Editor	1 edit		10 edits		100 edits	
	zsRE	COUNTERFACT	zsRE	COUNTERFACT	zsRE	COUNTERFACT
FT-c	0.57(s)	0.54(s)	6.58(s)	6.82(s)	21.77(s)	19.30(s)
ROME	20.47(s)	18.27(s)	207.09(s)	179.30(s)	2184.16(s)	1810.42(s)
MEMIT	28.32(s)	23.71(s)	108.67(s)	96.71(s)	862.20(s)	847.72(s)
Full-FT	0.76(s)	0.72(s)	7.8(s)	7.3(s)	25.36(s)	24.70(s)
F-Learning$_{\mathrm{FT}}$	**1.58(s)**	**1.47(s)**	**15.32(s)**	**14.9(s)**	**52.20(s)**	**50.12(s)**

Table 3: Editing time for 1 edit, 10 edits, 100 edits of the two dataset based on LLAMA2-7B.

5.7 Time Testing

In order to evaluate the efficiency of our proposed F-learning method, we calculated the editing time of several different knowledge updating and model editing methods for different numbers of edits. Taking LLAMA2-7B as an example. The results are shown in the Table 3.

We can find that the time consumed by the fine-tuning based method is significantly less than that of the locate-based method. This is because the locate-based method highly relies on the location of neurons and parameters, which increases the complexity and time of editing. Furthermore, since ROME can only edit a single piece of data at a time, while other methods can edit in batches, ROME is less efficient. Compared with other fine-tuning based methods, FT-c can be optimized faster with its norm constraint. The F-learning we proposed is a two-stage knowledge updating method that forgets before learning, as it takes about twice as long as Full-FT, but is still very fast and convenient. It is worth noting that although the forgetting operation requires an additional training process, once the training is completed, the parameters of this part of forgetting old knowledge can be reused during the subsequent optimization and inference process, which can save resources and time. Meanwhile, we can further accelerate supervised fine-tuning by deepspeed or other approaches.

5.8 Parametric Analysis of Forgetting Old Knowledge

The two-stage knowledge updating method we proposed highly utilizes the forgetting of old knowledge. From the perspective of interpretability, here we analyzed the parameters of old knowledge forgetting and further analyzed the parameter distribution and changes within the LLMs. The results are shown in Figure 4 and Figure 5 which are in the appendix. Taking the LLAMA2-7B and zsRE dataset as an example, specifically, we analyzed different parameters in two cases: forgetting old knowledge by full fine-tuning and forgetting old knowledge by LoRA (The hyperparameters λ of the rate of forgetting are both set to 1). We compare the parameter θ' of the model $f_{\theta'}$ after forgetting the old knowledge with the parameter θ of the original model f_θ, and calculate their Euclidean distance on each layer. We select the results with layer n=[6, 15, 24, 30] to exhibit for simplicity. In general, there is little difference in parameter changes between low-layers and high-layers within the model. For forgetting old knowledge by full fine-tuning, we can find that the parameter changes of the MLP layers are more significant than attention layers. This may be one of the reasons why "forgetting before learning" is effective, as knowledge is generally stored in the MLP layers. Relatively LoRA has less impact on parameters (Euclidean distances are less than 1), and only changes the parameters of "query" and "value" in the attention layers. Therefore, it is more limited than full fine-tuning. However, LoRA-based forgetting can help forget the patterns and relationships associated with old knowledge stored in the attention layers, and thus can also assist in knowledge updating.

5.9 Experiments on the Old Knowledge Forgetting

Here we evaluate the model performance on old knowledge data after performing only the forgetting operation by full fine-tuning or LoRA fine-tuning with different forgetting rates to verify the effectiveness of the forgetting operation. The results are shown in Figure 6 and Figure 7 which are in the appendix. Taking the LLAMA2-7B and zsRE dataset as an example. We set the hyperparameters λ of the rate of forgetting is set to 0.9, 0.7, 0.5, 0.3, 0.1, respectively. It can be easily found that the performance of the model on three metrics is negatively correlated with the rate of forgetting, i.e., the old knowledge in the model decreases as