Collective Noise Contrastive Estimation for Policy Transfer Learning

Supplementary Material

Weinan Zhang, Ulrich Paquet, Katja Hofmann

Detailed Derivations of Noise Contrastive Estimation

As discussed in the *Objectives* section of the paper, the gradient of objectives 1 and 2 w.r.t. the parameter θ involves the calculation of

$$\begin{aligned}\frac{\partial}{\partial\theta}\log\frac{e^{s_\theta(i,j,r)}}{\sum_{j'}e^{s_\theta(i,j',r)}} &= \frac{\partial s_\theta(i,j,r)}{\partial\theta} - \frac{\partial}{\partial\theta}\log\sum_{j'}e^{s_\theta(i,j',r)}\\ &= \frac{\partial s_\theta(i,j,r)}{\partial\theta} - \frac{\sum_{j'}e^{s_\theta(i,j',r)}\frac{\partial s_\theta(i,j',r)}{\partial\theta}}{\sum_{j'}e^{s_\theta(i,j',r)}}\\ &= \frac{\partial s_\theta(i,j,r)}{\partial\theta} - \mathbb{E}_{P_\theta(j'|i,r)}\Big[\frac{\partial s_\theta(i,j',r)}{\partial\theta}\Big], \end{aligned} \tag{1}$$

where the scoring function is implemented as

$$s_\theta(i,j,r) = r\cdot w_i^T w_j + b_j, \tag{2}$$

and θ denotes the general representation of parameters (w_i, w_j, b_j).

Unfortunately, the term $\mathbb{E}_{P_\theta(j'|i,r)}\Big[\frac{\partial s_\theta(i,j',r)}{\partial\theta}\Big]$ in Eq. (1) is of high computational complexity as it needs to iterate over all artists j' for every training instance (i,j,r). Noise-contrastive estimation (NCE) [1] addresses this problem by proposing a new objective with an easily calculated gradient that approximates the gradient in Eq. (1).

The basic idea of NCE is as follows. Instead of learning a model that maximises the log-likehood of the observed data, it learns a model that distinguishes the data samples from some artificially generated noise samples.

Assume each observation (i,j,r) is generated from a mixture of some true data distribution $P_d(j|i,r)$ and a noise distribution $P_n(j)$. Following [2], the method assumes that the noise samples are k times as frequent as data samples, so that the observations (i,j,r) come from a mixture $\frac{1}{k+1}P_d(j|i,r) + \frac{k}{k+1}P_n(j)$. We now want to use a probabilistic model $P_\theta(j|i,r)$ to approximate $P_d(j|i,r)$. Therefore, for every context i with the desired reward r, given the observed M_d true data with M_n noise data, the log-likelihood of correctly distinguishing the labels of these observations with respect to parameter θ is

$$\mathcal{L}^i(\theta) = \sum_{m=1}^{M_d}\log\frac{P_\theta(j_m|i,r)}{P_\theta(j_m|i,r)+kP_n(j_m)} + \sum_{m=1}^{M_n}\log\frac{kP_n(j_m)}{P_\theta(j_m|i,r)+kP_n(j_m)} \tag{3}$$

Now define the target function $J^i_{M_d,M_n}(\theta)$

$$J^i_{M_d,M_n}(\theta) = \frac{1}{M_d}\sum_{m=1}^{M_d}\log\frac{P_\theta(j_m|i,r)}{P_\theta(j_m|i,r)+kP_n(j_m)} + \frac{k}{M_n}\sum_{m=1}^{M_n}\log\frac{kP_n(j_m)}{P_\theta(j_m|i,r)+kP_n(j_m)} \tag{4}$$

Taking the limit $M_d, M_n \to +\infty$, we have

$$J^i(\theta) = \mathbb{E}_{P_d(j|i,r)}\Big[\log\frac{P_\theta(j|i,r)}{P_\theta(j|i,r)+kP_n(j)}\Big] + k\mathbb{E}_{P_n(j)}\Big[\log\frac{kP_n(j)}{P_\theta(j|i,r)+kP_n(j)}\Big] \tag{5}$$

which is just the expected log-likelihood of the policy $P_\theta(j|i,r)$ distinguishing the true data samples from the noise samples given the context i and desired reward r.

Take the derivative of $J^i(\theta)$ w.r.t. θ:

$$
\begin{aligned}
\frac{\partial J^i(\theta)}{\partial\theta} &= \mathbb{E}_{P_d(j|i,r)}\Big[\frac{kP_n(j)}{P_\theta(j|i,r)+kP_n(j)}\frac{\partial}{\partial\theta}\log P_\theta(j|i,r)\Big] - k\mathbb{E}_{P_n(j)}\Big[\frac{P_\theta(j|i,r)}{P_\theta(j|i,r)+kP_n(j)}\frac{\partial}{\partial\theta}\log P_\theta(j|i,r)\Big] \\
&= \sum_j \frac{kP_n(j)P_d(j|i,r)}{P_\theta(j|i,r)+kP_n(j)}\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) - \sum_j \frac{kP_\theta(j|i,r)P_n(j)}{P_\theta(j|i,r)+kP_n(j)}\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) \\
&= \sum_j \frac{kP_n(j)}{P_\theta(j|i,r)+kP_n(j)}(P_d(j|i,r)-P_\theta(j|i,r))\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) \qquad (6)
\end{aligned}
$$

When $k \to +\infty$, then

$$
\begin{aligned}
\frac{\partial J^i(\theta)}{\partial\theta} &\to \sum_j (P_d(j|i,r)-P_\theta(j|i,r))\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) \\
&= \sum_j P_d(j|i,r)\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) - \sum_{j'} P_\theta(j'|i,r)\frac{\partial}{\partial\theta}\log P_\theta(j'|i,r) \\
&= \sum_j P_d(j|i,r)\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) - \mathbb{E}_{P_\theta(j'|i,r)}\frac{\partial}{\partial\theta}\log P_\theta(j'|i,r) \\
&= \sum_j P_d(j|i,r)\Big(\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) - \mathbb{E}_{P_\theta(j'|i,r)}\Big[\frac{\partial}{\partial\theta}\log P_\theta(j'|i,r)\Big]\Big) \qquad (7) \\
&= \sum_j P_d(j|i,r)\Big(\frac{\partial}{\partial\theta}\log P_\theta(j|i,r) + \frac{\partial}{\partial\theta}\log\sum_k e^{s_\theta(i,k,r)} - \frac{\partial}{\partial\theta}\log\sum_l e^{s_\theta(i,l,r)} - \mathbb{E}_{P_\theta(j'|i,r)}\Big[\frac{\partial}{\partial\theta}\log P_\theta(j'|i,r)\Big]\Big) \\
&= \sum_j P_d(j|i,r)\Big(\frac{\partial}{\partial\theta}\log\Big(P_\theta(j|i,r)\sum_k e^{s_\theta(i,k,r)}\Big) - \mathbb{E}_{P_\theta(j'|i,r)}\Big[\frac{\partial}{\partial\theta}\log\Big(P_\theta(j'|i,r)\sum_l e^{s_\theta(i,l,r)}\Big)\Big]\Big) \\
&= \sum_j P_d(j|i,r)\Big(\frac{\partial}{\partial\theta}\log e^{s_\theta(i,j,r)} - \mathbb{E}_{P_\theta(j'|i,r)}\Big[\frac{\partial}{\partial\theta}\log e^{s_\theta(i,j',r)}\Big]\Big) \qquad (8) \\
&= \sum_j P_d(j|i,r)\Big(\frac{\partial}{\partial\theta}s_\theta(i,j) - \mathbb{E}_{P_\theta(k|i,r)}\Big[\frac{\partial}{\partial\theta}s_\theta(i,k)\Big]\Big) \\
&= \sum_j P_d(j|i,r)\frac{\partial\log P_\theta(j|i,r)}{\partial\theta} \qquad \text{by Eq. (1)} \qquad (9)
\end{aligned}
$$

which is the gradient of maximising the log-likelihood of generating the observed data.

Specifically, following [3], we can regard the normalisation factor for (i,r) a parameter $c_{i,r}$

$$c_{i,r} = \frac{1}{\sum_j e^{s_\theta(i,j,r)}}. \qquad (10)$$

As such,

$$P_\theta(j|i,r) = e^{s_\theta(i,j,r)} \cdot e^{c_{i,r}}. \qquad (11)$$

We can now calculate the gradient of $c_{i,r}$ just like that of other parameters. In fact, from the equivalence between Eq. (7) and Eq. (8), we can see that directly setting $P_\theta(j|i,r) = e^{s_\theta(i,j,r)}$ is no problem (also implemented in [3]).

In practice, given an observation $(i,j,1)$ from playlist data or an observation (i,j,r) from radio data, we generate k artists $j_1, j_2, \dots, j_k$ from a known distribution $P_n(j)$, e.g. by artist popularity, as noise data. Then we can calculate the gradient

$$\frac{\partial}{\partial\theta}J^{(i,j,r)}_{\text{NCE}}(\theta) = \frac{kP_n(j)}{e^{s_\theta(i,j,r)}+kP_n(j)}\frac{\partial}{\partial\theta}s_\theta(i,j,r) - \sum_{m=1}^{k}\frac{e^{s_\theta(i,j_m,r)}}{e^{s_\theta(i,j_m,r)}+kP_n(j_m)}\frac{\partial}{\partial\theta}s_\theta(i,j_m,r), \qquad (12)$$

which corresponds to Eq. (15) in our paper. The resulting method is very efficient if k is not large. In addition, both kinds of weights, i.e., $\frac{kP_n(j)}{e^{s_\theta(i,j,r)}+kP_n(j)}$ and $\frac{e^{s_\theta(i,j_m,r)}}{e^{s_\theta(i,j_m,r)}+kP_n(j_m)}$, on the partial derivatives are between 0 and 1, which makes NCE very stable.

NCE is a learning framework which can flexibly to incorporate different scoring functions s. The general update rule is

$$\theta \leftarrow \theta + \eta \frac{\partial}{\partial \theta} J_{\text{NCE}}^{(i,j,r)}(\theta), \tag{13}$$

where η is the learning rate. With the scoring function as in Eq. (2), the detailed gradients are

$$\frac{\partial}{\partial b_j} J_{\text{NCE}}^{(i,j,r)} = \frac{kP_n(j,r)}{e^{s_\theta(i,j,r)} + kP_n(j,r)} \tag{14}$$

$$\frac{\partial}{\partial b_{j_m}} J_{\text{NCE}}^{(i,j,r)} = -\frac{e^{s_\theta(i,j_m,r_m)}}{e^{s_\theta(i,j_m,r_m)} + kP_n(j_m,r_m)} \tag{15}$$

$$\frac{\partial}{\partial w_i} J_{\text{NCE}}^{(i,j,r)} = \frac{kP_n(j,r)}{e^{s_\theta(i,j,r)} + kP_n(j,r)} \cdot r \cdot w_j - \sum_{m=1}^{k} \frac{e^{s_\theta(i,j_m,r_m)}}{e^{s_\theta(i,j_m,r_m)} + kP_n(j_m,r_m)} \cdot r_m \cdot w_{j_m} \tag{16}$$

$$\frac{\partial}{\partial w_j} J_{\text{NCE}}^{(i,j,r)} = \frac{kP_n(j,r)}{e^{s_\theta(i,j,r)} + kP_n(j,r)} \cdot r \cdot w_i \tag{17}$$

$$\frac{\partial}{\partial w_{j_m}} J_{\text{NCE}}^{(i,j,r)} = -\frac{e^{s_\theta(i,j_m,r_m)}}{e^{s_\theta(i,j_m,r_m)} + kP_n(j_m,r_m)} \cdot r_m \cdot w_i. \tag{18}$$

Note that the regularisation terms should be further added into the gradient calculation.

Example Recommended Radio Streams

Here we further show some example radio streams of 15 artist sequentially recommended by our radio policy trained by collective NCE. To the authors' music knowledge, these radio streams are satisfactory.

Table 1: Case study of 4 recommended radio streams given specific seed artist.

Seed	**Queen**	**Blake Shelton**	**Billy Joel**	**Jessie James**
1	Status Quo	Eric Church	Don Henley	Lila McCann
2	Uriah Heep	James Otto	Mr. Mister	Sara Evans
3	The Romantics	Steve Holy	Little River Band	Jamie O'Neal
4	David Gilmour	Miss Willie Brown	Peter Cetera	Chely Wright
5	Duane Allman	Bobby Pinson	Rita Coolidge	Lorrie Morgan
6	Ash Wednesday	Jason Blaine	Janis Ian	Tanya Tucker
7	Michael Bolton	Chad Brock	Karla Bonoff	K.T. Oslin
8	Lobo	Easton Corbin	Bruce Cockburn	Briston Latina
9	Nils Lofgren	Love And Theft	Orleans	Beat This Summer
10	David Knopfler	John Rich	Nils Lofgren	The Charlie Daniels Band
11	Orleans	Eli Young Band	David Knopfler	The Statler Brothers
12	Bruce Cockburn	Josh Turner	Bob Dylan	Roger Miller
13	Karla Bonoff	Billy Currington	Lobo	Steve Earle
14	Janis Ian	Darius Rucker	Gino Vannelli	June Carter Cash
15	Marc Cohn	Craig Morgan	Rick Springfield	Charlie Louvin

References

[1] M. Gutmann and A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

[2] M. U. Gutmann and A. Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *The Journal of Machine Learning Research*, 13(1):307–361, 2012.

[3] A. Mnih and Y. W. Teh. A fast and simple algorithm for training neural probabilistic language models. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1751–1758, 2012.