CPSC 532P / LING 530A: Deep Learning for Natural Language Processing (DL-NLP)

Muhammad Abdul-Mageed

muhammad.mageed@ubc.ca

Natural Language Processing Lab

The University of British Columbia

Table of Contents

Many of the current slides are a summary Chapter 3 in Goodfellow et al. (2016). More information can be found therein. Note: The authors credit Pearl (1988) for a lot of the content of the chapter. Other sources used here are credited where approbriate.

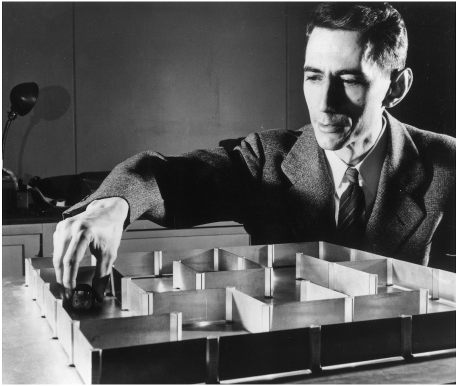

Figure: Claude Shannon. [From Time]. Check about Claude Shannon, e.g. short documentary [here] & lecture by Robert G. Gallager [here].

Information: A Book

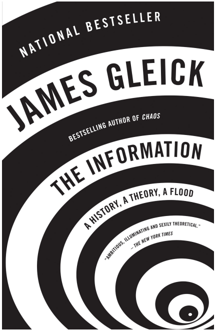

Figure: **Blurb**: A fascinating intellectual journey through the history of communication and information, from the language of Africa's talking drums to the invention of written alphabets; from the electronic transmission of code to the origins of information theory, into the new information age and the current deluge of news, tweets, images, and blogs...

Information Theory

What is information theory?

- Focused on **quantifying how much information is present in a signal**
- Originally **invented to study sending messages from discrete alphabets over a noisy channel**
- Communication via **radio transmission** is an example
- Answers **how to design optimal codes**
- Tells **how to calculate the expected length of messages** sampled from specific probability distributions

Information Theory: Basic Intuition

Intuition

- Learning that an **unlikely event** has occurred is **more informative** than learning that a likely event has occurred.
- “The sun rose this morning": **not informative enough** to send as a message
- “There was a solar eclipse this morning": **very informative**

Quantifying Information

Goal: Quantify Info. Such That:

- **Likely events**: have **low information content**, **events guaranteed to happen**: **no information content**
- **Less likely events**: higher information content.
- **Independent events**: have **additive information**. `Finding out that a tossed coin has come up as heads twice conveys twice as much information as finding out that a tossed coin has come up as heads once.`

Self-Information of Event X=x

- **Self-information** deals only with **a single outcome**.
- It is the **surprise** when a random variable is sampled.

1: Self-Information of Event X=x

$$\mathrm{I}(x) = -\log_2 P(x)$$

Example of Self-Information

- When we toss a fair coin, P(x="head"=0.5),
  $\mathrm{I}(x = 0.5) = -\log_2 P(0.5) = 1$ **bit** of information.
- **Note**: If we use base e, then the unit of measurement is **nats**. (Above gives ~ 0.693 nats).
- **Try it Python**: Base 2: -math.log(0.5,2); Base e: -math.log(0.5).

Surprisal of Document Language I

Figure: Probability of document language in and English& Russian collection.

Example: I(X) For Document Language

- If we know the language of the Russian document, we would have a **surprisal of 2 bits** ($-log_2(0.25)$)
- For English, we will have a **surprisal of 0.4150 bits** ($-log_2(0.75)$) (We expect most docs to be in English, 75%).
- On avg., we get (0.25 * 2) + (0.75 * 0.4150) = 0.81125 bits of information. **This is entropy!** (**Average surprise**)...

Example: I(X) For Document Language

- On avg., we get $(0.25 * 2) + (0.75 * 0.4150) = 0.81125$ bits of information. **This is entropy!**
- **Entropy measures the average amount of information we get from one sample from a given probability distribution P.**
- Equivalently, it tells us how **uncertain** we are.

Shannon Entropy

- The **Shannon entropy** of a distribution is the expected amount of information in an event drawn from that distribution.
- Thus allows us to quantify the amount of uncertainty in an entire probability distribution. (See below, but also denoted $H(P)$)

2: Shannon entropy

$$\textbf{Recall: self\_info.}: \mathrm{I}(x) = -\log_2 P(x)$$

$$\mathrm{H}(x) = \mathbb{E}_{x \sim P}\big[\mathrm{I}(x)\big] = -\mathbb{E}_{x \sim P}\big[\log_2 P(x)\big]$$

- Gives a **lower bound on the number of bits** (or nats) needed on avg to encode symbols drawn from a distribution P.
- **Nearly deterministic distributions**: have low entropy;
- Distributions **closer to uniform**: high entropy (i.e., high uncertainty).

Example on Entropy

3: Entropy (Expectation Re-Written)

$$\mathbb{E}(f(x)) = \sum_{x \in X} f(x)P(x)$$

$$\mathrm{H}(x) = -\sum_{x \in X} P(x) \log_2 P(x)$$

- Entropy is always **greater than or equal to zero**: $H(X) \geq 0$.
- Let **x**={1,2,3}, with P=(1/2, 1/4, 1/4). What is H(X)?

Example: H(X)

- **Note 1**: $-\log_2(1/2) = -(\log_2(1) - \log_2(2)) = -(0-1) = 1$
- **Note 2**: $-\log_2(1/4) = -(\log_2(1) - \log_2(4)) = -(0-2) = 2$
- **H(X)**= $-1/2 \log_2(1/2) - 1/4 \log_2(1/4) - 1/4 \log_2(1/4) = 3/2$

Remarks on Entropy

- **Recall**: $log_2(0) = -\infty$
- For any element of x_i for which $p(x_i) = 0$, we will get $0 * -\infty$ (undefined).
- In this case, we define $0 * log_2(0) = 0$.
- (**Recall**: $\lim_{x \to 0} = x \log_2(x) = 0$)

Kullback–Leibler Divergence (KL Divergence) I

KL Divergence

- **Measures how one probability distribution is different from a second probability distribution**.
- **Always greater than or equal to zero**
- A smaller KL divergence value means we can expect more similar behavior of the two distributions.

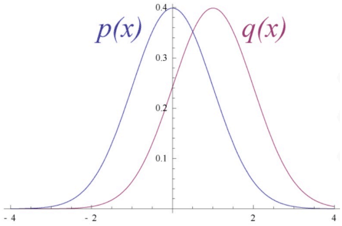

Figure: [From Wikipedia].

KL Divergence II

- With two prob distributions $P(x)$ and $Q(x)$ over the same r.v. x

4: KL Divergence

$$\mathrm{D}_{KL}(P||Q) =$$

$$\mathbb{E}_{x\sim P}\left[\log_2 \frac{P(x)}{Q(x)}\right] = \mathbb{E}_{x\sim P}\left[\log_2 P(x) - \log_2 Q(x)\right].$$

- For discrete variables, it is *the extra amount of info. needed* to send a message containing symbols drawn from prob distrib P, when we use a code designed to *minimize* the len of messages drawn from distrib Q.

Properties of KL Divergence

- KL divergence is **non-negative**.
- KL divergence is **not symmetric** (i.e., $D_{KL}(P||Q) \neq D_{KL}(Q||P)$ (and so it is not a measure of distance).
- The **KL divergence is 0 if and only** if P and Q are the same distribution in the case of discrete variables, or equal "almost everywhere" in the case of continuous variables.

KL Divergence for SRL I

- We are interested in **how much information a given verb can tell us about the possible semantic class of its argument**.
- This is useful for **semantic role labeling**.

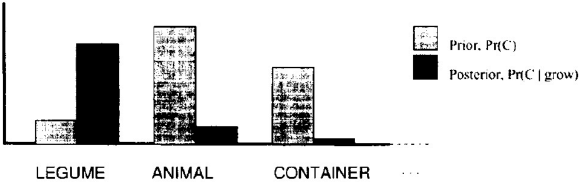

Fig. 1. Example of a prior distribution and a posterior distribution.

Figure: [From Resnik, 1996]

KL Divergence for SRL II

- We have **semantic class** c (e.g., "ANIMAL") to which an object argument of a verb v belongs.
- **P(c|v) is the posterior** and **Q(c) is the prior** of class c.
- Use KL-Divergence to calculate the **selectional preference strength** of a given verb v.

5: KL Divergence for SRL

$$S_R(v) = \sum_c P(c|v) \log_2 \frac{P(c|v)}{Q(c)}$$

KL Divergence for SRL III

```
import math
P_legume_given_grow= .80
P_animal_given_grow= .15
P_container_given_grow= .05
Q_legume = .20
Q_animal = .50
Q_container = .30
kl_c1=P_legume_given_grow  * math.log(P_legume_given_grow/Q_legume)
kl_c2=P_animal_given_grow  * math.log(P_animal_given_grow/Q_animal)
kl_c3=P_container_given_grow  * math.log(P_container_given_grow/Q_container)
KL=kl_c1+kl_c2+kl_c3
print(KL)
```

```
0.838851594786
```

```
print("kl_c1 % 1.4f"% kl_c1)
print("kl_c2 % 1.4f"% kl_c2)
print("kl_c3 % 1.4f"% kl_c3)

```

```
kl_c1  1.1090
kl_c2 -0.1806
kl_c3 -0.0896
```

Figure: [From Resnik, 1996. Note: Resnik's model proceeds with further steps beyond what is shown here.]

KL (Q||R) for SRL

Reversed: KL (Q||P)

```
kl_c1_r=Q_legume  * math.log(Q_legume/P_legume_given_grow)
kl_c2_r=Q_animal  * math.log(Q_animal/P_animal_given_grow)
kl_c3_r=Q_container  * math.log(Q_container/P_container_given_grow)
KL_R=kl_c1_r+kl_c2_r+kl_c3_r
print(KL_R)
```

```
0.862255370707
```

```
print("kl_c1_r % 1.4f"% kl_c1_r)
print("kl_c2_r % 1.4f"% kl_c2_r)
print("kl_c3_r % 1.4f"% kl_c3_r)
```

```
kl_c1_r -0.2773
kl_c2_r  0.6020
kl_c3_r  0.5375
```