WestlakeNLP

Natural Language Processing

Yue Zhang
Westlake University

西湖大學
WESTLAKE UNIVERSITY

WestlakeNLP

Chapter 5

Using Information Theory

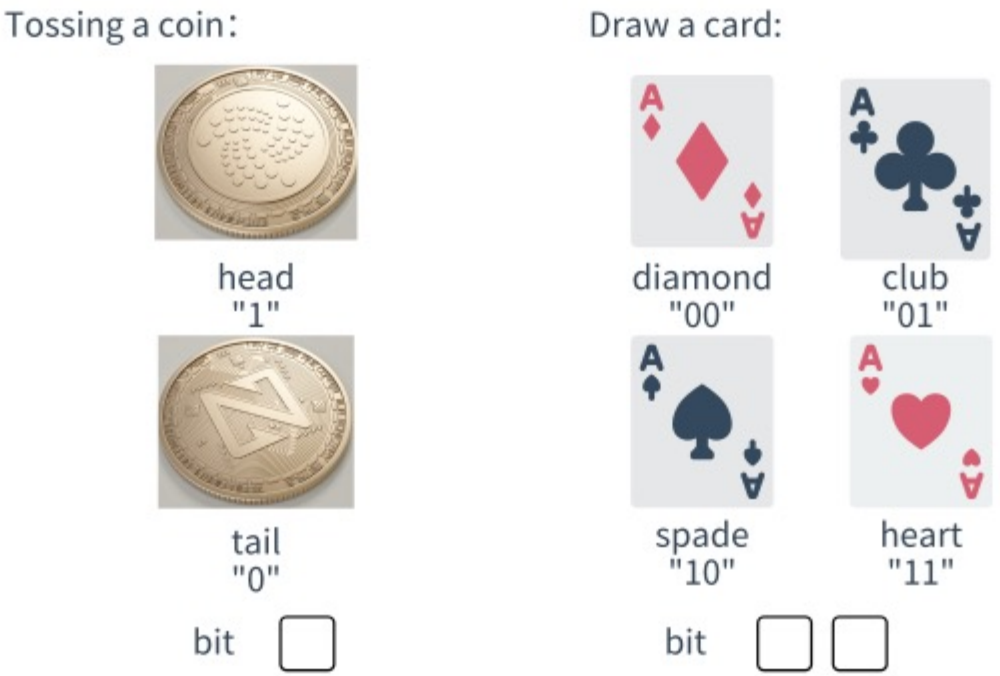
Contents

WestlakeNLP

Contents

What is information?

- Resolve uncertainty about random events.



- To learn the outcome of a random event with n equally possible results, $log_2 n$ bits of information is necessary.

Information of event.

- Original uncertainty ---- remaining uncertainty

- Spade Ace: $\log_2 52 - \log_2 1 = \log_2 52$
- Spade: $\log_2 52 - \log_2 13 = 2$ bits
- Ace: $\log_2 52 - \log_2 4 = \log_2 13$ bits

Information for non-uniform distributions

m red balls and n green balls in a box

Red ball:

$$\log_2(m+r) - \log_2 n$$
$$= \log_2 \frac{m+n}{n} = \log_2 \frac{1}{P(red)}$$

- The outcomes with higher probabilities contains less information.
- For a certain outcome r_i
  - Probability: $P(r_i)$
  - Information received:$-log_2 P(r_i)$.

Entropy

- Entropy analyzes information concerning events by considering all possible outcomes or random variables by considering all possible values.

- The entropy of distribution P is:

$$H(P) = -\sum_{i=1}^{n} P(r_i) log_2 P(r_i) = E\left(log_2 \frac{1}{P(r_i)}\right)$$

  - where E denotes a probability-weighted average, or the **mathematical expectation**

Entropy and distribution characteristics

WestlakeNLP

- Encoding six numbers

  - Uniform distribution

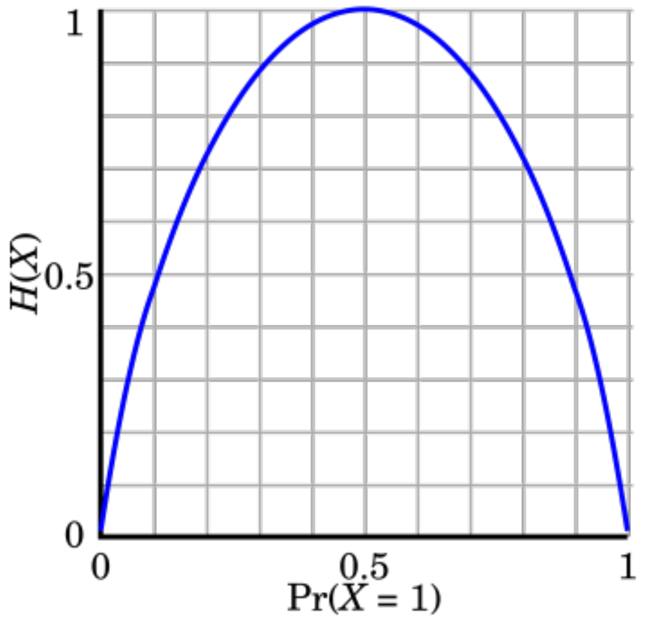
    3 bits 001 010 011 100 101 110

  - 90% case with number 1, and 2% with 2, 3, 4, 5, 6

    0 1010 1011 1110 1101 1110

    $0.9 \times 1 + 0.1 \times 4 = 1.3$ bits

Entropy



binary entropy for tossing a coin

the expected surprisal

X=1 represents a result of heads

Entropy H(X) of a coin flip corresponds to probability Pr(X)

- Events with uniform output distributions have the largest entropy.
- The more uneven the distribution is, the smaller the entropy is.

Contents

Occam's razor

WestlakeNLP

- A principle attributed to the fourteenth-century English Franciscan friar William of Ockham, which states that *"entities should not be multiplied beyond necessity"*

- **Occam's razor** shares underlying similarities with the **maximum entropy principle.**

A naive maximum entropy model

A probabilistic model for a random event e with possible outcomes $r_1, r_2, \ldots, r_M$:

$$\hat{P} = argmaxH(P) = argmax - \sum_{i=1}^{M} P(r_i) \log_2 P(r_i)$$

using $P(r_i)$ directly as parameters.

The training objective is to find:

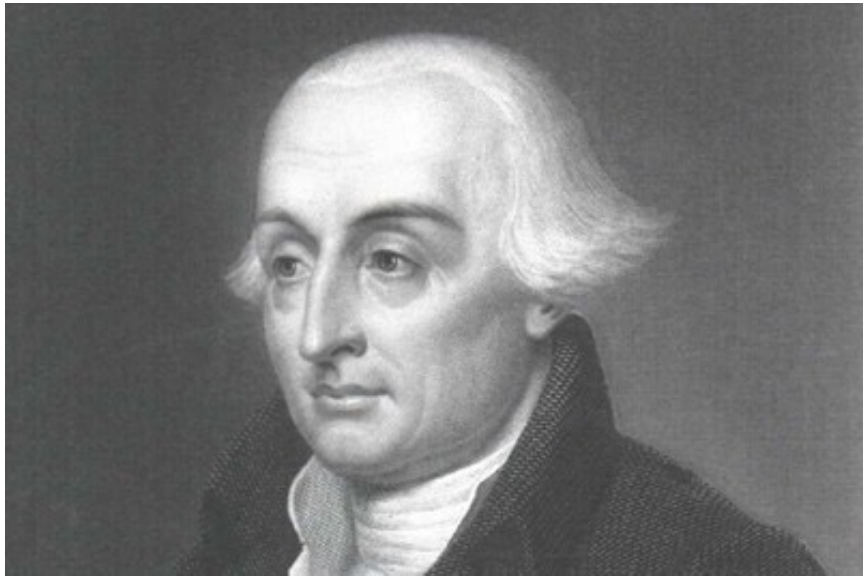
$$\hat{P}(e) = \arg\max H\,(e) = \arg\min \sum_{i=1}^{M} P(r_i) \log_2 P(r_i)$$

under the constraint that

$$\sum_{i=1}^{M} P(r_i) = 1$$

A naive maximum entropy model

Taking each $P(r_i)$ as a separate variable, we use **Lagrange multipliers** to do the optimization.



In mathematic optimization, the *method of Lagrange multipliers* is a strategy for finding the local maxima and minima of a function subject to equality constraints.

A naive maximum entropy model

- The Lagrangian equation is

$$\Lambda(P(r_1), P(r_2), \dots, P(r_M), \lambda) = \sum_{i=1}^{M} P(r_i) \log_2 P(r_i) + \lambda\left(\sum_{i=1}^{M} P(r_i) - 1\right),$$

where λ is a Lagrangian multiplier.

- A necessary condition for optimality in the constrained problem is that

$$\frac{\partial \Lambda}{\partial P(r_i)} = 0 \text{ for } i \in [1 \dots m] \quad => \quad 1 - \log_2 P(r_i) + \lambda = 0$$

which suggests that $P(r_1) = P(r_2) = \cdots = P(r_M)$

- The conclusion conforms the fact that the uniform distribution contains the most uncertainty.

Contents

WestlakeNLP

Conditional entropy

WestlakeNLP

For a conditional probability distribution $P(y|x)$, given that the random event X follows a probability distribution $P(x)$, the conditional entropy value:

$$H(Y|X) = -\sum_x \sum_y P(x)P(y|x)\log_2 P(y|x)$$

$$= -\sum_x \sum_y P(x,y)\log_2 P(y|x)$$

Conditional entropy

For a conditional probability distribution $P(y|x)$, given that the random event X follows a probability distribution $P(x)$, the conditional entropy value:

$$H(Y|X) = -\sum_x \sum_y \sum\_x P(x) P(y|x) \log_2 P(y|x)$$

$$= -\sum_x \sum_y P(x,y) \log_2 P(y|x)$$

Intuition

- Given x, $H(Y|x) = -\sum_y P(y|x) \log_2 P(y|x)$
- Expectation $H(Y|X) = \sum_x P(x)\, H(Y|x)$

Maximum entropy model and training data

WestlakeNLP

- We are to derive a maximum entropy model for feature-based discriminative classification.
  - Training data : $D = \{(x_i, y_i)\}|_{i=1}^{N}$
  - Feature instances for (x,y) : $f_1, f_2, \ldots, f_m$
  - Feature instance : $f_i(x, y)$ (count)

Notations

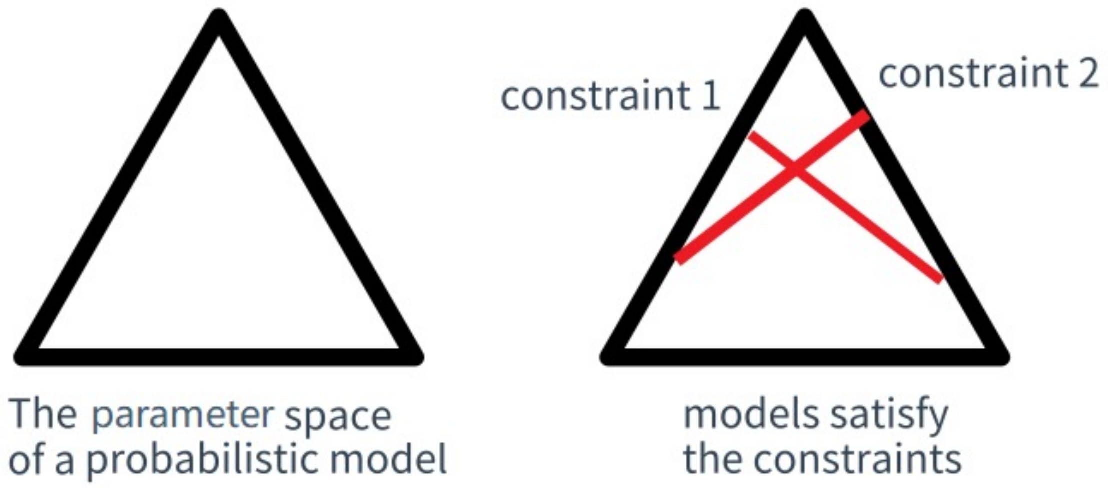
- The model to build : $P(y \mid x)$
- Prior distribution of x : $P(x)$
- Empirical distribution : $\tilde{P}(x) = \frac{\#x}{\sum_{\{x' \in D\}} \#x'} = \frac{\#x}{|D|}$
- Model expectation of f_i : $E(f_i)$
- Empirical count of f_i: $\tilde{E}(f_i)$

Modelling the problem

- Objective function : $H(P)$
- Goal : among all distributions that satisfy the constraints, choose the one $\hat{P}$ that maximizes $H(P)$

$$\hat{P} = argmaxH(P)$$

- Constraints: feature counts.



Objective

- The conditional entropy to maximize is
  $H(Y|X) = -\sum_x \sum_y P(x)P(y|x) \log_2 P(y|x)$

- We use $\tilde{P}(x) = \frac{\#x}{\sum_{x' \in D} \#x'} = \frac{\#x}{|D|}$ to represent $P(x)$, which is typically $\frac{1}{|D|}$.

- resulting in $H(Y|X) = -\sum_x \sum_y \tilde{P}(x)\, P(y|x) \log_2 P(y|x)$, which we maximize

Constraints

WestlakeNLP

- Model's feature expectation = observed feature expectation $E(f_i) = \tilde{E}(f_i)$

- The *model* feature

$$E(f_i) = \sum_{j=1}^{|D|} \tilde{P}(x_j) \sum_{y} P(y|x_j) f_i(x_i, y)$$

typically $\frac{1}{|D|} \sum_{j=1}^{|D|} \sum_y P(y|x_j) f_i(x_i, y)$.

- The *empirical* feature

$$\tilde{E}(f_i) = \sum_{j=1}^{|D|} \tilde{P}(x_j, y_j) f_i(x_j, y_j) \quad \left((x_j, y_j) \in D\right)$$

typically $\frac{1}{|D|} \sum_{j=1}^{|D|} f_i(x_j, y_j)$.

- One additional constraint, as before $\sum_y P(y|x) = 1$.

Using Lagrangian multipliers

WestlakeNLP

- A model $\hat{P}(y|x)$ that satisfies

$$\hat{P}(y|x) = \arg min \sum_{x} \sum_{y} \tilde{P}(x)\, P(y|x) \log_2 P(y|x)$$

s.t. for all $i, E(f_i) = \tilde{E}(f_i); \sum_y P(y|x) = 1$

- The Lagrangian equation is

$$\Lambda(P, \vec{\lambda}) = -H(Y|X) + \sum_{i=1}^{m} \lambda_i \left(E(f_i) - \tilde{E}(f_i) \right) + \sum_{x} \lambda_{m+1}^{x} \left(\sum_{y} P(y|x) - 1 \right)$$

Maximum entropy leads to log-linear models

WestlakeNLP

- A necessary condition to the constrained minimum value of *-H(Y | X)* is $\frac{\partial \Lambda}{\partial P} = 0$

- Solving these equations, we have

$$P(y|x) = \frac{\exp\left(\sum_i \lambda_i f_i(x,y)\right)}{\sum_{y'} \exp\left(\sum_i \lambda_i f_i(x,y')\right)}$$

- This is a log-linear form of *P(y | x)* using the maximum entropy principle, which is the same as a log linear model

- We further find $\vec{\lambda}$ via $\vec{\lambda} = \arg\min_{\vec{\lambda}} \Lambda(P, \lambda) = \arg\min_{\vec{\lambda}} - \sum_j P(y_i \,|x_i\,)$, which is the same as MLE.

Contents

KL-divergence

- **Revisit risk.** Parameter $\vec{\theta}$, data $D = \{d_i\}|_{i=1}^{N}$

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} loss\left(\vec{\theta} \cdot \vec{\phi}(d_i)\right)$$

KL-divergence

- **Revisit risk.** Parameter $\vec{\theta}$, data $D = \{d_i\}|_{i=1}^{N}$

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} loss\left(\vec{\theta} \cdot \vec{\phi}(d_i)\right)$$

- **For a probabilistic model**

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} diff(\tilde{P}(d_i),\ Q(d_i))$$

where $\tilde{P}(d_i)$ is data frequency , $Q(d_i)$ is model probability.

KL-divergence

WestlakeNLP

- **Revisit risk.** Parameter $\vec{\theta}$, data $D = \{d_i\}|_{i=1}^{N}$

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} loss\left(\vec{\theta} \cdot \vec{\phi}(d_i)\right)$$

- **For a probabilistic model**

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} diff(\tilde{P}(d_i),\ Q(d_i))$$

where $\tilde{P}(d_i)$ is data frequency , $Q(d_i)$ is model probability.

- *diff* **can be defined**

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N}\left(\log_2 \tilde{P}(d_i) - \log_2 Q(d_i)\right)$$

$$= \sum_{i=1}^{N} \tilde{P}(d_i)\left(\log_2 \tilde{P}(d_i) - \log_2 Q(d_i)\right) = \sum_{i=1}^{N} \tilde{P}(d_i) \log_2 \frac{\tilde{P}(d_i)}{Q(d_i)}$$

KL-divergence

WestlakeNLP

- **Revisit risk.** Parameter $\vec{\theta}$, data $D = \{d_i\}|_{i=1}^{N}$

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} loss\left(\vec{\theta} \cdot \vec{\phi}(d_i)\right)$$

- **For a probabilistic model**

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N} diff(\tilde{P}(d_i),\ Q(d_i))$$

where $\tilde{P}(d_i)$ is data frequency , $Q(d_i)$ is model probability.

- *diff* **can be defined**

$$\widetilde{risk}(\vec{\theta}) = \frac{1}{N}\sum_{i=1}^{N}\left(\log_2 \tilde{P}(d_i) - \log_2 Q(d_i)\right)$$

$$= \sum_{i=1}^{N} \tilde{P}(d_i)\left(\log_2 \tilde{P}(d_i) - \log_2 Q(d_i)\right) = \boxed{\sum_{i=1}^{N} \tilde{P}(d_i) \log_2 \frac{\tilde{P}(d_i)}{Q(d_i)}}$$

KL-divergence

- **Kullback-Leibler (KL) divergence** measures how different two distributions of the same random variable are.

$$KL(P,Q) = \sum_{i=1}^{M} P(z_i) \log_2 \frac{P(z_i)}{Q(z_i)} = E_{e \sim P(e)} \log_2 \frac{P(e)}{Q(e)}$$

- Not symmetric --- how different is Q according to P.

- Can measure a probabilistic model against a data distribution.
  $KL(P,Q) \geq 0$, $KL(P,Q) = 0$, where $P = Q$.

KL-divergence

- **Loss function:**

$$KL(P,Q) = \sum_{i=1}^{N} \tilde{P}(d_i)\Big(\log_2 \tilde{P}(d_i) - \log_2 Q(d_i)\Big)$$
$$= \sum_{i=1}^{N} \tilde{P}(d_i)\log_2 \tilde{P}(d_i) - \sum_{i=1}^{N} \tilde{P}(d_i)\log_2 Q(d_i)$$

- As the first term is constant, the loss effectively maximizes

$$\sum_{i=1}^{N} \tilde{P}(d_i)\log_2 Q(d_i) = \frac{1}{N}\sum_{i=1}^{N} \log_2 Q(d_i)$$

which is the exactly log-likelihood of the dataset *D*, namely **MLE.**

Contents

WestlakeNLP

Cross entropy

- The second term of KL-divergence is referred to **cross-entropy:**

$$H(P,Q) = -\sum_{i=1}^{M} P(r_i)\log_2 Q(r_i) = E_{e\sim P}\log_2 \frac{1}{Q(e)}$$

It also measures the similarity between two distributions of the **same** random variable.

Cross entropy

- The second term of KL-divergence is referred to **cross-entropy:**

$$H(P,Q) = -\sum_{i=1}^{M} P(r_i)\log_2 Q(r_i) = E_{e\sim P}\log_2\frac{1}{Q(e)}$$

It also measures the similarity between two distributions of the **same** random variable.

- Intuitively, it means the number of bits to encode a variable e distributed in Q using the encoding scheme defined by P.

Thus $H(P,Q) = H(P)$ if $Q = P$ and is larger when Q differs more form P.

Cross entropy

WestlakeNLP

- The second term of KL-divergence is referred to **cross-entropy:**

$$H(P,Q) = -\sum_{i=1}^{M} P(r_i)\log_2 Q(r_i) = E_{e\sim P}\log_2 \frac{1}{Q(e)}$$

It also measures the similarity between two distributions of the **same** random variable.

- Intuitively, it means the number of bits to encode a variable e distributed in Q using the encoding scheme defined by P.

Thus $H(P,Q) = H(P)$ if $Q = P$ and is larger when Q differs more form P.

- KL-divergence is non-negative because:

$$KL(P,Q) = \sum_{i=1}^{N} \tilde{P}(d_i)\log_2 \tilde{P}(d_i) - \sum_{i=1}^{N} \tilde{P}(d_i)\log_2 Q(d_i) = H(P,Q) - H(P)$$

As a result, KL-divergence is also called **relative entropy.**

Cross entropy loss

WestlakeNLP

- Cross-entropy: $$H(P, Q) = -\sum_{i=1}^{M} P(r_i) \log_2 Q(r_i)$$

- Cross-entropy loss:

$$H(\tilde{P}, Q) = -\sum_{i=1}^{N} \tilde{P}(d_i) \log_2 Q(d_i) = -\frac{1}{N} \sum_{i=1}^{N} \log_2 Q(d_i)$$

where $\tilde{P}(d_i)$ is data frequency, $Q(d_i)$ is model probability.

- The same as negative log-likelihood loss.

Contents

WestlakeNLP

Perplexity

- Formally $\Upsilon(P) = 2^{H(P)} = 2^{-\sum_i P(z_i)\log_2 P(z_i)}$

- Intuitively, **perplexity** represents the *expected* number of bits necessary for encoding each outcome.

Model Perplexity

WestlakeNLP

- Cross-entropy can also be used as the power term for calculating perplexity.
- This can be useful for model evaluation.

$$\Upsilon(Q,D) = 2^{H(\tilde{P}(d),Q)} = 2^{-\sum_{i=1}^{N} \tilde{P}(d_i)\log_2 Q(d_i)} = 2^{-\frac{1}{N}\sum_{i=1}^{N}\log_2 Q(d_i)}$$

(Model perplexity)

where $\tilde{P}(d_i)$ is data frequency, $Q(d_i)$ is model probability.

Evaluating language models

WestlakeNLP

- For classification, accuracy is the metric.
- For language modeling, no single correct answer.
  - For sentence level, $2^{-\frac{1}{N}\sum_{i=1}^{N}\log_2 Q(s_i)}$, typically 2^{190}.
  - A commonly used evaluation metric for language models is **per-word perplexity**:

$$2^{-\frac{1}{|D|}\sum_{i=1}^{|D|}\log Q(w_i)}$$

typically 10-250.

Contents

Entropy, Cross-Entropy, Mutual Information

WestlakeNLP

- Entropy – the *expected* number of bits to encode a random variable.
  The number of bits in average when encoding many outcome values.
  Optimal encoding scheme.

- Cross-Entropy – the *expected* number of bits to encode a random variable using a different encoding scheme.
  Two distributions concerning the same random variable.
  - Evaluate model distribution against data distribution
  - Calibrate model distribution against data.

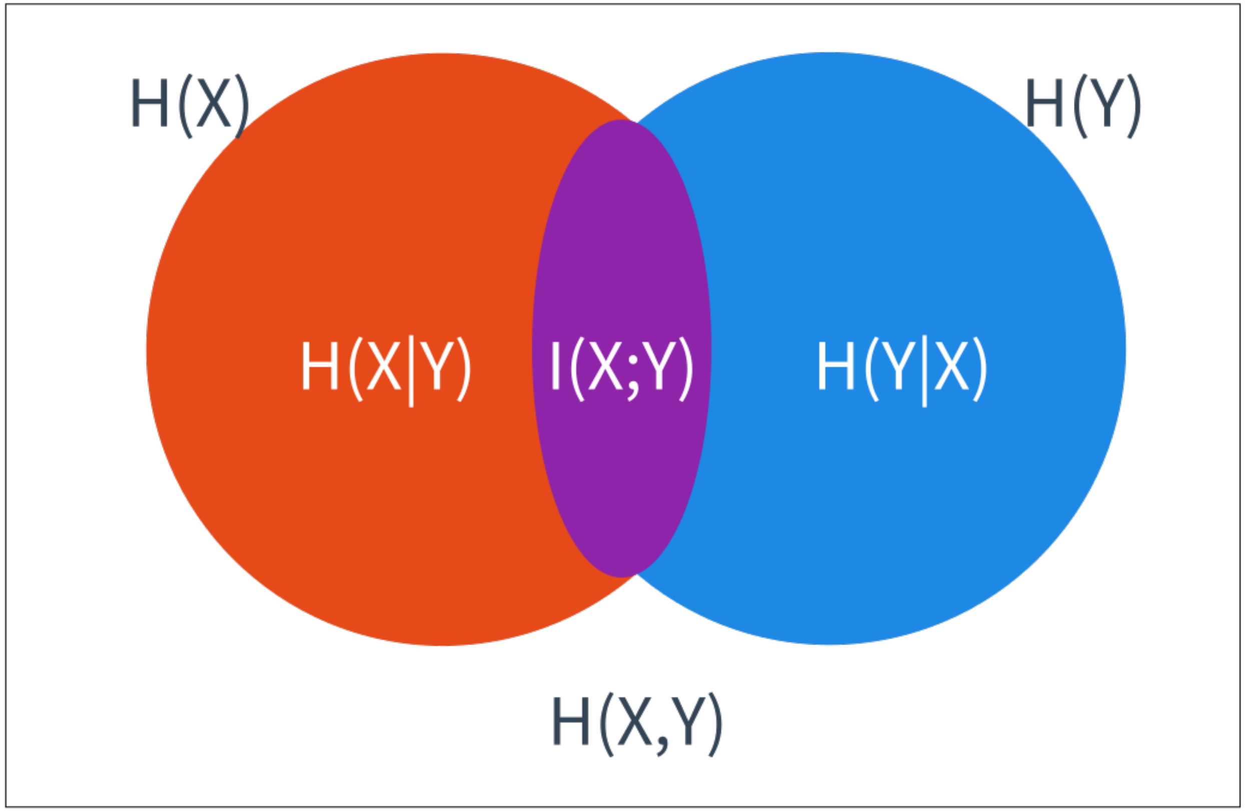
- Mutual Information – the *expected* number of bits you can save for encoding a random variable if a second random variable is known.
  About two random variables.

Mutual information

WestlakeNLP

- Measures the correlation between two different random variables *X* and *Y*.
- The difference between $H(Y)$ and $H(Y \mid X)$ is called the **mutual information** between *X* and *Y*, denoted as $I(X,Y)$
- $$\begin{aligned} I(X, Y) &= H(Y) - H(Y|X) \\ &= \sum_{x,y} P(x,y) \log_2 P(y|x) - \sum_y P(y) \cdot \log_2 P(y) \\ &= \sum_{x,y} P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)} \end{aligned}$$
- It measures the number of bits we can save for encoding *Y*, if *X* is known.

Conditional entropy



Contents

WestlakeNLP

Pointwise mutual information

WestlakeNLP

- Given two random events X and Y, their mutual information can be viewed as the expectation of $\log_2 \frac{P(x,y)}{P(x)P(y)}$ over all x, y:

$$I(x,y) = \sum_{x,y} P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)} = E_{x,y}\left(\log_2 \frac{P(x,y)}{P(x)P(y)}\right)$$

- For each outcome pair (x,y), $\log_2 \frac{P(x,y)}{P(x)P(y)}$ is called Pointwise Mutual information (PMI) between x and y.

Pointwise mutual information

WestlakeNLP

$$\log_2 \frac{P(x,y)}{P(x)P(y)}$$

- PMI represents the statistical correlation between two values of a random variable, or two outcomes of a random event.
- PMI --- Mutual information
  information --- entropy
- PMI can be negative!

Contents

WestlakeNLP

Using PMI to mine knowledge from texts

WestlakeNLP

- **The correlation between variables**.
  - Words and sentiment signals.
  - Neighboring words.
  - Features and class labels.

Learning sentiment lexicons

WestlakeNLP

- **Sentiment lexicon** contains information about the polarity and strength of sentiment words.

- *LEX(w)* represents the sentiment polarity, and the absolute value represents the strength.

$$SENTI(d) = \frac{\sum_i Lex(w_i)}{|\{w_i | Lex(w_i) \neq 0\}|}$$

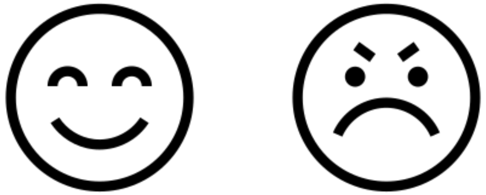
Learning sentiment lexicons

- The PMI between a word w and a *seed* word

$$PMI(w, seed) = log_2 \frac{P(w, seed)}{P(w)p(seed)}$$

$$LEX(w) = PMI(w, \mathrm{good}) - PMI(w, \mathrm{bad})$$

Emoticons in social media



Collocation extraction

- *Collocation* refers to words that are conventionally used together for certain meaning.

Mr President ✓

Mr Executive ✗

High temperature ✓

Big temperature ✗

- Given two words w_1 and w_2 and a corpus D, there association can be calculated using

$$PMI(w_1, w_2) = \log_2 \frac{P(w_1 w_2)}{P(w_1)P(w_2)}$$

Using PMI to select features

WestlakeNLP

- Feature selection: reduce the size of the feature vector

important features | unimportant features

goal , statement , president | a , in , does

- PMI between feature and class is a commonly used metric for feature selection, the higher PMI value, the more likely w is a strong indicator of c.

PMI and vector representations of words

- Representing a word in vector space
  - Useful for measuring semantic correlations
  - Thus far we have only learned a ``one-hot'' representation

PMI and vector representations of words

- *Distributional semantics* : the company of a word (*k-word windows*) tells us much information about its attributes.

Sentence	k	Context
s_1	2	{between, the, and, the }
	5	{the, water, halfway, between, the, and, the, island, ., <s>}
	7	{out, of, the, water, halfway, between, the, and, the, island, ., <s>, <s>, <s>}
s_2	2	{with, the, statement, are}
	5	{are, not, included, with, the, statement, are , called, outstanding, checks}
	7	{written but, are, not, included, with, the, statement, are , called, outstanding, checks, . , <s>}

- K-word windows for the word "bank" in
  s1 --- *There happened to be a rock sticking out of the water halfway between the bank and the island*
  s2 --- *The checks that have been written but are not included with the bank statement are called outstanding checks*

Word Representation

WestlakeNLP

Word	Representation	Feature vector
cat	One-hot	$\langle f_1 = 0, \ldots, f_{121} = 1, \ldots, f_{500} = 0, \ldots, f_{10000} = 0 \rangle$
	Context	$\langle f_1 = 1280, f_2 = 0, \ldots, f_{35} = 332, \ldots, f_{10000} = 0 \rangle$
dog	One-hot	$\langle f_1 = 0, \ldots, f_{121} = 0, \ldots, f_{500} = 1, \ldots, f_{10000} = 0 \rangle$
	Context	$\langle f_1 = 1190, f_2 = 19, \ldots, f_{35} = 271, \ldots, f_{10000} = 0 \rangle$

cat --- 121, dog --- 500, considering vector dot product.

PMI and vector representations of words

WestlakeNLP

- The vector representation of a word w_i is :

$$\overline{Vec(w_i)} = \left\langle PMI(w, w_1), PMI(w, w_2), \dots, PMI\left(w, w_{|V|}\right)\right\rangle$$

$$P(u,v) = \frac{\#(u \text{ and } v \text{ in each other's context window})}{\#(\text{any two words in each other's context window})}$$

$$= \frac{\#(u \text{ and } v \text{ in each other's context window})}{2k|D|^2}.$$

Word Representation

Word	Representation	Feature vector
cat	One-hot	$\langle f_1 = 0, \ldots, f_{121} = 1, \ldots, f_{500} = 0, \ldots, f_{10000} = 0 \rangle$
	Context	$\langle f_1 = 1280, f_2 = 0, \ldots, f_{35} = 332, \ldots, f_{10000} = 0 \rangle$
	PPMI	$\langle f_1 = 0.3, f_2 = 0, \ldots, f_{35} = 2.32, \ldots, f_{10000} = 0 \rangle$
dog	One-hot	$\langle f_1 = 0, \ldots, f_{121} = 0, \ldots, f_{500} = 1, \ldots, f_{10000} = 0 \rangle$
	Context	$\langle f_1 = 1190, f_2 = 19, \ldots, f_{35} = 271, \ldots, f_{10000} = 0 \rangle$
	PPMI	$\langle f_1 = 0.44, \ldots, f_{12} = 0.05, \ldots, f_{35} = 5.56, \ldots, f_{10000} = 0 \rangle$

- PMI and TF-IDF.

PMI and vector representations of words

WestlakeNLP

- $PPMI(u, v) = \max(PMI(u, v), 0)$

- We use positive PMI (PPMI) to reduce noise and the non-informative.

- Using PPMI, non differentiating words will have a small contribution to the distributional word representation.

Summary

WestlakeNLP

- Entropy and information.
- The maximum entropy principle for defining probabilistic models, and its application in deriving log-linear model forms
- Model perplexity, cross-entropy and KL-divergence for measuring the consistence between model distributions and data distributions
- Mutual information and pointwise mutual information (PMI) for natural language tasks
- Word representations and pointwise mutual information.