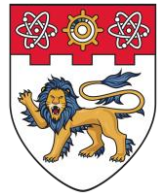
CLUE: CONFLICT-GUIDED LOCALIZATION FOR LLM UNLEARNING FRAMEWORK

Hang Chen, Jiaying Zhu, Xinyu Yang, Wenya Wang
Xi'an Jiaotong University, The Chinese University of HongKong,
Nanyang Technological University
Presenter: Hang Chen (albert2123@stu.xjtu.edu.cn)



Outline

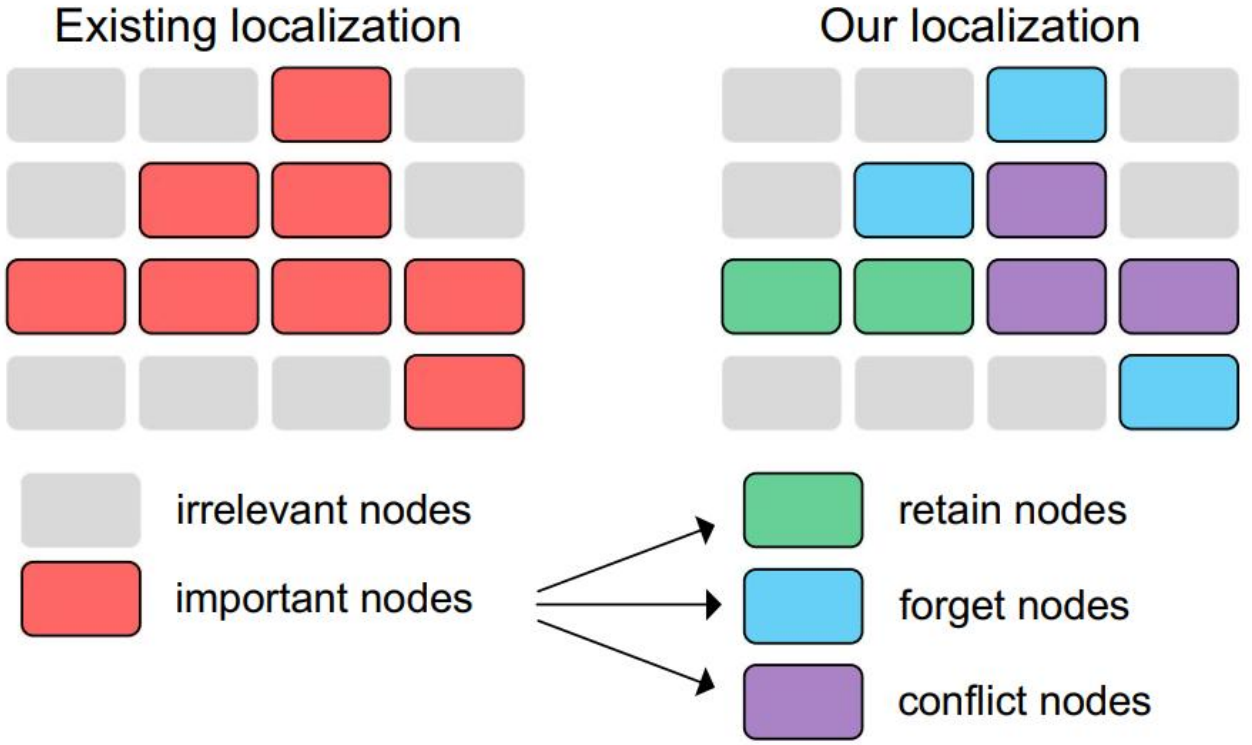
- Background & Motivation
- Preliminaries
- Method
- Experiment

Background & Motivation

- **LLM Unlearning**
  - eliminating the influence of specific ”unlearning targets” and removing associated model capabilities while preserving model performance for non-targets.
    - Forget set ⟶ removing target knowledge
    - Retain set ⟶ retaining non-target knowledge
    - $$\min_{\theta} \mathbb{E}_{(x,y_f)\in\mathcal{D}_f}[\mathcal{L}(y_f|x;\theta)] + \lambda\mathbb{E}_{(x,y)\in\mathcal{D}_r}[\mathcal{L}(y|x;\theta)]$$
  - Limitations：
    - Existing work only identifies two types of neurons：
      - Important neuron： its mechanism is related to forget set or retain set
      - Irrelevant neuron： its mechanism is not related to forget set or retain set

Motivation

- **Existing work**
  - **Important neuron**
  - **Irrelevant neuron**
- Ours
  - Retain neuron
  - Forget neuron
  - Conflict neuron
  - Irrelevant neuron



Preliminaries

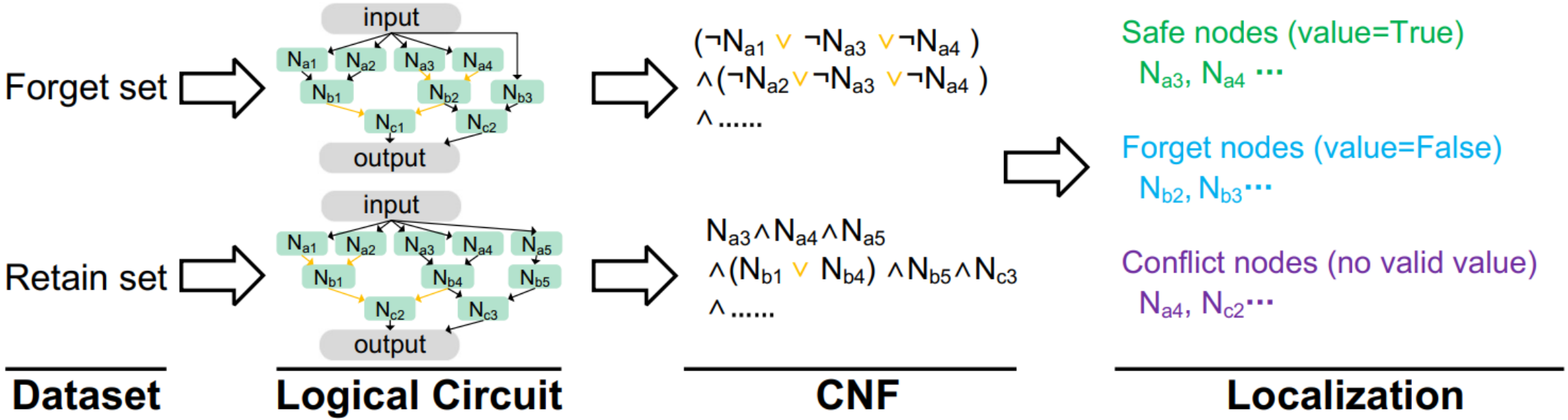
- Mechanistic Interpretability
  - Circuit discovery seeks to identify a minimal subgraph (circuit) $\mathcal{C} \subset \mathcal{G}$ that isolates the task-relevant causal mechanisms and capabilities for a target dataset $\mathcal{T}$.The optimization objective is formulated as:

$$\arg\min_{\mathcal{C}} \mathbb{E}_{(x)\in\mathcal{T}}[D(p_{\mathcal{G}}(y|x)||p_{\mathcal{C}}(y|x))], \quad s.t.\ 1 - |\mathcal{C}|/|\mathcal{G}| \geq s$$

  - The activation relationships and information flow between nodes in a circuit often exhibit strict logical structures.
    - **AND Gate**: The receiver node is activated if and only if all of its sender nodes are activatedsimultaneously.
    - **OR Gate**: The receiver node is activated as long as at least one of its sender nodes is activated.

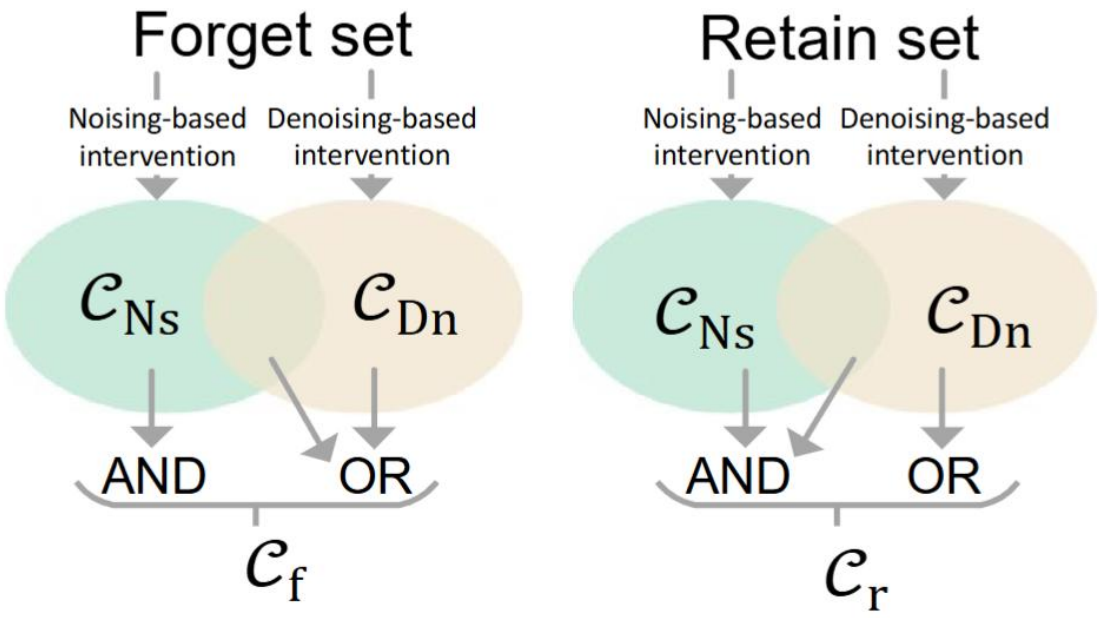
Method

- Overview



Method

- Step 1 Dataset to Circuit
  - Transform forget and retain data samples into logical circuits.
  - We extract two distinct subgraphs: the forget circuit $\mathcal{C}_f$ and the retain circuit $\mathcal{C}_r$ from their respective datasets. By applying noising and denoising interventions, we identify the most critical nodes and map their activation connections into logical AND and OR gates.



Method

- Step 2 Circuit to CNF
  - Convert the logical relationships within these circuits into Conjunctive Normal Form (CNF).
  - We apply Tseytin's transformation to convert the logical gates of $\mathcal{C}_f$ and $\mathcal{C}_r$ into strict CNF Boolean expressions (Φ_f and Φ_r). We construct a global CNF formula to enforce our unlearning objective—guaranteeing the forget circuit's functionality is disabled (False) while the retain circuit is perfectly preserved(True):

$$\Phi = \Phi_{\mathrm{f}} \wedge \Phi_{\mathrm{r}} \wedge (\neg \mathrm{output}_{\mathrm{f}}) \wedge (\mathrm{output}_{\mathrm{r}})$$

Method

- Step 3 Localization via SAT Solving
  - Solve the CNF satisfiability problem to strictly categorize each neuron's role.
  - We treat the unlearning localization as a boolean satisfiability problem by jointly solving the globalformula Φ.The optimal solution directly dictates the intervention strategy for each node:
    - Value = 1 (True) → Retain Nodes: Keeping them intact preserves non-target capabilities.
    - Value = 0 (False) → Forget Nodes: Removing them directly eliminates target knowledge.
    - Unsatisfiable → Conflict Nodes: Nodes lacking a valid assignment reveal an architectural bottleneck where forget and retain capabilities heavily overlap.

Method

- Step 4 Fine-Tuning
  - For forget neuron:

$$\min_{\theta_{\mathrm{f}}} \mathbb{E}_{(x,y_f)\in\mathcal{D}_f}[\mathcal{L}(y_f|x;\mathcal{M}_{\mathrm{f}}\odot\theta_{\mathrm{f}}+(1-\mathcal{M}_{\mathrm{f}})\odot\theta_{\mathrm{o}})]$$

  - For conflict neuron:

$$\min_{\theta_{\mathrm{c}}} \mathbb{E}_{(x,y_f)\in\mathcal{D}_f}[\mathcal{L}(y_f|x;\mathcal{M}_{\mathrm{f}}\odot\theta_{\mathrm{c}}+(1-\mathcal{M}_{\mathrm{f}})\odot\theta_{\mathrm{o}})]+\lambda\mathbb{E}_{(x,y)\in\mathcal{D}_r}[\mathcal{L}(y|x;\mathcal{M}_{\mathrm{f}}\odot\theta_{\mathrm{c}}+(1-\mathcal{M}_{\mathrm{f}})\odot\theta_{\mathrm{o}})]$$

Experiment

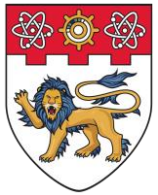
- a) RQ1: Can CLUE improve LLM unlearning through localization?
  - Yes. By precisely localizing forget and conflict nodes rather than applying uniform inter-ventions, CLUE consistently surpasses existing localization and fine-tuning methods in forget efficacy,retain utility, and general utility.

- b) RQ2: How does CLUE perform when a general corpus serves as the retain set?
  - Even when utilizing a multi-task general corpus (e.g., MMLU) or scaling up multiple specifictasks, CLUE maintains superior forget efficacy and effectively preserves general utility compared tobaselines like WAGLE.

Experiment

- a) RQ3: What is the relationship between unlearning performance and circuit sparsity?
  - Denser, more faithful circuits yield higher forget efficacy. Empirically, we find that theunlearning performance reaches its optimal range at a circuit sparsity level of 0.7.
- b) RQ4: How does node localization change after unlearning?
  - Post-unlearning, the proportion of forget nodes significantly increases while conflict nodesdecrease. This proves CLUE effectively decouples the targeted forget circuit from the mechanisms ofother preserved capabilities.

ICLR



- Thank you