

## **Project Diary**

This diary documents the timeline, tasks, and contribution made by Yunqiu Du and Tianyi Gong throughout the development of our hybrid dialogue system for contradiction detection and intelligent response generation.

### **October 2 – Topic Discussion**

At first, we considered creating a SpongeBob-themed project. After attending the project-idea discussion session in class, we held a follow-up meeting to rethink our direction. During this meeting, we brainstormed several possible topics and ultimately agreed that focusing on contradiction handling in dialogue systems.

### **October 6 – Topic selection and Project planning**

We thought contradiction handling in dialogue systems is a meaningful and feasible direction. Also it is aligned with the course objectives. After reviewing the technical requirements and potential workflow, we finalized our project topic: designing a contradiction-aware dialogue system. In addition, we outlined the key system components, initial pipeline, and work distribution plan. Later, Tianyi emailed Vlad to confirm if our project topic was appropriate for the course.

### **October 8 – Background Research**

Yunqiu conducted initial background research on existing dialogue systems such as Siri, Replika, and ChatGPT, focusing on their limitations in recognizing and handling contradictions. This helped us clarify the motivation for our system.

### **October 9 – search dataset**

We explored available datasets for contradiction detection independently. Each of us searched for potential resources, and together we compiled a list of candidate datasets for further evaluation and comparison.

### **October 10 – Literature Review**

We began reviewing relevant literature on dialogue consistency and contradiction detection. Both of us searched for references individually, and Tianyi found a particularly useful paper that aligned with our project. We held a meeting to discuss the insights from this reference and how they could inform our system design.

### **October 13 – Discuss and design test questions**

We met to design a set of contradiction-focused test questions for evaluating existing dialogue systems. After discussion, we finalized two test cases:

Q1: "It's rainy outside, I won't go out today." -> later: "I will go out by bike today."

Q2: "I only have one cat, it's white." -> later: "I like my black cat."

We selected these paired because they represent common forms of everyday contradictions.

### **October 15 – Test dialogue system**

Tianyi started the first round of system testing. Tianyi evaluated several widely used dialogue systems, including Apple Siri & Google Assistant & Amazon Alexa & Microsoft Cortana & SoundHound Houndify. Tianyi tested them and documented how each system responded.

### **October 16 – Test dialogue system**

Yunqiu continued testing additional dialogue systems, including: Replika & Character.AI & Mitsuku (Kuki) by Pandorabots, as well as further tests with Apple Siri and ChatGPT to see if there's any difference. Yunqiu tested them, documented, and categorized the responses into three contradiction-handling types: Clarification, Assumption, and Ignorance.

### **October 20 – Analysis of existing dialogue system examples**

We held a meeting to review and analyze all collected test results. We examined the different ways existing dialogue systems responded to contradictions and identified several recurring failure patterns. Through discussion, we think clarification is generally the most desirable response strategy. This analysis helped shape the design goals of our own system.

### **October 21 – System Architecture Design**

Yunqiu drafted the initial version of the system architecture diagram, outlining the overall workflow and component structure. Later Tianyi refined the design by adding more technical details. We finalized the architecture for our hybrid dialogue system, making sure that the contradiction detection module and LLM response generator were integrated.

### **October 22 – Develop the contradiction detection module**

We decided to refer to the implementation approach of Lab 2 and begin developing the conflict detection module based on it. Then Tianyi began developing the contradiction detection module, but he encountered many TypeScript-related issues during the process.(Emailed [Vladislav Maraev](#) and received great suggestion) We discussed and debugged these issues together. And Yunqiu assisted in fixing several minor issues and bugs in the module as well.

### **October 24 – Integration with LLM**

Tianyi integrated the contradiction prediction into the LLM process, making sure the system to adjust the generated response strategy based on whether a contradiction is detected.

### **October 26 – System Test**

Yunqiu tested on the integrated system to evaluate its performance under different inputs. We focused on whether the system could correctly identify contradictions in the dialogue and whether the generated responses became more intelligent and context-appropriate.

### **October 28 – Demo Preparation**

We collaboratively prepared the demo examples to showcase the system's functionality for the presentation.

### **October 30 – presentation outline discussion**

We drafted the structure of the presentation and allocated responsibilities: Yunqiu covers system motivation and architecture, while Tianyi explains the contradiction detection model and presents the demo.

### **November 1 – prepare presentation ppt**

Yunqiu prepared and designed the PPT slides. Tianyi reviewed the slides and added technical details for the detection module.

### **November 2 – write final report**

We worked together on the final written report. Tianyi focused on writing model-related sections, while Yunqiu focused on architecture, background, and overall structuring. The final document was polished collaboratively.