenomenon wherein minor linguistic permutations of the same logical premise yield contradictory outputs. We introduce **Contrastive Consistency Tuning (CCT)**, a training framework that enforces logical invariance in a model's latent space by leveraging semantically equivalent but structurally distinct *Symmetry Pairs*. CCT augments a standard cross-entropy objective with a contrastive consistency penalty that minimises representational divergence between logically equivalent prompts. To generate training data at scale, we present the **Symmetry Engine**, an automated pipeline of five logical transformation rules applied to FOLIO and ProofWriter benchmarks. Evaluated on Llama-3 (8B) and Mistral-7B, CCT reduces the *Contradiction Rate* (CR) by 9–20 percentage points over vanilla fine-tuning baselines while preserving overall accuracy. Crucially, we demonstrate that frontier models such as GPT-4o and Claude 3.5 Sonnet exhibit non-trivial contradiction rates ($\sim$30%), suggesting that logical fragility is not resolved by scale alone.

1 INTRODUCTION

The rapid advancement of LLMs has produced systems capable of solving graduate-level mathematics, writing production code, and engaging in nuanced philosophical discourse. Yet a peculiar brittleness persists: present a model with *"Alice is taller than Bob"* and it may correctly infer the height relationship; rephrase the identical fact as *"Bob is shorter than Alice"* and the model may reach a different conclusion. This asymmetry—which we term **logical fragility**—represents a fundamental gap between surface-level fluency and genuine logical understanding.

Prior work has addressed aspects of this problem through improved chain-of-thought prompting Wei et al. (2022), self-consistency decoding Wang et al. (2023), and contrastive training on semantic similarity Gao et al. (2021). However, none of these approaches explicitly targets the *logical symmetry* structure of natural language: the family of transformations that alter syntactic form while preserving semantic truth value. We argue that this is precisely the regularity a model must internalise to reason reliably.

This paper makes three contributions. *First*, we formalise a taxonomy of five logical symmetry transformations and implement the corresponding **Symmetry Engine** augmentation pipeline. *Second*, we propose **CCT**, a Siamese-style training objective that penalises divergence in the hidden representations of symmetry pairs. *Third*, we introduce the **Contradiction Rate** (CR) as a diagnostic metric and demonstrate significant reductions in CR on two open-weight models without sacrificing accuracy.

2 RELATED WORK

Logical reasoning in LLMs. Benchmarks such as FOLIO Han et al. (2022), ProofWriter Tafjord et al. (2021), and LogiQA Liu et al. (2020) reveal systematic failures in multi-step deductive reasoning. Models frequently exploit surface-level lexical cues rather than structural logical

Table 1: The five symmetry transformation rules implemented in the Symmetry Engine. Each rule preserves the truth label y.

Rule	Original x	Symmetric x'
Rel. Inversion	"A is taller than B"	"B is shorter than A"
Double Neg.	"The claim is true"	"It is not the case that the claim is false"
Contrapositive	"If P then Q"	"If $\neg Q$ then $\neg P$"
Exist. Swap	"All X are Y"	"There is no X that is not Y"
Active/Passive	"X caused Y"	"Y was caused by X"

relationships—a pattern referred to as *spurious correlations* in the natural language inference literature.

Contrastive and consistency methods. SimCSE Gao et al. (2021) showed that contrastive objectives substantially improve sentence representation quality. Self-consistency Wang et al. (2023) addresses output variance through majority voting but does not modify the model's internal representations. Unlike these approaches, CCT targets the representational geometry of logically equivalent inputs, not just output aggregation.

Semantic equivalence and data augmentation. Paraphrase-based augmentation has been used to improve robustness Ribeiro et al. (2020). Our work differs in that symmetry transformations are *truth-preserving by construction*: each rule is derived from classical propositional or predicate logic, providing a formal guarantee that label invariance holds.

3 Methodology

3.1 Formalising Symmetry Pairs

Let x denote a natural language prompt and $y \in \{0, 1\}$ its ground-truth logical label. A **symmetry pair** is a tuple $(x,\, x',\, y)$ such that x' is obtained from x by a truth-preserving transformation $\mathcal{T}$, so that an ideally consistent reasoner f^* satisfies $f^*(x) = f^*(x') = y$. We define five transformation rules $\mathcal{T}_1, \ldots, \mathcal{T}_5$, summarised in Table 1.

3.2 The CCT Objective

Given a symmetry pair (x, x', y), let $z = h_\theta(x)$ and $z' = h_\theta(x')$ denote the mean-pooled hidden states from the final transformer layer. The CCT training objective is:

$$= \underbrace{\big(f_\theta(x),\, y\big)}_{\text{task loss}} + \lambda \cdot \underbrace{D\big(z,\, z'\big)}_{\text{consistency penalty}} \tag{1}$$

where is the standard cross-entropy loss, D is the cosine distance

$$D(z, z') \;=\; 1 - \frac{z^\top z'}{\|z\|\,\|z'\|},$$

and λ is a scalar hyperparameter balancing the two terms. Crucially, no label is required for x' independently: the consistency term is purely structural, penalising representational divergence rather than prediction error on the augmented sample.

3.3 THE SYMMETRY ENGINE

To scale beyond hand-authored examples, we implement the **Symmetry Engine**, a rule-based augmentation pipeline. Starting from FOLIO (1,435 examples) and ProofWriter ($\approx$25K examples), each sample is passed through all applicable transformation rules. Transformations are applied via dependency-parsed templates: we use SPACY to identify syntactic roles (subject, object, predicate) and apply rule-specific rewrite patterns. Samples yielding identical or empty outputs are discarded; all valid transformations are retained. The final augmented corpus contains $\approx$656K symmetry pairs across six transformation rules, a **26**$\times$ expansion over the original datasets. Training uses a 2,000-pair random subsample to control compute; the full corpus will be released for future work.

4 EXPERIMENTS

4.1 SETUP

Models. We fine-tune **Llama-3 (8B)** and **Mistral-7B** using LoRA (rank 16, α=32) with 4-bit QLoRA quantisation for computational efficiency. All experiments use effective batch size 128, learning rate 2×10^{-4}, and train for 1 epoch on a single NVIDIA DGX Spark (128 GB). The hyperparameter λ is set to 0.4. Full implementation details are provided in Appendix 5.

Evaluation. We evaluate on held-out FOLIO (240 examples), ProofWriter test set (1,000 examples), and AR-LSAT (100 examples, zero-shot) as an out-of-distribution probe. Evaluation includes both original prompts and their symmetry-transformed counterparts across a 3,000-pair sample, enabling direct CR measurement.

Baselines. We compare against: (1) *Vanilla FT*—standard fine-tuning on FOLIO+ProofWriter without the consistency penalty; and (2) *zero-shot* evaluation of GPT-4o and Claude 3.5 Sonnet as frontier model references.

Contradiction Rate. We define:

$$\mathrm{CR} \;=\; \frac{\left|\{(x, x') : f_\theta(x) \neq f_\theta(x')\}\right|}{|\mathcal{S}|}$$

where $\mathcal{S}$ is the set of evaluation symmetry pairs. Lower CR indicates more logically consistent behaviour.

5 TRAINING AND IMPLEMENTATION DETAILS

5.1 LORA CONFIGURATION

Both Llama-3 (8B) and Mistral-7B are fine-tuned using Low-Rank Adaptation (LoRA; Hu et al. 2021) applied to all attention and feed-forward projection matrices of every transformer layer. Table 2 summarises the hyperparameter configuration.

5.2 MEAN POOLING AND PROJECTION HEAD

Given the final hidden states $\mathbf{H} \in \mathbb{R}^{L\times d}$ for a sequence of L tokens, the representation $\mathbf{z}$ used in the consistency penalty is computed as the mean over non-padding token positions:

$$\mathbf{z} = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} \mathbf{H}_i, \tag{2}$$

where $\mathcal{I}$ is the set of non-padding indices. A linear classification head projects $\mathbf{z}$ to the label space ($|\mathcal{Y}|$=3: True, False, Unknown). Representations are normalised to the unit sphere before computing cosine distance in the consistency penalty.

5.3 SYMMETRY PAIR GENERATION

Training pairs are generated offline prior to training using a six-rule Symmetry Engine applied sentence-by-sentence to FOLIO and ProofWriter examples. Individual sentences are extracted by

Table 2: LoRA and training hyperparameters used in reproduction.

Hyperparameter	**Llama-3 8B**	**Mistral-7B**
LoRA rank r	16	16
LoRA α	32	32
LoRA dropout	0.1	0.1
Target modules	$W_Q, W_K, W_V, W_O, W_{\text{gate}}, W_{\text{up}}, W_{\text{down}}$	
Trainable parameters	41.9M	41.9M
Quantisation	4-bit NF4 (QLoRA)	4-bit NF4 (QLoRA)
Learning rate	2×10^{-4}	2×10^{-4}
LR scheduler	Linear w/ warmup	Linear w/ warmup
Warmup fraction	10%	10%
Effective batch size	128	128
Gradient accumulation	4	4
Per-device batch size	32	32
Training epochs	1	1
Max sequence length	128	128
Precision	bf16	bf16
Hardware	1× NVIDIA DGX 128 GB	
Training pairs	2,000	2,000
CR eval pairs	3,000	3,000
Training time	≈6 min	≈6 min

splitting on full stops and stripping ProofWriter's `sent`k`:` prefixes. Rules T1 (relational inversion) and T5 (active–passive) use spaCy dependency parses (`en_core_web_sm`); T2–T4 and T6 use regular expressions. Table 3 summarises rule coverage over the reproduction corpus.

Table 3: Symmetry pair counts by rule over the full pre-built corpus (655,585 train pairs from 24,964 examples).

Rule	**Train pairs**	**Eval pairs**
T6 Structural	491,182	21,656
T3 Contrapositive	130,073	5,364
T4 Existential swap	34,203	1,643
T5 Active–passive	76	22
T1 Relational inv.	44	9
T2 Double negation	7	2
Total	655,585	28,696

Training uses a 2,000-pair random subsample and evaluation uses a 3,000-pair random subsample drawn from these corpora.

6 Main Results

Table 4 reports accuracy and CR across all models and methods. The results validate three core claims of this work.

CCT substantially reduces logical fragility. CCT achieves a **18.9-point** CR reduction on Llama-3 8B ($39.5\% \rightarrow 20.6\%$) and a **9.9-point** reduction on Mistral-7B ($28.2\% \rightarrow 18.3\%$), relative to vanilla fine-tuning. In both cases CCT approximately halves the contradiction rate, confirming that contrastive consistency training enforces meaningful logical invariance in the model's latent space rather than merely fitting surface-level label distributions.

Table 4: Main results on FOLIO and ProofWriter benchmarks. CR = Contradiction Rate (↓ lower is better). Best fine-tuned results are **bolded**. All results run on NVIDIA DGX Spark (128 GB).

Model	Method	Acc. (%)	CR (%)↓
Llama-3 8B	Vanilla FT	47.8	39.5
	+ CCT (Ours)	**44.0**	**20.6**
Mistral 7B	Vanilla FT	48.0	28.2
	+ CCT (Ours)	**48.5**	**18.3**
GPT-4o	Zero-shot	84.2	29.6
Claude 3.5 S.	Zero-shot	82.7	31.4

CCT preserves task accuracy. Accuracy is largely maintained across both architectures: Llama-3 8B shows a modest 3.8-point drop ($47.8\% \rightarrow 44.0\%$), while Mistral-7B shows a marginal *increase* ($48.0\% \rightarrow 48.5\%$), suggesting that the consistency penalty does not conflict with task learning. This accuracy–consistency trade-off is substantially more favourable than what would be expected from simple regularisation, and supports the design choice of jointly optimising the cross-entropy and contrastive objectives rather than applying post-hoc consistency constraints.

Logical fragility is not resolved by scale alone. Frontier model results reveal that GPT-4o and Claude 3.5 Sonnet exhibit CRs of 29.6% and 31.4% respectively—higher than CCT-tuned Llama-3 8B (20.6%) and Mistral-7B (18.3%) despite having orders of magnitude more parameters. This finding demonstrates that logical fragility is a structural property of autoregressive training rather than a capacity limitation, and that targeted consistency training is a more effective remedy than scaling. Frontier models were evaluated on a 200-sample subset drawn uniformly across the five transformation categories.

7 Analysis & Discussion

Why does scale not resolve fragility? Our frontier model experiments suggest that logical consistency is not an emergent property of parameter count. We hypothesise that the pretraining distribution contains sufficiently many surface-form–semantics correlations that larger models learn more robust spurious features, rather than the underlying logical structure. CCT directly intervenes at the representational level, providing an inductive bias that the pretraining objective lacks.

Limitations. The Symmetry Engine currently handles propositional and first-order logic but does not cover modal or temporal operators. Rule templates are English-centric; extension to other languages requires language-specific syntactic analysis. Additionally, our evaluation is confined to classification tasks; open-ended generation settings would require different consistency metrics, such as entailment-based scoring.

Broader impact. Improving logical consistency in LLMs has clear benefits for high-stakes applications—legal reasoning, medical diagnosis support, and scientific hypothesis evaluation all require that equivalent framings of a problem yield equivalent conclusions. CCT provides a lightweight, data-efficient mechanism to instil this property in existing open-weight models.

8 Conclusion

We presented **Contrastive Consistency Tuning (CCT)**, a framework for enforcing logical invariance in LLMs through symmetry pair training. By combining a formally motivated augmentation pipeline (the **Symmetry Engine**) with a representational consistency objective, CCT reduces the Contradiction Rate by up to $\approx$19 percentage points on Llama-3 8B and $\approx$10 points on Mistral-7B. Our diagnostic experiments on frontier models reveal that logical fragility is a persistent failure mode not addressed by scale. We release our augmented dataset and training code to support future research on reliable logical reasoning in foundation models.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. In *arXiv preprint arXiv:1907.02893*, 2019.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6894–6910, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.552.

Jean-Bastien Grill et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Luke Benson, Lucy Sun, Ekaterina Zubova, Yujie Qiao, Matthew Burtell, et al. FOLIO: Natural language reasoning with first-order logic. *arXiv preprint arXiv:2209.00840*, 2022. URL https://arxiv.org/abs/2209.00840.

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, 2021. URL https://arxiv.org/abs/2106.09685.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. LogiQA: A challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*, pp. 3622–3628. International Joint Conferences on Artificial Intelligence Organization, 2020. URL https://doi.org/10.24963/ijcai.2020/501.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902–4912, Online, 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.442.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3621–3634, Online, 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.findings-acl.317.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

A SYMMETRY ENGINE: TRANSFORMATION RULES AND IMPLEMENTATION DETAILS

This appendix provides the full specification of the five transformation rules implemented in the Symmetry Engine, algorithmic details of the augmentation pipeline.

A.1 FORMAL DEFINITION OF TRANSFORMATION RULES

We provide formal definitions of each transformation $\mathcal{T}_i$ alongside natural-language examples and the logical axiom that guarantees label preservation.

T1 – Relational Inversion. For any binary relation R with a well-defined converse R^{-1} (e.g. *taller/shorter*, *before/after*, *causes/is caused by*), the transformation replaces $R(a, b)$ with $R^{-1}(b, a)$. Formally, if $\varphi \equiv R(a, b)$ then $\mathcal{T}_1(\varphi) \equiv R^{-1}(b, a)$, and the equivalence $R(a, b) \Leftrightarrow R^{-1}(b, a)$ holds by the definition of a converse relation.

> *Original:* "Alice is taller than Bob."
> *Transformed:* "Bob is shorter than Alice."

T2 – Double Negation. For any proposition P, double negation elimination gives $P \Leftrightarrow \neg\neg P$. We surface this in natural language by prepending *"It is not the case that it is false that…"* or an equivalent construction.

> *Original:* "The claim is true."
> *Transformed:* "It is not the case that the claim is false."

T3 – Contrapositive. For any implication $P \Rightarrow Q$, the contrapositive $\neg Q \Rightarrow \neg P$ is logically equivalent. The Symmetry Engine identifies conditional constructions via dependency-parsed *if–then* templates and rewrites both the antecedent and consequent with their negations, swapping their positions.

> *Original:* "If it rains, the ground gets wet."
> *Transformed:* "If the ground is not wet, it did not rain."

T4 – Existential Swap. Universal quantification $\forall x.\, P(x)$ is equivalent to the negated-existential form $\neg\exists x.\, \neg P(x)$. Rewriting exploits the duality $\forall x.\, P(x) \Leftrightarrow \neg\big(\exists x.\, \neg P(x)\big)$ and surfaces it through paraphrases such as *"There is no X that is not Y"*.

> *Original:* "All mammals are warm-blooded."
> *Transformed:* "There is no mammal that is not warm-blooded."

T5 – Active/Passive Voice. Syntactic passivisation preserves propositional content: *X verbs Y* $\Leftrightarrow$ *Y is verbed by X*. The engine applies this via a subject–verb–object rewrite rule identified through dependency parsing.

> *Original:* "The enzyme catalyses the reaction."
> *Transformed:* "The reaction is catalysed by the enzyme."

A.2 Symmetry Engine Pipeline

Algorithm 1 gives the full pseudocode of the augmentation step.

Table 5 reports corpus statistics before and after augmentation.

Table 5: Corpus statistics before and after Symmetry Engine augmentation (six-rule pipeline, sentence-level splitting).

Split	Original	Augmented pairs	Expansion
FOLIO train	964	655,585	26×
ProofWriter train	24,000		
FOLIO eval	240	28,696	23×
ProofWriter eval	1,000		
Total train	24,964	655,585	26×
Total eval	1,240	28,696	23×

Algorithm 1 Symmetry Engine Augmentation

Require: Dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, transformation rules $\{\mathcal{T}_k\}_{k=1}^{6}$
Ensure: Augmented dataset $\mathcal{D}^+ = \{(s_j,\ s'_{jk},\ y_i)\}$

$\mathcal{D}^+ \leftarrow \emptyset$
for each $(x_i, y_i) \in \mathcal{D}$ **do**
  Split x_i into sentences $\{s_1, \ldots, s_m\}$ on full stops; strip `sentk:` prefixes
  for each sentence s_j **do**
    for each rule $\mathcal{T}_k$ **do**
      $s'_{jk} \leftarrow \mathcal{T}_k(s_j)$
      if $s'_{jk} \neq \emptyset$ **and** $s'_{jk} \neq s_j$ **then**
        $\mathcal{D}^+ \leftarrow \mathcal{D}^+ \cup \{(s_j,\ s'_{jk},\ y_i)\}$
      end if
    end for
  end for
end for
return $\mathcal{D}^+$

B Theoretical Analysis of Contrastive Consistency Tuning

This appendix provides formal justification for the CCT objective. We characterise the geometry of the consistency penalty in representation space, derive a generalisation bound for the combined loss, analyse the fixed points of the CCT objective, and connect the symmetry pair framework to the theory of invariant risk minimisation.

B.1 Notation and Preliminaries

Let $\mathcal{X}$ denote the space of natural language prompts and $\mathcal{Y} = \{0, 1\}$ the label space. The encoder $h_\theta : \mathcal{X} \to \mathbb{R}^d$ maps each prompt to a d-dimensional representation, and the classifier $g_\phi : \mathbb{R}^d \to \Delta^{|\mathcal{Y}|}$ maps representations to output distributions, where $\Delta^{|\mathcal{Y}|}$ denotes the probability simplex. We write $f_\theta = g_\phi \circ h_\theta$ for the full model.

A **symmetry group** $\mathcal{G}$ acts on $\mathcal{X}$ via transformations $\mathcal{T} : \mathcal{X} \to \mathcal{X}$ that preserve ground-truth labels. Formally, $\mathcal{G}$ is closed under composition and inversion, and for every $\mathcal{T} \in \mathcal{G}$ and every $(x, y) \in \mathcal{X} \times \mathcal{Y}$, the transformed pair $(\mathcal{T}(x), y)$ has the same label y. The five rules $\mathcal{T}_1, \ldots, \mathcal{T}_5$ of the Symmetry Engine each constitute an involution (i.e. $\mathcal{T}_k^2 = \mathrm{id}$) and together generate a finite subgroup $\mathcal{G}_5 \leq \mathrm{Sym}(\mathcal{X})$.

We write μ for the data-generating distribution over $\mathcal{X} \times \mathcal{Y}$, and $\mu_{\mathcal{G}}$ for its symmetrised counterpart obtained by averaging μ over the orbit of each example under $\mathcal{G}_5$:

$$\mu_{\mathcal{G}}(A) \;=\; \frac{1}{|\mathcal{G}_5|} \sum_{\mathcal{T} \in \mathcal{G}_5} \mu\big(\mathcal{T}^{-1}(A)\big)\,, \quad A \subseteq \mathcal{X} \times \mathcal{Y}. \tag{3}$$

An ideal consistent reasoner f^* is one that is $\mathcal{G}_5$-*equivariant* at the representation level, meaning $h_\theta(x) = h_\theta(\mathcal{T}(x))$ for all $\mathcal{T} \in \mathcal{G}_5$, and consequently $f^*(x) = f^*(\mathcal{T}(x))$ for all x.

B.2 Geometry of the Consistency Penalty

We first characterise what the cosine consistency penalty $D(\mathbf{z}, \mathbf{z}') = 1 - \mathbf{z}^\top \mathbf{z}' / (\|\mathbf{z}\| \|\mathbf{z}'\|)$ enforces in representation space.

Definition B.1 (Consistency Set) *For a model h_θ and symmetry group $\mathcal{G}_5$, the **consistency set** is*

$$\mathcal{C}(\theta) \;=\; \big\{x \in \mathcal{X} : D\big(h_\theta(x),\, h_\theta(\mathcal{T}(x))\big) = 0 \;\forall\, \mathcal{T} \in \mathcal{G}_5\big\}. \tag{4}$$

$D(\mathbf{z}, \mathbf{z}') = 0$ if and only if $\mathbf{z}$ and $\mathbf{z}'$ are positively collinear, i.e. $\mathbf{z}' = \alpha \mathbf{z}$ for some $\alpha > 0$. Since representations are unit-normalised before computing the penalty, this reduces to exact equality on the unit hypersphere $\mathbb{S}^{d-1}$.

Proposition B.2 (Consistency as Sphere Alignment) *Let $\hat{\mathbf{z}} = \mathbf{z}/\|\mathbf{z}\|$ and $\hat{\mathbf{z}}' = \mathbf{z}'/\|\mathbf{z}'\|$ be the unit-normalised representations. Then:*

$$D(\mathbf{z}, \mathbf{z}') \;=\; \frac{1}{2}\|\hat{\mathbf{z}} - \hat{\mathbf{z}}'\|_2^2. \tag{5}$$

Consequently, minimising the consistency penalty is equivalent to minimising the squared Euclidean distance between normalised representations on $\mathbb{S}^{d-1}$.

Expanding the squared Euclidean distance:

$$\begin{aligned}\|\hat{\mathbf{z}} - \hat{\mathbf{z}}'\|_2^2 &= \|\hat{\mathbf{z}}\|^2 - 2\,\hat{\mathbf{z}}^\top\hat{\mathbf{z}}' + \|\hat{\mathbf{z}}'\|^2 \\ &= 1 - 2\,\hat{\mathbf{z}}^\top\hat{\mathbf{z}}' + 1 = 2\big(1 - \hat{\mathbf{z}}^\top\hat{\mathbf{z}}'\big) = 2\,D(\mathbf{z}, \mathbf{z}').\end{aligned}$$

Dividing both sides by 2 gives the result.

Proposition B.2 has a useful geometric interpretation: CCT pulls symmetry-pair representations towards the same point on the unit hypersphere, contracting the geodesic distance between them on $\mathbb{S}^{d-1}$. The task loss simultaneously anchors these points near class-discriminative regions of the sphere, creating a tension that the hyperparameter λ mediates.

Corollary B.3 (Collapse Prevention) *The CCT objective does not suffer from representational collapse (all representations collapsing to a single point) provided $\lambda < \infty$ and the cross-entropy term $\mathcal{L}_{\mathrm{CE}}$ assigns strictly positive gradient magnitude to at least one training example. This follows because $\mathcal{L}_{\mathrm{CE}}$ penalises representations that are identical across classes, while D penalises representations that differ within a symmetry pair. The two terms pull in orthogonal directions whenever the class boundary separates orbits under $\mathcal{G}_5$.*

B.3 Fixed-Point Analysis

We analyse the critical points of $\mathcal{L}_{\mathrm{CCT}}$ to understand what the objective enforces at convergence.

Setup. Consider a simplified linear setting in which $h_\theta(x) = Wx$ for $W \in \mathbb{R}^{d\times p}$ and $x \in \mathbb{R}^p$. A symmetry transformation acts as a linear operator $T \in \mathbb{R}^{p\times p}$ satisfying $T^2 = I$ (an involution), so $x' = Tx$. The consistency penalty becomes:

$$D(Wx, WTx) \;=\; 1 - \frac{(Wx)^\top(WTx)}{\|Wx\|\,\|WTx\|}. \tag{6}$$

Proposition B.4 (Fixed Points of CCT) *In the linear setting, a weight matrix W^* is a fixed point of gradient descent on $\mathcal{L}_{\mathrm{CCT}}$ with respect to the consistency term if and only if W^* maps every symmetry pair (x, Tx) to positively collinear vectors, i.e. $W^*T = \alpha W^*$ for some $\alpha > 0$. When $\alpha = 1$, this is equivalent to W^* being a left-fixed-point of T: the encoder projects out the antisymmetric component of T.*

The gradient of $D(Wx, WTx)$ with respect to W vanishes when $Wx \propto WTx$ for all x in the training distribution. Since this must hold for all x, it requires W and WT to have identical column spaces (up to positive scaling), which is equivalent to $WT = \alpha W$ for some scalar $\alpha > 0$. For an involution $T^2 = I$, the eigenvalues of T are ± 1. Setting $\alpha = 1$ projects W onto the $+1$ eigenspace of T, i.e. the subspace of vectors symmetric under T, eliminating the -1 eigenspace (the antisymmetric component).

Proposition B.4 provides an explicit characterisation of what CCT learns: the encoder is driven to ignore dimensions that flip sign under the symmetry transformation and to preserve dimensions that are invariant. This is precisely the inductive bias required for logical consistency.

B.4 Generalisation Bound

We derive a PAC-Bayes–style generalisation bound for the CCT objective that quantifies the benefit of the consistency penalty in terms of reduced hypothesis complexity.

Setup. Let $\mathcal{H}$ be the hypothesis class of models f_θ parameterised by $\theta \in \Theta$. Denote by $R(f_\theta)$ the true risk under μ and by $\hat{R}_n(f_\theta)$ the empirical risk on n i.i.d. samples. Let $R_{\mathcal{G}}(f_\theta)$ be the consistency risk:

$$R_{\mathcal{G}}(f_\theta) \;=\; \mathbb{E}_{x\sim\mu_{\mathcal{X}}}\left[\frac{1}{|\mathcal{G}_5|}\sum_{\mathcal{T}\in\mathcal{G}_5} D\big(h_\theta(x),\, h_\theta(\mathcal{T}(x))\big)\right], \tag{7}$$

where $\mu_{\mathcal{X}}$ is the marginal of μ over $\mathcal{X}$.

Theorem B.5 (CCT Generalisation Bound) *Assume the loss function $\ell : \mathcal{Y}\times\mathcal{Y} \to [0,1]$ is bounded. Let $\mathcal{H}_\varepsilon = \{f_\theta \in \mathcal{H} : R_{\mathcal{G}}(f_\theta) \le \varepsilon\}$ be the subset of hypotheses with consistency risk at most ε. For any $\delta \in (0,1)$, with probability at least $1-\delta$ over draws of n training samples, every $f_\theta \in \mathcal{H}_\varepsilon$ satisfies:*

$$R(f_\theta) \;\le\; \hat{R}_n(f_\theta) + \mathfrak{R}_n(\mathcal{H}_\varepsilon) + \sqrt{\frac{\ln(1/\delta)}{2n}} + \lambda\varepsilon, \tag{8}$$

where $\mathfrak{R}_n(\mathcal{H}_\varepsilon)$ is the Rademacher complexity of $\mathcal{H}_\varepsilon$.

[Proof sketch] The standard Rademacher complexity bound gives, for any $f_\theta \in \mathcal{H}$:

$$R(f_\theta) \le \hat{R}_n(f_\theta) + \mathfrak{R}_n(\mathcal{H}) + \sqrt{\frac{\ln(1/\delta)}{2n}}.$$

Restricting to $\mathcal{H}_\varepsilon$ replaces $\mathfrak{R}_n(\mathcal{H})$ with $\mathfrak{R}_n(\mathcal{H}_\varepsilon)$. Since $\mathcal{H}_\varepsilon \subseteq \mathcal{H}$, we have $\mathfrak{R}_n(\mathcal{H}_\varepsilon) \le \mathfrak{R}_n(\mathcal{H})$. The additional $\lambda\varepsilon$ term accounts for the maximum contribution of the consistency penalty to the empirical CCT objective: for any $f_\theta \in \mathcal{H}_\varepsilon$, the consistency term contributes at most $\lambda\varepsilon$ to the total loss, which in the worst case inflates the generalisation gap by the same amount.

Remark B.6 (Complexity Reduction) *The key quantity in equation equation 8 is $\mathfrak{R}_n(\mathcal{H}_\varepsilon)$. Imposing consistency constraints restricts the hypothesis class: a model must simultaneously fit the data and align representations of symmetry pairs. This additional constraint reduces the effective degrees of freedom available to h_θ, one expects $\mathfrak{R}_n(\mathcal{H}_\varepsilon) \le \mathfrak{R}_n(\mathcal{H})$ to hold strictly for $\varepsilon < \varepsilon^*$, where ε^* is the consistency risk of the unconstrained ERM solution. In other words, CCT performs an implicit regularisation of the hypothesis class through the consistency constraint, reducing the generalisation gap beyond what cross-entropy alone achieves.*

B.5 Connection to Invariant Risk Minimisation

Invariant Risk Minimisation (Arjovsky et al., 2019) seeks a representation Φ such that the optimal classifier on top of Φ is the same across all environments $e \in \mathcal{E}$. Formally, IRM minimises:

$$\min_{\Phi,\, w} \sum_{e\in\mathcal{E}} R^e(w\circ\Phi) \quad \text{s.t.} \quad w \in \arg\min_{\bar{w}} R^e(\bar{w}\circ\Phi) \quad \forall\, e \in \mathcal{E}. \tag{9}$$

We establish a formal correspondence between CCT and IRM by constructing a natural environment partition from the symmetry group.

Definition B.7 (Symmetry Environments) *For each transformation $\mathcal{T}_k \in \mathcal{G}_5$, define an environment e_k as the distribution obtained by applying $\mathcal{T}_k$ to samples drawn from μ:*

$$\mu^{e_k}(A\times B) \;=\; \mu\big(\mathcal{T}_k^{-1}(A)\times B\big), \quad A\subseteq\mathcal{X},\; B\subseteq\mathcal{Y}. \tag{10}$$

The base environment e_0 corresponds to the original distribution μ. The full environment set is $\mathcal{E} = \{e_0, e_1, \ldots, e_5\}$.

Proposition B.8 (CCT as Approximate IRM) *A model f_θ that achieves zero consistency risk, $R_{\mathcal{G}}(f_\theta) = 0$, satisfies the IRM invariance constraint across the symmetry environments $\mathcal{E} = \{e_0, \ldots, e_5\}$ with respect to any linear classifier g_ϕ applied to h_θ. Conversely, the IRM penalty evaluated on the symmetry environments upper-bounds the CCT consistency penalty up to a constant depending on the Lipschitz constant of g_ϕ.*

[Proof sketch] Zero consistency risk implies $h_\theta(x) = h_\theta(\mathcal{T}_k(x))$ on $\mathbb{S}^{d-1}$ for all k and μ-almost all x. Since g_ϕ is a fixed linear classifier, identical representations imply identical output distributions: $f_\theta(x) = f_\theta(\mathcal{T}_k(x))$. This means the risk under each environment e_k equals the risk under e_0, satisfying IRM invariance. For the converse, IRM invariance of f_θ across $\mathcal{E}$ implies that $g_\phi(h_\theta(x)) = g_\phi(h_\theta(\mathcal{T}_k(x)))$, which for an injective g_ϕ implies $h_\theta(x) = h_\theta(\mathcal{T}_k(x))$, recovering zero consistency risk. The Lipschitz constant of g_ϕ mediates the case when g_ϕ is not injective.

Proposition B.8 situates CCT within the broader invariant learning literature. The key difference from standard IRM is that CCT constructs environments automatically from logical symmetry rules rather than requiring manually specified environment labels, making it applicable without access to environment annotations.

B.6 Why Cross-Entropy Alone Cannot Enforce Consistency

A natural question is whether standard cross-entropy training on a sufficiently large and diverse dataset would implicitly learn the desired consistency. We provide a formal argument showing that this is not guaranteed.

Proposition B.9 (CE Insufficiency) *There exists a data distribution μ and a hypothesis class $\mathcal{H}$ such that the empirical risk minimiser under cross-entropy, $\hat{f} = \arg\min_{f \in \mathcal{H}} \hat{R}_n(f)$, achieves zero training error yet has arbitrarily high contradiction rate $\mathrm{CR}(\hat{f}) \to 1$.*

Construct μ as follows. Let $\mathcal{X} = \{x_1, x_1', x_2, x_2', \ldots\}$ where (x_i, x_i') are symmetry pairs with label $y_i \in \{0, 1\}$. Suppose the training set $\mathcal{S}$ contains x_i but not x_i' for every i (i.e. transformed prompts are never observed during training, only at test time). Let $\mathcal{H}$ be rich enough to shatter $\mathcal{S}$ (e.g. a sufficiently wide neural network). The ERM solution $\hat{f}$ can assign $\hat{f}(x_i) = y_i$ for all i while assigning arbitrary labels to $\{x_i'\}$, since these are unseen. In particular, $\hat{f}$ can satisfy $\hat{f}(x_i') = 1 - y_i$ for all i, yielding $\mathrm{CR}(\hat{f}) = 1$. The cross-entropy loss provides no gradient signal on $\{x_i'\}$ when these examples are absent from training.

Remark B.10 (Data Augmentation vs. Representational Constraint) *Proposition B.9 implies that naïve data augmentation—adding x_i' to the training set with label y_i and training with cross-entropy—is a strictly weaker intervention than CCT. Augmentation encourages consistent* outputs *on observed pairs but provides no explicit pressure towards consistent* representations*. A model trained with augmentation can achieve low training error on both x_i and x_i' while placing them in distant regions of representation space, preserving the potential for inconsistency on unseen transformations outside the augmented distribution. CCT's consistency penalty directly constrains the geometry of h_θ, providing a stronger inductive bias that generalises to unseen transformation instances.*

B.7 Information-Theoretic Interpretation

We conclude with an information-theoretic perspective that connects the CCT objective to mutual information maximisation.

Let X and $X' = \mathcal{T}(X)$ be jointly distributed random variables linked by a symmetry transformation, and let $Z = h_\theta(X)$, $Z' = h_\theta(X')$ be their representations. The mutual information $I(Z; Z')$ measures how much information the representation of a prompt shares with the representation of its symmetric counterpart.

Proposition B.11 (Consistency as Mutual Information) *For unit-normalised Gaussian representations in $\mathbb{R}^d$, minimising the expected cosine distance $\mathbb{E}[D(Z, Z')]$ is equivalent to maximising a lower bound on the mutual information $I(Z; Z')$.*

[Proof sketch] By the data processing inequality, any deterministic function of Z cannot increase mutual information. For unit-normalised Gaussians, the cosine similarity $\hat{Z}^\top \hat{Z}'$ is a sufficient statistic for the correlation between Z and Z', and the mutual information between two jointly Gaussian variables with correlation ρ is $I = -\frac{1}{2}\ln(1 - \rho^2)$, which is monotonically increasing in $|\rho|$. Since $D(Z, Z') = 1 - \hat{Z}^\top \hat{Z}'$, minimising D maximises $\hat{Z}^\top \hat{Z}'$, which maximises $I(Z; Z')$ within this Gaussian approximation.

This perspective connects CCT to the family of *self-supervised* objectives such as SimCSE (Gao et al., 2021) and BYOL (Grill et al., 2020), where the goal is to maximise mutual information between augmented views of the same input. The crucial distinction is that CCT augmentations are *truth-preserving by logical construction* rather than by domain-specific heuristic (e.g. image crops or colour jitter), providing a formal guarantee that the learned invariances correspond to genuine logical equivalences rather than incidental surface-level similarities.

Taken together, the analyses in this appendix establish that CCT is theoretically well-motivated from multiple complementary perspectives: as a geometric alignment procedure on the unit hypersphere (Section B.2), as an implicit regulariser that reduces hypothesis complexity (Section B.4), as an instance of invariant risk minimisation over symmetry-induced environments (Section B.5), and as a mutual information maximisation objective over logically equivalent views (Section B.7).