Instruction Tuning With Loss Over Instructions

Zhengxiang Shi[1], Adam X. Yang[2], Bin Wu[1], Laurence Aitchison[2], Emine Yimaz[1], Aldo Lipani[1]

[1]University College London, [2]University of Bristol

NEURAL INFORMATION PROCESSING SYSTEMS

Paper Link Github

Takeaways

- If the number of training data is limited and completions are short, including the prompt loss during instruction tuning might be advantageous on various NLP and open-ended generation tasks.
- We identify two scenarios where including the prompt loss is particularly useful: (1) The ratio between instruction length and output length in the training data is high; and (2) The number of training examples is limited.
- The improvement stems from reducing the tendency to overfit, particularly under limited training resource conditions: Instruction tuning on brief outputs or a small amount of data can potentially lead to rapid overfitting.

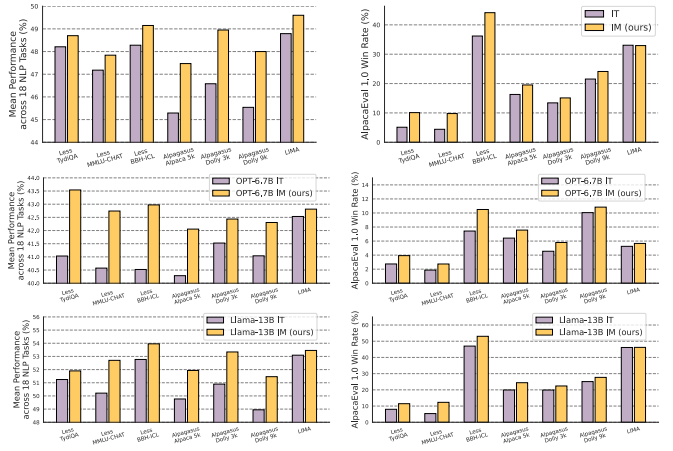
Limitations

- **The success of our approach relies on the quality and diversity of the instructions and prompts in the training datasets.**
- **It is crucial to ensure that the instructions are ethically sound and free from harmful or biased content. Training on inappropriate or toxic instructions may result in undesirable outputs.**

Main Experiments

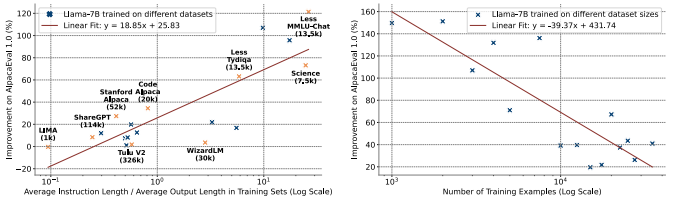
Main experiment 1: Performance differences between INSTRUCTION TUNING (IT, without the prompt loss) and INSTRUCTION MODELLING (IM, with the prompt loss) on 7 datasets.

Our findings: In many scenarios, IM can effectively improve the model performance on both NLP tasks (*e.g.,* MMLU, TruthfulQA, and HumanEval) and open-ended generation benchmarks (*e.g.,* MT-Bench and AlpacaEval).



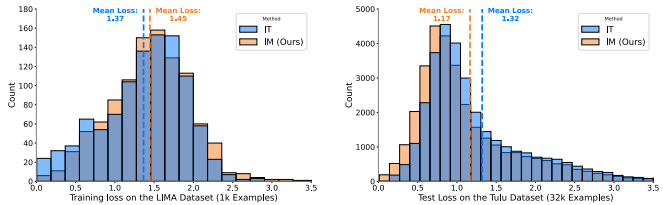
Main experiment 2: Performance improvement, achieved by our approach INSTRUCTION MODELLING (IM) compared to INSTRUCTION TUNING (IT) on the AlpacaEval 1.0.

Our findings: We identify two key factors influencing the effectiveness of IM: (1) The ratio between instruction length and output length in the training data; and (2) The number of training examples.



Additional Experiments

Additional Experiment 1: Train and test loss analysis.
Our findings: IM has a higher train loss with lower test loss, suggesting that IM effectively mitigates the overfitting issues compared to IT.



Additional Experiment 2: Average BLEU Score comparison.
Our findings: IM produces outputs have less overlap with the ground truth outputs in training examples, indicating less overfitting.

	LIMA	Less Tydiqa	Less MMLU Chat	Less BBH ICL	Alpagasus Alpaca 5k	Alpagasus Dolly 9k	Alpagasus Dolly 3k
IT	18.15	69.21	72.43	60.96	72.26	61.76	60.99
IM (ours)	17.30↓0.85	65.63↓3.58	69.20↓3.23	53.94↓7.02	70.50↓1.76	60.61↓1.15	59.04↓1.95

Additional Experiment 3: Can KL divergence loss, as regularization, easily address overfitting?
Our findings: (1) Incorporating KL Loss reduces overfitting and reduces the performance degradation on traditional NLP tasks; (2) KL Loss detrimentally affects model performance on open-ended generation tasks.

		LIMA (1K)		ALPAGASUS DOLLY (9K)	
	LLAMA-2-7B-BASE	IT w/o KL Loss	IT w/ KL Loss	IT w/o KL Loss	IT w/ KL Loss
NLP Tasks	49.32	48.79↓0.53	49.26↓0.06	45.54↓3.78	49.31↓0.01
AlpacaEval 2.0	0.01	2.58↑2.57	0.06↑0.05	2.28↑2.27	0.04↑0.03

Additional Experiment 4: Performance comparison of IM and IM +NEFTUNE on AlpacaEval 1.0 and various NLP benchmarks.
Our findings: Our proposed method IM could further improve the model performance with NEFTUNE.

	LIMA	Less Tydiqa	Less MMLU Chat	Less BBH ICL	Alpagasus Alpaca 5k	Alpagasus Dolly 9k	Alpagasus Dolly 3k
AlpacaEval 1.0 Win Rate							
IM	32.94	10.10	9.78	44.15	19.52	30.77	15.11
IM +NEFTUNE	30.77↓2.17	23.41↑13.31	12.45↑2.67	48.25↑4.10	32.07↑12.55	38.28↑7.51	23.35↑8.24
Mean Performance Across 18 NLP Tasks							
IM	49.60	48.70	47.84	49.15	47.47	48.00	48.95
IM +NEFTUNE	49.47↓0.13	49.44↑0.74	47.73↓0.11	48.62↓0.53	48.70↑1.23	48.63↑0.63	49.54↑0.59