ASSIGNMENT - 3

MC - 307

Lokesh Godara
23/MC/082
E5

Ans 1) • parent (child, Parent):

parent (lili, paul) , parent (lili, helen)

parent (petunia, paul), parent (petunia, helen)

parent (james, albert), parent (james, ruth)

parent (dudley. vernon), parent (dudley, petunia)

parent (harry, lili) , parent (harry, james)

male (paul) , female (helen)

male (albert) , female (ruth)

male (vernon) , Female (petunia)

male (james) , Female (lili)

male (harry) , Female (dudley)

- grandparent (Grandchild, Grandparent) :-
  parent (Grand child, Parent),
  parent (Parent, Grand parent).

- grand father (Grandchild, Grand father) :-
  grandparent (Grandchild, Grand Father),
  male (Grandfather).

- grandmother (Grandchild, Grandmother) :-
  grandparent (Grandchild, Grandmother),
  female (Grandmother)

?- grand father (dudley, Grand father).

(Grand father = paul)

?- grandmother (dudley, Grandmother).

(Grand mother = helen)

?- grand father (harry, Grand father).

(Grand father = paul), (Grandfather = Albert)

?- grand mother (harry, Grandmother).

(Grand mother = helen), (Grand mother = ruth)

Ans2) a. Concatenation of two lists.

(append / 3):

```
append ([], L, L).
append ([H|T1], L2, [H|T3]):-
    append (T1, L2, T3).
```

---

b. Reverse (reverse /2):

```
reverse (L, R):-
    rev_aux ( L, [], R).

rev_aux ([], Acc, Acc).
rev_aux([H|T], Acc, R):-
    rev_aux (T, [H|Acc], R).
```

---

c. Divide (in Two equal List):

```
split ( L, , L1, L2) :-
    length ( L, Len),
    Mid is Len // 2,
    length ( L1, Mid).
    append ( L1, L2, L).
```

---

d) Sort the elements of a list:

```
sort ( [], []).
sort ([H|T]. Sorted ) :-
    sort (T, SortedT),
    insert ( H, SortedT, Sorted).

insert ( E, [], [E] ).
insert ( E, [H|T], [E,H|T]):-
    E =< H.
insert ( E, [H|T] , [H|NewT]) :-
    E > H,
    insert (E, T, NewT).
```

Ans 3) 2 Pass , 6 Fail

Entropy $E(S)$

$$E(S) = -p_{pass} \log_2 (p_{pass}) - p_{fail} \log_2 (p_{fail})$$

$$E(S) = -\left(\frac{2}{8}\right) \log_2\left(\frac{2}{8}\right) - \left(\frac{6}{8}\right) \log_2\left(\frac{6}{8}\right)$$

$$= -(0.25 \times (-2)) - (0.75 \times (-0.415))$$

$$= 0.5 + 0.311$$

$$\boxed{E(S) \approx 0.811}$$

Information Gain:

$$Gain(S, A) = E(S) - \sum_{v} \frac{|S_v|}{|S|} E(S_v)$$

Attribute: Reading:

$E(S, Reading) = 0.5 + 0 = 0.5$

$Gain(S, Reading) = 0.811 - 0.5$

$Gain(S, Redding) = \underline{0.311}$

Attribute: Assignment:

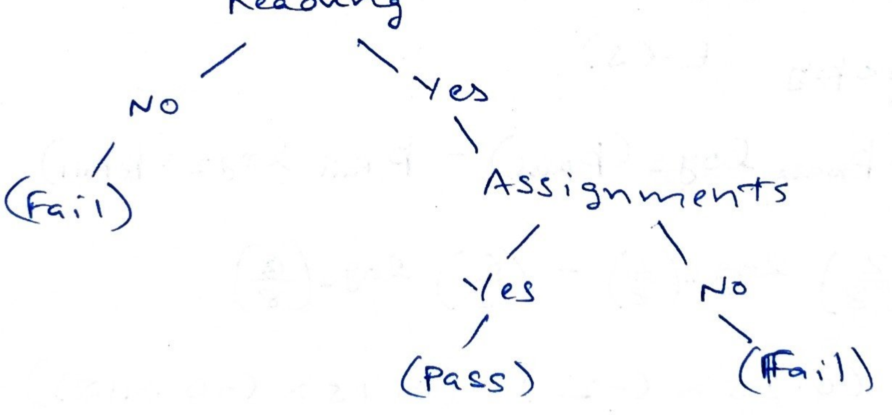
$E(S, Assignment) = 0.5 + 0 = 0.5$

$Gain(S, Assignment) = 0.811 - 0.5 \approx \underline{0.311}$

Attribute: End Term Paper:

$E(S, End) = 0.4055 + 0.4055 = 0.811$

$Gain(S, End) = 0.811 - 0.811 = \underline{0.0}$



Ans 4)

Naive Bayes Classifier is a simple, highly effective probabilistic machine learning algorithm based on Bayes' Theorem with 'naive" assumption of conditional independence b/w features

It classifies an email message (D) into one of two categories (classes, C): Spam or Useful (Ham) by calculating the probabilities that the document belongs to each class given its content.

Naive Bayes Formula:

$$P(C|D) \propto P(C) \cdot \prod_{i=1}^{n} P(w_i|C)$$

$P(C|D)$: Prob. ~~that~~ email is of class C given document D

Ans 5)

$C(x)$: x is a child.

$L(x,y)$: x loves y.

$R(x)$: x is Reindeer

$N(x)$: x is a red nose

$W(x)$: x is weird

$K(x)$: x is clown.

a) $\forall x\ (C(x) \rightarrow L(x, santa))$

b) $\forall x \forall y\ ((L(x, santa) \wedge R(y)) \rightarrow L(x,y))$

c) $R(x) \wedge N(x)$

d) $\forall x\ (N(x) \rightarrow (W(x) \vee K(x)))$

e) $\forall x\ (R(y) \rightarrow \sim K(x))$

~~f) $\forall x (L(Alice, x) \rightarrow \sim KW(x))$~~

f) $\forall x\ (W(x) \rightarrow \sim L(Alice, x))$

CNF: $\sim C(Alice)$ by Resolution :

$\sim C(x) \vee L(x, santa)$, $\sim L(x, santa) \vee R(y) \vee L(x,y)$,

$\sim N(x) \vee W(x) \vee K(x)$, $\sim R(y) \vee \sim K(x)$,

$\sim W(x) \vee \sim L(Alice, x)$

Ans 6) $L(x,y)$: x loves y

$F(x)$: x is a football star

$S(x)$: x is a student

$P(x)$: x passes

$Y(x)$: x plays

$U(x)$: x studies

a) $\forall x \left(L(Mary, x) \rightarrow F(x) \right)$

b) $\forall x \; (S(x) \wedge \sim P(x)) \rightarrow \; \sim Y(x)$

c) $S(John)$

d) $\forall x \left((S(x) \wedge \sim U(x)) \rightarrow \; \sim P(x) \right)$

e) $\forall x \; (\sim Y(x) \rightarrow \; \sim F(x))$

$(\sim U(John) \rightarrow \; \sim L(Mary, John))$:

Ans 7) Cryptarithmetic Problems as CSPs.

Constraints Satisfaction Problems (CSPs)

V : variables

D : Domain

C : constraints

eg. SEND + MORE = MONEY

V = {S, E, N, D, M, O, R, Y}

D = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

C : All diff (S, E, N, D, M, O, R, Y)

$S \neq 0$, $M \neq 0$

$C_i \in \{0, 1\}$

$D + E = Y + 10 C_1$

$C_1 + N + R = E + 10 \cdot C_2$

$C_2 + E + O = N + 10 C_3$

$C_3 + S + M = O + 10 C_4$

$C_4 = M$

SEND + MORE = MONEY is :

~~95~~ 9567 + 1085 = 10652

S = 9, E = 5, N = 6, D = 7, M = 1, O = 0, R = 8

Y = 2

Ans8) Backtracking seach algorithm for constraint satisfaction problems (CSPs) is ~~a~~ inefficient ~~to~~ because it wastes time exploring irrelevant sub-trees.

Techniques:

1. Filtering : Constraint Propogation

- Forward checking
- Arc consistency

2. ordering:

- variable ordering : Minimum Remaining values
- variable ordering : Degree Heuristic
- value ordering : Least constraining value

3. Structure - Based:

- Tree Structured CSPs.
- Cycle cutset