



# Adaptive Reinforcement Tuning Language Models as Hard Data Generators for Sentence Representation

Bo Xu, **Yifei Wu**, Shouang Wei, Ming Du, Hongya Wang

School of Computer Science and Technology, Donghua University, Shanghai, China





# Outline

- **Background**
- Motivation of Our Work
- Our Framework
- Experiments
- Conclusions



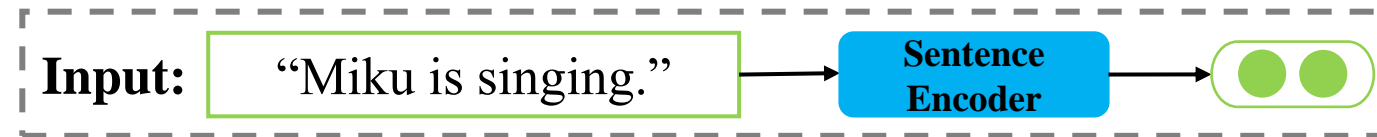


# Background


- Sentence Representation

- Task Definition

- Given a sentence as input, the task of sentence representation is to get the vectorized representation of the sentence.



- Synthetic Data Generation

- Commercial Large language models (LLMs) like ChatGPT and GPT4 can automatically generate high-quality contrastive data.
    - Data from the small-parameter LLMs is poor: 1) the generated data may contain errors. Requesting a positive sample might result in a negative one, and vice versa. 2) The generated data might not be sufficiently hard, leading to minimal improvements in the performance of existing sentence representation models. 



# Outline

- Background
- **Motivation of Our Work**
- Our Framework
- Experiments
- Conclusions





# Motivation of Our Work

- Limitations of Existing Work

- Large language models (LLMs) like ChatGPT and GPT4 can automatically generate high-quality contrastive data. A more cost-effective strategy is to utilize small-parameter LLMs to generate synthetic data and enhance sentence representation models. However, this alternative strategy also encounters challenges:
  - 1) Obtaining high-quality generated data from small-parameter LLMs is difficult.
  - 2) Inefficient utilization of the generated data. Existing methods typically generate a large amount of homogeneous data at once to ensure adequate model training.





# Motivation of Our Work

## • Our Opinion

- Specifically, to address the first challenge, we introduce a reinforcement learning approach for fine-tuning small-parameter LLMs, enabling the generation of high-quality contrastive data without human feedback.
- To address the second challenge, we propose an adaptive iterative framework to guide small-parameter LLMs to generate progressively harder samples through multiple iterations, thereby maximizing the utility of generated sentence data.





# Outline

- Background
- Motivation of Our Work
- **Our Framework**
- Experiments
- Conclusions





# Our Framework

## • Problem Formulation

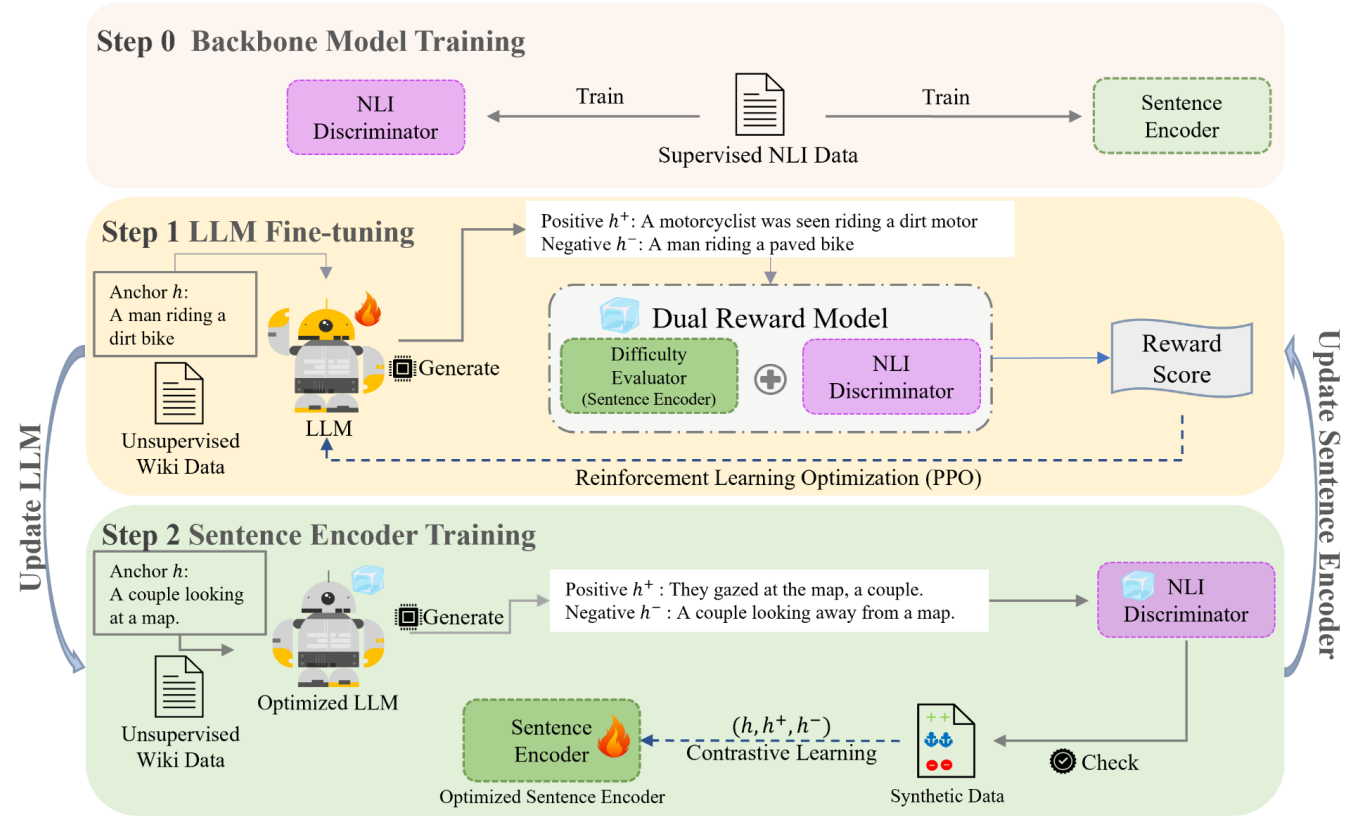
- Given the supervised training corpus  $X$ , unsupervised corpus  $D$  and the existing supervised sentence encoder  $f_\theta$ , where  $X$  consists of a set of labeled contrastive data  $\{x_i, x_i^+, x_i^-\}$ ,  $D$  consists of a set of unlabeled sentences and the supervised sentence encoder  $f_\theta$  is initially trained with  $X$ .
- Our objective is to generate synthetic contrastive sentence sample data  $\tilde{X}$  from unsupervised corpus  $D$  to improve the performance of existing supervised sentence encoder  $f_\theta$ .





# Our Framework

- Overall Architecture of Our framework



Our adaptive reinforcement tuning (ART) framework consisting of an initialization backbone model training step followed by two adaptive iterative steps, namely the LLM fine-tuning step and the sentence encoder training step.

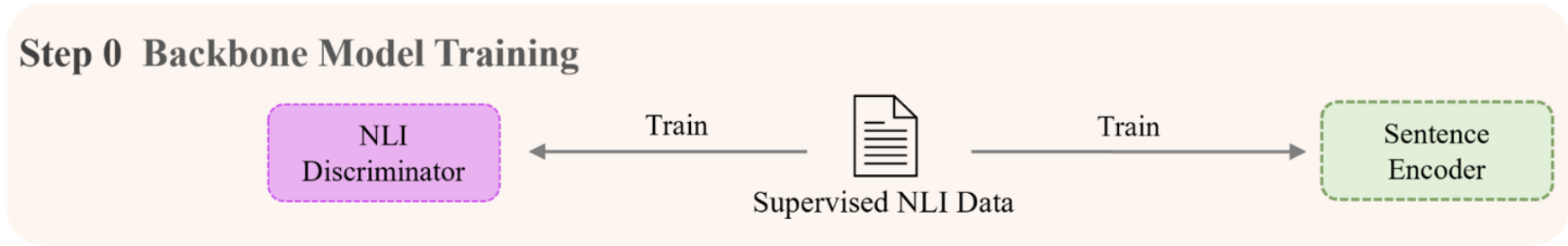




# Our Framework

- Backbone Model Training Step

- The backbone model training step utilizes the NLI dataset to train an initial supervised sentence encoder and an NLI discriminator.



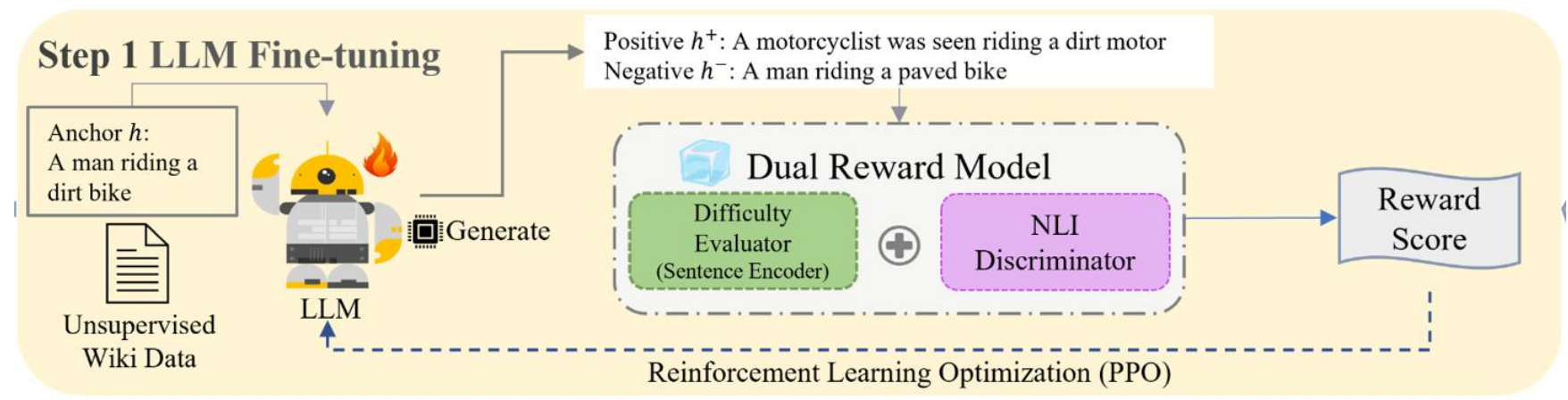
Backbone model training



# Our Framework

- LLM Fine-tuning Step

- The LLM fine-tuning step aims to improve the ability of the small-parameter LLM to generate hard synthetic data from unlabeled sentences  $D$ . Due to the lack of supervised data, we use reinforcement learning to fine-tune the small-parameter LLM.



LLM Fine-tuning





# Our Framework

- LLM Fine-tuning Step
  - Reinforcement Learning Optimization
    - We employ the proximal policy optimization (PPO) algorithm to optimize the LLM on our environment to generate harder samples. The environment is a bandit environment that presents a positive/negative prompt and an anchor sample  $x$  from the unlabeled training corpora  $D$  and expects a positive/negative sample  $y$  to the corresponding prompts.

$$E_{(x,y) \sim \mathcal{D}} [r_{\theta}(x, y) - \beta D_{KL}(\pi_{\phi}^{\text{RL}}(y | x) || \pi^{\text{ref}}(y | x))]$$

PPO Algorithm we used

---

**Positive Prompt:** Generate a positive variation of Original Sentence, ensuring it has same meaning, exhibits different syntactical and grammatical structures. Original: "[X]" Positive:

---

**Negative Prompt:** Generate a negative variation of Original Sentence, ensuring it has a completely different meaning, similar syntax and grammar. Original: "[X]" Negative:

---

Prompts used to generate hard positive and negative samples, respectively. [X] refers to the input (anchor) sentence.





# Our Framework

- LLM Fine-tuning Step

- Dual Reward Model

- We introduce a novel dual-reward model to assess the quality of samples generated by the LLM, which consists of an NLI discriminator and a difficulty evaluator based on the sentence encoder. The former is used to determine the correctness of the generated data, while the latter assesses the difficulty of the generated data.



Reward Model

$$r_1 = P(t|x, y) - \Omega$$
$$r_2 = \begin{cases} (1 - \text{sim}(x, y^+)) \cdot \text{sgn}(\text{sim}(x, y^+) - \alpha^+) \\ \text{sim}(x, y^-) \cdot \text{sgn}(\alpha^- - \text{sim}(x, y^-)), \end{cases}$$
$$r_\theta(x, y) = w_1 \times r_1 + w_2 \times r_2,$$

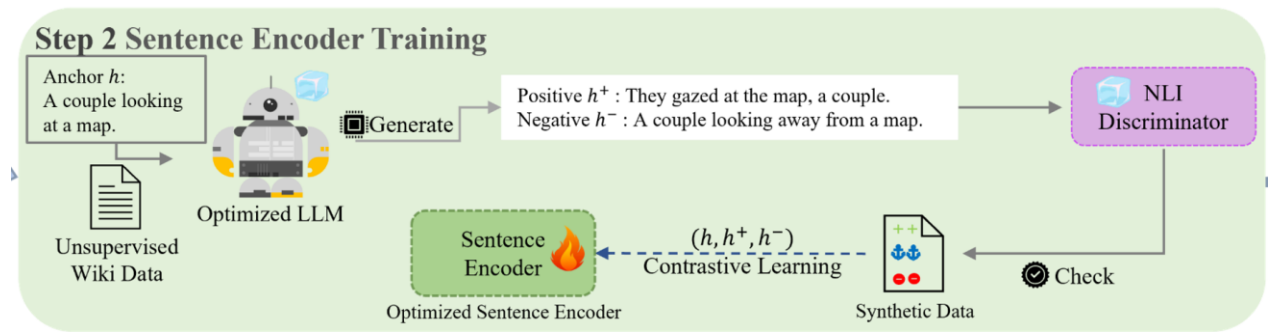
Score Equation



# Our Framework

- Sentence Encoder Training Step

- The sentence encoder training step uses the synthetic data generated by the fine-tuned small-parameter LLM to further train the sentence encoder, which consists of a data synthesis process and an encoder training process.



Sentence Encoder Training

$$\mathcal{L} = -\log \frac{e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_i^+) / \tau}}{\sum_{j=1}^N (e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^+) / \tau} + e^{\text{sim}(\mathbf{h}_i, \mathbf{h}_j^-) / \tau})}$$

SimCSE loss





# Outline

- Background
- Motivation of Our Work
- Our Framework
- **Experiments**
- Conclusions





# Experiments

## • Data

- The training data is sourced from two categories: 1) Supervised NLI dataset, which is a combination of SNLI and MNLI datasets. 2) Synthetic data, generated by our large language model through multi-round generation based on sentences from English Wikipedia.
- At each round of training, we generated 20k triplets with vLLM.
- The testing data consists of seven standard STS datasets.



# Experiments



## • Training Details

- We conduct all the experiments on 2 Nvidia RTX A6000 GPUs with PyTorch 2.0.0 in Nvidia-docker image
- In the LLM fine-tuning step, we employ WizardLM 7B as the small-parameter LLM, and use LoRA for efficient finetune, with parameters  $r = 16$  and  $\alpha = 32$ .





# Experiments

## • Result

### • Performance Comparison

- Our method achieves strong overall results with far fewer data, yet achieves results comparable to or even surpassing those obtained by directly using large-parameter LLMs.

Methods	Extra Data	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	AVG.
<b>BERT-large</b>									
SimCSE	-	75.78	86.33	80.44	86.06	80.86	84.87	<b>81.14</b>	82.21
SimCSE*	270K Raw-LLM	77.50	87.11	81.44	<u>87.09</u>	82.96	<u>85.45</u>	<u>80.74</u>	83.18
PromCSE	-	<u>78.43</u>	<u>87.31</u>	<u>82.09</u>	<b>87.85</b>	<u>83.16</u>	<b>85.62</b>	80.74	<u>83.60</u>
SynCSE	270K ChatGPT	78.30	87.26	81.27	86.87	82.88	85.44	80.73	83.25
Ours	3*20K RL-LLM	<b>79.00</b>	<b>87.85</b>	<b>82.25</b>	87.42	<b>83.51</b>	85.35	80.17	<b>83.65</b>
<b>RoBERTa-large</b>									
SimCSE	-	77.46	87.27	82.36	86.66	83.93	86.70	81.95	83.76
SimCSE*	270K Raw-LLM	<u>79.98</u>	87.57	82.80	86.67	84.64	86.03	81.58	84.18
PromCSE	-	79.14	<b>88.64</b>	<b>83.73</b>	87.33	84.57	<b>87.84</b>	<u>82.07</u>	<b>84.76</b>
SynCSE	270K ChatGPT	77.13	87.61	82.82	<u>87.67</u>	<b>85.66</b>	<u>87.22</u>	<b>82.45</b>	84.37
Ours	3*20K RL-LLM	<b>80.38</b>	<u>88.63</u>	<u>83.61</u>	<b>87.70</b>	<u>85.05</u>	86.45	80.78	<u>84.66</u>
<b>T5-base</b>									
GenSE	-	80.72	87.43	83.96	88.63	85.19	87.65	79.87	84.78
GenSE*	270K Raw-LLM	<u>80.84</u>	87.54	84.23	88.72	85.31	87.72	79.63	84.86
GenSE+	4M QA (Real)	80.65	<b>88.18</b>	<b>84.69</b>	<b>89.03</b>	<b>85.82</b>	<b>87.88</b>	<u>80.10</u>	<b>85.19</b>
Ours	3*20K RL-LLM	<b>81.21</b>	<u>87.93</u>	<u>84.41</u>	<u>88.83</u>	<u>85.36</u>	<u>87.87</u>	<b>80.21</b>	<u>85.12</u>

Table 2: Results on seven STS datasets. (**Bold**: the best. Underlined: the second best.)



# Experiments



## • Result

### • Effectiveness of Adaptive Iterative Framework

- We compare our method to several state-of-the-art approaches based on two existing supervised sentence representation methods
- Our results show an adaptive enhancement in the model's performance with each successive round. As expected, a stronger baseline gains fewer performance upraise.

Base Model	Methods	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	AVG.
BERT-large	Round 0	75.78	86.33	80.44	86.06	80.86	84.87	<b>81.14</b>	82.21
	Round 1	78.36	87.26	81.66	87.22	83.15	85.51	80.85	83.43
	Round 2	78.66	87.64	82.08	87.39	83.37	<b>85.52</b>	80.47	83.59
	Round 3	<b>79.00</b>	<b>87.85</b>	<b>82.25</b>	<b>87.42</b>	<b>83.51</b>	85.35	80.17	<b>83.65</b>
RoBERTa-large	Round 0	77.46	87.27	82.36	86.66	83.93	86.70	<b>81.95</b>	83.76
	Round 1	78.98	88.43	83.54	87.61	85.00	<b>86.81</b>	81.25	84.52
	Round 2	79.42	<b>88.64</b>	<b>83.62</b>	87.69	<b>85.27</b>	86.64	80.95	84.60
	Round 3	<b>80.38</b>	88.63	83.61	<b>87.70</b>	85.05	86.45	80.78	<b>84.66</b>
GenSE-T5-Base	Round 0	80.72	87.43	83.96	88.63	85.19	87.65	79.87	84.78
	Round 1	81.07	87.80	84.33	88.75	85.34	87.75	79.81	84.98
	Round 2	<b>81.42</b>	<b>88.18</b>	<b>84.53</b>	88.67	<b>85.44</b>	87.53	79.70	85.07
	Round 3	81.21	87.93	84.41	<b>88.83</b>	85.36	<b>87.87</b>	<b>80.21</b>	<b>85.12</b>

Table 3: Results of each round on seven STS datasets.



# Experiments



- Result

- Ablation Study 1: Accuracy of Synthetic Data

- The accuracy of the synthetic positive and negative samples improves over multiple rounds of adaptive training.

Round	Pos. Acc.	Neg. Acc.	Avg. Acc.
Round 0	85.96%	78.72%	82.34%
Round 1	92.89%	95.67%	94.27%
Round 2	<b>93.54%</b>	96.52%	95.03%
Round 3	93.51%	<b>96.68%</b>	<b>95.09%</b>

Table 4: Result of synthetic sample accuracy, based on RoBERTa-large sentence encoder.





# Experiments

- Result

- Ablation Study 2: Difficulty of Synthetic Data

- Our proposed model produces harder samples which better enhances the capability of semantic similarity modeling.

Syn Data	Wizard	SynCSE	GenSE	Ours
$x, x^+$ <b>cos</b> ↓	0.94	0.91	0.85	<b>0.84</b>
$x, x^-$ <b>cos</b> ↑	0.57	0.60	0.51	<b>0.83</b>

Table 5: Result of average synthetic sample difficulty, ours from round 3, calculated by supervised SimCSE RoBERTa-large, Wizard stands for WizardLM.



# Experiments



## • Result

### • Case Study

- Our method uses more hard positive samples and hard negative samples for training, so we can better identify hard positive samples and hard negative samples in practice.
- Several trends emerge, on lexically and syntactically.

Round	Positive Sample	Negative Sample
	<b>Anchor:</b> <i>A girl is sitting on the side of a mountain.</i>	
Round 1	Aside the mountain, a girl is sitting.	A girl is standing on the side of a mountain.
Round 2	She sat on the side of the mountain, a girl.	A boy is sitting on the side of a mountain.
Round 3	The slope of the mountain is where a girl is seated.	Boy is sitting on the side of a mountain.
	<b>Anchor:</b> <i>A man is kneeling down and using a paintbrush.</i>	
Round 1	Painting with a brush, a man knelt down.	A man is standing up and using a paintbrush.
Round 2	He is painting with a brush while standing on his knees.	A paintbrush is kneeling down and using a woman.
Round 3	A man is on bended knee, daubing with his brush.	A woman is kneeling down and using a paintbrush.

Table 6: Comparison of different data synthesis results at different rounds for RoBERT-large





# Outline

- Background
- Motivation of Our Work
- Our Framework
- Experiments
- **Conclusions**



# Conclusions



## • Conclusions

- In this paper, we proposed a cost-effective strategy to utilize small-parameter LLMs to generate synthetic data and enhance sentence representation models.
- Specifically, we propose a novel adaptive reinforcement tuning (ART) framework to optimize small-parameter LLMs in a reinforcement learning manner to adaptively generate hard contrastive samples with multiple iterations. These samples are then used to enhance the performance of existing sentence representation models.
- Experiments conducted on seven semantic text similarity tasks demonstrate that the sentence representation models trained using the synthetic data generated by our proposed method achieve state-of-the-art performance. We also conducted ablation studies to showcase the critical roles played by the reinforcement learning approach and the adaptive iterative framework within our proposed framework.

