NAACL2022

Embedding Hallucination for Few-shot Language Fine-tuning

Yiren Jian *, Chongyang Gao *, Soroush Vosoughi

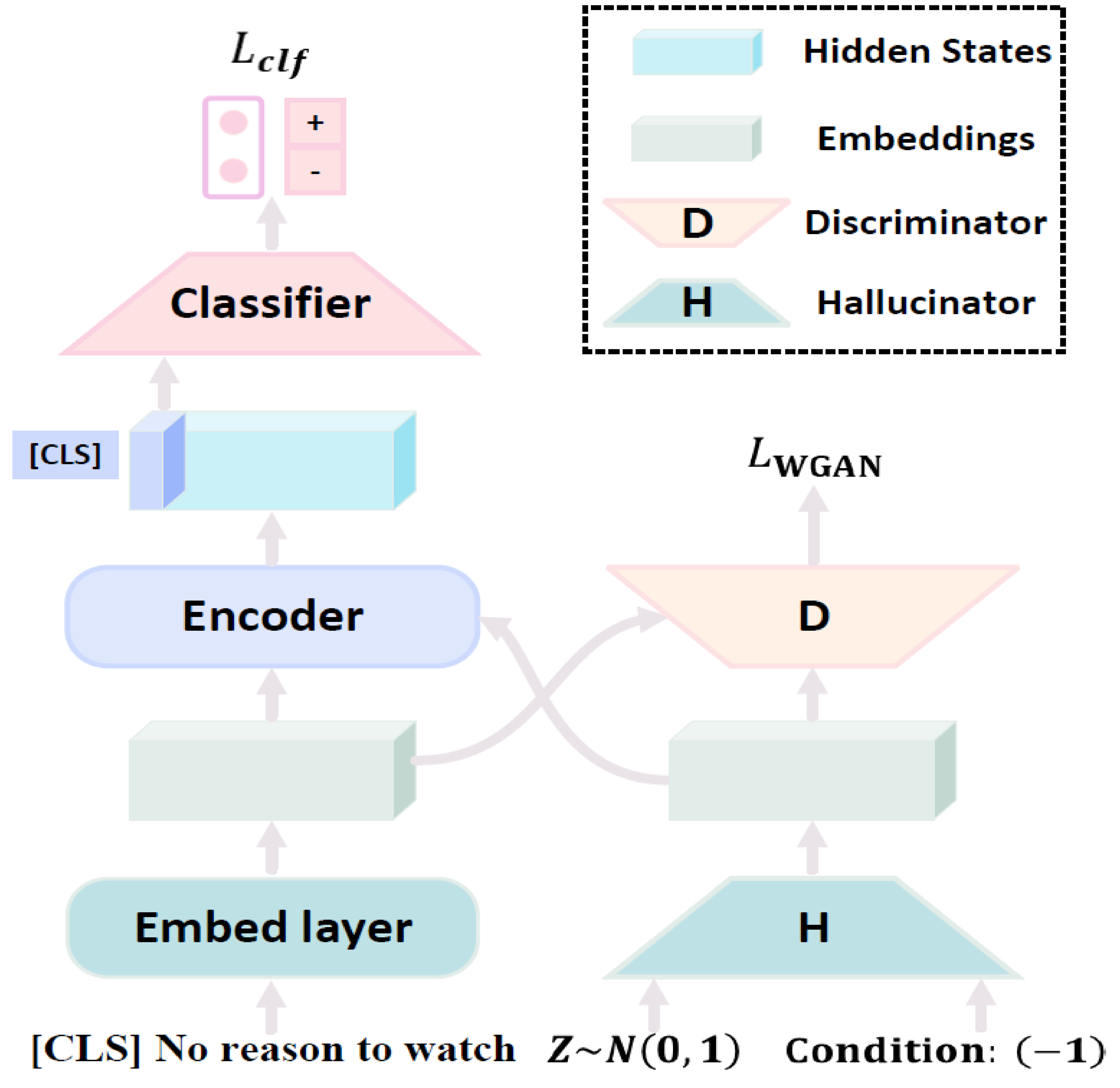
DARTMOUTH
Department of Computer Science

Northwestern | McCORMICK SCHOOL OF ENGINEERING
Computer Science

Contributions

- An embedding hallucination method for data augmentation for few-shot learning, based on cWGAN [1].
- Evaluate Embedding Hallucination on 15 tasks and show that it generally improves over recent fine-tuning methods.
- Show the overall superiority of EmbedHalluc when comparing to regularization methods proposed to address the problem of over-fitting during fine-tuning of LMs and outperforms a common augmentation method.

Overview



Overview of our method

Hallucinated Embedding

$$\mathcal{L}_{\text{halluc}} = \text{KL}(\mathcal{M}(s_{\text{halluc}}(c_i)), c_{\text{pseudo},i})$$

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{real}} + \mathcal{L}_{\text{halluc}}$$

Algorithm

Algorithm 1 Our method: EmbedHalluc

$Max\_Step = 1000$,
LM: Language model,
H: Emebedding hallucinator (pre-trained),
$Train\_Set$: Training set,
$Sample$: Randomly sampling function,
CE: Cross Entropy loss,
KL: KL-divergence loss.
for i in Max_Step **do** ▷ Training LM_1
  $sent, y = Sample(Train\_Set)$
  $output_1 = LM_1(sent)$
  $L = CE(output_1, y)$
  $L.backward()$
  $optimizer.step()$
end for
for i in Max_Step **do** ▷ Training LM_2
  $sent, y = Sample(Train\_Set)$
  $embed = H(\mathcal{N}(0,1)), c)$
  ▷ Learning from real text
  $output_1 = LM_2(sent)$
  $L_{real} = CE(output_1, y)$
  $L_{real}.backward()$
  $optimizer.step()$
  ▷ Learning from hallucination
  $prob_2 = LM_1(embed)$
  $output_2 = LM_2(embed)$
  $L_{halluc} = KL(prob_2, output_2)$
  $L_{halluc}.backward()$
  $optimizer.step()$
end for
return LM_2

Conventional fine-tuning

Task	Fine-tuning	EmbedHalluc	w/LabelCalib
SST-2 (acc)	76.8 (4.2)	**82.6** (5.6)	82.0 (4.7)
Subj (acc)	90.3 (1.5)	**91.3** (0.8)	**91.3** (0.9)
SST-5 (acc)	40.6 (2.2)	40.3 (1.5)	**41.6** (2.6)
CoLA (Matt.)	36.0 (9.9)	**39.7** (10.8)	38.1 (11.8)
TREC (acc)	83.0 (4.9)	**88.1** (2.5	87.9 (1.0)
MNLI (acc)	41.6 (5.2)	48.0 (9.5)	**49.6** (5.8)
MNLI-mm (acc)	42.7 (5.9)	49.7 (10.5)	**51.8** (6.1)
SNLI (acc)	52.9 (6.7)	**54.4** (3.4)	52.3 (5.3)
QNLI (acc)	55.3 (2.7)	60.2 (5.3)	**64.9** (5.1)
QQP (acc)	59.2 (8.6)	64.6 (5.0)	**66.7** (5.3)
RTE (acc)	52.9 (1.4)	53.4 (1.7)	**55.9** (4.3)
MRPC (F1)	76.3 (5.2)	**78.7** (1.9)	78.1 (3.0)
MR (acc)	74.5 (5.9)	79.4 (5.5)	**80.8** (3.2)
MPQA (acc)	65.0 (1.5)	70.1 (7.0)	**70.5** (4.6)
CR (acc)	71.7 (7.5)	75.1 (5.6)	**78.0** (3.8)

Comparison of conventional fine-tuning and our EmbedHalluc

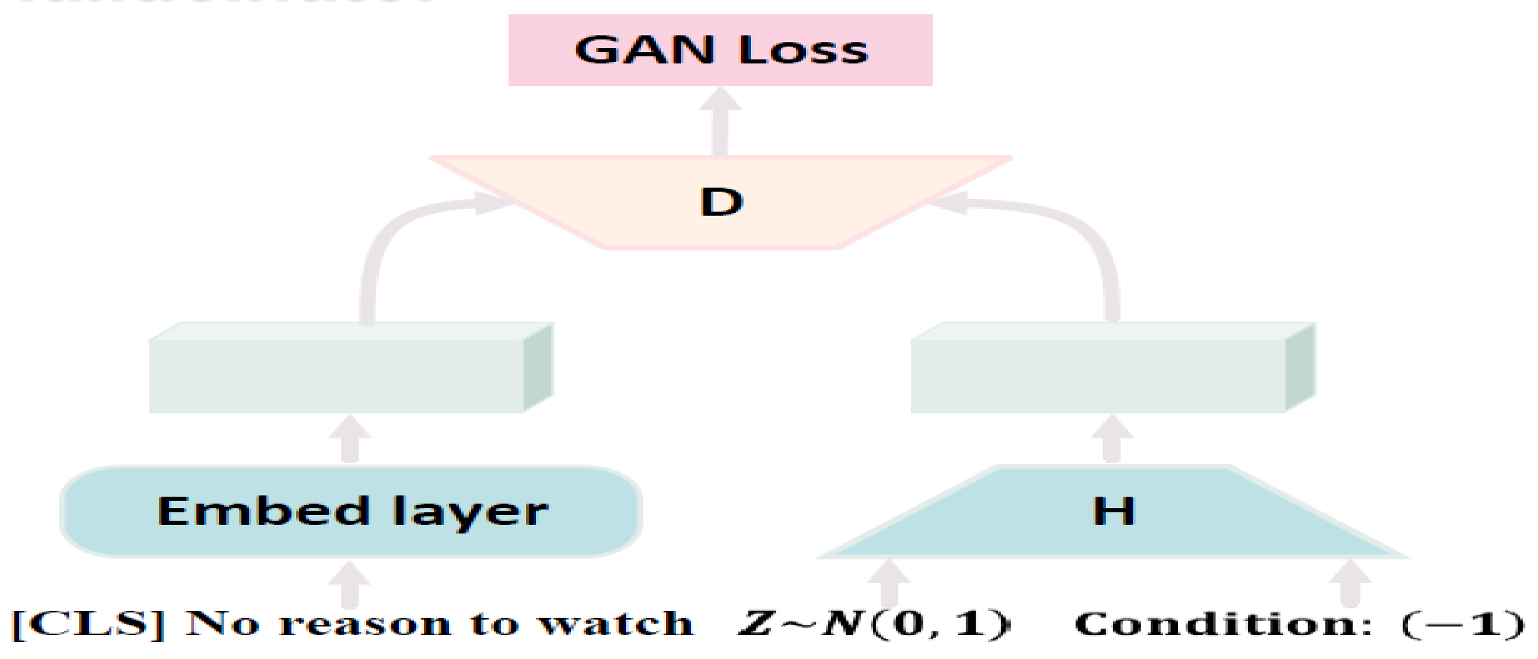
Prompt-based fine-tuning

Task	Prompt-based	EmbedHalluc	w/LabelCalib
SST-2 (acc).	92.7 (0.4)	92.8 (0.7)	**93.1** (0.7)
Subj (acc)	91.3 (1.0)	**92.0** (0.4)	91.7 (1.3)
SST-5 (acc)	48.8 (1.0)	49.0 (2.2)	**49.4** (1.4)
CoLA (Matt.)	7.3 (5.8)	12.3 (7.6)	**22.1** (15.6)
TREC (acc)	83.8 (5.3)	85.5 (3.3)	**87.1** (2.9)
MNLI (acc)	**69.7** (2.0)	68.0 (2.8)	68.5 (1.7)
MNLI-mm (acc)	**71.5** (1.9)	69.9 (3.0)	70.6 (1.7)
SNLI (acc)	78.0 (3.0)	**78.8** (2.3)	78.4 (2.3)
QNLI (acc)	68.6 (2.8)	69.6 (0.3)	**71.6** (2.0)
QQP (acc)	70.2 (4.3)	71.9 (5.2)	**74.2** (0.9)
RTE (acc)	**70.9** (3.3)	69.9 (3.3)	66.9 (3.4)
MRPC (F1)	74.6 (6.8)	78.0 (4.9)	**80.3** (3.5)
MR (acc)	86.8 (0.9)	87.2 (0.9)	**87.5** (0.9)
MPQA (acc)	85.4 (1.8)	84.2 (1.9)	**85.4** (1.9)
CR (acc)	91.1 (1.0)	91.1 (0.9)	**91.3** (0.3)

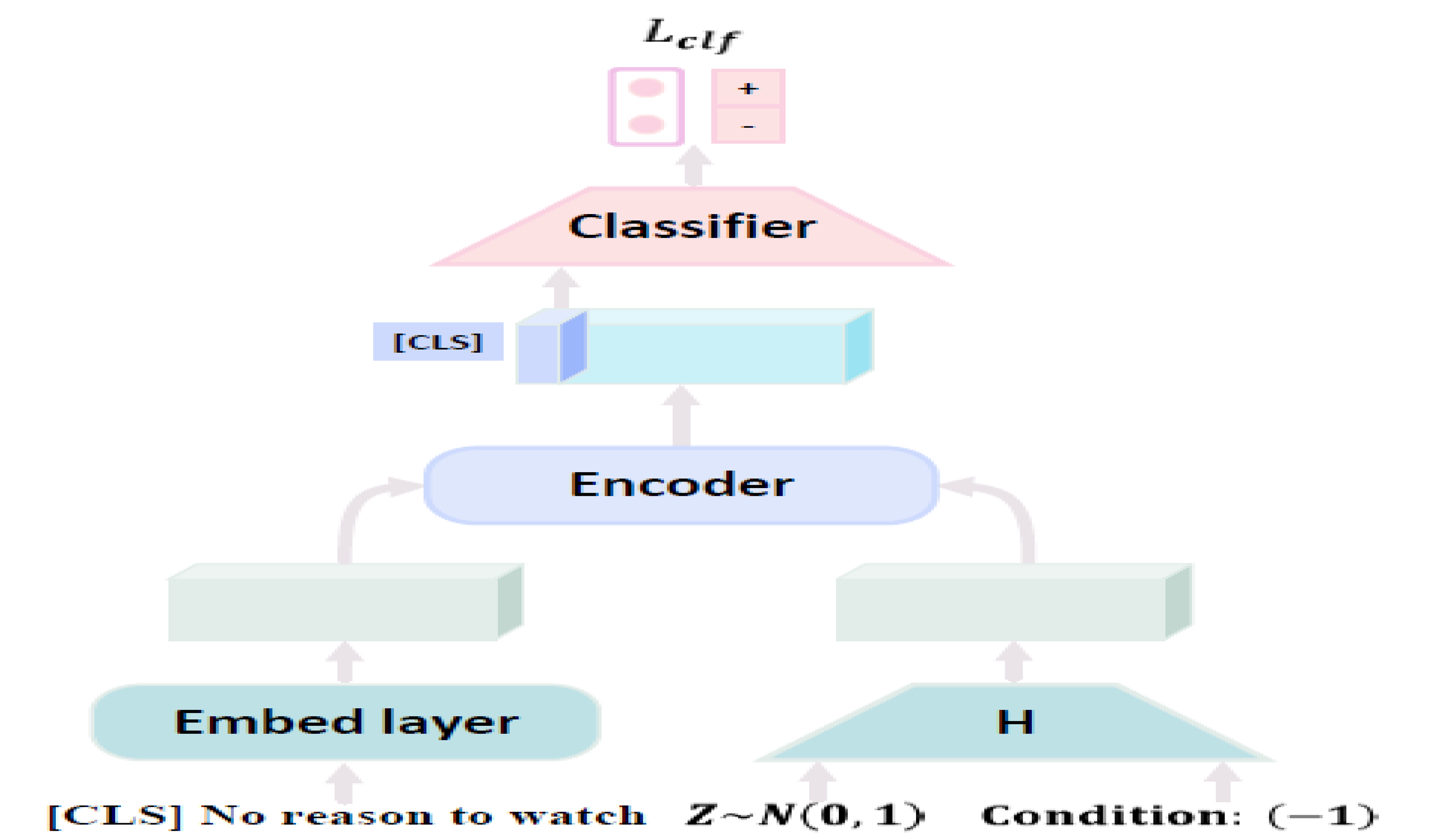
Comparison of prompt-based fine-tuning and our EmbedHalluc

[1] Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks." International conference on machine learning. PMLR, 2017.
[2] Zhang, Tianyi, et al. "Revisiting Few-sample BERT Fine-tuning." International Conference on Learning Representations. 2020.
[3] Lee, Cheolhyoung, Kyunghyun Cho, and Wanmo Kang. "Mixout: Effective Regularization to Finetune Large-scale Pretrained Language Models." (ICLR)., 2020.
[4] Cascante-Bonilla, Paola, et al. "Curriculum Labeling: Revisiting Pseudo-Labeling for Semi-Supervised Learning." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 35. No. 8. 2021.

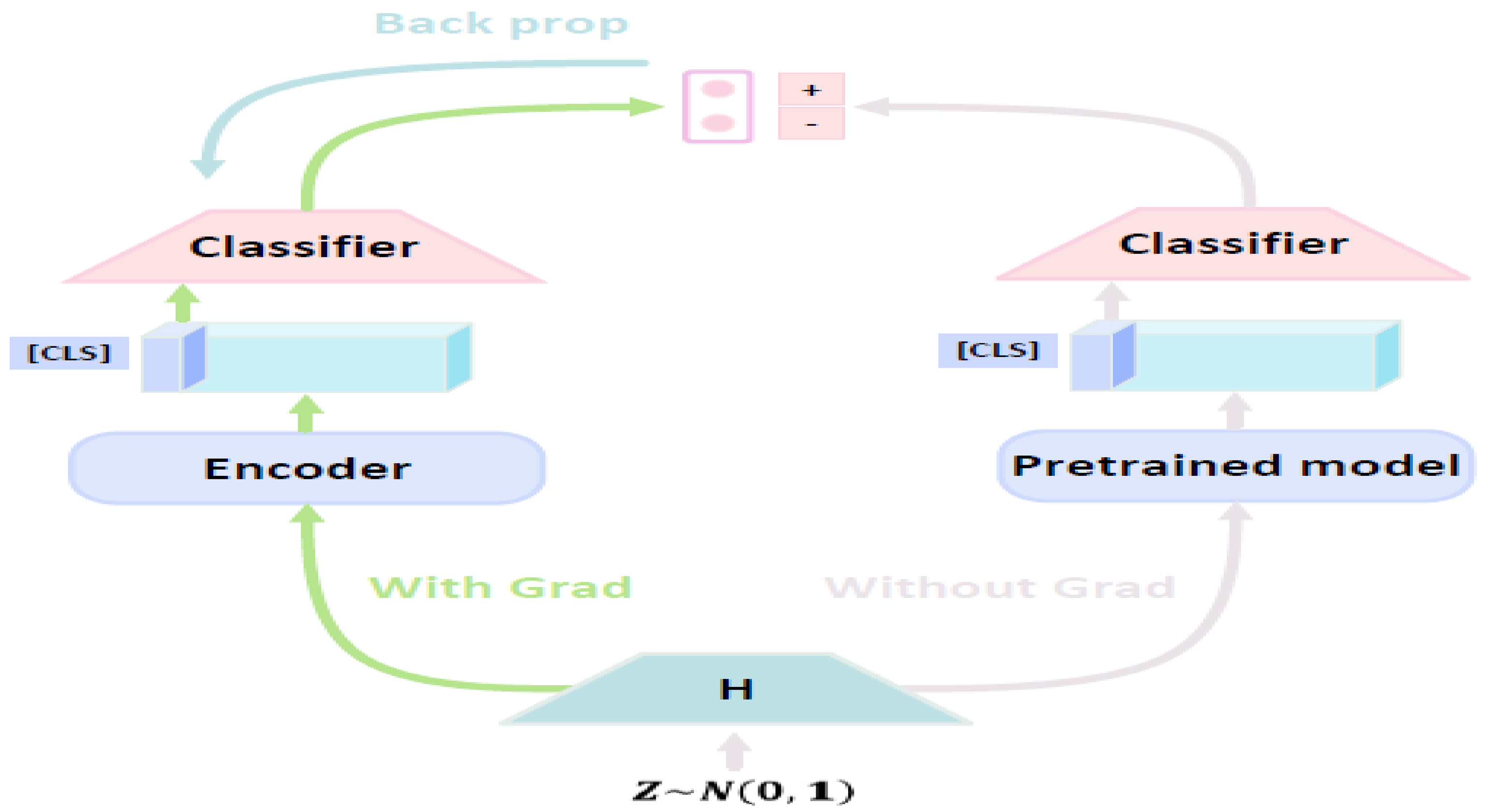
Hallucinator



Training of Hallucinator



Training with “hallucinated”, “augmented” pseudo-embeddings



Training with pseudo-labeled “hallucinated” embeddings.