Homework 4

Phillip Wang pw1287

November 30, 2021

1.1

a

Using the latent variable energy-based model to map one-to-many relationships. We can change the latent variables in order to get different predictions for the same input.

b

Probabilistic models are special cases of EBM. However, we have more choices of scoring function and objective function in EBMs.

c

Use the Gibbs distribution. $P(y|x) = \frac{\exp(-\beta F_W(x,y))}{\int_{y'} \exp(-\beta F_W(x,y'))}$

d

Energy function is minimized during inference, i.e. to produce low energy for the right (x,y) pairs. Loss function is minimized during training and is used to measure the quality of the available energy functions.

e

Yes. If the energy function is easy to learn and it is the only objective we are trying to optimize, we can set loss function equal to the energy function.

f

If we have similar input with different labels, only pushing down the energy for the correct inputs will not help the model differentiate the similar yet incorrect inputs from the ones that are correct.

g

- Push down of the energy of data points, push up everywhere else: we need to use variational approximation or tractable partition in order to compute the probabilities of all other locations. E.g. Max Likelihood
- Push down on a group of data points, push up on a chosen group of points: we can specifically choose the incorrect pairs that are similar to push up, hence we do not need to compute exaustively in the space. E.g. Siamese Nets
- Train a function that maps incorrect pairs to correct pairs: we can use a denoising autoencoder to map incorrect pairs to the correct pairs. E.g. BERT

h

Log loss is an example where both positive and negative examples are used.

$$\mathcal{L}(x, y, W) = \log(1 + \exp(F_W(x^i, y^i) - F_W(x^i, \bar{y}^i)))$$

The loss function aims to keep $F_W(Y^i, X^i)$ low for correct examples, and keep $F_W(\bar{Y}^i, X^i)$ high for incorrect examples.

1.2

a

Since we have a discrete y, we have

$$P_W(y|x) = \frac{e^{-\beta F_W(x,y)}}{\sum_{i=1}^{n} e^{-\beta F_W(x,y_i)}}$$

b

$$\begin{aligned}
\mathcal{L}_{nll}(x, y, W) &= -\frac{1}{\beta}\log(P_W(y|x)) = -\frac{1}{\beta}\log(\frac{e^{-\beta F_W(x,y)}}{\sum_{i=1}^{n} e^{-\beta F_W(x,y_i)}}) \\
&= -\frac{1}{\beta}(\log(e^{-\beta F_W(x,y)}) - \log(\sum_{i=1}^{n} e^{-\beta F_W(x,y_i)})) \\
&= -\frac{1}{\beta}(-\beta F_W(x, y) - \log(\sum_{i=1}^{n} e^{-\beta F_W(x,y_i)})) \\
&= F_W(x, y) + \frac{1}{\beta}\log(\sum_{i=1}^{n} e^{-\beta F_W(x,y_i)})
\end{aligned}$$

c

$$\begin{aligned}
\frac{\partial \mathcal{L}_{nll}(x, y, W)}{\partial W} &= \frac{\partial F_W(x, y)}{\partial W} + \frac{1}{\beta}\frac{\partial \log(\sum_{i=1}^{n} \exp(-\beta F_W(x, y_i))}{\partial W} \\
&= \frac{\partial F_W(x, y)}{\partial W} + \frac{1}{\beta}\frac{\frac{\partial}{\partial W}\sum_{i=1}^{n} \exp(-\beta F_W(x, y))}{\sum_{i=1}^{n} \exp(-\beta F_W(x, y_i)} \\
&= \frac{\partial F_W(x, y)}{\partial W} + \frac{1}{\beta}\frac{\sum_{i=1}^{n} -\beta \exp(-\beta F_W(x, y_i))\frac{\partial F_W(x,y_i)}{\partial W}}{\sum_{i=1}^{n} \exp(-\beta F_W(x, y_i))} \\
&= \frac{\partial F_W(x, y)}{\partial W} - \sum_{i=1}^{n} P_W(y_i|x)\frac{\partial F_W(x, y_i)}{\partial W}
\end{aligned}$$

It is intractable to compute the sum over all possible labels (or integral in continuous case) in $y \in \mathcal{Y}$ because the output is high-dimensional or has compositional structure or that the space is uncountable. In such case, we could use Monte-Carlo sampling, architectural design, variational methods etc.

d

The contrastive term in NLL causes energies for all the pairs to be pushed up. However, for the pairs with correct labels, they are pushed down even harder by the first term of the expression derived in c). Moreover, as the incorrect pairs is being pushed up proportional to the likelihood of the given pair, the adjacent incorrect pairs are pushed even higher.

1.3

a

$$\frac{\partial l_{simple}(x,y,\bar{y},W)}{\partial W} = \begin{cases} \frac{\partial F_W(x,y)}{\partial W} - \frac{\partial F_W(x,\bar{y})}{\partial W} & \text{when } F_W(x,y) > 0, F_W(x,\bar{y}) < m \\ \frac{\partial F_W(x,y)}{\partial W} & \text{when } F_W(x,y) > 0, F_W(x,\bar{y}) \geq m \\ -\frac{\partial F_W(x,\bar{y})}{\partial W} & \text{when } F_W(x,y) \leq 0, F_W(x,\bar{y}) < m \\ 0 & \text{when } F_W(x,y) \leq 0, F_W(x,\bar{y}) \geq m \end{cases}$$

b

$$\frac{\partial l_{hinge}(x,y,\bar{y},W)}{\partial W} = \begin{cases} \frac{\partial F_W(x,y)}{\partial W} - \frac{\partial F_W(x,\bar{y})}{\partial W} & \text{when } F_W(x,y) - F_W(x,\bar{y}) + m > 0 \\ 0 & \text{otherwise} \end{cases}$$

c

$$\frac{\partial l_{square-square}(x,y,\bar{y},W)}{\partial W} = \begin{cases} 2F_W(x,y)\frac{\partial F_W(x,y)}{\partial W} - 2(m - F_W(x,\bar{y}))\frac{\partial F_W(x,\bar{y})}{\partial W} & \text{when } F_W(x,y) > 0, F_W(x,\bar{y}) < m \\ 2F_W(x,y)\frac{\partial F_W(x,y)}{\partial W} & \text{when } F_W(x,y) > 0, F_W(x,\bar{y}) \geq m \\ -2(m - F_W(x,\bar{y}))\frac{\partial F_W(x,\bar{y})}{\partial W} & \text{when } F_W(x,y) \leq 0, F_W(x,\bar{y}) < m \\ 0 & \text{when } F_W(x,y) \leq 0, F_W(x,\bar{y}) \geq m \end{cases}$$

d

a

NLL loss considers the probability of y given x while computing the gradient while the other losses does not.

b

The margin ensures that the energy for the correct pair of input and label is at least m-smaller than the energy for the incorrect ones. If there were no m, then there wouldn't be a difference between the correct and incorrect data points. We only choose the positive part because the energy for the correct pair is not smaller than the energy for the incorrect pairs by at least m. Only in such case we want the gradients to be passed so that we will continue to push up the energy for the incorrect ones and push down the energy for the correct ones. If the expression evaluates to 0 or negative, it means that the energy for the correct ones is at least m-smaller than the incorrect ones, which means it met the objective.

c

The hinge loss combined the energy for the correct and incorrect pairs while the other two separated them. Also, the simple loss and square-square loss tries to situate the energy for the correct pairs to 0 or smaller, and the incorrect pairs to m or larger; hinge loss only cares about the distance between them. Simple Loss is much more robust while square-square is scaled by the energy. On the other hand, Simple Loss suffers from vanishing gradient, while square-square does not.