

propositional logic

$p \rightarrow q$ if p then q
Premise \downarrow consequence

If p is true then we can infer q .
when p is true then q must
be true.

Formal Representation of Entailment

$p, (\neg p \rightarrow q) \models q$

This means, given P and $P \rightarrow Q$, it logically follows that Q must be true.

The symbol \vdash represents logical entailment.

Knowledge Base

1. Alice is the mother of Bob $\rightarrow P_1$
 2. Bob is father of Charlie $\rightarrow P_2$
 3. Charlie is a parent $\rightarrow P_3$
 4. Alice is a parent $\rightarrow P_4$
 5. All parents have children $\rightarrow P_5$
 6. If someone is a parent, their children are siblings $\rightarrow P_6$
 7. Alice is married to David $\rightarrow P_7$

Hypothesis:

- Charlie is a sibling of Bob

\Rightarrow Implications:

1. From P₁: if Alice is the mother of Bob then Bob is a child of Alice.

P₁ \rightarrow Q₁: Bob is a child of Alice

2. From P₂: Bob is father of Charlie

P₂ \rightarrow Q₂: Charlie is a child of Bob.

3. From P₃ & P₄

P₃ \rightarrow P₄: Bob is a parent and Alice is a parent.

4. From P₅:

P₅ \rightarrow Q₃: Bob & Alice have children

5. From P₆:

P₆ \rightarrow H: Charlie & Bob are siblings

\Rightarrow Conclusion: P₁ \wedge P₂ \wedge P₃ \wedge P₄ \wedge P₅ \wedge P₆ \rightarrow H.

Sum
10/10/20

class Knowledge Base:

def __init__(self):

self.rules = []

self.facts = set()

def add_fact(self, fact):

self.facts.add(fact)

def add_rule(self, premise, conclusion):
self.rules.append((premise, conclusion))

def infer(self):

new_inferences = True

while new_inferences == True

for premise, conclusion in self.rules:

if all(fact in self.facts for fact in premise):

if conclusion not in self.facts:

self.facts.add(conclusion)

new_inferences = True

def entail(self, hypothesis):

return hypothesis in self.facts

kb = knowledge_base()

kb.add_fact("Alice is mother of Bob")

~~kb.add_fact("Bob is the father of Charlie")~~

~~kb.add_fact("If father is a parent")~~

~~kb.add_fact("All brothers are parents")~~

~~kb.add_fact("All parents have children")~~

~~kb.add_fact("Alice is married to David")~~

tb.add rule ("Bob is father of charlie",
"father is a parent"), "Bob is a parent"

tb.add rule ("Alice is the mother of boy",
"A mother is a parent"), "Alice is a parent"

tb.add rule ("Bob is a parent", "he
has children"), "charlie & bob are
siblings")

tb.infer()

hypothesis = "Charlie and Bob are
siblings"

if tb. entails (hypothesis):

print ("The hypothesis is entailed
by the knowledge base.")

else:

print ("The hypothesis is not entailed
by the knowledge base.")

out put:

The hypothesis, 'Charlie & Bob
are siblings' is entailed by the
knowledge base.

Jan 19/2024