Optimizing Partial Area Under the Top-k Curve: Theory and Practice

Zitai Wang, Qianqian Xu, Senior Member, IEEE, Zhiyong Yang, Yuan He, Xiaochun Cao, Senior Member, IEEE, and Qingming Huang, Fellow, IEEE

Abstract—Top-\(k\) error has become a popular metric for large-scale classification benchmarks due to the inevitable semantic ambiguity among classes. Existing literature on top-\(k\) optimization generally focuses on the optimization method of the top-\(k\) objective, while ignoring the limitations of the metric itself. In this paper, we point out that the top-\(k\) objective lacks enough discrimination such that the induced predictions may give a totally irrelevant label a top rank. To fix this issue, we develop a novel metric named partial Area Under the top-\(k\) Curve (AUTKC). Theoretical analysis shows that AUTKC has a better discrimination ability, and its Bayes optimal score function could give a correct top-\(k\) ranking with respect to the conditional probability. This shows that AUTKC does not allow irrelevant labels to appear in the top list. Furthermore, we present an empirical surrogate risk minimization framework to optimize the proposed metric. Theoretically, we present (1) a sufficient condition for Fisher consistency of the Bayes optimal score function; (2) a generalization upper bound which is insensitive to the number of classes under a simple hyperparameter setting. Finally, the experimental results on four benchmark datasets validate the effectiveness of our proposed framework.

Index Terms—Machine learning, label ambiguity, Top-k error, AUTKC optimization

1 INTRODUCTION

Top-1 error, which checks whether the annotated label has the highest score, is a ubiquitous measure in classification tasks. Implicitly, this metric assumes that (a) the annotated label is exactly the ground-truth, i.e., (no label noise [2]), and (b) the ground-truth label is easy to discriminate from the others, i.e., (no label ambiguity [3], [4]). To ensure (a), we can either improve the data quality to clean the possible noises, or improve the quality of the training model to resist the adverse effect of noises. Different from the noises, label ambiguity is an intrinsic property of the label semantic. Consequently, it cannot be eliminated by simply improving the quality of data once the label space is fixed. In fact, (b) does not necessarily hold as the number of classes in benchmark datasets grows rapidly [1], [5]. As shown in Fig. 1, when facing numerous classes, there are two major issues that make the ground-truth label ambiguous. One is that the semantic similarity among classes inevitably emerges as the size of label space grows larger, leading to high error rates of the top-1 predicted labels even for humans [4]. The other is that, when the number of classes is larger, each instance becomes more likely to be relevant with multiple classes, making each relevant class a reasonable prediction. Dealing with label ambiguity is thus a harder problem than label noises. In this sense, our goal is to seek a better solution against label ambiguity in this paper.

Historically, there exist two solutions against label ambiguity. One solution is to perform multi-label learning [3], [6], where each instance is annotated with multiple labels to cover ambiguous labels as much as possible. The other one is to relax the top-1 error and perform top-\(k\) optimization [4], [7], [8], [9], where ambiguous labels are allowed to be ranked higher than the ground-truth. In this way, one can
approximate the multi-label setting with a single-label dataset. Although the first solution is natural and straightforward, multi-label annotation is generally intractable for large-scale datasets. In practice, single-label datasets [1], [5] enjoy much higher scalability than multi-label datasets [10], [11] in terms of both the number of classes (hundreds versus dozens) and the number of instances (million versus thousands). Considering this fact, this paper focuses on the second solution, with a special interest in directly optimizing the top-k measure.

In this direction, existing literature has proposed a series of methods to optimize the top-k objective [4], [7], [8], [9]. However, the limitations of the metric itself have not been well understood. In this paper, we argue that the top-k error lacks enough discrimination, leaving some fatal errors free from penalty. Specifically, when the top-k error allows ambiguous labels to be ranked higher, it also relaxes the punishment on irrelevant labels. As shown in Fig. 1, even if an irrelevant label Dog is ranked higher than the ground-truth label Mountain and the ambiguous labels Valley and Sky, the top-k error can still be zero, as long as Mountain appears in the predicted top-k list. To fix this issue, this paper takes a further step toward a more discriminating metric.

First of all, in Section 3.2, the analysis suggests that looking at the performance at a single fixed k is insufficient to ensure the true prediction quality. In view of this, a natural solution is to evaluate the performance from a comprehensive perspective. To this end, we design a new measure named Partial Area Under the Top-k Curve (AUTKC) in Section 3.3, as an analogy with AUC [12]. The comparison in Section 3.4 reveals that AUTKC is strictly consistent and more discriminating than the top-k error, which helps it overcome the limitations of the top-k measure.

Considering its advantages, it is appealing to design learning algorithms that optimize AUTKC efficiently. To this end, we have the following abstract formulation of the optimization problem, whose details are shown in Eq. (9).

$$\min_{f, z} \mathbb{E}_{x} \left[ \sum_{k} \ell_k(f; x) \right], \tag{1}$$

where the loss $\ell_k$ takes the model prediction $f(x)$ as the input. As presented in Section 4, the main challenges are two-fold: (C1) $\ell_k$ is discrete and non-differentiable, making the objective difficult to optimize; (C2) the data distribution is generally unavailable, making it impossible to calculate the expectation.

Targeting (C1), one common strategy is to replace $\ell_k$ with a differentiable surrogate loss. But, what kind of surrogate loss should we select? The basic requirement is Fisher consistency, that is, we should recover the Bayes optimal solution of Eq. (1). In view of this, in Section 5.1, we present a necessary and sufficient property for the Bayes-optimal function induced by Eq. (1). It suggests that the Bayes optimal score function must preserve the ranking of the underlying top-K labels w.r.t. the conditional probability $P(y = i|x)$. Since the conditional probability for the irrelevant labels cannot surpass the relevant ones, the optimal solution naturally removes the irrelevant classes from the top list. On top of this, we develop a sufficient condition for the Fisher consistency for AUTKC in Section 5.2. Based on this condition, we find that common surrogate losses are consistent with AUTKC as long as the model outputs are bounded, including the square loss, the exponential loss and the logit loss. However, the hinge loss, which is the standard surrogate loss of the top-k optimization methods, is inconsistent.

In Section 6, we further construct an empirical optimization framework for AUTKC to relieve (C2). Specifically, we turn to minimize the unbiased empirical estimation of the surrogate loss over the training set, thus avoiding the direct calculation of the expectation. A basic question is whether the performance on the training set can generalize well to the unseen data. To answer this question in the context of label ambiguity, we construct the upper bound of the generalization error in terms of the number of classes C and deep neural networks. The result shows that we can obtain a generalization bound that is insensitive to C.

Finally, we conduct the experiments on four benchmark datasets in Section 7, where the number of classes ranges from 10 to 365.

In summary, the contribution of this paper is three-fold:

- **New measure**: We provide a novel performance measure named AUTKC to handle the label ambiguity in large-scale classification.
- **Theoretical guarantee**: The proposed framework for AUTKC optimization is supported by consistent surrogate losses and a generalization bound insensitive to the number of classes.
- **Empirical validation**: The empirical results not only show the superiority of the proposed framework, but also validate the theoretical results.

## 2 RELATED WORK

### 2.1 TOP-k Optimization

The early literature on top-k optimization focuses on the binary scenarios where only the top-ranked instances are of interest, such as retrieval (relevant/irrelevant) [13] and recommend (like/dislike) [14]. These methods minimize a pairwise ranking loss that punishes more on the errors related to the top-ranked instances [15], [16], [17].

As the scale of classes grows rapidly in modern benchmark datasets [1], [5], the semantic overlap between classes...
inevitably emerges. In this light, the top-\(k\) error has been extended to multiclass classification [11, 18], [19], [20]. Similar to the binary case, this metric evaluates the model performance on the top-ranked items. The difference lies in it measuring the ranking of the ground-truth label, rather than that of instance. The early work in this direction adopts an alternative approach to minimize the top-\(k\) error. For example, [21] models the classes missing from the annotation by the structured learning technique with latent variables. [22] analyzes the problem from the perspective of submodular reward functions. [23] proposes a probabilistic method explicitly modeling the total number of objects in an image. Besides, [24], [25] use metric learning to find potential labels, especially when the number of classes is dynamic.

Recently, more efforts focus on directly minimizing the top-\(k\) error. To this end, [7] relaxes the multiclass hinge loss defined on \(\mathcal{Y}\), where \(\mathcal{Y}\) represents the input space and \(\mathcal{Y}\) is the label space, that is the ground-truth label, consistent [4]. In other words, minimizing these losses does not guarantee an ideal top-\(k\) performance. To this end, [4] analyzes the calibration property of some top-\(k\) loss functions truncated from the multiclass hinge loss and the cross-entropy loss. The theoretical results are further extended to multi-label classification [8]. Most recently, [9] points out that the condition for top-\(k\) calibration proposed by [4] is invalid when ties exist in the prediction. To fix this issue, [9] develops a necessary and sufficient property for top-\(k\) optimality named top-\(k\) preserving. Based on this property, a top-\(k\) consistent hinge-like loss is finally established.

Compared with existing efforts on the optimization method of the top-\(k\) objective, this paper focuses on the limitations of the metric itself, whose details are shown in Section 3.2. To solve this problem, we propose a novel metric named AUTKC. Moreover, an efficient framework for AUTKC optimization is also established.

### 2.2 Label Ranking

The setting we discuss is naturally associated with the label ranking problem, where the predictions are also evaluated by the ranking of labels. Specifically, given the label space \(\mathcal{Y}\), label ranking aims to assign each instance \(x\) with the correct ranking of all the labels, that is, a complete/partial, transitive, and asymmetric relation \(\succ\) defined on \(\mathcal{Y}\), where \(i \succ j\) means that the label \(i\) precedes the label \(j\) in the ranking associated with \(x\). According to the taxonomy established by [26], label ranking methods can be divided into four categories: the ones decomposing the original problem to multiple simple objectives such as pointwise function [27], [28] and pairwise ranking loss [29], [30]; probabilistic methods including tree-based model [31], [32], [33], [34], [35], Gaussian mixture model [36] and structured learning [37]; the ones based on similarity [38], [39], [40]; and the rule-based ones [41], [42]. Please refer to the surveys for more details [26], [43].

Compared with label ranking, top-\(k\) optimization follows the setting of classification, where each instance \(x\) is assigned with a single label \(y \in \mathcal{Y}\), i.e., the ground-truth label. As a result, the common measures used in label ranking become invalid, such as Spearman’s rank [44] and Kendall’s tau [45]. And top-\(k\) optimization only considers the ranking associated with the ground-truth label, i.e., ensuring \(y \succ z, j \in \mathcal{Y}\). Concretely, top-\(k\) optimization requires that \(|\{j \in \mathcal{Y} : y \succ z\}| < k\), where \(|\cdot|\) represents the cardinality of a set. Besides, our analysis in Section 5.1 shows that optimizing AUTKC encourages the prediction to preserve the ranking of the underlying top-\(K\) labels. In this sense, AUTKC optimization essentially aims to recover the ranking of partial labels under a restricted condition where only the ground-truth label is available.

### 2.3 AUC Optimization

Our proposed framework is also related to AUC optimization, where the optimization objective is also a pairwise ranking loss. Specifically, AUC, the Area Under the receiver operating characteristic (ROC) Curve, measures the probability that the positive instances are ranked higher than the negative ones, with the assumption that the possibility to observe ties in the comparisons equals zero [12]. Being insensitive to the label distribution, AUC has become a popular metric in the class-imbalanced applications such as disease prediction [46] and rare event detection [47], [48], [49].

Traditional research on AUC optimization targets the whole area under the ROC curve. At the early stage, the major studies in this direction focus on the off-line setting [50], [51], [52], [53], [54]. As the scale of datasets increases rapidly, it becomes infeasible to optimize the pairwise loss in a full-batch manner. For this reason, more studies explore the extension of AUC optimization methods in the online setting [55], [56], [57], [58], [59], [60], [61], [62]. Besides, the theoretical guarantee of the learning framework is another appealing topic, including generalization bound [63], [64], [65], [66], [67] and consistency property [68], [69]. Recently, more literature focuses on the partial area under the ROC curve since only the order of the top-ranked instances are of interest in many practical applications [70], [71], [72], [73]. In this sense, partial AUC optimization is related to top-\(k\) optimization in binary scenarios.

Compared with partial AUC optimization, our proposed framework focuses on the ranking of multiple labels. Moreover, we assume that ties exist in the scores. These two differences will lead to totally different theoretical results, whose details are shown in Sections 5 and 6.

### 3 Preliminaries

In this section, we first describe the notations of the top-\(k\) error in Section 3.1 and some important notations are summarized in Table 1. Then, a detailed analysis on the limitations of this metric will be presented in Section 3.2. Motivated by this, we present the definition of AUTKC in Section 3.3 and further reveal its advantages over the top-\(k\) error in Section 3.4.

#### 3.1 Standard TOP-k Metric

In multi-class classification, one generally assumes that the samples are drawn i.i.d. from a product space \(\mathcal{Z} = \mathcal{X} \times \mathcal{Y}\), where \(\mathcal{X}\) is the input space and \(\mathcal{Y} = \{1, \ldots, C\}\) is the label space. Let \(D\) be the joint distribution defined on \(\mathcal{Z}\), and
TABLE 1
Some Important Notations Used in This Paper

| Notation | Description |
|----------|-------------|
| $C$      | the number of classes |
| $K$      | the maximum number of relevant labels |
| $\mathcal{X}, \mathcal{Y}$ | the input space and the label space |
| $\mathcal{D}$ | the joint distribution defined on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ |
| $f$      | the score function mapping $\mathcal{X}$ to $\mathbb{R}^C$ |
| $\mathcal{F}$ | the set of score functions |
| $L$      | the loss function mapping $\mathcal{Y} \times \mathcal{R}^C$ to $\mathbb{R}_+$ |
| $\pi_s(y)$ | the index of $s$, when sorting $\mathbf{s} \in \mathcal{R}^C$ descendingly |
| $\mathcal{S}$ | the $k$-th greatest entry of $\mathbf{s} \in \mathcal{R}^C$ |
| $\mathcal{S}^l_o$ | the $k$-th greatest element of $\mathbf{s} \in \mathcal{R}^C$ except $s_y$ |
| $\ell_{0-1, \mathcal{F}}$ | the 0-1 loss and its surrogate loss |
| $\mathbb{R}_K^l(f)$ | the original AUTKC risk |
| $\mathbb{R}_K^l(f, \mathcal{F})$ | the optimal AUTKC risk |
| $\mathbb{R}_K^l(f)$ | top-$K$ ranking-preserving property (Definition 5) |
| $\mathbb{R}_K^l(f, \mathcal{S})$ | the surrogate AUTKC risk |
| $\mathcal{D}$ | the dataset sampled from $\mathcal{D}$ |
| $\mathbb{R}_K^l(f, \mathcal{S}, \mathcal{F})$ | the empirical AUTKC risk |
| $\mathbb{R}_K^l(f, \mathcal{S}, \mathcal{F}, \mathcal{E}^{(\Theta, \epsilon, d)})$ | the empirical Gaussian Complexity |
| $\Phi(\mathcal{S}, \delta)$ | the generalization error of AUTKC |
| $\mathcal{E}^{(\Theta, \epsilon, d)}$ | covering number defined on the metric space $(\Theta, d)$ |
| $\mathcal{F}_{\beta, \epsilon}$ | the set of neural networks we discuss in Section 6.2.2 |
| vec$(\cdot)$ | the vectorization operation |
| mat$(\cdot)$ | the operator matrix |

$\eta(\mathbf{z}) = P[y|\mathbf{z}]$ denotes the conditional probability of instance $\mathbf{z}$ according to distribution $\mathcal{D}$. Assume that there exist no ties in conditional probability $\eta(\mathbf{z}) \in \mathcal{R}^C$. In other words, we have $\eta(\mathbf{z})_1 > \cdots > \eta(\mathbf{z})_C$, where $\eta(\mathbf{z})_k$ is the $k$-th greatest element of $\eta(\mathbf{z})$. Then, our task is to learn a score function $f: \mathcal{X} \rightarrow \mathcal{R}^C$ to estimate the conditional probability for each class. Generally, a loss function $L: \mathcal{Y} \times \mathcal{R}^C \rightarrow \mathbb{R}_+$ is used to measure the quality of the score function $f$ at $\mathbf{z} \in \mathcal{Z}$. For example, a standard measure for classification is top-1 error, which compares the ground-truth label with the only one guess, i.e., the class with the highest score.

Recently, top-$k$ error has raised more attention due to label ambiguity. As shown in Fig. 2(a), it indicates whether the ground-truth label appears in the top-$k$ ranking list:

$$\text{TOP-}k^l(f) = \mathbb{E}_{\mathbf{z} \sim \mathcal{D}}[\text{err}_k(y, f(\mathbf{z}))],$$

where

$$\text{err}_k(y, f(\mathbf{x})) = 1[f(y, \mathbf{x}) > k].$$

1[$\cdot$] is the indicator function, and $\pi_{f(\mathbf{z})}(y) \in \{1, \ldots, C\}$ denotes the index of $f(\mathbf{z})$ when we sort $f(\mathbf{z})$ in a descending order. Meanwhile, top-$k$ accuracy, denoted as TOP-$k^l$, is defined as $1 - \text{TOP-}k^l$; thus the two metrics are equivalent.

In the following discussion, we use TOP-$k$ to refer to top-$k$ error and accuracy when there exists no ambiguity.

Note that different ways of breaking ties will lead to different $\pi_{f(\mathbf{z})}(y)$. Traditional methods assume that no ties exist since the number of classes is relatively small. However, the problem becomes non-negligible when calculating top-$k$ since label ambiguity generally involves a large number of classes. Following the prior arts on TOP-$k$ optimization [9], we adopt the worst-case assumption:

**Assumption 1.** There exist ties in $f(\mathbf{z})$, and all the ties will be wrongly broken. In other words, we have the following formulations:

- **Given** $y_1, y_2$ such that $\eta(\mathbf{z})_1 > \eta(\mathbf{z})_2$, if $f(\mathbf{z})_{y_1} = f(\mathbf{z})_{y_2}$, then we have $\pi_{f(\mathbf{z})}(y_1) > \pi_{f(\mathbf{z})}(y_2)$.
- **Given** the ground-truth label $y$, for any $y' \neq y$ such that $f(\mathbf{z})_{y'} = f(\mathbf{z})_y$, we have $\pi_{f(\mathbf{z})}(y) > \pi_{f(\mathbf{z})}(y')$.

**Remark 1.** The two formulations apply to different situations:

- The first formulation is a population-level assumption where the conditional probability $\eta(\mathbf{z})$ is available. It states that, when ties exist, the ranking results of $f(\mathbf{z})$ are opposite to those of $\eta(\mathbf{z})$. We apply this formulation in the proof of Theorems 3, 4, and 5.
- The second formulation is an empirical-level assumption where only the ground-truth label is available. It states that, when ties exist, the ground-truth label will be ranked lower. We apply this formulation in the proof of Theorem 2.
- In summary, these two conditions ensure that we will always choose the worst result when there is a tie in a conservative sense.

**3.2 Motivation**

As mentioned in the introduction, TOP-$k$ has its limitations. In this subsection, we will make a further analysis to reveal its reasons.

---

**Fig. 2.** Comparisons of TOP-$k$ and AUTKC: (a) TOP-$k$ focuses on the performance at a single point of the TOP-$k$ curve; (b) AUTKC-W considers the entire area under the TOP-$k$ curve; (c) AUTKC-P emphasizes the partial area with $k$ ranging in $[k_1, k_2]$. We denote AUTKC-P as AUTKC in the rest discussion for convenience.
is not discriminating enough, leaving some fatal errors free from penalty (L1). As shown in Fig. 3, we assume that Lake is the ground-truth label; Sky and Cloud are the labels with ambiguity; Beach and Valley are the irrelevant labels. Consider the following two score function such that:

\[
\begin{align*}
&f_1: \text{Lake} \succ \text{Sky} \succ \text{Cloud} \succ \text{Beach} \succ \text{Valley}, \\
&f_2: \text{Beach} \succ \text{Lake} \succ \text{Sky} \succ \text{Cloud} \succ \text{Valley}.
\end{align*}
\]

Obviously, \(f_1\) is an ideal score function since it ranks Lake first, while \(f_2\) is a bad score function since it ranks the irrelevant label Beach higher than the ground-truth label Lake. Based on this observation, we expect

\[
\text{err}(y, f_2(x)) > \text{err}(y, f_1(x)).
\]

However, when \(k \geq 2\), we have

\[
\text{err}_k(y, f_1(x)) = \text{err}_k(y, f_2(x)) = 0.
\]

In other words, although using the top-\(k\) error can relax the punishment on label ambiguity, it also relaxes the punishment on the fatal errors. In this sense, we need to seek out a more discriminating metric to selectively relax the punishment.

One may wonder whether a clever selection of \(k\) could eliminate this limitation. Unfortunately, the number of ambiguous labels generally differs among classes, while the hyperparameter \(k\) is universal for the classes. This contradiction leads to a dilemma during the training period: if the value of \(k\) is too small for some classes, label ambiguity might make these classes hard to optimize, which is the motivation of TOP-\(k\) optimization. But, if the value of \(k\) is too large, irrelevant labels will be more likely to be ranked higher than the ground-truth label, which is also unexpected. This suggests that merely looking at a single \(k\) is not enough to address (L1).

On the other hand, TOP-\(k\) is a static measure, ignoring the changes in operator’s decision condition (L2). To be specific, empirical results suggest that models trained with different hyperparameters \(k\) generally perform inconsistently on the top-\(k\) curve \([4], [8]\). As shown in Fig. 4, model \(f\) outperforms model \(g\) when \(k < 4\), while the results reverse when \(k > 4\). In view of this, the operator might ask: which model should we select to deploy? In practice, the answer depends on which \(k\) is of interest during the deployment phase. However, TOP-\(k\) optimization methods fix the value of \(k\) during the training phase. As a result, the model performance cannot be guaranteed if the \(k\) of interest changes after the model is deployed.

Being aware of the limitations of TOP-\(k\), we have the following question:

(Q1) Whether there exists a "better" metric for label ambiguity?

3.3 AUTKC and its Partial Variant

Our answer to (Q1) is to evaluate the score function from a more comprehensive perspective, rather than a single point of the top-\(k\) curve. Taking the inspiration from Area Under the ROC Curve (AUC) \([12]\), we propose to adopt the area under the top-\(k\) curve (AUTKC) to evaluate the quality of the given score function, which is also shown in Fig. 2(b):

\[
\text{AUTKC-W}^k(f) := \mathbb{E}_{x \sim D} \left[ \frac{1}{C} \int_{x \in C} \text{err}_k(y, f(x)) \, dk \right],
\]

(4)

The original definition of AUTKC considers the whole area under the top-\(k\) curve. However, we generally focus on the performance within a given range of \(k\). In this case, Eq. (4) provides a biased estimation of the expected performance by considering the unrelated regions. As shown in Fig. 2(c), this motivates us to adopt its variant involving the given range of \(k\):

\[
\text{AUTKC-P}^k(f) := \mathbb{E}_{x \sim D} \left[ \frac{1}{K_1 - K_2 + 1} \int_{K_1}^{K_2} \text{err}_k(y, f(x)) \, dk \right],
\]

(5)

where \(K_1, K_2\) are the minimum and the maximum number of ambiguous labels, respectively. Since TOP-1 is the most common metric of interest, and there generally exist some instances involving no ambiguous labels, we set \(K_1 = 1\) and denote \(K_2\) as \(K\) for the sake of conciseness in the following discussion.

Furthermore, since \(k\) is discrete, Eq. (5) enjoys a much simpler formulation:

\[
\text{AUTKC-P}^k(f) := \mathbb{E}_{x \sim D} [aerr_K(y, f(x))],
\]

(6)

where

\[
aerr_K(y, f(x)) = \frac{1}{K} \sum_{k=1}^{K} \text{err}_k(y, f(x)).
\]

(7)
Note that we denote AUTKC-P as AUTKC in the following discussion for the sake of convenience. Similar to TOP-k, the partial area under top-k accuracy curve, denoted as AUTKC, is equivalent to AUTKC, and we use AUTKC to refer to the two metrics when there exists no ambiguity.

### 3.4 AUTKC versus TOP-k

Intuitively, AUTKC is a better measure than TOP-k. On one hand, AUTKC is more discriminating than TOP-k, which helps overcome the limitation (L1). For example, when we set $K = 3$, AUTKC tells us $f_1$ is the ideal score function and indeed better than $f_2$:

$$aerr_K(y, f_1(z)) = \frac{1}{3} (0 + 0 + 0) = 0,$$

and

$$aerr_K(y, f_2(z)) = \frac{1}{3} (1 + 0 + 0) = \frac{1}{3}.$$  

On the other hand, AUTKC only requires a universal hyper-parameter $K$ for all the instances, that is, the maximum number of ambiguous labels. Such a simplification is beneficial to overcoming the limitation (L2).

Although the aforementioned example indicates the advantages of AUTKC, more theoretical clues are necessary to validate our argument. To this end, we first present the definition of consistent and discriminating, which is a classic technique for comparing measures [12].

#### Definition 1 (Degree of consistency [12]).

Given two measures $f$ and $g$ on domain $\Psi$ and two predictions $a, b \in \Psi$, let $R = \{(a, b) \mid f(a) > f(b), g(a) < g(b)\}$, $S = \{(a, b) \mid f(a) > f(b), g(a) > g(b)\}$. The degree of consistency between $f$ and $g$ is defined as $C = \frac{|R|}{|R| + |S|} \in [0, 1]$, where $|\cdot|$ is the cardinality of a set.

#### Remark 2.

The degree of consistency estimates the probability that the two measures make the same judgment on the given domain. A relatively high degree of consistency is a necessary condition for measure comparison.

#### Definition 2 (Degree of discrimination [12]).

Given two measures $f$ and $g$ on domain $\Psi$ and two predictions $a, b \in \Psi$, let $P = \{(a, b) \mid f(a) > f(b), g(a) = g(b)\}$, $S = \{(a, b) \mid g(a) > g(b), f(a) = f(b)\}$. The degree of discrimination between $f$ and $g$ is $D = \frac{|P|}{|P| + |S|}$.

#### Remark 3.

Degree of discrimination describes whether a measure could find the discrepancy that the other measure fails to discover. For example, the top-1 error is intuitively more discriminating than the top-k error when $k > 1$.

#### Definition 3 (Consistent and discriminating [12]).

The measure $f$ is consistent and more discriminating than $g$ if and only if $C > 0.5$ and $D > 1$. In this case, we conclude, intuitively, that $f$ is a "better" measure than $g$. Especially, if $C = 1.0$ and $D = \infty$, we say the measure $f$ is strictly consistent and more discriminating than $g$.

On top of consistency and discrimination, the following theorem suggests that AUTKC is a better measure than TOP-k. Please see Appendix A, available in the online.

#### Theorem 1.

For any $k < K$, AUTKC is strictly consistent and more discriminating than TOP-k.

#### Remark 4.

Being strictly consistent with TOP-k for any $k < K$ means that AUTKC is insensitive to the $k$ of interest, which overcomes the limitation (L2). Meanwhile, being more discriminating implies that AUTKC might be a better optimization objective than TOP-k, thus providing a clue to overcoming the limitation (L1). Further analysis in Section 5.1 will support these conjectures.

### 4 Roadmap for AUTKC Optimization

So far, we have known the advantage of AUTKC. Then, a natural question is

[(Q2) How to design learning algorithm that optimizes AUTKC effectively?]

Following the standard machine learning paradigm [74], we first reformulate the metric optimization problem to a risk minimization problem:

$$\text{(OP}_0) \min_f R_K(f) := \text{AUTKC}(f)$$

$$= \mathbb{E}_{z \sim D} \left[ \frac{1}{K} \sum_{k=0}^{K-1} \ell_{k-1} \left( s_y - s_{\langle k \rangle} \right) \right],$$

where $s_y := f(z)_y$ denotes the score of the class $y$, $(s_{\langle k \rangle})$ is the $k$-th greatest entry of $f(z)$ except $s_y$, and $\ell_{t-1}(t) := 1$ for $t \leq 0$ is the 0-1 loss. Note that we set $\ell_{0-1}(0) = 1$ since according to Assumption 1, any ties will lead to an error.

For the sake of conciseness, we provide an equivalent reformulation of $(\text{OP}_0)$, where the annoying operation $\langle y \rangle$ is removed. See Appendix B, available in the online supplement material, for the proof.

#### Theorem 2.

When Assumption 1 holds, the original optimization problem $(\text{OP}_0)$ is equivalent to the following problem:

$$\text{(OP}_1) \min_f \mathbb{E}_{z \sim D} \left[ \frac{1}{K} \sum_{k=0}^{K-1} \ell_{k-1} \left( s_y - s_{\langle k \rangle} \right) \right].$$

#### Remark 5.

Thm. 2 bridges AUTKC optimization with two classic problems: AUC optimization and label ranking. Specifically, AUC optimization is a binary classification problem with a pairwise ranking objective [12]:

$$\text{(OP}_{\text{AUC}}) \min_f \mathbb{E}_{z \sim D^+} \left[ \ell_{0-1} (f(z^+) - f(z^-)) \right],$$

where $D^+, D^-$ are the distribution of positive and negative samples, respectively. If we view the ground-truth label as the positive instance and the other labels as the negative ones, then AUTKC optimization is similar to a partial variant of AUC optimization in the context of label ranking. Although partial AUC optimization has been well studied [70], [71], [72], [73], there still exist two main differences:
The objective of AUC optimization assumes that optimality does not involve only one label.

These differences will lead to different theoretical results and practical implementation. Please refer to Sections 5 and 6 for the details.

According to Eq. (9), the main challenges for directly minimizing $R_K(f)$ are two-fold:

1. The loss function $\ell_{0-1}$ is not differentiable;
2. The data distribution $D$ is unavailable.

Next, we will present the solutions to (C1) and (C2) in Sections 5 and 6, respectively. Finally, we will obtain an end-to-end framework that optimizes the empirical risk of AUTKC efficiently.

**5 Surrogate Risk Minimization for AUTKC**

In this section, our task is to handle (C1). Specifically, the discrete $\ell_{0-1}$ in (OP1) will be replaced by a differentiable surrogate loss $\ell$, leading to the following surrogate risk minimization problem:

$$(OP_2) \min_f R_K^\ell(f) : = \mathbb{E}_{z \sim D} \left[ \frac{1}{K+1} \sum_{k=1}^{K+1} \ell(s_y - s_{[k]}) \right]. \quad \text{(10)}$$

But,

$$Q2-1: \text{What kind of surrogate loss should we select?}$$

We next answer this question in three steps. First, we study the Bayes optimal function for AUTKC, that is, the optimal solution of (OP2). Next is the condition of $\ell$ that optimizing (OP2) leads to the Bayes optimal function. In other words, when the surrogate risk is Fisher consistent with the $\ell_{0-1}$-based AUTKC risk. Finally, we answer the question (Q2-1) by discussing the consistency property of some common surrogate losses.

**5.1 AUTKC Bayes Optimality**

We first define the Bayes optimal score function for AUTKC, which is the best score function we could approximate:

**Definition 4 (AUTKC Bayes optimal).** The score function $f^* : \mathcal{X} \to \mathbb{R}^C$ is AUTKC Bayes optimal if

$$f^* \in \arg \inf_f R_K(f). \quad \text{(11)}$$

Our goal is to find the solution to Eq. (11). To this end, we first present the top-$K$ ranking-preserving property:

**Definition 5 (Top-$K$ ranking-preserving property).** Given $a, b \in \mathbb{R}^C$, we say that $b$ is top-$K$ ranking-preserving with respect to $a$, denoted as $RP_K(b, a)$, if for any $k \leq K$,

$$\pi_{b}(y) = k \implies \pi_{a}(y) = k.$$
Remark 7. If a score function $f$ is AUTKC Bayes optimal, no ties will exist in
\[
\{f(x)_{[1]}, \ldots, f(x)_{[K]}, f(x)_{[K+1]}\}.
\]

In other words, optimizing AUTKC encourages breaking the ties, which is an appealing property. Meanwhile, ties might exist in
\[
\{f(x)_{[K+1]}, \ldots, f(x)_{[C]}\}
\]
since it does not break the top-$K$ ranking-preserving property.

5.2 Consistency Analysis for Surrogate Risk Minimization

On top of Theorem 3, we could investigate the answer to the question (Q2-1). A basic requirement for $\ell$ is consistency. That is, the solution returned by minimizing $R_K^\ell(f)$ should recover the Bayes optimal score function:

**Definition 6 (AUTKC consistency).** The surrogate loss $\ell : R \rightarrow R_+$ is consistent with AUTKC if for every function sequence $\{f_i\}_{i=1,2,\ldots}$, we have:
\[
\mathcal{R}_K^\ell(f_i) \rightarrow \inf_f \mathcal{R}_K^\ell(f) \Rightarrow \mathcal{R}_K(f_i) \rightarrow \inf_f \mathcal{R}_K(f).
\]

Next, we provide a sufficient condition for AUTKC consistency, which enjoys a simpler formulation than checking whether the optimal solutions of $(OP_3)$ are Bayes optimal. Please see Appendix C.2, available in the online supplemental material, for the proof.

**Theorem 4.** The surrogate loss $\ell(t)$ is AUTKC consistent if it is bounded, differentiable and strictly decreasing.

So far, we have discussed the consistency w.r.t. all the measurable functions. However, common surrogate losses are not bounded. To this end, we next restrict the functions within a special function set $F$, which induces the concept of $F$-consistency:

**Definition 7 (AUTKC $F$-consistency).** The surrogate loss $\ell : R \rightarrow R_+$ is consistent with AUTKC if for every function sequence $\{f_i\}_{i=1,2,\ldots}, f_i \in F$, we have:
\[
\mathcal{R}_K^\ell(f_i) \rightarrow \inf_{f \in F} \mathcal{R}_K^\ell(f) \Rightarrow \mathcal{R}_K(f_i) \rightarrow \inf_{f \in F} \mathcal{R}_K(f).
\]

According to Theorem 4, the following proposition and corollary clearly hold:

**Proposition 1.** Let $F$ denote the set of functions whose outputs are bounded in $[0, 1]$. Then the surrogate loss $\ell(t)$ is $F$-consistent with AUTKC if it is differentiable and strictly decreasing in $[0, 1]$.

**Corollary 1.** Let $F$ denote the set of functions whose outputs are bounded in $[0, 1]$. Then the following statements hold:
- The square loss $\ell_{sq}(t) = (1-t)^2$ is $F$-consistent with AUTKC.
- The exponential loss $\ell_{exp}(t) = \exp(-t)$ is $F$-consistent with AUTKC.
- The logit loss $\ell_{logit}(t) = \log(1 + \exp(-t))$ is $F$-consistent with AUTKC.

Traditional TOP-k optimization methods adopt the hinge loss $\ell_{hinge}(t) = |1-t|_+$ as the surrogate loss, where $|t|_+ = \max\{0, t\}$. However, we point out that its variant in AUTKC optimization is inconsistent, whose proof is shown in Appendix C.3, available in the online supplemental material.

**Theorem 5.** The hinge loss is inconsistent with AUTKC when there exists a $z$ such that $\sum_{k=K+1}^{\infty} \eta(z)_k > \frac{1}{K+1}$.

**Proof Sketch** Given any sample $z$ satisfying the above condition and its Bayes optimal score $s^*$, Theorem 5 is induced by the fact that we could construct a score $s$ such that $-\mathcal{R}_k(s, \eta(z))$ but $s$ has a smaller risk than $s^*$ does. This construction holds by the property that $\ell_{hinge}(t) = 0$ for any $t \geq 1$. Please refer to Appendix C.3, available in the online supplemental material, for the details.

6 Generalization Analysis for AUTKC Optimization

In this section, we aim to relieve (C2). Since the data distribution $D$ is unknown, we turn to optimizing its empirical estimation based on the given dataset $S = \{z(i)\}_{i=1}^n$ sampled i.i.d. from $D$. Let $s(i)_y$ denote $f(x(i))_{y(i)}$ for convenience. Then, we have the following optimization problem:
\[
(OP_3) \min_{f \in F} \hat{\mathcal{R}}_K^\ell(f, S) := \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K+1} \ell(s(i)_y - s(i)_k).
\]

The results in Section 5 guarantee a small AUTKC risk $\mathcal{R}_K$ when we minimize the surrogate risk $\hat{\mathcal{R}}_K^\ell$, as long as the surrogate loss $\ell$ is AUTKC consistent. But,

\[
(\text{Q2 - 2}) \text{Does minimizing the empirical risk } \hat{\mathcal{R}}_K^\ell(f, S) \text{ lead to a small generalization error } \mathcal{R}_K^\ell(f, S) \text{ in other words, we should ensure } \mathcal{R}_K(f) - \hat{\mathcal{R}}_K^\ell(f, S) \text{ is bounded with high probability, and the upper bound goes to zero as the number of samples increases, especially when the number of classes is large}. \]

To this end, we next provide a Gaussian-complexity-based analysis for this bound in Section 6.1. Since deep neural networks have been one of the most popular function sets for large-scale classification, we further present its result in Section 6.2.

6.1 General Result for AUTKC Generalization Bound

Gaussian Complexity [74] is a classic measure of generalization error. Here we present its formal definition.

**Definition 8 (Gaussian Complexity).** Given the dataset $S = \{z(i)\}_{i=1}^n$, let $F$ be a family of functions defined on the sample space $Z$. Then, the empirical Gaussian complexity is defined as:
\[
\Theta_S(F) := \mathbb{E}_{S} \left[ \sup_{f \in F} \frac{1}{n} \sum_{i=1}^{n} g(i) f(z(i)) \right],
\]

where $g(i)$ is a Gaussian weight.
where \( \mathbf{g} = (g^{(1)}, g^{(2)}, \ldots, g^{(n)}) \) are the independent random variables sampled from the standard normal distribution \( N(0, 1) \).

Then, the following lemma provides the upper bound of \( \mathcal{R}_K^g(f) - \hat{\mathcal{R}}_K^g(f, S) \) via Gaussian complexity.

**Lemma 1 (Generalization Bound with Gaussian Complexity [74]).** Suppose that the loss function \( L \) is bounded by \( B > 0 \). Then, for any \( \delta \in (0, 1) \), with a probability of at least \( 1 - \delta \), the following holds

\[
\Phi(S, \delta) \leq \sqrt{2\pi} \Phi_S(L \circ \mathcal{F}),
\]

where

\[
\Phi(S, \delta) = \sup_{f \in \mathcal{F}} \left( \mathcal{R}_K^g(f) - \hat{\mathcal{R}}_K^g(f, S) \right) - 3B \sqrt{\log \frac{2}{\delta}} \frac{1}{2n},
\]

and \( L \circ \mathcal{F} := \{ L \circ f | f \in \mathcal{F} \} \) is the set that consists of the composition of the loss function \( L : \mathbb{R}^C \to \mathbb{R} \) and the score function \( f \in \mathcal{F} \).

In multi-class classification, this classic bound scales w.r.t. \( \mathcal{O}(\sqrt{n}) \) [74], [75], [76]. This result is rather unfavorable when facing numerous classes. Thanks to the recent advance in learning theory, a Lipschitz assumption on the loss function \( L \) can help eliminate this problem [77]:

**Assumption 2 (Lipschitz Continuity, w.r.t. a Variant of the \( \ell_2 \)-Norm [77]).** The loss function \( L : \mathbb{R}^C \to \mathbb{R} \) is Lipschitz continuous w.r.t. a variant of the \( \ell_2 \)-norm involving index \( y \in \mathcal{Y} \) if for all \( \mathbf{s}, \mathbf{s}' \in \mathbb{R}^C \)

\[
|L(\mathbf{s}) - L(\mathbf{s}')| \\
\leq L_1 \| (s_1 - s'_1, \ldots, s_C - s'_C) \|_2 + L_2 \left| s_y - s'_y \right|,
\]

where \( L_1, L_2 \) are Lipschitz constants.

Then, according to Lemma 1 in [77], we obtain the following generalization bound:

**Proposition 2 (Generalization Bound with the Lipschitz Continuity Variant of the \( \ell_2 \)-Norm).** If the loss function \( L \) satisfies Assumption 2, then for any \( \delta \in (0, 1) \), with probability of at least \( 1 - \delta \), we have

\[
\Phi(S, \delta) \leq 2\sqrt{\pi} \left[ L_1 \Phi_S(\mathcal{F}) + L_2 \Phi_S(\mathcal{F}) \right],
\]

where

\[
\tilde{\mathcal{S}} = \left\{ \left( \mathbf{x}^{(1)}(1), \mathbf{x}^{(1)}(2), \ldots, \mathbf{x}^{(1)}(C) \right), \right. \\
\left. \mathbf{x}^{(2)}, \right. \\
\left. \mathbf{x}^{(3)}, \right. \\
\left. \mathbf{x}^{(n)}, \right. \\
\text{induced by } \mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \mathbf{z}^{(3)}, \mathbf{z}^{(n)} \}
\]

and \( \mathcal{F} = \{ \tilde{f}(\mathbf{x}, y) = \mathbf{e}_y^T f(\mathbf{x}) | f \in \mathcal{F}, \mathbf{x} \in \mathcal{X}, y \in \mathcal{Y} \} \),

and \( \mathbf{e}_y = (0, \ldots, 0, 1, 0, \ldots, 0)^T \).

Note that this formulation generalizes the Theorem 2 in [77], where \( f \) is restricted in linear models. Such generalization is necessary when we analyze the bound of neural networks.

According to Proposition 2, we need to present the Lipschitz property of the surrogate losses mentioned in Corollary 1. To this end, we first show the following theorem, whose proof is shown in Appendix D.1, available in the online supplemental material.

**Theorem 6.** Assume that \( \mathbf{s}_i \subset [0, B_i] \) for any \( i = 1, \ldots, C \), if \( \ell(t) \) is strictly decreasing and \( L \)-Lipschitz continuous in the range \([-B_i, B_i]\). Then

\[
L_K(\mathbf{s}, y) = \frac{1}{K} \sum_{k=1}^{K+1} \ell(s_y - s_k)
\]

is Lipschitz continuous w.r.t. a variant of the \( \ell_2 \)-norm with Lipschitz constant pair \((\frac{2\sqrt{2(K+1)}}{K}, \frac{2\sqrt{2(K+1)}}{K})\).

On top of Theorem 6, the following results are clear. We leave the rest of the proof in Appendix D.1, available in the online supplemental material.

**Corollary 2.** Let \( \mathbf{s} \) denote the normalized score \( \text{softmax}(\mathbf{s}) \), where \( \text{softmax} \) is the softmax function. Then, the following statements hold:

- The square loss

\[
L_K^{sq}(\mathbf{s}, y) = \frac{1}{K} \sum_{k=1}^{K+1} \ell_{sq}(s_y - s_k)
\]

is Lipschitz continuous w.r.t. a variant of the \( \ell_2 \)-norm with Lipschitz constant pair

\[
\left( \frac{2\sqrt{2(K+1)}}{K}, \frac{2\sqrt{2(K+1)}}{K} \right).
\]

- The exponential loss

\[
L_K^{exp}(\mathbf{s}, y) = \frac{1}{K} \sum_{k=1}^{K+1} \ell_{exp}(s_y - s_k)
\]

is Lipschitz continuous w.r.t. a variant of the \( \ell_2 \)-norm with Lipschitz constant pair

\[
\left( \frac{e\sqrt{2(K+1)}}{2K}, \frac{e\sqrt{2(K+1)}}{2K} \right).
\]

- The logit loss

\[
L_K^{logit}(\mathbf{s}, y) = \frac{1}{K} \sum_{k=1}^{K+1} \ell_{logit}(s_y - s_k)
\]

is Lipschitz continuous w.r.t. a variant of the \( \ell_2 \)-norm with Lipschitz constant pair

\[
\left( \frac{\sqrt{2(K+1)}}{2eK \ln 2}, \frac{\sqrt{2(K+1)}}{2eK \ln 2} \right).
\]
6.2 Generalization Bound for Convolutional Neural Networks

According to Proposition 2 and Theorem 6, we could obtain the final generalization bound as soon as a proper upper bound of Gaussian complexity is available. To this end, we first present an important technique for the bound of Gaussian complexity named chaining in Section 6.2.1. Then, the final result of convolutional neural networks is available under the given regularization condition in Section 6.2.2.

6.2.1 Chaining Bound for Gaussian Complexity

Similar to Gaussian complexity, covering number [78] is another technique to measure the complexity of the given (pseudo)metric space:

**Definition 9 (ε-covering).** Let \( \{F, d\} \) be a (pseudo)metric space and \( B(f, r) \) denote the ball of radius \( r \) centered at \( f \in F \). Given \( \Theta \in F \), we say \( F_m = \{f_i\}_{i=1}^m \) is ε-covering of \( \Theta \) if \( \Theta \subseteq \bigcup_{i=1}^m B(f_i, \varepsilon) \). In other words, for any \( \theta \in \Theta \), there exists \( f_i \in F_m \) such that \( d(\theta, f_i) < \varepsilon \).

**Definition 10 (Covering Number).** On top of Definition 9, covering number of \( \Theta \) is defined as the minimum cardinality of any \( \varepsilon \)-covering of \( \Theta \):

\[
\mathcal{C}(\Theta, \varepsilon, d) := \min \{ m : F_m \text{ is } \varepsilon \text{-covering of } \Theta \}. \tag{18}
\]

Our task is to bound the Gaussian Complexity with covering number. To this end, we first check the sub-Gaussian property of Gaussian Complexity. See Appendix D.2, available in the online supplemental material, for the proof.

**Definition 11 (Sub-Gaussian Process. w.r.t. the Metric \( d_X \)).** A collection of zero-mean random variables \( \{X_f, f \in F\} \) is a sub-Gaussian process w.r.t. a metric \( d_X \) on \( F \) if

\[
\mathbb{E}[e^{\lambda (X_f - X_{f'})}] \leq e^{\frac{\lambda^2}{2} (d(f, f'))}, \forall f, f' \in F, \lambda \in \mathbb{R}.
\]

**Proposition 3 (Sub-Gaussian Property of Gaussian Complexity).** For any \( f \in F \), let

\[
T_f(g) = \sum_{i=1}^n \sum_{j=1}^C g_{ij} s_j^{(i)}.
\]

Then, \( Z := \frac{1}{\sqrt{nC}} (T_f(g) - T_f(g)) \) is a sub-Gaussian process with metric

\[
d_{\infty, S}(s, s') = \max_{z \in S} |s - s'|,
\]

where \( s_j^{(i)} = f(x^{(i)}) \).

Since \( d_{\infty, S}(s, s') \leq 1 \) and

\[
\Theta_{\bar{s}}(F) = \mathbb{E}_g \left[ \sup_{f \in F} \frac{1}{nC} T_f(g) \right],
\]

we have the chaining bound of Gaussian Complexity according to Dudley’s entropy integral [78]:

**Proposition 4 (Chaining bound of Gaussian Complexity).**

\[
\Theta_{\bar{s}}(F) \leq \inf_{a \geq 0} \left( C_1 a + \frac{C_2}{\sqrt{nC}} \int_a^1 \sqrt{\log \mathcal{C}(F, \varepsilon, d_{\infty, S})} \, d\varepsilon \right), \tag{19}
\]

where \( C_1, C_2 \) are two constants.

6.2.2 Practical Result for Convolutional Neural Networks

On top of Eq. (19), we are now ready to present the generalization bound of a family of neural networks, where both convolution layers and fully-connected layers exist. In fact, it is an application of the recent advance in [79].

**The Family of Neural Networks.** The network consists of \( L_c \) convolutional layers and followed by \( L_f \) fully-connected layers. Each convolutional layer consists of a convolution followed by an activation function and an optional pooling operation. All the convolutions use zero-padding [80] with the kernel \( K^{(l)} \in \mathbb{R}^{r \times d \times d \times d_{l-1} \times d_l} \) for layer \( l \in \{1, \ldots, L_c\} \). All the activation functions and pooling operations are 1-Lipschitz continuous. Meanwhile, we denote the parameters of the \( l \)-the fully-connected layer as \( V^{(l)} \). Finally, Let \( \Theta = \{K^{(1)}, \ldots, K^{(L_c)}, V^{(1)}, \ldots, V^{(L_f)}\} \) denote the set of all the parameters in the given network.

**Regularization of the Parameters.** The model input \( x \in \mathbb{R}^{d \times d \times d} \) satisfies \( ||\text{vec}(x)|| \leq \chi \), where \( \chi \) is the number of channels and \( \text{vec}(\cdot) \) is the vectorization operation. Note that convolution is essentially a linear operation, and we denote the operator matrix of the kernel \( K^{(l)} \) as \( \text{mt}(K^{(l)}) \). We assume that the initial parameters, denoted as \( \Theta_0 \), satisfy

\[
||\text{mt}(K_0) - \text{mt}(K^{(l)})||_2 \leq 1 + \nu, l = 1, \ldots, L_c,
\]

\[
||\text{mt}(V_0) - V^{(l)}||_2 \leq 1 + \nu, l = 1, \ldots, L_f.
\]

And the distance from \( \Theta_0 \) to the current parameters \( \Theta \) is bounded:

\[
||\Theta - \Theta_0|| := \sum_{l=1}^{L_c} ||\text{mt}(K^{(l)}) - \text{mt}(K_0)||_2 + \sum_{l=1}^{L_f} ||V^{(l)} - V_0||_2 \leq \beta. \tag{20}
\]

Finally, let \( F_{\beta, \varepsilon} \) denote the set of neural networks described above. According to Proposition 5 in Appendix D.3, available in the online supplemental material, where the covering number in Eq. (19) and the Gaussian complexity in Eq. (17) are bounded, we have the final result for the generalization bound of \( F_{\beta, \varepsilon} \). See Appendix D.3, available in the online supplemental material, for the details of the proof.

**Theorem 7 (Generalization bound of Convolutional Neural Networks).** Given the set of convolutional networks \( F_{\beta, \varepsilon} \) if the loss function \( L \) satisfies Assumption 2, then for any \( \delta \in (0, 1) \), with a probability of at least \( 1 - \delta \), we have

\[
\Phi(S, \delta) \leq O \left( \frac{WL_1 \sqrt{C \log (nC B_{\beta, \varepsilon})}}{\sqrt{n}} \right) + O \left( \frac{WL_2 \log (nB_{\beta, \varepsilon})}{\sqrt{n}} \right).
\]
where $B_{\beta,v,K} := \chi(1 + v + \beta/L_\alpha)^{\frac{\alpha}{l}}$, $L_\alpha = L_c + L_f$, and $W$ is the number of parameters in $F_{\beta,v}$.

**Remark 8.** According to Theorem 3 and the generalization bound, it seems that we can simply set $K$ as the number of classes. However, only the ranking of the ground-truth label and the ambiguous labels is of interest, which are all top-ranked classes. In this sense, a large $K$ will provide a biased estimation of the performance by including more uninformative loss terms when taking the average. Motivated by this, a moderate $K$, which is exactly equal to or slightly larger than the maximum number of relevant labels, might be a better choice. Since the maximum number of relevant labels is proportional to the number of classes, we set $K \geq p \cdot C$, where $p$ is a universal constant. Note that $L_1 \sim \mathcal{O}(\frac{1}{\sqrt{K}})$ according to Theorem 6. In this sense, $\Phi(S, \delta)$ becomes irrelevant to $C$.

**Computational Complexity.** On one hand, the proposed method shares a similar convergence rate with prior arts since it can be optimized efficiently via common optimizers such as SGD [81]. On the other hand, the additional complexity of the proposed method comes from the top-$K$ ranking operator. With the help of heapsort, the average complexity of each sample is in order of $\mathcal{O}(C \log K)$. Then, the total complexity is in order of $\mathcal{O}(nC \log K + nK)$, where $n$ is the number of samples. As analyzed in the revised remark for Theorem 9, we prefer choosing a moderate $K$. Thus, compared with top-$k$ optimization, whose complexity is in order of $\mathcal{O}(nC + n)$, the complexity of the proposed method is still moderate.

### 7 EXPERIMENT

In this section, we analyze the empirical results on four benchmark datasets to validate the proposed AUTKC optimization framework as well as the theoretical results.

#### 7.1 Datasets

We first describe the datasets used in the following discussion:

- **Cifar-10** [82]: This dataset\(^1\) consists of 60,000 $32 \times 32$ color images in 10 classes, with 6,000 images per class. There are 50,000 and 10,000 images in the training set and the test set, respectively. At the first glance, Cifar-10 is not suitable for the context of label ambiguity since all the classes are completely mutually exclusive. For example, the class `Automobile` includes sedans, SUVs, things of that sort while the class `Truck` includes only big trucks. However, we have shown that AUTKC is a better metric than TOP-$k$ for any $k < K$ in Section 3.4. We adopt this dataset to justify this argument even when no label ambiguity involves.

- **Cifar-100** [82]: This dataset\(^1\) is similar to the Cifar-10, except it has 100 classes each containing 600 images. There are 500 training images and 100 testing images per class. As the number of classes increases, label ambiguity inevitably happens. For example, the class `Sea` tends to overlap with the class `Dolphin` and `Seal`; the class `Cloud` is widely distributed in outdoor classes such as `House`, `Mountain`, `Plain` and `Skyscraper`.

- **Tiny-imagenet-200**\(^2\): This dataset contains 100,000 $64 \times 64$ color images in 200 classes. Each class has 500 training images, 50 validation images and 50 test images. The classification challenge on this dataset adopts both TOP-1 and TOP-5 as the evaluation metrics. Since the test set does not provide annotations, we report the performances on the validation set.

- **Places-365**\(^3\): The standard Places-365\(^3\) dataset contains 1.8 million training images in 365 scene classes. There are 50 images per class in the validation set and 900 images per class in the test set. We use this dataset to validate the proposed framework on large-scale datasets.

#### 7.2 Competitors

Specifically, we hope to validate three arguments: (1) TOP-$k$ is not an ideal optimization objective, as presented in Section 3.2; (2) AUTKC is a better metric than TOP-$k$ since the models induced by AUTKC optimization consistently achieve better performance; (3) The hinge loss is inconsistent in AUTKC optimization. Motivated by this, the competitors include:

- Traditional loss functions for multiclass classification, such as cross-entropy loss ($CE$) and multi-class hinge loss ($Hinge$).

- Prior arts of TOP-$k$ optimization, whose details are presented in Table 2. Note that $TCE$ and $L1$ are induced by $CE$ and $Hinge$ via simple top-$k$ truncation, respectively. Meanwhile, $L2$, $L3$ and $L4$ are hinge-like top-$k$ loss functions consistent with TOP-$k$ when no ties exist in the scores. $L5$ is the recent consistent variant even when ties exist. The value of $k$ is searched in $\{3, 5\}$ on Cifar-10 and in $\{3, 5, 10\}$ on the other datasets.

Our AUTKC optimization framework implemented with the hinge surrogate loss (AUTKC-Hinge):

$$L_K^{\text{hinge}}(s, y) = \frac{1}{K} \sum_{k=1}^{K-1} \ell_{\text{hinge}}(s_y - s_{[k]}),$$

where $K$ is searched in the same space as $k$.

---

1. https://www.cs.toronto.edu/~kriz/cifar.html

2. https://www.kaggle.com/c/tiny-imagenet

3. http://places2.csail.mit.edu/download.html
### 7.3 Implementation Details

**Infrastructure.** We carry out all the experiments on an ubuntu 16.04 server equipped with Intel(R) Xeon(R) Silver 4110 CPU and an Nvidia(R) TITAN RTX GPU. The version of CUDA is 10.2 with GPU driver version 440.44. The codes are implemented via python (v-3.8.11) [84], and the main third-party packages include pytorch (v-1.9.0), numpy (v-1.20.3), scikit-learn (v-0.24.2) and torchvision (v-0.10.0).

**Evaluation Metric.** Given a trained score function $f$, we evaluate its performance with $\text{AUTKC}^i$ (the up the better):

$$\text{AUTKC}^i = \frac{1}{nK} \sum_{i=1}^{n} \left| \left\{ k : f(x_i)_{y_i} > f(x_i)_{[k]} , k \leq K + 1 \right\} \right| .$$

The value of $K$ ranges in $\{3, 5\}$ on Cifar-10 and $\{3, 5, 10\}$ on the other datasets. Meanwhile, we also record the performance at some points of the top-$k$ curve:

$$\text{TOP-k}^i = \frac{1}{n} \sum_{i=1}^{n} 1 \left[ f(x_i)_{y_i} > f(x_i)_{[k]} \right].$$

where the value of $k$ is in the range of $\{1, 5\}$ on Cifar-10 and $\{1, 10\}$ on the other datasets.

**Backbone and Optimization Method.** We utilize ResNet-18 [20] as the backbone on Cifar-10, Cifar-100 and Tiny-imagenet-200. The parameters of this neural network are initialized with the checkpoint pre-trained on ILSVRC2012 [5], and we finetune the whole model by Stochastic Gradient Descent (SGD) [81] with Nesterov momentum. Empirically, the batch size and momentum are set as 128 and 0.9, respectively; the learning rate at the first epoch is searched in $\{0.001, 0.0001, 0.00001\}$, and decays exponentially with a ratio of 0.1 every 30 epochs; the $\ell_2$ regularization term is searched in $\{0.001, 0.0001, 0.00001\}$. On Places-365, to scale up the training period, we extract the original images to the features with 2,048 dimensions via the official pre-trained ResNet-50$^4$. Then, the backbone is implemented with a three-layer fully-connected neural network, whose parameters are randomly initialized. The optimization strategy is similar to that on the other datasets, except that the batch size is 4096 and the initial learning rate is searched in $\{0.1, 0.01, 0.001, 0.0001\}$. The total number of epochs is set as 50 on Tiny-imagenet-200 and 90 on the other datasets. Note that Tiny-imagenet-200 has the same domain as ILSVRC2012, which helps the training process converge in fewer epochs.

**Warm-Up Training.** Focusing on a few classes during the early training period makes it easy to suffer from over-fitting. Thus, we adopt a warm-up strategy to encourage the

---

**Table 3**

| Type          | Dataset      | Cifar-10            | Cifar-100          | Tiny-imagenet-200 | Places-365 |
|--------------|--------------|---------------------|--------------------|-------------------|-----------|
|              | $K$          |                     |                    |                   |           |
|              | 3            | 5                   | 3                  | 5                 | 3         | 5        |
| Traditional  | CE           | 92.87               | 95.24              | 71.81             | 76.70     | 83.05    | 81.56    | 85.34    | 89.66 |
|              | Hinge        | 92.77               | 95.20              | 71.33             | 76.65     | 83.35    | 79.85    | 83.95    | 88.74 |
| Top-$k$      | L1           | 89.58               | 93.25              | 69.90             | 76.19     | 83.19    | 81.41    | 85.22    | 89.54 |
|              | L2           | 92.04               | 94.70              | 71.34             | 77.19     | 83.98    | 81.43    | 85.30    | 89.64 |
|              | L3           | 92.56               | 94.92              | 71.98             | 76.78     | 82.81    | 81.87    | 85.48    | 89.66 |
|              | L4           | 91.64               | 94.48              | 71.47             | 77.28     | 84.05    | 81.16    | 85.11    | 89.48 |
|              | L5           | 88.88               | 92.79              | 68.88             | 75.27     | 82.31    | 81.24    | 85.10    | 89.43 |
|              | TCE          | 90.29               | 93.74              | 71.41             | 76.99     | 83.73    | 79.45    | 83.82    | 88.76 |
| AUTKC (Ours) | AUTKC-Hinge  | 92.93               | 95.27              | 73.29             | 78.33     | 84.57    | 82.20    | 85.73    | 89.99 |
|              | AUTKC-Sq     | 93.00               | 95.01              | 73.39             | 78.43     | 84.76    | 82.05    | 85.70    | 89.99 |
|              | AUTKC-Exp    | 93.01               | 95.30              | 73.60             | 78.70     | 84.87    | 82.87    | 85.77    | 90.01 |
|              | AUTKC-Logit  | 93.05               | 95.30              | 73.29             | 78.42     | 84.69    | 82.12    | 85.81    | 90.05 |

The Best and the runner-up methods on each metric are marked with red and blue, respectively. The best method of the competitors is marked with underline.
attention on all the classes. Specifically, the model is trained with CE at the first $E_w$ epochs. Afterward, we start the AUTKC training phase by optimizing the proposed objectives. The corresponding ablation study is shown in Section 7.4.2, where $E_w$ is searched in $\{5, 10, 20, 30, 40\}$.

### 7.4 Results and Analysis

#### 7.4.1 Overall Performance

**The Comparison on AUTKC.** First, we record the best performance of each method on each dataset in Table 3. From the results, we have the following observations:

- The AUTKC methods outperform all the competitors consistently on all the metrics on Cifar-100, Tiny-imagenet-200 and Places-365. This validates the effectiveness of the proposed optimization framework on the datasets where label ambiguity exists.
- A slight improvement is also attainable with AUTKC optimization methods on Cifar-10, where the number of classes is small. This result evidences our theoretical results in Sections 3.4 and 5.1, that is, AUTKC is more discriminating than TOP-$k$.
- Among the four surrogate losses for AUTKC optimization, the hinge loss performs inferior to the others, which is consistent with our theoretical analysis in Section 5.
- Top-$k$ optimization methods achieve a slight improvement w.r.t. CE on Cifar-100, Tiny-imagenet-200 and Places-365. This empirical result is consistent with the observations in [8], [9].
- On Cifar-10, the performances of top-$k$ optimization methods fail to surpass CE, which also validates the limitation of TOP-$k$ shown in Section 3.2.

**The Comparison on the Top-$k$ Curve.** Then, we plot the top-$k$ curve in Fig. 6 for more fine-grained comparison. Note that we summarize the results of each type of method for concise illustration, where the scattered points represent the average performance of each type of method, and the shadow represents the standard deviation. Meanwhile, the top-$k$ accuracy
ranges from 50 to 90 at different $k$ points, while the performance gap between different methods is significantly smaller. This scale difference makes the performance gap almost invisible. To fix this issue, we normalize the performance gap on each dataset and each $k$ point by the following strategy:

$$\text{Normalized Top-}k \text{ Acc Gain of method } m = \begin{cases} \frac{\text{Gain}(m, k)}{G_+}, & \text{Gain}(m, k) > 0; \\ \frac{\text{Gain}(m, k)}{G_-}, & \text{otherwise}. \end{cases}$$

where $\text{Gain}(m, k)$ is the top-$k$ accuracy gain of method $m$ with respect to CE, and

$$G_+ := \max_{m,k} \text{Gain}(m, k),$$
$$G_- := \min_{m,k} \text{Gain}(m, k),$$

are the maximum and minimum of all the gains on the dataset, respectively. According to Fig. 6, we have the following observations:

- In most cases, the average gains of the proposed framework are positive, especially when $k$ is small. This again suggests that optimizing AUTKC is beneficial to the top-$k$ accuracy with $k < K$.
- The gains of top-$k$ optimization methods tend to be negative. Moreover, we can see that the gains are increasing at the first several $k$s. All these observations again validate our analysis of the limitation of TOP-$k$. Review Section 3.2 for the details.

**Case Study.** After the quantitative comparison, we further visualize the ranking results of different types of methods in Fig. 7 to further investigate the performance. As illustrated by the demo images, the class beach house is semantically related to the class balcony-exterior and roof garden, and the class lagoon naturally overlaps with the class islet, ocean, coast and sky. As a result, all these methods fail to rank the annotated label first due to label ambiguity. For these instances, CE ranks the ambiguous labels higher than the annotated label, thus degenerating the top-$k$ performance. Meanwhile, the top-$k$ method succeeds in improving the order of the annotated label, while the irrelevant labels are ranked higher, such as chalet and mansion in the first case and ice shelf and beer hall in the second one. In contrast, the annotated label and the ambiguous labels are both ranked higher by the proposed AUTKC method. These cases again validate the superiority of AUTKC and the proposed framework.

### 7.4.2 Sensitivity Analysis

**The Effect of $K$.** In Fig. 8, we show the sensitivity in terms of hyperparameter $K$ on Places-365 for the AUTKC methods with $E_w = 40$. Note that the space of $K$ includes the total number of classes. We have the following observations from the results:

- AUTKC-Hinge has a different trend from the other methods. To be specific, its median performance is insensitive to the value of $K$, and the worst performance becomes worse as the value of $K$ increases. This counter-intuitive phenomenon again validates Theorem 5, that is, AUTKC-Hinge is not AUTKC consistent.
- The consistent methods, e.g., AUTKC-Sq, AUTKC-Exp and AUTKC-Logit, achieve the best median performance on AUTKC@5 when $K$ is set as 3 or 5. This observation is consistent with our goal to design the AUTKC objective.
- When $K$ equals 1, the consistent methods generally have the worst median performance. This is not surprising since this case does not properly handle label ambiguity.
The consistent methods with $K = 365$ also perform competitively but are inferior to the ones with $K = 3$ or $K = 5$. This confirms the necessity to select a moderately large $K$.

The Effect of $E_w$. Furthermore, we show the sensitivity in terms of hyperparameter $E_w$ on Cifar-100 for the AUTKC methods in Fig. 9. Note that we record the results only when the method succeeds to converge. From the results, one can observe that:

- On one hand, the increasing $E_w$ from 5 to 10 brings an increasing trend of the median performance. On the other hand, a larger $K$ generally leads to a decreasing trend of the performance deviation. This shows that the warm-up strategy is beneficial to the performance of the AUTKC methods.

- The AUTKC methods achieve the best median performance when $E_w$ is equal to 10 or 20. This suggests that a moderate warm-up strategy might be a better strategy for AUTKC optimization.

### 8 Conclusion

This paper proposes a new metric named AUTKC to overcome the limitations of the top-$k$ objective in the context of label ambiguity. An empirical surrogate risk minimization framework is further established for efficient AUTKC optimization. Theoretically, we show that the surrogate loss is Fisher consistent with the 0-1-loss-based AUTKC risk if it is differentiable, bounded, and strictly decreasing. Most common surrogate losses satisfy this condition when the model outputs are bounded, such as the square loss, the exponential loss, and the logit loss. However, the hinge loss, the standard surrogate loss for top-$k$ optimization, is inconsistent. Moreover, we construct the generalization bound for empirical AUTKC optimization, which is insensitive to the number of classes. Finally, the empirical results on four benchmark datasets justify the proposed framework and the theoretical results.

### References

[1] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, “Places: A 10 million image database for scene recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 6, pp. 1452–1464, Jun. 2018.

[2] B. Frenay and M. Verleysen, “Classification in the presence of label noise: A survey,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 5, pp. 845–869, May 2014.

[3] R. Jin and Z. Ghahramani, “Learning with multiple labels,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2002, pp. 897–904.

[4] M. Lapin, M. Hein, and B. Schiele, “Loss functions for top-k error: Analysis and insights,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 1468–1477.

[5] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2009, pp. 248–255.

[6] M.-L. Zhang and Z.-H. Zhou, “A review on multi-label learning algorithms,” IEEE Trans. Knowl. Data Eng., vol. 26, no. 8, pp. 1819–1837, Aug. 2014.

[7] M. Lapin, M. Hein, and B. Schiele, “Top-k multiclass SVM,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2015, pp. 325–333.

[8] M. Lapin, M. Hein, and B. Schiele, “Analysis and optimization of loss functions for multiclass, top-k, and multilabel classification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 7, pp. 1533–1554, Jul. 2018.

[9] F. Yang and S. Koyejo, “On the consistency of top-k surrogate losses,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 1073–10735.

[10] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, “The pascal visual object classes (VOC) challenge,” Int. J. Comput. Vis., vol. 88, pp. 303–338, 2010.

[11] L. Lin et al., “Microsoft COCO: Common objects in context,” in Proc. Eur. Conf. Comput. Vis., 2014, pp. 740–755.

[12] C. X. Ling, J. Huang, and H. Zhang, “AUC: A statistically consistent and more discriminating measure than accuracy,” in Proc. Int. Joint Conf. Artif. Intell., 2003, pp. 519–526.

[13] T. Liu, “Learning to rank for information retrieval,” Found. Trends Inf. Retrieval, vol. 3, pp. 225–331, 2009.

[14] S. Rendle, L. B. Marrinho, A. Nanopoulos, and L. Schmidt-Thieme, “Learning optimal ranking with tensor factorization for tag recommendation,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2009, pp. 727–736.

[15] S. Agarwal, “The infinite push: A new support vector ranking algorithm that directly optimizes accuracy at the absolute top of the list,” in Proc. Int. Conf. Data Mining, 2011, pp. 839–840.

[16] C. Rudin, “The P-norm push: A simple convex ranking algorithm that concentrates at the top of the list,” J. Mach. Learn. Res., vol. 10, pp. 2223–2271, 2009.

[17] N. Usunier, D. Buffoni, and P. Gallinari, “Ranking with ordered weighted pairwise classification,” in Proc. Int. Conf. Mach. Learn., 2009, pp. 1057–1064.

[18] F. Perronnin, Z. Akata, Z. Harchaoui, and C. Schmid, “Towards good practice in large-scale learning for image classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2012, pp. 3480–3489.

[19] O. Russakovsky et al., “ImageNet large scale visual recognition challenge,” Int. J. Comput. Vis., vol. 115, pp. 211–252, 2015.

[20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

[21] J. J. McAuley, A. Ramisa, and T. S. Caetano, “Optimization of ranking dist loss functions for weakly-labeled image taxonomies,” Int. J. Comput. Vis., vol. 104, pp. 343–361, 2013.

[22] S. Ross, J. Zhou, Y. Yue, D. Dey, and D. Bagnell, “Learning policies that are robust loss functions for weakly-labeled image taxonomies,” Int. J. Comput. Vis., 2015, pp. 1364–1372.

[23] K. Swersky, D. Tarlow, R. P. Adams, R. S. Zemel, and B. J. Frey, “Probabilistic n-choose-k models for classification and ranking,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2012, pp. 3089–3097.

[24] M. Guillaumin, T. Mensink, J. J. Verbeek, and C. Schmid, “TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation,” in Proc. IEEE Int. Conf. Comput. Vis., 2009, pp. 309–316.

[25] T. Mensink, J. J. Verbeek, F. Perronnin, and G. Csurka, “Distance-based image classification: Generalizing to new classes at near-zero cost,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 11, pp. 2624–2637, Nov. 2013.

[26] Y. Zhou, Y. Liu, J. Yang, X. He, and L. Liu, “A taxonomy of label ranking algorithms,” J. Comput., vol. 9, pp. 557–565, 2014.

[27] S. Har-Peled, D. Roth, and D. Zimak, “Constraint classification for multiclass classification and ranking,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2002, pp. 785–792.

[28] O. Dekel, C. D. Manning, and Y. Singer, “Log-linear models for label ranking,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2003, pp. 497–504.

[29] E. Hüllermeier, J. Fünkranz, W. Cheng, and K. Brinker, “Label ranking by learning pairwise preferences,” Artif. Intell., vol. 172, pp. 1897–1916, 2008.

[30] J. Fünkranz and E. Hüllermeier, “Preference learning and ranking by pairwise comparison,” in Preference Learning, Berlin, Germany: Springer, 2010, pp. 65–82.

[31] W. Cheng, J. C. Huhn, and E. Hüllemeier, “Decision tree and instance-based learning for label ranking,” in Proc. Int. Conf. Mach. Learn., 2009, pp. 161–168.

[32] P. L. H. Yu, W. M. Pan, and P. H. Lee, “Decision tree modeling for ranking data,” in Preference Learning, Berlin, Germany: Springer, 2010, pp. 83–106.

[33] J. A. Alejo, J. A. Gámez, and D. Molina, “Tackling the supervised label ranking problem by bagging weak learners,” Inf. Fusion, vol. 35, pp. 38–50, 2017.

[34] C. R. de Sá, C. Soares, A. J. Knobbe, and P. Cortez, “Label ranking forests,” Expert Syst. J. Knowl. Eng., vol. 34, no. 1, 2017, Art. no. e12166.
Zitai Wang received the BS degree in computer science and technology from Beijing Jiaotong University in 2019. He is currently working toward the PhD degree with the University of Chinese Academy of Sciences. His research interests include machine learning and data mining, with special focus on top-k optimization and preference learning.
WANG ET AL.: OPTIMIZING PARTIAL AREA UNDER THE TOP-K CURVE: THEORY AND PRACTICE

Qianqian Xu (Senior Member, IEEE) received the BS degree in computer science from the China University of Mining and Technology, in 2007, and the PhD degree in computer science from the University of Chinese Academy of Sciences, in 2013. She is currently an associate professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. Her research interests include statistical machine learning, with applications in multimedia and computer vision. She has authored or coauthored more than 50 academic papers in prestigious international journals and conferences (including the IEEE Transactions on Pattern Analysis and Machine Intelligence, International Journal of Computer Vision, IEEE Transactions on Image Processing, NeurIPS, ICM, CVPR, AAAI, etc). Moreover, she serves as an associate editor of the IEEE Transactions on Circuits and Systems for Video Technology, ACM Transactions on Multimedia Computing, Communications, and Applications, and Multimedia Systems.

Zhiyong Yang received the MSc degree in computer science and technology from the University of Science and Technology Beijing (USTB), in 2017, and the PhD degree from the University of Chinese Academy of Sciences (UCAS), in 2021. He is currently a postdoctoral research fellow with the University of Chinese Academy of Sciences. His research interests lie in machine learning and learning theory, with special focus on AUC optimization, meta-learning/multi-task learning, and learning theory for recommender systems. He has authored or coauthored about 32 academic papers in top-tier international conferences and journals including the IEEE Transactions on Pattern Analysis and Machine Intelligence, ICML, NeurIPS, CVPR. He served as a senior PC member for IJCAI 2021 and a reviewer for several top-tier journals and conferences such as the IEEE Transactions on Image Pattern Analysis and Machine Intelligence, Transactions on Machine Learning Research, ICML, NeurIPS, and ICLR.

Yuan He received the BS and PhD degrees from Tsinghua University, China. He is a senior staff engineer with the Security Department, Alibaba Group, and working on artificial intelligence-based content moderation and intellectual property protection systems. Before joining Alibaba, he was a research manager with Fujitsu working on document analysis system. He has published more than 30 papers in computer vision and machine learning related conferences and journals including CVPR, ICCV, ICML, NeurIPS, AAAI, and ACM MM. His research interests include computer vision, machine learning, and AI security.

Xiaochun Cao (Senior Member, IEEE) received the BS and ME degrees both in computer science from Beihang University (BUAA), China, and the PhD degree in computer science from the University of Central Florida, USA, with his dissertation nominated for the university level Outstanding Dissertation Award. He is a professor with the School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen University. After graduation, he spent about three years with ObjectVideo Inc. as a research scientist. From 2008 to 2012, he was a professor with Tianjin University. Before joining SYSU, he was a professor with the Institute of Information Engineering, Chinese Academy of Sciences. He has authored and coauthored more than 200 journal and conference papers. In 2004 and 2010, he was the recipient of the Piero Zamperoni Best Student Paper Award at the International Conference on Pattern Recognition. He is on the editorial boards of the IEEE Transactions on Image Processing and IEEE Transactions on Multimedia, and was on the editorial board of the IEEE Transactions on Circuits and Systems for Video Technology.

Qingming Huang (Fellow, IEEE) received the bachelor’s degree in computer science and the PhD degree in computer engineering, both from the Harbin Institute of Technology, China, in 1988 and 1994. He is a chair professor with the University of Chinese Academy of Sciences and an adjunct research professor with the Institute of Computing Technology, Chinese Academy of Sciences. His research areas include multimedia computing, image processing, computer vision, and pattern recognition. He has authored or coauthored more than 400 academic papers in prestigious international journals and top-level international conferences. He was the associate editor of the IEEE Transactions on Circuits and Systems for Video Technology and Acta Automatica Sinica, and the reviewer of various international journals including the IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, etc. He has served as a general chair, program chair, track chair and TPC member for various conferences, including ACM Multimedia, CVPR, ICCV, ICME, ICMR, PCM, BigMM, PSIVT, etc.

*For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.*