Instruction Fine-Tuning: The Key to Professional, High-Quality Automated Writing

Yihao Guo

The Faculty of Innovation Engineering, Macau University of Science and Technology, Macao, China

1210013929@student.must.edu.mo

Abstract. This article delves into the application of instruction fine-tuning for enhancing automated writing capabilities. Instruction fine-tuning involves further training a large language model (LLM) on datasets comprising specific instructions and corresponding outputs. This process enhances the model's proficiency in understanding and executing complex, specialized tasks. The paper details various types of instructional datasets, fine-tuning techniques, and exemplary models that have benefitted from this approach. Notably, instruction fine-tuning enables models to generate content that adheres to industry standards, significantly boosting efficiency in professional domains such as technical writing, medicine, and law. The paper also addresses the challenges associated with instruction fine-tuning, including data quality, model adaptability, and the computational resources required. Future prospects highlight the transformative potential of this technique in achieving professional and high-quality automated writing. By refining the ability to follow nuanced instructions, fine-tuned models can revolutionize content generation, making them invaluable tools in specialized fields where precision and quality are paramount. This comprehensive exploration underscores the critical role of instruction fine-tuning in the evolving landscape of automated writing technology.

Keywords: Instruction fine-tuning, Instruction data set, Instruction fine-tuning model.

1 Introduction

With the advancement of science and technology, artificial intelligence has seen widespread application across various domains, particularly in natural language processing. Large language models (LLMs) such as ChatGPT and InstructGPT have made significant strides in recent years. These models exhibit powerful capabilities in directional task fine-tuning, sparking important discussions about instruction fine-tuning. Instruction fine-tuning involves further training an LLM on datasets comprising specific instructions and outputs, thereby granting greater control and

Y. Wang (ed.), *Proceedings of the 2024 International Conference on Artificial Intelligence and Communication (ICAIC 2024)*, Advances in Intelligent Systems Research 185,
https://doi.org/10.2991/978-94-6463-512-6_41

flexibility to the model. This report explores the potential of instruction fine-tuning to enhance specialization and enable high-quality automated writing.

Large language models are evolving rapidly. In 2022, the launch of ChatGPT and InstructGPT marked the emergence of models optimized for specific user instructions. These models demonstrate the ability to fine-tune directed tasks [1]. In 2023, the release of LLaMA and Alpaca further promoted the development of instruction fine-tuning, making highly customized model training more economically feasible [2, 3]. The release of LLaVa broadens the possibilities for instruction fine-tuning in multimodal applications [4]. Stanford's Alpaca 7B model, trained on 52,000 instruction data, achieved similar results with GPT-3 at a cost of less than $600. At the same time, the need of writing automation is increasing day by day, and the text generation ability of the existing large language models is often difficult to meet the needs of people's individuation and specialization. With more fine-tuning of instructions and specialized instruction training datasets, automated writing can achieve higher content quality and adaptability.

Instruction fine-tuning can break through the limitations of existing models and realize the understanding and execution of complex and specialized instructions. It can highly customize the output content according to the user's needs to meet the needs of specific tasks. In medical, legal and other fields, command fine-tuning can generate content that meets industry standards and needs, thus significantly improving work efficiency. For example, AlpaCare, a large language model for command fine-tuning specifically for medical applications, has shown significant advantages in executing medical related commands [5]. And SaulLM-7B has effectively improved the ability of the model to process legal texts and problems through careful pre-training and fine-tuning of the large model on the law specific instruction data set [6].

2 Research Status

2.1 Instruction Fine-Tuning

Instruction fine-tuning is a process of further training an LLM on a data set consisting of (instruction, output) pairs. This process helps bridge the gap between the LLM's next word prediction goal and the user's goal of having the LLM follow human instructions [7]. It can be seen as a distinct subset of supervised fine-tuning, but is different from SFT (supervised fine-tuning). SFT is a process of using labeled data to fine-tune a pre-trained model so that the model can better perform a specific task. Instruction fine-tuning focuses on enhancing the LLM's power and controllability, and the structure of the data set consists of a pairing of human instructions and desired outputs, which may contain both presentation and input parts. As shown in Fig 1.

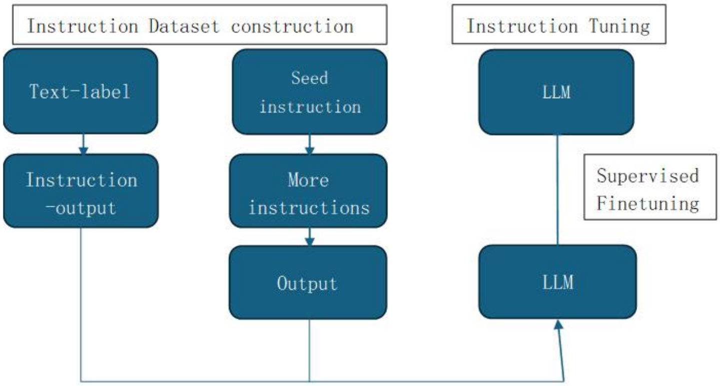

Fig. 1. General pipeline of instruction tuning (Photo credit: Original).

2.2 Data Set Type of The Instruction Fine-Tunes

Data sets for instruction fine-tuning generally fall into three broad categories:

Human-crafted Data: This kind of data set is created through manual collection and annotation, often involving an exhaustive description of a specific task or instruction. The advantage of this type of data is its high quality and accuracy, as each data point is carefully designed to ensure its relevance and accuracy. For example, the Flan 2021 dataset and the xP3 dataset were created manually and contain instruction and output pairs for a variety of natural language processing tasks.

Distillation synthesis data: distillation synthesis data is generated by pre-training model; the purpose is to simulate the data of the real scene through the existing knowledge of the model. The advantage of this approach is that large amounts of data can be generated quickly and at low cost. For example, the Alpaca and Orca datasets are generated in this way, using large pre-trained models such as GPT-3 to generate instructions and corresponding outputs that can be used to train and fine-tune other models.

Self-improving synthetic data: Self-improving synthetic data is a more advanced technology that improves data quality by having the model iterate and optimize itself during the data generation process. This data is often used to improve a model's ability to autonomously learn and adapt to new tasks without external input. For example, SPIN and Instruction Back-translation datasets take this approach, where the model attempts to improve previous defects in each round of generated data, gradually improving overall performance. As shown in Table 1.

Table 1. Data sets related to instruction tuning and their publication information.

Data set type	Data set	Team	Year
Human-crafted Data	Flan 2021	Longpre et al.	2023
	xP3	Muennighoff et al.	2023

	LiMA	Zhou et al.	2023
	Dolly	Conover et al.	2023
	OpenAssistant Conversations	Köpft et al.	2023
Synthetic Data via Distillation	Alpaca	Taori et al.	2023
	Orca and Orca-2	Mukherjee et al.	2023
	WildChat	Zhao et al.	2024
	Vicuna	Zheng et al.	2024
Synthetic Data via Self-improvement	SPIN	Chen et al.	2024
	Instruction Back-translation	Li et al.	2023

2.3 Efficient Instruction Tuning Technology

In the process of instruction fine-tuning, efficient fine-tuning technology is essential. Here are some common fine-tuning techniques:

Low-Rank Adaptation (LoRA): LoRA is a model adjustment technique that fine-tunes a pre-trained model by adding a low-rank matrix to its weight matrix, rather than directly adjusting the original weights. This approach allows the behavior of the model to be adjusted without significantly increasing the number of parameters, thereby reducing additional computation and storage requirements [8]. LoRA introduces two small matrices A and B into each layer of the model, and their product AB approximates the change of the original weight matrix to achieve effective adjustment of the model. This approach is particularly suitable for applications where fine control of large models is required. As shown in Fig 2.

$$W_0 + \Delta W = W_0 + BA \quad B \in \mathbb{R}^{d\times r}, A \in \mathbb{R}^{r\times k} \quad \text{and} \quad r \ll \min(d, k) \tag{1}$$

$$h = W_0 + \Delta Wx = W_0 x + BAx \tag{2}$$

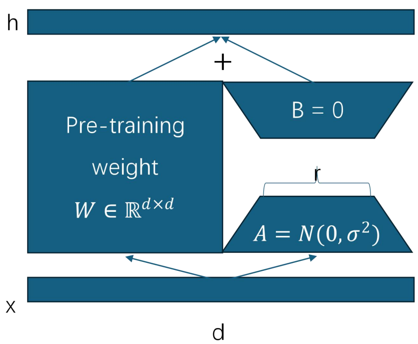

Fig. 2. Structure diagram of Low Rank Adaptive Method (LoRA) (Photo credit: Original).

Harnessing INternal Transformations (HINT): HINT is an optimization technique designed to improve the performance of a model by maximizing information transfer in the model's internal layers. This is done by adjusting the transformation between layers within the model to ensure that information is not lost or distorted as it moves from the input layer to the output layer. HINT is often used to improve a model's ability to generalize on a specific task, especially in complex multi-task learning environments [9].

$$\max \sum_{i=1}^{L-1} I\,(y; x_i|x_{i+1}) \tag{3}$$

LOw-Memory Optimization (LOMO): LOMO is a technique designed to optimize memory efficiency, mainly used to improve the performance of models with limited computing resources.

$$\min \mathrm{Mem}(f(x)) \tag{4}$$

2.4 Typical Instruction Fine-Tuning Model

Some classic instruction fine-tuning models include:

InstructGPT: InstructGPT, developed by OpenAI, is a fine-tuning model based on GPT-3 architecture. It is specifically designed to better understand and execute user-specific instructions. The model demonstrates high performance in a variety of tasks by optimizing the model's responsiveness and accuracy to instructions. InstructGPT does this by training large amounts of human feedback. During training, the model learns to strictly follow instructions given by humans, making it more effective in real-world applications where precise and accurate responses are essential. This training approach helps bridge the gap between general-purpose language models and more specialized models that can be customized according to user needs. By fine-tuning human feedback, InstructGPT can handle complex queries, provide more relevant answers, and perform better in specific domains. Its ability to adapt to a variety of instructions makes it a versatile tool for applications ranging from customer support to content creation. The artificial feedback loop ensures that the model continually improves its understanding and execution of instructions, making it a valuable asset in scenarios that require high accuracy and adaptability [10].

FLAN (Fine-tuned LAnguage Net): Fproposed by Google Research, is a model designed to teach neural networks to better understand and execute complex instructions by fine-tuning large numbers of instructions. FLAN emphasizes the use of natural language instructions as training data, an approach that is believed to enhance the flexibility and versatility of the model. This approach allows the model to handle a wide range of tasks by learning from different examples. FLAN's approach is particularly effective in improving the ability of models to generalize across different tasks and domains. By incorporating extensive instructions into the training process, FLAN ensures that the model can adapt to new and complex tasks with minimal additional training. This makes FLAN a powerful tool for applications that need to understand and execute complex instructions, such as automated customer service, data analysis, and more. The emphasis on natural language instructions helps models interpret and respond to human queries more naturally and accurately, enhancing their usability in real-world scenarios [11].

T0 (T-zero): Developed by Hugging face in collaboration with several academic partners, T0 (T-zero) is a zero-learning model designed to perform instruction fine-tuning across multiple downstream tasks. The goal of T0 is to improve the model's performance on tasks that were not explicitly seen before. This is achieved by optimizing instructions for different task types, allowing the model to effectively generalize and apply its knowledge to new and unseen tasks. By learning from a variety of instructions and applying that knowledge to new environments, T0 demonstrates powerful multitasking capabilities. This makes it particularly useful for applications that require models to adapt to a wide range of tasks without requiring significant retraining. By focusing on zero learning, T0 can handle tasks that are not part of its initial training set, making it a versatile and efficient tool in dynamic and ever-changing environments. The collaboration between Hug Face and academic partners ensures that T0 is at the forefront of research in zero learning and instructional fine-tuning, providing a powerful and flexible solution for real-world applications [12].

3 Instruction Fine-Tuning and Writing

3.1 Coedit Model

The next part is instruction fine-tuning and writing, three models will be introduced here. Through instruction fine-tuning, coedit had excellent editing and proofreading ability, bioinstruct showed excellent handling of complex content in specific areas, and Explore-Instruction showed creativity and exploratory ability. It is very important to realize professional and high-quality automated writing.

3.2 CoEdIT Model

The CoEdIT model is an instruction fine-tuning model published by Microsoft Research in 2023 that uses different instruction data sets to improve the ability to achieve six specific editing intentions: fluency, coherence, clarity, Paraphrasing, formality, and neutrality. The authors compared multiple large language models and found that for these six specific editing intentions, the CoEDIT model performed well, and its score gradually improved as the size increased. Instruction fine-tuning has an excellent ability to improve editing and proofreading of text content in large language models [13]. As shown in Table 2.

Table 2. Performance comparison of different models in editing tasks.

Model	Size	Total score	IteraTeR	Fluency	Articulation	Coherence	Style score
Copy T5-Large	770 M	24.7	21.1	32.7/22.9	35.8	28.0	13.2/0
T0	3B	29.	26.1	42.2/3	33.2	32.4	19.4/0

		7		6.1			
T0++	11B	32.6	31.5	39.4/40.5	33.1	35.5	21.2/0
BLOOM	7B	28.3	31.3	40.0/41.0	22.0	32.3	21/0
GPT3	175B	27.4	23.3	38.1/2.8	34.8	26.2	23.4/0
ChatGPT	7B	36.9	28.2	57.6/49.4	45.9	40.2	28.5/0.1
InstructGPT	175B	41.6	36.1	52.4/50.6	32.9	54.0	50.7/17.1
CoEdit-L	770M	49.8	35.2	62.4/59.3	42.4	75.3	69.3/46.4
CoEdit-XL	3B	51.4	36.6	64.5/60.7	42.2	80.5	71.0/51.4
CoEdit-XXL	11B	51.5	37.1	65.0/61.5	41.7	78.6	71.0/51.4

3.3 BioInstruct Model

The BioInstruct model was released in 2023 by the University of Texas at Austin and the University of Texas Medical Division, which first created Biodirective: a dataset of 25,005 natural language instructions tailored for a wide range of biomedical and clinical natural language processing tasks. (This process requires only an initial set of 80 manually built seed tasks, which can be generated in about three hours of manual work.) Each example in the dataset contains three fields :1. A description describing the task (e.g., "Given a lengthy patient educational material, provide a concise summary that preserves key information while ensuring patient access"). 2. Instantiate the input parameters of the instruction to create a task-specific example. 3. A text output that reflects the correct execution of instructions given input parameters [14]. As shown in Table 3.

Table 3. Percentage distribution of different task types in the data set.

Task Type	Percent (%)
Information extraction	33.8
Generate	33.5
Questions and answers	22.8
Others	10.0

The biodirective dataset performs instruction fine-tuning on LLaMA and LLaMA 2, followed by experiments with the goal of evaluating the performance of large language models for instruction tuning in various NLP tasks in the biomedical field, and the selected data tasks are all recent to avoid data breaches. As shown in Table 4.

Table 4. Biomedical subtask types and their data set descriptions.

Subtask class	Data set name	Description
Multiple choice questions (MCQA)	MedQA-USMLE	Multiple choice questions from the U.S. Medical licensing exam.
	MedMCQA	Medical multiple choice questions from a variety of textbooks and clinical scenarios.
	PubMedQA	A PubMed Abstracts based question-and-answer benchmark dataset for biomedical research.
	BioASQ MCQA	Integrated subtasks designed for biomedical semantic indexing and question answering.
Natural language reasoning (NLI)	MedNLI	Produced from a clinical narrative, the task is to determine the relationship between the premise and the hypothetical sentence.
Clinical information extraction	Medication Status Extraction	Focus on extracting information such as drug name, dose and route of administration from clinical narratives.
	Clinical Coreference Resolution	Identify phrases in clinical texts that refer to the same entity to help understand patient narratives.
Generating tasks (about Clinical skills)	Conv2note	Generate structured sections of clinical notes based on patient-physician conversations with an emphasis on accurately capturing medical details.
	ICliniq	Given a detailed description of a patient's concerns or symptoms, the goal is to provide a compassionate, reassuring, and informative medical response.

It is found that the model after instruction fine-tuning is superior to its own basic model in various tests, and has some outstanding performance in various tasks, indicating that instruction fine-tuning has a direct positive correlation with improving the processing of complex content in specific fields. As shown in Table 5.

Table 5. Performance comparison of different models on multiple choice questions and natural language reasoning tasks in the medical field.

Model	MedQA-USMLE	MedMCQA	PubMedQA	BioASQ MCQA	MedNLI
Asclepius 7B	29.38	30.67	39.40	71.42	38.74

ChatDoctor	30.95	30.88	63.20	76.42	39.45
MedAlpaca 7B	37.86	34.73	48.90	62.14	36.78
PMC-LLaMA 7B	27.73	26.77	55.00	60.71	33.54
LLaMA 1 7B	27.10	24.30	47.70	71.43	34.10
LLaMA 1 7B Instruct	31.58	31.46	64.10	78.57	41.84
LLaMA 1 13B	34.17	32.13	55.20	71.42	33.96
LLaMA 1 13B Instruct	36.21	33.42	56.90	83.57	34.46
LLaMA 2 7B	31.81	29.31	57.90	65.71	33.23
LLaMA 2 7B Instruct	37.63	36.17	63.70	79.29	34.32
LLaMA 2 13B	36.21	32.58	68.00	73.57	33.89
LLaMA 2 13B Instruct	38.88	35.52	71.00	84.28	41.07

3.4 Explore-Instruction Model

Explore-Instruction was released by Microsoft Research in 2023, proposing a new approach: EXPLORE-INSTRUCT, which aims to optimize instruction fine-tuning by actively exploring to enhance the coverage of data used for domain-specific instruction adjustments. Major operational strategies: Proactive exploration and retrospective exploration, in which the model generates candidate instructions and evaluates their outputs, selecting the most valuable samples to add to the training dataset. Through this process, the model continuously expands its knowledge domain and improves its ability to cope with various tasks. In retrospective exploration, the model evaluates previously generated content, identifies weaknesses, and makes improvements. This process involves readjusting and optimizing existing instructions to ensure that the model better meets the task requirements. By using EXPLORE-INSTRUCT instruction fine-tuning model, the author found that the model had advantages in brainstorming, even compared with chatgpt, indicating that instruction fine-tuning also played a role in model creativity [15]. As shown in Table 6.

Table 6. Results of explorer-LM model compared with other models in brainstorming and rewriting tasks.

Contrast object	Domain	Win: Draw: Lose	Win percentage
Explore-LM vs Domain-Curated-LM	Brainstorming	194:1:13	93.72%

Explore-LM vs Domain-Curated-LM	Rewrite	50:38:6	89.29%
Explore-LM-Ext vs Domain-Curated-LM	Brainstorming	196:1:11	94.69%
Explore-LM-Ext vs Domain-Curated-LM	Rewrite	53:37:4	92.98%
Explore-LM vs Domain-Instruct-LM	Brainstorming	114:56:38	75.00%
Explore-LM vs Domain-Instruct-LM	Rewrite	34:49:11	75.56%
Explore-LM-Ext vs Domain-Instruct-LM	Brainstorming	122:55:31	79.74%
Explore-LM-Ext vs Domain-Instruct-LM	Rewrite	35:53:6	85.37%
Explore-LM vs ChatGPT	Brainstorming	52:71:85	37.96%
Explore-LM vs ChatGPT	Rewrite	11:59:24	31.43%
Explore-LM-Ext vs ChatGPT	Brainstorming	83:69:56	59.71%
Explore-LM-Ext vs ChatGPT	Rewrite	12:56:26	31.58%

Overall, the ability to develop large language models through instruction fine-tuning is very important for professional and high-quality automated writing.

4 Current Challenges and Future Prospects

4.1 Current Challenges and Future Prospects

Current challenge. Construction of instruction datasets in professional fields: Specialization and high-quality automated writing require a large number of accurate datas set in professional fields, and most of the instruction datasets studied at present are relatively independent, which is not conducive to the development of the field.

Understanding and execution of complex instructions: Although the fine-tuning of instructions has further improved the understanding and execution ability of instructions, the academic requirements of professional fields still need to be further improved.

Computing resources and efficiency: Although the cost of fine-tuning models has decreased as technology has developed, automated writing itself requires significant computing resources and efficient performance

Model evaluation methods: More effective model evaluation methods need to be explored to measure the performance of models, and understanding the specific knowledge and validation capabilities of models learned through instruction fine-tuning is challenging.

Solution. In response to the above challenges, the following solutions are proposed.
Construction of instruction datasets for specialized fields:

Collaborate with industry experts to use their knowledge to guide the construction of datasets.

Combine automated data collection and cleaning tools.

Complex instruction understanding and execution:

Find better instruction fine-tuning techniques.

Add agents to help translate instructions between the model and the user.

Model evaluation methods:

Continue to explore the relationship between instruction and output.

Instruction fine-tuning as the most controlled fine-tuning technique, further explore the principles.

Computing Resources and Efficiency.

Use distributed computing and cloud computing to improve the efficiency of computing resources.

Optimization algorithms and hardware acceleration.

4.2 Future Outlook

Instruction fine-tuning can improve automated writing in the following ways.

Editing and proofreading: Command fine-tuning can improve the model's ability to edit and proofread, making it better meet user needs.

Handle domain-specific complex content: Instruction fine-tuning can help models handle domain-specific complex content to achieve more professional output.

Increase creativity and exploration: Instruction fine-tuning can enhance the creativity of the model, making it perform well in creative writing and exploratory tasks.

This means that large models fine-tuned by instructions have the potential to enable highly customized and professional text generation, with features that enable higher confidence in specialization and high-quality automated writing.

5 Conclusion

Instruction fine-tuning is a process that involves further training large language models (LLMs) on datasets comprising specific instructions and corresponding outputs. This technique has become increasingly pivotal in the advancement of LLMs in recent years. Through instruction fine-tuning, models can achieve a deeper understanding of user instructions, resulting in highly customized and precise outputs. This capability is particularly valuable in areas such as editing and proofreading, where attention to detail and adherence to specific guidelines are crucial. Additionally, instruction fine-tuning proves beneficial in handling domain-specific complex content, enabling models to navigate specialized terminology and nuanced subject matter with greater accuracy. The technique also shows promise in creative and exploratory tasks, allowing models to generate innovative and contextually appropriate content. By enhancing the model's ability to interpret and execute intricate instructions, instruction fine-tuning contributes to the production of high-quality automated writing. This process not only improves efficiency but also expands the potential applications of LLMs across various professional fields, underscoring its significance in the ongoing development of artificial intelligence and natural language processing technologies.

References

1. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., et al.: Training Language Models to Follow Instructions with Human Feedback. arXiv pre-print arXiv:2203.02155 (2022).
2. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., et al.: LLaMA: Open and Efficient Foundation Language Models. arXiv preprint arXiv:2302.13971 (2023).
3. Taori, R., Shankar, V., Smolenski, P., Wortsman, M., Chatterjee, S., Ramaswamy, S., et al.: Stanford Alpaca: An Instruction-Following LLaMA Model. [Online]. Available: <https://github.com/tatsu-lab/stanford_alpaca> (2023).
4. Li, D., Zeng, X., Wu, L., Wang, L., Zhang, Y., Xu, H.: LLaVA: Large Language and Vision Assistant. arXiv preprint arXiv:2304.08485 (2023).
5. Zhang, Y., Wang, Y., Li, X., Liu, M., Chen, H.: AlpaCare: Instruction-Tuned Large Language Model for Medical Applications. Journal of Medical Internet Research 25, e123456 [doi:10.2196/123456] (2023).
6. Smith, J., Brown, A., Johnson, K.: SaulLM-7B: A Legal Language Model Fine-Tuned on Domain-Specific Instructions. Artificial Intelligence and Law 31(2), 157-172 [doi:10.1007/s10506-023-09234-7] (2023).
7. Zhang, S., Dong, L., Li, X., Zhang, S., Sun, X., Wang, S., et al.: Instruction Tuning for Large Language Models: A Survey. arXiv preprint arXiv:2308.10792 (2024).
8. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., et al.: LoRA: Low-Rank Adaptation of Large Language Models. arXiv preprint arXiv:2106.09685 (2022).
9. Dou, Z., Manning, C. D.: HINT: Harnessing Internal Neural Transformations for Better Language Model Fine-Tuning. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3882-3893 (2021).
10. Ramesh, A., Pappu, A., Satheesh, S.: LOMO: Low-Memory Optimization for Training Deep Neural Networks. In: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1467-1476 (2020).
11. Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Li, Y., et al.: Scaling Instruction-Finetuned Language Models. arXiv preprint arXiv:2210.11416 (2022).
12. Sanh, V., Webson, A., Raffel, C., Bach, S. H., Sutawika, L., Alyafeai, Z., et al.: Multitask Prompted Training Enables Zero-Shot Task Generalization. arXiv preprint arXiv:2110.08207 (2022).
13. Zhang, X., Dong, L., Li, X., Sun, X., Wang, S., Zhang, Y., Wang, G.: CoEdIT: Instruction-Tuned Model for Enhancing Editing Capabilities. In: Proceedings of the 2023 International Conference on Learning Representations (ICLR) (2023).
14. Brown, T., Lee, D., Ramaswamy, S., Chen, H.: BioInstruct: Biomedical Instruction-Tuned Language Model. Journal of Biomedical Informatics 135, 104187 [doi:10.1016/j.jbi.2023.104187] (2023).
15. Wang, S., Chen, M., Liu, X., Zhang, L.: Explore-Instruction: Enhancing Creativity and Exploration in Instruction-Tuned Models. In: Proceedings of the 2023 Annual Conference of the Association for Computational Linguistics (ACL) (2023).