Supplementary Material

A. Proofs

In the theoretical analysis, we fix $s_K(x, \theta) = 0$. Then, we only need to consider $C_x \cup N_x = \{1, \ldots, K - 1\}$. Now, the normalization factor becomes

$$E(x, j) = 1 + \sum_{k' \in C_x} e^{s_k(x, \theta)} + e^{s_j(x, \theta)}/q_x(j),$$

with some sampled class $j \in N_x$. Now, we can rewrite $R$ and $\hat{R}$ as

$$R(\theta) = \mathbb{E}_x \sum_{k \in C_x} p(y = k|x) \sum_{j \in N_x} q_x(j) \frac{e^{s_k(x, \theta)}}{E(x, j)} + \sum_{k \in N_x} p(y = k|x) \log \frac{e^{s_k(x, \theta)}}{E(x, k)} + p(y = K|x) \sum_{j \in N_x} q_x(j) \frac{1}{E(x, j)}.$$

$$\hat{R}_n(\theta) = \frac{1}{n} \sum_{i=1}^n \left[ \sum_{k \in C_x} \mathbb{I}(y_i = k) \sum_{j \in N_x} \frac{q_x(j)}{E(x, j)} \log \frac{e^{s_k(x, \theta)}}{E(x, j)} + \sum_{k \in N_x} \frac{1}{E(x, k)} \right].$$

In the proofs, we will use point-wise notations $p_k, s_k, q_k$ and $E_k$ to represent $p(y = k|x), s_k(x, \theta), q_x(k)$ and $E(x, k)$ for simplicity.

A.1. Useful Lemma

We will need the following lemma in our analysis.

**Lemma 1.** For any norm $|| \cdot ||$ defined on the parameter space of $\theta$, assume the quantities $||\nabla \theta s_k||, ||\nabla \theta s_k||$ and $||\nabla \theta s_k||$ for $k = 1, \ldots, K - 1$ are bounded. Then, for any compact set $S$ defined on the parameter space, we have

$$\sup_{\theta \in S} ||\hat{R}_n(\theta) - R(\theta)|| \xrightarrow{p} 0, \quad \sup_{\theta \in S} ||\nabla \hat{R}_n(\theta) - \nabla R(\theta)|| \xrightarrow{p} 0, \quad \text{and} \quad \sup_{\theta \in S} ||\nabla^2 \hat{R}_n(\theta) - \nabla^2 R(\theta)|| \xrightarrow{p} 0.$$

**Proof.** For fixed $\theta$, let

$$\psi(x, y, \theta) = \sum_{j \in C_x} \mathbb{I}(y = k) \sum_{j \in N_x} q_j \log \frac{e^{s_k}}{1 + \sum_{k' \in C_x} e^{s_{k'}} + \frac{e^{s_j}}{q_j}} + \mathbb{I}(y = K) \sum_{j \in N_x} q_j \log \frac{1}{1 + \sum_{k' \in C_x} e^{s_{k'}} + \frac{e^{s_j}}{q_j}}.$$

Then we have $\hat{R}_n(\theta) = \frac{1}{n} \sum_{i=1}^n \psi(x_i, y_i, \theta)$ and $R(\theta) = \mathbb{E}_{x,y} \psi(x, y, \theta)$. By the Law of Large Numbers, we know that $\hat{R}_n(\theta)$ converges point-wisely to $R(\theta)$ in probability.

According to the assumption, there exists a constant $M > 0$ such that

$$||\nabla \theta \psi(x, y, \theta)|| \leq \sum_{k=1}^{K-1} ||\nabla \theta s_k|| \leq M.$$

Given any $\epsilon > 0$, we may find a finite cover $S_x \subset S$ so that for any $\theta \in S$, there exists $\theta' \in S_x$ such that $||\psi(x, y, \theta) - \psi(x, y, \theta')|| \leq M||\theta - \theta'|| < \epsilon$. Since $S_x$ is finite, as $n \to \infty$, $\sup_{\theta \in S_x} ||\hat{R}_n(\theta) - R(\theta)||$ converges to 0 in probability. Therefore, as $n \to \infty$, with probability 1, we have

$$\sup_{\theta \in S} ||\hat{R}_n(\theta) - R(\theta)|| < 2\epsilon + \sup_{\theta \in S_x} ||\hat{R}_n(\theta) - R(\theta)|| \to 2\epsilon.$$

Let $\epsilon \to 0$, we obtain the first bound. The second and the third bounds can be similarly obtained.

A.2. Proof of Theorem 1

**Proof.** $R$ can be re-written as

$$R = \mathbb{E}_x \sum_{j \in N_x} q_j \left( \sum_{k \in C_x} p_k \log \frac{e^{s_k}}{1 + \sum_{k' \in C_x} e^{s_{k'}} + e^{s_j}/q_j} + p_K \log \frac{1}{1 + \sum_{k' \in C_x} e^{s_{k'}} + e^{s_j}/q_j} \right) + \frac{p_j}{q_j} \log \frac{1}{1 + \sum_{k' \in C_x} e^{s_{k'}} + e^{s_j}/q_j}. $$
For $i \in C_x$, we have
\[
\nabla_i R = \mathbb{E}_x \sum_{j \in \mathcal{N}_x} q_j \left[ p_i \left( 1 - \frac{e^{s_i}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j} \right) - \sum_{k \neq i \in \mathcal{C}_x} p_k \frac{e^{s_i}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j} \right] - p_k \frac{e^{s_i}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j} - p_j / q_j \frac{e^{s_j}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j} \right] \\
= \mathbb{E}_x \sum_{j \in \mathcal{N}_x} q_j \left[ p_i \left( p_k + \sum_{k \in \mathcal{C}_x} p_k + p_j / q_j \right) \right] \frac{e^{s_i}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j}.
\]

Similarly, for $j \in \mathcal{N}_x$, we have
\[
\nabla_j R = \mathbb{E}_x q_j \left[ - \left( p_k + \sum_{k \in \mathcal{C}_x} p_k \right) \right] \frac{e^{s_j}/q_j}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j} + p_j / q_j \left( 1 - \frac{e^{s_j}/q_j}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j} \right) \\
= \mathbb{E}_x p_j \left( p_k + \sum_{k \in \mathcal{C}_x} p_k + p_j / q_j \right) \frac{e^{s_j}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s_{k'}} + e^{s_j}/q_j}.
\]

By measuring $s_k = \log \frac{p_k}{p_R}$, we see that $\nabla s_k R = 0$ for $k = 1, \cdots, K - 1$. Therefore, $s_k = \log \frac{p_k}{p_R}$ is an extrema of $R$.

Now, for $i, i' \in C_x$ and $j, j' \in \mathcal{N}_x$, we have
\[
\mathbb{H}_{ii} = \nabla^2_{s_i s_i} R = -\mathbb{E}_x \sum_{j \in \mathcal{N}_x} q_j D_j \frac{e^{s_i}(E_j - e^{s_i})}{E_j^2}, \\
\mathbb{H}_{ii'} = \nabla^2_{s_i s_i} R = \mathbb{E}_x \sum_{j \in \mathcal{N}_x} q_j D_j \frac{e^{s_i} e^{s_{i'}}}{E_j^2}, \\
\mathbb{H}_{ij} = \mathbb{H}_{ji} = \nabla^2_{s_i s_j} R = \nabla^2_{s_j s_i} R = \mathbb{E}_x \sum_{j \in \mathcal{N}_x} D_j \frac{e^{s_i} e^{s_j}}{E_j^2}, \\
\mathbb{H}_{jj} = \nabla^2_{s_j s_j} R = -\mathbb{E}_x D_j \frac{e^{s_j}(E_j - e^{s_j}/q_j)}{E_j^2}, \\
\mathbb{H}_{jj'} = \nabla^2_{s_j s_j} R = 0,
\]

where
\[
D_j = p_k + \sum_{k' \in \mathcal{C}_x} p_k + p_j / q_j.
\]

Now, we can write
\[
\nabla^2 R = \mathbb{E}_x \sum_{j \in \mathcal{N}_x} q_j D_j \frac{E_j}{E_j} \left[ \text{diag}(\mathbf{v}_j) - \frac{1}{E_j} \mathbf{v}_j \mathbf{v}_j^\top \right],
\]

where $\mathbf{v}_j = (e^{s_{i_1}}, \cdots, e^{s_{i_{|\mathcal{C}_x|}}}, 0, \cdots, e^{s_j}/q_j, \cdots, 0)^\top$. Let
\[
A_j = \text{diag}(\mathbf{v}_j) - \frac{1}{E_j} \mathbf{v}_j \mathbf{v}_j^\top.
\]
A.4. Proof of Theorem 3

Proof. Let \( \varphi = (\varphi_1, \cdots, \varphi_{K-1})^T \in \mathbb{R}^{K-1} \), we have

\[
\varphi^T A_j \varphi = \sum_{i \in C_x} e^{\xi_i} \varphi_i^2 + \frac{e^{\xi_j}}{q_j} \varphi_j^2 - \frac{1}{E_j} \left( \sum_{i \in C_x} e^{\xi_i} \varphi_i + \frac{e^{\xi_j}}{q_j} \varphi_j \right)^2 \geq \frac{1}{E_j} \left( \sum_{i \in C_x} e^{\xi_i} \varphi_i + \frac{e^{\xi_j}}{q_j} \varphi_j \right)^2 - \frac{1}{E_j} \left( \sum_{i \in C_x} e^{\xi_i} \varphi_i + \frac{e^{\xi_j}}{q_j} \varphi_j \right)^2 > 0,
\]

for every \( j \in N_x \), where the first inequality is by the Cauchy-Schwarz inequality and the second inequality is because \( 0 < \sum_{i \in C_x} e^{\xi_i} + \frac{e^{\xi_j}}{q_j} < E_j \). Therefore, \( -\nabla^2 R = \mathbb{E}_x \sum_{j \in N_x} \frac{D_j}{q_j} A_j \) is positive-definite and \( R \) is strongly concave with respect to \( s \). Hence, \( s_k = \log \frac{p_k}{p_{k \cdot}} \) for \( k = 1, \cdots, K-1 \) is the only maxima of \( R \).

\[
\sqrt{n}(\hat{\theta} - \theta^*) = -\nabla^2 \tilde{R}_n(\hat{\theta})^{-1} \nabla \tilde{R}_n(\theta^*),
\]

(12)

where \( \hat{\theta} = t\theta^* + (1-t)\hat{\theta} \) for some \( t \in [0, 1] \). Note that Lemma 1 implies that \( \nabla^2 \tilde{R}_n(\theta^*)^{-1} \) converges to \( \nabla^2 R(\hat{\theta})^{-1} \) in probability; moreover, \( \hat{\theta} \to \theta^* \) in probability and hence \( \theta \to \theta^* \) in probability. By the Slutsky’s Theorem, the limit distribution of \( \sqrt{n}(\hat{\theta} - \theta^*) \) is given by

\[
-\nabla^2 R(\theta^*)^{-1} \nabla \tilde{R}_n(\theta^*).
\]

Observe that \( \sqrt{n} \nabla \tilde{R}_n(\theta^*) \) is the sum of \( n \) i.i.d. random vectors with mean \( \mathbb{E} \sqrt{n} \nabla \tilde{R}_n(\theta^*) = \sqrt{n} \mathbb{E} \nabla R(\theta^*) = 0 \), and the variance of \( \sqrt{n}(\hat{\theta} - \theta^*) \) is

\[
\text{Var} \left( \sqrt{n}(\hat{\theta} - \theta^*) \right) = \nabla^2 R(\theta^*)^{-1} \text{Var} \left( \sqrt{n} \nabla \tilde{R}_n(\theta^*) \right) \nabla^2 R(\theta^*)^{-1}.
\]
From the proof of Theorem 1, we have
\[
\nabla^2 R(\theta) = -\mathbb{E}_x \nabla \left[ \sum_{j \in \mathcal{N}_x} q_j \frac{D_{ij} A_j}{E_j} \right] \nabla^\top,
\]
where
\[
\nabla = \text{diag} \left( \left( \nabla_i, \cdots, \nabla_{i|\mathcal{C}_x|}, \nabla_j, \cdots, \nabla_{j|\mathcal{N}_x|} \right)^\top \right)
\]
and \( \nabla_k = \nabla \theta_k \).

Measuring \( \nabla^2 R(\theta) \) at \( \theta^* \), we have
\[
\nabla^2 R(\theta^*) = -\mathbb{E}_x \nabla M \nabla^\top
\]
where
\[
M = \sum_{j \in \mathcal{N}_x} q_j \left[ \text{diag}(u_j) - \frac{1}{D_j} u_j u_j^\top \right],
\]
where \( u_j = (p_{i_1}, \cdots, p_{i|\mathcal{C}_x|}, 0, \cdots, p_j/q_j, 0, \cdots)^\top \). By following the proof of Theorem 1, it is easy to show that \( M \succ 0 \) is positive definite.

Next, we derive \( \text{Var} \left( \sqrt{n} \nabla \hat{R}_n(\theta^*) \right) \). Introduce some Bernoulli variables \( Q_j \) for \( j \in \mathcal{N}_x \) with \( p(Q_j = 1|x) = q_j \). Now, for \( i, i' \in C_x \) and \( j, j' \in \mathcal{N}_x \), we have
\[
\mathbb{V}_{ii} = \text{Var} \left( \nabla_i \hat{R}_n(\theta^*), \nabla_i \hat{R}_n(\theta^*) \right)
\]
\[
= \mathbb{E}_x Q \left[ p_i \left( 1 - \frac{e^{s^*_i}}{1 + \sum_{k' \in \mathcal{C}_x} e^{-s^*_i} / q_{k'}} \right)^2 + (D_j - p_i) \left( \frac{e^{s^*_i}}{1 + \sum_{k' \in \mathcal{C}_x} e^{s^*_i} / q_{k'}} \right)^2 \right] \cdot \nabla_i \nabla_{i'}^\top.
\]
\[
\mathbb{V}_{ii'} = \text{Var} \left( \nabla_i \hat{R}_n(\theta^*), \nabla_{i'} \hat{R}_n(\theta^*) \right) = \mathbb{E}_x Q \left[ (D_j - p_i - p_i') \frac{p_i p_i'}{D_j^2} - p_i (1 - \frac{p_i}{D_j}) \frac{p_i'}{D_j} - p_i' (1 - \frac{p_i'}{D_j}) \frac{p_i}{D_j} \right] \cdot \nabla_i \nabla_{i'}^\top.
\]
\[
\mathbb{V}_{jj} = \text{Var} \left( \nabla_j \hat{R}_n(\theta^*), \nabla_j \hat{R}_n(\theta^*) \right) = \mathbb{E}_x Q \left[ \frac{p_j}{q_j} (1 - \frac{p_j/q_j}{D_j})^2 + (D_j - p_j/q_j) \frac{p_j^2/q_j^2}{D_j^2} \right] \cdot \nabla_j \nabla_j^\top.
\]
\[
\mathbb{V}_{jj'} = 0.
\]
\[
\mathbb{V}_{ij} = \mathbb{V}_{ji} = \text{Var} \left( \nabla_i \hat{R}_n(\theta^*), \nabla_j \hat{R}_n(\theta^*) \right)
\]
\[
= \mathbb{E}_x Q \left[ (D_j - p_i - p_j/q_j) \frac{p_j p_j/q_j}{D_j^2} - p_i (1 - \frac{p_i}{D_j}) \frac{p_j/q_j}{D_j} - p_j/q_j (1 - \frac{p_j/q_j}{D_j}) \frac{p_i}{D_j} \right] \cdot \nabla_i \nabla_{i'}^\top.
\]
\[
= -\mathbb{E}_x \sum_{j \in \mathcal{N}_x} \frac{p_j (D_j - p_j/q_j)}{D_j} \cdot \nabla_i \nabla_{i'}^\top.
\]
Now, the variance can be written as

\[ V(\theta^*) = Var \left( \sqrt{n} \nabla \tilde{R}_n(\theta^*) \right) \]

\[ \begin{bmatrix}
  \mathbb{V}_{i_1} & \cdots & \mathbb{V}_{i_1|C_m|} \\
  \vdots & \ddots & \vdots \\
  \mathbb{V}_{i|C_m|} & \cdots & \mathbb{V}_{i|C_m|} \\
  \mathbb{V}_{j_1} & \cdots & \mathbb{V}_{j_1|C_m|} \\
  \vdots & \ddots & \vdots \\
  \mathbb{V}_{j|C_m|} & \cdots & \mathbb{V}_{j|C_m|} \\
  \end{bmatrix} \]

\[ = \begin{bmatrix}
  0 & \cdots & 0 \\
  \vdots & \ddots & \vdots \\
  0 & \cdots & 0 \\
  \end{bmatrix} \]

By comparing \( \nabla^2 R(\theta^*) \) and \( V(\theta^*) \), we immediately have \( -\nabla^2 R(\theta^*) = V(\theta^*) \) and hence

\[ Var \left( \sqrt{n}(\hat{\theta} - \theta^*) \right) = \left[ E_x \nabla M \nabla^T \right]^{-1}. \]

**A.5. Proof of Corollary 1**

**Proof.** By following the proof of Theorem 3, it is easy to show that the statistical variance of the softmax logistic regression in Eq. (1) is \( [E_x \nabla M^{mle} \nabla^T]^{-1} \) (with \( s_K = 0 \) fixed), where

\[ M^{mle} = diag \left( \begin{bmatrix} p_1 \\ \vdots \\ p_{K-1} \end{bmatrix} \right) - \begin{bmatrix} p_1 \\ \vdots \\ p_{K-1} \end{bmatrix}^T. \]

When \( \sum_{k \in C_x \cup \{K\}} p(k, x) \to 1 \), we have \( \sum_{j' \in N_x} p_{j'} \to 0 \) and \( D_j \to 1 \). Then,

\[ M = diag \left( \begin{bmatrix} p_{i_1} \\ \vdots \\ p_{i|C_m|} \\ p_{j_1} \\ \vdots \\ p_{j|N_x|} \end{bmatrix} \right) - \begin{bmatrix} p_{i_1} \sum_{j' \in N_x} p_{j'} \\ \vdots \\ p_{i|C_m|} \sum_{j' \in N_x} p_{j'} \\ p_{j_1} \sum_{j' \in N_x} p_{j'} \\ \vdots \\ p_{j|N_x|} \sum_{j' \in N_x} p_{j'} \end{bmatrix} \]

If we arrange the index order in \( M^{mle} \) according to the index order in \( M \) and denote \( \Delta = M - M^{mle} \), we have

\[ \Delta = \begin{bmatrix} \Delta_1 & \Delta_2 \\ \Delta_2^T & \Delta_3 \end{bmatrix} \to 0, \]

because

\[ \Delta_1 = 0, \]

\[ \Delta_2 = \begin{bmatrix} p_{i_1} (p_{j_1} - \sum_{j' \in N_x} p_{j'}) \\ \vdots \\ p_{i|C_m|} (p_{j|N_x|} - \sum_{j' \in N_x} p_{j'}) \end{bmatrix} \to 0, \]

\[ \Delta_3 = \begin{bmatrix} p_{j_1} (1 - 1/q_{j_1}) \\ \vdots \\ p_{j|N_x|} (1 - 1/q_{j|N_x|}) \end{bmatrix} \to 0. \]

This completes the proof.
B. The Beam Search Algorithm

The beam search algorithm used in both training and testing is depicted in Algorithm 3.

Algorithm 3 The Beam Search Algorithm.

1: **Input:** The root of the tree, input data point $x$ and Beam width $J$.
2: **Output:** The $J$ candidate classes.
3: Initialize stack $S \leftarrow \text{root}$ and stack $S' \leftarrow \emptyset$;
4: Initialize the candidate class set $E \leftarrow \emptyset$;
5: **while** true **do**
6: **if** $S$ is empty **then**
7: Break;
8: **end if**
9: **for** $i = 1$ to $S$.size() **do**
10: **if** $S_i$ is a leaf **then**
11: $E$.pushback($S_i$);
12: **else**
13: **for** $c = 1$ to $S_i$.Child.size() **do**
14: Accumulate the score to $S_i$.Child($c$);
15: $S'$.pushback($S_i$.Child($c$));
16: **end for**
17: **end if**
18: **end for**
19: $S$.clear();
20: **if** $S'$.size() > $J$ **then**
21: // Using the max heap.
22: Find the top-$J$ nodes with the highest accumulated scores in $S'$ and push them into $S$;
23: **else**
24: $S \leftarrow S'$;
25: **end if**
26: $S'$.clear();
27: **end while**
28: // Using the max heap.
29: Return the top-$J$ classes with the highest scores in $E$;

C. A Hierarchical Clustering Method for Generating the Tree Structure

Given the data points of a dataset, we can obtain the center, i.e., the average data point, of each class by scanning the data once and get $\bar{X} \in \mathbb{R}^{K \times d}$, where $K$ is the number of classes and $d$ is the feature dimension. Then, a hierarchical clustering algorithm in Algorithm 4 is performed by viewing each row of $\bar{X}$ as a separate data point. In Algorithm 4, the function ‘Split(root)’ in step 16 has already constructed a $b$-nary tree, which can be used by the Beam Tree Algorithm. However, the clustering algorithm, e.g., the $k$-means algorithm, may generate imbalanced clusters in step 9, and the resulting $b$-nary tree in step 16 may be imbalanced and affect the efficiency of Beam Tree. A simple way to fix this problem is to fetch the labels (leaves) in the tree in step 16 from left to right, where the obtained label order maintains a rough similarity relationship among the classes. We then assign the ordered labels to the leaves of a new balanced $b$-nary tree from left to right.

D. Experimental Details

---

![Training images resized as 224 * 224](image)

*Figure 4.* The neural network structure used for the ImageNet datasets. ‘FC’ indicates fully-connected layer.
Algorithm 4 A Hierarchical Clustering Algorithm for Generating the Tree over Class Labels.

1: Input: $K$, $b$ and $X$.
2: Output: a $b$-nary tree.

3: Function Split(node $o$)
4: while true do
5: if $o$ is assigned with only one label then
6: $o.isleaf = true$;
7: Return;
8: end if
9: Perform any clustering algorithm, e.g., k-means, on the labels associated with the node $o$ and obtain $b$ clusters $\{L_1, \cdots, L_b\}$;
10: Split $o$ into $b$ children $\{o_1, \cdots, o_b\}$ and assign the label clusters $\{L_1, \cdots, L_b\}$ to them respectively;
11: for $i = 1$ to $b$ do
12: Split($o_i$);
13: end for
14: end while

15: Assign root with all labels $\{1, 2, \cdots, K\}$;
16: Split(root);
17: Get the label order in the leaves from left to right;
18: Assign the labels to the leaves of a new balanced $b$-nary tree from left to right;
19: Return the balanced $b$-nary tree;

Hyper-parameter tuning is computationally expensive. In order to efficiently select a good setting of the hyper-parameters, we let each method process half epoch of the training data and use another 10% held-out subset of the training set to tune hyper-parameters. For every classifier, the learning rate $\eta$ needs to be tuned. For the LOMTree method, by following (Choromanska & Langford, 2015), we choose the number of the internal nodes in its binary tree from a set $\{K - 1, 4K - 1, 16K - 1, 64K - 1\}$, and tune the swap resistance from $\{4, 16, 64, 256\}$. The Recall Tree method has a default setting for large class problem in (Daume III et al., 2017), which is also adopted in the experiments.

The VGG-16 network structure used in ImageNet-2010 and ImageNet-10K datasets is provided in Fig. 4. Parameters of Conv layers 1-13, FC14 and FC15 are pre-trained on the ImageNet 2012 dataset.