TTIC 31230 Fundamentals of Deep Learning

Problems for Graphical Models.

Problem 1. Dynamic Programing for HMMs Assume we have an input sequence $x_1, \ldots, x_T$ and a phoneme gold label $y_1, \ldots, y_T$ with $y_t \in \mathcal{P}$. This problem is simpler than CTC because the gold label has the same length as the input sequence.

In an HMM we assume a hidden state sequence $s_1, \ldots, s_T$ with $s_t \in \mathcal{S}$ where $\mathcal{S}$ is some finite sets of "hidden states". Here will assume that then some deep network has computed transition probabilities and emission probabilities.

$$P_{\mathrm{Trans}}(s_{t+1} \mid s_t)$$

$$P_{\mathrm{Emit}}(y_t \mid s_t)$$

We assume an initial state s_{init} and a stop state s_{stop} such that $s_1 = s_{\mathrm{init}}$ (before emitting any phonemes). The length T is determined by when the hidden state becomes s_{stop} giving $s_{T+1} = s_{\mathrm{stop}}$.

For a given gold sequence $y_1, \ldots, y_T$ we define a "forward tensor" as

$$F[t,s] = P(y_1, \ldots, y_{t-1} \ \wedge s_t = s)$$

We have

$$\begin{array}{rcl} F[1, s_{\mathrm{init}}] & = & 1 \\ F[1, s] & = & 0 \quad \text{for } s \neq s_{\mathrm{init}} \end{array}$$

(a) Write a dynamic programming equation to compute $F[t,s]$ from $F[t-1,s']$ for various values of s'.

(b) Express $P(y_1, \ldots, y_T)$ in terms of $F[t,s]$.

(c) EM for HMMs involves computing a "backward" tensor

$$B[t,s] = P(y_t, \ldots, y_T \mid s_t = s).$$

Explain why, if the forward equations are written in a framework, we do not need to also compute the backward tensor.

Problem 2. CTC for image labeling

Suppose that the training data consists of pairs (I, S) where I is an image and S is a set of object types occuring in the image. For example S might be $\{\mathrm{Person}, \mathrm{Dog}, \mathrm{Car}\}$. To be concrete we can take $\mathcal{C}$ to be the set of image labels used in CIFAR 100 and take S to be a subset of $\mathcal{C}$ containing no more than five labels ($|S| \leq 5$). We want to do SGD on a model defining $P_\Phi(S \mid I)$.

We will use a latent variable $z[X,Y]$ such that for pixel coordinates (x,y) we have $z[x,y] \in \mathcal{C} \cup \{\perp\}$. For a given $z[X,Y]$ define $S(z[X,Y])$ to be the set of classes appearing in $z[X,Y]$, i.e., $S(z[X,Y]) = \{c \; \exists \; x,y \; z(x,y) = c\}$. Here the "semantic segmentation" $Z[X,Y]$ is analogous to the phoneme sequence $z[T]$ in CTC. Unlike the CTC model, the label S is a set rather than a sequence.

We assume a CNN (with convolutions of stride 1 to preserve spatial dimensions) followed by a softmax at each pixel to get a probability $P_\Phi(z[x,y] = c)$ for each pixel location (x,y) and each $c \in \mathcal{C} \cup \{\perp\}$ and where each pixel location has an independent probability distribution over classes. To simplify notation we can reshape the pixel locations into a linear sequence and replace $z[X,Y]$ by $z[T]$ with $T = X \times Y$ so we have $z[1], z[1], \ldots, z[T]$.

Define

$$S_t = \{c \in \mathcal{C} \; \exists t' \leq t \; z[t'] = c\}$$

For $U \subseteq S$ define

$$F[U,t] = P(S_t = U)$$

Note that for $|S| \leq 5$ there are at most 32 possible values of U. Give dynamic programming equations defining $F[U,0]$ and defining $F[U,t+1]$ in term of $F[U',t]$ for various U'.

Problem 3. Pseudolikelihood of a one dimensional spin glass. We let $\hat{x}$ be an assignment of a value to every node where the nodes are numbered from 1 to N_{nodes} and for every node i we have $\hat{x}[i] \in \{0,1\}$. We define the score of $\hat{x}$ by

$$f(\hat{x}) = \sum_{i=1}^{N-1} \mathbf{1}[\hat{x}[i] = \hat{x}[i+1]]$$

The probability distribution over assignments is defined by a softmax. We let $\hat{x}[i := v]$ be the assignment identical to $\hat{x}$ except that node i is assigned the value v. The expression $\hat{x}[i] = v$ is either true or false depending on whether no i is assigned value v in $\hat{x}$. So these are quite different.

$$P_f(\hat{x}) = \operatorname*{softmax}_{\hat{x}} f(\hat{x})$$

Pseudolikelihood is defined in terms of the softmax probability P_f as follows.

$$\tilde{P}_f(\hat{x}) \;\; = \;\; \prod_i P_f(\hat{x}[i] \mid \hat{x} \backslash i)$$

What is the pseudolikelihood of the all ones assignment under the definition of f given above?

Problem 4. Pseudolikelihood for images. Consider a semantic segmentation $\hat{y}[i]$ on pixels i with $\hat{y}[i]$ a semantic class label in $\{C_1, \ldots, C_K\}$. We also assume a scoring function s_Φ on semantic segmentations defining

$$P_\Phi(\hat{y}) = \underset{\hat{y}}{\mathrm{softmax}}\ s_\Phi(\hat{y})$$

Pseudolikelihood is defined by

$$\tilde{P}_\Phi(\hat{y}) \quad = \quad \prod_i P_\Phi(\hat{y}[i] \mid \hat{y}\backslash i)$$

where $\hat{y}\backslash i$ assigns a class to every pixel other than i, and $\hat{y}[i := c]$ is the semantic segmentation identical to $\hat{y}$ except that pixel i is labeled with semantic class c. In a typical graphical model for images we have

$$P_\Phi(\hat{y}[i] \mid \hat{y}\backslash i) = P_\Phi(\hat{y}[i] \mid \hat{y}[N(i)])$$

where $\hat{y}[N(i)]$ is $\hat{y}$ restricted to those pixels which are neighbors of pixel i.

(a) show

$$\frac{P_\Phi(\hat{y})}{\sum_c P_\Phi(\hat{y}[i := c])} = \underset{c}{\mathrm{softmax}}\, s_\Phi(\hat{y}[i := c]) \quad \text{evaluated at } c = y[i]$$

(b) How many scores need to be computed in the worst case for computing $P_\Phi(\hat{y})$. Given the result of part (a), how many for computing $\tilde{P}_\Phi(\hat{y})$?

(c) Consider a distribution on semantic segmentations where for each pixel the class assigned to that pixel is uniquely determined by the classes of its neighbors. Can this distribution be defined by a softmax over scores? Explain your answer.

(d) If P_Φ is a distribution defined in some other way such that the class of each pixel is completely determined by the other pixels, given a simple expression for $\tilde{P}_\Phi(\hat{y})$ in the case where $\hat{y}$ has nonzero probability under P_Φ.

Problem 5. Pseudolikelihood for Monocular Distance Estimation. (25 points) Here we are interested in labeling each pixed with a distance from the camera. Each pixel i is to be labeled with a real number $\hat{y}[i] > 0$ giving the distance in (say) meters from the camera to the point on the object displayed by that pixel. We assume a neural network that computes for each pixel i an expected distance μ_i and a variance $\sigma_i > 0$. For each pair of neighboring pixels i and j the neural network computes a real number $\lambda_{\langle i,j \rangle} \geq 0$. For each assignment $\hat{y}$ of distances to pixels we then define the score $s(\hat{y})$ by

$$s(\hat{y}) \quad = \quad \sum_{i \in \text{nodes}} -(\hat{y}[i] - \mu_i)^2/\sigma_i^2 \quad + \sum_{\langle i,j \rangle \in \text{edges}} -\lambda_{\langle i,j \rangle} |\hat{y}[i] - \hat{y}[j]|$$

(a) This scoring function determines a continuous softmax distribution defined by

$$p(\hat{y}) = \frac{1}{Z} e^{s(\hat{y})}$$

where Z is an integral rather than a sum. What is the dimension of the space to be integrated over in computing Z?

(b) We now consider pseudolikelihood for this problem. Give an expression for the continuous conditional probability density on $\hat{y}[i]$ for the distance $\hat{y}[i]$ conditioned on the value of the neighbors $N(i)$ of node i. This probability is written $p(\hat{y}[i] \mid \hat{y}[N(i)])$. You answer should be given as a function of the values $\hat{y}[j]$ for the nodes j neighboring i written $j \in N(i)$. Write Z as an integral but do not bother trying to solve it. What is the dimention of the integral for this conditional probability?

Problem 6. Computing the Partition Function for a Chain Graph. Consider a graphical model defined on a sequence of nodes $n_1, \ldots, n_T$. We are interested in "colorings" $\hat{\mathcal{Y}}$ which assign a color $\hat{\mathcal{Y}}[n]$ to each node n. We will use y to range over the possible colors. Suppose that we assign a score $s(\hat{\mathcal{Y}})$ to each coloring defined by

$$s(\hat{\mathcal{Y}}) = \left(\sum_{t=1}^{T} S^N[t, \hat{\mathcal{Y}}[n_t]]\right) + \left(\sum_{t=1}^{T-1} S^E[t, \hat{\mathcal{Y}}[n_t], \hat{\mathcal{Y}}[n_{t+1}]]\right)$$

In this problem we derive an efficient way to exactly compute the partition function

$$Z = \sum_{\hat{\mathcal{Y}}} e^{s(\hat{\mathcal{Y}})}.$$

Let $\hat{\mathcal{Y}}_t$ range over colorings of $n_1, \ldots n_t$ and define the score of $\hat{\mathcal{Y}}_t$ by

$$s(\hat{\mathcal{Y}}_t) = \left(\sum_{s=1}^{t} S^N[s, \hat{\mathcal{Y}}[n_s]]\right) + \left(\sum_{s=1}^{t-1} S^E[s, \hat{\mathcal{Y}}_t[n_s], \hat{\mathcal{Y}}_t[n_{s+1}]]\right)$$

Now define $Z_t(y)$ by

$$\begin{aligned} Z_1(y) &= e^{S^N[1,y]} \\ Z_{t+1}(y) &= \sum_{\hat{\mathcal{Y}}_t} e^{s(\hat{Y}_t)} e^{S^E[t, \hat{\mathcal{Y}}_t[n_t], y]} e^{S^N[t+1,y]} \end{aligned}$$

(a) Give dynamic programming equations for computing $Z_t(y)$ efficiently. You do not have to prove that your equations are correct — just writing the correct equations gets full credit.

(b) show that $Z = \sum_y Z_T(y)$

Problem 7. Consider a probability distribution on structured labels $\mathcal{Y}[N]$ where $\mathcal{Y}[n]$ is either -1, 0 or 1. Consider a score function $s(\mathcal{Y})$ defined by

$$s(\mathcal{Y}) = \left(\sum_{n=0}^{N-2} \mathcal{Y}[n]\ \mathcal{Y}[n+1]\right) + \mathcal{Y}[N-1]\mathcal{Y}[0]$$

We can think of this as a ring of edge potentials with no node potentials. We are interested in the probability defined by the exponential softmax

$$P_s(\mathcal{Y}) = \frac{1}{Z_s} e^{s(\mathcal{Y})}$$
$$Z_s = \sum_{\mathcal{Y}} e^{s(\mathcal{Y})}$$

(a) Given an expression for the negative log pseud-likelihood $-\ln \tilde{P}_s(\mathcal{Y})$ where $\mathcal{Y}$ is the constant assignment defined by $\mathcal{Y}[n] = 0$ for all n. Your expression should be a simple function of N.

(b) Repeat part (a) but for the constant structured label defined by $\mathcal{Y}[n] = 1$.