

Detecting Patient Information Conflicts Through Conflict Reasoning in Knowledge Graphs

Research Question: “How can conflict resolution in knowledge graphs enhance the accuracy and reliability of lifestyle management systems?”

Jochem van Paridon | j.m.vanparidon@student.tudelft.nl

June 25, 2025

TU Delft | CSE3000

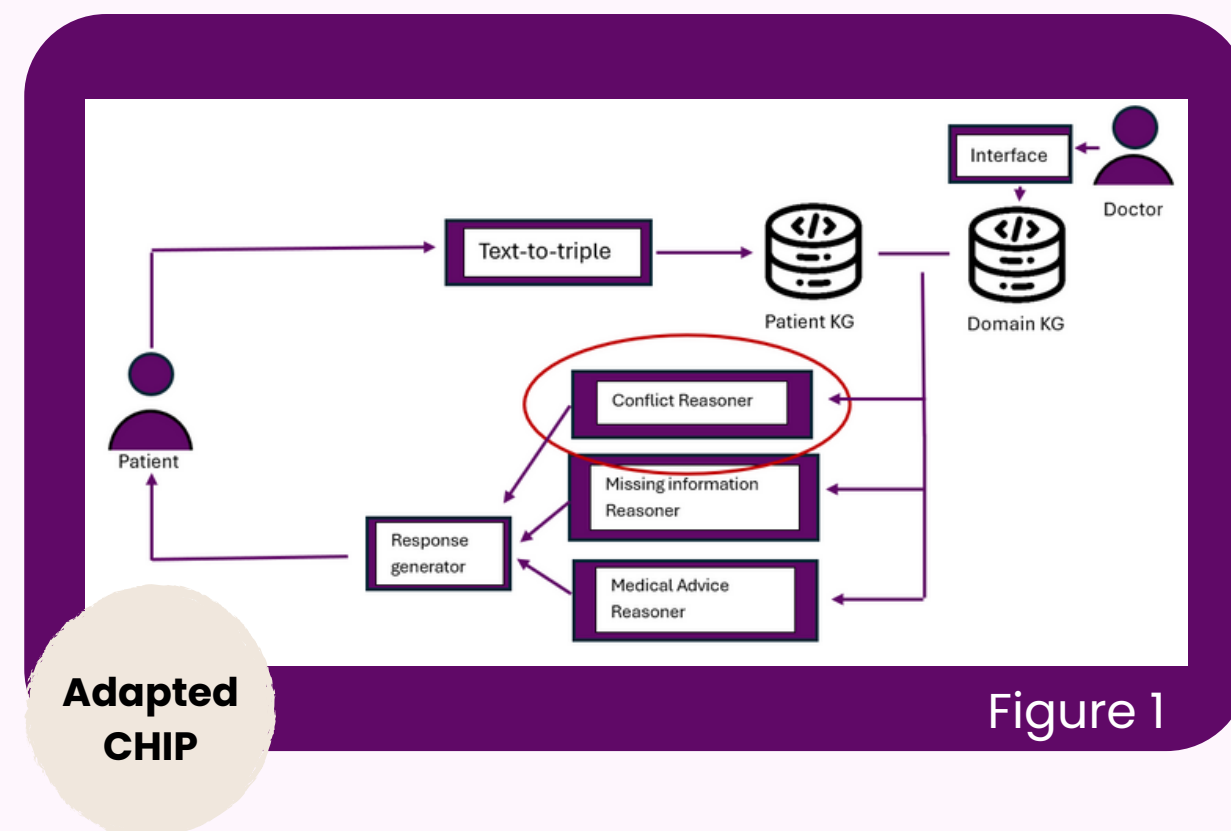
Prof. Catholijn Jonker

J.D. Top, MSc



What is it about?

- CHIP is a chatbot for diabetes lifestyle management, developed by TNO with the Hybrid Intelligence Project (HI).
- It offers personalized lifestyle recommendations based on patient-provided information.
- Patients have been known to not always provide true information.
- Certain falsehoods create conflicts in the patient knowledge graph.
- A conflict reasoner was added to the CHIP system to identify these knowledge graph inconsistencies (see Figure 1).



How does the Conflict Reasoner work?



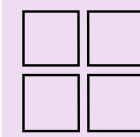
Knowledge Graphs

The data is stored in knowledge graphs (KGs). These consist of triples (s, p, o) : subject s , predicate p , object o . For example: (John, hasAge, 60). If inconsistent information is stored, temporal conflicts may arise.



Temporal Conflicts

Temporal conflicts exist when information contradicts itself over time. These highlight mismatches between the reported timeline and what is medically plausible. To track temporal information, quads are used in CHIP, and constraints help identify potential inconsistencies.



Quads

Instead of using triples (s, p, o) to store the data in the KG, quads are added to store an additional timestamp (s, p, o, t) .



Constraints

The temporal constraints are created by hand, based on causal relationships from a medical perspective.

What did we do?

1. Literature study

Different conflict types were studied: temporal conflicts, semantic conflicts and fact-validation conflicts. Different methods for conflict handling were examined for temporal conflicts and the best applicable was chosen.

2. Designing

A design for within the CHIP system was made. Detecting of conflicts is done within the conflict reasoner.

3. Implementation

The design is implemented in the system to detect inconsistencies in patient-reported data. It ensures that conflicting information in the knowledge graph is identified and addressed.

4. Experimentation (using scenarios)

Multiple scenarios are created synthetically to test if the system is able to detect conflicts in different situations.

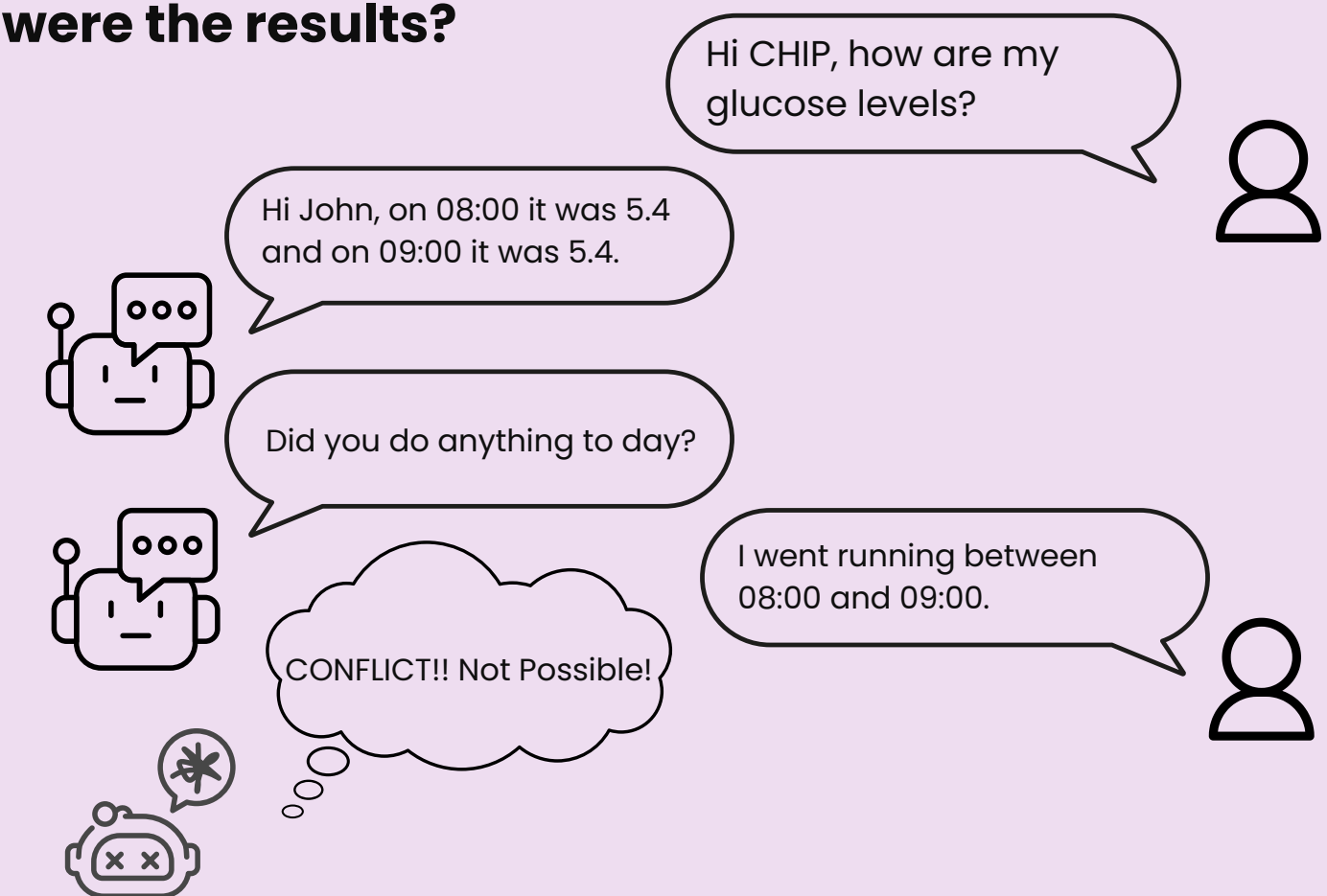
What does a constraint look like?

```
precedes( $t_1, t_2$ ) := ( $x, \text{glucoseMeasurement}, a, t_1$ )  
( $x, \text{activity}, y, t_2$ )  
( $x, \text{glucoseMeasurement}, b, t_2$ )  
 $a > b + \epsilon$ 
```

Constraint 1

Constraint 1 states that after activity your glucose levels should have dropped.

What does a scenario look like and what were the results?



- In the example above an scenario is illustrated on how the experiment is done. CHIP detects that the constraint is violated (see Constraint 1). The glucose levels do not correspond with the activity of the patient.
- Each scenario resulted in successful conflict detection by CHIP.