19/11/24

Week 8

Date / /
Page

Unification in First-Order Logic

Key conditions

i) Same predicate symbol : the Predicate symbols in the expressions must match

ii) Same number of arguments : the expressions must have an equal number of arguments

iii) Variable conflict resolution : variables cannot takes multiple conflicting values

iv) No conflicting function symbols : Different function symbols cannot unify

Examples :-

(1) expression A : Knows(f(x,y), g(x))
expression B : Knows(f(Alice, Bob), g(z))

Steps :-

i) By comparing the predicates, we arrive that. Both are Knows

ii) Consider f(x, y) => f(Alice, Bob)
x = Alice, y = Bob
g(x) ⟶ g(z)
z = Alice (since x = Alice)

iii) x = Alice
y = Bob
z = Alice

iv) unified expression
Knows (f(Alice, Bob), g(Alice))

fm 26/11/24

Date __/__/__
Page _____

code :-

```
Knowledge_base = [
{"type": "rule", "rule": "∀x∀y (Doctor(x) ∧ Sickly(y) → Treats(x, y))"},
{"type": "fact", "fact": "Doctor(Jophen)"},
{"type": "fact", "fact": "∀x (Doctor(x) → ∃h (Hospital(h) ∧
        WorksAt(x, h)))"},
{"type": "fact", "fact": "Hospital(GeneralHospital)"},
{"type": "fact", "fact": "WorksAt(John, GeneralHospital)"},
]

query = {"predicate": "Treats", "arguments": ["?", "Mary"]}.

def unify(kb, query):
    predicate = query["predicate"]
    target_args = query["arguments"][1]
    result = None.

for item in kb:
    if item["type"] == "rule" and predicate in item["rule"]:
    rule = item["rule"]
    if "Doctor(x)" in rule and "Sickly" in rule:
        doctor = None
        sick_person = None

for fact in kb:
    if fact["type"] == "fact" and "Doctor" in fact["fact"]:
        doctor = fact["fact"].split("(")[1][:-1]
    if fact["type"] == "fact" and "Sick" in fact["fact"]:
        sick_person = fact["fact"].split("(")[1][:-1]
```

LAB 9

Date __/__/__
Page

First

```
if result:
    return f"The query '{query['predict']}({result},
        {target_args})' is unified: {result} treats
            {target_arg2}."
else:
    return "The query '{query} {query['predicates']}({
        query['arguments'][0]}, {target_args})'
        could not required with knowledge base."

result = unify(knowledge_base, query)
print(result).
```

Output :-

The query can't Access? , Project x ' is unified:
John Treats Mary