Maximum Entropy Models

Zongheng Yang

Last Modified: Jun 9, 2012

1 Overview

Maximum entropy models in NLP seek to maximize the entropy of observed data (training data) under the constraint that the model distribution of every feature should equal the corresponding empirical distribution.

Entropy H is defined as the expected surprise:

$$H(p) = E_p log_2(\frac{1}{p_x}) = -\sum_x p_x log_2 p_x \tag{1}$$

where p_x denotes the probability of event x. (See Slide 12-16 of Stanford NLP's AdvancedMaxent.pdf for demonstration of calculating Maxent probabilities.)

The theorectical foundation of this collection of models is the principle of maximum entropy, which states "the probability distribution which best represents the current state of knowledge is the one with largest information theoretical entropy."

Maxent models are discriminative models, different from generative models such as Naive Bayes model. In particular, unlike Naive Bayes, Maxent models do not assume feature independence.

2 Generative Models vs. Discriminative Models

Generative models place probabilities over observed data + "hidden stuff" (generate the observed data from hidden stuff); or **joint** models, $P(c, d)$.

Discriminative models take the data as given, and put a probability over hidden structure given the data (use the observed data to calculate and put probabilities on hidden stuff, such as classes in classification); or **conditional** models, $P(c|d)$. Includes logistic regression, conditional loglinear, **maxent models**.

3 Features

A feature is a function: $f : C \times D \to \mathbb{R}$ (C: classes, D: data (documents)). Usually we limit the range of features to $\{0, 1\}$. The model assigns **a weight** λ_i

to each feature f_i. Positive weights: feature is likely correct, negative weights: likely incorrect.

4 Conditional Likelihood of Data

In solving classification problems with exponential models (log-linear, maxent, logistic, Gibbs), the **conditional likelihood** (or "vote") that a class c will get at a particular data item d (not necessarily a word) is defined as

$$P(c|d,\lambda) = \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}. \tag{2}$$

The denominator normalizes the votes, and therefore the votes can be deemed as probabilities. Suming up all the data points and using logarithms, we have the goal, the conditional log likelihood, for maximization:

$$\begin{aligned} \log P(C|D,\lambda) &= \sum_{(c,d)\in(C,D)} \log P(c|d,\lambda) & (3) \\ &= \sum_{(c,d)\in(C,D)} \log \frac{\exp \sum_i \lambda_i f_i(c,d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c',d)}. & (4) \end{aligned}$$

5 Maximizing the Likelihood

Taking the partial derivative $\log P(C|D,\lambda)$ with respect to λ_i and simplifying gives

$$\frac{\partial \log P(C|D,\lambda)}{\partial \lambda_i} = \text{empiricalCnt}(f_i, C) - \text{predictedCnt}(f_i, \lambda).$$

6 Smoothing: L2 Prior

The **gaussian prior** (or L_2 **prior**) is defined to be

$$P(\lambda_i) = \frac{1}{\sigma_i\sqrt{2\pi}} \exp(-\frac{(\lambda_i-\mu_i)^2}{2\sigma_i^2})$$

where μ denotes the mean and σ^2 denotes the variance. Ususally we take $\mu = 0$ and $2\sigma^2 = 1$ (works very well).

The maximization objective thus changes to

$$\begin{aligned} \log P(C,\lambda|D) &= \log P(C|D,\lambda) - \log P(\lambda) \\ &= \sum_{(c,d)\in(C,D)} \log P(c|d,\lambda) - \sum_i \frac{(\lambda_i-\mu_i)^2}{2\sigma_i^2} + k. \end{aligned}$$

Taking the partial derivative:

$$\frac{\partial \log P(C,\lambda|D)}{\partial \lambda_i} = \text{empiricalCnt}(f_i, C) - \text{predictedCnt}(f_i, \lambda) - \frac{\lambda_i - \mu_i}{\sigma^2}.$$