Loss Functions for Top-k Error: Analysis and Insights

Maksim Lapin,[1] Matthias Hein[2] and Bernt Schiele[1]
[1]Max Planck Institute for Informatics, Saarbrücken, Germany
[2]Saarland University, Saarbrücken, Germany

Abstract

In order to push the performance on realistic computer vision tasks, the number of classes in modern benchmark datasets has significantly increased in recent years. This increase in the number of classes comes along with increased ambiguity between the class labels, raising the question if top-1 error is the right performance measure. In this paper, we provide an extensive comparison and evaluation of established multiclass methods comparing their top-k performance both from a practical as well as from a theoretical perspective. Moreover, we introduce novel top-k loss functions as modifications of the softmax and the multiclass SVM loss and provide efficient optimization schemes for them. In the experiments, we compare on various datasets all of the proposed and established methods for top-k error optimization. An interesting insight of this paper is that the softmax loss yields competitive top-k performance for all k simultaneously. For a specific top-k error, our new top-k losses lead typically to further improvements while being faster to train than the softmax.

1. Introduction

The number of classes is rapidly growing in modern computer vision benchmarks [46, 61]. Typically, this also leads to ambiguity in the labels as classes start to overlap. Even for humans the error rates in top-1 performance are often quite high ($\approx 30\%$ on SUN 397 [59]). While previous research has focused on minimizing top-1 error, we address in this paper top-k error optimization. We are interested in two cases: a) achieving small top-k error for *all* reasonably small k; and b) minimization of a specific top-k error.

While it is argued in [2] that the one-versus-all (OVA) SVM scheme performs on par in top-1 and top-5 accuracy with the other SVM variations based on ranking losses, it has been recently shown [28] that the minimization of what they call the top-k hinge loss leads to an improvement in top-k error compared to OVA SVM, multiclass SVM, and other ranking-based formulations. In this paper, we study the optimization of the top-k error from a wider perspective. On the one hand, we compare existing OVA schemes and direct multiclass losses in extensive experiments, and on the other, we present theoretical discussion regarding calibration of certain losses for top-k error. Based on these insights, we suggest three new families of loss functions for top-k error. One is a smoothed convex version of the top-k hinge loss of [28], and the other two are convex and nonconvex top-k versions of the softmax loss. We discuss their advantages and disadvantages, and for the convex losses provide an efficient implementation based on stochastic dual coordinate ascent (SDCA) [47].

We evaluate a battery of loss functions on 11 datasets of different tasks ranging from text classification to large scale vision benchmarks, including fine-grained and scene classification. We systematically optimize and report results separately for each top-k accuracy. One interesting message that we would like to highlight is that the softmax loss is able to optimize *all top-k error measures simultaneously*. This is in contrast to multiclass SVM and is also reflected in our experiments. Finally, we show that our new top-k variants of (smooth) multiclass SVM and the softmax loss can further improve top-k performance for a specific k.

Related work. Top-k optimization has recently received revived attention with the advent of large scale problems [21, 28, 30, 31]. The top-k error in multiclass classification, which promotes good ranking of class *labels* for each example, is closely related to the precision@k metric in information retrieval, which counts the fraction of positive instances among the top-k ranked *examples*. The classic approaches optimize pairwise ranking with $\mathrm{SVM}^{\mathrm{struct}}$ [25, 52], $\mathrm{RankNet}$ [12], or LaRank [8].

An alternative direction was proposed by Usunier *et al.* [53], who described a general family of convex loss functions for ranking and classification. One of the loss functions that we consider (top-k SVM_β of [28]) also falls into that family. Weston *et al.* [58] then introduced Wsabie, which optimizes an approximation of a ranking-based loss from [53]. A Bayesian approach was suggested by [50].

Recent works focus more on the top of the ranked list [45, 1, 39, 10], the scalability to large datasets [21, 28, 30], and explore the setting of transductive learning [31].

Contributions. We study the problem of top-k error optimization on a diverse range of learning tasks. We consider existing methods (OVA/direct multiclass) as well as propose three novel loss functions for minimizing the top-k error and compare them both theoretically and empirically. We discover that the softmax loss and the proposed smooth top-1 SVM are astonishingly competitive in all top-k errors. Further small improvements can be obtained by our new top-k variants. Finally, we propose a novel optimization scheme based on SDCA for training the softmax loss.

2. Loss Functions for the Top-k Error

We consider multiclass problems with m classes where the training set $(x_i, y_i)_{i=1}^n$ consists of n examples $x_i \in \mathbb{R}^d$ along with the corresponding labels $y_i \in \mathcal{Y} \triangleq \{1, \ldots, m\}$. We use the notation $p_x(j) \triangleq \Pr(Y = j \,|\, X = x)$. For a $g \in \mathbb{R}^m$, let $g_{[i]}$ denote the i-th largest component, *i.e.*

$$g_{[1]} \geq g_{[2]} \geq \ldots \geq g_{[m]}.$$

We also use a slightly more verbose notation g_{τ_i} and g_{π_i} for the same purpose when the permutation of indexes is transferred across two vectors or two permutations need to be compared. While we specialize later to linear predictors, all loss functions below are formulated for the general setting, where one learns a function $f : \mathbb{R}^d \to \mathbb{R}^m$, and prediction at test time is done via $\arg\max_{j=1,\ldots,m} f_j(x)$. For the linear case, all predictors f_j have the form $f_j(x) = \langle w_j, x \rangle$.

Given that the number of classes m is large, it is likely that the decision between two or more classes is ambiguous. In this case it is natural to use the top-k error instead of the top-1 error, which means the classifier has k guesses and only suffers a loss if all the k guesses are wrong.

Formally, we define the top-k multiclass loss as

$$L_k(y, f(x)) = \mathbb{1}_{f_{[k]}(x) > f_y}, \tag{1}$$

where $\mathbb{1}_P = 1$ if P is true and 0 otherwise. Note that for $k = 1$ we recover the standard multiclass error measure.

Proposition 1. *The conditional Bayes optimal top-k error at x is:* $\min_{g \in \mathbb{R}^m} \mathbb{E}[L_k(y, g) | X = x] = 1 - \sum_{l=1}^k p_x(\tau_l)$, *where the ordering τ is such that $p_x(\tau_1) \geq p_x(\tau_2) \geq \ldots \geq p_x(\tau_m)$. Any f^* is Bayes optimal for the top-k error if $f^*_{\tau_i}(x) > f^*_{\tau_j}(x)$, for all $i = 1, \ldots, k$, $j = k+1, \ldots, m$.*

Proof. Let $g \in \mathbb{R}^m$ and π be an ordering such that $g_{\pi_1} \geq g_{\pi_2} \geq \ldots \geq g_{\pi_m}$. The conditional expected top-k error at x is given by

$$\begin{aligned} \mathbb{E}[L_k(y, g) \,|\, X = x] &= \sum_{l=1}^m p_x(l) \mathbb{1}_{g_{\pi_k} > g_l} \\ &= \sum_{l=1}^m p_x(\pi_l) \mathbb{1}_{g_{\pi_k} > g_{\pi_l}} = \sum_{l=k+1}^m p_x(\pi_l) \\ &= 1 - \sum_{l=1}^k p_x(\pi_l). \end{aligned}$$

The conditional expected top-k error is minimized if the k largest scoring components $g_{\pi_1}, \ldots, g_{\pi_k}$ correspond to the k largest conditional probabilities

$$\Pr(Y = \tau_1 \,|\, X = x), \ldots, \Pr(Y = \tau_k \,|\, X = x).$$

Thus, any classifier $f^*(x)$, for which the two sets

$$\{\pi_1, \ldots, \pi_k\} \text{ and } \{\tau_1, \ldots, \tau_k\}$$

coincide, is Bayes optimal. The optimal top-k error is then

$$\mathbb{E}[\min_{g \in \mathbb{R}^m} \mathbb{E}[L_k(y, g) | X = x]] = \mathbb{E}[1 - \sum_{l=1}^k \Pr(Y = \tau_l \,|\, X)].$$

□

The use of the top-k error in a learning algorithm leads to hard combinatorial problems. In classification, the standard trick is to replace the 0-1-loss by a loss function which is convex and upper bounds the 0-1-loss. Then, under mild conditions on the loss [3, 51], one can prove that the optimal classifier with respect to the convex surrogate yields the Bayes optimal solution for the 0-1-loss. Such loss is called *classification calibrated*, which is known as a necessary condition in statistical learning theory for a classifier to be universally Bayes consistent [3]. We introduce now the notion of calibration for the top-k error.

Definition 1. *A loss function $L : \mathcal{Y} \times \mathbb{R}^m \to \mathbb{R}$ (or a reduction scheme) is called **top-k calibrated** if for all possible data generating measures on $\mathbb{R}^d \times \mathcal{Y}$ and all $x \in \mathbb{R}^d$,*

$$\arg\min_{g \in \mathbb{R}^m} \mathbb{E}[L(y, g) | X = x] \in \arg\min_{g \in \mathbb{R}^m} \mathbb{E}[L_k(y, g) | X = x].$$

If a loss is *not* top-k calibrated, it implies that even in the limit of infinite data, one does not obtain a classifier with the Bayes optimal top-k error from Proposition 1. It is thus an important property, as a loss which fails to be top-k calibrated is optimizing a different criterion than we are actually interested in.

All methods that we consider in this paper optimize regularized empirical loss given by the following objective

$$\frac{1}{n} \sum_{i=1}^n L(y_i, W x_i) + \lambda \left\| W \right\|_F^2,$$

where $W \in \mathbb{R}^{m \times d}$ is the stacked weight matrix.

A brief overview of all the methods that we consider in this paper is given in Table 1.

Method	Equivalent to	Name	Primal loss function	Details
SVM$^{\text{OVA}}$		OVA SVM hinge	$\max\{0,\ 1 - yf(x)\}$	§ 2.1, Lemma 1
LR$^{\text{OVA}}$		OVA logistic regression	$\log(1 + e^{-yf(x)})$	
SVM$^{\text{Multi}}$	top-1 SVM	multiclass SVM	$\max_j \left\{f_j(x) - f_y(x) + \mathbb{1}_{j \neq y}\right\}$	§ 2.1, Lemma 2
LR$^{\text{Multi}}$	top-1 Ent	softmax	$\log \left(\sum_{j=1}^m e^{f_j(x) - f_y(x)}\right)$	§ 2.1, Proposition 3
top-k SVM	top-k SVM$_\alpha$	top-k hinge (α)	$\max \left\{0, \frac{1}{k} \sum_{j=1}^k (a+c)_{[j]}\right\}$	§ 2.2
top-k SVM$^\gamma$	top-k SVM$_\alpha^\gamma$	smooth top-k hinge (α)	$\frac{1}{\gamma}\left(\langle a+c, p^\alpha\rangle - \frac{1}{2}\langle p^\alpha, p^\alpha\rangle\right)$	§ 2.2, Proposition 6
top-k SVM$_\beta$		top-k hinge (β)	$\frac{1}{k} \sum_{j=1}^k \max \left\{0, (a+c)_{[j]}\right\}$	§ 2.2
top-k SVM$_\beta^\gamma$		smooth top-k hinge (β)	$\frac{1}{\gamma}\left(\langle a+c, p^\beta\rangle - \frac{1}{2}\langle p^\beta, p^\beta\rangle\right)$	
top-k Ent		top-k entropy	$\max_{\substack{s, x \in \Delta_k \\ \langle \mathbf{1}, x\rangle = s}} \langle a^{\setminus y}, x\rangle - \langle x, \log x\rangle - (1-s)\log(1-s)$	§ 2.3, Proposition 8
top-k LR$_{\text{n}}$		truncated top-k softmax	$\log \left(1 + \sum_{\substack{j=k \\ [j] \neq y}}^m e^{f_{[j]}(x) - f_y(x)}\right)$	§ 2.4, Proposition 9

Table 1: Overview of the methods considered in the paper. We let $c \triangleq \mathbf{1} - e_y$, $a_j \triangleq f_j(x) - f_y(x)$, $p^\alpha \triangleq \operatorname{proj}_{\Delta_k(\gamma)}(a+c)$, $p^\beta \triangleq \operatorname{proj}_{\tilde{\Delta}_k(\gamma)}(a+c)$, $\Delta_k(r) \triangleq \left\{x \mid \langle \mathbf{1}, x\rangle \leq r,\ 0 \leq x_i \leq \frac{\langle \mathbf{1}, x\rangle}{k}\right\}$, $\tilde{\Delta}_k(r) \triangleq \left\{x \mid \langle \mathbf{1}, x\rangle \leq r,\ 0 \leq x_i \leq \frac{r}{k}\right\}$.

2.1. OVA and Direct Multiclass Approaches

The standard multiclass problem is often solved using the one-vs-all (OVA) reduction into a set of m binary classification problems. Every class $(+1)$ is trained versus the rest (-1), which yields m classifiers $\{f_j\}_{j=1}^m$, and prediction is done via $f(x) = \arg\max_{j=1,\ldots,m} f_j(x)$.

Typically, the binary classification problem is solved via a convex margin-based loss function, that is $L(y, f(x)) = L(yf(x))$ with $L : \mathbb{R} \to \mathbb{R}$. We consider in this paper:

$$\text{SVM}^{\text{OVA}}\text{: hinge loss, } L(yf(x)) = \max\{0,\ 1 - yf(x)\},$$
$$\text{LR}^{\text{OVA}}\text{: logistic loss, } L(yf(x)) = \log(1 + e^{-yf(x)}).$$

The hinge and logistic loss correspond to the SVM and logistic regression respectively. We now show when the OVA schemes are top-k calibrated, not only for $k = 1$ (standard multiclass loss) but for all k simultaneously.

Proposition 2. *The OVA reduction is top-k calibrated for any $1 \leq k \leq m$ if the Bayes optimal function of the convex margin-based loss $L(yf(x))$ is a strictly monotonically increasing function of* $\Pr(Y = 1 \mid X = x)$.

Proof. With the given choice of the labels for any binary problem (*e.g.* class k $(+1)$ versus the rest (-1)), we get that the corresponding Bayes optimal classifiers for the binary problems have the form

$$f_k(x) = g\big(\Pr(Y = k \mid X = x)\big), \quad k = 1, \ldots, m,$$

where g is a strictly monotonically increasing function. Thus, the ranking of f_k corresponds to the ranking of $\Pr(Y = k \mid X = x)$ and hence the OVA reduction is top-k calibrated for any $k = 1, \ldots, m$. □

Next, we provide the Bayes optimal functions for the hinge and logistic losses here for completeness.

Lemma 1. *The Bayes optimal functions of the binary (one-vs-all) hinge loss and logistic loss are respectively:*

$$f^*(x) = -1 + 2\mathbb{1}_{\Pr(Y=1 \mid X=x) > \frac{1}{2}}, \qquad \textit{(hinge loss)},$$
$$f^*(x) = \log\left(\frac{\Pr(Y = 1 \mid X = x)}{1 - \Pr(Y = 1 \mid X = x)}\right), \quad \textit{(logistic loss)}.$$

Proof. **The hinge loss.** The hinge loss is given as

$$L(y, f(x)) = \max\{1 - y\, f(x), 0\}.$$

Using the tower property we can decompose the expected loss as

$$\mathbb{E}[L(Y, f(X))] = \mathbb{E}[\mathbb{E}[L(Y, f(X)) \mid X]].$$

Thus one can compute the Bayes optimal classifier f^* pointwise by minimizing for each $x \in \mathbb{R}^d$,

$$\arg\min_{\alpha \in \mathbb{R}} \mathbb{E}[L(Y, \alpha) \mid X = x],$$

which leads to the following problem:

$$\arg\min_{\alpha \in \mathbb{R}}\ \max\{1 - \alpha, 0\} p_x(1) + \max\{1 + \alpha, 0\} p_x(-1),$$

where $p_x(y) \triangleq \Pr(Y = y \mid X = x)$. It is obvious that the optimal α^* has to be contained in $[-1, 1]$. We get

$$\arg\min_{-1 \leq \alpha \leq 1}\ (1 - \alpha) p_x(1) + (1 + \alpha) p_x(-1).$$

The minimum of an affine function is attained at the boundary and we get

$$f^*(x) = \begin{cases} 1 & \text{if } p_x(1) > \frac{1}{2}, \\ -1 & \text{if } p_x(1) \le \frac{1}{2}. \end{cases}$$

Therefore, the Bayes optimal classifier of the hinge loss is not a strictly monotonically increasing function of $p_x(1)$.

The logistic loss. The logistic loss is given as

$$L(y, f(x)) = \log\big(1 + \exp(-y f(x))\big).$$

Using the tower property we can decompose the expected loss as

$$\mathbb{E}[L(Y, f(X))] = \mathbb{E}[\mathbb{E}[L(Y, f(X)) \,|\, X]].$$

Thus one can compute the Bayes optimal classifier f^* pointwise by minimizing for each $x \in \mathbb{R}^d$,

$$\arg\min_{\alpha \in \mathbb{R}} \mathbb{E}[L(Y, \alpha) \,|\, X = x],$$

which leads to the following problem:

$$\arg\min_{\alpha \in \mathbb{R}} \; \log(1+\exp(-\alpha))p_x(1) + \log(1+\exp(\alpha))p_x(-1).$$

The logistic loss is known to be convex and differentiable and thus the optimum can be computed via

$$\frac{-\exp(-\alpha)}{1+\exp(-\alpha)} p_x(1) + \frac{\exp(\alpha)}{1+\exp(\alpha)} p_x(-1) = 0.$$

Re-writing the first fraction with $\exp(\alpha)$ in enumerator and denominator, we get

$$\frac{-1}{1+\exp(\alpha)} p_x(1) + \frac{\exp(\alpha)}{1+\exp(\alpha)} p_x(-1) = 0,$$

which can be solved as

$$\alpha^* = \log\Big(\frac{p_x(1)}{p_x(-1)}\Big).$$

The Bayes optimal classifier for the logistic loss can thus be written as

$$f^*(x) = \log\Big(\frac{p_x(1)}{1 - p_x(1)}\Big).$$

We check now that the function $\phi : (0,1) \to \infty$ defined as $\phi(x) = \log(\frac{x}{1-x})$ is strictly monotonically increasing. The derivative is given as

$$\begin{aligned} \phi'(x) &= \frac{1-x}{x}\big(\frac{1}{1-x} + \frac{x}{(1-x)^2}\big) \\ &= \frac{1-x}{x}\frac{1}{(1-x)^2} = \frac{1}{x(1-x)} > 0, \quad \forall x \in (0,1). \end{aligned}$$

The derivative is strictly positive on $(0,1)$, which implies that ϕ is strictly monotonically increasing. □

Thus, the logistic loss fulfills the conditions of Proposition 2, whereas the hinge loss does not. Therefore, OVA logistic regression is top-k calibrated for any $1 \le k \le m$, and the OVA SVM with the nonsmooth hinge loss is not.

The alternative to OVA is to use a multiclass loss $L : \mathcal{Y} \times \mathbb{R}^m \to [0, \infty)$. We consider the two direct generalizations of the hinge and the logistic losses:

$\text{SVM}^{\text{Multi}}$: multiclass hinge loss of Crammer/Singer [16]

$$L(y, f(x)) = \max_{j=1,\dots,m} \big\{ f_j(x) - f_y(x) + \mathbb{1}_{j \neq y} \big\}, \tag{2}$$

LR^{Multi}: softmax loss [7]

$$L(y, f(x)) = \log\Big(\sum_{j=1}^m \exp(f_j(x) - f_y(x))\Big). \tag{3}$$

Both are popular losses for multiclass problems. The softmax loss (also known as cross-entropy or multiclass logistic loss) is used often in the end-to-end training as the last layer of neural networks [6, 26, 49]. The multiclass hinge loss has also been shown to be competitive in large-scale image classification [2]. However, it is known that it is not multiclass calibrated [51]. In the following, we provide explicitly the Bayes optimal function of this loss which allows to discuss how "severe" its deficiency is.

Lemma 2. *Let $r^* \in \arg\max_{r=1,\dots,m} p_x(r)$ and fix any $c \in \mathbb{R}$. The Bayes optimal function $f^* : \mathbb{R}^d \to \mathbb{R}^m$ of the multiclass hinge loss (2) is*

$$\begin{aligned} f^*_{r^*}(x) &= c + \begin{cases} 1 & \textit{if } \max_{j=1,\dots,m} p_x(j) \ge \frac{1}{2}, \\ 0 & \textit{else}, \end{cases} \\ f^*_r(x) &= c, \quad r \in \{1,\dots,m\} \setminus \{r^*\}. \end{aligned}$$

Proof. Let $g \in \mathbb{R}^m$ with $g = f(x)$, then

$$\begin{aligned} &\mathbb{E}[L(y, g) \,|\, X = x] = \\ &\sum_{l=1}^m \Pr(Y = l \,|\, X = x) \max_{r=1,\dots,m} \big(g_r - g_l + \mathbb{1}_{r \neq l}\big). \end{aligned}$$

Suppose that the maximum of $(g_r)_{r=1}^m$ is not unique. In this case, one has

$$\max_{r=1,\dots,m} \big(g_r - g_l + \mathbb{1}_{r \neq l}\big) \ge 1,$$

as the term $\mathbb{1}_{r \neq l}$ is always active. Thus the best possible loss is obtained by setting $g_r = c$ for all $r = 1,\dots,m$, which yields an expected loss of 1. On the other hand, if the maximum of $(g_r)_{r=1}^m$ is unique and is achieved by r^*, then

$$\max_{r=1,\dots,m} \big(g_r - g_l + \mathbb{1}_{r \neq l}\big)$$

$$= \begin{cases} g_{r^*} - g_l + 1 & \text{if } l \neq r^*, \\ \max\{0, \max_{r \neq r^*} g_r - g_{r^*} + 1\} & \text{if } l = r^*. \end{cases}$$

As the loss only depends on the gap $g_{r^*} - g_l$ for $l \neq r^*$, we can optimize this with $\beta_l = g_{r^*} - g_l, l \neq r^*$.

$$\begin{aligned} &\mathbb{E}[L(y,g) \,|\, X = x] \\ &\quad= \sum_{l \neq r^*} \Pr(Y = l \,|\, X = x)(g_{r^*} - g_l + 1) \\ &\quad+ \Pr(Y = r^* \,|\, X = x) \max\{0, \max_{l \neq r^*} g_l - g_{r^*} + 1\} \\ &\quad= \sum_{l \neq r^*} \Pr(Y = l \,|\, X = x)(\beta_l + 1) \\ &\quad+ \Pr(Y = r^* \,|\, X = x) \max\{0, \max_{l \neq r^*}(-\beta_l) + 1\} \\ &\quad= \sum_{l \neq r^*} \Pr(Y = l \,|\, X = x)(\beta_l + 1) \\ &\quad+ \Pr(Y = r^* \,|\, X = x) \max\{0, 1 - \min_{l \neq r^*} \beta_l\}. \end{aligned}$$

As only the minimal β_l enters the last term, the optimum is achieved if all β_l are equal for $l \neq r^*$ (otherwise it is possible to reduce the first term without affecting the last term). Let $\alpha \triangleq \beta_l$ for all $l \neq r^*$. The problem becomes

$$\begin{aligned} &\min_{\alpha \geq 0} \sum_{l \neq r^*} \Pr(Y = l \,|\, X = x)(1 + \alpha) \\ &\qquad + \Pr(Y = r^* \,|\, X = x) \max\{0, 1 - \alpha\} \\ &\equiv \min_{0 \leq \alpha \leq 1} \alpha\big(1 - 2\Pr(Y = r^* \,|\, X = x)\big) \end{aligned}$$

Let $p \triangleq \Pr(Y = r^* \,|\, X = x)$. The solution is

$$\alpha^* = \begin{cases} 0 & \text{if } p < \frac{1}{2}, \\ 1 & \text{if } p \geq \frac{1}{2}. \end{cases}$$

The associated risk is

$$\mathbb{E}[L(y,g) \,|\, X = x] = \begin{cases} 1 & \text{if } p < \frac{1}{2}, \\ 2(1-p) & \text{if } p \geq \frac{1}{2}. \end{cases}$$

To construct the Bayes optimal classifier, we let

$$\begin{aligned} r^* &\triangleq \operatorname*{arg\,max}_{r=1,\ldots,m} \Pr(Y = r \,|\, X = x), \\ p &\triangleq \Pr(Y = r^* \,|\, X = x). \end{aligned}$$

If $p < \frac{1}{2}$, then $g_l^* = c$ for $l = 1, \ldots, m$ and any $c \in \mathbb{R}$. Otherwise, $p \geq \frac{1}{2}$ and

$$g_l^* = \begin{cases} c + 1 & \text{if } l = r^*, \\ c & \text{if } l \neq r^*, l = 1, \ldots m. \end{cases}$$

Moreover, $\mathbb{E}[L(y,g^*) \,|\, X = x] = \min\{1, 2(1-p)\} \leq 1$. □

We provided explicitly the Bayes optimal classifier of the multiclass hinge loss. One can deduce that it is not classification calibrated at any $x \in \mathbb{R}^d$, where $\max_j p_x(j) < \frac{1}{2}$. Moreover, it is not top-k calibrated for $k \geq 2$ as the optimal function is constant for all classes except one, which means the ordering of classifier scores need not be the same as that of the conditional class probabilities.

For the softmax loss, it is a folklore result that it is multiclass calibrated even though we could not find a reference for it. We now generalize this results to top-k calibration.

Proposition 3. *The softmax loss is top-k calibrated for any $1 \leq k \leq m$.*

Proof. The multiclass logistic loss is calibrated for the 0-1 multiclass loss in the following sense. If

$$f^*(x) = \operatorname*{arg\,min}_{g \in \mathbb{R}^m} \mathbb{E}[L(Y,g) \,|\, X = x],$$

then for $r = 1, \ldots, m$ and some $\alpha > 0$

$$f_r^*(x) = \begin{cases} \log\big(\alpha\, p_x(r)\big), & \text{if } p_x(r) > 0, \\ -\infty & \text{if } p_x(r) = 0, \end{cases}$$

where $p_x(r) \triangleq \Pr(Y = r \,|\, X = x)$, which then implies

$$\operatorname*{arg\,max}_{r=1,\ldots,m} f_r^*(x) = \operatorname*{arg\,max}_{r=1,\ldots,m} \Pr(Y = r \,|\, X = x).$$

We now prove this result and show that it also generalizes to top-k calibration for $k > 1$. Using the identity

$$L(r,g) = \log\big(\sum_{j=1}^m e^{g_j - g_r}\big) = -g_r + \log\big(\sum_{j=1}^m e^{g_j}\big)$$

and the fact that $\sum_{r=1}^m p_x(r) = 1$, we write for a $g \in \mathbb{R}^m$

$$\begin{aligned} \mathbb{E}[L(Y,g) \,|\, X = x] &= \sum_{r=1}^m p_x(r) L(r,g) \\ &= -\sum_{r=1}^m p_x(r) g_r + \log\big(\sum_{r=1}^m e^{g_r}\big). \end{aligned}$$

As the loss is convex and differentiable, we get the global optimum by computing a critical point. We have

$$\frac{\partial}{\partial g_s} \mathbb{E}[L(Y,g) \,|\, X = x] = -p_x(s) + \frac{e^{g_s}}{\sum_{k=1}^m e^{g_k}} = 0$$

for $s = 1, \ldots, m$. We note that the critical point is not unique as multiplication $g \to \kappa g$ leaves the equation invariant for any $\kappa > 0$. One can verify that $e^{g_s} = \alpha p_x(s)$ satisfies the equations for any $\alpha > 0$. This yields a solution

$$f_r^*(x) = \begin{cases} \log(\alpha p_x(r)) & \text{if } p_x(r) > 0, \\ -\infty & \text{if } p_x(r) = 0, \end{cases}$$

for any fixed $\alpha > 0$. We note that f_r^* is a strictly monotonically increasing function of the conditional class probabilities. Therefore, it preserves the ranking of $\Pr(Y = r \,|\, X = x)$ and implies that f^* is top-k calibrated for any $k \geq 1$. □

The implicit reason for top-k calibration of the OVA schemes and softmax loss is that one can estimate the conditional probabilities $p_x(j)$ from their Bayes optimal classifier. Loss functions which allow this are called *proper*. We refer to [41] and references therein for a detailed discussion.

At this point, one could ask why one should be interested in defining new loss functions for the top-k error given that both, OVA (for certain losses) and softmax, are top-k calibrated for *any* k. The reason is that calibration is an *asymptotic* property as the Bayes optimal functions are obtained by *pointwise* minimization of the expected loss. The picture changes if we use linear functions as classifiers, since they obviously cannot be minimized independently at each point. Indeed, most of the Bayes optimal functions of the different losses cannot be realized by linear functions.

In particular, a problematic property of the softmax and multiclass hinge losses (implicitly also of the OVA schemes) is the fact that even if the classifier $f(x)$ at a point x has *zero* top-k error, it can still be heavily penalized, *e.g.* if $f_{[1]}(x) \gg f_y(x) \geq f_{[k]}(x)$. This leads to a bias, in particular if one is working with "rigid" function classes such as linear ones. The loss functions, which we introduce in the following, are modifications of the above multiclass losses. The goal is to modify the losses such that in the case where the top-k error is zero, the loss should be smaller than the original multiclass loss.

2.2. Smooth Top-k Hinge Loss

Recently, a top-k version of the multiclass hinge loss (2) has been introduced [28]. We use their notation for direct comparison. Let $c = \mathbf{1} - e_j$, where $\mathbf{1}$ is the all ones vector and e_j is the j-th basis vector in $\mathbb{R}^m$, and $a \in \mathbb{R}^m$ be defined componentwise as $a_j \triangleq \langle w_j, x\rangle - \langle w_y, x\rangle$. The loss is

top-k SVM$_\alpha$: top-k hinge loss α [28]

$$L_k(a) = \max\Big\{0, \frac{1}{k}\sum_{j=1}^{k} (a + c)_{[j]}\Big\}. \tag{4}$$

They also propose a second top-k hinge loss based on the family of ranking losses proposed by [53] and show that the above top-k hinge loss is a tighter upper bound on the top-k error. Both versions performed similarly in the experiments, see § 5. The second loss is given as

top-k SVM$_\beta$: top-k hinge loss β [28]

$$L_k(a) = \frac{1}{k}\sum_{j=1}^{k} \max\big\{0, (a + c)_{[j]}\big\}.$$

The top-k hinge loss (either version) reduces to the multiclass hinge loss in (2) for $k = 1$. Thus, it is unlikely that it is top-k calibrated, even though we can currently neither prove nor disprove this for $k > 1$. The multiclass hinge loss fails to be calibrated as the loss is non-smooth and thus does not allow to estimate the class conditional probabilities. Our new family of smooth top-k hinge losses smoothes the top-k loss using the Moreau-Yosida regularization [34, 5]. This technique has been used in [47] to smooth the binary hinge loss. Interestingly, one can show that the smooth binary hinge loss fulfills the conditions of Proposition 2, see Proposition 4 below. Therefore, the hope is that the smoothed top-k hinge loss becomes top-k calibrated as well.

The smoothing works by adding a quadratic term to the conjugate function[1] which then becomes strongly convex. A classical result in convex analysis [22] states that a strongly convex function of modulus c has a conjugate function with Lipschitz smooth gradient with Lipschitz constant $\frac{1}{c}$. Smoothness of the loss typically leads to much faster optimization as we discuss in Section 3.

Proposition 4. *The OVA SVM with the smooth hinge loss is top-k calibrated for any $1 \leq k \leq m$.*

Proof. In order to derive the *smooth* hinge loss, we first compute the conjugate of the standard binary hinge loss,

$$\begin{aligned} L(\alpha) &= \max\{0, 1 - \alpha\}, \\ L^*(\beta) &= \sup_{\alpha \in \mathbb{R}} \{\alpha\beta - \max\{0, 1 - \alpha\}\} \\ &= \begin{cases} \beta & \text{if } -1 \leq \beta \leq 0, \\ \infty & \text{otherwise.} \end{cases} \end{aligned}$$

The smoothed conjugate is

$$L_\gamma^*(\beta) = L^*(\beta) + \frac{\gamma}{2}\beta^2.$$

The corresponding smooth primal hinge loss is given by

$$\begin{aligned} L_\gamma(\alpha) &= \sup_{-1 \leq \beta \leq 0} \{\alpha\beta - \beta - \tfrac{\gamma}{2}\beta^2\} \\ &= \begin{cases} 0, & \text{if } \alpha > 1, \\ \frac{(\alpha-1)^2}{2\gamma} & \text{if } 1 - \gamma \leq \alpha \leq 1, \\ 1 - \alpha - \frac{\gamma}{2} & \text{if } \alpha < 1 - \gamma. \end{cases} \end{aligned}$$

$L_\gamma(\alpha)$ is convex and differentiable with the derivative

$$L_\gamma'(\alpha) = \begin{cases} 0, & \text{if } \alpha > 1, \\ \frac{\alpha-1}{\gamma} & \text{if } 1 - \gamma \leq \alpha \leq 1, \\ -1 & \text{if } \alpha < 1 - \gamma. \end{cases}$$

Thus we can compute the Bayes optimal function via

$$\arg\min_{\alpha \in \mathbb{R}} \; p_x(1)L'(\alpha) - p_x(-1)L'(-\alpha).$$

[1] The **convex conjugate** of f is $f^*(x^*) = \sup\{\langle x^*, x\rangle - f(x)\}$.

Case $0 \leq \gamma \leq 1$**.** We do a case distinction and get for $1-\gamma \leq \alpha \leq 1$ and $0 \leq \gamma \leq 1$ with $p \triangleq p_x(1)$,

$$\begin{aligned} &p\frac{\alpha-1}{\gamma} + (1-p) = 0 \quad \Longrightarrow \\ &\alpha^* = 1 + \gamma\frac{p-1}{p} = \frac{p+\gamma(p-1)}{p}. \end{aligned}$$

If $p \geq \frac{1}{2}$ this solution fulfills $\alpha^* \geq 1-\gamma$. In the case where $\gamma - 1 \leq \alpha \leq 1-\gamma$ and $0 \leq \gamma \leq 1$ we get

$$-p + (1-p) = 1-2p \neq 0 \text{ if } p \neq \frac{1}{2},$$

and the last case is $-1 \leq \alpha \leq \gamma - 1 \leq 0$. Then

$$-p - (1-p)\frac{-\alpha-1}{\gamma} = 0 \quad \Longrightarrow \quad \alpha^* = -1 + \gamma\frac{p}{1-p},$$

where we have $-1 \leq \alpha^* \leq \gamma - 1$ if $p \leq \frac{1}{2}$. Thus we get the Bayes optimal solution for $0 \leq \gamma \leq 1$,

$$f^*(x) = \begin{cases} 1 + \gamma\frac{p_x(1)-1}{p_x(1)} & \text{if } p_x(1) \geq \frac{1}{2}, \\ -1 + \gamma\frac{p_x(1)}{1-p_x(1)} & \text{if } p_x(1) < \frac{1}{2}. \end{cases}$$

Note that while $f^*(x)$ is not a continuous function of $p_x(1)$ for $\gamma < 1$, it is still a strictly monotonically increasing function of $p_x(1)$ for any $0 < \gamma \leq 1$.

Case $\gamma > 1$**.** In the case when $\gamma > 1$, we have to distinguish the cases where $\gamma - 1 \leq \alpha \leq 1$, we get

$$\begin{aligned} &p\frac{\alpha-1}{\gamma} + (1-p) = 0 \quad \Longrightarrow \\ &\alpha^* = 1 + \gamma\frac{p-1}{p} = \frac{p+\gamma(p-1)}{p}. \end{aligned}$$

In order that $\alpha^* \geq \gamma - 1$ we get the condition $p \geq \frac{\gamma}{2}$. Then we have the case where $1 - \gamma \leq \alpha \leq \gamma - 1$, we get

$$p\frac{\alpha-1}{\gamma} - (1-p)\frac{-\alpha-1}{\gamma} = 0 \quad \Longrightarrow \quad \alpha^* = 2p-1,$$

which is in the range $1-\gamma \leq 2p-1 \leq \gamma-1$ if $1 - \frac{\gamma}{2} \leq p \leq \frac{\gamma}{2}$. Finally, we have the case where $-1 \leq \alpha \leq 1-\gamma$. In this case, we get

$$-p - (1-p)\frac{-\alpha-1}{\gamma} = 0 \quad \Longrightarrow \quad \alpha^* = -1 + \gamma\frac{p}{1-p},$$

where we have $-1 \leq \alpha^* \leq 1 - \gamma$ if $p \leq 1 - \frac{\gamma}{2}$. Overall, the Bayes optimal solution for $\gamma > 1$ is

$$f^*(x) = \begin{cases} 1 + \gamma\frac{p_x(1)-1}{p_x(1)} & \text{if } p_x(1) \geq \frac{\gamma}{2}, \\ 2p_x(1) - 1 & \text{if } 1 - \frac{\gamma}{2} \leq p_x(1) \leq \frac{\gamma}{2}, \\ -1 + \gamma\frac{p_x(1)}{1-p_x(1)} & \text{if } p_x(1) < 1 - \frac{\gamma}{2}. \end{cases}$$

Note that f^* is again a strictly monotonically increasing function of $p_x(1)$. Therefore, by Proposition 2, the smooth hinge loss with any $\gamma > 0$ is top-k calibrated for all k. □

We are now ready to introduce the smooth top-k hinge loss, which extends the multiclass top-k SVM loss proposed by [28]. The novel loss is defined via a conjugate, which requires the following set. The *top-k simplex* of radius r is defined as

$$\Delta_k(r) \triangleq \left\{ x \mid \langle \mathbf{1}, x \rangle \leq r,\ 0 \leq x_i \leq \tfrac{1}{k} \langle \mathbf{1}, x \rangle\,,\ \forall i \right\},$$

Proposition 5 ([28])**.** *The convex conjugate of $L_k(a)$ is $L_k^*(b) = -\langle c, b \rangle$, if $b \in \Delta_k(1)$, $+\infty$ otherwise.*

The application of the smoothing technique yields the following *smooth top-k hinge* loss.

Proposition 6. *Let $\gamma > 0$ be the smoothing parameter. The* ***smooth top-k hinge loss*** top-k SVM_α *and its conjugate are*

$$\begin{aligned} L_k^\gamma(a) &= \tfrac{1}{\gamma}\big(\langle a + c, p \rangle - \tfrac{1}{2} \langle p, p \rangle \big), \\ \big(L_k^\gamma\big)^*(b) &= \tfrac{\gamma}{2} \langle b, b \rangle - \langle c, b \rangle\,, \text{ if } b \in \Delta_k(1),\ \infty \text{ otherwise}, \end{aligned}$$

where $p = \operatorname{proj}_{\Delta_k(\gamma)}(a + c)$ is the Euclidean projection of $(a + c)$ on $\Delta_k(\gamma)$. Moreover, $L_k^\gamma(a)$ is $1/\gamma$-smooth.

Proof. We take the convex conjugate of the top-k hinge loss, which was derived in [28, Proposition 2],

$$L_k^*(b) = \begin{cases} -\langle c, b \rangle & \text{if } b \in \Delta_k(1), \\ +\infty & \text{otherwise}, \end{cases}$$

and add the regularizer $\frac{\gamma}{2} \langle b, b \rangle$ to obtain the γ-strongly convex conjugate loss $\big(L_k^\gamma\big)^*(b)$ as stated in the proposition. As mentioned above [22] (see also [47, Lemma 2]), the primal smooth top-k hinge loss $L_k^\gamma(a)$, obtained as the convex conjugate of $\big(L_k^\gamma\big)^*(b)$, is $1/\gamma$-smooth. We now obtain a formula to compute it based on the Euclidean projection onto the top-k simplex. By definition,

$$\begin{aligned} L_k^\gamma(a) &= \sup_{b \in \mathbb{R}^m} \{ \langle a, b \rangle - \big(L_k^\gamma\big)^*(b) \} \\ &= \max_{b \in \Delta_k(1)} \Big\{ \langle a, b \rangle - \frac{\gamma}{2} \langle b, b \rangle + \langle c, b \rangle \Big\} \\ &= - \min_{b \in \Delta_k(1)} \Big\{ \frac{\gamma}{2} \langle b, b \rangle - \langle a + c, b \rangle \Big\} \\ &= - \frac{1}{\gamma} \min_{b \in \Delta_k(1)} \Big\{ \frac{1}{2} \langle \gamma b, \gamma b \rangle - \langle a + c, \gamma b \rangle \Big\} \\ &= - \frac{1}{\gamma} \min_{\frac{b}{\gamma} \in \Delta_k(1)} \Big\{ \frac{1}{2} \langle b, b \rangle - \langle a + c, b \rangle \Big\}. \end{aligned}$$

For the constraint $\frac{b}{\gamma} \in \Delta_k(1)$, we have

$$\begin{aligned} \langle \mathbf{1}, b/\gamma \rangle \leq 1, \quad & 0 \leq b_i/\gamma \leq \frac{1}{k} \langle \mathbf{1}, b/\gamma \rangle \quad && \Longleftrightarrow \\ \langle \mathbf{1}, b \rangle \leq \gamma, \quad & 0 \leq b_i \leq \frac{1}{k} \langle \mathbf{1}, b \rangle \quad && \Longleftrightarrow \\ & b \in \Delta_k(\gamma). \end{aligned}$$

The final expression follows from the fact that

$$\begin{aligned}
&\operatorname*{arg\,min}_{b \in \Delta_k(\gamma)} \left\{ \frac{1}{2} \langle b, b \rangle - \langle a + c, b \rangle \right\} \\
&= \operatorname*{arg\,min}_{b \in \Delta_k(\gamma)} \|(a + c) - b\|^2 = \operatorname{proj}_{\Delta_k(\gamma)}(a + c).
\end{aligned}$$

□

The **smooth top-k hinge loss** top-k $\mathrm{SVM}_\beta^\gamma$ is defined analogously, but the top-k simplex $\Delta_k(r)$ is replaced by

$$\tilde{\Delta}_k(r) \triangleq \{x \,|\, \langle \mathbf{1}, x \rangle \leq r,\ 0 \leq x_i \leq \tfrac{r}{k},\ \forall i\}.$$

There is no analytic expression for these losses. The evaluation requires the projection onto the top-k simplex $\Delta_k(\gamma)$ or the set $\tilde{\Delta}_k(\gamma)$, which can be done in $O(m \log m)$ time as shown in [28]. The non-analytic nature of the losses currently prevents us from proving their top-k calibration.

2.3. Top-k Softmax Loss

We have shown that the softmax loss is top-k calibrated for all k *simultaneously*. As discussed above, this is a strong property, but of asymptotic nature and does not necessarily transfer to the case where only linear classifiers are used. In particular, we show in § 4 on a synthetic dataset that, when limited to linear classifiers, the top-1 and top-2 optimization lead to completely different solutions. The softmax loss, primarily aiming at top-1 performance, produces in that case a solution that is reasonably good in top-1 accuracy, but is far from what can be achieved in top-2 accuracy.

The above reasoning motivated us to search for a top-k version of the softmax loss. Our **top-k entropy loss**, is inspired by the conjugate of the top-k hinge loss. Recall that the conjugate functions of multiclass SVM of [16] and the top-k SVM of [28] differ just in their effective domain[2] Δ_k while the conjugate function is the same. Instead of the standard simplex, the conjugate of the top-k hinge loss is defined on a subset, the top-k simplex.

This suggests a way to *construct novel losses* with specific properties by taking the conjugate of an existing loss function, and shrinking/modifying its essential domain in a way that enforces the desired properties. The motivation for doing so comes from the interpretation of the dual variables as forces with which every training example pushes the decision surface in the direction given by the ground truth label. The absolute value of the dual variables determines the magnitude of these forces and the optimal values are often attained at the boundary of the feasible set (which coincides with the essential domain of the loss). Therefore, by reducing the feasible set we can limit the maximal contribution of a given training example.

As a first step in the construction of the top-k entropy loss, we compute the conjugate of the softmax loss. Let $W \in \mathbb{R}^{d \times m}$ be the matrix obtained by stacking the m weight vectors together and let $a^{\setminus j}$ be obtained by removing the j-th coordinate from vector a.

Proposition 7. *The convex conjugate of* (3) *with* $y = j$ *is*

$$L^*(v) = \begin{cases} \sum_{i \neq j} v_i \log v_i + (1 + v_j) \log(1 + v_j), \\ \qquad \textit{if } \langle \mathbf{1}, v \rangle = 0 \textit{ and } v^{\setminus j} \in \Delta, \\ +\infty \quad \textit{otherwise}, \end{cases} \tag{5}$$

where $\Delta \triangleq \{x \,|\, \langle \mathbf{1}, x \rangle \leq 1,\ 0 \leq x_i \leq 1,\ \forall i\}$.

Proof. We provide a derivation for the convex conjugate of the softmax loss which was already given in [32, Appendix D.2.3] without a proof. We also highlight the constraint $\langle \mathbf{1}, v \rangle = 0$ which can be easily missed when computing the conjugate and is re-stated explicitly in Lemma 3.

Consider a training example $x \in \mathbb{R}^d$ with a label $y \in \{1, \ldots, m\}$ and let $u \triangleq f(x) = W^\top x \in \mathbb{R}^m$, $j \triangleq y$. The softmax loss on example x is given by

$$L(u) = \log \Big(\sum_{i=1}^m \exp(u_i - u_j) \Big) = \log \Big(\sum_{i=1}^m \exp(u'_i) \Big),$$

where we let $u' \triangleq H_j u$ and $H_j \triangleq \mathbf{I} - \mathbf{1} e_j^\top$. Let

$$\phi(u) \triangleq \log \Big(\sum_{i=1}^m \exp(u_i) \Big),$$

then $L(u) = \phi(H_j u)$ and the convex conjugate is computed similar to [28, Lemma 2] as follows.

$$\begin{aligned}
L^*(v) &= \sup\{ \langle u, v \rangle - L(u) \,|\, u \in \mathbb{R}^m \} \\
&= \sup\{ \langle u, v \rangle - \phi(H_j u) \,|\, u \in \mathbb{R}^m \} \\
&= \sup\{ \langle u^\parallel, v \rangle + \langle u^\perp, v \rangle - \phi(H_j u^\perp) \,| \\
&\qquad u = u^\parallel + u^\perp, \\
&\qquad u^\parallel \in \operatorname{Ker} H_j, u^\perp \in \operatorname{Ker}^\perp H_j \},
\end{aligned}$$

where $\operatorname{Ker} H_j = \{u \,|\, H_j u = 0\} = \{t \mathbf{1} \,|\, t \in \mathbb{R}\}$ and $\operatorname{Ker}^\perp H_j = \{u \,|\, \langle \mathbf{1}, u \rangle = 0\}$. It follows that $L^*(v)$ can only be finite if $\langle u^\parallel, v \rangle = 0$, which implies $v \in \operatorname{Ker}^\perp H_j \iff \langle \mathbf{1}, v \rangle = 0$. Let $H_j^\dagger$ be the Moore-Penrose pseudoinverse of H_j. For a $v \in \operatorname{Ker}^\perp H_j$, we write

$$\begin{aligned}
L^*(v) &= \sup\{ \langle H_j^\dagger H_j u^\perp, v \rangle - \phi(H_j u^\perp) \,|\, u^\perp \in \operatorname{Ker}^\perp H_j \} \\
&= \sup\{ \langle z, (H_j^\dagger)^\top v \rangle - \phi(z) \,|\, z \in \operatorname{Im} H_j \},
\end{aligned}$$

where $\operatorname{Im} H_j = \{H_j u \,|\, u \in \mathbb{R}^m\} = \{u \,|\, u_j = 0\}$. Using rank-1 update of the pseudoinverse [37, § 3.2.7], we have

$$(H_j^\dagger)^\top = \mathbf{I} - e_j e_j^\top - \frac{1}{m} (\mathbf{1} - e_j) \mathbf{1}^\top,$$

[2] The **effective domain** of f is $\operatorname{dom} f = \{x \in X \,|\, f(x) < +\infty\}$.

which together with $\langle \mathbf{1}, v \rangle = 0$ implies

$$(H_j^\dagger)^\top v = v - v_j e_j .$$

Therefore,

$$\begin{aligned} L^*(v) &= \sup\{\langle u, v - v_j e_j \rangle - \phi(u) \,|\, u_j = 0\} \\ &= \sup\Big\{\langle u^{\setminus j}, v^{\setminus j} \rangle - \log\Big(1 + \sum_{i \neq j} \exp(u_i)\Big)\Big\}. \end{aligned}$$

The function inside sup is concave and differentiable, hence the global optimum is at the critical point [11]. Setting the partial derivatives to zero yields

$$v_i = \exp(u_i) / \big(1 + \sum_{i \neq j} \exp(u_i)\big)$$

for $i \neq j$, from which we conclude, similar to [47, § 5.1], that $\langle \mathbf{1}, v \rangle \leq 1$ and $0 \leq v_i \leq 1$ for all $i \neq j$, *i.e.* $v^{\setminus j} \in \Delta$. Let $Z \triangleq \sum_{i \neq j} \exp(u_i)$, we have at the optimum

$$u_i = \log(v_i) + \log(1 + Z), \quad \forall i \neq j.$$

Since $\langle \mathbf{1}, v \rangle = 0$, we also have that $v_j = -\sum_{i \neq j} v_i$, hence

$$\begin{aligned} L^*(v) &= \sum_{i \neq j} u_i v_i - \log(1 + Z) \\ &= \sum_{i \neq j} v_i \log(v_i) + \log(1 + Z)\Big(\sum_{i \neq j} v_i - 1\Big) \\ &= \sum_{i \neq j} v_i \log(v_i) - \log(1 + Z)(1 + v_j). \end{aligned}$$

Summing v_i and using the definition of Z,

$$\sum_{i \neq j} v_i = \sum_{i \neq j} \exp(u_i) / \big(1 + \sum_{i \neq j} \exp(u_i)\big) = Z / (1 + Z).$$

Therefore,

$$1 + Z = 1 / \big(1 - \sum_{i \neq j} v_i\big) = 1 / (1 + v_j),$$

which finally yields

$$L^*(v) = \sum_{i \neq j} v_i \log(v_i) + \log(1 + v_j)(1 + v_j),$$

if $\langle \mathbf{1}, v \rangle = 0$ and $v^{\setminus j} \in \Delta$ as stated in the proposition. □

The conjugate of the top-k entropy loss is obtained by replacing Δ with Δ_k in (5). There is no closed-form solution for the primal top-k entropy loss for $k > 1$, but we can evaluate the loss by solving an optimization problem.

Proposition 8. *The **top-k entropy** loss is defined via the following optimization problem with $u \triangleq W^\top x_i - \langle w_{y_i}, x_i \rangle \mathbf{1}$*

$$-\min_{\substack{s, x \in \Delta_k \\ \langle \mathbf{1}, x \rangle = s}} \langle x, \log x \rangle + (1 - s)\log(1 - s) - \langle u^{\setminus j}, x \rangle. \tag{6}$$

Moreover, we recover the softmax loss (3) if $k = 1$.

Proof. The convex conjugate of the top-k entropy loss is

$$L^*(v) \triangleq \begin{cases} \sum_{i \neq j} v_i \log v_i + (1 + v_j)\log(1 + v_j), \\ \qquad \text{if } \langle \mathbf{1}, v \rangle = 0 \text{ and } v^{\setminus j} \in \Delta_k, \\ +\infty \quad \text{otherwise,} \end{cases}$$

where we consider the same setting as in the proof of Proposition 7. The (primal) top-k entropy loss is defined as the convex conjugate of the $L^*(v)$ above. We have

$$\begin{aligned} L(u) &= \sup\{\, \langle u, v \rangle - L^*(v) \,|\, v \in \mathbb{R}^m \} \\ &= \sup\{\, \langle u, v \rangle - \sum_{i \neq j} v_i \log v_i - (1 + v_j)\log(1 + v_j) \\ &\qquad |\, \langle \mathbf{1}, v \rangle = 0,\ v^{\setminus j} \in \Delta_k \} \\ &= \sup\{\langle u^{\setminus j}, v^{\setminus j} \rangle - u_j \sum_{i \neq j} v_i - \sum_{i \neq j} v_i \log v_i \\ &\qquad - (1 - \sum_{i \neq j} v_i)\log(1 - \sum_{i \neq j} v_i) \,|\, v^{\setminus j} \in \Delta_k \}. \end{aligned}$$

We let $u \triangleq (W^\top x_i - \langle w_{y_i}, x_i \rangle \mathbf{1})$ be defined as usual for the multiclass losses, which then yields $u_j = 0$, and hence the corresponding term vanishes. Finally, we let $x \triangleq v^{\setminus j}$ and $s \triangleq \sum_{i \neq j} v_i = \langle \mathbf{1}, x \rangle$ and obtain (6).

Next, we discuss how this problem can be solved and show that it reduces to the softmax loss for $k = 1$.

Let $a \triangleq u^{\setminus j}$, the Lagrangian for (6) is given as

$$\begin{aligned} \mathcal{L} &= \langle x, \log x \rangle + (1 - s)\log(1 - s) - \langle a, x \rangle \\ &\quad + t(\langle \mathbf{1}, x \rangle - s) + \lambda(s - 1) - \langle \mu, x \rangle + \big\langle \nu, x - \tfrac{s}{k}\mathbf{1} \big\rangle, \end{aligned}$$

where $t \in \mathbb{R}$ and $\lambda, \mu, \nu \geq 0$ are the dual variables. Computing partial derivatives of $\mathcal{L}$ w.r.t. x_i and s, and setting them to zero, we obtain

$$\begin{aligned} \log x_i &= a_i - 1 - t + \mu_i - \nu_i, \quad \forall i \\ \log(1 - s) &= -1 - t - \tfrac{1}{k}\langle \mathbf{1}, \nu \rangle + \lambda. \end{aligned}$$

Note that $x_i = 0$ and $s = 1$ cannot satisfy the above conditions for any choice of the dual variables in $\mathbb{R}$. Therefore, $x_i > 0$ and $s < 1$, which implies $\mu_i = 0$ and $\lambda = 0$. The only constraint that might be active is $x_i \leq \frac{s}{k}$. Note, however, that in view of $x_i > 0$ it can only be active if either $k > 1$ or we have a one dimensional problem. We consider the case when this constraint is active below.

Consider x_i's for which $0 < x_i < \frac{s}{k}$ holds at the optimum. The complementary slackness conditions imply that the corresponding $\mu_i = \nu_i = 0$. Let $p \triangleq \langle \mathbf{1}, \nu \rangle$ and redefine t as $t \leftarrow 1 + t$. We obtain the simplified equations

$$\begin{aligned} \log x_i &= a_i - t, \\ \log(1 - s) &= -t - \tfrac{p}{k}. \end{aligned}$$

If $k = 1$, then $0 < x_i < s$ for all i in a multiclass problem as discussed above, hence also $p = 0$. We have

$$\begin{aligned} x_i &= \exp(a_i - t), \\ 1 - s &= \exp(-t), \end{aligned}$$

where $t \in \mathbb{R}$ is to be found. Plugging that into the objective,

$$\begin{aligned} &\sum_i (a_i - t)\exp(a_i - t) - t\exp(-t) - \sum_i a_i \exp(a_i - t) \\ &= \exp(-t)\Big[\sum_i (a_i - t)\exp a_i - t - \sum_i a_i \exp a_i\Big] \\ &= -t\exp(-t)\big[1 + \sum_i \exp a_i\big] \\ &= -t\big[\exp(-t) + \sum_i \exp(a_i - t)\big] \\ &= -t\big[1 - s + s\big] = -t. \end{aligned}$$

To compute t, we note that

$$\sum_i \exp(a_i - t) = \langle \mathbf{1}, x \rangle = s = 1 - \exp(-t),$$

from which we conclude

$$\begin{aligned} 1 &= \big(1 + \sum_i \exp a_i\big)\exp(-t) \implies \\ -t &= -\log\big(1 + \sum_i \exp a_i\big). \end{aligned}$$

Taking into account the minus in front of the $\min$ in (6) and the definition of a, we finally recover the softmax loss

$$L(u) = \log\big(1 + \sum_{y \neq y_i} \exp(\langle w_y, x_i \rangle - \langle w_{y_i}, x_i \rangle)\big).$$

□

The non-analytic nature of the loss for $k > 1$ does not allow us to check if it is top-k calibrated. We now show how this problem can be solved efficiently.

How to solve (6). We continue the derivation started in the proof of Propostion 8. First, we write the system that follows directly from the KKT [11] optimality conditions.

$$\begin{aligned} x_i &= \min\{\exp(a_i - t), \tfrac{s}{k}\}, \quad \forall i, \\ \nu_i &= \max\{0, a_i - t - \log(\tfrac{s}{k})\}, \quad \forall i, \\ s &= \langle \mathbf{1}, x \rangle, \\ p &= \langle \mathbf{1}, \nu \rangle, \\ 1 - s &= \exp(-t - \tfrac{p}{k}). \end{aligned} \tag{7}$$

Next, we define the two index sets U and M as follows

$$U \triangleq \{i \,|\, x_i = \tfrac{s}{k}\}, \qquad M \triangleq \{i \,|\, x_i < \tfrac{s}{k}\}.$$

Note that the set U contains at most k indexes corresponding to the largest components of a_i. Now, we proceed with finding a t that solves (7). Let $\rho \triangleq \frac{|U|}{k}$. We eliminate p as

$$\begin{aligned} p &= \sum_i \nu_i = \sum_U a_i - |U|\big(t + \log(\tfrac{s}{k})\big) \implies \\ \tfrac{p}{k} &= \tfrac{1}{k}\sum_U a_i - \rho\big(t + \log(\tfrac{s}{k})\big). \end{aligned}$$

Let $Z \triangleq \sum_M \exp a_i$, we write for s

$$\begin{aligned} s &= \sum_i x_i = \sum_U \tfrac{s}{k} + \sum_M \exp(a_i - t) \\ &= \rho s + \exp(-t)\sum_M \exp a_i = \rho s + \exp(-t) Z. \end{aligned}$$

We conclude that

$$\begin{aligned} (1 - \rho)s &= \exp(-t) Z \implies \\ t &= \log Z - \log\big((1 - \rho)s\big). \end{aligned}$$

Let $\alpha \triangleq \frac{1}{k}\sum_U a_i$. We further write

$$\begin{aligned} \log(1 - s) &= -t - \tfrac{p}{k} \\ &= -t - \alpha + \rho\big(t + \log(\tfrac{s}{k})\big) \\ &= \rho\log(\tfrac{s}{k}) - (1 - \rho)t - \alpha \\ &= \rho\log(\tfrac{s}{k}) - \alpha \\ &\quad - (1 - \rho)\big[\log Z - \log\big((1 - \rho)s\big)\big], \end{aligned}$$

which yields the following equation for s

$$\begin{aligned} &\log(1 - s) - \rho(\log s - \log k) + \alpha \\ &+ (1 - \rho)\big[\log Z - \log(1 - \rho) - \log s\big] = 0. \end{aligned}$$

Therefore,

$$\begin{aligned} &\log(1 - s) - \log s + \rho\log k + \alpha \\ &+ (1 - \rho)\log Z - (1 - \rho)\log(1 - \rho) = 0, \\ &\log\left(\frac{1 - s}{s}\right) = \log\left(\frac{(1 - \rho)^{(1 - \rho)}\exp(-\alpha)}{k^{\rho} Z^{(1 - \rho)}}\right). \end{aligned}$$

We finally get

$$\begin{aligned} s &= 1/(1 + Q), \text{ where} \\ Q &\triangleq (1 - \rho)^{(1 - \rho)}/(k^{\rho} Z^{(1 - \rho)} e^{\alpha}). \end{aligned}$$

We note that: *a)* Q is readily computable once the sets U and M are fixed; and *b)* $Q = 1/Z$ if $k = 1$ since $\rho = \alpha = 0$ in that case. This yields the formula for t as

$$t = \log Z + \log(1 + Q) - \log(1 - \rho). \tag{8}$$

As a sanity check, we note that we again recover the softmax loss for $k = 1$, since $t = \log Z + \log(1 + 1/Z) = \log(1 + Z) = \log(1 + \sum_i \exp a_i)$.

To verify that the computed s and t are compatible with the choice of the sets U and M, we check if this holds:

$$\begin{aligned} \exp(a_i - t) &\geq \tfrac{s}{k}, \quad \forall i \in U, \\ \exp(a_i - t) &\leq \tfrac{s}{k}, \quad \forall i \in M, \end{aligned}$$

which is equivalent to

$$\max_M a_i \leq \log(\tfrac{s}{k}) + t \leq \min_U a_i. \tag{9}$$

Algorithm to solve (6). The above derivation suggests a simple and efficient algorithm to compute a t that solves the KKT system (7) and, therefore, the original problem (6).

1. Compute $t = \log(1 + Z)$ (U is empty at this point).
2. If (9) holds, then stop.
3. Otherwise, $U \leftarrow U \cup \{\max_M a_i\}$.
4. Compute t as in (8) and go to step 2.

Note that the algorithm terminates after at most k iterations since $|U| \leq k$. The overall complexity is therefore $O(km)$.

To compute the actual loss (6), we note that if U is empty, *i.e.* there were no violated constraints, then the top-k entropy loss coincides with the softmax loss and is directly given by t. Otherwise, we have

$$\begin{aligned} &\langle a, x \rangle - \langle x, \log x \rangle - (1 - s)\log(1 - s) \\ &= \sum_U a_i \tfrac{s}{k} + \sum_M a_i \exp(a_i - t) - \sum_U \tfrac{s}{k} \log(\tfrac{s}{k}) \\ &- \sum_M (a_i - t)\exp(a_i - t) - (1 - s)\log(1 - s) \\ &= \alpha s - \rho s \log(\tfrac{s}{k}) + t \exp(-t) Z - (1 - s)\log(1 - s) \\ &= \alpha s - \rho s \log(\tfrac{s}{k}) + (1 - \rho) s t - (1 - s)\log(1 - s). \end{aligned}$$

Finally, we note that one has to take special care when performing exponentiation in the implementation as it can easily lead to an overflow in floating-point arithmetic.

2.4. Truncated Top-k Softmax Loss

A major limitation of the softmax loss regarding top-k error minimization is that it cannot ignore the highest scoring predictions, which yields a high loss even if top-k error is zero. This can be seen by rewriting softmax loss (3) as

$$L(y, f(x)) = \log\big(1 + \sum_{\substack{j=1 \\ j \neq y}}^{m} \exp(f_j(x) - f_y(x))\big). \tag{10}$$

Suppose we have for a *single* output $f_j \gg f_y$, then $f_j - f_y$ is heavily penalized even though the top-2 error is zero.

This problem is an inherent limitation of convex multiclass losses as it is also present in the smooth/nonsmooth top-k hinge loss. The origin of the problem is the fact that ranking based losses [53] are based on functions such as

$$\phi(f) = \frac{1}{m} \sum_{i=1}^{m} a_i f_{[i]} - f_y.$$

The function ϕ is convex if the sequence (a_i) is monotonically decreasing [11]. This implies that convex ranking based losses have to put *more* weight on the highest scoring classifiers. However, we would like to put *less* weight on them. Thus, the only way to reduce the influence of the highest scoring classifiers is to use a non-convex loss. In the **truncated top-k softmax loss** (top-k $\mathrm{LR_n}$), we simply drop the $(k - 1)$ highest scoring predictions from the sum in (10):

$$L_k(y, f(x)) = \log\big(1 + \sum_{\substack{j=k \\ [j] \neq y}}^{m} \exp(f_{[j]}(x) - f_y(x))\big), \tag{11}$$

This loss can be seen as a smoothed version of the original top-k error in (1) being small whenever the top-k error is zero, which is clearly a desirable property. Similar to the softmax loss, we can prove its top-k calibration.

Proposition 9. *The truncated top-k softmax loss is top-s calibrated for any $k \leq s \leq m$.*

Proof. The expected conditional loss can be written with $f(x) = g \in \mathbb{R}^m$ and $p_x(r) \triangleq \Pr(Y = r \,|\, X = x)$ as

$$\mathbb{E}[L(y, g) \,|\, X = x] = \sum_{r=1}^{m} p_x(r) \log\Big(1 + \sum_{\substack{l=k \\ l \neq r}}^{m} e^{g_{\pi_l} - g_r}\Big),$$

where π corresponds to a sorting of g in descending order, that is $g_{\pi_1} \geq g_{\pi_2} \geq \ldots \geq g_{\pi_m}$. Note that as the sum inside the logarithm starts with the k-th largest prediction g_{π_k}, the Bayes optimal solution will set $g_{\pi_1} = \ldots = g_{\pi_{k-1}} = \infty$ as this leads to the fact that the corresponding terms vanish given that g_{π_k} is finite. Let τ correspond to the ordering of $\big(p_x(l)\big)_{l=1}^{m}$ in descending order, that is

$$p_x(\tau_1) \geq p_x(\tau_2) \geq \ldots \geq p_x(\tau_m).$$

Then the optimal choice for $(\pi_l)_{l=1}^{k}$ is $(\tau_l)_{l=1}^{k}$, the exact ordering plays no role. Finally, we get

$$\begin{aligned} \mathbb{E}[L(y, g) \,|\, X = x] &= \sum_{r=k}^{m} p_x(\tau_r) \log\Big(1 + \sum_{\substack{l=k \\ \pi_l \neq \tau_r}}^{m} e^{g_{\pi_l} - g_{\tau_r}}\Big) \\ &= \sum_{r=k}^{m} p_x(\tau_r) \log\Big(\sum_{l=k}^{m} e^{g_{\pi_l} - g_{\tau_r}}\Big) \end{aligned}$$

$$= -\sum_{r=k}^{m} p_x(\tau_r) g_{\tau_r} + \log\Big(\sum_{l=k}^{m} e^{g_{\pi_l}}\Big)$$
$$= -\sum_{r=k}^{m} p_x(\tau_r) g_{\tau_r} + \log\Big(\sum_{l=k}^{m} e^{g_{\tau_l}}\Big)$$

where the second equality follows as $\tau_r \in (\pi_k, \dots, \pi_m)$ for every $r = k, \dots, m$. Note that the sum inside the logarithm in the third row is over all terms which are not yet fixed and thus the ordering plays no role anymore. Taking the derivative with respect to g_{τ_s} for $s \in (k, \dots, m)$, yields

$$\frac{\partial}{\partial g_{\tau_s}} \mathbb{E}[L(y, g) \,|\, X = x] = -p_x(\tau_s) + \frac{e^{g_{\tau_s}}}{\sum_{l=k}^{m} e^{g_{\tau_l}}}$$

We get as solution

$$g^*_{\tau_s} = \begin{cases} \log\big(\alpha p_x(\tau_s)\big) & \text{if } p_x(\tau_s) > 0, \\ -\infty & \text{otherwise,} \end{cases}$$

for any $\alpha > 0$ (the solution is not uniquely defined). We summarize the final Bayes optimal classifier g^* at x,

$$g^*_{\tau_s} = \begin{cases} +\infty & \text{for } s = 1, \dots, k-1, \\ \log\big(\alpha p_x(\tau_s)\big) & \text{if } s = k, \dots, m \text{ and } p_x(\tau_s) > 0, \\ -\infty & \text{if } s = k, \dots, m \text{ and } p_x(\tau_s) = 0. \end{cases}$$

We note that the Bayes optimal classifier preserves the ranking of the conditional distribution from the k-th term on and thus it is top-s calibrated for any $s \geq k$. □

As the optimization problem is nonconvex, we use the LR$^{\text{Multi}}$ solution as an initial point and minimize the ℓ_2-regularized top-k LR$_{\text{n}}$ loss using gradient descent. However, the optimization problem seems to be mildly "nonconvex" as the same-quality solution is obtained from different initializations (*e.g.*, we tried all zeros and Gaussian noise as initial solutions on toy data). In Section 4, we show a toy data experiment, where the advantage of discarding the highest scoring classifier in the loss becomes apparent.

3. Optimization Method

In this section we briefly discuss how the proposed loss functions top-k SVM$^\gamma$ and top-k Ent can be optimized efficiently within the SDCA framework of [47].

The primal and dual problems. Let $X \in \mathbb{R}^{d \times n}$ be the matrix of training examples $x_i \in \mathbb{R}^d$, $K = X^\top X$ the corresponding Gram matrix, $W \in \mathbb{R}^{d \times m}$ the matrix of primal variables, $A \in \mathbb{R}^{m \times n}$ the matrix of dual variables, and $\lambda > 0$ the regularization parameter. The primal and Fenchel dual [9] objective functions are given as

$$\begin{aligned} P(W) &= +\frac{1}{n} \sum_{i=1}^{n} L_i\left(W^\top x_i\right) + \frac{\lambda}{2} \operatorname{tr}\left(W^\top W\right), \\ D(A) &= -\frac{1}{n} \sum_{i=1}^{n} L_i^*\left(-\lambda n a_i\right) - \frac{\lambda}{2} \operatorname{tr}\left(A K A^\top\right), \end{aligned} \tag{12}$$

where L_i^* is the convex conjugate of $L_i(\cdot) = L(y_i, \cdot)$. SDCA proceeds by randomly picking a variable a_i (which in our case is a vector of dual variables over all m classes for a sample x_i) and modifying it to achieve maximal increase in the dual objective $D(A)$. It turns out that this update step is equivalent to a proximal problem, which can be seen as a regularized projection onto the essential domain of L_i^*.

The convex conjugate. An important ingredient in the SDCA framework is the convex conjugate L_i^*. We show that for all multiclass loss functions that we consider the fact that they depend on the differences $f_{y'}(x) - f_y(x)$ enforces a certain constraint on the conjugate function.

Lemma 3. *Let $H_j = \mathbf{I} - \mathbf{1} e_j^\top$ and let $\Phi(u) = \phi(H_j u)$. $\Phi^*(v) = +\infty$ unless $\langle \mathbf{1}, v \rangle = 0$.*

Proof. The proof follows directly from [28, Lemma 2] and was already reproduced in the proof of Proposition 7 for the softmax loss. We have formulated this simplified lemma since [28, Lemma 2] additionally required j-compatibility to show that if $\langle \mathbf{1}, v \rangle = 0$, then $\Phi^*(v) = \phi^*(v - v_j e_j)$, which does not hold *e.g*. for the softmax loss. □

Lemma 3 tells us that we need to enforce $\langle \mathbf{1}, a_i \rangle = 0$ at all times, which translates into $a_{y_i} = -\sum_{j \neq y_i} a_j$. The update steps are performed on the $(m-1)$-dimensional vector obtained by removing the coordinate a_{y_i}.

The update step for top-k SVM$^\gamma$**.** Let $a^{\setminus j}$ be obtained by removing the j-th coordinate from vector a. We show that performing an update step for the smooth top-k hinge loss is equivalent to projecting a certain vector b, computed from the prediction scores $W^\top x_i$, onto the essential domain of L_i^*, the top-k simplex Δ_k, with an added regularization $\rho \langle \mathbf{1}, x \rangle^2$, which biases the solution to be orthogonal to $\mathbf{1}$.

Proposition 10. *Let L_i and L_i^* in (12) be respectively the* top-k SVM$^\gamma_\alpha$ *loss and its conjugate as in Proposition 6. The update $\max_{a_i}\{D(A) \,|\, \langle \mathbf{1}, a_i \rangle = 0\}$ is equivalent with the change of variables $x \leftrightarrow -a_i^{\setminus y_i}$ to solving*

$$\min_x \{\|x - b\|^2 + \rho \langle \mathbf{1}, x \rangle^2 \mid x \in \Delta_k(\tfrac{1}{\lambda n})\}, \tag{13}$$

where $b = \frac{1}{\langle x_i, x_i \rangle + \gamma n \lambda}\left(q^{\setminus y_i} + (1 - q_{y_i})\mathbf{1}\right)$, $q = W^\top x_i - \langle x_i, x_i \rangle a_i$, and $\rho = \frac{\langle x_i, x_i \rangle}{\langle x_i, x_i \rangle + \gamma n \lambda}$.

Proof. We follow the proof of [28, Proposition 4]. We choose $i \in \{1, \dots, n\}$ and, having all other variables fixed, update a_i to maximize

$$-\frac{1}{n} L_i^*\left(-\lambda n a_i\right) - \frac{\lambda}{2} \operatorname{tr}\left(A K A^\top\right).$$

For the nonsmooth top-k hinge loss, it was shown [28] that

$$L_i^*\left(-\lambda n a_i\right) = \langle c, \lambda n (a_i - a_{y_i,i} e_{y_i}) \rangle$$

if $-\lambda n(a_i - a_{y_i,i} e_{y_i}) \in \Delta_k$ and $+\infty$ otherwise. Now, for the smoothed loss, we add regularization and obtain

$$-\frac{1}{n}\left(\frac{\gamma}{2}\left\|-\lambda n(a_i - a_{y_i,i}e_{y_i})\right\|^2 + \langle c, \lambda n(a_i - a_{y_i,i}e_{y_i})\rangle\right)$$

with $-\lambda n(a_i - a_{y_i,i}e_{y_i}) \in \Delta_k$. Using $c = \mathbf{1} - e_{y_i}$ and $\langle \mathbf{1}, a_i\rangle = 0$, one can simplify it to

$$-\frac{\gamma n \lambda^2}{2}\left\|a_i^{\setminus y_i}\right\|^2 + \lambda a_{y_i,i},$$

and the feasibility constraint can be re-written as

$$-a_i^{\setminus y_i} \in \Delta_k(\tfrac{1}{\lambda n}), \qquad a_{y_i,i} = \langle \mathbf{1}, -a_i^{\setminus y_i}\rangle.$$

For the regularization term $\operatorname{tr}\left(AKA^\top\right)$, we have

$$\operatorname{tr}\left(AKA^\top\right) = K_{ii}\left\langle a_i,a_i\right\rangle + 2\sum_{j\neq i} K_{ij}\left\langle a_i, a_j\right\rangle + \text{const.}$$

We let $q = \sum_{j\neq i} K_{ij} a_j = AK_i - K_{ii}a_i$ and $x = -a_i^{\setminus y_i}$:

$$\begin{aligned}\left\langle a_i,a_i\right\rangle &= \left\langle \mathbf{1},x\right\rangle^2 + \left\langle x,x\right\rangle, \\ \left\langle q, a_i\right\rangle &= q_{y_i}\left\langle \mathbf{1},x\right\rangle - \langle q^{\setminus y_i}, x\rangle.\end{aligned}$$

Now, we plug everything together and multiply with $-2/\lambda$.

$$\begin{aligned}\min_{x\in\Delta_k(\frac{1}{\lambda n})} &\gamma n\lambda\left\|x\right\|^2 - 2\left\langle\mathbf{1},x\right\rangle + 2\big(q_{y_i}\left\langle\mathbf{1},x\right\rangle - \langle q^{\setminus y_i},x\rangle\big) \\ &+ K_{ii}\big(\left\langle\mathbf{1},x\right\rangle^2 + \left\langle x,x\right\rangle\big).\end{aligned}$$

Collecting the corresponding terms finishes the proof. □

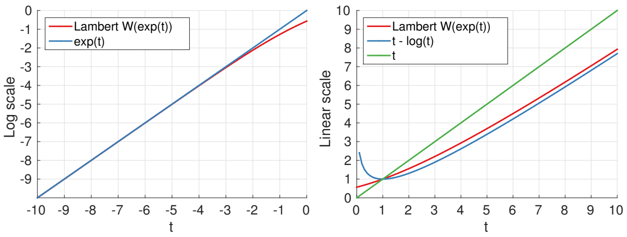
Note that setting $\gamma = 0$ we recover the update step for the non-smooth top-k hinge loss [28]. It turns out that we can employ their projection procedure for solving (13) with only minor modification of b and ρ. The smooth top-k hinge loss converges significantly faster than its nonsmooth variant as we show in the scaling experiments below. This can be explained by the theoretical results of [47] on the convergence rate of SDCA. Moreover, they had similar observations for the smoothed hinge loss for binary classification.

The update step for top-k Ent**.** We now discuss the optimization of the proposed top-k entropy loss in the SDCA framework. Note that the top-k entropy loss reduces to the softmax loss for $k = 1$. Thus our SDCA approach can be used for *gradient-free* optimization of the softmax loss without having to tune step sizes or learning rates.

Proposition 11. *Let L_i in (12) be the* top-k Ent *loss (6) and L_i^* be its convex conjugate as in (5) with Δ replaced by Δ_k. The update* $\max_{a_i}\{D(A) \mid \langle\mathbf{1}, a_i\rangle = 0\}$ *is equivalent with the change of variables $x \leftrightarrow -\lambda n a_i^{\setminus y_i}$ to solving*

$$\begin{aligned}\min_{x\in\Delta_k} &\tfrac{\alpha}{2}(\langle x,x\rangle + \langle\mathbf{1},x\rangle^2) - \langle b,x\rangle + \\ &\langle x, \log x\rangle + (1 - \langle\mathbf{1},x\rangle)\log(1 - \langle\mathbf{1},x\rangle)\end{aligned} \qquad (14)$$

where $\alpha = \frac{\langle x_i,x_i\rangle}{\lambda n}$ and $b = q^{\setminus y_i} - q_{y_i}\mathbf{1}$.



(a) $V(t) \approx e^t$ for $t \ll 0$.

(b) $V(t) \approx t - \log t$ for $t \gg 0$.

Figure 1: Behavior of the Lambert W function of the exponent ($V(t) = W(e^t)$). **(a)** Log scale plot with $t \in (-10, 0)$. **(b)** Linear scale plot with $t \in (0, 10)$.

Proof. Let $v \triangleq -\lambda n a_i$ and $j \triangleq y_i$. Using Proposition 7,

$$-\frac{1}{n}L_i^*(v) = -\frac{1}{n}\Big(\sum_{i\neq j} v_i\log v_i + (1+v_j)\log(1+v_j)\Big)$$

and $\langle\mathbf{1}, v\rangle = 0$, $v^{\setminus j} \in \Delta_k$. Let $x \triangleq v^{\setminus j}$ and $s \triangleq -v_j$. It follows that $s = \langle\mathbf{1}, x\rangle$ and

$$\begin{aligned}-\frac{\lambda}{2}\operatorname{tr}\left(AKA^\top\right) = -\frac{\lambda}{2}\Big(&K_{ii}(\langle x,x\rangle + s^2)/(\lambda n)^2 - \\ &2\left\langle q^{\setminus y_i} - q_{y_i}\mathbf{1}, x\right\rangle/(\lambda n)\Big)\end{aligned}$$

where $q = \sum_{j\neq i} K_{ij}a_j = AK_i - K_{ii}a_i$ as before. Finally, we plug everything together and multiply with $-n$. □

Note that this optimization problem is similar to (13), but is more difficult to solve due to the presence of logarithms in the objective. We propose next to tackle this problem using Lambert W functions, which we introduce below.

Lambert W function. The Lambert W function is defined to be the inverse of the function $w \mapsto we^w$ and is widely used in many fields [15, 19, 54]. Taking logarithms on both sides of the defining equation $z = We^W$, we obtain $\log z = W(z) + \log W(z)$. Therefore, if we are given an equation of the form $x + \log x = t$ for some $t \in \mathbb{R}$, we can directly "solve" it in closed-form as $x = W(e^t)$. The crux of the problem is that the function $V(t) = W(e^t)$ is transcendental [19] just like the logarithm and the exponent. There exist highly optimized implementations for the latter and we argue that the same can be done for the Lambert W function. In fact, there is already some work on this topic [19, 54], which we also employ in our implementation.

To develop intuition concerning the Lambert W function of the exponent, we now briefly discuss how the function $V(t) = W(e^t)$ behaves for different values of t. An illustration is provided in Figure 1. One can see directly from the equation $x + \log x = t$ that the behavior of $x = V(t)$ changes dramatically depending on whether t is a large positive or a large negative number. In the first case, the linear part dominates the logarithm and the function is approximately linear; a better approximation is $x(t) \approx t - \log t$, when $t \gg 1$. In the second case, the function behaves like an exponent e^t. To see this, we write $x = e^t e^{-x}$ and note that $e^{-x} \approx 1$ when $t \ll 0$, therefore, $x(t) \approx e^t$, if $t \ll 0$. We use these approximations as initial points for a 5-th order Householder method [23], which was also used in [19]. A *single* iteration is already sufficient to get full `float` precision and at most two iterations are needed for `double`.

How to solve (14). We present a similar derivation as was already done for the problem (6) above. The main difference is that we now encounter the Lambert W function in the optimality conditions. We re-write the problem as

$$\min_{\substack{s, x \in \Delta_k \\ \langle \mathbf{1}, x \rangle = s}} \tfrac{\alpha}{2}(\langle x, x \rangle + s^2) - \langle a, x \rangle + \langle x, \log x \rangle + (1 - s) \log(1 - s).$$

The Lagrangian is given by

$$\begin{aligned} \mathcal{L} =& \tfrac{\alpha}{2}(\langle x, x \rangle + s^2) - \langle a, x \rangle + \langle x, \log x \rangle \\ &+ (1 - s) \log(1 - s) + t(\langle \mathbf{1}, x \rangle - s) \\ &+ \lambda(s - 1) - \langle \mu, x \rangle + \left\langle \nu, x - \tfrac{s}{k}\mathbf{1} \right\rangle, \end{aligned}$$

where $t \in \mathbb{R}$, $\lambda, \mu, \nu \geq 0$ are the dual variables. Computing partial derivatives of $\mathcal{L}$ w.r.t. x_i and s, and setting them to zero, we obtain

$$\begin{aligned} \alpha x_i + \log x_i &= a_i - 1 - t + \mu_i - \nu_i, \quad \forall i, \\ \alpha(1 - s) + \log(1 - s) &= \alpha - 1 - t - \lambda - \tfrac{1}{k} \langle \mathbf{1}, \nu \rangle, \quad \forall i. \end{aligned}$$

Note that $x_i > 0$ and $s < 1$ as before, which implies $\mu_i = 0$ and $\lambda = 0$. We re-write the above as

$$\begin{aligned} &\alpha x_i + \log(\alpha x_i) = a_i - 1 - t + \log \alpha - \nu_i, \\ &\alpha(1 - s) + \log\big(\alpha(1 - s)\big) = \alpha - 1 - t + \log \alpha - \tfrac{\langle \mathbf{1}, \nu \rangle}{k}. \end{aligned}$$

Note that these equations correspond to the Lambert W function of the exponent, *i.e.* $V(t) = W(e^t)$ discussed above. Let $p \triangleq \langle \mathbf{1}, \nu \rangle$ and re-define $t \leftarrow 1 + t - \log \alpha$.

$$\begin{aligned} \alpha x_i &= W\big(\exp(a_i - t - \nu_i)\big), \\ \alpha(1 - s) &= W\big(\exp(\alpha - t - \tfrac{p}{k})\big). \end{aligned}$$

Finally, we obtain the following system:

$$\begin{aligned} x_i &= \min\{\tfrac{1}{\alpha} V(a_i - t), \tfrac{s}{k}\}, \quad \forall i, \\ \alpha x_i &= V(a_i - t - \nu_i), \quad \forall i, \\ s &= \langle \mathbf{1}, x \rangle, \\ p &= \langle \mathbf{1}, \nu \rangle, \\ \alpha(1 - s) &= V(\alpha - t - \tfrac{p}{k}). \end{aligned}$$

Note that $V(t)$ is a strictly monotonically increasing function, therefore, it is invertible and we can write

$$\begin{aligned} a_i - t - \nu_i &= V^{-1}(\alpha x_i), \\ \alpha - t - \tfrac{p}{k} &= V^{-1}\big(\alpha(1 - s)\big). \end{aligned}$$

Next, we defined the sets U and M as before and write

$$\begin{aligned} s &= \langle \mathbf{1}, x \rangle = \sum_U \tfrac{s}{k} + \sum_M \tfrac{1}{\alpha} V(a_i - t), \\ p &= \langle \mathbf{1}, \nu \rangle = \sum_U a_i - |U| \left(t + V^{-1}(\tfrac{\alpha s}{k})\right). \end{aligned}$$

Let $\rho \triangleq \frac{|U|}{k}$ as before and $A \triangleq \frac{1}{k} \sum_U a_i$, we get

$$\begin{aligned} (1 - \rho)s &= \tfrac{1}{\alpha} \sum_M V(a_i - t), \\ \tfrac{p}{k} &= A - \rho\big(t + V^{-1}(\tfrac{\alpha s}{k})\big). \end{aligned}$$

Finally, we eliminate p and obtain a system in *two* variables,

$$\begin{aligned} &\alpha(1 - \rho)s - \sum_M V(a_i - t) = 0, \\ &(1 - \rho)t + V^{-1}\big(\alpha(1 - s)\big) - \rho V^{-1}(\tfrac{\alpha s}{k}) + A - \alpha = 0, \end{aligned}$$

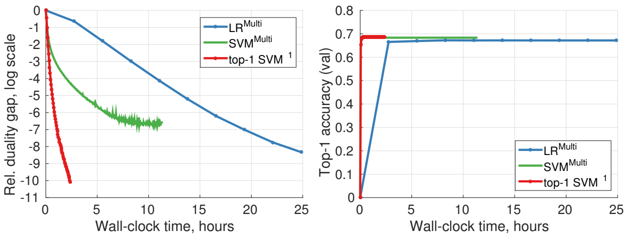
which can be solved using the Newton's method [36]. We also note the following identities, which are well-known [15] or can be easily derived. If $V(t) = W(\exp(t))$, then

$$\begin{aligned} \partial_t V(t) &= V(t)/(1 + V(t)), \\ V^{-1}(v) &= v + \log v, \\ \partial_v V^{-1}(v) &= 1 + 1/v. \end{aligned}$$

Therefore, at each iteration of the Newton's method, we only need to compute $V(a_i - t)$ **once** for each i in M and then can use it for the derivative of V.

The efficiency of this approach crucially depends on fast computation of $V(t)$. Our implementation was already sufficient enough to scale the training procedure to large datasets as we show next.

Runtime comparison. We compare the runtime of the top-1 SVM [28] (SVM$^{\text{Multi}}$) with our smooth top-1 SVM (top-1 SVM^1) and the top-1 entropy (LR$^{\text{Multi}}$) objectives in Figure 2. We plot the relative duality gap $((P(W) - D(A))/P(W))$ and the top-1 validation accuracy versus time for the best performing models on ILSVRC 2012. We obtain substantial improvement of the convergence rate for smooth top-1 SVM compared to the non-smooth baseline. Moreover, top-1 accuracy saturates after a few passes over the training data, which suggests the usage of a fairly loose stopping criterion (we used 10^{-3}). For LR$^{\text{Multi}}$, the cost of each epoch is significantly higher compared to the top-1 SVMs, which is due to the difficulty of solving (14). This suggests that one can use the smooth top-1 SVM^1 and obtain performance competitive with LR$^{\text{Multi}}$ but having a magnitude smaller training time (see also § 5).



(a) Relative duality gap vs. time (b) Top-1 accuracy vs. time

Figure 2: SDCA convergence with $\mathrm{LR}^{\mathrm{Multi}}$, $\mathrm{SVM}^{\mathrm{Multi}}$, and the proposed top-$k$ SVM^{1} objectives on ILSVRC 2012.

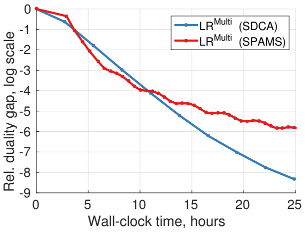


Figure 3: Convergence rate of SDCA (ours) and the SPAMS toolbox [33].

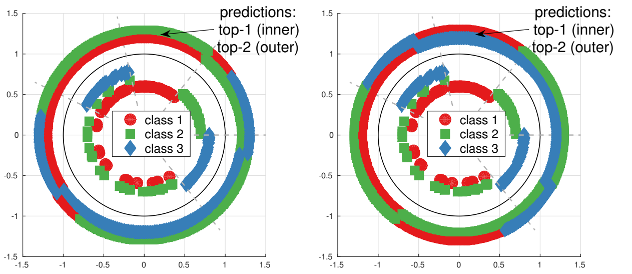
We also compare our SDCA-based implementation for optimizing the softmax loss, $\mathrm{LR}^{\mathrm{Multi}}$ (SDCA), with a state-of-the-art optimization toolbox SPAMS of Mairal *et al.* [33], denoted $\mathrm{LR}^{\mathrm{Multi}}$ (SPAMS). The latter provides efficient implementation of various optimization methods including FISTA [4], which we chose to compare against our method. We note that the rate of convergence of SDCA is competitive with FISTA for $\epsilon \geq 10^{-4}$ and is noticeably better for $\epsilon < 10^{-4}$. We conclude that our approach based on the Lambert W function is competitive with the state-of-the-art, and faster computation of $V(t)$ would lead to a further speedup.

4. Synthetic Example

In this section, we demonstrate in a synthetic experiment that our proposed top-2 losses outperform the top-1 losses when one aims at optimal top-2 performance. The dataset with three classes is shown in the inner circle of Figure 4.

Sampling. First, we generate samples in $[0, 7]$ which is subdivided into 5 segments. All segments have unit length, except for the 4-th segment which has length 3. We sample uniformly at random in each of the 5 segments according to the following class-conditional probabilities: $(0, 1, .4, .3, 0)$ for class 1, $(1, 0, .1, .7, 0)$ for class 2, and $(0, 0, .5, 0, 1)$ for class 3. Finally, the data is rescaled to $[0, 1]$ and mapped onto the unit circle.

Samples of different classes are plotted next to each other for better visibility as there is significant class overlap. We visualize top-1/2 predictions with two colored circles (outside the black circle). We sample 200/200/200K points for training/validation/testing and tune the $C = \frac{1}{\lambda n}$ parameter in the range 2^{-18} to 2^{18}. Results are in Table 2.



(a) top-1 SVM^{1} test accuracy (top-1 / top-2): 65.7% / 81.3% (b) top-2 LR_{n} test accuracy (top-1 / top-2): 29.4%, 96.1%

Figure 4: Synthetic data on the unit circle in $\mathbb{R}^2$ (inside black circle) and visualization of top-1 and top-2 predictions (outside black circle). **(a)** Smooth top-1 SVM^{1} optimizes top-1 error which impedes its top-2 error. **(b)** Trunc. top-2 softmax loss ignores top-1 scores and optimizes directly top-2 errors leading to a much better top-2 result.

Circle (synthetic)

Method	Top-1	Top-2	Method	Top-1	Top-2
$\mathrm{SVM}^{\mathrm{OVA}}$	54.3	85.8	top-1 SVM^{1}	**65.7**	83.9
$\mathrm{LR}^{\mathrm{OVA}}$	54.7	81.7	top-2 $\mathrm{SVM}^{0/1}$	54.4 / 54.5	87.1 / 87.0
$\mathrm{SVM}^{\mathrm{Multi}}$	58.9	89.3	top-2 Ent	54.6	87.6
$\mathrm{LR}^{\mathrm{Multi}}$	54.7	81.7	top-2 LR_{n}	58.4	**96.1**

Table 2: Top-k accuracy (%) on synthetic data. **Left:** Baselines methods. **Right:** Top-k SVM (nonsmooth / smooth) and top-k softmax losses (convex and nonconvex).

In each column we provide the results for the model that optimizes the corresponding top-k accuracy, which is in general different for top-1 and top-2. First, we note that all top-1 baselines perform similar in top-1 performance, except for $\mathrm{SVM}^{\mathrm{Multi}}$ and top-1 SVM^{1} which show better results. However, most importantly, we see that direct optimization of top-2 accuracy with our convex top-2 losses improves the results of the top-1 losses. Most significant is the improvement for the nonconvex top-2 LR_{n} loss (note that top-1 LR_{n} corresponds to $\mathrm{LR}^{\mathrm{Multi}}$), which is close to the Bayes optimal solution for this dataset. The reason for this good performance is that the non-convex loss top-2 LR_{n} is a very tight bound on the top-2 error, in particular it ignores higher top-1 scores in the loss. Unfortunately, this good performance does not transfer to the real-world data sets. The reason might lie in the high dimension of the feature space which yields relatively well separable problems.

5. Experimental Results

The goal of this section is to provide an extensive empirical evaluation of the top-k performance of different losses in multiclass classification. To this end, we evaluate the loss functions introduced in § 2 on 11 datasets (500 to 2.4M training examples, 10 to 1000 classes), from various prob-

	ALOI				**Letter**				**News 20**				**Caltech 101 Silhouettes**			
State-of-the-art	93 ± 1.2 [44]				97.98 [24] (RBF kernel)				86.9 [42]				62.1	79.6	83.4	[50]
Method	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	82.4	89.5	91.5	93.7	63.0	82.0	88.1	94.6	84.3	95.4	97.9	**99.5**	61.8	76.5	80.8	86.6
$\mathrm{LR}^{\mathrm{OVA}}$	86.1	93.0	94.8	96.6	68.1	86.1	90.6	96.2	84.9	96.3	97.8	99.3	63.2	80.4	84.4	89.4
$\mathrm{SVM}^{\mathrm{Multi}}$	90.0	95.1	96.7	98.1	76.5	89.2	93.1	97.7	85.4	94.9	97.2	99.1	62.8	77.8	82.0	86.9
$\mathrm{LR}^{\mathrm{Multi}}$	89.8	95.7	97.1	98.4	75.3	90.3	94.3	98.0	84.5	96.4	98.1	**99.5**	63.2	**81.2**	85.1	89.7
top-3 SVM	89.2	95.5	97.2	98.4	74.0	91.0	94.4	97.8	85.1	96.6	98.2	99.3	63.4	79.7	83.6	88.3
top-5 SVM	87.3	95.6	97.4	98.6	70.8	**91.5**	95.1	98.4	84.3	96.7	98.4	99.3	63.3	80.0	84.3	88.7
top-10 SVM	85.0	95.5	97.3	**98.7**	61.6	88.9	96.0	99.6	82.7	96.5	98.4	99.3	63.0	80.5	84.6	89.1
top-1 SVM^1	**90.6**	95.5	96.7	98.2	**76.8**	89.9	93.6	97.6	**85.6**	96.3	98.0	99.3	**63.9**	80.3	84.0	89.0
top-3 SVM^1	89.6	95.7	97.3	98.4	74.1	90.9	94.5	97.9	85.1	96.6	98.4	99.4	63.3	80.1	84.0	89.2
top-5 SVM^1	87.6	95.7	**97.5**	98.6	70.8	**91.5**	95.2	98.6	84.5	96.7	98.4	99.4	63.3	80.5	84.5	89.1
top-10 SVM^1	85.2	95.6	97.4	**98.7**	61.7	89.1	95.9	**99.7**	82.9	96.5	98.4	**99.5**	63.1	80.5	84.8	89.1
top-3 Ent	89.0	95.8	97.2	98.4	73.0	90.8	94.9	98.5	84.7	96.6	98.3	99.4	63.3	81.1	85.0	89.9
top-5 Ent	87.9	95.8	97.2	98.4	69.7	90.9	95.1	98.8	84.3	**96.8**	**98.6**	99.4	63.2	80.9	85.2	89.9
top-10 Ent	86.0	95.6	97.3	98.5	65.0	89.7	**96.2**	99.6	82.7	96.4	98.5	99.4	62.5	80.8	**85.4**	90.1
top-3 LR_{n}	89.3	**95.9**	97.3	98.5	63.6	91.1	95.6	98.8	83.4	96.4	98.3	99.4	60.7	81.1	85.2	**90.2**
top-5 LR_{n}	87.9	95.7	97.3	98.6	50.3	87.7	96.1	99.4	83.2	96.0	98.2	99.4	58.3	79.8	85.2	**90.2**
top-10 LR_{n}	85.2	94.8	97.1	98.5	46.5	80.9	93.7	99.6	82.9	95.7	97.9	99.4	51.9	78.4	84.6	**90.2**

	Indoor 67				**CUB**				**Flowers**				**FMD**		
State-of-the-art	82.0 [57]				62.8 [14] / 76.37 [60]				86.8 [40]				77.4 [14] / 82.4 [14]		
Method	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5
$\mathrm{SVM}^{\mathrm{OVA}}$	81.9	94.3	96.5	98.0	60.6	77.1	83.4	89.9	82.0	91.7	94.3	96.8	77.4	92.4	96.4
$\mathrm{LR}^{\mathrm{OVA}}$	82.0	94.9	97.2	98.7	62.3	80.5	87.4	93.5	82.6	92.2	94.8	97.6	79.6	94.2	**98.2**
$\mathrm{SVM}^{\mathrm{Multi}}$	82.5	**95.4**	97.3	99.1	61.0	79.2	85.7	92.3	82.5	92.2	94.8	96.4	77.6	93.8	97.2
$\mathrm{LR}^{\mathrm{Multi}}$	82.4	95.2	**98.0**	99.1	62.3	81.7	87.9	**93.9**	82.9	92.4	95.1	97.8	79.0	94.6	97.8
top-3 SVM	81.6	95.1	97.7	99.0	61.3	80.4	86.3	92.5	81.9	92.2	95.0	96.1	78.8	94.6	97.8
top-5 SVM	79.9	95.0	97.7	99.0	60.9	81.2	87.2	92.9	81.7	92.4	95.1	97.8	78.4	94.4	97.6
top-10 SVM	78.4	95.1	97.4	99.0	59.6	81.3	87.7	93.4	80.5	91.9	95.1	97.7			
top-1 SVM^1	**82.6**	95.2	97.6	99.0	61.9	80.2	86.9	93.1	**83.0**	92.4	95.1	97.6	78.6	93.8	98.0
top-3 SVM^1	81.6	95.1	97.8	99.0	61.9	81.1	86.6	93.2	82.5	92.3	95.2	97.7	79.0	94.4	98.0
top-5 SVM^1	80.4	95.1	97.8	99.1	61.3	81.3	87.4	92.9	82.0	**92.5**	95.1	97.8	79.4	94.4	97.6
top-10 SVM^1	78.3	95.1	97.5	99.0	59.8	81.4	87.8	93.4	80.6	91.9	95.1	97.7			
top-3 Ent	81.4	**95.4**	97.6	**99.2**	**62.5**	81.8	87.9	**93.9**	82.5	92.0	**95.3**	97.8	**79.8**	94.8	98.0
top-5 Ent	80.3	95.0	97.7	99.0	62.0	**81.9**	88.1	93.8	82.1	92.2	95.1	**97.9**	79.4	94.4	98.0
top-10 Ent	79.2	95.1	97.6	99.0	61.2	81.6	**88.2**	93.8	80.9	92.1	95.0	97.7			
top-3 LR_{n}	79.8	95.0	97.5	99.1	62.0	81.4	87.6	93.4	82.1	92.2	95.2	97.6	78.4	**95.4**	**98.2**
top-5 LR_{n}	76.4	94.3	97.3	99.0	61.4	81.2	87.7	93.7	81.4	92.0	95.0	97.7	77.2	94.0	97.8
top-10 LR_{n}	72.6	92.8	97.1	98.9	59.7	80.7	87.2	93.4	77.9	91.1	94.3	97.3			

	SUN 397 (10 splits)				**Places 205** (val)				**ILSVRC 2012** (val)			
State-of-the-art	66.9 [57]				60.6		88.5	[57]	76.3		93.2	[49]
Method	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10	Top-1	Top-3	Top-5	Top-10
$\mathrm{SVM}^{\mathrm{Multi}}$	65.8 ± 0.1	85.1 ± 0.2	90.8 ± 0.1	95.3 ± 0.1	58.4	78.7	84.7	89.9	68.3	82.9	87.0	91.1
$\mathrm{LR}^{\mathrm{Multi}}$	**67.5 ± 0.1**	**87.7 ± 0.2**	**92.9 ± 0.1**	**96.8 ± 0.1**	59.0	**80.6**	**87.6**	**94.3**	67.2	83.2	87.7	92.2
top-3 SVM	66.5 ± 0.2	86.5 ± 0.1	91.8 ± 0.1	95.9 ± 0.1	58.6	80.3	87.3	93.3	68.2	84.0	88.1	92.1
top-5 SVM	66.3 ± 0.2	87.0 ± 0.2	92.2 ± 0.2	96.3 ± 0.1	58.4	80.5	87.4	94.0	67.8	**84.1**	88.2	92.4
top-10 SVM	64.8 ± 0.3	87.2 ± 0.2	92.6 ± 0.1	96.6 ± 0.1	58.0	80.4	87.4	**94.3**	67.0	83.8	88.3	**92.6**
top-1 SVM^1	67.4 ± 0.2	86.8 ± 0.1	92.0 ± 0.1	96.1 ± 0.1	**59.2**	80.5	87.3	93.8	**68.7**	83.9	88.0	92.1
top-3 SVM^1	67.0 ± 0.2	87.0 ± 0.1	92.2 ± 0.1	96.2 ± 0.0	58.9	80.5	**87.6**	93.9	68.2	**84.1**	88.2	92.3
top-5 SVM^1	66.5 ± 0.2	87.2 ± 0.1	92.4 ± 0.2	96.3 ± 0.0	58.5	80.5	87.5	94.1	67.9	**84.1**	**88.4**	92.5
top-10 SVM^1	64.9 ± 0.3	87.3 ± 0.2	92.6 ± 0.2	96.6 ± 0.1	58.0	80.4	87.5	**94.3**	67.1	83.8	88.3	**92.6**
top-3 Ent	67.2 ± 0.2	**87.7 ± 0.2**	**92.9 ± 0.1**	**96.8 ± 0.1**	58.7	**80.6**	**87.6**	94.2	66.8	83.1	87.8	92.2
top-5 Ent	66.6 ± 0.3	**87.7 ± 0.2**	**92.9 ± 0.1**	**96.8 ± 0.1**	58.1	80.4	87.4	94.2	66.5	83.0	87.7	92.2
top-10 Ent	65.2 ± 0.3	87.4 ± 0.1	92.8 ± 0.1	**96.8 ± 0.1**	57.0	80.0	87.2	94.1	65.8	82.8	87.6	92.1

Table 3: Top-k accuracy (%) on various datasets. The first line is a reference to the state-of-the-art on each dataset and reports top-1 accuracy except when the numbers are aligned with Top-k. We compare the one-vs-all and multiclass baselines with the top-k SVM [28] as well as the proposed smooth top-k SVM^{γ}, top-k Ent, and the nonconvex top-k LR_{n}.

lem domains (vision and non-vision; fine-grained, scene and general object classification). The detailed statistics of the datasets is given in Table 4.

Dataset	m	n	d	Dataset	m	n	d
ALOI [44]	1K	54K	128	Indoor 67 [38]	67	5354	4K
Caltech 101 Sil [50]	101	4100	784	Letter [24]	26	10.5K	16
CUB [56]	202	5994	4K	News 20 [27]	20	15.9K	16K
Flowers [35]	102	2040	4K	Places 205 [61]	205	2.4M	4K
FMD [48]	10	500	4K	SUN 397 [59]	397	19.9K	4K
ILSVRC 2012 [46]	1K	1.3M	4K				

Table 4: Statistics of the datasets used in the experiments (m – # classes, n – # training examples, d – # features).

Please refer to Table 1 for an overview of the methods that we consider and our naming convention. A broad selection of results is also reported at the end of the paper. As other ranking based losses did not perform well in [28], we do no further comparison here.

Solvers. We use LibLinear [18] for the one-vs-all baselines $\mathrm{SVM}^{\mathrm{OVA}}$ and $\mathrm{LR}^{\mathrm{OVA}}$; and the code of [28] for top-k SVM. We extend[3] the latter to support the smooth top-k SVM^{γ} and top-k Ent. The multiclass loss baselines $\mathrm{SVM}^{\mathrm{Multi}}$ and $\mathrm{LR}^{\mathrm{Multi}}$ correspond respectively to top-1 SVM and top-1 Ent. For the nonconvex top-k LR_{n}, we use the $\mathrm{LR}^{\mathrm{Multi}}$ solution as an initial point and perform gradient descent with line search. We cross-validate hyper-parameters in the range 10^{-5} to 10^{3}, extending it when the optimal value is at the boundary.

Features. For ALOI, Letter, and News20 datasets, we use the features provided by the LibSVM [13] datasets. For ALOI, we randomly split the data into equally sized training and test sets preserving class distributions. The Letter dataset comes with a separate validation set, which we used for model selection only. For News20, we use PCA to reduce dimensionality of sparse features from 62060 to 15478 preserving all non-singular PCA components[4].

For Caltech101 Silhouettes, we use the features and the train/val/test splits provided by [50].

For CUB, Flowers, FMD, and ILSVRC 2012, we use MatConvNet [55] to extract the outputs of the last fully connected layer of the imagenet-vgg-verydeep-16 model which is pre-trained on ImageNet [17] and achieves state-of-the-art results in image classification [49].

For Indoor 67, SUN 397, and Places 205, we use the Places205-VGGNet-16 model by [57] which is pre-trained on Places 205 [61] and outperforms the ImageNet pre-trained model on scene classification tasks [57]. Further results can be found at the end of the paper. In all cases we obtain a similar behavior in terms of the ranking of the considered losses as discussed below.

Discussion. The experimental results are given in Table 3. There are several interesting observations that one can make. While the OVA schemes perform quite similar to the multiclass approaches (logistic OVA vs. softmax, hinge OVA vs. multiclass SVM), which confirms earlier observations in [43, 2], the OVA schemes performed worse on ALOI and Letter. Therefore it seems safe to recommend to use multiclass losses instead of the OVA schemes.

Comparing the softmax vs. multiclass SVM losses, we see that there is no clear winner in top-1 performance, but softmax consistently outperforms multiclass SVM in top-k performance for $k > 1$. This might be due to the strong property of softmax being top-k calibrated for all k. Please note that this trend is uniform across all datasets, in particular, also for the ones where the features are not coming from a convnet. Therefore, the positive trend of softmax cannot be directly explained by the fact that the convnet-features were trained using the softmax loss.

Both the smooth top-k hinge loss and the top-k entropy loss perform slightly better than softmax if one compares the corresponding top-k errors. A similar statement holds for the nonconvex top-k loss even though the good performance from the synthetic dataset does not transfer to the real world datasets. This might have to do with the relatively high dimension of the feature spaces, but requires further investigation.

We conclude that if one just wants to use a single loss for multiclass problems, then the softmax is the method of choice as it yields competitive results for all top-k errors. A competitive alternative is the smooth top-1 hinge loss which is much faster to train (see Section 3). If one is willing to sacrifice the top-1 error and optimize directly for a top-k error, then further improvements are possible using either the smooth SVM or the entropy top-k loss.

6. Conclusion

We have done an extensive experimental study of top-k performance optimization. We observed that the softmax loss and the smooth top-1 hinge loss are competitive across all top-k errors and should be considered the primary candidates in practice. Our new top-k loss functions can further improve these results slightly, especially if one is targeting a particular top-k error as the performance measure. Finally, we would like to highlight our new optimization scheme based on SDCA for the top-k entropy loss which also includes the softmax loss and is of an independent interest.

References

[1] S. Agarwal. The infinite push: A new support vector ranking algorithm that directly optimizes accuracy at the absolute top of the list. In *SDM*, pages 839–850, 2011. 1

[2] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Good practice in large-scale learning for image classifica-

[3] https://github.com/mlapin/libsdca

[4] The top-k SVM solvers that we used were designed for dense inputs.

tion. *Pattern Analysis and Machine Intelligence*, 36(3):507–520, 2014. 1, 4, 17

[3] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe. Convexity, classification and risk bounds. *Journal of the Americal Stat. Assoc.*, 101:138–156, 2006. 2

[4] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. 15

[5] A. Beck and M. Teboulle. Smoothing and first order methods, a unified framework. *SIAM J. Optimization*, 22:557–580, 2012. 6

[6] Y. Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, 2009. 4

[7] C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, 2006. 4

[8] A. Bordes, L. Bottou, P. Gallinari, and J. Weston. Solving multiclass support vector machines with LaRank. In *ICML*, pages 89–96, 2007. 1

[9] J. M. Borwein and A. S. Lewis. *Convex Analysis and Nonlinear Optimization: Theory and Examples*. Cms Books in Mathematics Series. Springer Verlag, 2000. 12

[10] S. Boyd, C. Cortes, M. Mohri, and A. Radovanovic. Accuracy at the top. In *NIPS*, pages 953–961, 2012. 1

[11] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. 9, 10, 11

[12] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML*, pages 89–96, 2005. 1

[13] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:1–27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 17

[14] M. Cimpoi, S. Maji, and A. Vedaldi. Deep filter banks for texture recognition and segmentation. In *CVPR*, 2015. 16

[15] R. M. Corless, G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth. On the lambert W function. *Advances in Computational mathematics*, 5(1):329–359, 1996. 13, 14

[16] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292, 2001. 4, 8

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 17

[18] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. 17

[19] T. Fukushima. Precise and fast computation of Lambert W-functions without transcendental function evaluations. *Journal of Computational and Applied Mathematics*, 244:77 – 89, 2013. 13, 14

[20] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *ICCV*, pages 221–228, 2009. 24, 25

[21] M. R. Gupta, S. Bengio, and J. Weston. Training highly multiclass classifiers. *JMLR*, 15:1461–1492, 2014. 1

[22] J.-B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of Convex Analysis*. Springer, Berlin, 2001. 6, 7

[23] A. S. Householder. *The Numerical Treatment of a Single Nonlinear Equation*. McGraw-Hill, 1970. 14

[24] C.-W. Hsu and C.-J. Lin. A comparison of methods for multiclass support vector machines. *Neural Networks*, 13(2):415–425, 2002. 16, 17

[25] T. Joachims. A support vector method for multivariate performance measures. In *ICML*, pages 377–384, 2005. 1

[26] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012. 4

[27] K. Lang. Newsweeder: Learning to filter netnews. In *ICML*, pages 331–339, 1995. 17

[28] M. Lapin, M. Hein, and B. Schiele. Top-k multiclass SVM. In *NIPS*, 2015. 1, 6, 7, 8, 12, 13, 14, 16, 17, 29, 33, 35, 37

[29] M. Lapin, B. Schiele, and M. Hein. Scalable multitask representation learning for scene classification. In *CVPR*, 2014. 32

[30] N. Li, R. Jin, and Z.-H. Zhou. Top rank optimization in linear time. In *NIPS*, pages 1502–1510, 2014. 1

[31] L. Liu, T. G. Dietterich, N. Li, and Z. Zhou. Transductive optimization of top k precision. *CoRR*, abs/1510.05976, 2015. 1

[32] J. Mairal. *Sparse Coding for Machine Learning, Image Processing and Computer Vision*. PhD thesis, Ecole Normale Superieure de Cachan, 2010. 8

[33] J. Mairal, R. Jenatton, F. R. Bach, and G. R. Obozinski. Network flow algorithms for structured sparsity. In *NIPS*, pages 1558–1566, 2010. 15

[34] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005. 6

[35] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, pages 722–729, 2008. 17

[36] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Science+ Business Media, 2006. 14

[37] K. B. Petersen, M. S. Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 450:7–15, 2008. 8

[38] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, 2009. 17

[39] A. Rakotomamonjy. Sparse support vector infinite push. In *ICML*, pages 1335–1342. ACM, 2012. 1

[40] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: an astounding baseline for recognition. In *CVPRW, DeepVision workshop*, 2014. 16

[41] M. Reid and B. Williamson. Composite binary losses. *Journal of Machine Learning Research*, 11:2387–2422, 2010. 6

[42] J. D. Rennie. Improving multi-class text classification with naive bayes. Technical report, Massachusetts Institute of Technology, 2001. 16

[43] R. Rifkin and A. Klautau. In defense of one-vs-all classification. *The Journal of Machine Learning Research*, 5:101–141, 2004. 17

[44] A. Rocha and S. Klein Goldenstein. Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches. *Neural Networks and Learning Systems, IEEE Transactions on*, 25(2):289–302, 2014. 16, 17

[45] C. Rudin. The p-norm push: A simple convex ranking algorithm that concentrates at the top of the list. *The Journal of Machine Learning Research*, 10:2233–2271, 2009. 1

[46] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge, 2014. 1, 17

[47] S. Shalev-Shwartz and T. Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, pages 1–41, 2014. 1, 6, 7, 9, 12, 13

[48] L. Sharan, R. Rosenholtz, and E. Adelson. Material perception: What can you see in a brief glance? *Journal of Vision*, 9(8):784–784, 2009. 17

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 4, 16, 17

[50] K. Swersky, B. J. Frey, D. Tarlow, R. S. Zemel, and R. P. Adams. Probabilistic n-choose-k models for classification and ranking. In *NIPS*, pages 3050–3058, 2012. 1, 16, 17

[51] A. Tewari and P. Bartlett. On the consistency of multiclass classification methods. *Journal of Machine Learning Research*, 8:1007–1025, 2007. 2, 4

[52] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, pages 1453–1484, 2005. 1

[53] N. Usunier, D. Buffoni, and P. Gallinari. Ranking with ordered weighted pairwise classification. In *ICML*, pages 1057–1064, 2009. 1, 6, 11

[54] D. Veberič. Lambert W function for applications in physics. *Computer Physics Communications*, 183(12):2622–2628, 2012. 13

[55] A. Vedaldi and K. Lenc. Matconvnet – convolutional neural networks for matlab. In *Proceeding of the ACM Int. Conf. on Multimedia*, 2015. 17

[56] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical report, California Institute of Technology, 2011. 17

[57] L. Wang, S. Guo, W. Huang, and Y. Qiao. Places205-vggnet models for scene recognition. *CoRR*, abs/1508.01667, 2015. 16, 17

[58] J. Weston, S. Bengio, and N. Usunier. Wsabie: scaling up to large vocabulary image annotation. *IJCAI*, pages 2764–2770, 2011. 1

[59] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 1, 17

[60] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based rcnn for fine-grained detection. In *ECCV*, 2014. 16

[61] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, 2014. 1, 17

ALOI

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	82.4	87.4	89.5	90.7	91.5	92.1	92.6	93.1	93.4	93.7
$\mathrm{LR}^{\mathrm{OVA}}$	86.1	91.1	93.0	94.1	94.8	95.4	95.8	96.1	96.4	96.6
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	90.0	93.4	95.1	96.0	96.7	97.1	97.5	97.7	97.9	98.1
top-2 SVM_α	90.0	94.0	95.5	96.4	97.0	97.4	97.7	97.9	98.1	98.3
top-3 SVM_α	89.2	94.2	95.5	96.7	97.2	97.6	97.8	98.1	98.2	98.4
top-4 SVM_α	88.4	94.3	95.6	96.8	97.4	97.8	98.0	98.2	98.4	98.5
top-5 SVM_α	87.3	94.1	95.6	96.9	97.4	97.8	98.0	98.3	98.4	98.6
top-10 SVM_α	85.0	93.0	95.5	96.5	97.3	97.8	98.2	98.4	98.6	98.7
top-1 SVM^1_α	90.6	94.2	95.5	96.2	96.7	97.0	97.6	97.9	98.1	98.2
top-2 SVM^1_α	90.3	94.3	95.6	96.3	96.8	97.1	97.7	98.0	98.2	98.3
top-3 SVM^1_α	89.6	94.4	95.7	96.7	97.3	97.6	97.9	98.1	98.3	98.4
top-4 SVM^1_α	88.7	94.4	95.7	96.9	97.4	97.8	98.0	98.2	98.4	98.5
top-5 SVM^1_α	87.6	94.3	95.7	96.9	97.5	97.8	98.1	98.3	98.4	98.6
top-10 SVM^1_α	85.2	93.1	95.6	96.6	97.4	97.8	98.2	98.4	98.6	98.7
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	90.0	93.4	95.1	96.0	96.7	97.1	97.5	97.7	97.9	98.1
top-2 SVM_β	90.2	93.9	95.2	96.0	96.7	97.1	97.5	97.7	97.9	98.1
top-3 SVM_β	90.2	94.1	95.4	96.0	96.5	96.9	97.5	97.8	98.0	98.1
top-4 SVM_β	90.1	94.2	95.4	96.1	96.6	97.0	97.3	97.5	98.0	98.2
top-5 SVM_β	90.0	94.3	95.5	96.2	96.7	97.1	97.3	97.5	97.7	98.2
top-10 SVM_β	89.5	94.2	95.7	96.5	96.9	97.3	97.5	98.0	97.9	98.3
top-1 SVM^1_β	90.6	94.2	95.5	96.2	96.7	97.0	97.6	97.9	98.1	98.2
top-2 SVM^1_β	90.6	94.2	95.5	96.2	96.7	97.0	97.3	97.9	98.1	98.2
top-3 SVM^1_β	90.4	94.3	95.6	96.3	96.7	97.1	97.4	97.6	98.1	98.2
top-4 SVM^1_β	90.3	94.4	95.6	96.3	96.8	97.1	97.4	97.6	97.8	97.9
top-5 SVM^1_β	90.2	94.4	95.7	96.3	96.8	97.2	97.5	97.7	97.8	98.0
top-10 SVM^1_β	89.5	94.3	95.7	96.6	97.0	97.4	97.6	97.8	98.1	98.2
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	89.8	94.2	95.7	96.5	97.1	97.5	97.8	98.0	98.2	98.4
top-2 Ent	89.4	94.2	95.8	96.6	97.1	97.5	97.8	98.0	98.2	98.4
top-3 Ent	89.0	94.3	95.8	96.6	97.2	97.5	97.8	98.0	98.2	98.4
top-4 Ent	88.5	94.2	95.8	96.7	97.2	97.6	97.8	98.1	98.3	98.4
top-5 Ent	87.9	94.2	95.8	96.7	97.2	97.6	97.9	98.1	98.3	98.4
top-10 Ent	86.0	93.2	95.6	96.7	97.3	97.7	98.0	98.2	98.4	98.5
top-2 LR_n	89.8	94.4	95.9	96.7	97.2	97.6	97.9	98.1	98.3	98.5
top-3 LR_n	89.3	94.3	95.9	96.7	97.3	97.7	98.0	98.2	98.3	98.5
top-4 LR_n	88.7	94.0	95.8	96.7	97.3	97.7	98.0	98.2	98.4	98.5
top-5 LR_n	87.9	93.7	95.7	96.7	97.3	97.7	98.0	98.2	98.4	98.6
top-10 LR_n	85.2	92.4	94.8	96.3	97.1	97.6	98.0	98.2	98.4	98.5

Table 5: Comparison of different methods in top-k accuracy (%).

Letter

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
SVM$^{\text{OVA}}$	63.0	75.7	82.0	85.7	88.1	89.9	91.4	92.8	93.9	94.6
LR$^{\text{OVA}}$	68.1	81.1	86.1	88.4	90.6	92.2	93.4	94.6	95.3	96.2
top-1 SVM$_\alpha$ / SVM$^{\text{Multi}}$	76.5	85.5	89.2	91.5	93.1	94.3	95.5	96.5	97.0	97.7
top-2 SVM$_\alpha$	76.1	86.9	90.1	92.2	93.3	94.8	96.0	96.5	97.2	97.7
top-3 SVM$_\alpha$	74.0	87.0	91.0	93.0	94.4	95.4	96.2	96.7	97.3	97.8
top-4 SVM$_\alpha$	71.4	86.4	91.4	93.5	94.8	95.7	96.6	97.2	97.6	98.2
top-5 SVM$_\alpha$	70.8	85.9	91.5	93.9	95.1	96.2	96.9	97.5	98.1	98.4
top-10 SVM$_\alpha$	61.6	82.5	88.9	93.6	96.0	97.6	98.3	98.9	99.2	99.6
top-1 SVM$^1_\alpha$	76.8	86.0	89.9	92.1	93.6	94.9	95.8	96.3	97.0	97.6
top-2 SVM$^1_\alpha$	76.2	87.0	90.3	92.5	94.0	94.9	96.0	96.6	97.3	97.7
top-3 SVM$^1_\alpha$	74.1	87.1	90.9	93.2	94.5	95.6	96.4	96.9	97.3	97.9
top-4 SVM$^1_\alpha$	72.1	86.7	91.4	93.2	94.7	95.7	96.7	97.3	97.8	98.0
top-5 SVM$^1_\alpha$	70.8	86.2	91.5	93.8	95.2	96.3	97.0	97.6	98.2	98.6
top-10 SVM$^1_\alpha$	61.7	82.9	89.1	93.6	95.9	97.5	98.3	98.9	99.3	99.7
top-1 SVM$_\beta$ / SVM$^{\text{Multi}}$	76.5	85.5	89.2	91.5	93.1	94.3	95.5	96.5	97.0	97.7
top-2 SVM$_\beta$	76.5	86.4	90.2	92.1	93.8	94.8	95.9	96.7	97.3	97.8
top-3 SVM$_\beta$	75.6	86.9	90.6	92.7	94.2	95.3	95.8	96.7	97.4	97.9
top-4 SVM$_\beta$	74.9	86.9	90.9	93.1	94.6	95.5	96.3	96.9	97.5	98.1
top-5 SVM$_\beta$	74.5	86.9	91.0	93.4	94.9	95.7	96.5	97.2	97.8	98.2
top-10 SVM$_\beta$	72.9	85.9	90.8	93.5	95.3	96.2	97.2	97.8	98.3	98.8
top-1 SVM$^1_\beta$	76.8	86.0	89.9	92.1	93.6	94.9	95.8	96.3	97.0	97.6
top-2 SVM$^1_\beta$	76.5	86.6	90.2	92.4	93.8	95.0	96.0	96.8	97.4	97.8
top-3 SVM$^1_\beta$	75.6	86.9	90.6	92.8	94.2	95.3	96.1	96.6	97.3	97.9
top-4 SVM$^1_\beta$	75.1	87.0	90.8	93.1	94.6	95.5	96.3	97.0	97.8	98.1
top-5 SVM$^1_\beta$	74.6	87.0	91.1	93.4	94.9	95.8	96.6	97.2	97.8	98.2
top-10 SVM$^1_\beta$	72.9	86.1	90.8	93.6	95.2	96.3	97.1	97.9	98.3	98.8
top-1 Ent / LR$^{\text{Multi}}$	75.2	86.5	90.1	93.1	94.5	95.7	96.5	97.4	98.1	98.1
top-2 Ent	74.6	86.5	90.6	93.2	94.7	95.7	96.5	97.5	98.0	98.4
top-3 Ent	73.0	86.4	90.8	93.3	94.9	95.9	96.8	97.6	98.2	98.5
top-4 Ent	71.9	86.1	91.0	93.6	95.3	96.3	97.1	97.7	98.3	98.6
top-5 Ent	69.7	85.6	90.9	93.7	95.1	96.7	97.3	97.9	98.6	98.8
top-10 Ent	65.0	82.5	89.7	93.6	96.2	97.7	98.4	98.9	99.3	99.6
top-2 LR$_{\text{n}}$	70.1	86.3	90.7	93.7	95.1	96.1	97.0	97.7	98.1	98.5
top-3 LR$_{\text{n}}$	63.6	83.9	91.1	93.9	95.6	96.7	97.5	98.1	98.5	98.8
top-4 LR$_{\text{n}}$	58.7	80.3	90.0	93.9	96.1	97.2	98.0	98.5	98.9	99.3
top-5 LR$_{\text{n}}$	50.3	76.4	87.7	93.7	96.1	97.5	98.3	98.7	99.2	99.4
top-10 LR$_{\text{n}}$	46.5	67.9	80.9	88.9	93.7	96.3	97.7	98.7	99.3	99.6

Table 6: Comparison of different methods in top-k accuracy (%).

News 20

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	84.3	93.0	95.4	97.0	97.9	98.5	98.8	99.0	99.3	99.5
$\mathrm{LR}^{\mathrm{OVA}}$	84.9	93.1	96.3	97.2	97.8	98.3	98.7	98.8	99.0	99.3
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	85.4	92.7	94.9	96.4	97.2	97.6	98.2	98.5	98.7	99.1
top-2 SVM_α	85.3	94.3	96.4	97.2	97.9	98.4	98.5	98.9	99.0	99.1
top-3 SVM_α	85.1	94.8	96.6	97.7	98.2	98.6	99.0	99.2	99.3	99.3
top-4 SVM_α	85.0	94.5	96.7	97.7	98.3	98.6	98.9	99.1	99.3	99.4
top-5 SVM_α	84.3	94.2	96.7	97.8	98.4	98.7	99.0	99.2	99.3	99.3
top-10 SVM_α	82.7	93.3	96.5	97.7	98.4	98.6	99.0	99.2	99.3	99.3
top-1 SVM^1_α	85.6	94.5	96.3	97.3	98.0	98.5	98.8	99.0	98.9	99.3
top-2 SVM^1_α	85.2	94.6	96.5	97.6	98.1	98.6	98.9	99.1	99.3	99.3
top-3 SVM^1_α	85.1	94.7	96.6	97.8	98.4	98.7	99.0	99.1	99.3	99.4
top-4 SVM^1_α	84.9	94.4	96.7	97.8	98.4	98.7	99.0	99.1	99.3	99.4
top-5 SVM^1_α	84.5	94.4	96.7	97.9	98.4	98.8	99.0	99.1	99.3	99.4
top-10 SVM^1_α	82.9	93.5	96.5	97.8	98.4	98.7	99.0	99.2	99.3	99.5
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	85.4	92.7	94.9	96.4	97.2	97.6	98.2	98.5	98.7	99.1
top-2 SVM_β	85.7	94.3	96.1	97.4	97.8	97.8	98.2	98.6	98.9	99.1
top-3 SVM_β	86.0	94.5	96.5	97.5	98.1	98.4	98.7	99.0	99.2	99.2
top-4 SVM_β	85.9	94.8	96.6	97.7	98.1	98.4	98.7	99.0	99.2	99.2
top-5 SVM_β	85.4	94.8	96.7	97.7	98.3	98.7	99.0	99.1	99.3	99.4
top-10 SVM_β	84.9	94.6	96.7	98.0	98.5	98.7	99.0	99.2	99.3	99.3
top-1 SVM^1_β	85.6	94.5	96.3	97.3	98.0	98.5	98.8	99.0	98.9	99.3
top-2 SVM^1_β	85.8	94.5	96.3	97.4	98.0	98.4	98.8	99.0	99.2	99.3
top-3 SVM^1_β	85.7	94.5	96.5	97.6	98.1	98.5	98.8	99.0	99.3	99.3
top-4 SVM^1_β	85.6	94.6	96.6	97.6	98.2	98.6	98.9	99.1	99.0	99.4
top-5 SVM^1_β	85.6	94.7	96.6	97.6	98.3	98.7	99.0	99.2	99.3	99.3
top-10 SVM^1_β	84.9	94.7	96.7	97.9	98.5	98.7	99.0	99.1	99.3	99.3
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	84.5	94.2	96.4	97.6	98.1	98.5	99.0	99.1	99.3	99.5
top-2 Ent	84.7	94.4	96.5	97.7	98.3	98.6	98.9	99.2	99.3	99.3
top-3 Ent	84.7	94.6	96.6	97.8	98.3	98.7	98.9	99.2	99.3	99.4
top-4 Ent	84.5	94.5	96.7	97.9	98.5	98.7	99.0	99.2	99.3	99.4
top-5 Ent	84.3	94.3	96.8	97.8	98.6	98.8	99.0	99.1	99.3	99.4
top-10 Ent	82.7	93.3	96.4	97.8	98.5	98.7	98.9	99.1	99.3	99.4
top-2 LR_n	84.2	93.9	96.6	97.6	98.4	98.7	98.9	99.1	99.3	99.4
top-3 LR_n	83.4	93.6	96.4	97.7	98.3	98.6	98.9	99.2	99.2	99.4
top-4 LR_n	83.3	93.5	96.2	97.6	98.3	98.6	98.9	99.1	99.2	99.3
top-5 LR_n	83.2	93.3	96.0	97.7	98.2	98.5	98.9	99.1	99.2	99.4
top-10 LR_n	82.9	92.7	95.7	97.0	97.9	98.4	98.8	99.0	99.2	99.4

Table 7: Comparison of different methods in top-k accuracy (%).

Caltech 101 Silhouettes

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	61.8	73.1	76.5	78.5	80.8	82.5	83.6	84.6	85.6	86.6
$\mathrm{LR}^{\mathrm{OVA}}$	63.2	77.1	80.4	82.6	84.4	85.7	87.0	87.9	88.6	89.4
top-1 SVM_{α} / $\mathrm{SVM}^{\mathrm{Multi}}$	62.8	74.6	77.8	80.0	82.0	83.3	84.4	85.0	85.9	86.9
top-2 SVM_{α}	63.1	76.2	79.0	81.0	82.7	84.4	85.5	86.2	87.0	87.6
top-3 SVM_{α}	63.4	76.7	79.7	81.5	83.6	85.1	85.8	86.6	87.6	88.3
top-4 SVM_{α}	63.2	76.6	79.8	82.4	84.0	85.3	86.0	86.9	87.8	88.6
top-5 SVM_{α}	63.3	76.8	80.0	82.7	84.3	85.6	86.5	87.5	88.1	88.7
top-10 SVM_{α}	63.0	77.3	80.5	82.7	84.6	85.9	86.8	88.0	88.8	89.1
top-1 $\mathrm{SVM}^{1}_{\alpha}$	63.9	76.9	80.3	82.4	84.0	85.4	86.4	87.2	88.4	89.0
top-2 $\mathrm{SVM}^{1}_{\alpha}$	63.8	76.9	80.5	82.3	84.1	85.3	86.5	87.4	88.4	89.0
top-3 $\mathrm{SVM}^{1}_{\alpha}$	63.3	77.2	80.1	82.3	84.0	85.7	86.5	87.3	88.0	89.2
top-4 $\mathrm{SVM}^{1}_{\alpha}$	63.1	77.0	80.3	82.4	84.3	85.5	86.6	87.3	88.2	89.2
top-5 $\mathrm{SVM}^{1}_{\alpha}$	63.3	77.1	80.5	82.7	84.5	85.9	86.7	87.6	88.5	89.1
top-10 $\mathrm{SVM}^{1}_{\alpha}$	63.1	77.2	80.5	82.8	84.8	86.0	87.0	88.1	88.5	89.1
top-1 SVM_{β} / $\mathrm{SVM}^{\mathrm{Multi}}$	62.8	74.6	77.8	80.0	82.0	83.3	84.4	85.0	85.9	86.9
top-2 SVM_{β}	63.5	76.2	79.3	81.1	82.6	84.3	85.3	86.3	87.0	87.9
top-3 SVM_{β}	63.9	76.6	79.7	81.4	83.4	84.9	85.7	86.6	87.4	88.0
top-4 SVM_{β}	63.9	76.9	80.1	82.0	83.5	85.0	85.9	87.0	87.8	88.7
top-5 SVM_{β}	63.6	77.0	80.4	82.6	84.2	85.3	86.3	87.4	88.1	89.0
top-10 SVM_{β}	64.0	77.1	80.5	83.0	84.9	86.2	87.3	87.9	88.8	89.4
top-1 SVM^{1}_{β}	63.9	76.9	80.3	82.4	84.0	85.4	86.4	87.2	88.4	89.0
top-2 SVM^{1}_{β}	63.9	77.1	80.4	82.3	84.1	85.5	86.5	87.3	88.4	89.0
top-3 SVM^{1}_{β}	64.0	77.0	80.4	82.4	84.2	85.4	86.6	87.3	88.4	89.1
top-4 SVM^{1}_{β}	63.7	77.0	80.2	82.6	84.4	85.7	86.7	87.4	88.4	89.0
top-5 SVM^{1}_{β}	63.7	77.2	80.5	82.4	84.4	85.9	86.9	87.6	88.4	89.1
top-10 SVM^{1}_{β}	64.2	77.3	80.6	83.2	85.0	86.2	87.1	88.0	88.8	89.5
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	63.2	77.8	81.2	83.4	85.1	86.5	87.3	88.7	89.3	89.7
top-2 Ent	63.3	77.5	81.1	83.3	85.0	86.5	87.4	88.6	89.4	89.8
top-3 Ent	63.3	77.5	81.1	83.3	85.0	86.6	87.4	88.7	89.2	89.9
top-4 Ent	63.2	77.5	81.0	83.2	85.0	86.5	87.5	88.6	89.3	89.9
top-5 Ent	63.2	77.5	80.9	83.1	85.2	86.4	87.5	88.5	89.3	89.9
top-10 Ent	62.5	77.3	80.8	83.4	85.4	86.6	87.9	88.5	89.3	90.1
top-2 LR_{n}	62.7	77.5	81.3	83.5	85.5	86.9	87.8	88.7	89.3	90.0
top-3 LR_{n}	60.7	76.9	81.1	83.7	85.2	87.0	88.2	88.8	89.8	90.2
top-4 LR_{n}	60.0	75.9	80.4	83.1	85.1	86.9	88.2	88.9	89.7	90.5
top-5 LR_{n}	58.3	75.4	79.8	82.7	85.2	86.9	88.0	88.8	89.6	90.2
top-10 LR_{n}	51.9	72.1	78.4	82.0	84.6	86.2	87.5	88.5	89.4	90.2

Table 8: Comparison of different methods in top-k accuracy (%).

Caltech 101 (average of 39 kernels from [20], 5 splits)

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-2 SVM_α	73.2 ± 0.9	81.7 ± 0.7	85.5 ± 0.9	87.8 ± 0.7	89.6 ± 0.6	90.9 ± 0.5	91.9 ± 0.4	92.7 ± 0.3	93.3 ± 0.4	93.9 ± 0.5
top-3 SVM_α	73.0 ± 1.0	81.6 ± 0.7	85.4 ± 0.9	87.8 ± 0.6	89.6 ± 0.6	90.9 ± 0.4	91.8 ± 0.3	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.5
top-4 SVM_α	72.8 ± 1.0	81.4 ± 0.7	85.4 ± 0.9	87.9 ± 0.6	89.6 ± 0.5	90.9 ± 0.4	91.8 ± 0.3	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-5 SVM_α	72.6 ± 1.0	81.2 ± 0.7	85.2 ± 0.8	87.7 ± 0.6	89.5 ± 0.6	90.8 ± 0.5	91.7 ± 0.4	92.6 ± 0.4	93.2 ± 0.5	93.7 ± 0.5
top-10 SVM_α	71.0 ± 0.8	80.2 ± 1.0	84.2 ± 0.7	87.0 ± 0.8	88.8 ± 0.7	90.1 ± 0.5	91.3 ± 0.4	92.2 ± 0.5	92.9 ± 0.4	93.5 ± 0.4
top-1 SVM^1_α	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-2 SVM^1_α	73.2 ± 0.9	81.7 ± 0.7	85.5 ± 0.9	87.8 ± 0.7	89.6 ± 0.6	90.9 ± 0.5	91.9 ± 0.4	92.7 ± 0.3	93.3 ± 0.4	93.9 ± 0.5
top-3 SVM^1_α	73.0 ± 1.0	81.6 ± 0.7	85.4 ± 0.8	87.8 ± 0.6	89.6 ± 0.6	90.9 ± 0.4	91.8 ± 0.3	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-4 SVM^1_α	72.8 ± 1.0	81.4 ± 0.7	85.4 ± 0.9	87.9 ± 0.6	89.6 ± 0.5	90.9 ± 0.4	91.8 ± 0.3	92.7 ± 0.4	93.3 ± 0.5	93.8 ± 0.4
top-5 SVM^1_α	72.5 ± 1.0	81.2 ± 0.7	85.2 ± 0.8	87.7 ± 0.6	89.5 ± 0.6	90.8 ± 0.4	91.7 ± 0.4	92.6 ± 0.4	93.2 ± 0.5	93.8 ± 0.4
top-10 SVM^1_α	71.0 ± 0.8	80.2 ± 1.0	84.2 ± 0.7	87.0 ± 0.8	88.8 ± 0.7	90.1 ± 0.5	91.3 ± 0.4	92.2 ± 0.5	92.9 ± 0.4	93.5 ± 0.4
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-2 SVM_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-3 SVM_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-4 SVM_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-5 SVM_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-10 SVM_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.9 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.5
top-1 SVM^1_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-2 SVM^1_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-3 SVM^1_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-4 SVM^1_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-5 SVM^1_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-10 SVM^1_β	73.2 ± 1.0	81.6 ± 0.8	85.5 ± 0.8	87.8 ± 0.7	89.5 ± 0.6	90.8 ± 0.5	91.8 ± 0.4	92.7 ± 0.4	93.3 ± 0.4	93.8 ± 0.4
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	72.7 ± 0.8	80.9 ± 0.9	84.9 ± 0.9	87.4 ± 0.7	89.1 ± 0.7	90.5 ± 0.6	91.6 ± 0.5	92.4 ± 0.3	93.0 ± 0.5	93.6 ± 0.5
top-2 Ent	72.6 ± 0.9	80.9 ± 0.8	85.0 ± 0.9	87.5 ± 0.8	89.2 ± 0.5	90.6 ± 0.5	91.6 ± 0.5	92.4 ± 0.4	93.1 ± 0.4	93.6 ± 0.4
top-3 Ent	72.5 ± 0.9	80.8 ± 0.8	85.0 ± 0.9	87.3 ± 0.7	89.2 ± 0.6	90.5 ± 0.5	91.6 ± 0.4	92.4 ± 0.4	93.1 ± 0.4	93.6 ± 0.5
top-4 Ent	72.2 ± 1.0	80.7 ± 0.8	84.8 ± 0.9	87.3 ± 0.7	89.0 ± 0.7	90.5 ± 0.5	91.5 ± 0.5	92.4 ± 0.3	93.1 ± 0.5	93.6 ± 0.5
top-5 Ent	72.0 ± 0.8	80.5 ± 0.9	84.7 ± 0.8	87.2 ± 0.7	89.0 ± 0.6	90.4 ± 0.5	91.5 ± 0.3	92.3 ± 0.2	93.0 ± 0.4	93.6 ± 0.4
top-10 Ent	70.2 ± 0.9	79.8 ± 1.1	83.5 ± 0.6	86.6 ± 0.5	88.4 ± 0.6	89.7 ± 0.8	90.8 ± 0.7	91.9 ± 0.4	92.7 ± 0.4	93.0 ± 0.5

Table 9: Comparison of different methods in top-k accuracy (%).

Caltech 256 (average of 39 kernels from [20])

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	47.1	56.3	61.4	64.7	67.3	69.3	71.5	72.6	74.1	75.2
top-2 SVM_α	47.1	56.3	61.4	64.6	67.2	69.4	71.3	72.6	74.1	75.2
top-3 SVM_α	47.0	56.3	61.5	64.6	67.3	69.4	71.3	72.6	74.1	75.1
top-4 SVM_α	46.9	56.2	61.4	64.5	67.3	69.3	71.3	72.8	74.1	75.2
top-5 SVM_α	46.8	56.1	61.2	64.4	67.1	69.3	71.2	72.8	74.2	75.1
top-10 SVM_α	45.4	55.3	60.7	64.1	66.7	69.0	71.0	72.6	73.9	75.2
top-1 SVM^1_α	47.1	56.3	61.4	64.5	67.1	69.3	71.5	72.8	74.1	75.2
top-2 SVM^1_α	47.1	56.3	61.4	64.4	67.0	69.3	71.3	72.7	74.0	75.2
top-3 SVM^1_α	47.0	56.2	61.5	64.5	67.3	69.4	71.3	72.7	74.1	75.1
top-4 SVM^1_α	46.9	56.2	61.4	64.5	67.3	69.3	71.1	72.8	74.2	75.2
top-5 SVM^1_α	46.8	56.1	61.3	64.3	67.0	69.4	71.1	72.9	74.2	75.2
top-10 SVM^1_α	45.4	55.3	60.8	64.1	66.6	68.9	71.0	72.6	74.0	75.2
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	47.1	56.3	61.4	64.7	67.3	69.3	71.5	72.6	74.1	75.2
top-2 SVM_β	47.2	56.3	61.4	64.7	67.2	69.4	71.4	72.6	74.1	75.2
top-3 SVM_β	47.2	56.3	61.4	64.7	67.2	69.4	71.4	72.6	74.1	75.2
top-4 SVM_β	47.2	56.3	61.4	64.7	67.2	69.4	71.4	72.6	74.1	75.2
top-5 SVM_β	47.2	56.3	61.4	64.7	67.2	69.3	71.4	72.6	74.1	75.2
top-10 SVM_β	47.2	56.3	61.4	64.7	67.2	69.3	71.5	72.7	74.1	75.2
top-1 SVM^1_β	47.1	56.3	61.4	64.5	67.1	69.3	71.5	72.8	74.1	75.2
top-2 SVM^1_β	47.1	56.3	61.4	64.5	67.1	69.3	71.5	72.8	74.1	75.2
top-3 SVM^1_β	47.1	56.3	61.4	64.5	67.1	69.3	71.5	72.7	74.1	75.2
top-4 SVM^1_β	47.1	56.3	61.4	64.5	67.1	69.3	71.5	72.7	74.1	75.2
top-5 SVM^1_β	47.1	56.3	61.4	64.5	67.1	69.3	71.5	72.7	74.1	75.2
top-10 SVM^1_β	47.2	56.3	61.4	64.5	67.1	69.3	71.5	72.7	74.1	75.2
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	46.2	55.6	60.9	64.1	66.7	69.1	71.0	72.5	73.7	74.9
top-2 Ent	46.2	55.7	60.9	64.1	66.7	69.0	71.0	72.0	73.3	74.5
top-3 Ent	46.1	55.8	61.0	64.2	66.8	69.0	70.9	72.5	73.7	74.8
top-4 Ent	46.1	55.5	60.9	64.2	66.7	68.9	70.9	71.5	73.9	74.8
top-5 Ent	46.0	55.4	60.7	64.1	66.7	68.8	70.7	72.7	73.2	74.2
top-10 Ent	45.0	54.6	59.9	63.7	66.2	68.4	70.4	72.3	73.5	74.7

Table 10: Comparison of different methods in top-k accuracy (%).

CUB

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	60.6	71.4	77.1	80.7	83.4	85.2	86.6	87.9	89.0	89.9
$\mathrm{LR}^{\mathrm{OVA}}$	62.3	74.8	80.5	84.6	87.4	89.4	90.7	91.9	92.8	93.5
top-1 SVM_{α} / $\mathrm{SVM}^{\mathrm{Multi}}$	61.0	73.3	79.2	82.8	85.7	87.8	89.5	90.7	91.6	92.3
top-2 SVM_{α}	61.2	73.7	79.9	83.5	85.9	88.2	89.9	91.0	91.9	92.6
top-3 SVM_{α}	61.3	74.9	80.4	83.9	86.3	88.1	89.9	91.3	91.9	92.5
top-4 SVM_{α}	61.1	75.1	81.1	84.6	86.7	88.7	89.8	90.9	91.9	92.8
top-5 SVM_{α}	60.9	74.7	81.2	84.7	87.2	89.0	90.4	91.0	92.1	92.9
top-10 SVM_{α}	59.6	73.9	81.3	85.1	87.7	89.6	90.7	91.7	92.7	93.4
top-1 $\mathrm{SVM}_{\alpha}^{1}$	61.9	74.3	80.2	84.0	86.9	88.6	90.1	91.4	92.3	93.1
top-2 $\mathrm{SVM}_{\alpha}^{1}$	62.0	74.7	80.5	84.0	86.9	88.6	90.1	91.3	92.2	93.0
top-3 $\mathrm{SVM}_{\alpha}^{1}$	61.9	75.1	81.1	84.2	86.6	88.6	90.2	91.4	92.2	93.2
top-4 $\mathrm{SVM}_{\alpha}^{1}$	61.7	75.1	81.2	84.7	87.1	89.0	90.3	91.4	92.3	93.0
top-5 $\mathrm{SVM}_{\alpha}^{1}$	61.3	75.0	81.3	85.0	87.4	89.2	90.6	91.2	92.2	92.9
top-10 $\mathrm{SVM}_{\alpha}^{1}$	59.8	73.9	81.4	85.2	87.8	89.7	90.7	91.8	92.7	93.4
top-1 SVM_{β} / $\mathrm{SVM}^{\mathrm{Multi}}$	61.0	73.3	79.2	82.8	85.7	87.8	89.5	90.7	91.6	92.3
top-2 SVM_{β}	61.0	73.4	79.2	83.1	86.0	88.1	89.8	91.1	91.9	92.6
top-3 SVM_{β}	61.3	73.7	79.8	83.5	86.4	88.2	89.9	91.2	91.9	92.5
top-4 SVM_{β}	61.5	73.9	79.7	83.8	86.6	88.3	89.8	90.9	91.9	92.8
top-5 SVM_{β}	61.8	74.4	79.8	83.8	86.4	88.5	90.1	91.1	91.6	92.4
top-10 SVM_{β}	62.1	74.9	80.8	84.4	86.8	88.8	90.1	91.3	92.2	93.1
top-1 SVM_{β}^{1}	61.9	74.3	80.2	84.0	86.9	88.6	90.1	91.4	92.3	93.1
top-2 SVM_{β}^{1}	61.9	74.3	80.2	84.0	86.9	88.6	90.2	91.3	92.2	93.1
top-3 SVM_{β}^{1}	62.0	74.3	80.2	83.9	86.7	88.6	90.2	91.4	92.2	93.2
top-4 SVM_{β}^{1}	61.9	74.4	80.2	83.9	86.8	88.5	90.2	91.3	92.3	93.1
top-5 SVM_{β}^{1}	61.9	74.4	80.4	83.9	86.7	88.7	90.1	91.5	92.4	93.0
top-10 SVM_{β}^{1}	62.4	74.9	80.9	84.3	86.8	88.8	90.3	91.3	92.1	92.9
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	62.3	75.2	81.7	85.2	87.9	89.6	91.3	92.5	93.2	93.9
top-2 Ent	62.4	75.4	81.6	85.2	87.9	89.7	91.3	92.5	93.2	93.9
top-3 Ent	62.5	75.6	81.8	85.4	87.9	89.6	91.2	92.4	93.2	93.9
top-4 Ent	62.3	75.6	81.4	85.4	87.8	89.6	91.1	92.4	93.2	93.9
top-5 Ent	62.0	75.4	81.9	85.4	88.1	90.0	91.2	92.4	93.2	93.8
top-10 Ent	61.2	74.7	81.6	85.3	88.2	90.0	91.3	92.4	93.3	93.8
top-2 LR_{n}	61.9	74.8	81.2	84.7	87.6	89.4	90.9	91.9	92.7	93.4
top-3 LR_{n}	62.0	75.0	81.4	84.9	87.6	89.4	90.8	91.9	93.0	93.4
top-4 LR_{n}	61.8	74.5	81.1	84.9	87.7	89.6	91.0	92.0	93.0	93.6
top-5 LR_{n}	61.4	74.8	81.2	85.0	87.7	89.7	91.1	92.0	92.9	93.7
top-10 LR_{n}	59.7	73.2	80.7	84.7	87.2	89.4	90.8	91.9	92.8	93.4

Table 11: Comparison of different methods in top-k accuracy (%).

Flowers

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	82.0	89.2	91.7	93.2	94.3	95.2	95.9	95.8	96.2	96.8
$\mathrm{LR}^{\mathrm{OVA}}$	82.6	89.4	92.2	93.9	94.8	95.8	96.4	96.9	97.3	97.6
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	82.5	89.5	92.2	93.8	94.8	95.6	96.2	95.5	96.0	96.4
top-2 SVM_α	82.3	89.5	92.3	93.9	95.0	95.7	96.5	95.6	95.9	96.2
top-3 SVM_α	81.9	89.3	92.2	93.8	95.0	95.8	96.4	97.0	96.0	96.1
top-4 SVM_α	81.8	89.3	92.3	94.0	95.0	95.9	96.6	97.0	95.6	97.8
top-5 SVM_α	81.7	89.1	92.4	94.1	95.1	95.8	96.6	97.1	97.4	97.8
top-10 SVM_α	80.5	88.8	91.9	93.7	95.1	95.9	96.5	97.1	97.4	97.7
top-1 SVM^1_α	83.0	89.8	92.4	94.0	95.1	95.9	96.4	96.7	97.3	97.6
top-2 SVM^1_α	82.6	89.6	92.4	94.0	95.2	95.9	96.5	96.9	97.3	97.6
top-3 SVM^1_α	82.5	89.7	92.3	94.1	95.2	95.8	96.5	97.1	97.3	97.7
top-4 SVM^1_α	82.3	89.3	92.4	94.1	95.2	96.0	96.6	97.1	97.5	97.7
top-5 SVM^1_α	82.0	89.3	92.5	94.1	95.1	95.9	96.6	97.1	97.5	97.8
top-10 SVM^1_α	80.6	88.8	91.9	93.7	95.1	95.9	96.6	97.0	97.4	97.7
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	82.5	89.5	92.2	93.8	94.8	95.6	96.2	95.5	96.0	96.4
top-2 SVM_β	82.5	89.6	92.2	93.7	94.9	95.6	96.2	95.6	95.9	96.3
top-3 SVM_β	82.4	89.8	92.1	93.7	94.8	95.7	96.2	96.8	95.8	96.1
top-4 SVM_β	82.4	89.5	92.1	93.7	94.8	95.6	96.2	96.7	95.6	96.0
top-5 SVM_β	82.5	89.7	92.0	93.7	94.9	95.6	96.2	96.7	95.8	96.2
top-10 SVM_β	82.7	89.7	92.3	93.9	95.0	95.6	96.2	96.7	97.2	97.5
top-1 SVM^1_β	83.0	89.8	92.4	94.0	95.1	95.9	96.4	96.7	97.3	97.6
top-2 SVM^1_β	83.0	89.8	92.4	94.0	95.1	95.9	96.4	96.7	97.3	97.6
top-3 SVM^1_β	83.0	89.8	92.4	94.0	95.1	95.9	96.5	96.7	97.4	97.6
top-4 SVM^1_β	82.9	89.7	92.4	94.0	95.0	96.0	96.5	96.9	97.3	97.6
top-5 SVM^1_β	83.0	89.9	92.4	93.9	95.1	95.9	96.5	96.7	97.4	97.6
top-10 SVM^1_β	82.7	89.8	92.4	94.0	95.1	96.0	96.2	96.7	97.3	97.6
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	82.9	89.7	92.4	94.0	95.1	96.0	96.6	97.1	97.4	97.8
top-2 Ent	82.6	89.7	92.4	94.0	95.3	96.0	96.6	97.1	97.5	97.8
top-3 Ent	82.5	89.5	92.0	94.1	95.3	96.1	96.6	97.1	97.4	97.8
top-4 Ent	82.2	89.5	92.4	94.1	95.3	96.0	96.7	97.1	97.5	97.8
top-5 Ent	82.1	89.4	92.2	94.1	95.1	95.9	96.6	97.1	97.5	97.9
top-10 Ent	80.9	88.9	92.1	93.9	95.0	95.9	96.5	97.0	97.4	97.7
top-2 LR_{n}	82.1	89.4	92.3	93.8	95.0	96.0	96.5	97.2	97.4	97.8
top-3 LR_{n}	82.1	89.2	92.2	94.2	95.2	96.0	96.7	97.0	97.4	97.6
top-4 LR_{n}	81.9	88.8	92.3	94.1	95.0	95.9	96.6	97.2	97.5	97.7
top-5 LR_{n}	81.4	89.0	92.0	93.8	95.0	95.7	96.2	97.0	97.3	97.7
top-10 LR_{n}	77.9	87.8	91.1	93.0	94.3	95.2	96.0	96.6	96.9	97.3

Table 12: Comparison of different methods in top-k accuracy (%).

FMD

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	77.4	87.2	92.4	94.4	96.4	97.8	99.0	99.8	100.0	100.0
$\mathrm{LR}^{\mathrm{OVA}}$	79.6	90.2	94.2	96.6	98.2	98.8	99.2	100.0	99.8	100.0
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	77.6	88.8	93.8	95.4	97.2	98.4	100.0	100.0	100.0	100.0
top-2 SVM_α	78.2	89.2	94.0	95.8	97.4	98.6	99.8	100.0	100.0	100.0
top-3 SVM_α	78.8	89.2	94.6	96.4	97.8	98.8	99.4	99.8	100.0	100.0
top-4 SVM_α	78.2	89.4	94.6	96.8	98.0	98.8	99.4	99.6	100.0	100.0
top-5 SVM_α	78.4	89.2	94.4	96.8	97.6	98.6	99.2	99.6	99.8	100.0
top-1 SVM^1_α	78.6	89.4	93.8	96.0	98.0	99.0	99.4	99.6	100.0	100.0
top-2 SVM^1_α	78.4	90.2	93.8	96.2	97.6	99.0	99.2	99.6	99.8	100.0
top-3 SVM^1_α	79.0	89.6	94.4	96.2	98.0	99.0	99.2	99.6	99.8	100.0
top-4 SVM^1_α	79.2	89.4	94.6	96.6	97.8	98.8	99.2	99.6	99.8	100.0
top-5 SVM^1_α	79.4	89.2	94.4	96.8	97.6	98.8	99.2	99.2	99.8	100.0
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	77.6	88.8	93.8	95.4	97.2	98.4	100.0	100.0	100.0	100.0
top-2 SVM_β	79.0	89.6	93.6	95.4	97.4	98.6	99.8	100.0	100.0	100.0
top-3 SVM_β	79.8	89.6	93.6	95.4	97.8	98.8	99.2	99.8	100.0	100.0
top-4 SVM_β	80.4	90.0	93.8	95.6	97.8	98.8	99.2	99.6	100.0	100.0
top-5 SVM_β	80.2	90.2	94.8	95.8	97.4	98.8	99.6	100.0	100.0	100.0
top-1 SVM^1_β	78.6	89.4	93.8	96.0	98.0	99.0	99.4	99.6	100.0	100.0
top-2 SVM^1_β	78.6	89.4	93.8	96.0	98.0	99.0	99.4	99.6	100.0	100.0
top-3 SVM^1_β	79.6	89.8	94.0	96.2	98.2	99.0	99.2	99.6	100.0	100.0
top-4 SVM^1_β	79.4	90.2	94.2	96.2	97.8	98.8	99.4	99.6	100.0	100.0
top-5 SVM^1_β	80.0	90.2	94.6	96.0	97.6	98.8	99.4	100.0	100.0	100.0
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	79.0	90.6	94.6	96.6	97.8	98.8	99.2	100.0	99.8	100.0
top-2 Ent	79.4	89.6	94.6	97.6	98.0	98.8	99.2	100.0	99.8	100.0
top-3 Ent	79.8	89.4	94.8	97.4	98.0	98.8	99.2	99.2	99.8	100.0
top-4 Ent	79.2	90.2	94.8	97.0	97.8	98.8	99.0	99.2	99.8	100.0
top-5 Ent	79.4	90.4	94.4	97.2	98.0	98.8	99.2	99.2	99.8	100.0
top-2 LR_{n}	79.0	89.4	94.4	96.6	98.2	99.0	99.2	100.0	100.0	100.0
top-3 LR_{n}	78.4	89.6	95.4	97.4	98.2	98.6	99.2	100.0	100.0	100.0
top-4 LR_{n}	77.8	89.4	94.8	96.6	98.0	98.8	99.4	99.8	100.0	100.0
top-5 LR_{n}	77.2	89.4	94.0	96.4	97.8	98.8	99.2	100.0	100.0	100.0

Table 13: Comparison of different methods in top-k accuracy (%).

Indoor 67 (AlexNet trained on Places 205, FC7 output, provided by [28])

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
SVM$^{\mathrm{OVA}}$	71.7	80.1	85.9	90.2	92.6	94.3	94.9	95.7	96.4	96.9
LR$^{\mathrm{OVA}}$	73.1	84.5	89.6	91.8	93.3	94.3	95.3	96.1	96.6	97.0
top-1 SVM$_\alpha$ / SVM$^{\mathrm{Multi}}$	74.0	85.2	89.3	91.9	93.4	94.9	95.6	95.8	96.4	96.9
top-2 SVM$_\alpha$	73.1	85.7	90.4	92.2	94.5	95.1	96.2	96.6	97.0	97.3
top-3 SVM$_\alpha$	71.6	86.3	91.1	93.2	94.7	95.7	96.4	96.6	97.1	97.2
top-4 SVM$_\alpha$	71.4	85.7	90.7	93.3	94.8	95.7	96.2	96.6	97.2	97.8
top-5 SVM$_\alpha$	70.7	85.7	90.4	93.2	94.7	95.5	96.1	96.9	97.5	97.9
top-10 SVM$_\alpha$	70.0	85.4	90.0	93.1	94.6	95.6	96.2	97.1	97.5	97.5
top-1 SVM$^1_\alpha$	74.0	86.0	90.7	92.8	94.5	95.9	96.1	96.8	97.1	97.4
top-2 SVM$^1_\alpha$	72.7	85.9	90.5	93.0	94.3	95.8	96.3	96.6	97.1	97.4
top-3 SVM$^1_\alpha$	72.2	86.1	90.8	93.1	94.6	95.4	96.3	96.6	97.3	97.7
top-4 SVM$^1_\alpha$	71.3	86.2	90.7	93.1	94.6	95.5	96.4	96.7	97.4	97.7
top-5 SVM$^1_\alpha$	71.0	86.2	90.2	92.8	94.7	95.4	96.3	97.0	97.5	97.8
top-10 SVM$^1_\alpha$	70.3	85.2	89.9	93.1	94.6	95.7	96.0	97.0	97.4	97.7
top-1 SVM$_\beta$ / SVM$^{\mathrm{Multi}}$	74.0	85.2	89.3	91.9	93.4	94.9	95.6	95.8	96.4	96.9
top-2 SVM$_\beta$	74.0	85.9	89.8	92.2	94.1	95.1	95.7	96.4	97.0	97.3
top-3 SVM$_\beta$	73.0	86.3	90.6	92.8	94.4	95.9	96.1	96.3	96.9	97.2
top-4 SVM$_\beta$	73.1	86.2	90.8	92.7	94.5	95.8	96.2	96.6	97.1	97.7
top-5 SVM$_\beta$	72.6	85.6	90.7	93.0	94.5	95.7	96.2	96.8	97.4	97.6
top-10 SVM$_\beta$	71.9	85.3	90.4	93.4	94.4	95.6	96.2	97.2	97.5	97.8
top-1 SVM$^1_\beta$	74.0	86.0	90.7	92.8	94.5	95.9	96.1	96.8	97.1	97.4
top-2 SVM$^1_\beta$	73.9	85.9	90.8	92.8	94.5	95.7	96.2	96.6	97.1	97.5
top-3 SVM$^1_\beta$	73.7	86.3	90.7	92.7	94.5	95.6	96.3	96.6	97.3	97.5
top-4 SVM$^1_\beta$	73.0	86.2	90.7	92.5	94.3	95.7	96.3	96.7	97.4	97.5
top-5 SVM$^1_\beta$	72.7	85.9	90.8	92.7	94.4	95.6	96.2	97.0	97.5	97.8
top-10 SVM$^1_\beta$	72.2	85.5	90.7	93.4	94.6	95.6	96.3	97.1	97.5	97.8
top-1 Ent / LR$^{\mathrm{Multi}}$	72.5	86.0	91.2	93.8	94.9	95.7	96.6	97.1	97.5	97.8
top-2 Ent	72.1	85.9	91.3	93.7	94.8	95.9	96.6	97.0	97.5	97.8
top-3 Ent	71.4	86.0	91.0	93.6	94.9	95.7	96.6	97.2	97.6	97.7
top-4 Ent	71.3	86.0	91.2	93.4	94.8	95.7	96.6	97.2	97.5	97.6
top-5 Ent	71.0	85.7	90.6	93.0	94.7	95.8	96.5	97.1	97.5	97.6
top-10 Ent	69.6	84.8	90.1	92.5	94.9	95.9	96.4	97.1	97.4	97.5
top-2 LR$_{\mathrm{n}}$	68.2	85.3	90.4	92.8	94.6	95.6	96.5	97.1	97.6	97.8
top-3 LR$_{\mathrm{n}}$	68.3	83.9	89.8	92.6	94.5	95.1	96.3	97.2	97.5	97.5
top-4 LR$_{\mathrm{n}}$	68.2	82.4	89.3	92.2	94.2	95.4	96.2	97.0	97.4	97.5
top-5 LR$_{\mathrm{n}}$	67.1	82.8	89.6	92.8	94.3	95.3	95.9	96.7	97.4	97.3
top-10 LR$_{\mathrm{n}}$	66.4	82.6	87.9	91.9	93.8	95.3	95.9	96.8	97.0	97.3

Table 14: Comparison of different methods in top-k accuracy (%).

Indoor 67 (VGG-16 trained on Places 205, FC6 output)

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	80.4	89.9	94.3	95.3	96.0	96.9	97.3	97.7	97.9	98.3
$\mathrm{LR}^{\mathrm{OVA}}$	82.5	92.6	95.1	96.3	97.5	97.5	98.3	98.6	98.4	98.7
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	82.6	91.4	95.0	96.1	96.0	97.9	97.5	97.8	98.3	98.9
top-2 SVM_α	83.1	92.3	95.4	96.8	97.7	98.0	98.3	98.7	98.7	98.5
top-3 SVM_α	83.0	92.5	95.4	97.5	97.8	98.3	98.5	98.7	98.7	98.9
top-4 SVM_α	82.9	92.5	95.6	97.0	97.8	98.2	98.7	98.7	98.8	99.0
top-5 SVM_α	82.5	92.3	95.7	97.0	97.7	98.1	98.5	98.7	98.8	99.0
top-10 SVM_α	80.4	92.2	95.8	96.9	97.7	98.4	98.8	99.0	99.1	99.2
top-1 SVM^1_α	83.1	92.5	95.1	97.1	97.8	98.1	98.5	98.3	98.4	99.0
top-2 SVM^1_α	83.0	93.1	95.2	97.1	97.8	98.1	98.4	98.8	98.4	99.0
top-3 SVM^1_α	82.9	93.0	95.4	97.2	97.8	98.2	98.5	98.8	98.8	98.9
top-4 SVM^1_α	82.5	92.5	95.7	97.2	97.8	98.2	98.7	98.7	98.8	99.0
top-5 SVM^1_α	82.1	92.5	95.7	97.0	97.7	98.3	98.7	98.7	99.0	99.1
top-10 SVM^1_α	80.1	92.1	95.9	96.9	97.5	98.1	98.7	99.0	99.2	99.3
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	82.6	91.4	95.0	96.1	96.0	97.9	97.5	97.8	98.3	98.9
top-2 SVM_β	83.2	92.1	95.2	96.6	97.5	98.0	98.4	98.7	98.7	98.5
top-3 SVM_β	82.9	92.5	95.1	96.9	97.9	98.3	98.4	98.7	98.4	98.7
top-4 SVM_β	82.5	92.6	95.2	97.1	97.8	98.4	98.6	98.8	98.8	98.8
top-5 SVM_β	82.2	92.8	95.4	97.1	97.8	98.3	98.6	98.8	98.8	98.9
top-10 SVM_β	81.9	92.8	95.7	97.1	97.7	98.4	98.7	99.0	99.0	99.2
top-1 SVM^1_β	83.1	92.5	95.1	97.1	97.8	98.1	98.5	98.3	98.4	99.0
top-2 SVM^1_β	82.9	92.8	95.1	97.0	97.8	98.2	98.5	98.7	98.4	99.0
top-3 SVM^1_β	82.8	92.9	95.1	97.0	97.8	98.4	98.5	98.8	98.8	99.0
top-4 SVM^1_β	82.2	92.7	95.4	97.1	97.8	98.4	98.6	98.8	98.8	99.0
top-5 SVM^1_β	82.3	92.8	95.4	97.0	97.8	98.4	98.7	98.8	98.8	99.0
top-10 SVM^1_β	81.6	92.9	95.5	97.0	97.7	98.4	98.7	98.9	99.0	99.3
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	82.4	92.9	95.4	97.1	97.2	98.1	98.7	99.1	99.1	99.1
top-2 Ent	82.4	93.2	95.7	97.2	97.2	98.1	98.7	99.0	99.0	99.2
top-3 Ent	81.9	93.3	95.7	97.1	97.5	98.1	98.8	98.9	99.0	99.1
top-4 Ent	81.9	92.8	95.7	96.9	97.6	98.2	98.7	98.9	99.0	99.2
top-5 Ent	81.9	92.5	95.8	96.9	97.5	98.4	98.7	99.0	99.0	99.2
top-10 Ent	81.2	92.5	95.7	96.9	97.6	98.4	98.7	99.0	99.1	99.2
top-2 LR_n	81.0	92.0	95.5	96.9	97.6	98.1	98.4	98.7	98.8	99.0
top-3 LR_n	81.2	92.0	95.7	96.9	97.6	98.4	98.6	98.7	99.0	99.1
top-4 LR_n	81.1	91.9	95.2	97.0	97.7	98.4	98.6	98.7	98.7	99.1
top-5 LR_n	80.9	91.9	95.1	96.5	97.5	98.1	98.6	98.9	99.1	99.0
top-10 LR_n	78.7	90.9	94.6	96.2	97.5	98.3	98.6	98.8	99.0	99.2

Table 15: Comparison of different methods in top-k accuracy (%).

Indoor 67 (VGG-16 trained on Places 205, FC7 output)

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	81.9	91.0	94.3	95.7	96.5	97.2	97.4	97.7	97.8	98.0
$\mathrm{LR}^{\mathrm{OVA}}$	82.0	91.6	94.9	96.3	97.2	97.8	98.1	98.3	98.6	98.7
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	82.5	91.7	95.4	96.9	97.3	97.8	97.6	98.5	98.8	99.1
top-2 SVM_α	82.0	91.3	95.1	96.6	97.6	97.8	98.0	98.6	98.7	99.0
top-3 SVM_α	81.6	91.4	95.1	96.8	97.7	98.2	98.6	98.7	98.8	99.0
top-4 SVM_α	80.8	91.3	95.2	96.9	97.7	98.6	98.6	98.7	98.9	99.1
top-5 SVM_α	79.9	91.3	95.0	96.5	97.7	98.4	98.6	98.8	98.8	99.0
top-10 SVM_α	78.4	91.0	95.1	96.6	97.4	98.3	98.5	98.8	99.0	99.0
top-1 SVM^1_α	82.6	91.6	95.2	96.9	97.6	98.1	98.4	98.6	98.7	99.0
top-2 SVM^1_α	82.5	91.6	95.0	96.7	97.8	98.0	97.9	98.1	98.9	98.7
top-3 SVM^1_α	81.6	91.5	95.1	96.6	97.8	98.3	98.7	98.7	98.7	99.0
top-4 SVM^1_α	81.0	91.3	95.1	96.8	97.8	98.4	98.6	98.7	99.0	99.3
top-5 SVM^1_α	80.4	91.3	95.1	96.6	97.8	98.4	98.6	98.8	98.8	99.1
top-10 SVM^1_α	78.3	91.0	95.1	96.6	97.5	98.4	98.5	98.8	99.0	99.0
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	82.5	91.7	95.4	96.9	97.3	97.8	97.6	98.5	98.8	99.1
top-2 SVM_β	82.5	91.3	95.1	96.9	97.7	97.8	98.3	98.5	98.7	98.7
top-3 SVM_β	82.1	91.4	95.0	97.1	97.7	98.1	98.2	98.6	98.7	99.0
top-4 SVM_β	82.2	91.6	95.2	96.9	97.5	98.1	98.4	98.7	98.9	98.6
top-5 SVM_β	82.2	91.3	94.9	96.9	97.6	98.2	98.4	98.7	98.9	99.0
top-10 SVM_β	81.7	91.8	95.2	96.7	97.3	98.2	98.7	98.9	99.0	99.1
top-1 SVM^1_β	82.6	91.6	95.2	96.9	97.6	98.1	98.4	98.6	98.7	99.0
top-2 SVM^1_β	82.7	91.5	95.1	96.9	97.7	98.2	97.9	98.1	98.8	98.7
top-3 SVM^1_β	82.5	91.4	95.1	96.8	97.6	98.3	98.6	98.1	98.4	98.7
top-4 SVM^1_β	82.6	91.3	95.0	97.0	97.6	98.3	98.5	98.1	98.9	98.7
top-5 SVM^1_β	82.5	91.6	95.1	96.7	97.5	98.3	98.5	98.7	98.9	98.8
top-10 SVM^1_β	81.6	91.7	95.3	96.8	97.2	98.2	98.6	98.8	99.0	99.2
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	82.3	91.4	95.2	97.2	98.0	98.4	98.7	98.8	99.0	99.1
top-2 Ent	81.9	91.9	95.1	96.9	97.8	98.4	98.7	98.7	99.0	99.3
top-3 Ent	81.4	91.5	95.4	96.7	97.6	98.3	98.7	98.8	99.0	99.2
top-4 Ent	80.8	91.6	95.4	96.6	97.7	98.3	98.7	98.8	99.0	99.1
top-5 Ent	80.3	91.3	95.0	96.6	97.7	98.2	98.7	98.8	99.0	99.0
top-10 Ent	79.2	91.0	95.1	96.7	97.6	98.3	98.7	98.7	99.0	99.0
top-2 LR_{n}	79.6	90.8	94.4	96.3	97.0	97.9	98.4	98.7	98.9	99.0
top-3 LR_{n}	79.8	90.9	95.0	96.2	97.5	98.1	98.5	98.7	99.0	99.1
top-4 LR_{n}	78.7	90.1	94.9	96.3	97.4	98.1	98.5	98.9	99.0	99.1
top-5 LR_{n}	76.4	90.2	94.3	96.0	97.3	97.8	98.4	98.6	98.7	99.0
top-10 LR_{n}	72.6	88.7	92.8	95.7	97.1	97.6	98.3	98.6	98.7	98.9

Table 16: Comparison of different methods in top-k accuracy (%).

SUN 397 (Fisher Vector kernels provided by [29])

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-2 SVM_α	48.9 ± 0.3	60.5 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-3 SVM_α	48.9 ± 0.3	60.5 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.3	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-4 SVM_α	48.8 ± 0.3	60.5 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.6 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-5 SVM_α	48.7 ± 0.3	60.5 ± 0.3	66.8 ± 0.2	70.9 ± 0.3	73.9 ± 0.3	76.2 ± 0.2	78.1 ± 0.2	79.6 ± 0.2	80.9 ± 0.2	82.1 ± 0.2
top-10 SVM_α	48.3 ± 0.4	60.4 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.1 ± 0.2	76.5 ± 0.2	78.4 ± 0.2	80.0 ± 0.2	81.3 ± 0.2	82.5 ± 0.2
top-1 SVM_α^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-2 SVM_α^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-3 SVM_α^1	48.9 ± 0.3	60.6 ± 0.2	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-4 SVM_α^1	48.8 ± 0.3	60.6 ± 0.2	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.2	78.4 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-5 SVM_α^1	48.7 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.4 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-10 SVM_α^1	48.3 ± 0.4	60.4 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.1 ± 0.2	76.5 ± 0.2	78.5 ± 0.2	80.0 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-2 SVM_β	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-3 SVM_β	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-4 SVM_β	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-5 SVM_β	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.1 ± 0.2	78.0 ± 0.2	79.5 ± 0.2	80.9 ± 0.2	82.0 ± 0.2
top-10 SVM_β	48.9 ± 0.3	60.6 ± 0.3	66.8 ± 0.2	70.8 ± 0.3	73.8 ± 0.2	76.3 ± 0.2	78.2 ± 0.2	79.8 ± 0.2	81.1 ± 0.2	82.3 ± 0.2
top-1 SVM_β^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-2 SVM_β^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-3 SVM_β^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-4 SVM_β^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.4 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-5 SVM_β^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.1 ± 0.3	74.2 ± 0.2	76.5 ± 0.3	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-10 SVM_β^1	48.9 ± 0.3	60.6 ± 0.3	66.9 ± 0.2	71.0 ± 0.3	74.2 ± 0.2	76.5 ± 0.2	78.5 ± 0.2	80.1 ± 0.2	81.4 ± 0.2	82.6 ± 0.2
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	48.5 ± 0.3	60.5 ± 0.2	66.9 ± 0.3	71.2 ± 0.3	74.3 ± 0.3	76.7 ± 0.3	78.6 ± 0.2	80.2 ± 0.2	81.6 ± 0.2	82.7 ± 0.2
top-2 Ent	48.5 ± 0.3	60.5 ± 0.2	66.9 ± 0.3	71.2 ± 0.3	74.4 ± 0.3	76.7 ± 0.3	78.6 ± 0.2	80.2 ± 0.2	81.6 ± 0.2	82.7 ± 0.2
top-3 Ent	48.5 ± 0.3	60.5 ± 0.2	66.9 ± 0.3	71.2 ± 0.3	74.4 ± 0.3	76.7 ± 0.3	78.6 ± 0.2	80.2 ± 0.2	81.6 ± 0.2	82.8 ± 0.2
top-4 Ent	48.5 ± 0.3	60.5 ± 0.2	66.9 ± 0.3	71.2 ± 0.3	74.3 ± 0.3	76.7 ± 0.3	78.6 ± 0.2	80.2 ± 0.2	81.6 ± 0.2	82.8 ± 0.2
top-5 Ent	48.4 ± 0.3	60.5 ± 0.2	66.9 ± 0.3	71.2 ± 0.3	74.3 ± 0.3	76.7 ± 0.3	78.6 ± 0.2	80.2 ± 0.3	81.6 ± 0.2	82.8 ± 0.2
top-10 Ent	48.0 ± 0.4	60.3 ± 0.3	66.8 ± 0.3	71.1 ± 0.3	74.3 ± 0.3	76.7 ± 0.2	78.7 ± 0.2	80.2 ± 0.2	81.6 ± 0.2	82.8 ± 0.2

Table 17: Comparison of different methods in top-k accuracy (%).

SUN 397 (AlexNet trained on Places 205, FC7 output, provided by [28])

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	44.1 ± 0.4	60.8 ± 0.3	69.9 ± 0.2	75.8 ± 0.1	79.8 ± 0.1	82.8 ± 0.1	85.1 ± 0.2	86.9 ± 0.2	88.3 ± 0.2	89.6 ± 0.2
$\mathrm{LR}^{\mathrm{OVA}}$	53.9 ± 0.2	69.2 ± 0.2	76.5 ± 0.2	80.9 ± 0.2	84.0 ± 0.3	86.2 ± 0.2	87.9 ± 0.2	89.2 ± 0.2	90.3 ± 0.2	91.2 ± 0.2
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	58.2 ± 0.2	71.7 ± 0.2	78.2 ± 0.1	82.3 ± 0.2	85.0 ± 0.2	87.1 ± 0.2	88.8 ± 0.2	90.0 ± 0.2	91.0 ± 0.2	91.9 ± 0.2
top-2 SVM_α	58.8 ± 0.2	72.7 ± 0.2	79.3 ± 0.2	83.3 ± 0.2	85.9 ± 0.2	87.8 ± 0.2	89.2 ± 0.2	90.3 ± 0.2	91.3 ± 0.2	92.2 ± 0.2
top-3 SVM_α	59.0 ± 0.1	73.2 ± 0.2	79.9 ± 0.2	83.8 ± 0.2	86.5 ± 0.2	88.3 ± 0.2	89.7 ± 0.2	90.9 ± 0.2	91.8 ± 0.2	92.6 ± 0.2
top-4 SVM_α	58.9 ± 0.1	73.5 ± 0.2	80.3 ± 0.2	84.2 ± 0.2	86.8 ± 0.2	88.6 ± 0.2	90.0 ± 0.2	91.1 ± 0.2	92.0 ± 0.2	92.8 ± 0.2
top-5 SVM_α	58.9 ± 0.1	73.7 ± 0.2	80.5 ± 0.2	84.4 ± 0.3	87.0 ± 0.2	88.8 ± 0.2	90.2 ± 0.2	91.3 ± 0.2	92.2 ± 0.2	93.0 ± 0.2
top-10 SVM_α	58.0 ± 0.2	73.6 ± 0.1	80.8 ± 0.1	84.8 ± 0.2	87.4 ± 0.2	89.3 ± 0.2	90.7 ± 0.2	91.8 ± 0.2	92.7 ± 0.2	93.4 ± 0.2
top-1 SVM^1_α	59.7 ± 0.1	73.8 ± 0.1	80.3 ± 0.2	84.2 ± 0.2	86.8 ± 0.2	88.6 ± 0.2	90.0 ± 0.2	91.1 ± 0.2	92.0 ± 0.2	92.8 ± 0.2
top-2 SVM^1_α	59.6 ± 0.1	73.8 ± 0.1	80.4 ± 0.2	84.3 ± 0.2	86.8 ± 0.2	88.6 ± 0.3	90.1 ± 0.2	91.2 ± 0.2	92.1 ± 0.2	92.9 ± 0.2
top-3 SVM^1_α	59.4 ± 0.1	73.8 ± 0.1	80.4 ± 0.2	84.4 ± 0.2	86.9 ± 0.3	88.7 ± 0.3	90.1 ± 0.2	91.3 ± 0.2	92.2 ± 0.2	92.9 ± 0.2
top-4 SVM^1_α	59.2 ± 0.1	73.9 ± 0.1	80.6 ± 0.2	84.5 ± 0.2	87.1 ± 0.2	88.8 ± 0.2	90.2 ± 0.2	91.4 ± 0.2	92.3 ± 0.2	93.0 ± 0.2
top-5 SVM^1_α	59.1 ± 0.1	74.0 ± 0.2	80.7 ± 0.2	84.6 ± 0.3	87.2 ± 0.2	89.0 ± 0.2	90.4 ± 0.2	91.5 ± 0.2	92.4 ± 0.2	93.1 ± 0.2
top-10 SVM^1_α	58.1 ± 0.2	73.7 ± 0.2	80.9 ± 0.2	84.9 ± 0.2	87.5 ± 0.2	89.4 ± 0.2	90.7 ± 0.2	91.8 ± 0.2	92.7 ± 0.1	93.4 ± 0.1
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	58.2 ± 0.2	71.7 ± 0.2	78.2 ± 0.1	82.3 ± 0.2	85.0 ± 0.2	87.1 ± 0.2	88.8 ± 0.2	90.0 ± 0.2	91.0 ± 0.2	91.9 ± 0.2
top-2 SVM_β	58.8 ± 0.2	72.7 ± 0.2	79.3 ± 0.2	83.2 ± 0.2	85.9 ± 0.2	87.7 ± 0.2	89.0 ± 0.2	90.3 ± 0.2	91.3 ± 0.2	92.2 ± 0.2
top-3 SVM_β	59.1 ± 0.2	73.2 ± 0.2	79.8 ± 0.2	83.8 ± 0.2	86.4 ± 0.2	88.2 ± 0.2	89.6 ± 0.2	90.8 ± 0.2	91.7 ± 0.2	92.5 ± 0.2
top-4 SVM_β	59.2 ± 0.1	73.6 ± 0.2	80.2 ± 0.2	84.2 ± 0.2	86.7 ± 0.2	88.5 ± 0.2	89.9 ± 0.2	91.1 ± 0.2	92.0 ± 0.2	92.7 ± 0.2
top-5 SVM_β	59.3 ± 0.2	73.8 ± 0.2	80.4 ± 0.3	84.4 ± 0.3	87.0 ± 0.3	88.7 ± 0.3	90.1 ± 0.2	91.2 ± 0.2	92.1 ± 0.2	92.9 ± 0.2
top-10 SVM_β	59.3 ± 0.1	74.1 ± 0.2	80.9 ± 0.2	84.9 ± 0.2	87.5 ± 0.2	89.3 ± 0.2	90.7 ± 0.2	91.7 ± 0.2	92.6 ± 0.2	93.4 ± 0.2
top-1 SVM^1_β	59.7 ± 0.1	73.8 ± 0.1	80.3 ± 0.2	84.2 ± 0.2	86.8 ± 0.2	88.6 ± 0.2	90.0 ± 0.2	91.1 ± 0.2	92.0 ± 0.2	92.8 ± 0.2
top-2 SVM^1_β	59.7 ± 0.1	73.8 ± 0.2	80.3 ± 0.2	84.3 ± 0.2	86.8 ± 0.2	88.6 ± 0.3	90.0 ± 0.2	91.2 ± 0.2	92.1 ± 0.2	92.9 ± 0.2
top-3 SVM^1_β	59.6 ± 0.1	73.8 ± 0.1	80.4 ± 0.3	84.3 ± 0.2	86.9 ± 0.3	88.7 ± 0.3	90.1 ± 0.2	91.2 ± 0.2	92.1 ± 0.2	92.9 ± 0.2
top-4 SVM^1_β	59.6 ± 0.2	73.9 ± 0.2	80.5 ± 0.2	84.4 ± 0.2	87.0 ± 0.2	88.8 ± 0.3	90.2 ± 0.2	91.3 ± 0.2	92.2 ± 0.2	93.0 ± 0.2
top-5 SVM^1_β	59.5 ± 0.1	74.1 ± 0.1	80.7 ± 0.2	84.6 ± 0.2	87.1 ± 0.2	88.9 ± 0.3	90.3 ± 0.2	91.4 ± 0.2	92.3 ± 0.2	93.1 ± 0.2
top-10 SVM^1_β	59.3 ± 0.1	74.2 ± 0.2	81.0 ± 0.2	85.0 ± 0.2	87.5 ± 0.2	89.3 ± 0.2	90.7 ± 0.2	91.8 ± 0.2	92.7 ± 0.2	93.4 ± 0.2
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	59.5 ± 0.2	74.2 ± 0.2	81.1 ± 0.2	85.1 ± 0.2	87.7 ± 0.2	89.6 ± 0.2	91.0 ± 0.1	92.1 ± 0.2	93.0 ± 0.2	93.7 ± 0.2
top-2 Ent	59.5 ± 0.2	74.3 ± 0.2	81.1 ± 0.2	85.1 ± 0.2	87.7 ± 0.2	89.6 ± 0.2	91.0 ± 0.2	92.1 ± 0.2	93.0 ± 0.2	93.7 ± 0.2
top-3 Ent	59.4 ± 0.1	74.3 ± 0.2	81.2 ± 0.2	85.2 ± 0.2	87.8 ± 0.2	89.6 ± 0.2	91.0 ± 0.2	92.1 ± 0.2	93.0 ± 0.2	93.7 ± 0.2
top-4 Ent	59.2 ± 0.2	74.3 ± 0.2	81.2 ± 0.2	85.2 ± 0.2	87.8 ± 0.2	89.7 ± 0.2	91.1 ± 0.2	92.2 ± 0.2	93.0 ± 0.2	93.7 ± 0.2
top-5 Ent	58.9 ± 0.1	74.3 ± 0.2	81.2 ± 0.2	85.2 ± 0.2	87.8 ± 0.2	89.7 ± 0.2	91.0 ± 0.2	92.1 ± 0.1	93.0 ± 0.2	93.7 ± 0.2
top-10 Ent	58.0 ± 0.2	73.7 ± 0.2	81.0 ± 0.2	85.1 ± 0.2	87.8 ± 0.2	89.7 ± 0.2	91.0 ± 0.2	92.2 ± 0.1	93.1 ± 0.2	93.8 ± 0.1

Table 18: Comparison of different methods in top-k accuracy (%).

SUN 397 (VGG-16 trained on Places 205, FC7 output)

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
$\mathrm{SVM}^{\mathrm{OVA}}$	65.4 ± 0.2	77.6 ± 0.1	83.8 ± 0.2	87.2 ± 0.1	89.6 ± 0.1	91.3 ± 0.1	92.6 ± 0.2	93.6 ± 0.2	94.3 ± 0.2	94.9 ± 0.1
$\mathrm{LR}^{\mathrm{OVA}}$	67.6 ± 0.1	81.5 ± 0.2	87.2 ± 0.2	90.4 ± 0.2	92.4 ± 0.1	93.7 ± 0.1	94.7 ± 0.1	95.4 ± 0.1	96.0 ± 0.1	96.4 ± 0.1
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	65.8 ± 0.1	79.0 ± 0.2	85.1 ± 0.2	88.4 ± 0.2	90.8 ± 0.1	92.3 ± 0.1	93.3 ± 0.1	94.2 ± 0.1	94.8 ± 0.1	95.3 ± 0.1
top-2 SVM_α	66.4 ± 0.2	80.2 ± 0.2	86.1 ± 0.1	89.4 ± 0.1	91.5 ± 0.1	92.9 ± 0.2	93.8 ± 0.2	94.6 ± 0.2	95.3 ± 0.1	95.7 ± 0.1
top-3 SVM_α	66.5 ± 0.2	80.6 ± 0.2	86.5 ± 0.1	89.7 ± 0.2	91.8 ± 0.1	93.2 ± 0.1	94.2 ± 0.1	95.0 ± 0.1	95.3 ± 0.1	95.9 ± 0.1
top-4 SVM_α	66.4 ± 0.2	80.8 ± 0.2	86.8 ± 0.1	90.0 ± 0.2	92.1 ± 0.2	93.4 ± 0.1	94.4 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.2 ± 0.1
top-5 SVM_α	66.3 ± 0.2	81.0 ± 0.2	87.0 ± 0.2	90.2 ± 0.1	92.2 ± 0.2	93.6 ± 0.1	94.5 ± 0.1	95.2 ± 0.1	95.8 ± 0.1	96.3 ± 0.1
top-10 SVM_α	64.8 ± 0.3	80.9 ± 0.1	87.2 ± 0.2	90.5 ± 0.2	92.6 ± 0.1	93.9 ± 0.1	94.9 ± 0.1	95.6 ± 0.1	96.2 ± 0.1	96.6 ± 0.1
top-1 SVM^1_α	67.4 ± 0.2	81.1 ± 0.2	86.8 ± 0.1	90.0 ± 0.1	92.0 ± 0.1	93.4 ± 0.1	94.3 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.1 ± 0.1
top-2 SVM^1_α	67.2 ± 0.2	81.1 ± 0.2	86.9 ± 0.2	90.1 ± 0.2	92.1 ± 0.1	93.4 ± 0.1	94.4 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.2 ± 0.1
top-3 SVM^1_α	67.0 ± 0.2	81.2 ± 0.2	87.0 ± 0.1	90.2 ± 0.2	92.2 ± 0.1	93.5 ± 0.1	94.5 ± 0.1	95.2 ± 0.1	95.7 ± 0.1	96.2 ± 0.0
top-4 SVM^1_α	66.8 ± 0.2	81.2 ± 0.2	87.1 ± 0.1	90.3 ± 0.2	92.3 ± 0.2	93.6 ± 0.1	94.5 ± 0.1	95.2 ± 0.1	95.8 ± 0.1	96.3 ± 0.0
top-5 SVM^1_α	66.5 ± 0.2	81.3 ± 0.2	87.2 ± 0.1	90.4 ± 0.2	92.4 ± 0.2	93.7 ± 0.1	94.6 ± 0.1	95.3 ± 0.1	95.9 ± 0.1	96.3 ± 0.0
top-10 SVM^1_α	64.9 ± 0.3	80.9 ± 0.1	87.3 ± 0.2	90.6 ± 0.2	92.6 ± 0.2	94.0 ± 0.1	94.9 ± 0.1	95.6 ± 0.1	96.2 ± 0.1	96.6 ± 0.1
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	65.8 ± 0.1	79.0 ± 0.2	85.1 ± 0.2	88.4 ± 0.2	90.8 ± 0.1	92.3 ± 0.1	93.3 ± 0.1	94.2 ± 0.1	94.8 ± 0.1	95.3 ± 0.1
top-2 SVM_β	66.4 ± 0.2	80.1 ± 0.1	86.0 ± 0.2	89.3 ± 0.2	91.4 ± 0.1	92.7 ± 0.2	93.8 ± 0.1	94.6 ± 0.2	95.3 ± 0.1	95.8 ± 0.1
top-3 SVM_β	66.8 ± 0.2	80.7 ± 0.2	86.5 ± 0.1	89.7 ± 0.2	91.7 ± 0.1	93.1 ± 0.2	94.2 ± 0.1	94.7 ± 0.1	95.3 ± 0.1	95.9 ± 0.1
top-4 SVM_β	66.9 ± 0.2	80.9 ± 0.2	86.8 ± 0.1	90.0 ± 0.2	92.0 ± 0.2	93.4 ± 0.1	94.4 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.1 ± 0.1
top-5 SVM_β	67.0 ± 0.2	81.2 ± 0.2	87.0 ± 0.1	90.2 ± 0.2	92.2 ± 0.1	93.6 ± 0.1	94.5 ± 0.1	95.2 ± 0.1	95.8 ± 0.1	96.2 ± 0.1
top-10 SVM_β	66.9 ± 0.2	81.5 ± 0.2	87.4 ± 0.2	90.7 ± 0.1	92.6 ± 0.1	93.9 ± 0.1	94.8 ± 0.1	95.5 ± 0.1	96.1 ± 0.1	96.5 ± 0.0
top-1 SVM^1_β	67.4 ± 0.2	81.1 ± 0.2	86.8 ± 0.1	90.0 ± 0.1	92.0 ± 0.1	93.4 ± 0.1	94.3 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.1 ± 0.1
top-2 SVM^1_β	67.3 ± 0.2	81.1 ± 0.2	86.8 ± 0.1	90.0 ± 0.2	92.0 ± 0.1	93.4 ± 0.1	94.4 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.2 ± 0.1
top-3 SVM^1_β	67.3 ± 0.2	81.2 ± 0.2	86.9 ± 0.1	90.1 ± 0.1	92.1 ± 0.1	93.4 ± 0.1	94.4 ± 0.1	95.1 ± 0.1	95.7 ± 0.1	96.2 ± 0.1
top-4 SVM^1_β	67.2 ± 0.2	81.3 ± 0.2	87.0 ± 0.1	90.2 ± 0.2	92.2 ± 0.1	93.5 ± 0.1	94.5 ± 0.1	95.2 ± 0.1	95.8 ± 0.1	96.2 ± 0.1
top-5 SVM^1_β	67.2 ± 0.2	81.4 ± 0.2	87.2 ± 0.2	90.3 ± 0.1	92.3 ± 0.1	93.6 ± 0.1	94.6 ± 0.1	95.3 ± 0.1	95.9 ± 0.1	96.3 ± 0.1
top-10 SVM^1_β	66.9 ± 0.2	81.5 ± 0.2	87.5 ± 0.2	90.7 ± 0.1	92.6 ± 0.1	93.9 ± 0.1	94.9 ± 0.1	95.5 ± 0.1	96.1 ± 0.1	96.5 ± 0.1
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	67.5 ± 0.1	81.7 ± 0.2	87.7 ± 0.2	90.9 ± 0.2	92.9 ± 0.1	94.2 ± 0.1	95.1 ± 0.1	95.8 ± 0.1	96.4 ± 0.1	96.8 ± 0.1
top-2 Ent	67.4 ± 0.2	81.8 ± 0.2	87.7 ± 0.2	90.9 ± 0.1	92.9 ± 0.1	94.2 ± 0.1	95.1 ± 0.1	95.8 ± 0.1	96.4 ± 0.1	96.8 ± 0.1
top-3 Ent	67.2 ± 0.2	81.8 ± 0.2	87.7 ± 0.2	90.9 ± 0.1	92.9 ± 0.1	94.2 ± 0.1	95.1 ± 0.1	95.8 ± 0.1	96.4 ± 0.1	96.8 ± 0.1
top-4 Ent	66.9 ± 0.2	81.7 ± 0.2	87.7 ± 0.2	91.0 ± 0.1	92.9 ± 0.1	94.2 ± 0.1	95.2 ± 0.1	95.9 ± 0.1	96.4 ± 0.1	96.8 ± 0.1
top-5 Ent	66.6 ± 0.3	81.6 ± 0.2	87.7 ± 0.2	91.0 ± 0.1	92.9 ± 0.1	94.2 ± 0.1	95.2 ± 0.1	95.9 ± 0.1	96.4 ± 0.1	96.8 ± 0.1
top-10 Ent	65.2 ± 0.3	81.0 ± 0.2	87.4 ± 0.1	90.8 ± 0.2	92.8 ± 0.1	94.2 ± 0.1	95.2 ± 0.1	95.9 ± 0.1	96.4 ± 0.1	96.8 ± 0.1

Table 19: Comparison of different methods in top-k accuracy (%).

Places 205 (AlexNet trained on Places 205, FC7 output, provided by [28])

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	50.6	64.5	71.4	75.5	78.5	80.7	82.5	84.0	85.1	86.2
top-2 SVM_α	51.1	65.7	73.1	77.5	80.7	83.1	84.9	86.3	87.5	88.4
top-3 SVM_α	51.3	66.2	73.2	77.9	81.3	83.6	85.6	87.1	88.3	89.4
top-4 SVM_α	51.2	66.3	73.5	78.1	81.4	83.7	85.7	87.3	88.7	89.7
top-5 SVM_α	50.8	66.2	73.7	78.2	81.4	83.9	85.8	87.5	88.9	90.0
top-10 SVM_α	50.1	65.8	73.4	78.3	81.6	84.0	86.0	87.6	89.0	90.1
top-1 SVM^1_α	51.8	66.4	73.5	78.1	81.4	83.9	85.7	87.4	88.7	89.8
top-2 SVM^1_α	51.5	66.4	73.5	78.1	81.4	83.8	85.7	87.3	88.6	89.7
top-3 SVM^1_α	51.5	66.4	73.5	78.1	81.4	83.8	85.7	87.4	88.7	89.8
top-4 SVM^1_α	51.3	66.4	73.7	78.1	81.5	83.8	85.9	87.5	88.8	89.9
top-5 SVM^1_α	50.9	66.2	73.6	78.2	81.5	83.9	85.9	87.5	88.9	90.0
top-10 SVM^1_α	50.2	65.8	73.4	78.3	81.7	84.0	86.0	87.6	89.0	90.2
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	50.6	64.5	71.4	75.5	78.5	80.7	82.5	84.0	85.1	86.2
top-2 SVM_β	51.0	65.6	72.7	77.4	80.6	82.9	84.9	86.1	87.4	88.4
top-3 SVM_β	51.3	66.0	73.4	77.9	81.3	83.6	85.6	87.1	88.3	89.3
top-4 SVM_β	51.4	66.2	73.6	78.0	81.4	83.8	85.7	87.3	88.7	89.8
top-5 SVM_β	51.3	66.3	73.7	78.3	81.4	83.8	85.8	87.5	88.8	89.9
top-10 SVM_β	50.9	66.1	73.5	78.4	81.7	84.0	86.0	87.6	89.0	90.2
top-1 SVM^1_β	51.8	66.4	73.5	78.1	81.4	83.9	85.7	87.4	88.7	89.8
top-2 SVM^1_β	51.7	66.4	73.5	78.0	81.4	83.9	85.7	87.4	88.7	89.8
top-3 SVM^1_β	51.5	66.3	73.7	78.2	81.4	83.8	85.8	87.4	88.8	89.8
top-4 SVM^1_β	51.5	66.4	73.7	78.3	81.5	83.8	85.8	87.4	88.8	89.9
top-5 SVM^1_β	51.3	66.4	73.7	78.3	81.4	83.8	85.8	87.5	88.8	90.0
top-10 SVM^1_β	51.0	66.1	73.5	78.3	81.7	84.0	86.0	87.6	89.0	90.2
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	51.1	66.1	73.5	78.1	81.5	84.1	85.9	87.6	88.9	90.0
top-2 Ent	51.0	66.1	73.4	78.1	81.5	84.0	85.8	87.6	88.9	90.0
top-3 Ent	50.9	66.1	73.4	78.1	81.5	83.9	85.8	87.5	88.9	89.9
top-4 Ent	50.7	66.0	73.3	78.0	81.5	83.9	85.7	87.5	88.9	89.9
top-5 Ent	50.3	65.8	73.3	77.8	81.3	83.9	85.7	87.3	88.8	89.9
top-10 Ent	48.9	64.9	72.7	77.5	81.0	83.7	85.6	87.2	88.7	89.8

Table 20: Comparison of different methods in top-k accuracy (%).

Places 205 (VGG-16 trained on Places 205, FC7 output)

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	58.4	72.5	78.7	82.3	84.7	86.4	87.5	88.4	89.2	89.9
top-2 SVM_α	58.6	73.4	80.1	84.1	86.6	88.5	89.9	90.8	91.6	92.2
top-3 SVM_α	58.6	73.7	80.3	84.5	87.3	89.3	90.8	91.8	92.6	93.3
top-4 SVM_α	58.6	73.8	80.5	84.6	87.4	89.5	91.0	92.1	93.0	93.8
top-5 SVM_α	58.4	73.8	80.5	84.5	87.4	89.5	91.1	92.3	93.2	94.0
top-10 SVM_α	58.0	73.2	80.4	84.6	87.4	89.6	91.2	92.5	93.5	94.3
top-1 SVM^1_α	59.2	74.2	80.5	84.6	87.3	89.6	91.1	92.2	93.2	93.8
top-2 SVM^1_α	59.0	73.9	80.4	84.6	87.5	89.6	91.1	92.2	93.1	93.7
top-3 SVM^1_α	58.9	74.0	80.5	84.6	87.6	89.6	91.1	92.3	93.2	93.9
top-4 SVM^1_α	58.7	73.8	80.5	84.6	87.4	89.6	91.1	92.3	93.2	94.1
top-5 SVM^1_α	58.5	73.8	80.5	84.5	87.5	89.5	91.2	92.3	93.2	94.1
top-10 SVM^1_α	58.0	73.2	80.4	84.5	87.5	89.6	91.3	92.5	93.5	94.3
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	58.4	72.5	78.7	82.3	84.7	86.4	87.5	88.4	89.2	89.9
top-2 SVM_β	58.6	73.6	80.0	83.9	86.4	88.3	89.6	90.6	91.4	92.0
top-3 SVM_β	58.8	73.9	80.4	84.5	87.2	89.2	90.7	91.7	92.6	93.2
top-4 SVM_β	58.8	73.9	80.6	84.6	87.4	89.6	91.0	92.1	93.0	93.7
top-5 SVM_β	58.9	74.0	80.6	84.7	87.5	89.6	91.0	92.2	93.2	94.0
top-10 SVM_β	58.7	74.0	80.7	84.8	87.6	89.7	91.3	92.4	93.5	94.2
top-1 SVM^1_β	59.2	74.2	80.5	84.6	87.3	89.6	91.1	92.2	93.2	93.8
top-2 SVM^1_β	59.0	74.2	80.5	84.6	87.4	89.6	91.0	92.2	93.1	93.7
top-3 SVM^1_β	59.0	74.1	80.6	84.7	87.5	89.7	91.1	92.2	93.1	93.8
top-4 SVM^1_β	58.9	74.0	80.7	84.7	87.5	89.7	91.1	92.3	93.2	94.0
top-5 SVM^1_β	58.9	74.0	80.7	84.7	87.5	89.7	91.1	92.3	93.3	94.2
top-10 SVM^1_β	58.7	74.0	80.7	84.8	87.6	89.7	91.3	92.4	93.5	94.3
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	59.0	73.9	80.6	84.8	87.6	89.7	91.3	92.5	93.5	94.3
top-2 Ent	58.9	73.8	80.6	84.7	87.6	89.6	91.2	92.4	93.5	94.3
top-3 Ent	58.7	73.8	80.6	84.8	87.6	89.6	91.2	92.5	93.5	94.2
top-4 Ent	58.5	73.6	80.5	84.6	87.5	89.6	91.2	92.4	93.4	94.2
top-5 Ent	58.1	73.5	80.4	84.5	87.4	89.6	91.2	92.4	93.4	94.2
top-10 Ent	57.0	72.8	80.0	84.2	87.2	89.3	91.0	92.3	93.4	94.1
top-2 LR_n	57.7	73.4	80.1	84.2	87.2	89.4	90.8	92.0	93.2	94.0
top-3 LR_n	56.8	72.8	80.0	84.1	87.1	89.3	90.9	92.2	93.3	94.2
top-4 LR_n	55.2	71.6	79.2	83.6	86.9	89.2	90.8	92.0	93.0	94.1
top-5 LR_n	54.2	71.2	79.0	83.5	86.9	89.1	90.8	92.0	93.1	94.0
top-10 LR_n	51.1	68.4	76.9	82.2	85.8	88.3	90.2	91.7	92.9	93.8

Table 21: Comparison of different methods in top-k accuracy (%).

ILSVRC 2012 (AlexNet trained on ImageNet, FC7 output, provided by [28])

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	56.6	67.3	72.4	75.4	77.7	79.3	80.8	82.0	82.9	83.7
top-2 SVM_α	56.6	68.1	73.2	76.4	78.6	80.3	81.7	82.8	83.7	84.6
top-3 SVM_α	56.6	68.3	73.6	76.8	79.0	80.7	82.1	83.2	84.2	85.0
top-4 SVM_α	56.5	68.4	73.8	77.1	79.3	81.1	82.4	83.5	84.5	85.3
top-5 SVM_α	56.5	68.4	73.8	77.2	79.4	81.1	82.5	83.7	84.6	85.4
top-10 SVM_α	55.9	68.2	73.8	77.3	79.8	81.4	82.8	84.0	85.0	85.8
top-1 SVM_α^1	57.1	68.3	73.5	76.7	78.9	80.6	82.0	83.1	84.1	84.9
top-2 SVM_α^1	56.7	68.4	73.6	76.8	79.0	80.8	82.1	83.2	84.2	85.0
top-3 SVM_α^1	56.6	68.4	73.8	77.0	79.2	80.9	82.4	83.5	84.4	85.2
top-4 SVM_α^1	56.6	68.5	73.9	77.1	79.4	81.1	82.5	83.7	84.6	85.3
top-5 SVM_α^1	56.5	68.5	73.9	77.3	79.5	81.2	82.6	83.7	84.6	85.5
top-10 SVM_α^1	55.9	68.2	73.8	77.4	79.7	81.4	82.8	84.0	85.0	85.8
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	56.6	67.3	72.4	75.4	77.7	79.3	80.8	82.0	82.9	83.7
top-2 SVM_β	56.9	68.0	73.2	76.2	78.5	80.2	81.6	82.7	83.6	84.4
top-3 SVM_β	57.0	68.3	73.5	76.7	78.9	80.6	82.0	83.1	84.0	84.8
top-4 SVM_β	57.0	68.4	73.6	76.9	79.1	80.8	82.2	83.4	84.3	85.1
top-5 SVM_β	57.1	68.4	73.7	76.9	79.3	81.0	82.4	83.5	84.5	85.2
top-10 SVM_β	56.9	68.4	73.9	77.3	79.5	81.2	82.7	83.8	84.8	85.6
top-1 SVM_β^1	57.1	68.3	73.5	76.7	78.9	80.6	82.0	83.1	84.1	84.9
top-2 SVM_β^1	57.1	68.3	73.6	76.7	78.9	80.6	82.0	83.2	84.2	84.9
top-3 SVM_β^1	54.6	66.1	71.5	74.7	77.1	78.7	80.2	81.3	82.3	83.1
top-4 SVM_β^1	57.1	68.5	73.8	77.0	79.2	80.9	82.3	83.5	84.5	85.2
top-5 SVM_β^1	57.1	68.5	73.8	77.0	79.3	81.0	82.5	83.6	84.6	85.3
top-10 SVM_β^1	56.9	68.5	73.9	77.3	79.6	81.2	82.7	83.8	84.8	85.7
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	55.8	67.4	73.1	76.6	79.0	80.8	82.2	83.4	84.4	85.3
top-2 Ent	55.6	67.4	73.0	76.5	79.0	80.8	82.2	83.4	84.4	85.2
top-3 Ent	55.5	67.4	73.0	76.5	78.9	80.7	82.2	83.4	84.4	85.2
top-4 Ent	55.4	67.3	73.0	76.5	78.9	80.7	82.1	83.4	84.4	85.2
top-5 Ent	55.2	67.2	72.9	76.5	78.9	80.7	82.1	83.4	84.3	85.2
top-10 Ent	54.8	66.9	72.7	76.4	78.8	80.6	82.1	83.3	84.3	85.2

Table 22: Comparison of different methods in top-k accuracy (%).

ILSVRC 2012 (VGG-16 trained on ImageNet, FC7 output)

Method	Top-1	Top-2	Top-3	Top-4	Top-5	Top-6	Top-7	Top-8	Top-9	Top-10
top-1 SVM_α / $\mathrm{SVM}^{\mathrm{Multi}}$	68.3	78.6	82.9	85.4	87.0	88.2	89.2	89.9	90.6	91.1
top-2 SVM_α	68.3	79.3	83.7	86.3	87.8	89.0	89.9	90.6	91.3	91.8
top-3 SVM_α	68.2	79.5	84.0	86.5	88.1	89.3	90.2	91.0	91.6	92.1
top-4 SVM_α	68.0	79.6	84.1	86.6	88.3	89.5	90.4	91.2	91.8	92.3
top-5 SVM_α	67.8	79.5	84.1	86.6	88.2	89.5	90.5	91.2	91.9	92.4
top-10 SVM_α	67.0	79.0	83.8	86.5	88.3	89.6	90.6	91.4	92.1	92.6
top-1 SVM^1_α	68.7	79.5	83.9	86.4	88.0	89.3	90.2	90.9	91.6	92.1
top-2 SVM^1_α	68.5	79.6	84.0	86.5	88.1	89.3	90.2	91.0	91.7	92.2
top-3 SVM^1_α	68.2	79.6	84.1	86.6	88.2	89.4	90.3	91.1	91.8	92.3
top-4 SVM^1_α	68.0	79.7	84.2	86.7	88.4	89.6	90.5	91.2	91.8	92.3
top-5 SVM^1_α	67.9	79.6	84.1	86.6	88.4	89.6	90.5	91.3	92.0	92.5
top-10 SVM^1_α	67.1	79.1	83.8	86.5	88.3	89.6	90.6	91.4	92.1	92.6
top-1 SVM_β / $\mathrm{SVM}^{\mathrm{Multi}}$	68.3	78.6	82.9	85.4	87.0	88.2	89.2	89.9	90.6	91.1
top-2 SVM_β	68.6	79.2	83.6	86.1	87.6	88.9	89.8	90.6	91.2	91.7
top-3 SVM_β	68.5	79.5	83.9	86.4	88.0	89.2	90.1	90.8	91.5	91.9
top-4 SVM_β	68.4	79.5	84.0	86.6	88.2	89.4	90.3	91.0	91.6	92.1
top-5 SVM_β	68.4	79.6	84.1	86.6	88.2	89.5	90.4	91.1	91.7	92.2
top-10 SVM_β	68.0	79.5	84.0	86.6	88.3	89.6	90.6	91.3	92.0	92.5
top-1 SVM^1_β	68.7	79.5	83.9	86.4	88.0	89.3	90.2	90.9	91.6	92.1
top-2 SVM^1_β	68.7	79.5	84.0	86.5	88.1	89.3	90.2	91.0	91.6	92.1
top-3 SVM^1_β	68.6	79.5	84.1	86.6	88.1	89.4	90.3	91.0	91.6	92.2
top-4 SVM^1_β	68.5	79.6	84.1	86.6	88.3	89.5	90.4	91.1	91.7	92.2
top-5 SVM^1_β	68.4	79.6	84.1	86.6	88.3	89.5	90.4	91.2	91.8	92.3
top-10 SVM^1_β	68.0	79.5	84.1	86.6	88.3	89.6	90.6	91.4	92.0	92.5
top-1 Ent / $\mathrm{LR}^{\mathrm{Multi}}$	67.2	78.5	83.2	85.9	87.7	89.1	90.1	91.0	91.7	92.2
top-2 Ent	67.1	78.4	83.2	85.9	87.8	89.1	90.1	91.0	91.7	92.2
top-3 Ent	66.8	78.4	83.1	85.9	87.8	89.1	90.1	91.0	91.7	92.2
top-4 Ent	66.7	78.3	83.1	85.8	87.8	89.1	90.1	91.0	91.7	92.2
top-5 Ent	66.5	78.2	83.0	85.8	87.7	89.1	90.1	91.0	91.6	92.2
top-10 Ent	65.8	77.8	82.8	85.7	87.6	89.0	90.0	90.9	91.6	92.1
top-2 $\mathrm{LR_n}$	66.6	78.1	83.0	85.7	87.6	89.0	90.0	90.8	91.6	92.1
top-3 $\mathrm{LR_n}$	65.9	77.8	82.8	85.7	87.5	89.0	89.9	90.8	91.5	92.1
top-4 $\mathrm{LR_n}$	66.0	77.7	82.7	85.6	87.5	88.9	89.9	90.8	91.5	92.1
top-5 $\mathrm{LR_n}$	65.0	77.1	82.3	85.2	87.3	88.7	89.8	90.6	91.3	91.9
top-10 $\mathrm{LR_n}$	64.6	76.7	82.0	85.0	87.0	88.5	89.6	90.5	91.2	91.8

Table 23: Comparison of different methods in top-k accuracy (%).