Local Bayes Risk Minimization Based Stopping Strategy for Hierarchical Classification

Yu Wang, Qinghua Hu, Yucan Zhou, Hong Zhao
Tianjin University
Tianjin, China
{armstrong_wangyu, huqinghua, yucanzhou, hongzhaocn}@tju.edu.cn

Yuhua Qian, Jiye Liang
Shanxi University
Taiyuan, China
{jinchengqyh, ljy}@sxu.edu.cn

***Abstract*—In large-scale data classification tasks, it is becoming more and more challenging in finding a true class from a huge amount of candidate categories. Fortunately, a hierarchical structure usually exists in these massive categories. The task of utilizing this structure for effective classification is called hierarchical classification. It usually follows a top-down fashion which predicts a sample from the root node with a coarse-grained category to a leaf node with a fine-grained category. However, misclassification is inevitable if the information is insufficient or large uncertainty exists in the prediction process. In this scenario, we can design a stopping strategy to stop the sample at an internal node with a coarser category, instead of predicting a wrong leaf node. Several studies address the problem by improving performance in terms of hierarchical accuracy and informative prediction. However, all of these researches ignore an important issue: when predicting a sample at the current node, the error is inclined to occur if large *uncertainty* exists in the next lower level children nodes. In this paper, we integrate this uncertainty into a *risk problem*: when predicting a sample at a decision node, it will take precipitance risk in predicting the sample to a children node in the next lower level on one hand, and take conservative risk in stopping at the current node on the other. We address the risk problem by designing a Local Bayes Risk Minimization (LBRM) framework, which divides the prediction process into recursively deciding to stop or to go down at each decision node by balancing these two risks in a top-down fashion. Rather than setting a global loss function in the traditional Bayes risk framework, we replace it with different uncertainty in the two risks for each decision node. The uncertainty on the precipitance risk and the conservative risk are measured by information entropy on children nodes and information gain from the current node to children nodes, respectively. We propose a Weighted Tree Induced Error (WTIE) to obtain the predictions of minimum risk with different emphasis on the two risks. Experimental results on various datasets show the effectiveness of the proposed LBRM algorithm.**

I. Introduction

With the advent of the big data era, we are often confronted with a huge amount of categories in classification tasks. It poses great challenges on finding a true class from these massive candidate categories. Fortunately, a hierarchical relationship of tree structure or directed acyclic graph (DAG) structure always exists in these massive classes, such as the large taxonomies of Google for web page classification, the semantic hierarchy of ImageNet [1] for image classification, and the gene hierarchy of National Center for Gene Research for gene classification. Classes in the hierarchy have a parent-children relationship, in which concepts are from the abstract to the concrete and granularities are from the coarse to the fine. We can utilize this hierarchical relationship to realize more effective and efficient classification, referred as hierarchical classification.

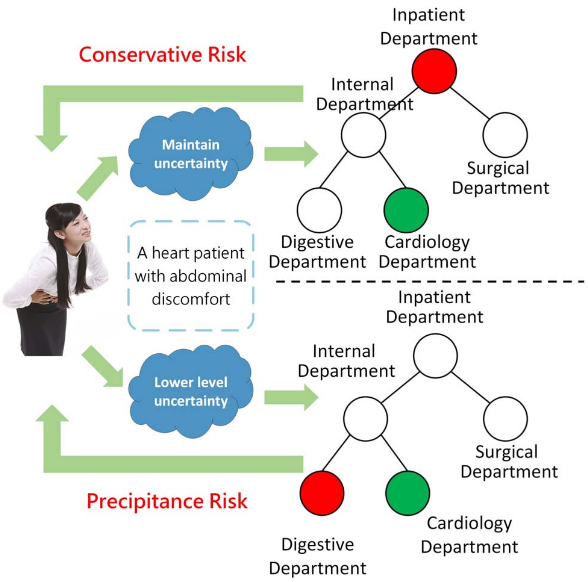


Fig. 1: A patient example illustrating the precipitance risk and the conservative risk. The green node is the ground truth, while the red node is the prediction with high risk.

In recent years, hierarchical classification receives increasing attention in gene classification [2], [3], [4], image classification [5], [6], [7], [8], medical image annotation [9], [10] and many other domains [11], [12]. Most of them predict a sample starting at the root node with a coarse-grained category until a leaf node with a fine-grained category is reached. However, misclassification is inevitable if the information is insufficient or huge uncertainty exists. In this scenario, we can stop the sample at an internal node with a coarser category, instead of predicting a wrong leaf node in the hierarchy (Fig. 2(a)). To this end, it is of great significance to design an effective stopping strategy.

Several studies have been devoted to designing stopping strategies in the past few years. Sun et al. introduce this

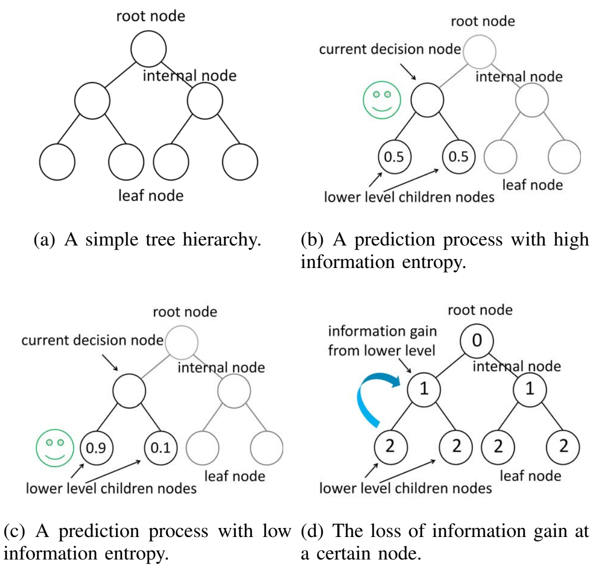
2374-8486/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDM.2017.61

problem into hierarchical classification in [13] and utilize a threshold method in [14] by setting a threshold for each class node. They stop pushing the sample down to the next lower level only if the probability is not greater than the threshold of this level [12]. Ceci et al. [15] find the best values of these thresholds by optimizing F1 score to encourage the samples to go deeply. And they improve their algorithm by optimizing the tree induced error (TIE) to leverage hierarchy information in [16]. Deng et al. [17] first propose the accuracy-specificity trade-off in hierarchical classification, obtaining the prediction which maximizes the information gain while ensuring a fixed accuracy guarantee. However, all of these researches ignore an important issue: when predicting a sample at the current decision node, the error is inclined to occur if large *uncertainty* exists in the next lower level children nodes.

In this paper, we integrate this issue into a *risk problem*: when predicting a sample at a decision node, it will take precipitance risk in predicting to the next lower level children node and take conservative risk in stopping at the current node. For example, a patient with abdominal discomfort may get a stomach trouble or have a heart disease, as shown in Fig. 1. Suppose she has a heart disease in fact. She will be faced with the precipitance risk and be sent to department of digestive if the doctor has a smattering of the phenomenon with large uncertainty. On the contrary, if the doctor is afraid of misdiagnosing and asks her to have a thorough examination, she will face the conservative risk of wasting time on useless examinations due to the high risk of sudden heart attack. Our goal is to obtain the prediction by balancing these two risks.

We address this risk problem by designing a *local Bayes risk minimization (LBRM)* framework. Rather than setting a global loss function in the traditional Bayes risk framework, we replace it with different uncertainty for the two risks at the current decision node. Based on these two risks, we decide whether to permit a sample going down to the next lower level or not. This process starts at the root node recursively until a stopping decision is made or a leaf node is reached. To measure the uncertainty on children nodes in the next lower level, we utilize information entropy on all the posterior probabilities of them, as illustrated in Fig. 2(b)(c). For the measurement of the uncertainty on stopping at the current node, we develop the idea of information gain by Deng et al. [17] by calculating the decrease on numbers of leaf nodes from the current node to its children nodes, shown in Fig. 2(d). To obtain the prediction of minimum risk emphasizing differently on the two risks, we propose a weighted tree induced error (WTIE) which extends the tree induced error [22] by adding coefficients on corresponding precipitance error and conservative error. The contributions of this work are summarized as follows.

- We introduce the risk problem in stopping strategy for hierarchical classification which seeks a balance between the precipitance risk and conservative risk.
- We propose a local Bayes risk minimizing framework (LBRM) which replaces the traditional loss function in Bayes risk framework with the uncertainty in precipitance risk and conservative risk on each decision node.
- We introduce information entropy to measure the uncertainty on children nodes in the next lower level in stopping strategy of hierarchical classification.
- We design a new metric weighted tree induced error (WTIE) which puts different emphasis on precipitance error and conservative error.



(a) A simple tree hierarchy.

(b) A prediction process with high information entropy.

(c) A prediction process with low information entropy.

(d) The loss of information gain at a certain node.

Fig. 2: Illustration of how to measure the two risks. The numbers in (b) and (c) are the posterior probabilities of the nodes, while the numbers in (d) is the information gain of each node. The smile face represents the final decision.

The rest of this paper is organized as follows. Section II reviews some related work briefly. Section III introduces preliminary knowledge for this paper. Section IV shows the details of the proposed LBRM algorithm and how to measure the uncertainty in the precipitance risk and the conservative risk. Experimental results on various datasets are presented in Section V. Finally, we conclude our work in the last section.

II. Related Work

There are some related studies on the issue of stopping strategy in hierarchical classification. Sun et al. [13] introduce the stopping strategy as predicting on all nodes in the hierarchy. To decide whether to stop at the internal nodes or leaf nodes, thresholds are set to each class node in the hierarchy. If the posterior probabilities or confidence scores are greater than the threshold on the corresponding class, the samples are sent to the classifiers on next lower level. Roughly speaking, they can be categorized into three classes.

Firstly, some researches try to avoid misclassification by predicting categories with coarser granularity. Selective rejection prediction [18], [19], [20] refuses to classify when the classifier is confused. Deng et al. extend it to hierarchical classification for comparison experiments in [17]. This algorithm predicts the leaf node class with the largest probability if it is not below a global threshold and stops at the root node otherwise. These strategies often make "right" decisions because all of the ancestor classes can be regarded as generalization of the leaf-node class. However, too conservative prediction, such as predicting the sample on the root node, takes high risk in losing information, as discussed above.

To overcome this problem, the second kind of strategy encourages samples to go more deeply in the tree hierarchy. Sun et al.[14] and D'Alessio [21] achieve this goal by reducing the thresholds of upper levels optimized by flat F1 score. The differences between these two algorithms is that [14] uses one threshold per level and optimizing macro F1 score, while [21] assigns one specific threshold to each node and optimizes TP (True Positive) minus FP (False Positive) score. Ceci et al.[15] develop the reject option to all of the class nodes in hierarchy instead of the root node only. Although these methods provide predictions with more information, they are inclined to make more wrong decisions in the process of letting samples going down through the hierarchy. If information is insufficient or huge uncertainty exists, predictions with this kind of stopping strategy will take precipitance risk in classification.

To solve the problems of the two kinds of methods above, a balance is sought in some researches. Deng et al.[17] propose Dual Accuracy Reward Trade-off Search (DARTS) algorithm which optimizes trade-off between hierarchical accuracy and specification described as information gain in the paper. By maximizing the information gain while maintaining a given hierarchical accuracy, this work integrates the advantages of both the first and the second kind of strategies. Ceci et al.[16] optimize the tree instance (or tree induced error [22]), which measures the distance between the prediction and the ground truth. However, all of these researches ignore the uncertainty of the next lower level on predicting a sample at the current level class, where the error is inclined to occur.

Furthermore, our work is also related with some methods applied in other hierarchical classification issues. Stopping the classification at a specific node can also be considered as a problem of finding the best path in the hierarchy from the root to the leaves. Sun et al. [23] and Qu et al. [24] address this problem by calculating the scores of each path. Specifically, they select the candidate paths of a required number from the first level to the current decision level. But, they just aim to solve mandatory leaf nodes prediction problem [12] that all the predictions are the leaf nodes and cannot address the problem of stopping strategy.

Besides, hierarchical multi-label classification has relationship with our work. Cesa et al. [25] use H-Loss as an optimization metric in threshold learning. This method can make prediction more conservatively, as it penalizes only the error nodes at the uppermost level for multiple labels of one sample. Triguero et al. utilizes H-Loss for threshold learning in [26]. To solve the disadvantage of H-Loss, Bi et al.[27] proposes a new loss function called hierarchical multi-label classification loss (HMC-Loss), which takes all the error nodes into account. HMC-Loss is applied in a Bayes risk framework to find the final prediction class, but it is not suitable for stopping strategy in hierarchical classification. Some other loss functions used in multi-label classification are discussed comprehensively in [28]. But these methods are all designed to solve multi-label classification problem. As for the field of stopping strategy in hierarchical classification, they just encourage all samples to go to leaf nodes as the second strategy we have introduced before.

III. Preliminaries

A. Class Hierarchy

There are two kinds of structures in class hierarchy, tree and directed acyclic graph (DAG). We focus on tree hierarchy for easy understanding.

A tree hierarchy organizes the class labels into a tree-like structure to represent a kind of "IS-A" relationship between labels [29]. Specifically, Kosmopoulos et al. points out that the properties of the "IS-A" relationship can be described as asymmetry, anti-reflexivity and transitivity [30]. We define a tree as a pair $(D, \prec)$, where $D = \{d_1, d_2, ...\}$ is the set of class labels and "$\prec$" denotes the "IS-A" relationship. The three properties of this relationship are formulated as follows:

(1) Asymmetry: if $d_i \prec d_j$ then $d_j \not\prec d_i$ for $\forall d_i, d_j \in D$;

(2) Anti-reflexivity: $d_i \prec d_i$ for $\forall d_i \in D$;

(3) Transitivity: if $d_i \prec d_j$ and $d_j \prec d_k$, then $d_i \prec d_k$ for $\forall d_i, d_j, d_k \in D$;

Generally, there are several types of nodes in a tree hierarchy. For node d_i:

(1) Its parent node is denoted by p_i;

(2) Its children nodes is denoted by C_i, and $|C_i|$ denotes the number of children nodes of d_i;

(3) Its ancestor nodes is denoted by $An(d_i)$, and $|An(d_i)|$ denotes the number of ancestor nodes of d_i;

(4) Its leaf nodes are denoted by $Le(d_i)$, and $|Le(d_i)|$ denotes the number of leaf nodes of d_i. Specially, L denotes the leaf nodes of the tree, and $|L|$ denotes the number of all leaf nodes.

B. Hierarchy Constraints and Augmented Set of True Classes

In hierarchical classification, for a given sample $s_i = (x, y)$, x is the sample data, and y is the corresponding label. In a hierarchy, labels in higher levels represent more general classes and labels of lower levels correspond to more specific classes. If a given sample belongs to a certain class, it must belong to the ancestor nodes of this class, i.e., if $s_i(x) = y$ then $s_i(x) \in An(y)$. To this end, it is intuitively right to assign a sample to its ancestor classes of the true class. [30] defines this as augmented set of true classes, i.e., $Y_{aug} = An(y)$.

C. Information Gain in Hierarchical Classification

Information gain measures the decrease of uncertainty by adding some information to the decision process. In hierarchical classification, it can be described as the decrease in number of nodes when taking a step forward from the current level to the next introduced by Deng et al. [17]. The information gain in hierarchical classification at node v can be defined formally as the following formula:

$$\begin{aligned} I(v) &= H(Y) - H(Y|v) \\ &= \log|L| - \log|L_{(}d_v)|. \end{aligned} \tag{1}$$

As we assume that the true labels are all at the leaf nodes and all the nodes are of equal significance, the uncertainty can be measured by the number of the corresponding leaf nodes.

D. Bayesian Decision Theory in Hierarchical Classification

Bayesian Decision Theory [31] calculates risks of all possible actions by multiplying posterior probabilities and loss function. The former comes from Bayes formula and the latter is set according to different scenarios. Typically, in hierarchical classification, we calculate the risks of all possible prediction nodes and choose the one of the minimum risk.

Formally, given the true label y, predicted label $\widehat{y}$, nodes set D in tree T and data x at node v, the loss function can be written as $l(y, \widehat{y})$, where $\widehat{y}$ can be any node in D.

The posterior probability of the predicted node through the corresponding classifier is denoted as $P(y|x)$. Thus we can obtain the Bayesian risk function:

$$R(\widehat{y_v}) = \sum_{y} l(\widehat{y}, y)P(y|x) \tag{2}$$

For all the possible predicted nodes, we evaluate the risk for each node and predict the node with minimum risk:

$$\widehat{y} = \text{argmin}_{\widehat{y_v}}(R(\widehat{y})), \tag{3}$$

where $v \in D$.

IV. Local Bayes Risk Minimization

A. Measuring uncertainty in prediction

Traditionally, a global loss function is applied in Bayes risk framework to measure risk. However, In the scenario of stopping strategy, uncertainty brings risks at each decision node in stopping or going down. The loss on each node should be treat differently.

On one hand, we observe that misclassification often occurs if the posterior probabilities of several classes in the next lower level are very close. In this scenario, the classifier is hard to distinguish them successfully. Recall the example in Fig. 1, if we just know that the patient has abdominal discomfort, it is not easy to distinguish whether to guide her to department of digestive or department of cardiology. This is a low-level classification uncertainty for classifier on the current node. Information entropy can measure the uncertainty that a discrete random variable contains. Given a discrete variable x with possible values $\{x_1, x_2, \cdots, x_n\}$, the information entropy is explicitly written as:

$$H(x) = \sum_{i=1}^{n} -p(i)\text{log}p(i), \tag{4}$$

where $p(i)$ is the probability of the value x_i.

In our case, the discrete random variable is the posterior probabilities of all children nodes provided by the classifier of the parent node. Formally, given a nonleaf node v and the sample x, the classifier C_v at node v provides probabilities $\{p_1, p_2, \cdots, p_i\}$, corresponding to ith children node $c(i)$ of the node v. The information Entropy of node v is:

$$L_v^M(x) = \sum_{i=1}^{|C_v|} -p(c(i))\text{log}p(c(i)). \tag{5}$$

On the other hand, if we make a prediction of stopping a sample at the current node to maintain great uncertainty, information provided by the lower level node will be lost. This uncertainty of conservative prediction brings risk to our prediction. Recall the example in Section I, if the patient with a heart attack be sent to the inpatient department, she will have huge uncertainty on her sickness. We can use information gain from the current node to the next lower level node to measure this uncertainty. This uncertainty of conservative prediction can be written as:

$$L_v^G(x) = I(u) - I(v), \tag{6}$$

where $I(u)$ is the information gain of the children node from the root node, $I(v)$ is the information gain of the current node from the root node.

B. Local Bayes Risk Minimization (LBRM)

A top-down hierarchical classification process is obtaining a prediction at each decision node until a leaf node is reached. Thus we can design a stopping strategy by dividing this process into a recursive binary decision on each node, i.e, stopping or going down. For each decision node v, if we consider all children nodes of equal importance, the candidate children nodes can be narrowed to the node of maximum posterior probability. To this end, we propose a Local Bayes Risk Minimization (LBRM) framework, which utilizes Bayesian decision theory to choose the action of the minimum risk. By balancing precipitance risk and conservative risk, it decides if the sample is stopped at the current node or sent to the children node of the maximum probability.

In LBRM, we need compare the precipitance risk and the conservative risk at each decision node. For the precipitance risk, we measure its uncertainty with information entropy. At each nonleaf node v, the loss function of precipitance risk is computed with (5), and we can obtain the risk integrated into Bayesian decison theory based on (2) and (3):

$$R_v^M(x) = [\sum_{i=1}^{|C_v|} -p(c(i))\text{log}p(c(i))]p(u|x), \tag{7}$$

where $p(u|x)$ is the maximum posterior probability of the children node in C_i, and u is the corresponding node in C_i as there are only two action nodes in the transformed problem.

For the conservative risk, we compute information gain from the current node to the children node as the loss function:

$$R_v^G(x) = \log\frac{|L(d_v)|}{|L(d_u)|}p(v|x), \tag{8}$$

where $p(v|x) = 1 - p(u|x)$.To balance precipitance risk and conservative risk at one specific node, we add a coefficient λ to get the minimum risk. When a sample arrives at v, we can just use the transformed formula based on (1),(4), (5) and (6) to decide whether to let the sample goes down or stop at v:

$$R_v^t(x) = \log\frac{|L(d_v)|}{|L(d_u)|}p(v|x) + \lambda[\sum_{i=1}^{|C_v|}p(c(i))\log p(c(i))]p(u|x), \tag{9}$$

where $p(u|x) + p(v|x) = 1$. According to (9), we assign the sample x to the current node v if the value of (9) ≥ 0 or push it down to the children node of the maximum probability otherwise, i.e.,

$$\Phi_v(x) = \begin{cases} 0 & R_v^t(x) > 0 \\ 1 & \text{otherwise.} \end{cases} \tag{10}$$

Calculating (10) from the root node to leaf nodes recursively, we can obtain the final prediction.

C. Optimization

Parameter λ balances the precipitance risk and the conservative risk, thus we can optimize it to get predictions meeting the need of different scenarios for these two risks. Tree induced error (TIE) [22] measures the error of prediction by calculating the distance in tree between prediction and real label, which reflects the error degree in a tree hierarchy. Errors can be categorized into precipitance error and conservative error. The former measures the errors not in the augmented true label set, while the latter measures those in the set but on a more general class. In some scenarios, we need to treat these two kinds of errors differently. However, TIE regards these two types of risk equally important. To this end, we modify the tree induced error to the weighted tree induced error (WTIE), which can set different weights to the precipitance error and conservative error:

$$WTIE = \frac{1}{NS} * (\alpha T_M + \beta T_G), \tag{11}$$
$$\text{subject to} \quad \alpha + \beta = 2,$$

where NS denotes the number of sample, T_M denotes the TIE of prediction $\widehat{y}$ for the sample $x \notin Y_{aug}$, while F_G denotes $\widehat{y} \in Y_{aug}$. T_M and T_G measure the precipitance error and the conservative error, respectively. So we can put different emphasis on these two kinds of risk through α and β.

Typically, when $\alpha = \beta = 1$, the weighted tree induced error is the traditional tree induced error. When $\alpha = 0, \beta = 2$, the weighted tree induced error just penalizes the conservative error. When $\alpha = 2, \beta = 0$, the weighted tree induced error just penalizes the precipitance error.

In the prediction process, if we pursuit the decisions with sufficient information even maybe the wrong one, just take out the low-level classification uncertainty into account and all the samples will reside in leaf nodes. On the other hand, if we cannot stand any risk of misclassification, just ignore the low-level uncertainty. By optimizing the WTIE of different weights, we can obtain series of predictions which taking this two risks into account to different degrees.

Our goal is to optimize the WTIE while ensuring that the risk of the node in lower level is lower than that of its parent node. Especially, we regard the root node itself as its parent node. And the optimizing objective is:

$$\text{minimize} \quad WTIE, \tag{12}$$
$$\text{subject to} \quad R_v^t < R_{pa(v)}^t,$$

where $v \in$ node set $D = \{d_1, d_2, ..., d_n\}$.

It is pointed out by [32] that optimizing the TIE is not a convex optimization problem, so as to the WTIE. Furthermore, the variables in the objective function are discrete, which cannot be solved by convex optimization. Fortunately, this problem has a global optimal solution as the value of the WTIE first decreases when λ is small and then increases with the increment of λ. This is because when the λ is small the sample is pushed down to the lower-level nodes with much precipitance risk; when the λ is too large, the sample is blocked at the higher-level nodes with much conservative risk. Only balancing these two kinds of risk can get the prediction of minimum risk.

To this end, we turn to random optimization methods to obtain the global optimal solution, such as generic algorithm (GA). Inspired by the process of natural selection, GA starts from a population of randomly generated creatures and reproduces iteratively. In each iteration, the fitness of every creature in the population is evaluated. The fit individuals are stochastically selected from the current population, and the genome of each creature is modified to form a new generation. Iteration terminates when a solution is found that satisfies minimum criteria or the fixed number of generations set by user is reached [33].

In our case, the fitness function of GA is our objective function (12), and we can optimize it to find the global optimal solution. However, GA has a disadvantage that it converges slowly if the candidate search space is very large. To reduce the time cost of this process, we try to derive the bounds of parameter λ.

Recall (9) is the transformed risk function at a node v. If $R_v^t(x)$ of a sample is larger than 0, we push it down to the children node of the maximum probability and stop it at v. We can obtain a relative narrow search bound for optimization of parameter λ.

We summarize LBRM in Algorithm 1. Given the input data vector X, tree hierarchy T and trained classifiers CT, we first obtain all posterior probabilities $p(y|x)$ via CT, where $p_{root}(y|x) = 1$. We get the precipitance risk $R_v^M(x)$ and conservative risk $R_v^G(x)$ of the root node from (6) and (8), respectively (step 3-6). Then we get the decision of the root node by (9) and (10) with parameter λ (step 7). By recursively proceed this process, we obtain the final prediction. Given the user-defined α and β, we optimize parameter λ through GA algorithm until the maximum iteration number is reached. In the GA iteration process, we calculate the WTIE in each iteration and finally choose the parameter with the minimum WTIE. And we can obtain a set of predictions emphasizing on different risks through various combinations of α and β.

Algorithm 1 Local Bayes Risk Minimization (LBRM)

```
Input: data vector $X = \{x_1, x_2, ..., x_n\}$,
       tree hierarchy $T$, trained classifiers $CT = \{CT_1, CT_2, ..., CT_s\}$,
Output: predictions $P_{final}$,
Obtain all $p(y|x), y \in L$ except the root via $CT$.
for $v$ from the root to the leaves do
    Get $L_v^M(x) = \sum_{i=1}^{|C_v|} -p(c(i))\text{log}p(c(i))$,
    Calculate $R_v^M(x) = [\sum_{i=1}^{|C_v|} -p(c(i))\text{log}p(c(i))]p(u|x)$,
    Get $L_v^G(x) = I(u) - I(v)$,
    Calculate $R_v^G(x) = \log\frac{|L(d_v)|}{|L(d_u)|}p(v|x)$,
    Obtain $\Phi_v(x)$ through $R_v^t(x)$,
end for
Get the predictions $P_c$,
$bestWTIE = inf$,
$\alpha = a$,
$\beta = b$,
$P_{final}$ = $P_c$,
for user-defined $\alpha, \beta$ do
  while iteration number < max iteration number do
    use GA with lower bound $\frac{\log\frac{|L|}{|L|-1}}{-\text{log}C_m}$ and upper bound $\frac{\text{log}|L|}{H_{min}}$
    to find a optimal $\lambda$:
      Calculate $WTIE$,
      if $WTIE < bestWTIE$ then
        $bestWTIE = WTIE$,
        $P_final = P_c$,
      end if
  end while
end for
return $P_{final}$
```

Obviously, if we ignore the precipitance risk, the sample will go down until a leaf node is reached. That is to say, $\lambda = 0$ can be a lower bound for the search space. Now we consider the upper bound that makes the sample stop at the node on upper level. The optimal solution then can be obtained through GA with the search bound.

We aim to find the λ that makes (9) < 0, so we can transform the equation into:

$$\begin{aligned}\lambda &< \frac{-\log\frac{|L(d_v)|}{|L(d_u)|}p(v|x)}{[\sum_{i=1}^{|C_v|} p(c(i))\text{log}p(c(i))]p(u|x)} \\ &= \frac{-\log\frac{|L(d_v)|}{|L(d_u)|}p(v|x)}{H_{C_v}(x)p(u|x)}\end{aligned} \tag{13}$$

Information entropy measures the uncertainty of random variable X. In our case, the posterior probability of each c in C_v is such random variable that we can describe the uncertainty in it. As we know, information entropy reaches the maximum value if the random variable respects a uniform distribution, i.e., the posterior probabilities are equal of all the label nodes. Furthermore, the more the number of nodes is, the larger the information entropy is. Thus we can maximize the information entropy to obtain a tighter lower bound based on (13):

$$\begin{aligned}H_{C_v}(x) &= [\sum_{i=1}^{|C_v|} p(c(i))\text{log}p(c(i))]p(v|x) \\ &< C_m * \frac{-1}{C_m} * \text{log}C_m \\ &= -\text{log}C_m,\end{aligned} \tag{14}$$

where C_m is the maximum number of the children nodes that one node has in the tree.

Furthermore, to obtain a lower bound, we minimize the information gain loss of (8) by assuming $C_m = |L| - 1$. Thus the lower bound based on (13) and (14) is:

$$B_L = \frac{\log\frac{|L|}{|L|-1}}{-\text{log}C_m}. \tag{15}$$

Moreover, we need maximize (8) by assuming $C_m = 1$. However, we have to calculate the minimum value of information H_{min} through the sample x as the minimum value of information entropy is zero. Thus we can obtain the lower bound:

$$B_U = \frac{\text{log}|L|}{H_{min}}. \tag{16}$$

Given the upper bound and the lower bound, we can find the optimal λ in the large search space more efficiently.

V. Experiments

A. Datasets

We perform our experiments on four datasets with tree-structured hierarchies (TABEL I):

- VOC [34]: It is a PASCAL visual object classes dataset which is a benchmark in visual object category recognition and detection. The tree hierarchy in this dataset is five-level.
- Cifar-100 [35]: This is an image datasets containing 60000 samples in 100 classes, with 600 images in each class. Its tree-structured hierarchy has three levels and 21 internal nodes with no samples.
- SUN [36]: This is a scene understanding datasets with 397 kinds of scenes. The original dataset has a completed four-level tree taxonomy. In the tree taxonomy, the second level contains 3 superordinate categories and the third level has 15 basic-level categories. We modify it by leaving out the categories that has more than one parent labels and samples with multiple labels. Finally SUN dataset turns into 324 classes with at least 100 images per category.
- ILSVRC65 [17]: This is a subset of ImageNet with 65 classes and four-level tree taxonomy.

TABLE I: Datasets description.

Datasets	#Sample	#Class	#Leaf	Depth
VOC	34828	30	20	5
Cifar-100	60000	121	100	3
SUN	90212	343	324	4
ILSVRC65	17100	65	57	4

B. Implementation

Data preprocessing. To represent the images, we use the GIST features for VOC and cifar-100 dataset, LLC features from densely sampled SIFT for ILSVRC65 dataset, VGG16 features for SUN dataset.

Classifier training. We train a logistic regression multi-class classifier for each nonleaf node and use liblinear toolbox [37] for implementation. For classification scheme, we follow the top-down fashion that a given sample traverses the tree hierarchy from the root node to one of the leaf nodes.

Dataset split. We split each dataset into training subset, validation subset and testing subset randomly, using 64%, 16%, 20% of data, respectively.

All experiments are executed on an Windows 8 operating system of Intel Core i7-600 running at 3.40GHz with 16 GB memory.

C. Comparison Methods

Dual Accuracy Reward Trade-off Search (DARTS): it maximizes the information gain while maintaining a certain hierarchical accuracy. Specifically, it first finds the search bound which ensures a hierarchical accuracy threshold preset by users, and then searches for the solution of the maximum information gain until the maximum iteration number is reached.

Threshold-TIE (TH-TIE): TH-TIE is a threshold-based method which takes hierarchy information into account by optimizing the traditional tree induced error [16]. It selects thresholds from a candidate set consisting of posterior probability values. As it is of high computational cost to find a global optimal solution in threshold-learning-based algorithm, we learns a specific threshold for nodes at the same level. By limiting the bounds of all candidate threshold values, we obtain the predictions of different TH-TIE with dynamic values of hierarchical accuracy.

Threshold-Rejection (TH-REJ): TH-REJ is a threshold-based method developing from the selective rejection method of flat classification [17]. It sets a global threshold and predicts the root node if the posterior probability of the prediction at the leaf node lower than this threshold. Obtaining different values of hierarchical accuracy with this algorithm can be achieved by setting different bounds of the global threshold.

Standard Top-Down Hierarchical Classification (STD): This is the standard and classical hierarchical classification with all the labels residing in the leaf nodes. At each nonleaf node from the root, STD predicts a sample to a children node of the maximum posterior probability and proceeding this process recursively until a leaf node is reached.

Rejection-Root (REJ-RT): REJ-RT makes conservative decisions by only predicting the root node for all of the samples.

D. Evaluation Metrics

In the scenario of hierarchical classification, hierarchy information should be taken into account when evaluating the performance of algorithms, instead of metrics applied in flat classification. Specifically, we use three hierarchical metrics as follows:

Weighted Tree Induced Error (WTIE): WTIE is introduced in Section IV, and we use it to measure the degree of errors with different emphasis on the precipitance error and the conservative error.

Hierarchical Accuracy (HA): HA adds all the ancestor nodes of the real label nodes to the real label set, i.e. the real label nodes and its ancestors are considered to be ground truth [17].

Hierarchical F1 Score (HF): HF extends traditional F1 score to a hierarchical version. It extends the real label and predicted labels to corresponding augmented set, respectively. It represents the overall performance on both hierarchical accuracy and information obtained in predictions [30].

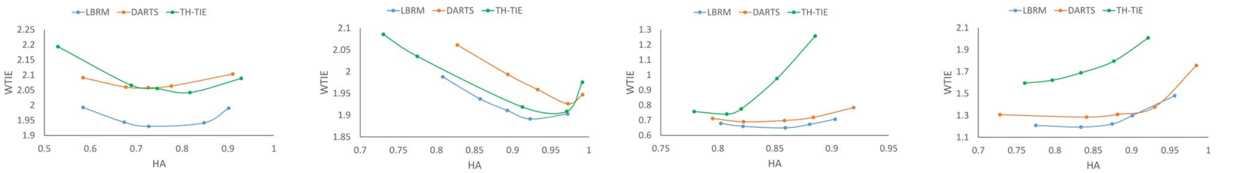
E. Experimental Results and Discussion

We compare all algorithms in four datasets with tree hierarchy. Note that we try to focus on the risk problem of stopping strategy in hierarchical classification, thus interpret the issue on tree hierarchy only. Actually our algorithm can deal with the DAG hierarchy as well. We first transform the proposed WTIE to the traditional TIE by setting both the parameters α and β as 1.0 for the common case and then test with different combinations of α and β . We run all the algorithms including the proposed one and the comparisons ten times, and calculate the average mean value of all metrics.

Results of the best WTIE on four datasets. From TABLE II we can see that the WTIE of STD and REJ-RT are both significantly higher than other algorithms. This result infers that it is dangerous for predicting either too informative or too conservative. On one hand, More informative predictions pushes all samples to the leaf nodes. However, it is challenging and struggling for distinguishing all the fine-grained classes of high uncertainty in the whole predicting process. On the other hand, the WTIE is also very high if the predictions are too conservative, as the conservative error is dominant overall even the precipitance error is zero. Furthermore, TH-REJ also performs poorly by assigning samples to the root node or one of the leaf nodes, as shown in TABLE II. This indicates that it is ineffective for only leveraging the root node and leaf nodes bring high risk on the predictions. As the performance of STD, REJ-RT and TH-REJ are poorer than others obviously, we will compare the state-of-the-art algorithms DARTS, TH-TIE and LBRM only.

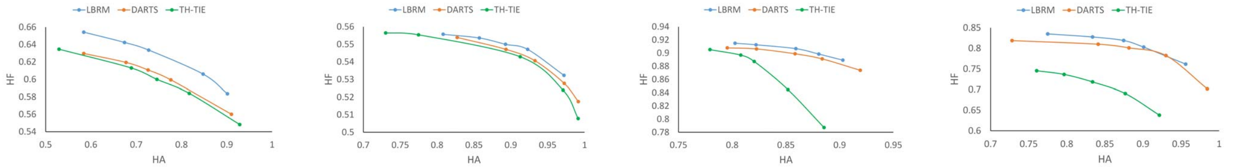
Results on VOC dataset. VOC dataset has a deep five-level tree hierarchy. It is essential and representative for testing the performance of stopping strategy as performance gap becomes larger on the hierarchy of a deep tree.

Fig. 3 is the result of Weighted Tree Induced Error - Hierarchical Accuracy curve (WTIE-HA). This curve can reflect the overall performance in terms of hierarchical accuracy and errors in a tree structure. Given a certain hierarchical



(a) VOC (b) Cifar-100 (c) Sun (d) ILSVRC65

Fig. 3: WTIE-HA curve of three algorithms on four datasets ($\alpha = \beta = 1$).



(a) VOC (b) Cifar-100 (c) Sun (d) ILSVRC65

Fig. 4: HF-HA curve of three algorithms on four datasets ($\alpha = \beta = 1$).

TABLE II: Results of the best WTIE on four datasets ($\alpha = 1, \beta = 1$).

Algorithms	VOC	Cifar-100	SUN	ILSVRC65
STD	2.4175	2.8575	0.7994	1.8535
REJ-RT	2.2697	2.0000	2.0000	3.0000
DARTS	2.0562	1.9257	0.6899	1.2263
TH-TIE	2.0420	1.9085	0.7411	1.4754
TH-REJ	2.2680	1.9445	0.7979	2.2890
LBRM	**1.9136**	**1.8880**	**0.6450**	**1.1939**

accuracy value by a set of predictions, we can find the corresponding WTIE value. The smaller the area enclosed by a curve and coordinate axes, is the better the performance of an algorithm is. Fig. 4 is the result of Hierarchical F1 Score curve - Hierarchical Accuracy (HF-HA). This curve can describe the overall performance of both specificity and accuracy in hierarchical classification. Neither too conservative nor too aggressive but wrong decisions will get better evaluations on this metric. The curve with a larger area enclosed by a curve and coordinate axes performs better that which with a small one.

We can see from Fig.3(a) that our proposed LBRM algorithm outperforms the DARTS algorithm and TH-TIE algorithms 6.3% for the best performance. Different from the static standard metrics in hierarchical classification, the point of minimum risk in the curve varies from different algorithms. For example, the optimal WTIE reaches the point where the HA gets lower than 75% in DARTS and our proposed algorithm, and larger than 75% in TH-TIE algorithm. Furthermore, DARTS outperforms TH-TIE where the HA is larger than 73%, inferring that DARTS is used in the application requiring high accuracy. Our LBRM outperforms those two state-of-the-art algorithms obviously for all hierarchical accuracy required.

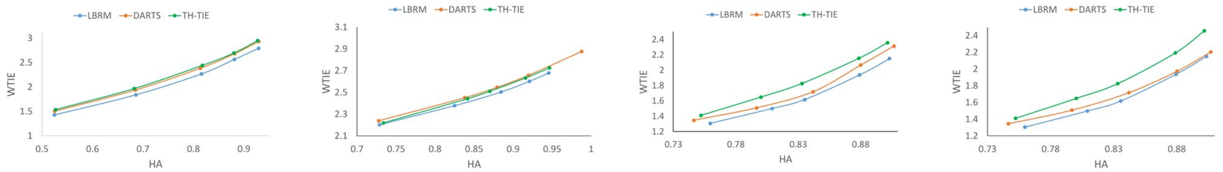
Fig. 4(a) demonstrates that the overall performance of LBRM is better than the two state-of-the-art algorithms. result shows that LBRM algorithm is neither not too precipitate but wrong nor too conservative. Note that the HF-HA curve keeps going down in the figure with the increase of hierarchical accuracy because the hierarchical F1 score weighs the precision and recall equally, the decrease in recall exceeds the increase in precision. Besides, the HF-HA curve first goes up and then goes down. We see the decreasing trend in the figure only because of the intervals we select. The result of a typical example is shown in Fig. 7.

Results on Cifar-100 dataset. Cifar-100 dataset is a large dataset with three-level tree hierarchy, thus the result can reflect the performance on a large dataset with shallow tree hierarchy. From Fig. 3(b) and TABLE II we can see that our algorithm LBRM performs better than the other two algorithms in general. Furthermore, the minimum WTIE is better than the other two algorithms by at least 1.1%. Fig. 4(b) demonstrates that the overall performance of our algorithm also outperforms DARTS and TH-TIE.

Results on SUN dataset. Testing on SUN dataset can reflect the performance on a large dataset with deeper tree hierarchy. The WTIE-HA curve and the HF-HA curve are shown in Fig. 3(c) and Fig. 4(c), respectively. It is clear that our algorithm is better than the other two comparison algorithms not only in the WTIE but also in the HF score.

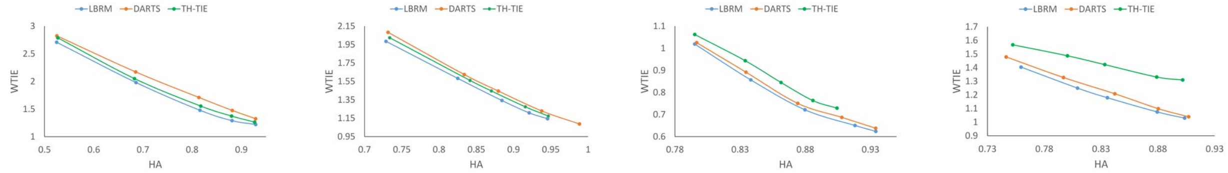
Results on ILSVRC65 dataset. It is a subset of ImageNet which can reflect the features of this large-scale dataset. The results can reflect the performance on a large-scale dataset to some degree. Experimental results of the WTIE-HA curve and the HF-HA curve are shown in Fig. 3(d) and Fig. 4(d), respectively. Our LBRM shows advantages on both of risk and overall performance over the other two algorithms on most of the intervals of HA. Different from the results above, the performance of DARTS is a little better than ours where hierarchical accuracy is at 93%.

Results on various risks. The WTIE is proposed in this paper to weigh differently on the precipitance error and the



(a) VOC. (b) Cifar-100. (c) Sun. (d) ILSVRC65.

Fig. 5: WTIE-HA curve of four datasets ($\alpha = 0.5, \beta = 1.5$). Results show the performance emphasizing on conservative error.



(a) VOC. (b) Cifar-100. (c) Sun. (d) ILSVRC65.

Fig. 6: WTIE-HA curve of four datasets ($\alpha = 1.5, \beta = 0.5$). Results show the performance emphasizing on precipitance error.

conservative error. We test LBRM, DARTS and TH-TIE on four datasets with $\alpha = 0.5, \beta = 1.5$ and $\alpha = 1.5, \beta = 0.5$ to show the performance of the three algorithms on different scenarios of errors. The results are shown in Fig. 5 and Fig. 6, respectively. The WTIE-HA curve demonstrates that our proposed LBRM algorithm is better in general.

VI. Conclusions and Future Work

In this paper, we have introduced the risk problem of stopping strategy in hierarchical classification containing the precipitance risk and the conservative risk. We propose a local Bayes risk minimization framework (LBRM) to address this problem. For each decision node, it balances these two risks based on Bayesian decision theory. The final prediction can be obtained by proceeding this process from the root node recursively. Uncertainty is used to replace the global loss function in the traditional Bayes risk minimization framework as it can describe the risk taken in the predicting process more appropriately. We utilizes information entropy to measure precipitance risk, which takes the information of the next lower level children nodes into account advantaging over other methods. For conservative risk, the loss of information gain from the current node to its children nodes is used to measure the uncertainty of stopping the sample going down. As we need emphasize differently on the two risks in various applications, weighted tree induced error (WTIE) is proposed to address this issue. Experiments on several datasets show the effectiveness of our method compared to the state-of-the-art algorithms.

It is time-consuming to find the global optimal solution and can only reach a local optimum solution sometimes in this work. In the future, we will design a more efficient optimization method for the WTIE.

Acknowledgment

This work was partly supported by National Natural Science Foundation of China under Grants 61432011, U1435212 and 61703196, and Key Project of Natural Science Foundation of China under Grant 61732011.

References

[1] J. Deng, W. Dong, R. Socher, L. J. Li, and F. F. Li, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 248–255.

[2] S. Oh, "Top-k hierarchical classification," in *Proceedings of the Thirty-First Conference on Artificial Intelligence (AAAI)*, 2017.

[3] R. Cerri, R. C. Barros, and A. C. de Carvalho, "Hierarchical classification of gene ontology-based protein functions with neural networks," in *International Joint Conference on Neural Networks*, 2015, pp. 1–8.

[4] Y. Song and D. Roth, "On dataless hierarchical text classification," in *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014, pp. 1579–1585.

[5] H. Zhao, P. Zhu, P. Wang, and Q. Hu, "Hierarchical feature selection with recursive regularization," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2017.

[6] C. Luo, B. Ni, S. Yan, and M. Wang, "Image classification by selective regularized subspace learning," *IEEE Transactions on Multimedia*, vol. 18, no. 1, pp. 40–50, 2015.

[7] T. Hoyoux, A. J. Rodrlguez-Snchez, J. H. Piater, and S. Szedmak, "Can computer vision problems benefit from structured hierarchical classification?" *ACM Transactions on Knowledge Discovery from Data*, vol. 9, no. 3, pp. 1–23, 2015.

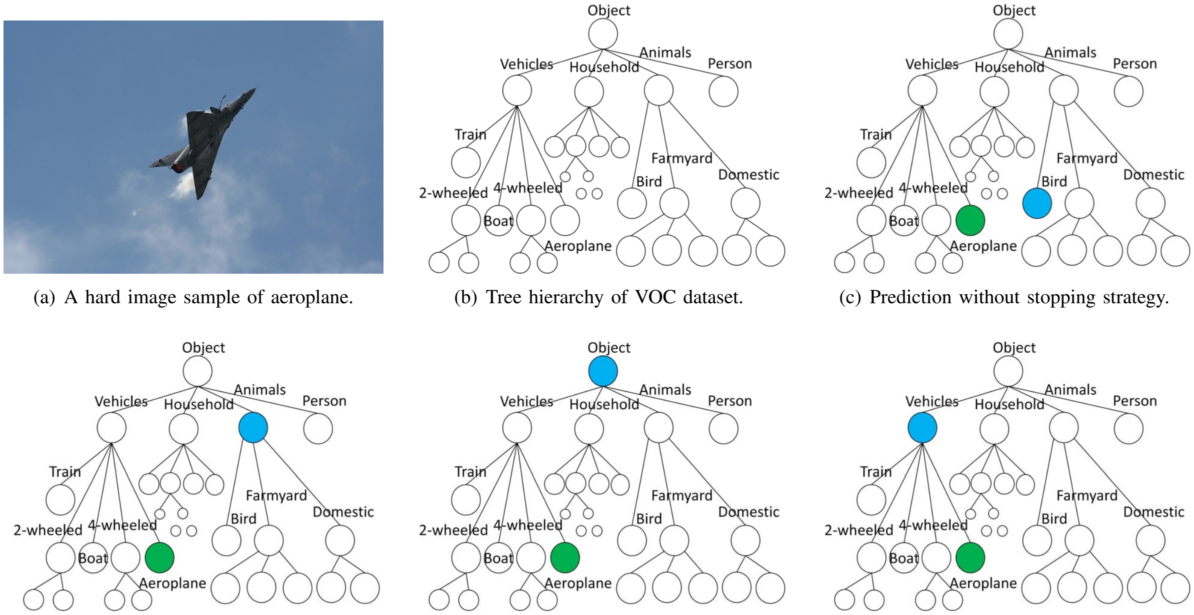
[8] S. Gopal and Y. Yang, "Hierarchical bayesian inference and recursive regularization for large-scale classification," *Acm Transactions on Knowledge Discovery from Data*, pp. 403–414, 2015.

[9] I. Dimitrovski, D. Kocev, S. Loskovska, and S. s. Dˇ zeroski, "Hierarchical annotation of medical images," *Pattern Recognition*, vol. 44, no. 10, pp. 2436–2449, 2011.

[10] C. Kurtz, C. F. Beaulieu, S. Napel, and D. L. Rubin, "A hierarchical knowledge-based approach for eving similar medical images described with semantic annotations," *Journal of Biomedical Informatics*, vol. 49, no. C, pp. 227–244, 2014.

[11] Y. Otani Naoki and H. Kashima, "Quality control for crowdsourced hierarchical classification," in *IEEE International Conference on Data Mining (ICDM)*. IEEE, 2015.

[12] C. N. S. Jr and A. A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining and Knowledge Discovery*, vol. 22, no. 1, pp. 31–72, 2011.



(a) A hard image sample of aeroplane.

(b) Tree hierarchy of VOC dataset.

(c) Prediction without stopping strategy.

(d) Prediction of DARTS.

(e) Prediction of TH-TIE.

(f) Prediction of Ours.

Fig. 7: A hard sample from VOC dataset. The green node is the ground truth, and the blue nodes are the predictions of DARTS, TH-TIE and our proposed LBRM.

[13] A. Sun and E. P. Lim, "Hierarchical text classification and evaluation," in *IEEE International Conference on Data Mining*, 2001, pp. 521–528.

[14] A. Sun, E. P. Lim, W. K. Ng, and J. Srivastava, "Blocking reduction strategies in hierarchical text classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 10, pp. 1305–1308, 2004.

[15] M. Ceci and D. Malerba, "Hierarchical classification of html documents with webclassii," *Advances in Information Retrieval*, vol. 2633, pp. 57–72, 2003.

[16] M. Ceci and D. Malerba, "Classifying web documents in a hierarchy of categories: a comprehensive study," *Journal of Intelligent Information Systems*, vol. 28, no. 1, pp. 37–78, 2007.

[17] Jia Deng, Jonathan Krause, Alexander C. Berg and L. Fei-Fei, "Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3450–3457.

[18] E. Y. Ran and Y. Wiener, "On the foundations of noise-free selective classification," *Journal of Machine Learning Research*, vol. 11, no. 15, pp. 1605–1641, 2014.

[19] M. Yuan and M. Wegkamp, "Classification methods with reject option based on convex risk minimization," *Journal of Machine Learning Research*, vol. 11, no. 1, pp. 111–130, 2010.

[20] B. Hanczar and E. R. Dougherty, "Classification with reject option in gene expression data," *Bioinformatics*, vol. 24, no. 17, pp. 1889–1895, 2008.

[21] S. D'Alessio, K. Murray, R. Schiaffino, and A. Kershenbaum, "The effect of using hierarchical classifiers in text categorization," in *Content-Based Multimedia Information Access*, 2000, pp. 302–313.

[22] F. Esposito, D. Malerba, V. Tamma, and H. H. Bock, "Classical resemblance measures," *Analysis of Symbolic Data*, 2000.

[23] M. Sun, W. Huang, and S. Savarese, "Find the best path: An efficient and accurate classifier for image hierarchies," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 265–272.

[24] Y. Qu, L. Lin, F. Shen, C. Lu, Y. Wu, Y. Xie, and D. Tao, "Joint hierarchical category structure learning and large-scale image classification," *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, 2016.

[25] Cesa-Bianchi, Nicol, D. U. Itgentile, Claudio, U. Itzaniboni, Luca, D. U. Itcollins, and Michael, "Incremental algorithms for hierarchical classification." *Journal of Machine Learning Research*, vol. 7, no. 1, pp. 31–54, 2006.

[26] I. Triguero and C. Vens, "Labelling strategies for hierarchical multi-label classification techniques," *Pattern Recognition*, vol. 56, no. C, pp. 170–183, 2016.

[27] W. Bi and J. T. Kwok, "Hierarchical multilabel classification with minimum bayes risk," in *IEEE International Conference on Data Mining*, 2013, pp. 101–110.

[28] W. Bi, "Bayes-optimal hierarchical multilabel classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 2907–2918, 2015.

[29] F. Wu, J. Zhang, and V. Honavar, "Learning classifiers using hierarchically structured class taxonomies," *Abstraction, Reformulation and Approximation*, pp. 313–320, 2005.

[30] A. Kosmopoulos, I. Partalas, E. Gaussier, G. Paliouras, and I. Androutsopoulos, "Evaluation measures for hierarchical classification: a unified view and novel approaches," *Data Mining and Knowledge Discovery*, vol. 29, no. 3, pp. 820–865, 2015.

[31] T. Bayes, R. Price, and J. Canton, *An essay towards solving a problem in the doctrine of chances*. C. Davis, Printer to the Royal Society of London, 1763.

[32] O. Dekel, J. Keshet, and Y. Singer, "Large margin hierarchical classification," in *International Conference on Machine Learning*, 2004.

[33] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, 1994.

[34] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[35] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009.

[36] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 3485–3492.

[37] 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*, vol. 9, no. 9, pp. 1871–1874, 2008.