Contrastive Loss Integration in Student-Teacher Paradigms for CodeT5 Robustness on MBXP Python

Assignee Research

June 7, 2026

Abstract

This report synthesises findings from 16 peer-reviewed papers addressing the following research question: How does integrating contrastive loss into student-teacher paradigms affect CodeT5's robustness against adversarial perturbations in the MBXP Python subset. 11 claims were extracted from source literature; 0 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 3.5/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: A Deep Dive into Adversarial Robustness in Zero-Shot Learning. Research question: How does integrating contrastive loss into student-teacher paradigms affect CodeT5's robustness against adversarial perturbations in the MBXP Python subset?.

2 Methodology

Systematic literature search across multiple databases yielded 16 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 3.5/10.

3 Results

16 papers retrieved. 11 claims extracted; 0 independently verified. Quality review score: 3.5/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
The CUB dataset has 312 attributes, 200 classes, and 11788 images.	×	0.02
The SUN dataset has 102 attributes, 717 classes, and 14340 images.	×	0.02
The AWA2 dataset has 85 attributes, 50 classes, and 37322 images.	×	0.02
The standard per-class top-1 accuracy is used for ZSL evaluation.	×	0.05
For GZSL, per-class top-1 accuracy values for seen and unseen classes are used to compute harmonic-scores.	×	0.04
The ResNet-101 feature extractor is merged with the ALE model to make the computational graph end-to-end differentiable.	×	0.03
The ALE model is trained with a frozen feature extractor for each dataset.	×	0.03
PyTorch is used for the experiments.	×	0.04
The ALE model is formulated as F(x, y; W) = θ(x)W T φ(y), where θ(x) is the visual and φ(y) is the class embeddings.	×	0.04
The compatibility function F() in the ALE model is parametrized by learnable weights W.	×	0.02
The ALE model is selected because it showed direct mapping by exploiting data and auxiliary information is more effectiv	×	0.03

References

- http://arxiv.org/abs/1807.09380v3
- http://arxiv.org/abs/2403.13322v3

- http://arxiv.org/abs/2008.07651v1