The Maximum Entropy

Special Topics in Spoken Language Processing

Guan-Yu, Menphis, Chen

Department of Computer Science & Information Engineering,
National Taiwan Normal University, Taipei, Taiwan
696470203@ntnu.edu.tw

Main Reference:

1. Ronald Rosenfeld, “A Maximum Entropy Approach to Adaptive Statistical Language Modeling,” Computer Science Department Carnegie Mellon University,1996
2. Adwait Ratnaparkhi, “A Simple Introduction to Maximum Entropy Models for Natural Language Processing,” University of Pennsylvania, 1997
3. Adam L. Berger, “A Maximum Entropy Approach to Natural Language Processing,” Columbia University, 1996
4. Paul Penfield, Jr., “Information and Entropy,” Massachusetts Institute of Technology, 2007

Outline

- Introduction to the Maximum Entropy
- Inference
- Principle of Maximum Entropy
- ME for Natural Language Processing
- ME for NLP in detail
- Conclusion
  - What is the main idea of Entropy?
  - Why do we want to “Maximum” the entropy?
  - What is the relation between ME and ML?
- Reference
- Appendix

Introduction to the Maximum Entropy - 1

- Suppose there are n events in the sample space $X = \{w_1, w_2, ..., w_n\}$,and the probability of w_i is p_i , where $\sum_i p_i = 1$.

- We define a function H which the domain is the sample space and the value is $H(p_1, ..., p_n)$. We hope that we can describe the “uncertainty” of events $w_1, w_2, ..., w_n$ by the H function.

- Here, the Entropy function H must satisfy three properties：

  $1. \forall n , H(p_1, ..., p_n)$ *be a continuoue function*

  $2. if\ p_i = \frac{1}{n}, \forall i = 1,2,...,n$

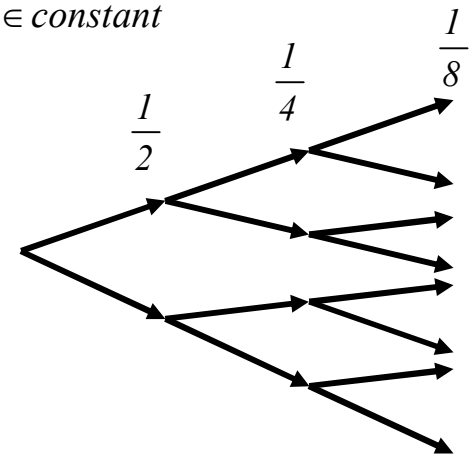
  $then\ H(\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n})$ *be a monotonically increasing function*

  3.if we divide a task into several small tasks

  then the origial H must be the weight sum of the several small tasks' H

Introduction to the Maximum Entropy - 2

- We can prove the entropy function $H(p_1,\ldots,p_n) = -k\sum_i p_i \log(p_i)$, $k \in constant$



proof： Step1. let $A(n) = H(\frac{1}{n},\ldots,\frac{1}{n})$, where $n \in N$

suppose $A(s^m) = mA(s)$, where $s,m \in N$

by example : if $s = 2$, $m = 3$

$$\text{then } A(2^3) = A(8) = H(\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8}) = 3A(2) = 3H(\frac{1}{2},\frac{1}{2},\frac{1}{2})$$

$$\text{by property3 we know : } H(\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8},\frac{1}{8})$$

$$= H(\frac{1}{2},\frac{1}{2}) + \frac{1}{2}\left[H(\frac{1}{2},\frac{1}{2}) + H(\frac{1}{2},\frac{1}{2})\right] + \frac{1}{4}\left[H(\frac{1}{2},\frac{1}{2}) + H(\frac{1}{2},\frac{1}{2}) + H(\frac{1}{2},\frac{1}{2}) + H(\frac{1}{2},\frac{1}{2})\right]$$

$$= H(\frac{1}{2},\frac{1}{2}) + H(\frac{1}{2},\frac{1}{2}) + H(\frac{1}{2},\frac{1}{2}) = 3H(\frac{1}{2},\frac{1}{2}) = 3A(2)$$

$$\text{so, by induction we have } H(\frac{1}{s^m},\ldots,\frac{1}{s^m}) = H(\frac{1}{s},\ldots,\frac{1}{s}) + \frac{1}{s}\left[sH(\frac{1}{s},\ldots,\frac{1}{s})\right] + \frac{1}{s^2}\left[s^2 H(\frac{1}{s},\ldots,\frac{1}{s})\right] + \ldots + \frac{1}{s^{m-1}}\left[s^{m-1} H(\frac{1}{s},\ldots,\frac{1}{s})\right]$$

$$= mH(\frac{1}{s},\ldots,\frac{1}{s}) = mA(s)$$

Introduction to the Maximum Entropy - 3

now, if we have t, s, n and m

and $s^m \le t^n \le s^{m+1}$, where $t, s, n, m \in N$

$$\Rightarrow m\log s \le n\log t \le (m+1)\log s \qquad \Rightarrow m \le n\frac{\log t}{\log s} \le m+1$$

$$\Rightarrow \frac{m}{n} \le \frac{\log t}{\log s} \le \frac{m+1}{n} \qquad \Rightarrow -\frac{1}{n} \le \frac{m}{n} - \frac{\log t}{\log s} \le 0 \quad \ldots(1)$$

by property2 we have : $A(s^m) \le A(t^n) \le A(s^{m+1})$

$$\Rightarrow mA(s) \le nA(t) \le (m+1)A(s) \qquad \Rightarrow \frac{m}{n} \le \frac{A(t)}{A(s)} \le \frac{m+1}{n} \quad \ldots(2)$$

combine(1) ,(2): $\frac{m}{n} - \frac{1}{n} \le \frac{m}{n} + \frac{A(t)}{A(s)} - \frac{\log t}{\log s} \le \frac{m+1}{n}$

$$\Rightarrow -\frac{1}{n} \le \frac{A(t)}{A(s)} - \frac{\log t}{\log s} \le \frac{1}{n} \qquad \Rightarrow \frac{A(t)}{A(s)} = \frac{\log t}{\log s} \quad (\text{by } n \in N)$$

$$\Rightarrow \frac{A(t)}{\log t} = \frac{A(s)}{\log s} = k \text{ , } k \in \text{constant}$$

$$\therefore A(n) = k\log(n) = -k\sum \frac{1}{n}\log(\frac{1}{n})$$

Introduction to the Maximum Entropy - 4

Step2. given another example :

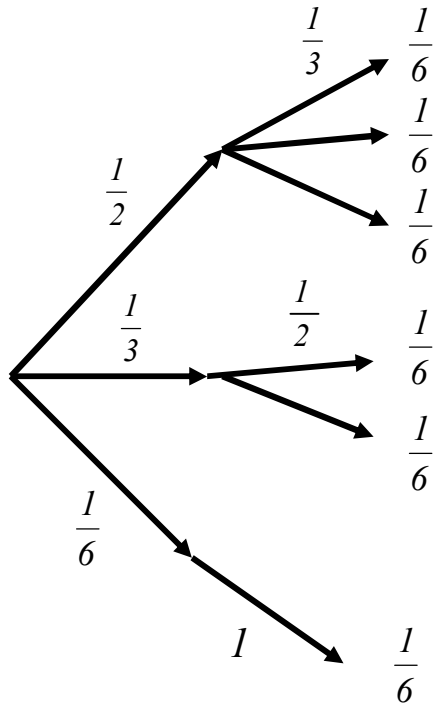
$$\because H(\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6}) = H(\frac{1}{2},\frac{1}{3},\frac{1}{6}) + \frac{1}{2}H(\frac{1}{3},\frac{1}{3},\frac{1}{3}) + \frac{1}{3}H(\frac{1}{2},\frac{1}{2}) + \frac{1}{6}H(1)$$

$$\therefore H(\frac{1}{2},\frac{1}{3},\frac{1}{6}) = H(\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6},\frac{1}{6}) - \frac{1}{2}H(\frac{1}{3},\frac{1}{3},\frac{1}{3}) - \frac{1}{3}H(\frac{1}{2},\frac{1}{2}) - \frac{1}{6}H(1)$$

$$\Rightarrow \text{if } p_1 = \frac{n_1}{n_1+n_2+n_3},\ p_2 = \frac{n_2}{n_1+n_2+n_3},\ p_3 = \frac{n_3}{n_1+n_2+n_3}$$

$$\text{then } H(p_1,p_2,p_3) = A(\sum n_i) - \sum p_i A(n_i)$$

$$\Rightarrow \text{if } p_i = \frac{n_i}{\sum n_i},\ \text{then } H(p_1,\ldots,p_i) = A(\sum n_i) - \sum p_i A(n_i)$$



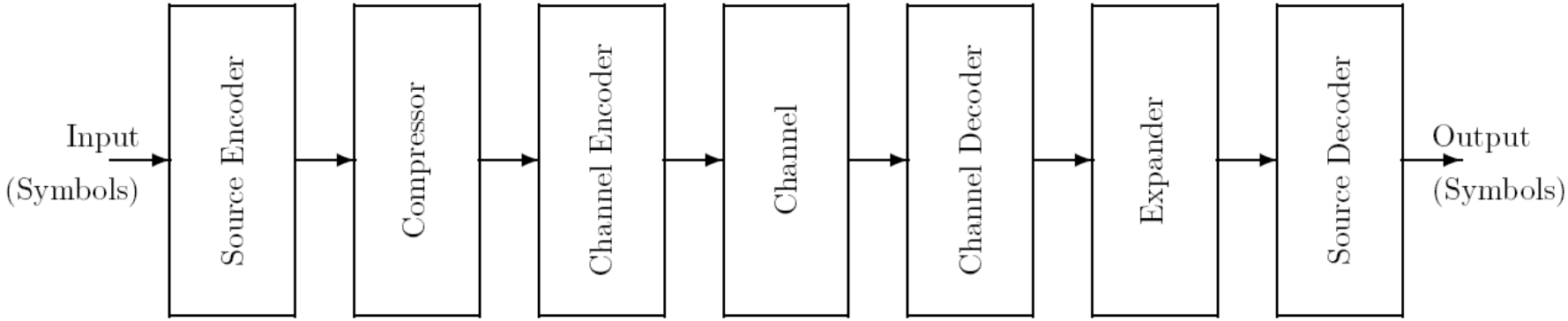
$$\text{by step1: } H(p_1,\ldots,p_i) = k\log(\sum n_i) - \sum p_i k\log(n_i)$$

$$= k\sum p_i \log(\sum n_i) - k\sum p_i \log(n_i)$$

$$= k\sum p_i \log\frac{n_i}{\sum n_i}$$

$$= -k\sum p_i \log(p_i)$$

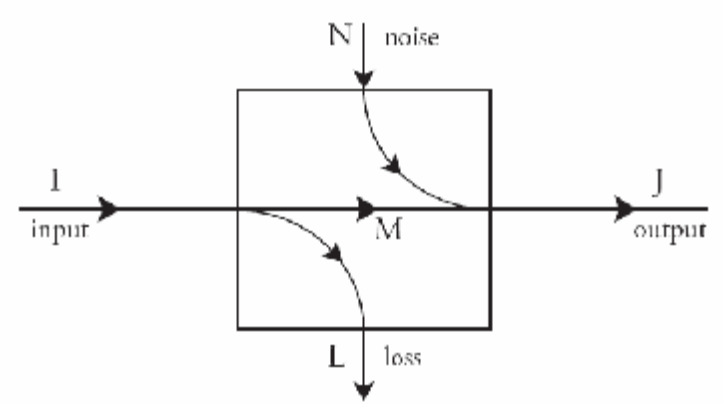
Inference - 1



- The model of a communication system that we have been developing is shown above, where the source is assumed to emit a stream of symbols or digits!
- Each boxes in this diagram can be represented by a “process” and they have some property:
  - Discrete
  - Finite
  - Memoryless
  - Nondeterministic
  - Lossy

Inference - 2

- In the case some internal state of the process would be set by the input, and the probability distribution leading to the output and the next state would depend on the current state.



- Now, we often necessary to determine the input event when only the output event has been observed.

- This is the case for communication systems, in which the objective is to eventually infer the symbol emitted by the source so that it can be created at the output.

- It is not always possible to infer the input event of a process from knowledge of the output. If the system has no loss, then inference is possible, but this is generally not the case.

Inference - 3

- The best that can generally be done is to refine the probabilities of the input events once the output event has been observed.

- If the input probability distribution $p(A_i)$ and the conditional output probabilities, which conditioned on the input events, $p(B_j \mid A_i) = c_{ji}$ are known.

- The unconditional probability $p(B_j)$ of each output event B_j is $p(B_j) = \sum_i c_{ji} p(A_i)$ and the joint probability of each input with each output $p(A_i, B_j)$ and the conditional probabilities $p(A_i \mid B_j)$ can be found using Bayes' Theorem as :

$$\begin{aligned} p(A_i, B_j) &= p(B_j) p(A_i \mid B_j) \\ &= p(A_i) p(B_j \mid A_i) \\ &= p(A_i) c_{ji} \end{aligned}$$

- So, for each input event A_i and for the particular output event B_j we have :

$$p(A_i \mid B_j) = \frac{p(A_i)}{p(B_j)} c_{ji}$$

Inference - 4

- Note that this approach only works if the input probability distribution is know. If the input probability distribution is not known, then another technique is required. One such technique is the Principle of Maximum Entropy.

- Here we can discuss about the uncertainty between before the output is known and after some particular output event is known.

$$U_{before} = \sum_i p(A_i) log(\frac{1}{p(A_i)})$$

$$U_{after}(B_j) = \sum_i p(A_i \mid B_j) log(\frac{1}{p(A_i \mid B_j)})$$

- Although it is not always true that $U_{after}(B_j) \le U_{before}$, we can prove that the average over all output states of the residual uncertainty is less than the original uncertainty：

$$\sum_j p(B_j) U_{after}(B_j) \le U_{before}$$

Inference - 5

$$We\ have\ U_{before} = \sum_i p(A_i)\log(\frac{1}{p(A_i)})\ and\ U_{after}(B_j) = \sum_i p(A_i \mid B_j)\log(\frac{1}{p(A_i \mid B_j)})$$

$$prove\ that: \sum_j p(B_j)U_{after}(B_j) \le U_{before}$$

$$show: \because \sum_j p(B_j)U_{after}(B_j) \le U_{before}$$

$$\sum_j p(B_j)U_{after}(B_j) - U_{before} = \sum_j p(B_j)\sum_i p(A_i \mid B_j)\log(\frac{1}{p(A_i \mid B_j)}) - \sum_i p(A_i)\log(\frac{1}{p(A_i)})$$

$$= \sum_j \sum_i p(B_j)\frac{p(A_i \cap B_j)}{p(B_j)}\log(\frac{p(B_j)}{p(A_i \cap B_j)}) - \sum_i p(A_i)\log(\frac{1}{p(A_i)})$$

$$= \sum_j \sum_i p(B_j)\frac{p(A_i)p(B_j)}{p(B_j)}\log(\frac{p(B_j)}{p(A_i)p(B_j)}) + \sum_i p(A_i)\log(p(A_i))$$

$$= \sum_j \sum_i p(A_i)p(B_j)\log(\frac{1}{p(A_i)}) + \sum_i p(A_i)\log(p(A_i))$$

$$= \sum_j \sum_i p(A_i)\log(p(A_i))\left[1 - p(B_j)\right] \ldots (\Phi)$$

$$Here,\ p(A_i)\log(p(A_i)) \le 0\ \ and\ \ 1 - p(B_j) \ge 0$$

$$\therefore \Phi \le 0$$

Inference - 6

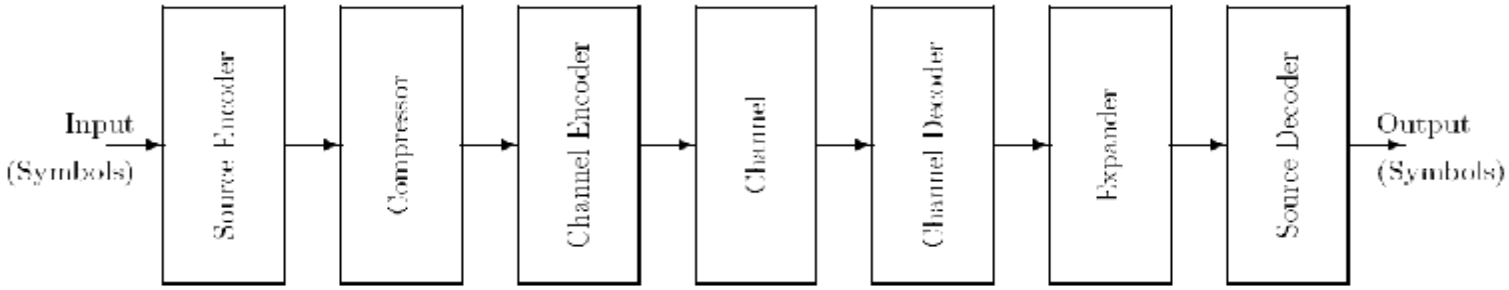
- In words, this statement says that on average, our uncertainty about the input state is never increased by learning something about the output state.
- In other words, on average, this technique of inference helps us get a better estimate of the input state.
- Often, it is not sufficient to calculate the probabilities of the various possible input events. The correct operation of a system may require that a definite choice be made of exactly one input event.
- For processes without loss, this can be done accurately. However, for processes with loss, some strategy must be used to convert probabilities to a single choice.
- One simple strategy, “Maximum likelihood,” is to decide on whichever input event has the highest probability after the output event is known. However, sometimes it does not work at all, especially in the case without any historical information.

Principle of Maximum Entropy - 1

- The Principle of Maximum Entropy is a technique that can be used to estimate input probabilities more generally.
- The result is a probability distribution that is consistent with known constraints expressed in terms of averages, or expected values, but is otherwise as unbiased as possible.
- This principle has applications in many domains, but was originally motivated by statistical physics, which attempts to relate macroscopic, measurable properties of physical systems to description at the atomic or molecular level.
- Particularly the definition of information in terms of probability distributions, provides a quantitative measure of ignorance that can be maximized mathematically to find the probability distribution that is maximally unbiased.

Principle of Maximum Entropy - 2

- We assume that each of the possible states A_i has some probability of occupancy $p(A_i)$ where i is an index running over the possible states.



- Our uncertainty is expressed quantitatively by the information which we do not have about the state occupied. This information is

$$S = \sum_i p(A_i) log(\frac{1}{p(A_i)})$$

- One person may have different knowledge of the system from another, and therefore would calculate a different numerical value for entropy. The Principle of Maximum Entropy is used to discover the probability distribution which leads to highest value for this uncertainty, thereby assuring that no information is inadvertently assumed.

Principle of Maximum Entropy - 3

- It is a property of the entropy formula above that it has its maximum value when all probabilities are equal.

- If we have no additional information about the system, then such a result seems reasonable. However, if we have additional information then we ought to be able to find a probability distribution that is better in the sense that it has less uncertainty.

- Here we consider one constraint, namely that we know the expected value of some quantity.

- For which each of the states has a value $g(A_i)$ then we want to consider those probability distributions for which the expected value is

$$G = \sum_i p(A_i)g(A_i)$$

Principle of Maximum Entropy - 4

Item	Entree	Cost	Calories	Probability of arriving hot	Probability of arriving cold
Value Meal 1	Burger	$1.00	1000	0.5	0.5
Value Meal 2	Chicken	$2.00	600	0.8	0.2
Value Meal 3	Fish	$3.00	400	0.9	0.1

- Example：There is a restaurant named Berger's Burgers. Suppose we are told that the average price of a meal is $2.5, and we want to estimate the separate probabilities of the various meals without making any other assumptions. Then our constraint would be

$$\$1.75 = \$1.0p(B) + \$2.0p(C) + \$3.0p(F)$$

$$1 = p(B) + p(C) + p(F)$$

- The amount of uncertainty about the probability distribution is

$$S = p(B)\log(\frac{1}{p(B)}) + p(C)\log(\frac{1}{p(C)}) + p(F)\log(\frac{1}{p(F)})$$

Principle of Maximum Entropy - 5

- Working with the two constraints, two of the unknown probabilities can be expressed in terms of the third. So we have

$$p(C) = 0.75 - 2p(F)$$
$$p(B) = 0.25 + p(F)$$

- The possible range of values of probabilities can be determined.

$$0 \le p(F) \le 0.375$$
$$0 \le p(C) \le 0.75$$
$$0.25 \le p(B) \le 0.625$$

- These expressions can be substituted into the formula for entropy so we have

$$S = (0.25 + p(F))\log\left(\frac{1}{0.25 + p(F)}\right) + (0.75 - 2p(F))\log\left(\frac{1}{0.75 - 2p(F)}\right) + p(F)\log\left(\frac{1}{p(F)}\right)$$

ME for Natural Language Processing - 1

- Many problems in natural language processing can be re-formulated as statistical classification problem, in which the task is to estimate the probability of “class” a occurring with “context” b, or p(a,b).

- Large text corpora usually contain some information about the cooccurrence of a’s and b’s, but never enough to completely specify p(a,b) pairs, since the word in b are typically spare.

- Consider the principle of maximum entropy which states that the correct distribution p(a,b) is that which maximizes entropy, or “uncertainty”, subject to the constraints, which represent “evidence”.

- [Jaynes, 1957]discusses its advantages: ”…in making inferences on the basis of partial information we must use that probability distribution which has maximum entropy subject to whatever is known. This is the only unbiased assignment we can make; to use any other would amount to arbitrary assumption of information which by hypothesis we do not have.”

ME for Natural Language Processing - 2

- More explicitly, if A denotes the set of possible classes, and B denotes the set of possible context, p should maximize the entropy

$$H(p) = -\sum_{x \in \varepsilon} p(x)\log(p(x))$$

where $x = (a,b), a \in A, b \in B$, and $\varepsilon = A \times B$, and should remain consistent with the evidence, or “partial information”.

- One way to represent evidence is to encode useful facts as features and to impose constraints on the values of those feature expectations.

ME for Natural Language Processing - 3

- A feature is a binary value function on events： $f_j : \varepsilon \to \{0,1\}$. Given k features, the constraints have the form $E_p f_j = E_{\tilde{p}} f_j$, $where\ 1 \le j \le k$.

- $E_p f_j$ is the model $p's$ expectation of f_j ： $E_p f_j = \sum_{x \in \varepsilon} p(x) f_j(x)$

  and is constrained to match the observed expectation, $E_{\tilde{p}} f_j$ ： $E_{\tilde{p}} f_j = \sum_{x \in \varepsilon} \tilde{p}(x) f_j(x)$

  where $\tilde{p}$ is the observed probability of x in some training sample S .

ME for Natural Language Processing - 4

- Definition 1：Relative Entropy, or Kullback-Liebler Distance
  - The relative entropy D between two probability distributions p and q is given by

$$D(p,q) = \sum_{x \in \varepsilon} p(x) \log \frac{p(x)}{q(x)}$$

- Definition 2：

$$A = set\ of\ possible\ classes \quad , \quad B = set\ of\ possible\ contexts$$

$$\varepsilon = A \times B \quad , \quad S = finite\ training\ sample\ of\ events$$

$$\tilde{p}(x) = observed\ probability\ of\ x\ in\ S$$

$$p(x) = the\ model\ p'\ s\ probability\ of\ x$$

$$f_j = A\ function\ of\ type\ \varepsilon \to \{0,1\}$$

$$E_p f_j = \sum_{x \in \varepsilon} p(x) f_j(x) \quad , \quad E_{\tilde{p}} f_j = \sum_{x \in \varepsilon} \tilde{p}(x) f_j(x)$$

$$P = \{p \mid E_p f_j = E_{\tilde{p}} f_j \ , j = \{1 \ldots k\}\} \quad , \quad Q = \{p \mid p(x) = \pi \prod_{j=1}^{k} \alpha_j^{f_j(x)}, 0 < \alpha_j < \infty\}$$

$$H(p) = \sum_{x \in \varepsilon} p(x) \log(p(x)) \quad , \quad L(p) = \sum_{x \in \varepsilon} \tilde{p}(x) \log(p(x))$$

ME for Natural Language Processing – 5

- Lemma 1
  - For any two probability distributions p and q, $D(p,q) \geq 0$, and $D(p,q) = 0$ if and only if $p = q$.
- Lemma 2 (Pythagorean Property)
  - Given P and Q from Definition 2, if $p \in P$, $q \in Q$, and $p^* \in P \cap Q$, then

$$D(p,q) = D(p,p^*) + D(p^*,q)$$

proof. *for any* $r, s \in P$, *and* $t \in Q$

$$\sum_{x \in \varepsilon} r(x)\log(t(x)) = \sum_x r(x)\left[\log\pi + \sum_j f_j(x)\log(\alpha_j)\right] = \log\pi\left[\sum_x r(x)\right] + \left[\sum_j \log(\alpha_j)\sum_x r(x)f_j(x)\right]$$

$$= \log\pi\left[\sum_x s(x)\right] + \left[\sum_j \log(\alpha_j)\sum_x s(x)f_j(x)\right] = \sum_x s(x)\left[\log\pi + \sum_j f_j(x)\log(\alpha_j)\right]$$

$$= \sum_x s(x)\log(t(x))$$

then let $p \in P, q \in Q,$ *and* $p^* \in P \cap Q$

$$D(p,p^*) + D(p^*,q) = \sum_x p(x)\log(p(x)) - \sum_x p(x)\log(p^*(x)) + \sum_x p^*(x)\log(p^*(x)) - \sum_x p^*(x)\log(q(x))$$

$$= \sum_x p(x)\log(p(x)) - \sum_x p(x)\log(p^*(x)) + \sum_x p(x)\log(p^*(x)) - \sum_x p(x)\log(q(x))$$

$$= \sum_x p(x)\log(p(x)) - \sum_x p(x)\log(q(x)) = D(p,q)$$

ME for Natural Language Processing – 6

- Theorem 1 (from lemmas 1 and 2) :
  - If $p^* \in P \cap Q$, then $p^* = arg\ max_{p \in P} H(p)$. Furthermore, p^* is unique.

proof. *Suppose* $p \in P$ *and* $p^* \in P \cap Q$.

Let $u \in Q$ *be the uniform distributi on*

so that $\forall x \in \varepsilon$ *we have* $u(x) = \frac{1}{|\varepsilon|}$

Step1. show that $H(p) \leq H(p^*)$:

By Lemma 2 ,

$$D(p, u) = D(p, p^*) + D(p^*, u)$$

and by lemma 1 ,

$$D(p, u) \geq D(p^*, u)$$

$$-H(p) - \log \frac{1}{|\varepsilon|} \geq -H(p^*) - \log \frac{1}{|\varepsilon|}$$

$$H(p) \leq H(p^*)$$

Step2. show p^* *is unique :*

$$H(p) = H(p^*) \Rightarrow D(p, u) = D(p^*, u) \Rightarrow D(p, p^*) = 0 \Rightarrow p = p^*$$

ME for Natural Language Processing – 7

- Example：
  - Suppose the task is to estimate a probability distribution $p(a,b)$, where $a \in \{x, y\}$ and $b \in \{0,1\}$. Furthermore suppose that the only fact known about p is that $p(x,0) + p(y,0) = 0.6$.

$p(a,b)$	0	1	
x	?	?	
y	?	?	
$total$	.6		1.0

  - Clearly there are many consistent ways to fill in the cells of the table; one way is shown as：

	0	1	
x	.5	.1	
y	.1	.3	
$total$	.6		1.0

  - However, the principle of Maximum Entropy recommends the assignment as：

	0	1	
x	.3	.2	
y	.3	.2	
$total$	.6		1.0

which is the most non-committal assignment of probabilities that meets the constraints on p.

ME for Natural Language Processing – 8

– Formally, under the maximum entropy framework, the fact
$p(x,0)+p(y,0)=0.6$
is implemented as a constraint on the model $p's$ expectation of a feature f : $E_p f = 0.6$

$$\text{where} \quad E_p f = \sum_{a\in\{x,y\}, b\in\{0,1\}} p(a,b) f(a,b)$$

and the f is defined as follows :

$$f(a,b) = \begin{cases} 1, & if\ b=0 \\ 0, & otherwise \end{cases}$$

– The observed expectation of f , or $E_{\tilde{p}} f$, is 0.6. The objective is then to maximize

$$H(p) = - \sum_{a\in\{x,y\}, b\in\{0,1\}} p(a,b) log(p(a,b))$$

ME for Natural Language Processing – 9

- The features typically express a cooccurrence relation between something in the linguistic context and a particular prediction.

- For example, estimates a model $p(a,b)$ where a is a possible part-of-speech tag and b contains the word to be tagged.

- A useful feature might be：

$$f(a,b) = \begin{cases} 1, & if\ a = DETERMINER\ and\ currentword(b) = "that" \\ 0, & otherwise \end{cases}$$

- The observed expectation $E_{\tilde{p}} f$ of this feature would then be the number of times we would expect to see the word “that” with the tag DETERMINER in the training sample, normalized over the number of training sample.

- The advantage of the maximum entropy framework is that experimenters need only focus their efforts on deciding ***what*** features to use, not on ***how*** to use them.

ME for NLP in detail - 1

- The quality of a language model M can be judged by its cross entropy with the distribution of some hitherto unseen text T

$$H(P_T; P_M) = -\sum_x P_T(x) \cdot \log P_M(x)$$

- The goal of statistical language modeling is to identify and exploit sources of information in the language stream, so as to bring the cross entropy down, as close as possible to the true entropy

ME for NLP in detail - 2

- Information Sources (in the Document's History)
  - Context-free estimation (unigram)
    - The most obvious information source for predicting the current word w_i is the prior distribution of words. Without this source, entropy is $log(V)$,where V is the vocabulary size.
    - The information provide by the priors is：

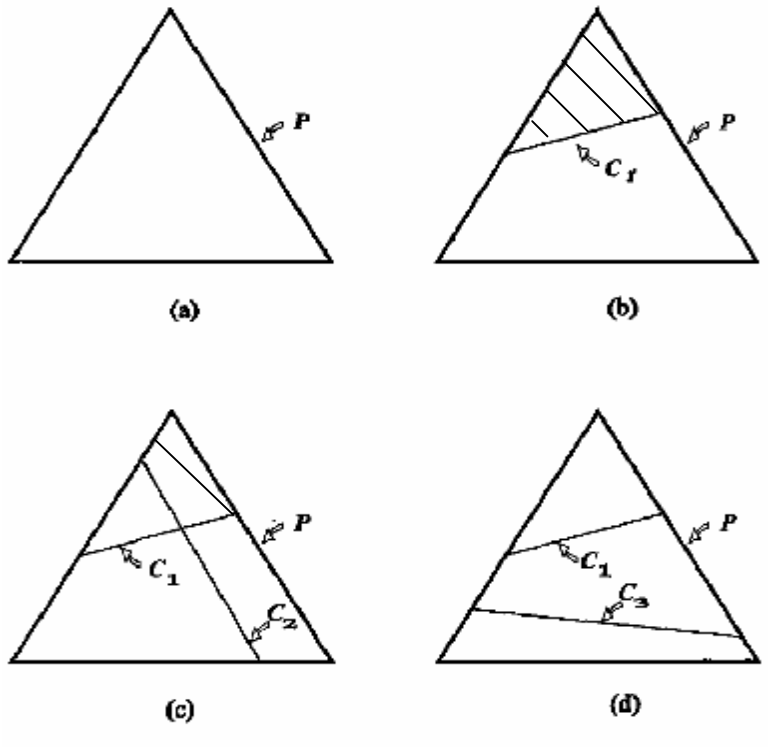
$$H(w_i) - H(w_i \mid \langle PRIORS \rangle) = \sum_{w \in V} log(V) + \sum_{w \in V} P(w) log(P(w)) = log(V) + \sum_{w \in V} P(w) log(P(w))$$

  - Short-term history (n-gram)
    - They are completely "blind" to any phenomenon, or constraint, this is outside their limited scope.
  - Short-term class history (class n-gram)
  - Intermediate distance (skip n-gram)
  - Long distance (trigger)
  - (Observed information)
  - Syntactic constraints

ME for NLP in detail - 3

- Under the Maximum Entropy approach, one does not construct separate models. Instead, one builds a single, combined model, which attempts to capture all the information form various knowledge source.

- Constrained optimization
  - a) all p are allowable
  - b) p lying on the line are allowable
  - c) p at intersection are allowable
  - d) no model can satisfy



ME for NLP in detail - 4

- Indicator function

$$f(h,w) \overset{def}{=} \begin{cases} 1 & if\ (h,w) \in S \\ 0 & otherwise \end{cases}$$

- Expected value of f with respect to empirical distribution and model

$$\tilde{p}(f) = \sum_{(h,w)} \tilde{P}(h,w) f(h,w)$$

$$p(f) = \sum_{(h,w)} P(h,w) f(h,w) \approx \sum_{(h,w)} \tilde{P}(h) P(w \mid h) f(h,w)$$

- Constraint:

$$P(f) = \tilde{p}(f)$$

ME for NLP in detail - 5

- Define the subset C :

$$C = \{p \in P \mid p(f_i) = \tilde{p}(f_i) \ \ \forall i = 1..n\}$$

- Among those models $p \in C$, the ME philosophy dictates that we select the most uniform distribution.

- A mathematical measure of uniformity of a conditional distribution $p(w \mid h)$ is provided by the conditional entropy

$$H(p) = \left(-\sum_{h,w} \tilde{P}(h) P(w \mid h) \log P(w \mid h) \right)$$

- A more common notation for the conditional entropy is $H(w \mid h)$, where w and h are random variables with joint distribution $\tilde{p}(h)p(w \mid h)$.

ME for NLP in detail - 6

- Constrained optimization (primal)

$$p^* = \arg\max_{p \in C} H(p)$$

$$= \arg\max_{p \in C} \left(-\sum_{h,w} \tilde{P}(h) P(w \mid h) \log P(w \mid h) \right)$$

- Lagrangian

$$\Lambda(P, \Lambda, \gamma) = -\sum_{h,w} \tilde{P}(h) P(w \mid h) \log P(w \mid h)$$

$$+ \sum_i \lambda_i \left(\sum_{h,w} \tilde{P}(h) P(w \mid h) f_i(h,w) - \sum_{h,w} \tilde{P}(h,w) f_i(h,w) \right)$$

$$- \gamma_h \left(\sum_w P(w \mid h) - 1 \right)$$

ME for NLP in detail - 7

$$\frac{\partial \Lambda}{\partial P(w|h)} = -\tilde{P}(h)(1+\log P(w|h)) + \sum_i \lambda_i \tilde{P}(h) f_i(h,w) - \gamma = 0$$

$$\Rightarrow \tilde{P}(h)(1+\log P(w|h)) = \sum_i \lambda_i \tilde{P}(h) f_i(h,w) - \gamma$$

$$\Rightarrow \log P(w|h) = \sum_i \lambda_i f_i(h,w) - \frac{\gamma}{\tilde{P}(h)} - 1$$

$$\Rightarrow P(w|h) = exp\left(\sum_i \lambda_i f_i(h,w)\right) exp\left(-\frac{\gamma}{\tilde{P}(h)} - 1\right)$$

$$\because \forall h, \sum_w P(w|h) = 1$$

$$\Rightarrow \sum_w P(w|h) = \sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right) \exp\left(-\frac{\gamma}{\tilde{P}(h)} - 1\right) = 1$$

$$\Rightarrow \exp\left(-\frac{\gamma}{\tilde{P}(h)} - 1\right) \cdot \sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right) = 1$$

$$\Rightarrow \exp\left(-\frac{\gamma}{\tilde{P}(h)} - 1\right) = \frac{1}{\sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right)}$$

ME for NLP in detail - 8

- So, we can know $P^*(w|h)$ will be an exponential form

$$P^*(w|h)=\frac{1}{\sum_{\hat{w}} exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)}\cdot exp\left(\sum_i \lambda_i f_i(h,w)\right)=\frac{1}{Z(h)}exp\left(\sum_i \lambda_i f_i(h,w)\right) \quad , \quad where\ Z(h)=\sum_{\hat{w}} exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)$$

$$\Rightarrow \Lambda(P,\Lambda,\gamma)=H(p)+\sum_i \lambda_i(p(f_i)-\tilde{p}(f_i))$$

$$=-\sum_{h,w}\tilde{P}(h)P(w|h)log\,P(w|h)+\sum_i \lambda_i\left(\sum_{h,w}\tilde{P}(h)P(w|h)f_i(h,w)-\tilde{p}(f_i)\right)$$

$$=-\sum_h \tilde{P}(h)\sum_w P(w|h)log\,P(w|h)+\sum_i \lambda_i\sum_h \tilde{P}(h)\sum_w P(w|h)f_i(h,w)-\sum_i \lambda_i\tilde{p}(f_i)$$

$$=-\sum_h \tilde{P}(h)\sum_w P(w|h)log\,P(w|h)+\sum_h \tilde{P}(h)\sum_w P(w|h)\sum_i \lambda_i f_i(h,w)-\sum_i \lambda_i\tilde{p}(f_i)$$

$$=-\sum_h \tilde{P}(h)\sum_w P(w|h)\left(\sum_i \lambda_i f_i(h,w)-log\,Z_\lambda(h)-\sum_i \lambda_i f_i(h,w)\right)-\sum_i \lambda_i\tilde{p}(f_i)$$

$$=-\sum_h \tilde{P}(h)\sum_w P(w|h)(-log\,Z_\lambda(h))-\sum_i \lambda_i\tilde{p}(f_i)$$

$$=\sum_h \tilde{P}(h)\sum_w P(w|h)(log\,Z_\lambda(h))-\sum_i \lambda_i\tilde{p}(f_i) \quad ...(\Psi)$$

ME for NLP in detail - 9

- IIS performs hill-climbing with enforcement of two relax lower bounds
  - Adam Berger, “The Improved Iterative Scaling Algorithm: A Gentle Introduction”, 1997
  - Rong Yan, “A variant of IIS algorithm”

- We want to update the parameter λ

$$\Rightarrow \sum_{h,w} \tilde{P}(h,w) log\ P_{\Lambda+\Delta}(w|h) - \sum_{h,w} \tilde{P}(h,w) log\ P_{\Lambda}(w|h)\ must \geq 0$$

ME for NLP in detail - 10

$$\sum_{h,w} \tilde{P}(h,w) log\, P_{\Lambda+\Delta}(w|h) - \sum_{h,w} \tilde{P}(h,w) log\, P_{\Lambda}(w|h)$$

$$= \sum_{h,w} \tilde{P}(h,w) log \left[\frac{1}{Z_{\Lambda+\Delta}(h)} exp\left(\sum_i (\lambda_i + \delta_i) f_i(h,w) \right) \right] - \sum_{h,w} \tilde{P}(h,w) log \left[\frac{1}{Z_{\Lambda}(h)} exp\left(\sum_i \lambda_i f_i(h,w) \right) \right]$$

$$= \sum_{h,w} \tilde{P}(h,w) \left[\sum_i (\lambda_i + \delta_i) f_i(h,w) + log\left(\frac{1}{Z_{\Lambda+\Delta}(h)} \right) \right] - \sum_{h,w} \tilde{P}(h,w) \left[\sum_i \lambda_i f_i(h,w) + log\left(\frac{1}{Z_{\Lambda}(h)} \right) \right]$$

$$= \sum_{h,w} \tilde{P}(h,w) \left[\sum_i \delta_i f_i(h,w) + log\left(\frac{1}{Z_{\Lambda+\Delta}(h)} \right) - log\left(\frac{1}{Z_{\Lambda}(h)} \right) \right]$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) - \sum_{h,w} \tilde{P}(h,w) log\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)} \right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) - \sum_{h} \tilde{P}(h) log\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)} \right)$$

ME for NLP in detail - 11

$$\therefore \sum_{h,w} \tilde{P}(h,w) log\ P_{\Lambda+\Delta}(w|h) - \sum_{h,w} \tilde{P}(h,w) log\ P_{\Lambda}(w|h) = \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) - \sum_{h} \tilde{P}(h) log\ \frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)}$$

$$\geq \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) + \sum_{h} \tilde{P}(h) \left(1 - \frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)}\right) \qquad (by\ -log(a) \geq 1 - a\ ,\ a > 0)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) + 1 - \sum_{h} \tilde{P}(h) \left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)}\right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) + 1 - \sum_{h} \tilde{P}(h) \frac{\sum_{w} exp\left(\sum_{i} (\lambda_i + \delta_i) f_i(h,w)\right)}{\sum_{\hat{w}} exp\left(\sum_{i} \lambda_i f_i(h,\hat{w})\right)}$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) + 1 - \sum_{h} \tilde{P}(h) \frac{\sum_{w} exp\left(\sum_{i} \lambda_i f_i(h,w)\right) \cdot exp\left(\sum_{i} \delta_i f_i(h,w)\right)}{\sum_{\hat{w}} exp\left(\sum_{i} \lambda_i f_i(h,\hat{w})\right)}$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) + 1 - \sum_{h} \tilde{P}(h) \sum_{w} \frac{exp\left(\sum_{i} \lambda_i f_i(h,w)\right)}{\sum_{\hat{w}} exp\left(\sum_{i} \lambda_i f_i(h,\hat{w})\right)} \cdot exp\left(\sum_{i} \delta_i f_i(h,w)\right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_{i} \delta_i f_i(h,w) + 1 - \sum_{h} \tilde{P}(h) \sum_{w} p_{\Lambda}(w|h) exp\left(\sum_{i} \delta_i f_i(h,w)\right) \quad \ldots lower\ bound\ (A)$$

ME for NLP in detail - 12

$$A(\Delta|\Lambda) = \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w p_\Lambda(w|h) exp\left(\sum_i \delta_i f_i(h,w) \right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w p_\Lambda(w|h) exp\left(f^{\#}(h,w) \sum_i \delta_i \frac{f_i(h,w)}{f^{\#}(h,w)} \right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w|h) exp\left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \left(\delta_i f^{\#}(h,w) \right) \right)$$

(by Jensen Inequality)

$$\geq \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w|h) \left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} exp\left(\delta_i f^{\#}(h,w) \right) \right) \quad \ldots \textit{lower bound (B)}$$

where $f^{\#}(h,w) = \sum_i f_i(h,w)$

Jensen Inequality : $\sum M(x) exp(N(x)) \geq exp(\sum M(x) N(x))$

ME for NLP in detail - 13

$$B(\Delta \mid \Lambda)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w \mid h) \left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \exp\left(\delta_i f^{\#}(h,w)\right) \right)$$

$$\frac{\partial B(\Delta \mid \Lambda)}{\partial \delta_i} = \sum_{h,w} \tilde{P}(h,w) f_i(h,w) - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w \mid h) f_i(h,w) \exp\left(\delta_i f^{\#}(h,w)\right)$$

- It is straightforward to solve for each of the n free parameters individually by differentiating with respect to δ in turn
- In case $f^{\#}(h,w)$ is constant for each (h,w) pair, IIS can be degraded to the GIS algorithm and simply solved in close-form
- Otherwise, this can solve with numeric root-finding procedure

ME for NLP in detail - 14

- Now, if we define a Dual function

$$\Psi(\lambda) = -\sum_{h} p(h)\log Z_{\lambda}(h) + \sum_{i}\lambda_i \sum_{h,w}\tilde{P}(h,w) f_i(h,w)$$

- Suppose that λ^* of $\Psi(\lambda)$ is the solution of the dual problem, then p_{λ^*} is the solution of the primal problem
  - The maximum entropy model subject to the constraints has parametric form p_{λ^*}, where the parameter values λ^* can be determined by maximizing the dual function $\Psi(\lambda)$

ME for NLP in detail - 15

- Definition of log-likelihood

$$L_{\tilde{P}}(P) \equiv \log \prod_{h,w} P(w \mid h)^{\tilde{P}(h,w)} = \sum_{h,w} \tilde{P}(h,w) \log P(w \mid h)$$

- Replace p with exponential form

$$L_{\tilde{P}}(P) = \sum_{h,w} \tilde{P}(h,w) \left(\sum_i \lambda_i f_i(h,w) \right) - \sum_{h,w} \tilde{P}(h,w) \log \sum_{\hat{w}} \exp\left(\sum_i \lambda_i f_i(h,\hat{w}) \right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \lambda_i f_i(h,w) - \sum_h \tilde{P}(h) \log \sum_w \exp\left(\sum_i \lambda_i f_i(h,w) \right)$$

  - Dual function equals to log-likelihood of the training data

- The model with maximum entropy is the model in the parametric family that maximizes the likelihood of the training data

Conclusion

- What is the main idea of Entropy?
- Why does we want to “Maximum” the entropy?
- What is the relation between ME and ML?

Reference

- Ronald Rosenfeld, “A Maximum Entropy Approach to Adaptive Statistical Language Modeling,” Computer Science Department Carnegie Mellon University,1996
- Adwait Ratnaparkhi, “A Simple Introduction to Maximum Entropy Models for Natural Language Processing,” University of Pennsylvania, 1997
- Adam L. Berger, “A Maximum Entropy Approach to Natural Language Processing,” Columbia University, 1996
- Paul Penfield, Jr., “Information and Entropy,” Massachusetts Institute of Technology, 2007
- Marian Grendár, jr, “Maximum Entropy: Clearing up Mysteries,” Institute of Measurement Science, Slovak Academy of Sciences, 2001
- Rong Yan, “A variant of IIS algorithm.”
- Richard O. Duda, “Pattern Classification,” second edition, 2001
- 李天岩, “熵 (Entropy),”數學傳播十三卷三期, Michigan State University, Distinguished Professor of Mathematics

Appendix - 1

- IIS performs hill-climbing in the log-likelihood space with enforcement of two relax lower bounds
  - Adam Berger, “The Improved Iterative Scaling Algorithm: A Gentle Introduction”, 1997
  - Rong Yan, “A variant of IIS algorithm”

- Definition of difference of likelihood function

$$L_{\tilde{P}}(\Lambda+\Delta)-L_{\tilde{P}}(\Lambda)=\sum_{h,w}\tilde{P}(h,w)\log P_{\Lambda+\Delta}(w\mid h)-\sum_{h,w}\tilde{P}(h,w)\log P_{\Lambda}(w\mid h)$$

Appendix - 2

- derivation

$$
\begin{aligned}
& L_{\tilde{P}}(\Lambda+\Delta)-L_{\tilde{P}}(\Lambda) \\
&= \sum_{h,w} \tilde{P}(h,w)\log P_{\Lambda+\Delta}(w\mid h)-\sum_{h,w} \tilde{P}(h,w)\log P_{\Lambda}(w\mid h) \\
&= \sum_{h,w} \tilde{P}(h,w)\log\left[\frac{1}{Z_{\Lambda+\Delta}(h)}\exp\left(\sum_i(\lambda_i+\delta_i)f_i(h,w)\right)\right]-\sum_{h,w} \tilde{P}(h,w)\log\left[\frac{1}{Z_{\Lambda}(h)}\exp\left(\sum_i\lambda_i f_i(h,w)\right)\right] \\
&= \sum_{h,w} \tilde{P}(h,w)\left[\sum_i(\lambda_i+\delta_i)f_i(h,w)+\log\left(\frac{1}{Z_{\Lambda+\Delta}(h)}\right)\right]-\sum_{h,w} \tilde{P}(h,w)\left[\sum_i\lambda_i f_i(h,w)+\log\left(\frac{1}{Z_{\Lambda}(h)}\right)\right] \\
&= \sum_{h,w} \tilde{P}(h,w)\left[\sum_i\delta_i f_i(h,w)+\log\left(\frac{1}{Z_{\Lambda+\Delta}(h)}\right)-\log\left(\frac{1}{Z_{\Lambda}(h)}\right)\right] \\
&= \sum_{h,w} \tilde{P}(h,w)\sum_i\delta_i f_i(h,w)-\sum_{h,w} \tilde{P}(h,w)\log\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)}\right) \\
&= \sum_{h,w} \tilde{P}(h,w)\sum_i\delta_i f_i(h,w)-\sum_{h} \tilde{P}(h)\log\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_{\Lambda}(h)}\right)
\end{aligned}
$$

Appendix - 3

$$L_{\tilde{P}}(\Lambda+\Delta)-L_{\tilde{P}}(\Lambda)=\sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)-\sum_h \tilde{P}(h)\log\frac{Z_{\Lambda+\Delta}(h)}{Z_\Lambda(h)}$$

$$\geq \sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)+\sum_h \tilde{P}(h)\left(1-\frac{Z_{\Lambda+\Delta}(h)}{Z_\Lambda(h)}\right) \qquad (by\ -\log(a)\geq 1-a\ ,\ a>0)$$

$$=\sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)+1-\sum_h \tilde{P}(h)\left(\frac{Z_{\Lambda+\Delta}(h)}{Z_\Lambda(h)}\right)$$

$$=\sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)+1-\sum_h \tilde{P}(h)\frac{\sum_w \exp\left(\sum_i(\lambda_i+\delta_i)f_i(h,w)\right)}{\sum_{\hat{w}}\exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)}$$

$$=\sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)+1-\sum_h \tilde{P}(h)\frac{\sum_w \exp\left(\sum_i \lambda_i f_i(h,w)\right)\cdot\exp\left(\sum_i \delta_i f_i(h,w)\right)}{\sum_{\hat{w}}\exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)}$$

$$=\sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)+1-\sum_h \tilde{P}(h)\sum_w \frac{\exp\left(\sum_i \lambda_i f_i(h,w)\right)}{\sum_{\hat{w}}\exp\left(\sum_i \lambda_i f_i(h,\hat{w})\right)}\cdot\exp\left(\sum_i \delta_i f_i(h,w)\right)$$

$$=\sum_{h,w}\tilde{P}(h,w)\sum_i \delta_i f_i(h,w)+1-\sum_h \tilde{P}(h)\sum_w p_\Lambda(w\mid h)\exp\left(\sum_i \delta_i f_i(h,w)\right) \quad \ldots lower\ bound\ (A)$$

Appendix - 4

$$A(\Delta \mid \Lambda) = \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w p_\Lambda(w \mid h) exp\left(\sum_i \delta_i f_i(h,w) \right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w p_\Lambda(w \mid h) exp\left(f^{\#}(h,w) \sum_i \delta_i \frac{f_i(h,w)}{f^{\#}(h,w)} \right)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w \mid h) exp\left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \left(\delta_i f^{\#}(h,w) \right) \right)$$

(by Jensen Inequality)

$$\geq \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w \mid h) \left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} exp\left(\delta_i f^{\#}(h,w) \right) \right) \quad \ldots \textit{lower bound (B)}$$

where $f^{\#}(h,w) = \sum_i f_i(h,w)$

Jensen Inequality : $\sum M(x) exp(N(x)) \geq exp(\sum M(x) N(x))$

Appendix - 5

$$B(\Delta \mid \Lambda)$$

$$= \sum_{h,w} \tilde{P}(h,w) \sum_i \delta_i f_i(h,w) + 1 - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w \mid h) \left(\sum_i \frac{f_i(h,w)}{f^{\#}(h,w)} \exp\left(\delta_i f^{\#}(h,w)\right) \right)$$

$$\frac{\partial B(\Delta \mid \Lambda)}{\partial \delta_i} = \sum_{h,w} \tilde{P}(h,w) f_i(h,w) - \sum_h \tilde{P}(h) \sum_w P_\Lambda(w \mid h) f_i(h,w) \exp\left(\delta_i f^{\#}(h,w)\right)$$

- It is straightforward to solve for each of the n free parameters individually by differentiating with respect to δ in turn
- In case $f^{\#}(h,w)$ is constant for each (h,w) pair, IIS can be degraded to the GIS algorithm and simply solved in close-form
- Otherwise, this can solve with numeric root-finding procedure