

T04 Machine Learning

17341137 Zhenpeng Song

December 16, 2019

Contents

1	Q1	2
2	Q2	4
2.1	Bayesian	4
2.2	MAP	4
2.3	ML	5
3	Q3	6
4	Q4	7

1 Q1

To Judge which attribute to select, we should calculate these two important indices:

- Entropy: $Ent(D) = - \sum_{k=1}^{|Y|} p_k \log_2 p_k$, Y stands for the possible values for the label (In this case: User Action, $-Y- = 2$).
- Info_Gain: $Gain(D, a) = Ent(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} Ent(D^v)$

Therefore, I produce the graph below:

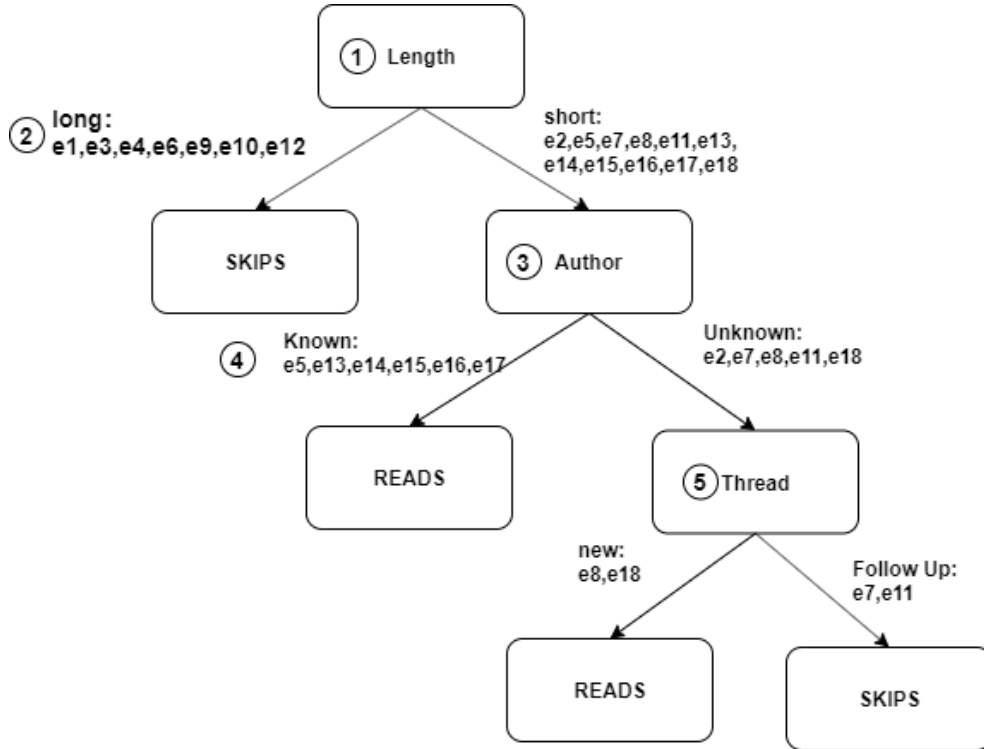


Figure 1: Decision Tree

Where steps 1 to 5 are:

1. Chosen attribute: **Length**. $D \rightarrow D1, D2$.
2. Calculate Info_Gain: $Ent(D1) = 0, Ent(D2) = -\frac{2}{11} \times \log_2(\frac{2}{11}) - \frac{9}{11} \times \log_2(\frac{9}{11}) = 0.6840$.
 - $Author = 0.6840 - \frac{5}{11} \times 0.9710 - \frac{6}{11} \times 0 = 0.2426$
 - $Thread = 0.6840 - \frac{4}{11} \times 1 - \frac{7}{11} \times 1 = -0.3160$
 - $WhereRead = 0.6840 - \frac{5}{11} \times 0.7219 - \frac{6}{11} \times 0.6500 = 0.0013$

3. Choose attribute: **Author**. $D2 \rightarrow D3, D4$.
4. Calculate Info_Gain: $Ent(D3) = 0, Ent(D4) = -\frac{2}{5} \times \log_2(\frac{2}{5}) - \frac{3}{5} \times \log_2(\frac{3}{5}) = 0.9710$
 - $Thread = 0.9710 - \frac{3}{5} \times 0 - \frac{2}{5} \times 0 = 0.9710$
 - $WhereRead = 0.9710 - \frac{1}{5} \times 0 - \frac{4}{5} \times 0.8113 = 0.3219$
5. Choose attribute: **Thread**.
6. Done.

2 Q2

2.1 Bayesian

According to the description:

- $h = \langle 0.1, 0.2, 0.4, 0.2, 0.1 \rangle$
- $d = \langle \text{lime}, \text{cherry}, \text{cherry}, \text{lime}, \text{lime} \rangle$

hypothesis	h1	h2	h3	h4	h5
$P(\text{lime} \mathbf{h_i})$	0	0.25	0.5	0.75	1
$P(\mathbf{h_i})$	0.1	0.2	0.4	0.2	0.1
$P(\mathbf{d} \mathbf{h_i})$	0	0.578125	0.03125	0.484375	0

To predict the 6th candy, calculate:

$$P(\text{lime}|d) = \sum_i P(\text{lime}|h_i)P(h_i|d) \quad (1)$$

$$= \frac{1}{P(d)} \sum_i P(\text{lime}|h_i)P(d|h_i)P(h_i) \quad (2)$$

$$= \frac{0.25 \times 0.2 \times 0.578125 + 0.5 \times 0.4 \times 0.03125 + 0.75 \times 0.2 \times 0.484375}{0.2 \times 0.578125 + 0.4 \times 0.03125 + 0.2 \times 0.454375} \quad (3)$$

$$= \frac{0.10781175}{0.219} \quad (4)$$

$$= 0.49 \quad (5)$$

$$P(\text{cherry}|d) = 1 - 0.49 = 0.51 \quad (6)$$

The 6th candy is more possibly to be of cherry taste.

2.2 MAP

MAP make prediction based on most probable hypothesis:

$$h_{MAP} = \operatorname{argmax}_{h_i}$$

$$P(h_i|d) = P(X|d) \simeq P(X|h_{MAP})$$

With $d = \langle \text{lime}, \text{cherry}, \text{cherry}, \text{lime}, \text{lime} \rangle$:

Prediction after:

1. lime: $h_{MAP} = h_3, Pr(\text{lime}|h_{MAP}) = 0.5$
2. cherry: $h_{MAP} = h_3, Pr(\text{lime}|h_{MAP}) = 0.5$

3. cherry: $h_{MAP} = h_3, Pr(lime|h_{MAP}) = 0.5$

4. lime: $h_{MAP} = h_3, Pr(lime|h_{MAP}) = 0.5$

5. lime: $h_{MAP} = h_3, Pr(lime|h_{MAP}) = 0.5$

With hypothesis 3, $Pr(lime|h_3) = 0.5 = Pr(cherry|h_3)$

The 6th candy has the same possibility to be of cherry or lime taste.

2.3 ML

ML make prediction based on most probable conditional hypothesis:

$$h_{ML} = \operatorname{argmax}_h P(d|h)$$

With $d = \langle lime, cherry, cherry, lime, lime \rangle$:

Prediction after:

1. lime: $h_{ML} = h_5, Pr(lime|h_{ML}) = 1$

2. cherry: $h_{ML} = h_3, Pr(lime|h_{ML}) = 0.5$

3. cherry: $h_{ML} = h_2, Pr(lime|h_{ML}) = 0.25$

4. lime: $h_{ML} = h_3, Pr(lime|h_{ML}) = 0.5$

5. lime: $h_{ML} = h_4, Pr(lime|h_{ML}) = 0.75$

With hypothesis 4, $Pr(lime|h_4) = 0.75 > Pr(cherry|h_3) = 0.25$

The 6th candy is more possibly to be of lime taste.

3 Q3

According to Naive Bayes Classification, these two rules are followed:

$$P(y|x_1, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i|y)$$

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i|y)$$

Then, build the truth table without term $A = 1, B = 1, C = 1, D = 1$.

A	B	C	D	Value
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0

So with the input $A = 1, B = 1, C = 1, D = 1$:

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i|y) \quad (7)$$

$$= \arg \max_y \{ \quad \quad \quad \} \quad (8)$$

$$P(1) \times P(x_1 = 1|1) \times P(x_2 = 1|1) \times P(x_3 = 1|1) \times P(x_4 = 1|1), \quad (9)$$

$$P(0) \times P(x_1 = 1|0) \times P(x_2 = 1|0) \times P(x_3 = 1|0) \times P(x_4 = 1|0) \} \quad (10)$$

$$= \arg \max_y \left\{ \frac{4}{15} \times \frac{1^4}{2}, \frac{11}{15} \times \frac{5^4}{11} \right\} \quad (11)$$

$$= \arg \max_y \{0.0167, 0.0313\} \quad (12)$$

$$= 0 \quad (13)$$

4 Q4

Since XOR operation is NOT linearly separable, a straight forward neural network without hidden layer will definitely not learn well.

However, AND operation, NOT operation, OR operation is linearly separable, it's possible to build an XOR Neural Network by combining those operations using a hidden layer.

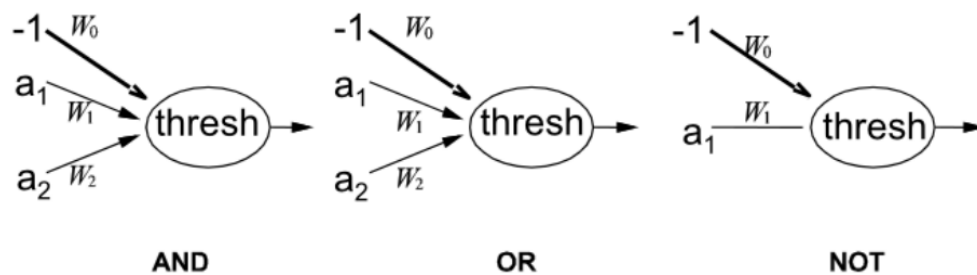


Figure 2: References

For convenience, for sigmoid function: $g(z) = \frac{1}{1 + e^{-z}}$, I define a threshold of 10, which means for z over 10 or less than -10, the value certainly approach 1 or 0.

So, define **NOT** layer, **AND** layer and **OR** layer:

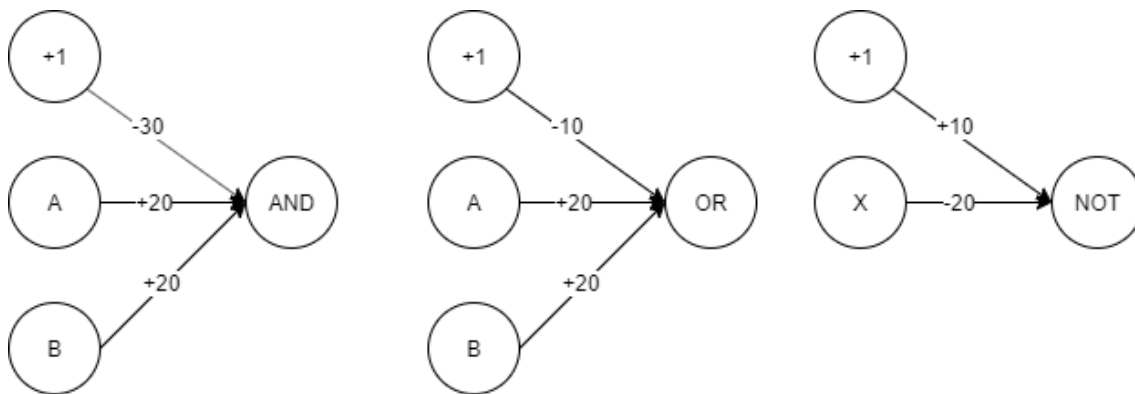


Figure 3: Implementations

And then, build the XOR NN as below:

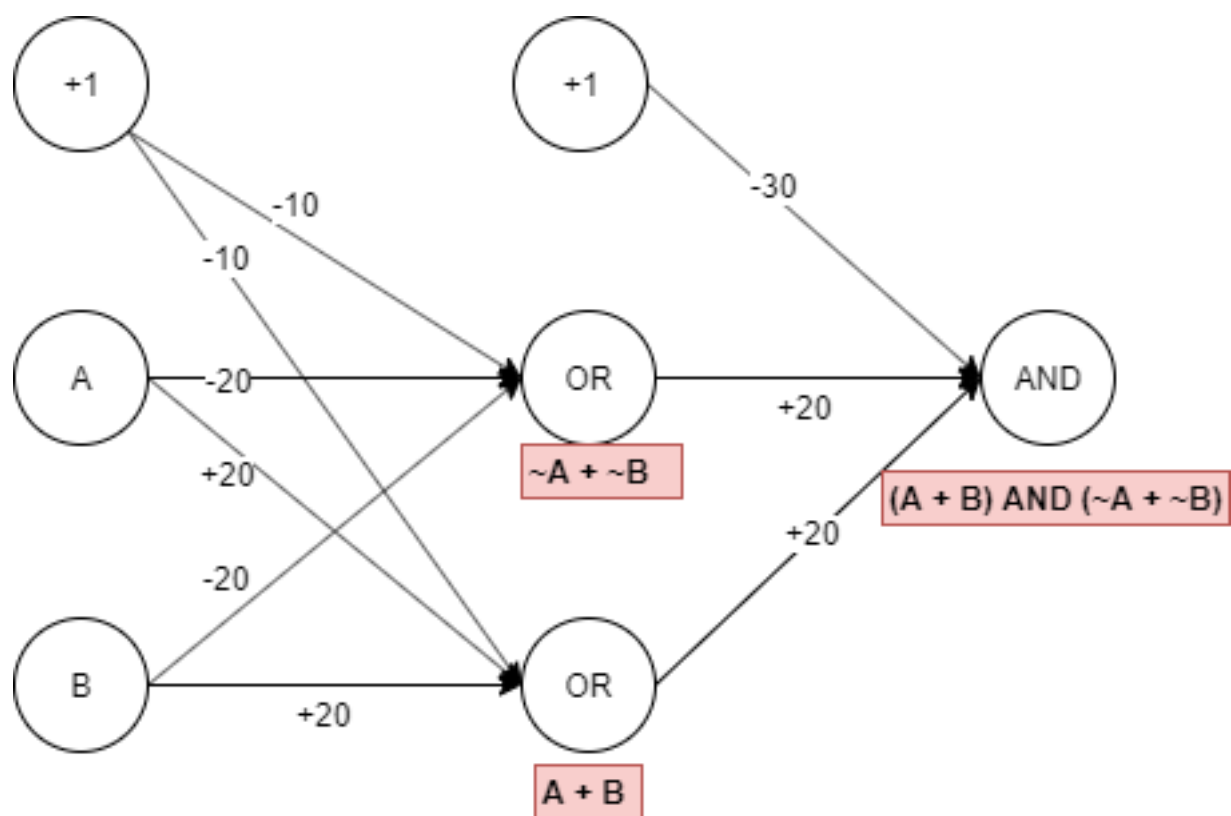


Figure 4: XOR

For a certain input, this model will compute $A \text{ XOR } B$.