Answer1

(a)(i)

p	q	r	$q \vee r$	$p \wedge (q \vee r)$	$p \wedge q$	$p \wedge r$	$(p \wedge q) \vee (q \wedge r)$	
T	T	T	T	T	T	T	T	√
T	T	F	T	T	T	F	T	√
T	F	T	T	T	F	T	T	√
T	F	F	F	F	F	F	F	
F	T	T	T	F	F	F	F	
F	T	F	T	F	F	F	F	
F	F	T	T	F	F	F	F	
F	F	F	F	F	F	F	F	

Therefore, it could prove the inference hold in propositional logic using truth table.

(a)(ii)

$p \wedge (q \vee r) \vdash (p \wedge q) \vee (p \wedge r)$

$CNF[\neg((p \wedge q) \vee (p \wedge r))]$

$\equiv \neg(p \wedge q) \wedge \neg(p \wedge r)$

$\equiv (\neg p \vee \neg q) \wedge (\neg p \vee \neg r)$

1. p (premise)
2. $q \vee r$ (premise)
3. $\neg p \vee \neg q$ (negated conclusion)
4. $\neg p \vee \neg r$ (negated conclusion)
5. $\neg q$ (1, 3 resolution)
6. r (2, 5 resolution)
7. $\neg p$ (4, 6 resolution)
8. $\square$ (1, 7. resolution)

Therefore, it could prove the inference hold in propositional logic using resolution.

(b)(i)

p	q	$q \to p$	$p \to (q \to p)$	
T	T	T	T	✓
T	F	T	T	✓
F	T	F	T	✓
F	F	T	T	✓

Therefore, it could prove the inference hold in propositional logic using truth table.

(b)(ii)

$CNF[\neg(p \to (q \to p))]$

$\equiv \neg(\neg p \vee (\neg q \vee p))$

$\equiv \neg\neg p \wedge \neg(\neg q \vee p)$

$\equiv \neg\neg p \wedge (\neg\neg q \wedge \neg p)$

$\equiv p \wedge (q \wedge \neg p)$

$\equiv p \wedge q \wedge \neg p$

1. p (negated conclusion)
2. q (negated conclusion)
3. $\neg p$ (negated conclusion)
4. $\square$ (1. 3. resolution).

Therefore, it could not prove the inference hold in propositional logic using resolution.

(c)(i)

p	q	$p \to q$	$\neg p$	$\neg q$	$\neg p \to \neg q$	
T	T	T	F	F	T	✓
T	F	F	F	T	T	
F	T	$\boxed{T}$	T	F	$\boxed{F}$	✗
F	F	T	T	T	T	✓

Therefore, it could not prove the inference hold in propositional logic using truth table

since $p \to q$ is truth but $\neg p \to \neg q$ is false in the third row.
(c)(ii)

$CNF[p \to q]$

$\equiv \neg p \vee q$ (remove $\to$)

$CNF[\neg(\neg p \to \neg q)]$

$\equiv \neg(\neg\neg p \vee \neg q)$ (remove $\to$)

$\equiv \neg(p \vee \neg q)$ (double negation)

$\equiv \neg p \wedge \neg\neg q$ (De Morgan)

$\equiv \neg p \wedge q$ (double negation)

1. $\neg p \vee q$ (premise)
2. $\neg p$ (negated conclusion)
3. q (negated conclusion)

Therefore, it could not get empty clause using resolution.
(d)(i)

p	q	$\neg p$	$\neg q$	$q \to p$	$\neg q \to \neg p$	$p \leftrightarrow q$	
T	T	F	F	T	T	T	V
T	F	F	T	T	F	F	
F	T	T	F	F	T	F	
F	F	T	T	T	T	T	✓

Therefore, it could prove the inference hold in propositional logic using truth table.
(d)(ii)

$CNF[q \to p]$

$\equiv \neg q \vee p$ (remove $\to$)

$CNF(\neg q \to \neg p)$

$\equiv \neg\neg q \vee \neg p$ (remove $\to$)

$\equiv q \vee \neg p$ (double negation)

$a \lor \neg(\neg p \leftrightarrow \neg q))$

$\equiv \neg((p \to q) \land (p \leftarrow q))$

$\equiv \neg(p \to q) \lor \neg(p \leftarrow q)$ (De Margon)

$\equiv \neg(\neg p \lor q) \lor \neg(\neg q \lor p)$ (remove $\to$)

$\equiv (\neg\neg p \land \neg q) \lor (\neg\neg q \land \neg p)$ (De Margon)

$\equiv (p \land \neg q) \lor (q \land \neg p)$ (double negation)

$\equiv (p \lor q) \land (\neg q \lor \neg p) \land (p \land \neg p) \land (\neg q \land q)$

1. $\neg q \lor p$ (premise)
2. $q \lor \neg p$ (premise)
3. $p \lor q$ (negated conclusion)
4. $\neg q \lor \neg p$ (negated conclusion)
5. $p \lor \neg p$ (negated conclusion)
6. $\neg q \lor q$ (negated conclusion)
7. p (1, 3, resolution)
8. $\neg q$ (4, 7, resolution)
9. $\neg p$ (2, 8, resolution)
10. $\square$ (7, 9, resolution)

Therefore, it could not prove the inference hold in propositional logic using resolution.
(e)(i)

p	q	r	$p \to q$	$q \to r$	$r \to q$	
T	T	T	T	T	T	√
T	T	F	T	F	T	
T	F	T	F	T	F	
T	F	F	F	T	T	
F	T	T	T	T	T	√
F	T	F	T	F	T	
F	F	T	$\boxed{T}$	$\boxed{T}$	$\boxed{F}$	X
F	F	F	T	T	T	√

Therefore, it could not prove the inference hold in propositional logic using truth table since $p \rightarrow q$ and $q \rightarrow r$ are truth but $r \rightarrow q$ is false in the second row from the bottom.

(e)(ii)

$CNF[(p \rightarrow q)]$

$\equiv \neg p \vee q$ (remove $\rightarrow$)

$CNF[(q \rightarrow r)]$

$\equiv \neg q \vee r$ (remove $\rightarrow$)

$CNF[\neg(r \rightarrow q)]$

$\equiv \neg(\neg r \vee q)$ (remove $\rightarrow$)

$\equiv \neg\neg r \wedge \neg q$ (De Morgan)

$\equiv r \wedge \neg q$ (double negation)

1. $\neg p \vee q$ (premise)
2. $\neg q \vee r$ (premise)
3. r (negated conclusion)
4. $\neg q$ (negarted conclusion)
5. $\neg p$ (1, 4 resolution)

Therefore, it could not get empty clause using resolution.

Answer 2

(a)

① "I never stole the jam!" March Hare

$\neg Lying(marchHare) \leftrightarrow \neg stole(marchHare, jam)$

② "One of us stole it, but it wasn't me!" Mad Hatter.

$\neg Lying(madHatter) \leftrightarrow ((stole(marchHare, jam) \wedge \neg stole(doormouse)) \vee$
$(\neg stole(marchHare, jam) \wedge stole(doormouse))) \wedge$
$\neg stole(madHatter)$

③ "At least one of them did" Doormouse.

$\neg Lying(doormouse) \leftrightarrow stole(marchHare, jam) \vee stole(madHatter, jam)$

④ the March Hare and the Doormouse were not both tell the truth.

$Lying(marchHare) \vee Lying(doormouse)$

(b)

$S = \{ \neg Lying(marchHare) \leftrightarrow \neg stole(marchHare, jam),$

$\neg Lying(madHatter) \leftrightarrow ((stole(marchHare, jam) \wedge \neg stole(doormouse, jam)) \vee$

$(\neg stole(marchHare, jam) \wedge stole(doormouse, jam))) \wedge$

$\neg stole(marchHatter),$

$\neg Lying(doormouse) \leftrightarrow stole(marchHare, jam) \vee stole(madHatter, jam),$

$Lying(marchHare) \vee Lying(doormouse)$

$\}$

$\alpha = \exists x \; stole(x, jam)$

Claim $S \models \alpha$

Proof:

Let I be any interpretation such that $I \models S$

Case 1: $I \models Lying(marchHare) \wedge \neg Lying(doormouse) \wedge \neg Lying(madHatter)$

$\therefore I \models \neg Lying(doormouse) \rightarrow stole(marchHare, jam) \vee stole(madHatter, jam)$

$\therefore I \models Lying(marchHare) \rightarrow stole(marchHare, jam)$

$\therefore I \models stole(marchHare)$

$\therefore I \models \alpha$

Case 2: $I \models \neg Lying(marchHare) \wedge Lying(doormouse) \wedge \neg Lying(madHatter)$

$\therefore I \models \neg Lying(marchHare) \rightarrow \neg stole(marchHare, jam)$

$\therefore I \models \neg Lying(madHatter) \rightarrow ((stole(marchHare, jam) \wedge \neg stole(doormouse, jam)) \vee$

$(\neg stole(marchHare, jam) \wedge stole(doormouse, jam))) \wedge$

$\neg stole(madHatter)$

$\therefore I \models Lying(doormouse) \rightarrow \neg stole(marchHare) \wedge \neg stole(madHatter)$

$\therefore I \models stole(doormouse)$

$\therefore I \models \alpha$

Case 3: $I \models \neg Lying(marchHare) \wedge \neg Lying(doormouse) \wedge Lying(madHatter)$

$\therefore I \models \neg Lying(marchHare) \leftrightarrow \neg stole(marchHare, jam)$

$\therefore I \models \neg Lying(doormouse) \leftrightarrow stole(marchHare, jam) \vee stole(madHatter, jam)$

$\therefore I \models Lying(madHatter) \rightarrow ((\neg stole(marchHare, jam) \vee stole(doormouse, jam)) \wedge$
$(stole(marchHare, jam) \vee \neg stole(doormouse, jam)) \vee$
$stole(madHatter)$

$\therefore I \models \neg Lying(marchHare) \wedge \neg Lying(doormouse)$

It is controdictory with the truth $Lying(march) \wedge Lying(doormouse)$ and $I \models Lying(madHatter)$ require DoorMouse stole the jam, but I can not get it.

Therefore, I could not determine who stole the jam. In the case 1 and 2, I got two different result. The reason is that there have three conditions from Door Mouse. The first condition is that March Hare stole the jam, the second condition is that Mad Hatter stole the jam and the third condition is that both of them stole the jam. In case 3, it is conflict with the truth that the March Hare and Doormouse were not both speaking the truth. So, the answer is no.

(c)

As above condition from case 1 and 2, I think that if assuming only one person stole the jam, it could know that Mad Hatter stole the jam. The sentence I intend to add is as follows.

$$\exists x\, stole(x, jam) \wedge (x = marchHare \vee x = madHatter \vee x = doormouse)$$

(d)

1. $Lying(marchHare) \vee Lying(doormouse)$ (premise)
2. $\exists x\, stole(x, jam) \wedge (x = marchHare \vee x = madHatter \vee x = doormouse)$ (premise)
3. $\neg Lying(madHatter) \rightarrow ((stole(marchHare, jam) \wedge \neg stole(doormouse, jam)) \vee$
$(\neg stole(marchHare, jam) \wedge stole(doormouse, jam))) \wedge$
$\neg stole(madHatter)$ (premise)
4. $\neg Lying(doormouse) \leftrightarrow stole(marchHare, jam) \vee stole(madHatter, jam)$ (premise)
5. $Lying(madHatter)$ (1, 2. resolution)
6. $Stole(marchHare)$ (4, 5 resolution)

Answer 3

1 Introduction

It is also known that 3-SAT exhibits an easy-hard-easy computational pattern. Determining the satisifiability of sets of clauses that are small in relation to the total number of distinct propositional variables in the set is usually easy because there are fewer constraints in assigning truth values to the propositional variables. Determining the satisifiability of sets of clauses that are large in relation to the total number of distinct propositional variables in the set is usually easy because there are too many constraints to assign truth values to the propositional variables and the set is unsatisfiable. Somewhere in between these two extremes the satisfiability problem becomes hard.

2 Generating Test Cases

I wrote a short program *generator.py* and used it to generate 13 cases for this question from *file0.cnf* to *file12.cnf* in the current folder.

Example: *file0.cnf*
c example CNF file with 1000 propositional variables and 100 clauses
p cnf 1000 100
-405 -89 -266 0
-780 58 202 0
223 -663 -487 0
-149 570 936 0
…

3 Testing

I tested these 13 cnf files by **minisat** on the CSE machines. I recorded the details in the txt format files from *file0Statistics.txt* to *file12Statistics.txt* in the current folder. In these two extremes, it's easy to solve since some of them have zero or almost zero CPU time. Even some of them are unsatisfiable, but the CPU time is very low. Some of cases are as follows.

Case 0:
Number of variables: 990
Number of clauses: 100
restarts: 1
conflicts: 0 (-nan /sec)
decisions: 1 (0.00 % random) (inf /sec)
propagations: 0 (-nan /sec)
conflict literals: 0 (-nan % deleted)
Memory used: 5.00 MB
CPU time: 0 s
SATISFIABLE

Case 1:
Number of variables: 100

Number of clauses:100
restarts: 1
conflicts: 0 (-nan /sec)
decisions: 1 (0.00 % random) (inf /sec)
propagations: 0 (-nan /sec)
conflict literals: 0 (-nan % deleted)
Memory used: 5.00 MB
CPU time: 0 s
SATISFIABLE

Case 2:
Number of variables: 400
Number of clauses: 1200
restarts: 1
conflicts: 18 (inf /sec)
decisions: 128 (0.00 % random) (inf /sec)
propagations: 1390 (inf /sec)
conflict literals: 325 (6.88 % deleted)
Memory used: 5.00 MB
CPU time: 0 s
SATISFIABLE

Case 6:
Number of variables: 100
Number of clauses: 986
restarts: 1
conflicts: 3 (inf /sec)
decisions: 2 (0.00 % random) (inf /sec)
propagations: 50 (inf /sec)
conflict literals: 3 (0.00 % deleted)
Memory used: 5.00 MB
CPU time: 0 s
UNSATISFIABLE

Case 7:
Number of variables:100
Number of clauses:1486
restarts: 1
conflicts: 3 (inf /sec)
decisions: 3 (0.00 % random) (inf /sec)
propagations: 36 (inf /sec)
conflict literals: 2 (0.00 % deleted)
Memory used: 5.00 MB
CPU time: 0 s

UNSATISFIABLE

Case 8:
Number of variables:100
Number of clauses:2000
restarts: 0
conflicts: 0 (-nan /sec)
decisions: 0 (-nan % random) (-nan /sec)
propagations: 8 (inf /sec)
conflict literals: 0 (-nan % deleted)
Memory used: 5.00 MB
CPU time: 0 s
UNSATISFIABLE

At the beginning, I only generated 8 cases. However, most of them are easy to solve except two cases which C is 4 and 5. Therefore, I guess that C value may between 4 and 5 and then generated other cases for testing. Based on the 13 cases, I found that when C approaches 4.25, the CPU time gradually becomes very high and the number of conflicts is very large. It means that the problem becomes very hard. When C equals to 4.25, CPU time is 102.248s. It also has many conflicts and unsatisfiable after a very long time. The details are as follows. Except it, when C equals 4.5, the CPU time is 52.1s. When C equals 4.75, the CPU time is 13.416s.

Case 12:
Number of variables: 400
Number of clauses:1700
restarts: 8190
conflicts: 4784225 (46790 /sec)
decisions: 5807724 (0.00 % random) (56800 /sec)
propagations: 283229619 (2770026 /sec)
conflict literals: 74493486 (25.87 % deleted)
Memory used: 6.00 MB
CPU time: 102.248 s
UNSATISFIABLE

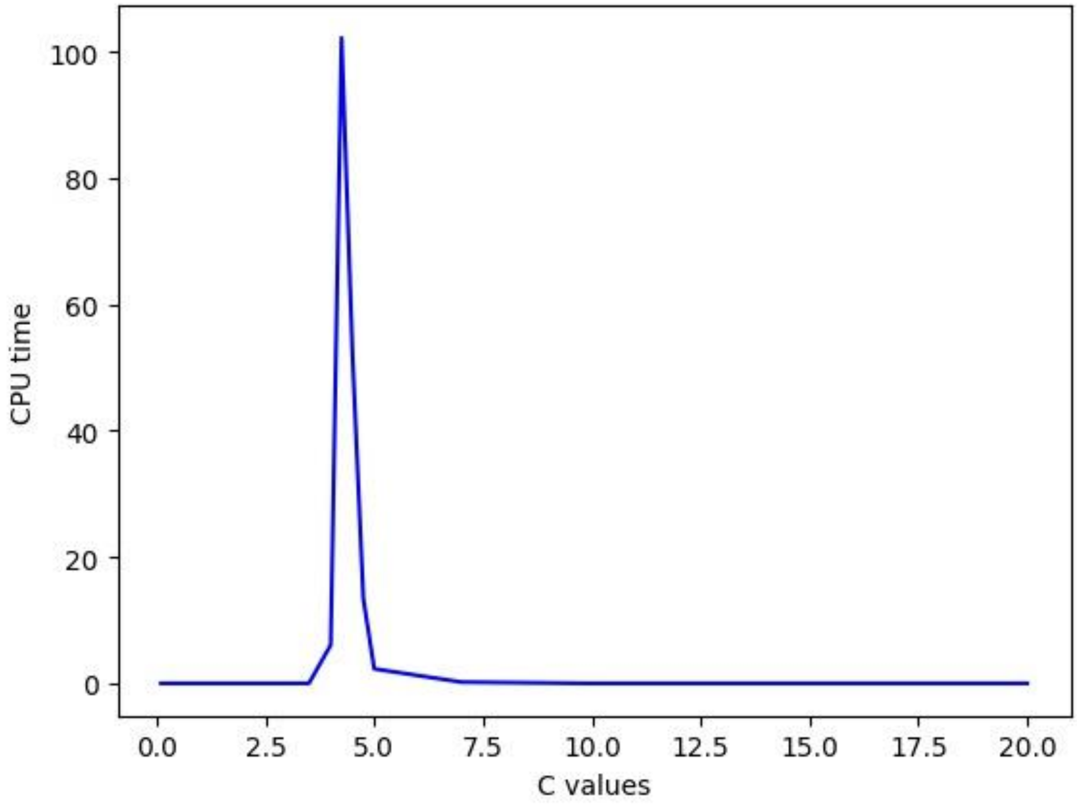
Case 10:
Number of variables: 400
Number of clauses: 1800
restarts: 4350
conflicts: 2552014(48983 /sec)
decisions: 3110877 (0.00 % random) (59710 /sec)
propagations: 145426024 (2791286 /sec)
conflict literals: 39410100 (24.78 % deleted)
Memory used: 5.00 MB

CPU time: 52.1 s
UNSATISFIABLE

Case 11:
Number of variables: 400
Number of clauses: 1900
restarts: 1534
conflicts: 729867 (54403 /sec)
decisions: 883583(0.00 % random) (65860 /sec)
propagations: 41083843 (3062302 /sec)
conflict literals: 9998794 (26.75 % deleted)
Memory used: 5.00 MB
CPU time: 13.416 s

4 Conclusion

According to the test, I came up with the constant value C≈4.25 empirically. The number of variables is 400 and the number of clauses is 1700. I also made a bar chart(*barchat.jpg*) by *barchat.py* in the folder in order to show the conditions of all cases. The 13 cases show the point where the problem is hard to determine.



Answer 4

1 Brief Summary

The recent success of neural networks in applications makes many people describe the automations of these tasks as having reached human level intelligence. Many researchers have a dilemma of "What just happened in AI?". Therefore, the author wanted to trigger a discussion about recent developments in AI. Firstly, the author introduced model-based and function-based approaches. But function-based approach has a question that it highlights problems and thresholds more than it highlights technology. Then, two key questions are the following. Are the functions simple enough and do we have the ability to estimate these functions? There are three developments influencing these questions. The first is our improved ability to fit functions to data. The second is we have identified applications. The third is we changed the measures for success. The author thought the development of AI in some certain areas can not be called a breakthrough. But AI has impact on automation. He believed that attributing human level intelligence to the tasks currently conquered by neural networks is questionable. The current derivative for progress on neural has not been sustained long, so there are two questions. The first is about the whether the functions of cognitive tasks reach the thresholds and the second is about the functions are only approximations. Then, the author said that we face a bullied-by-success phenomena, so the government has the responsibility to guide junior researchers. In the end, he thought the need is cognitive function which captures a relationship that is typically associated with cognition. It has a catalogue of cognitive functions, a study of their representational complexity and a study of their learnability and approximateability. So, he said that he prefers to rename the field of deep learning to the field of *learning approximations of cognitive functions*.

2 Agreement

I agree with that while the current AI technology is still very limited, the impact it may have on automation and, hence, society may be substantial. The reason is that we could see many AI technologies such as some neural networks just apply to some certain commercial tasks and many of them are just emulate some processes which could reach some expected point of these tasks, like accuracy. In other words, we may do not know a precise underlying principle. However, we still use these technologies to solve some redundancy and duplication of work(automations), which has good results.

3 Disagreement

I disagree with that the combination of some behaviors of researchers and other members in community is harmful to scientific inquiry. The reason is that even we have many lessons of failure, such as the symbolic logic, we still can not predicate the future clearly since the wisest decision maker may make some mistakes. Therefore, from my perspective, industry which will trigger more discussion and research would bear more responsibilities in order to make the AI community have a huge progress since the practice will let us know more. It is a little bit difficult for the decision makers and senior members to control the current development and the current commercial success.