从NCE到InfoNCE

2021 年 10 月 30 日

Huacheng Li

摘要

NCE的作用就是

表 1: 符号总结

Symbol	Defination
$\tilde{p}(w\|c)$	从句子(特定上下文)中取样的w的数据分布（经验分布）

1 从NLP视角引入NCE

1.1 n-gram

语言模型假设所有可能的句子服从一个概率分布，每个句子出现的概率之和为1。句子$s = \{w_1, w_2, \ldots, w_m\}$，其出现的概率可以表示为公式 (1)，其中$c_i$表示单词$w_i$的上下文。

$$\begin{aligned} p(s) = p(w_1, w_2, \ldots, w_m) &= p(w_1) * p(w_2|w_1) * p(w_3|w_1, w_2) * \cdots * p(w_m|w_1, \ldots, w_{m-1}) \\ &= \prod_{i=1}^{m} p(w_i|w_1, \ldots, w_{i-1}) \\ &= \prod_{i=1}^{m} p(w_i|c_i) \end{aligned} \tag{1}$$

n-gram假设一个词的上下文只与前n个词有关，因此n-gram可以将模型进化为公式(2)

$$p(w_1, w_2, \ldots, w_m) = \prod_{i=1}^{m} p(w_i|w_{i-n+1}, \ldots, w_{i-1}) \tag{2}$$

1.2 最大似然估计

n-gram模型建立之后，根据语料库中w_i和c_i，计算$p(w_i|c_i)$。 在计算一个句子出现的概率时，将这个句子中出现的$p(w_i|c_i)$连乘。 因此，可以不计算每个w和c的概率，而是建立一个目标函数，根据语料库去求解这些概率。可以将w和c的关系看成一个依赖参数θ的条件分布$p_\theta(w_i|c_i)$。因此优化目标可以如下所示

$$\mathcal{L}_{MLE} = \sum_{w_i \in s} \log p_\theta(w_i|c_i) \tag{3}$$

条件概率$p_\theta(w_i|c_i)$可以看作参数为θ的函数$F(w, c; \theta)$。通过最优化参数θ^*得到函数F。

1.3 神经概率语言模型

上述方法问题主要在于两点：

- 如何构造函数F

- 最大似然估计理论可行，但计算量太大

针对第一个问题，引入神经网络来拟合函数F，[Bengio et al., 2003]等人提出NPLM (Neural Probabilistic Language Model)，不再受限与gram的大小，可以包含任意大小上下文的情况建模词的条件概率$p(w|c)$。单词库$V = \{v_1, v_2, \ldots, v$将$(w, c)$看作训练样本，通过神经网络和softmax后会输出一个向量$\hat{y}_i = \{\hat{y}_{i,1}\}$。每一维$\hat{y}_{i,j} = p(v_j|c_i)$表示当上下文为$c_i$时，对应的第$i$个位置的单词$w_i$是单词库中第$j$个单词$v_j$的概率。训练过程要求最后单词库中概率最大的单词就是训练样本中对应的单词w_i。因此，**NPLM将语言模型的建立看作是多分类问题**。

假定输入softmax之前的结果用$s_\theta(w, c)$表示，那么w的条件概率可以表示为：

$$p_\theta(w|c) = \frac{e^{s_\theta(w,c)}}{\sum_{w' \in V} e^{s_\theta(w',c)}} = \frac{u_\theta(w, c)}{Z(c)} \tag{4}$$

其中，$u_\theta(w, c) = e^{s_\theta(w,c)}$表示下一个单词是这个单词$w$的概率，$Z(c) = \sum_{w' \in V} e^{s_\theta(w^{'},c)}$表示当前单词库中所有单词的概率累计，也叫配分函数或归一化因子。 **由于单词库规模很大，配分函数难以计算，因此引入NCE**。

将公式(4)看作函数F的具体形式，使用章节1.2提到的最大似然估计求解参数θ。 **将从句子c中取样的关于w的分布看作经验分布**$\tilde{p}(w|c)$，可以将公式(3)重写为下式：

$$\mathcal{L}_{MLE} = \sum_{w \sim \tilde{p}(w|c)} \log p_\theta(w|c) = \mathbb{E}_{w \sim \tilde{p}(w|c)} \log \frac{u_\theta(w, c)}{Z(c)} \tag{5}$$

最大化经验分布即是对θ求导：

$$\begin{aligned} \frac{\partial}{\partial \theta} \mathcal{L}_{MLE} &= \mathbb{E}_{w \sim \tilde{p}(w|c)} \frac{\partial}{\partial \theta} \log \frac{u_\theta(w, c)}{Z(c)} \\ &= \mathbb{E}_{w \sim \tilde{p}(w|c)} [\frac{\partial}{\partial \theta} \log u_\theta(w, c) - \frac{\partial}{\partial \theta} \log Z(c)] \\ &= \mathbb{E}_{w \sim \tilde{p}(w|c)} \frac{\partial}{\partial \theta} \log u_\theta(w, c) - \frac{\partial}{\partial \theta} \log Z(c) \end{aligned} \tag{6}$$

上式最后一步中，$Z(c) = \sum_{w' \in V} e^{s_\theta(w^{'},c)}$表示当前单词库中所有单词的概率累计，$w^{'}$为单词库中所有的单词，每个单词的概率由$p_\theta(w|c)$产生。 **尽管$w^{'} \sim p_\theta(w|c)$与参数$\theta$有关，但与经验分布$w \sim \tilde{p}(w|c)$无关，因此需要对后一项求导，但可以把期望去掉**。对$Z(c)$求导如下：

$$\begin{aligned} \frac{\partial}{\partial \theta} \log Z(c) &= \frac{1}{Z(c)} \frac{\partial}{\partial \theta} Z(c) \\ &= \frac{1}{Z(c)} \frac{\partial}{\partial \theta} \sum_{w' \in V} u_\theta(w^{'}, c) \\ &= \frac{1}{Z(c)} \frac{\partial}{\partial \theta} \sum_{w' \in V} \exp(u_\theta(w^{'}, c)) \\ &= \sum_{w' \in V} \frac{1}{Z(c)} \exp(s_\theta(w^{'}, c)) \frac{\partial}{\partial \theta} s_\theta(w^{'}, c) \\ &= \sum_{w' \in V} p_\theta(w^{'}, c) \frac{\partial}{\partial \theta} s_\theta(w^{'}, c) \\ &= \mathbb{E}_{w \sim p_\theta(w|c)} \frac{\partial}{\partial \theta} s_\theta(w, c) \text{//因为这里} w \text{和} w^{'} \text{没有区别} \\ &= \mathbb{E}_{w \sim p_\theta(w|c)} \frac{\partial}{\partial \theta} \log u_\theta(w, c) \text{//这个和上一个等价，一个是指数，一个是对数} \end{aligned} \tag{7}$$

将公式(7)代回公式(6)，可以得到下式：

$$\begin{aligned} \frac{\partial}{\partial \theta} \mathcal{L}_{MLE} &= \mathbb{E}_{w \sim \tilde{p}(w|c)} \frac{\partial}{\partial \theta} \log u_\theta(w, c) - \frac{\partial}{\partial \theta} \log Z(c) \\ &= \mathbb{E}_{w \sim \tilde{p}(w|c)} \frac{\partial}{\partial \theta} \log u_\theta(w, c) - \mathbb{E}_{w \sim p_\theta(w|c)} \frac{\partial}{\partial \theta} \log u_\theta(w, c) \\ &= \sum_w \tilde{p}(w|c) \frac{\partial}{\partial \theta} \log u_\theta(w, c) - \sum_w p_\theta(w, c) \frac{\partial}{\partial \theta} \log u_\theta(w, c) \\ &= \sum_w [(\tilde{p}(w|c) - p_\theta(w|c)) \frac{\partial}{\partial \theta} \log u_\theta(w, c)] \end{aligned} \tag{8}$$

如上式所示，最终还是绕不开归一化常数$Z(c)$

2 什么是NCE

为解决上一章中$Z(c)$计算复杂的问题，有以下几种思路：

- 将$Z(c)$看作一个参数训练：**不可行**，因为由公式(5)可以看出，训练过程有可能会直接让$Z(c)$趋向于0
- 不定义$Z(c)$，直接使用$u_\theta(w,c)$估计模型：不在文本讨论范围，如 Contrastive Divergence [Hinton, 2002]，Score Matching [Hyvärinen, 2005]。
- NCE：核心思想就是通过学习数据分布样本和噪声分布样本之间的差别，从而发现数据中的特性。通过最大化同一个目标函数来估计模型参数θ和归一化常数。*NCE将Z(c)推断问题转化为二分类问题，训练分类器能够对数据样本和噪声样本进行二分类，这样分类器的参数θ等价于我们要学习的参数θ*

假设特定上下文c中取出的**正样本**数据分布为$\tilde{p}(w|c)$，令其为类别$D=1$；与上下文无关的噪声分布为$q(w)$，即随机采的，取出的为**负样本**，类别为$D=0$。如果取出的正样本数和负样本数分别为k_d和k_n。正负样本混合得到混合分布$p(w|c)$。

$$\begin{cases} p(D=1)=\frac{k_d}{k_d+k_n} \\ p(D=0)=\frac{k_n}{k_d+k_n} \\ p(w|D=1,c)=\tilde{p}(w|c) \\ //\text{上式可以理解为，已知取正例，且从上下文}c\text{相关空间取样的概率} \\ p(w|D=0,c)=q(w) \end{cases} \tag{9}$$

所以后验概率可以计算为

$$\begin{aligned} p(D=0|w,c) &= \frac{p(D=0,w|c)}{p(w|c)} \\ &= \frac{p(D=0)p(w|D=0,c)}{p(D=0)p(w|D=0,c)+p(D=1)p(w|D=1,c)} \\ &= \frac{\frac{k_n}{k_n+k_d}\times q(w)}{\frac{k_d}{k_d+k_n}\times\tilde{p}(w|c)+\frac{k_n}{k_n+k_d}\times q(w)} \\ &= \frac{\frac{k_n}{k_d}\times q(w)}{\tilde{p}(w|c)+\frac{k_n}{n_d}\times q(w)} \end{aligned} \tag{10}$$

$$\begin{aligned} p(D=1|w,c) &= \frac{p(D=1,w|c)}{p(w|c)} \\ &= \frac{p(D=1)p(w|D=1,c)}{p(D=0)p(w|D=0,c)+p(D=1)p(w|D=1,c)} \\ &= \frac{\frac{k_d}{k_d+k_n}\times\tilde{p}(w|c)}{\frac{k_d}{k_d+k_n}\times\tilde{p}(w|c)+\frac{k_n}{k_n+k_d}\times q(w)} \\ &= \frac{\tilde{p}(w|c)}{\tilde{p}(w|c)+\frac{k_n}{n_d}\times q(w)} \end{aligned} \tag{11}$$

令负样本和正样本数目的比例为$k=\frac{k_n}{k_d}$：

$$\begin{cases} p(D=0|w,c)=\frac{k\times q(w)}{\tilde{p}(w|c)+k\times q(w)} \\ p(D=1|w,c)=\frac{\tilde{p}(w|c)}{\tilde{p}(w|c)+k\times q(w)} \end{cases} \tag{12}$$

从公式(12)可以看出，**NCE是将式中的经验分布$\tilde{p}(w|c)$来替换为概率模型$p_\theta(w|c)$，使后验概率成为参数为θ的函数**，如公式(13)。 NCE做了两个假设：

- 将$Z(c)$作为一个参数z_c来估计，相当于引进了新的参数。
- 将z_c固定为1. 这个设定减少了参数的数量，也使模型符合归一化的性质，即$Z(c)\approx 1$。

那么由公式(4)，可以得到$p_\theta(w|c) = u_\theta(w|c)$。由此，公式(12)可以表示为

$$\begin{cases} p_\theta(D=0|w,c) = \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \\ p_\theta(D=1|w,c) = \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} \end{cases} \tag{13}$$

假设标签D_t服从伯努利分布，其对数似然函数$\mathcal{L}^c_{NCE}$可以表示如下：

$$\begin{aligned} \mathcal{L}^c_{NCE} &= \sum_{t=1}^{k_d+k_n} [D_t \log P(D=1|w_t,c) + (1-D_t)\log P(D=0|w_t,c)] \\ &= \sum_{t=1}^{k_d} \log P(D=1|w_t,c) + \sum_{t=1}^{k_n} \log P(D=0|w_t,c) \\ &= \sum_{t=1}^{k_d} \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + \sum_{t=1}^{k_n} \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \end{aligned} \tag{14}$$

NCE的目标函数要在上式的基础上除以正样本的数量k_d，我认为这里是要假设正样本为*1*，求的是概率，因而要去掉不同样本数量的影响。

$$\begin{aligned} \mathcal{J}^c_{NCE} &= \frac{1}{k_d}[\sum_{t=1}^{k_d} \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)}] \\ &= \frac{1}{k_d} \sum_{t=1}^{k_d} \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + \frac{1}{k_d} \sum_{t=1}^{k_n} \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \\ &= \frac{1}{k_d} \sum_{t=1}^{k_d} \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + \frac{k}{k_n} \sum_{t=1}^{k_n} \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \end{aligned} \tag{15}$$

因此，当数据数量很大时，上式可以写为公式(16)，k实际上就是设置二分类问题时选取的负样本与正样本的比例。文章[Gutmann and Hyvärinen, 2012]有如下结论：

- 噪声分布$q(w)$应当尽可能接近数据分布$p(\tilde{w}|c)$，否则分类任务过于简单，也很难学习到数据特性。
- 负样本和正样本数量之比k越大，则NCE对噪声分布好坏的依赖程度越小。因此应当尽可能增大比值k。

$$\begin{aligned} \mathcal{J}^c_{NCE} &= \frac{1}{k_d} \sum_{t=1}^{k_d} \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + \frac{k}{k_n} \sum_{t=1}^{k_n} \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \\ &= \mathbb{E}_{w \sim \tilde{p}(w|c)} \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + k\mathbb{E}_{w \sim q(w)} \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \\ &= \mathbb{E}_{w \sim \tilde{p}(w|c)} \log \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + k\mathbb{E}_{w \sim q(w)} \log \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \text{//转化为log，即最大化对数似然函数} \end{aligned} \tag{16}$$

同时NCE训练时应当考虑所有上下文，如公式(17)所示。总结就是从上下文c中取出单词作为正样本，从噪声分布中取出的单词作为负样本，正负样本数量之比为$1:k$，然后训练二分类器，通过类比交叉熵的损失函数的目标函数来进行训练。

$$\mathcal{J}_{NCE} = \sum_c P(c) \mathcal{J}^c_{NCE} \tag{17}$$

3 NCE原理

这一章推理为什么NCE是正确的。对公式(16)求导如公式(18)：

$$\begin{aligned} \frac{\partial}{\partial \theta} \mathcal{J}^c_{NCE}(\theta) &= \frac{\partial}{\partial \theta} [\mathbb{E}_{w \sim \tilde{p}(w|c)} \log \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + k\mathbb{E}_{w \sim q(w)} \log \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)}] \\ &= \frac{\partial}{\partial \theta} \sum_{w \sim \tilde{p}(w|c)} \tilde{p}(w|c) \log \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + \frac{\partial}{\partial \theta} k \sum_{w \sim q(w)} q(w) \log \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \\ &= \sum_{w \sim \tilde{p}(w|c)} \tilde{p}(w|c) \frac{\partial}{\partial \theta} \log \frac{u_\theta(w,c)}{u_\theta(w,c)+k \times q(w)} + k \sum_{w \sim q(w)} q(w) \frac{\partial}{\partial \theta} \log \frac{k \times q(w)}{u_\theta(w,c)+k \times q(w)} \end{aligned} \tag{18}$$

对上式左右两边展开：

$$\begin{aligned}
\frac{\partial}{\partial\theta}\log\frac{u_\theta(w,c)}{u_\theta(w,c)+k\times q(w)} &= -\frac{\partial}{\partial\theta}\log(1+\frac{k\times q(w)}{u_\theta(w,c)}) \\
&= -\frac{1}{1+\frac{k\times q(w)}{u_\theta(w,c)}}\frac{\partial}{\partial\theta}\frac{k\times q(w)}{u_\theta(w,c)} \\
&= -\frac{1}{1+\frac{k\times q(w)}{u_\theta(w,c)}}(k\times q(w))\frac{-1}{[u_\theta(w,c)]^2}\frac{\partial}{\partial\theta}\frac{1}{u_\theta(w,c)} \\
&= \frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}\frac{1}{u_\theta(w,c)}\frac{\partial}{\partial\theta}\frac{1}{u_\theta(w,c)} \\
&= \frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}\log u_\theta(w,c)
\end{aligned} \tag{19}$$

$$\begin{aligned}
\frac{\partial}{\partial\theta}\log\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)} &= -\frac{\partial}{\partial\theta}\log(1+\frac{u_\theta(w,c)}{k\times q(w)}) \\
&= -\frac{1}{1+\frac{u_\theta(w,c)}{k\times q(w)}}\frac{\partial}{\partial\theta}\frac{u_\theta(w,c)}{k\times q(w)} \\
&= -\frac{1}{1+\frac{u_\theta(w,c)}{k\times q(w)}}\frac{1}{k\times q(w)}\frac{\partial}{\partial\theta}u_\theta(w,c) \\
&= -\frac{1}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}u_\theta(w,c) \\
&= -\frac{u_\theta(w,c)}{u_\theta(w,c)+k\times q(w)}\frac{1}{u_\theta(w,c)} \\
&= -\frac{u_\theta(w,c)}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}\log u_\theta(w,c)
\end{aligned} \tag{20}$$

将公式(19)和(20) 结果代回公式(18)，且根据$Z(c)\approx 1$的设定，即$p_\theta(w,c)=u_\theta(w,c)$

$$\begin{aligned}
\frac{\partial}{\partial\theta}\mathcal{J}^c_{NCE}(\theta) &= \sum_{w\sim\tilde{p}(w|c)}\tilde{p}(w|c)\frac{\partial}{\partial\theta}\log\frac{u_\theta(w,c)}{u_\theta(w,c)+k\times q(w)} + k\sum_{w\sim q(w)}q(w)\frac{\partial}{\partial\theta}\log\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)} \\
&= \sum_{w\sim\tilde{p}(w|c)}\tilde{p}(w|c)\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}\log u_\theta(w,c) - k\sum_{w\sim q(w)}q(w)\frac{u_\theta(w,c)}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}\log u_\theta(w,c) \\
&= \sum_{w\sim\tilde{p}(w|c)}\tilde{p}(w|c)\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}\log u_\theta(w,c) - \sum_{w\sim q(w)}u_\theta(w,c)\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}\frac{\partial}{\partial\theta}\log u_\theta(w,c) \\
&= \sum_w[\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}(\tilde{p}(w|c)-u_\theta(w,c))\frac{\partial}{\partial\theta}\log u_\theta(w,c)] \\
&= \sum_w[\frac{k\times q(w)}{u_\theta(w,c)+k\times q(w)}(\tilde{p}(w|c)-p_\theta(w|c))\frac{\partial}{\partial\theta}\log u_\theta(w,c)]
\end{aligned} \tag{21}$$

因此，当$k\to\infty$，如公式(22)所示，其形式与公式(8)类似。

$$\begin{aligned}
\lim_{k\to\infty}\frac{\partial}{\partial\theta}\mathcal{J}^c_{NCE}(\theta) &= \lim_{k\to\infty}\sum_w[\frac{q(w)}{\frac{u_\theta(w,c)}{k}+q(w)}(\tilde{p}(w|c)-p_\theta(w|c))\frac{\partial}{\partial\theta}\log u_\theta(w,c)] \\
&= \sum_w[(\tilde{p}(w|c)-p_\theta(w|c))\frac{\partial}{\partial\theta}\log u_\theta(w,c)]
\end{aligned} \tag{22}$$

4 从NCE到InfoNCE

InfoNCE是在论文 [van den Oord et al., 2018] 提出的，主要用于对比预测编码(Contrastive Predictive Coding, CPC)。 CPC核心思想是通过无监督任务来学习（编码）高维数据特征，通常采取无监督策略根据上下文预测未来或缺失的信息。

要构建这样的预测任务，直接的方法是建模条件生成模型$p(x_{t+k}|c_t)$，即根据当前上下文c_t预测k个时刻之后的数据x_{t+k}。但是这种思路过于细节，因此作者引入**互信息**的思想，最大化当前上下文c_t和未来数据x_{t+k}之间的互信息来预测。公式(23)是互信息的表示，由于无法知道联合概率分布，因此要最大化互信息，需要最大化$\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$

$$I(x_{t+k};c_t)=\sum_{x,c}p(x_{t+k})\log\frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \tag{23}$$

定义$\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$为密度比，分子相当于$p_d$，相当于目标函数，分母相当于$p_n$，相当于参考分布，即噪声。因此，根据NCE中的思路，将问题转化为二分类问题。具体而言：

- 从条件概率$p(x_{t+k}|c_t)$中采出的数据称为正样本，它是根据上下文c_t做出的预测数据，将正样本与上下文一起组成正样本对，类别标签设置为1.
- 将从$p(x_{t+k})$中取出的样本称为负样本，是与当前上下文没有必然关系的随机数据，与上下文组成负样本对，标签设置为0.
- 正样本也就是与c_t间隔步长为k的数据，根据NCE中的规定，正样本选取1个；因为因为NCE中证明噪声分布和数据分布越接近越好，因此负样本直接在序列中随机采样，且采的越多越好。

4.1 举例

假设一组数量为N的$X=\{x_1,\dots,x_N\}$。其中包含1个从$p(x_{t+k}|c_t)$中取正样本和$N-1$个从指定分布(用于对比的噪声分布)$p(x_{t+k})$。假设第x_i是正样本，且$i=t+k$，上下文c_t表示t之前的数据，那么能够正确的同时找到一个正样本和$N-1$个负样本的情况可以写成如下形式：

$$\begin{aligned}p(d=i|X,c_t)&=p(x_{t+k}|c_t)\\&=\frac{p(x_{t+k}|c_t)\prod_{l\neq t+k}p(x_l)}{\sum_{j=1}^{N}p(x_j|c_t)\prod_{l\neq j}p(x_l)}\\&=\frac{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}}{\sum_{j=1}^{N}\frac{p(x_j|c_t)}{p(x_j)}}\end{aligned} \tag{24}$$

最大化上式可以最大化模型分辨出每个正负样本的能力，也就是最大化密度比，最大化互信息。

参考上文公式(4)，可以将表达式写为softmax形式，

$$\begin{aligned}p(x_{t+k}|c_t)&=\frac{\exp(s_\theta(x_{t+k},c_t))}{\sum_{x_j\in X}\exp(s_\theta(x_j,c_t))}\\&=\frac{f_k(x_{t+k},c_t)}{\sum_{x_j\in X}f_k(x_j,c_t)}\end{aligned} \tag{25}$$

根据公式(24)，函数$f_k(x_j,c_t)$可以看作密度比$\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$的一种形式。【即使不直接等价，但含义相关】，即

$$f_k(x_{t+k},c_t)\propto\frac{p(x_{t+k}|c_t)}{p(x_{t+k})} \tag{26}$$

优化目标时候是使公式(24)和(26)最大，因此交叉熵损失函数可以写为：

$$\begin{aligned}\mathcal{L}_N&=-\sum_X[p(x,c)\log\frac{f_k(x_{t+k},c_t)}{\sum_{x_j\in X}f_k(x_j,c_t)}]\\&=-\mathbb{E}_X[\log\frac{f_k(x_{t+k},c_t)}{\sum_{x_j\in X}f_k(x_j,c_t)}]\end{aligned} \tag{27}$$

上式为InfoNCE损失函数，最小化InfoNCE的Loss，也就相当于最大化x_{t+k}和c_t之间的互信息的下限，也就是最大化互信息$I(x_{t+k};c_t)$。

$$
\begin{aligned}
\mathcal{L}_N^{opt} &= -\mathbb{E}_X \log[\frac{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})}}{\frac{p(x_{t+k}|c_t)}{p(x_{t+k})} + \sum_{x_j \in X_{neg}} \frac{p(x_j|c_t)}{p(x_j)}}] \\
&= \mathbb{E}_X \log[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)} \sum_{x_j \in X_{neg}} \frac{p(x_j|c_t)}{p(x_j)}] \\
&\approx \mathbb{E}_X \log[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)}(N-1)\mathbb{E}_{x_j} \frac{p(x_j|c_t)}{p(x_j)}] \\
&= \mathbb{E}_X \log[1 + \frac{p(x_{t+k})}{p(x_{t+k}|c_t)}(N-1)]//\text{因为负例随机取，与上下文无关} \\
&\geq \mathbb{E}_X \log[\frac{p(x_{t+k})}{p(x_{t+k}|c_t)}N] \\
&= -I(x_{t+k}, c_t) + \log(N)
\end{aligned}
\tag{28}
$$

5 附录：

NCE实际是要解决归一化参数密度估计问题

参考文献

[Bengio et al., 2003] Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155.

[Gutmann and Hyvärinen, 2012] Gutmann, M. and Hyvärinen, A. (2012). Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *J. Mach. Learn. Res.*, 13:307–361.

[Hinton, 2002] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Comput.*, 14(8):1771–1800.

[Hyvärinen, 2005] Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.*, 6:695–709.

[van den Oord et al., 2018] van den Oord, A., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748.