

## A BASIC PROMPTS AND REASONING PIPELINE OF BOT

### A.1 THOUGHT GENERATION PART OF BOT

This subsection presents the basic prompts used by the reasoning generation of the proposed Boosting of Thoughts (BoT). For details, one can also access the source code *examples/BoostingOfThought/BoT\_reasoner.py*.

**System prompt  $S$ :** You are an expert on mathematical problems. Perform step-by-step reasoning toward problem solving by first learning from an ensemble of trial-and-error reasoning experiences. Such trial-and-error reasoning experience specifically contains error reports and detailed advice on how to revise historical reasoning steps. Always recall these listed experiences before generating a new reasoning step, thereby avoiding making the same mistakes and reusing correct steps to generate better reasoning steps to solve the task.

**Prompt for the next thought generation:**

```
f"""{task_prompt}. \n First of all , Recall historical reasoning experience: \n\n {
  experiences} \n\n Please make one step of reasoning to generate only one next
  possible reasoning step . This next reasoning step is the subsequential step from
  the following ordered previous steps , accompanied by their evaluated scores (A
  higher score means the reasoning step is more likely to complete the task .): \n\t{
  chain_prompt}\n\n Based on listed previous reasoning steps (ignore them when the
  above space is empty), generate one single next possible step following the Task
  rule . (Emphasize: Please generate only one single next possible reasoning step of
  the given steps .)"""
```

where the `task_prompt` contains the  $X$  and  $Q$  for the task, `experiences` is the historical reasoning experience  $\mathbf{F}^{1..t}$ , and `chain_prompt` is the  $\{G_i\}$ , which is a placeholder to be replaced by the preceding chain of thoughts  $z_{1..,i-1}$  of the current thought  $z_i$ .

**Prompt for the thought evaluation:**

```
f"""{task_prompt}. \n\n Below are the generated reasoning steps , presented in order ,
  accompanied by their evaluated scores (A higher score means the reasoning step is
  more likely to complete the task .):\n\t{chain_prompt}\n\t{thought}\n\n What is your
  evaluation score for the logic , correctness , and benefit to reaching a final
  solution for these reasoning steps ? Please select one value from [0.1, 0.3, 0.5,
  0.7, 0.9, 1.0] as the score , where a higher score means better reasoning steps . The
  score should be placed after 'Evaluation score:' for users to read ."""
```

where the `thought` is the current thought  $z_i$ .

### A.2 EXPERIENCE GENERATION PART OF BOT

To generate feedback for the aggregated chain, LLMs use the following basic prompts. For details, one can also access the source code *examples/BoostingOfThought/BoT\_commenter.py*

**System prompt  $S$ :** You are an expert AI checker for math answers, dedicated to evaluating the reasoning chain generated towards addressing the mathematical problem. Judge each reasoning step of this reasoning chain by providing detailed analyses on whether the current step is a logical inference of the previous step and whether the reasoning step is beneficial to the correct solution. Provide advice and suggestions for each reasoning step with errors. Provide recommendation or rejection descriptions for each correct reasoning step.

**Prompt for the feedback:**

```
f"""Given task:{task_prompt}.\n\t{chain_prompt}\n\n Please evaluate this reasoning chain
  by giving detailed comments containing the following content.\n 1.{
  chain_feedback_format } . 2.{ step_feedback_format } . 3.{ confidence_feedback_format }.\n
  \n Notice: {Notice}.
```