텐서플로로 배우는 딥러닝

정보 이론

정보 이론 - 정보량(Quantity of information)

정보량 (Quantity of information)

우리는 삶 속에서 늘 정보를 교환한다. 인간이라면 누구든 사용하는 언어가 그 매체 중 하나이다. 그렇다면, 우리가 질량이나 높이, 속도와 같은 것들을 단위 량으로 측정하 듯 정보 또한 그렇게 측정할 수 있을까? 그때 등장하는 개념이 **"정보량"**이다. 이 정보량의 기원은 어떤 내용을 표현하기 위해 물어야 하는 최소한의 질문 개수에서 출발한다.

예를 들어 이진수 0과 1로 **동전을 5번 던지 결과를 전송**해야 한다고 가정해 보자.

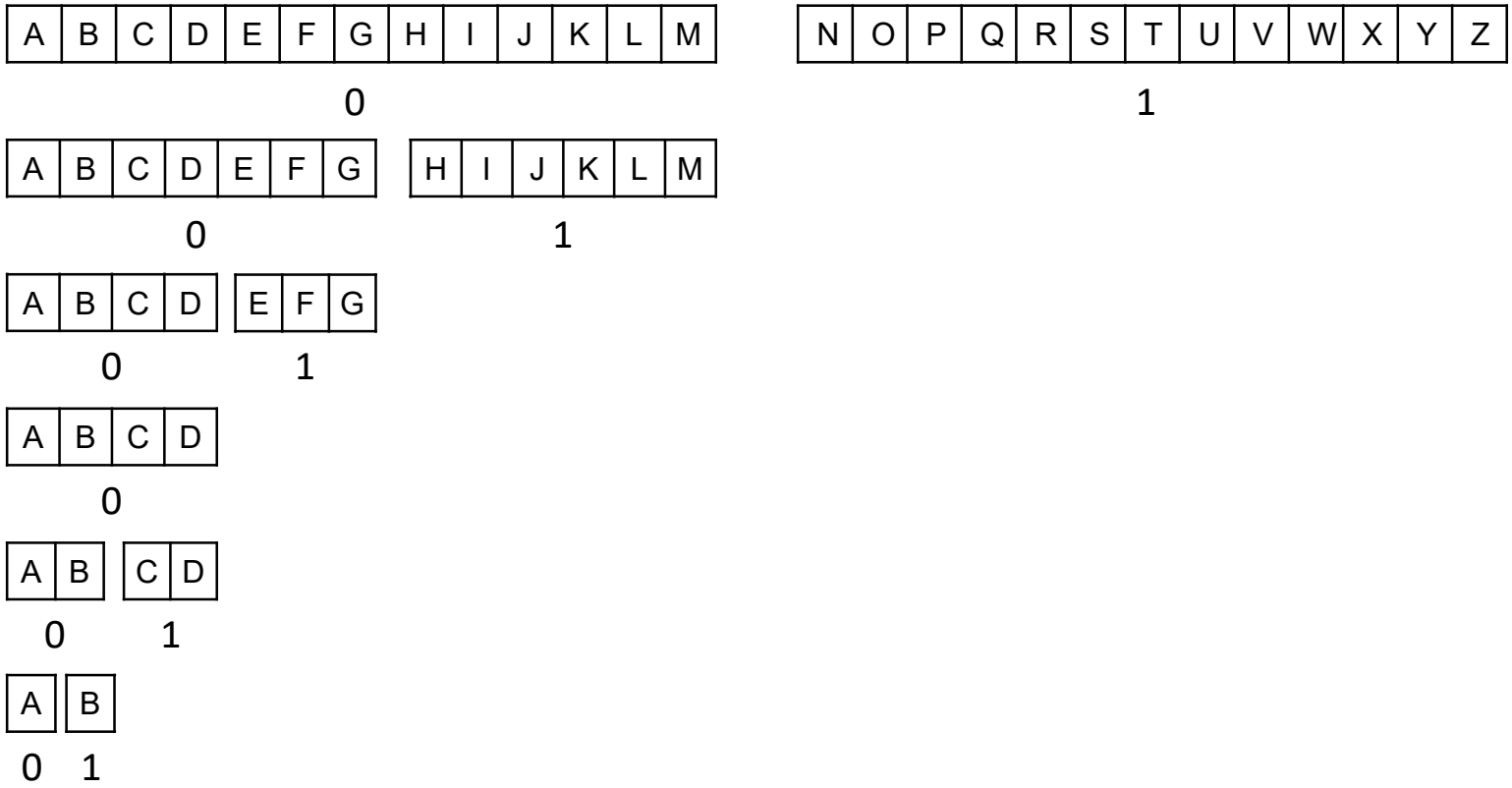
앞면인가? 1을 보내면 앞면이고, 0을 보내면 뒷면이다.

동전을 5번 던지고, 결과를 5번 연결해서 다음과 같이 전송하면 된다.

11010(앞면, 앞면, 뒷면, 앞면, 뒤면)

정보 이론 - 정보량(Quantity of information)

이번엔 알파벳 6개를 보내야 한다고 가정해 보자. 어떻게 알파벳 한 글자를 0과 1로 보낼 수 있을까? 하나의 방법을 예로 들면, 글자가 26개의 알파벳 중에서 앞쪽 절반(A~M)에 속하는지 뒷쪽 절반(M~Z)에 속하는지 확인하는 것이다. 그리고 다시 아래와 같이 여러 번의 질문을 통해 글자 하나를 추려낼 수 있다.



정보 이론 - 정보량(Quantity of information)

$$2^{\text{질문갯수}} = 26$$

$$\text{질문갯수} = \log_2 26$$

$$6 \times 5 = 30$$

$$\text{질문갯수} = \log_2(\text{가능한 결과의 수})$$

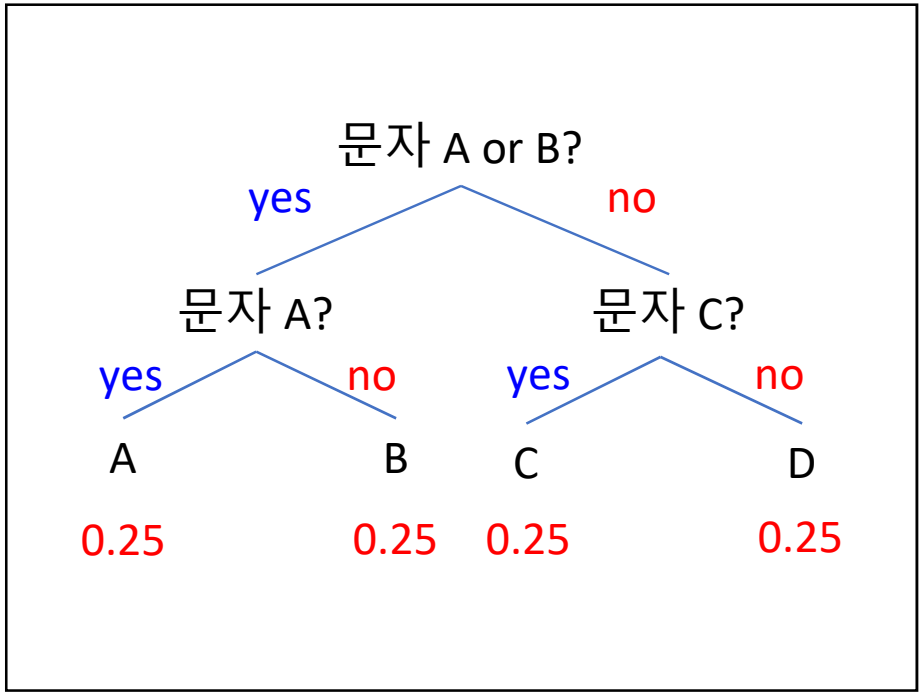
정보 이론 - Entropy

문자열을 출력하는 2대의 머신 X와 Y가 있다.

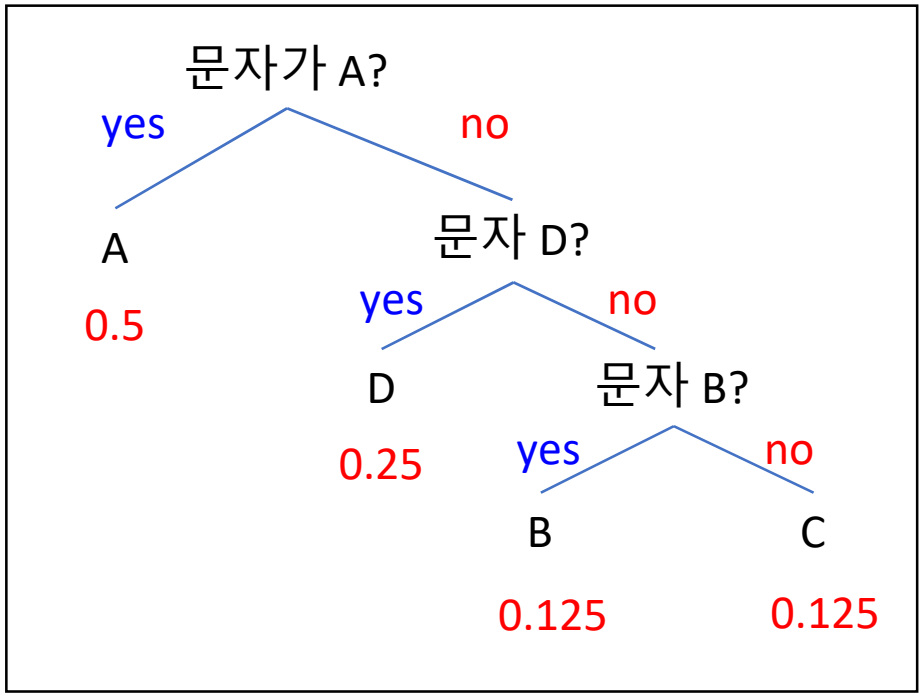
각 머신은 오른쪽의 확률로 문자열을 출력한다.

머신 X는	머신 Y는
A : 0.25	A : 0.5
B : 0.25	B : 0.125
C : 0.25	C : 0.125
D : 0.25	D : 0.25

Machine X



Machine Y

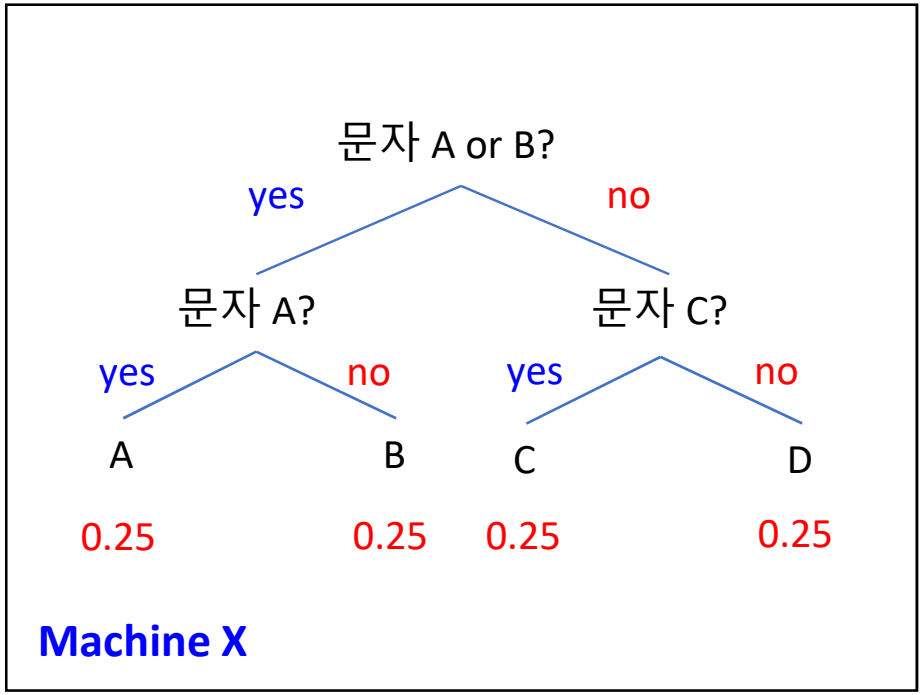


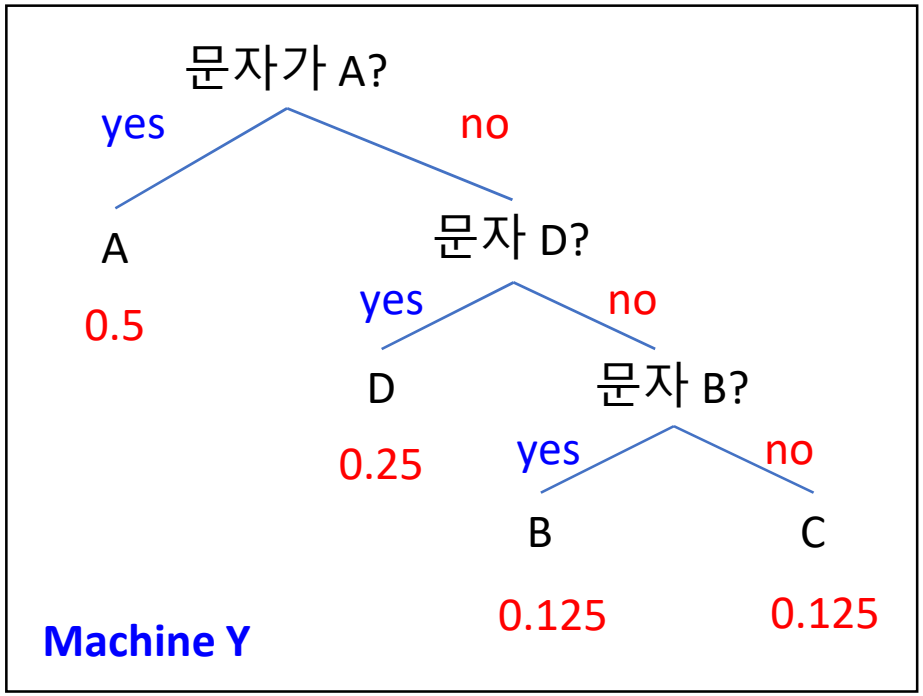
정보 이론 - Entropy

머신 X는 출력하는 문자 1개를 구분하기 위한 최소 질문의 개수는 2개이다.

머신 Y는 A이거나 (B,C,D)이거나로 묻는게 좋다. 이렇게 질문할 경우 최소 질문의 개수는 문자에 따라 달라진다. 이를 수식으로 표현하면 아래와 같다. 즉, 한 글자를 맞추기 위해 1.75 개의 질문이 필요하다.

$$p(A) \cdot 1 + p(B) \cdot 3 + p(C) \cdot 3 + p(D) \cdot 2 = 1.75$$





정보 이론 - Entropy

만약 우리가 각 기계가 출력한 100개의 글자를 맞히기 위해서는 기계 X는 200번, 기계 Y는 175번의 질문을 해야 한다. 그러므로 기계 Y가 X보다 더 적은 정보량을 생산한다고 볼 수 있다. 왜냐하면 불확실성이 더 적기 때문이다. 이를 식으로 정립한 것이 크로드 셰넌(Claude Shannon)이다. 셰넌은 이 불확실성의 측정을 엔트로피(Entropy)라고 불렀다. 이를 H라고 표시하였고, 단위를 bit라고 하였다.

A : 0.5 = 1/2

B: 0.125 = 1/8

C: 0.125 = 1/8

D: 0.25 =1/4

$$\log_2(\frac{1}{1/2}) = 1$$

$$\log_2(\frac{1}{1/8}) = 3$$

$$\log_2(\frac{1}{1/4}) = 2$$

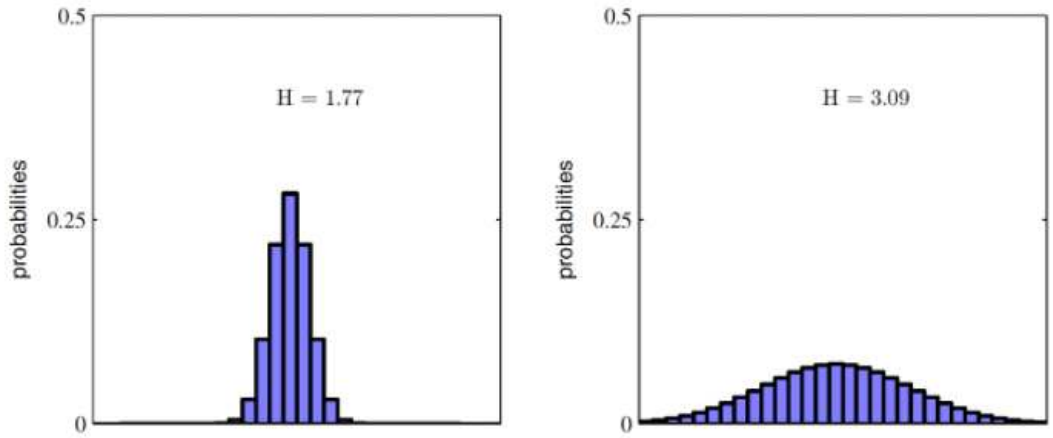
$$H = \sum(\text{사건 발생확률}) \cdot \log_2(\frac{1}{\text{사건 발생확률}})$$

$$= \sum_i p_i \ \log_2(\frac{1}{p_i})$$

$$= -\sum_i p_i \ \log_2(p_i)$$

정보 이론 - Entropy

엔트로피(Entropy)는 가능한 모든 사건이 같은 확률로 일어날 때 그 최댓값을 갖는다. 그 각각의 확률이 모두 동등한 상황에서 조금만 벗어나도 엔트로피는 감소한다. 아래 두 경우를 생각해 보자. 왼쪽의 데이터의 분포가 오른쪽 보다 엔트로피가 작다.



정리하자면, 엔트로피란 최적의 전략 하에서 그 사건을 예측하는 데에 필요한 질문 개수를 의미한다. 다른 표현으로는 최적의 전략 하에서 필요한 질문 개수에 대한 **기댓값**이다. 따라서, 이 entropy가 감소한다는 것은 우리가 그 사건을 맞히기 위해서 필요한 질문의 개수가 줄어드는 것을 의미한다. 질문의 개수가 줄어든다는 사실은 정보량도 줄어든다는 의미이다.

Cross-Entropy

$$H(p,q) = \sum_i p_i \log_2 \frac{1}{q_i}$$
$$= -\sum_i p_i \log_2 q_i$$

p_i가 특정 확률에 대한 참값 또는 목표 확률이고,

q_i가 우리가 현재 학습한 확률 값이다.

예를 들어 여기서는 p=[0.5, 0.125, 0.125, 0.25]이고,

q=[0.25, 0.25, 0.25, 0.25]이면,

따라서, 우리가 어떤 q_i를 학습하고 있는 상태라면,

p_i에 가까워질수록 크로스 엔트로피 값은 작아지게 된다.

이런 특성 때문에 크로스 엔트로피를 머신 러닝에서 많이 쓰는 것이다.

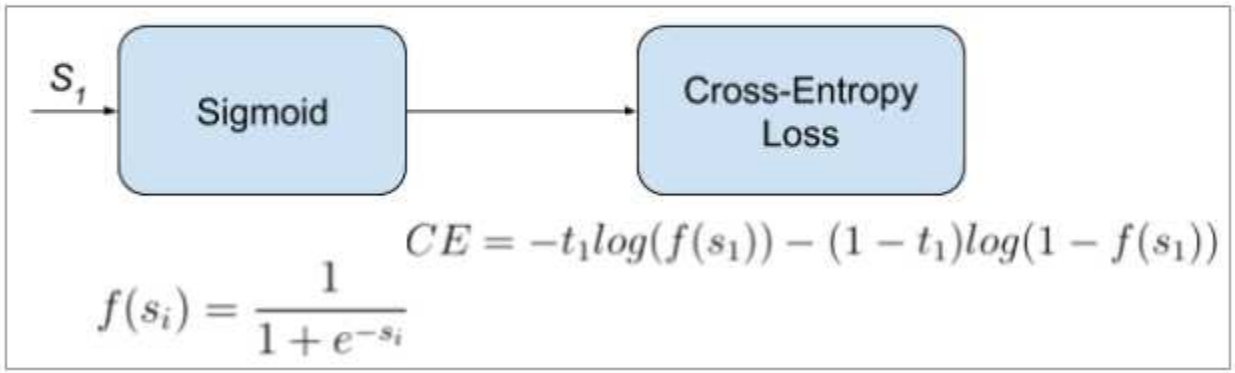
이산형이 아니라 연속형인 확률분포에서는 시그마가 아니라 integral 이다.

$$-\int p(x) \log q(x) dx$$

Binary Cross-Entropy Loss

Sigmoid activation 뒤에 Cross-Entropy loss를 붙인 형태로 주로 사용하기 때문에 Sigmoid CE loss라고도 불린다.

$$CE = -\sum_{i=1}^{C'=2} t_i log(f(s_i)) = -t_1 log(f(s_1)) - (1 - t_1) log(1 - f(s_1))$$



Categorical Cross-Entropy Loss

- Softmax 뒤에 Cross-Entropy loss를 붙인 형태로 주로 사용하기 때문에 Softmax loss 라고도 부른다. → Multi-class classification에 사용
- 우리가 분류 문제에서 주로 사용하는 Loss이다. 분류 문제 에서는 MSE(mean square error) loss 보다 CE loss가 더 빨리 수렴한다는 사실이 알려져 있다. 따라서 multi class에서 클래스를 구분할 때 Softmax와 CE loss의 조합을 많이 사용한다.
- categorical cross-entropy는 분류해야 할 클래스가 3개 이상인 경우, 즉 멀티클래스 분류에 사용됩니다. 라벨이 [0,0,1,0,0], [1,0,0,0,0], [0,0,0,1,0]과 같이 one-hot 형태로 제공될 때 사용된다.