Detecting Toxic content from social media using LLMs

Group 3:
Dondapati Chandrahas Reddy (AI21BTECH11010),
Jupally Sriram (CS21BTECH11025)

Problem Statement

The objective of this work is to detect toxic social media content by utilizing language models.

One of the major issues plaguing social media is toxicity, whether it be hate speech, threats or insults. Due to this content moderation has become a crucial task of any social media platform. However due to the enormous scale, it cannot be done manually and must be automated by using computer algorithms. In this project, we aim to build a model that can accurately filter toxic content from social media.

Dataset - 1

The first dataset is from Learning from the Worst: Dynamically Generated Datasets to Improve Online Hate Detection (Vidgen et al., ACL-IJCNLP 2021).
The dataset contains 40,000+ entries generated and labelled by trained annotators using a human-and-model-in-the-loop process with 54% categorized as hate.

Examples:

- Women shouldn't be allowed to drive (hate)
- Muslims are a cancer to the world (hate)
- The government is full of incompetents (not hate)
- You couldn't be more stupid (not hate)

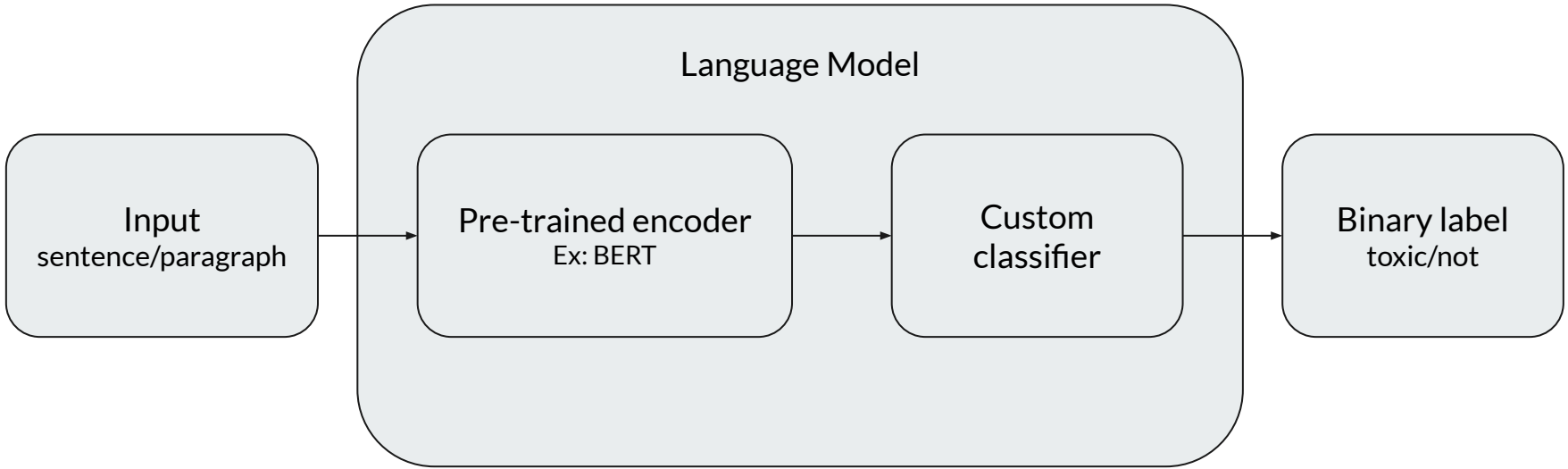
Dataset - 2

The second dataset (from kaggle) is a collection of labelled datasets from various social media platforms containing hate speech, aggression, insults and toxicity. The dataset contains ~330,000 entries of which ~44,000 (13%) can be classified as toxic

Examples:

- You seem a very strange and peculiar person. (not toxic)
- I didn't, you piece of garbage. (toxic)
- Yes it is, don't argue what u dont know. (not toxic)
- You are a loser. Stop being pathetic (toxic)

Proposed Method - 1



Proposed Method - 2 (BD-LLM)

Decision-Tree-of-Thought (DToT) to improve the zero-shot and few-shot in-context learning performance of LLMs as well as extract better rationales.

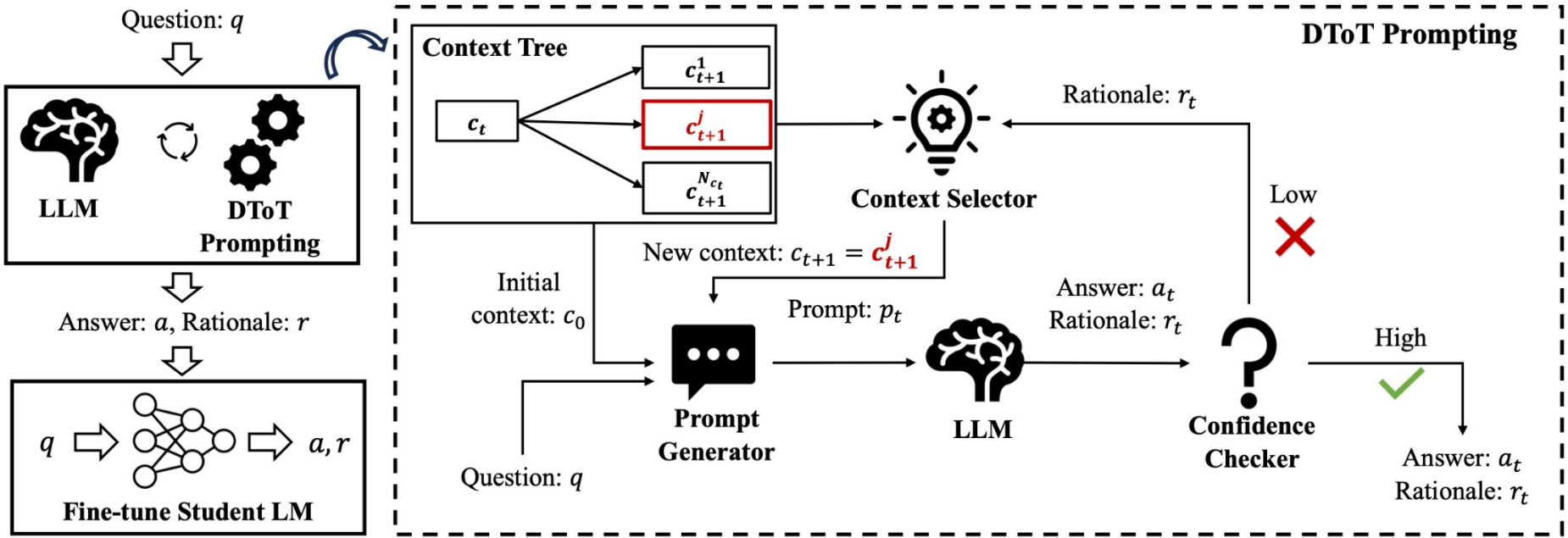
Iteratively extracting high confidence answers a and rationales r from LLM given input question q by using more fine-grained contexts.

Four Modules needs to be designed

- Context Tree
- Prompt generator
- Context Selector
- Confidence Checker

The authors observed that DTOT improved the performance(accuracy, F1 score, AUC) of LLMS when compared to simple CoT prompting.

We train a student LM to predict both answers and rationales, given query q.(Our Final Model). It Outperforms both the parent model and the SOTA on various datasets. The authors also observed that It performs well on other datasets.


Question: q
LLM
DToT Prompting
Answer: a, Rationale: r
q
a, r
Fine-tune Student LM
DToT Prompting
Context Tree
c_t
c_{t+1}^1
c_{t+1}^j
$c_{t+1}^{N_{c_t}}$
Context Selector
Rationale: r_t
Low
New context: $c_{t+1} = c_{t+1}^j$
Initial context: c_0
Prompt: p_t
Answer: a_t
Rationale: r_t
High
Prompt Generator
LLM
Confidence Checker
Question: q
Answer: a_t
Rationale: r_t

Plan

- **Weeks 1-2**: Develop and implement the first approach for the project.
- **Weeks 3-5**: Construct all essential modules for the DToT framework and integrate API calls with the Language Learning Model (LLM).
- **Week 6**: Perform fine-tuning of the student LLM.
- **Week 7**: Experiment with various configurations and assess performance improvements.
- **Week 8**: Compile the final report and submit the completed project.

Thank You!