Collaborative Error Reduction for Hierarchical Classification

Shiai Zhu[a], Xiao-Yong Wei[b], Chong-Wah Ngo[a]

[a] *Dept of Computer Science, City University of Hong Kong, Kowloon, Hong Kong*
[b] *College of Computer Science, Sichuan University, Chengdu, China*

Abstract

Hierarchical classification (HC) is a popular and efficient way for detecting the semantic concepts from the images. The conventional method always selects the branch with the highest classification response. This branch selection strategy has a risk of propagating classification errors from higher levels of the hierarchy to the lower levels. We argue that the local strategy is too arbitrary, because the candidate nodes are considered individually, which ignores the semantic and context relationships among concepts. In this paper, we first propose a novel method for HC, which is able to utilize the semantic relationship among candidate nodes and their children to recover the responses of unreliable classifiers of the candidate nodes. Thus the error is expected to be reduced by a collaborative branch selection scheme. The approach is further extended to enable multiple branch selection, where other relationships (e.g., contextual information) are incorporated, with the hope of providing the branch selection a more globally valid, semantically and contextually consistent view. An extensive set of experiments on three large-scale datasets shows that the proposed methods outperform the conventional HC method, and achieve a satisfactory balance between the effectiveness and efficiency.

Keywords: Concept Detection, Large-scale Hierarchy, Error Propagation

[*] Corresponding author: Xiao-Yong Wei

Email addresses: zshiai@gmail.com (Shiai Zhu), cswei@scu.edu.cn (Xiao-Yong Wei), cwngo@cs.cityu.edu.hk (Chong-Wah Ngo)

1. Introduction

With the increasing demand for organizing the large-scale image data effectively, image classification, a problem of labeling a target image with a set of predefined semantic concept label(s) [1, 2], has received intensive studies in the last decade. The task is usually simplified as a binary classification problem, in which a binary classifier is learnt for each concept and used on-the-fly to determine the semantic content of the target image. The binary classification scheme, even has been popularly employed and demonstrated encouraging performance, is impractical when facing large-scale datasets (e.g., ImageNet [3] which includes 21,841 concepts with each of them associated with 1,000 images), because all classifiers have to be called at runtime for every image.

To tackle the scalability issue, hierarchical classification is commonly adopted. Instead of applying all classifiers to a target image, hierarchical classification organizes concept classifiers into a hierarchy (e.g., Figure 1) according to the semantic relationship among concepts, and only selects a small set of classifiers for testing. The selection procedure usually starts from the root of the hierarchy and proceeds in a top-down manner that, for each node under investigation, hierarchical classification first applies the classifiers of the child nodes to the target, and then selects the child node(s) with the highest response(s) on its (their) classifier(s) as the next node(s) to investigate. The procedure repeats recursively and results in a path (paths) from the root to a leaf node (e.g., Animal-Fish-Salmon), on the basis of which all the concept labels on the path will be assigned to the target image.

By reducing the number of classifiers to be visited, hierarchical classification significantly improves the efficiency of multiple concept detection, and thus has been widely employed (e.g, web categorization [4, 5] and gene function prediction [6]). However, as pointed out by Bennett *et al.* [5], the improvement is paid at the price of sacrificing effectiveness of the classification. More specifically, the top-down classification procedure will make the classification errors included at the higher levels of the hierarchy be propagated to the lower levels, and in turn significantly degrades the accuracy of those leaf nodes. We argue that the major cause of the error propagation is the arbitrariness of the branch selection. It is well known that the performance of visual concept detectors is still not satisfactory in a general sense, and thus selecting a branch under a unreliable node may seriously ruin the classification procedure follows. In addition, the branch selection strategy is

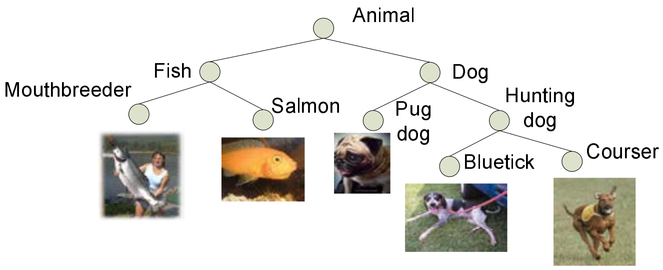


Figure 1: An example of image concept hierarchy. Instead of brute-force search of the best classifier, hierarchical classification performs search by traversing the hierarchy in a top-down manner, resulting in significance speed up but with the price of sacrificing classification accuracy.

lack of global perspective, in the way that the classification only focuses on the local responses of the nodes to be investigated, but never verifies if these responses are valid or globally consistent to other related concepts. With Figure 1 as an example, assuming that a target image includes a fish, the classification will be led to the branch under Dog if the classifier for Dog is unreliable and outputs higher response than that of Fish. When deciding next node to move on, the classifiers of the Pug Dog and Hunting Dog (if reliable enough) will output low responses. Globally speaking, this is conflicting to the semantic relationship in the hierarchy, in the sense that a parent node is with high response while all of its child nodes are with low responses. However, the procedure will never "doubt" the decision at the node Dog and continue to select between Pug Dog and Hunting Dog, resulting in further error propagation.

In this paper, we propose a novel branch selection scheme to address the arbitrariness of the branch selection. Instead of selecting the node(s) with the highest response(s), we introduce an error recovery scheme which first verifies the consistency between the observation of the candidate node and those of its semantically and contextually related concepts, and then adjusts the output of the node accordingly. The decision of which branch to go will be delayed when the verifications of all candidate nodes are finished, and the decision is made only when more observations are available and it is "confident" to do so. Compared with the highest-response-first strategy, this is more reliable because the proposed branching decision is made *collectively* by investigating the responses of the candidate node and its relatives instead of that of the candidate node alone. In the example mentioned above, the scheme is able to detect the inconsistency among the observation of Dog and those of its children nodes, and thus increase the chance of leading the classification to the correct branch (i.e., that under Fish in this example).

The main contribution of this work is the proposal of collaborative error reduction method for addressing the issue of error propagation in hierarchical classification, which is a problem rarely studied in the literature. The employment of proposed approach for both single-branch and multi-branch hierarchical classification methods is also demonstrated. Particularly, multiple concept relationships are novelly encapsulated in provision of a more globally consistent branch selection procedure. Eventually, a satisfactory balance between effectiveness and efficiency can be achieved.

2. Related Works

Although image classification has been intensively studied for more than one decade with numerous methods proposed, the majority of the efforts have been put on improving the *effectiveness* (i.e., how to construct effective learning mechanisms for building the classifiers [7, 8, 9, 10, 11]). Only until recently when the task is challenged by the scalability issue raised by the overwhelming growth of concepts and images, the attention starts shifting towards how to utilize inter-concept hierarchical relationship for improving the *efficiency* [12, 13, 14, 15, 16], which is also the focus of this paper. Therefore, we mainly review the works following this trend in this section. According to how a hierarchy is learnt, we roughly categorize these methods into two groups: pre-defined hierarchy and data-driven hierarchy.

Given the fact that there are many hierarchies developed for linguistic studies or information retrieval (e.g., WordNet[1]), borrowing these pre-defined hierarchies is popular in image classification [17, 18, 19]. Such a hierarchy is usually organized according to the semantic relationships (mostly "is-a" relation) among concepts, resulting in a tree-liked structure with the general concepts (e.g., vehicle) in the higher levels and the specific concepts (e.g., bus) in the lower levels. Marszalek *et al.* [18] perform the conventional hierarchical classification on a hierarchy extracted from WordNet according to two relationships (i.e., "is-a" and "is-part-of"). In [17], the authors propose a method for learning hierarchy aware classifiers. Instead of using 0-1 loss

[1] http://wordnet.princeton.edu/

as in conventional method, they utilize a loss function which assigns a lower loss for mis-classifying a training instance to a concept closer to the target concept. The hierarchy is used for measuring the distance between concepts. In [19], Deng *et al.* propose a selective flat classification framework, which only outputs classification results for classifiers that are the most informative concepts and with satisfactory accuracies. The hierarchy is used for measuring the specificity of a concept (because more specific a concept is, more informative it is).

Representative examples of pre-defined hierarchies include Enzyme Commission [20] and the Gene Ontology [21] for gene function prediction, Yahoo! Directory[2] for text categorization, and Caltech256 [22] hierarchy for image classification. The advantage of pre-defined hierarchy is that it is easy to obtain, and the structure is consistent to human perception, which makes the analysis of the classification results can be carried out intuitively, because it is easy to check whether each of the branch selections is correct. However, due to the *semantic gap*, the disadvantage of the pre-defined hierarchy is that the semantic relationships of some concepts may not be consistent to their visual relationships in the feature space, which makes the branch selection at a node with visually similar children (e.g., “Paris daisy” and “Easter daisy”) lack of discriminative-ness and be random.

In methods using data-driven hierarchy, the concept hierarchy is constructed by top-down [23, 24, 25, 26, 27, 28] or bottom-up [29, 30] hierarchical clustering with all the target concepts as leaf nodes. The inter-concept distance is generally defined on the distribution distance of their training examples in the low-level feature space (e.g., BoW [27, 26, 28]), and used as the metric for the clustering. The advantage of using data-driven hierarchy is that the resulting hierarchical structure has encapsulated the low-level relationships between concepts (i.e., how their visual appearances are similar or different from each other) and thus is able to provide a handy discriminative power for the classification. This is especially useful for filtering out the irrelevant concepts during the branch selection effectively and efficiently. However, the data-driven hierarchy may work awkwardly for a leaf node with diverse visual appearances (e.g., “vehicle”), because the node can only be linked to the branch representing one type of its visual characteristics (e.g., “car”) and thus examples (e.g., “boat”) with visual appearances vary-

[2] http://dir.yahoo.com/

ing from this branch will always be directed to other leaf nodes. Furthermore, the automatically generated inner nodes for a data-driven hierarchy are usually not associated with any semantic meanings, which makes the analysis of the classification results less intuitive.

By using either pre-defined or data-driven hierarchy, as aforementioned, the top-down branching process will impose error propagation. However, as one of the open questions for hierarchical classification, it is rarely addressed in the literature. Two works we found from very few examples are [4] and [5]. In [4], Xue *et al.* address the problem by selecting only a small number of nodes from the original hierarchy to construct a simplified hierarchy for classification. Therefore, the chance of error propagation will be reduced because the classification path from root to leaf node is significantly shortened. In [5], when building the set of negative instances for training each node, Bennett *et al.* also include the false positive instances which have been misclassified at its ancestor nodes, in the hope that those instances can be rejected by current classifier and the mis-classification errors will not be further propagated to the lower level nodes. Even those methods are able to improve the accuracy of hierarchical classification, the optimization schemes employed still have not addressed the arbitrariness of the branch selection which is the core issue of the error propagation.

In this paper, we argue that the arbitrariness of the branch selection is caused by the local view of the decision making, in which the concepts are investigated individually with its relationship to other concepts being ignored. Therefore, we propose a collaborative branch selection scheme which makes decision by further utilizing the semantic and contextual consistency of the concepts under investigation. The idea is primarily studied in [31], which only supports single branch selection that assigns an image only to one leaf node in the hierarchy, with the assumption that all the leaf nodes are exclusive to each other. The assumption is popularly employed in the existing methods (e.g., [24, 25, 26, 27, 28]) to ease the problem solving. However, in a real application, an image always contains several concepts that may not in the same branch because they are not necessarily being semantically related (e.g., car and road). In this paper, we extend the method to enable multiple branch selection[3]. Furthermore, we have incorporated the contextual relation

[3] In [32], singe branch selection is termed as single-label classification while multiple branch selection is termed as multi-label classification. We change the terminology in this

among concepts into branch selection for a more reasonable decision making process. In the following sections, we first present the basic framework of the proposed method in Section 3 with its application to singe branch selection. In Section 4, we further extend the framework to enable multiple branch selection.

3. The Basic Framework

In this section, we describe the basic framework of the proposed method under the scenario of single branch selection. Instead of considering only the candidate nodes, we also involve their children and siblings to form a committee for decision making. In practice, before deciding which branch to go, we adjust the response of each candidate node to be semantically consistent with those of other nodes in the committee, with the hope that the unreliable response of a candidate can be fixed if it is conflicting with those of its relatives in the hierarchy. Therefore, making decision on the adjusted response is with a more global view and can avoid the arbitrariness of the branch selection with the highest-response-first local strategy.

3.1. Problem Formulation

Given an instance (image) x and a node (concept) c_t as the current node in a hierarchy, we denote the branch to go at next moment as a node c_{t+1}, so that

$$c_{t+1} = \operatorname*{arg\,max}_{c \in \mathcal{N}(c_t)} \hat{f}_c(x) \tag{1}$$

where $\mathcal{N}(c_t)$ is a set of child nodes of c_t, and $\hat{f}_c(x)$ is the adjusted response of c to x. Further denoting the original response of c as $f_c(x)$, we can formulate the highest-response-first local strategy by replacing $\hat{f}_c(x)$ with the original response $f_c(x)$. Moreover, our committee for decision making is $\mathcal{T} = \bigcup_{c \in \mathcal{N}(c_t)} \bar{\mathcal{N}}(c)$, where $\bar{\mathcal{N}}(c) = \mathcal{N}(c) \cup \{c\}$. In other words, $\mathcal{T}$ is a union of c_t's children and its grandchildren. The problem to solve is then how to define the adjusting function $\hat{f}_c(x)$ with respect to both the semantical relationship and the observations of the nodes in the committee $\mathcal{T}$. Let us

paper, because multi-label classification is easy to be confused with the multiple label assignment in singe branch selection where concepts along the resulting branch are all assign to a target image.

denote the semantical relationship among nodes as $\Phi_{\mathcal{T}}$ and compose the observations of the committee $\mathcal{T}$ into a vector $\mathbf{f}_{\mathcal{T}}(x) = [f_{c_1}(x), f_{c_2}(x), f_{c_3}(x), \dots]$ where $c_1, c_2, c_3, \dots \in \mathcal{T}$, the problem can be formulated as

$$\hat{f}_c(x) = P(c|\Phi_{\mathcal{T}}, \mathbf{f}_{\mathcal{T}}(x)). \tag{2}$$

3.2. Committee-based Response Adjustment

Once the hierarchy is known, there are a lot of priori can be utilized for modeling $\hat{f}_c(x)$. For example, in the single-label hierarchy as shown in Figure 1, the siblings are semantically exclusive. If all the classifiers are reliable, the response for a candidate node (e.g., "Fish") should be approaching 1 if those of its siblings (e.g., "Dog") are all with responses close to 0. In addition, a parent node represents a union of the instances of its child nodes, so that the response for a candidate node (e.g., "Dog") should be close to 0 if those of its child nodes (e.g., "Pug dog" and "Hunting dog") are all with responses close to 0. In brief, confined by the semantic relationship $\Phi_{\mathcal{T}}$, the responses of nodes in a committee should always follow certain patterns. Unreliable classifiers will produce responses conflicting to the patterns within the nodes of committee. Therefore, we can use the observations of the committee $\mathcal{T}$ to predict that of a candidate node so as to implement the response adjustment.

Intuitively, this can be simply modeled by logistic regression, where we use the observations of the committee $\mathcal{T}$ as predictors for estimating a reasonable output for a candidate node c. The impacts of semantic constraints $\Phi_{\mathcal{T}}$ on predicting the response is then modeled by a set of weights (i.e., a weight vector $\mathbf{w}_c = [w_1, w_2, \dots])$ associated with the predictors (nodes in $\mathcal{T}$). A weight given to a predictor reflects the ability of the predictor to estimate the output of the candidate node. By further expanding the logistic regression to all the candidate nodes, we can learn their weights at the same time by multi-class regression (MCR), resulting in a weight matrix $\mathbf{W}$. It is worth mentioning that learning the weights together not only brings convenience for the learning but also makes the inter-concept relationship among candidate nodes be modeled during the learning. Thus the adjusted responses would follow the specific patterns embedded in the hierarchy. By replacing the semantic relationship $\Phi_{\mathcal{T}}$ with $\mathbf{W}$, Equation (2) can be implemented with MCR as

$$P(c|\mathbf{f}_{\mathcal{T}}(x), \mathbf{W}) = \frac{\exp(\mathbf{w}_c^T \mathbf{f}_{\mathcal{T}}(x))}{\sum_{c_k \in \mathcal{N}(c_t)} \exp(\mathbf{w}_k^T \mathbf{f}_{\mathcal{T}}(x))}, \tag{3}$$

where $\mathbf{w}_k$ is the weight vector for the corresponding candidate node $c_k \in \mathcal{N}(c_t)$. Equation (3) takes the responses of committee as input, and computes adjusted responses for the candidate nodes.

Given a set of training instances $\mathbf{X} = \{x_1, x_2, \dots\}$ with each of them associated with a class label $y_i \in \mathcal{N}(c_t)$, an optimal weight matrix $\mathbf{W}^*$ can be obtained by

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} - \sum_{x_i \in \mathbf{X}} \log P(y_i | \mathbf{W}, \mathbf{f}_{\mathcal{T}}(x_i)) + \lambda \|\mathbf{W}\|^2, \tag{4}$$

where the second term is a regularizer used to control the model complexity, and λ is regularization parameter. Equation (4) is referred to as L2-regularized MCR. This problem can be efficiently solved by Quasi-Newton method. In the experiment, we adopt the package released by Mark Schmidt[4]. The proposed method for hierarchical classification with single-branch selection is summarized in Algorithm 1.

3.3. Verification and Recovery

Since the labels given to each instance always follow certain patterns which reflect the inter-concept semantic relationship in the hierarchy, the resulting weight matrix **W*** in Equation (4) is also embedded with those relationship which can be used to verify if a set of responses is consistent to those patterns, in the way that it results in larger response in Equation (3) when it is consistent and smaller response otherwise. Note that, during learning, we put the candidate node itself in the committee, with the hope that the resulting weight can also reflect the classification reliability of the candidate node. According to the principle of MCR, if a candidate node is with an unreliable classifier, it will be assigned with a small weight to weaken its impact to the final results (i.e., the adjusted response Equation (3)), and the predication for its label will mainly rely on the responses of other nodes in the committee. By contrast, if the node is a reliable classifier, it earns a large weight so its impacts will dominate those of others. This also explains why we put the candidate node itself in the committee. Therefore, the proposed method fulfills the semantic relationship verification and error recovery at the same time by MCR.

[4] http://www.di.ens.fr/∼mschmidt/Software/code.html

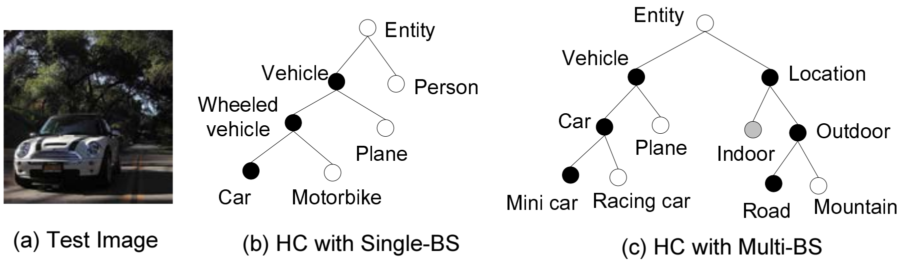
Algorithm 1 Error Reduction HC with Single Branch Selection (ER-SHC)

```
Input:
⋆ Testing instance $x$.
Initialization:
⋆ Set $t = 1$ and current node $c_t$ as root node.
⋆ Initialize the detected set of classes as $\mathcal{C} = \varPhi$.
Hierarchical classification
    1. Set $c_t$'s child nodes $\mathcal{N}(c_t)$ as candidate nodes.
    2. Construct a committee $\mathcal{T} = \{\mathcal{N}(c_t), \bigcup_{c \in \mathcal{N}(c_t)} \mathcal{N}(c)\}$, which consists of $c_t$'s child and grandchild nodes.
    3. Compute the responses of the classifiers in committee $\mathcal{T}$. The responses are composed into a vector $\mathbf{f}_{\mathcal{T}}(x) = [f_{c_1}(x), f_{c_2}(x), f_{c_3}(x), \dots]$.
    4. Get the adjusted responses of candidate nodes by using MCR model in Equation (3).
    5. Select a node $c$ from the candidate nodes for further investigating using Equation (1). Update the prediction results $\mathcal{C} = \mathcal{C} \cup \{c\}$.
    6. If $c$ is not a leaf node, set $c_t = c$, $t = t + 1$ and return to step 1.
Output
The prediction results $\mathcal{C}$.
```

3.4. Complexity Analysis

One may argue that the Error Reduction HC proposed in this paper is less efficient than the conventional HC, because an additional error recovery process is included. However, we will show that this is not a critical issue. In this section, we provide a theoretical analysis on the time complexity of the conventional HC and the Error Reduction HC. A more comprehensive empirical study will be given in Section 6.

First, let us define the $TC = 1 - \frac{\#model}{\#concept}$, where $\#model$ and $\#concept$ denote the number of activated classifiers and the number of concepts respectively. TC is the saved computational cost compared to standard one-vs-all approach. We further assume that a hierarchy is a binary tree with $2^{L+1} - 2$ nodes (or L levels). It can be easily calculated that $2L$ and $4L - 2$ classifiers will be activated by the conventional HC and our method respectively. Therefore, the TCs of the two HC methods are $1 - \frac{L}{2^L - 1}$ and $1 - \frac{2L-1}{2^L-1}$. We will see that the percentage of saved computational cost by using our method is still significant for large hierarchy with many levels. This is consistent with the results in Figure 3, which shows the TCs using conventional HC and our



(a) Test Image (b) HC with Single-BS (c) HC with Multi-BS

Figure 2: Given a test image (a), the hierarchical classification results with (b) single branch selection (Single-BS), and (c) multiple branch selection (Multi-BS). Single-BS assigns the concepts on a single path to the image, and Multi-BS labels the image with multiple paths. Correct and wrong assignments are marked as black and gray respectively.

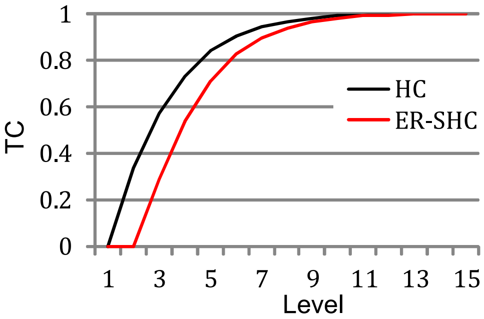


Figure 3: The saved computational costs of conventional HC and ER-SHC compared to one-vs-all approach on binary hierarchies with different number of levels.

method on hierarchies with different levels. We can see that the advantage of both approaches is more obvious for larger hierarchy in terms of efficiency. Furthermore, the two curves are closer for larger hierarchy. Thus the advantage of hierarchical classification can be maintained at large scale by using our proposed error reduction approach.

4. Extension to Multiple Branch Selection

In Section 3, to simplify the description, the basic framework is built only for supporting single branch selection (Single-BS). We will extend the framework to enable multiple branch selection (Multi-BS) in this section so as to increase its generalizability. The difference between Single-BS and Multi-BS is shown in Figure 2. In Single-BS, we assume the sibling nodes are semantically exclusive, and the classification result is thus a single path from root to leaf. For example, in Figure 2(b), the test image is annotated with “vehicle”, “wheeled vehicle” and “car”. However, Single-BS will cause missing labeling when some of the related concepts like “road” are also in the concept set. To address this problem, as shown in Figure 2(c), in Multi-BS, we select multiple branches, which result in a more comprehensive labeling for the target image with both semantically and contextually related concepts.

4.1. Combining Semantic and Contextual Relations for Multi-BS

The most intuitive way for combining the semantic and contextual consistency into the framework is to add the concepts that are contextually related to the target node into the committee, so that the weight matrix **W**, after being learnt, will carry both semantic and contextual relationships between the committee members and the target concept. Therefore, two questions needed to be answered are 1) how to find contextually related concepts? and 2) how to use the new committee to select multiple branches?

4.1.1. Selecting Contextually Related Concepts

Contextual relationship has been intensively studied in literature (e.g., [15, 33, 34]), resulting in a lot of metrics to measure the contextual similarity between two concepts. To select the contextual relatives for a given concept from a candidate concept set, we can simply use any of the existing measures to calculate its contextual similarities to the candidate concepts and select the top-k concepts with the largest similarities (or similarities exceeding a threshold). In this paper, we adopt Flickr Context Similarity (FCS) [35] for this purpose. It estimates inter-concept contextual similarity based on the statistics derived from tags associated with the images in Flickr. FCS is defined as:

$$FCS(c_i, c_j) = e^{-NGD(c_i,c_j)/\rho} \tag{5}$$

where

$$NGD(c_i, c_j) = \frac{\max\{\log h(c_i), \log h(c_j)\} - \log h(c_i, c_j)}{\log N - \min\{\log h(c_i), \log h(c_j)\}}. \tag{6}$$

Here NGD stands for Normalized Google Distance [36], $h(c_i)$ is the number of Flickr images associated with concept c_i, $h(c_i, c_j)$ is the number of images associated with both c_i and c_j, and N is the number of images indexed by Flickr. The function h is computed by querying Flickr API. According to [35], the parameter ρ is empirically set as average NGD among a set of randomly selected words. The contextually related concepts can then be selected based on their similarities with the target concept measured by Equation (5).

4.1.2. Multiple Branch Selection

With the new committee generated using both semantic and contextual relationships, we can use similar process as in Section 3.2 to adjust the output for each candidate concept. In Section 3.2, the adjustments for all the candidate concepts are conducted simultaneously within the framework of MCR because the candidate concepts are assumed to be exclusive to each other. In this section, we have to conduct the adjustment for each candidate concept individually, since the assumption of exclusion is no longer valid. To this end, a binary regression model (LR) is learnt for each concept c_{t+1} under current node c_t by using responses of its committee concepts as predictors. That is

$$P(c_{t+1}|\mathbf{w}_{c_{t+1}}, \mathbf{f}_{\mathcal{T}}(x)) = \frac{\exp(\mathbf{w}^T_{c_{t+1}}\mathbf{f}_{\mathcal{T}}(x))}{1+\exp(\mathbf{w}^T_{c_{t+1}}\mathbf{f}_{\mathcal{T}}(x))}, \tag{7}$$

where the relationship constraints $\Phi_{\mathcal{T}}$ is modeled by weight vector $\mathbf{w}_{c_{t+1}}$. Consequently, we can select the branch where the adjusted response of the candidate concept exceeds a threshold [16, 4, 5]. The selection criteria is

$$P(c_{t+1}|\Phi_{\mathcal{T}}, \mathbf{f}_{\mathcal{T}}(x)) > th_{c_{t+1}}, \tag{8}$$

where $P(c_{t+1}|\Phi_{\mathcal{T}}, \mathbf{f}_{\mathcal{T}}(x))$ is the adjusted response based on all the observations of committee members, and $th_{c_{t+1}}$ is the threshold for the new response. The proposed method with multi-branch selection is summarized in Algorithm 2.

4.2. Exclusion-aware Multi-BS

In this section, we have to release the mutually exclusive assumption among candidate concepts to allow multiple branch selection. It is worth mentioning that this may impose errors for the selection when the selected candidates are with (either semantically or contextually) exclusive relationship. For example, both “indoor” and “outdoor” might be assigned to the image in Figure 2(c), even these two concepts rarely appear in the same image. We believe that using the prior knowledge on the exclusive relation to remove this type of conflicting selections could help lead the classification to correct branch. Thus we propose an exclusion-aware Multi-BS method, where the exclusive relationship is explicitly considered. In specific, for the candidates with exclusive relationship (e.g., “indoor” and “outdoor”), only one candidate can be selected according to Equation (1). Otherwise, Equation (8) is adopted for selecting multiple nodes (e.g., “vehicle” and “location”) to investigate.

Algorithm 2 Error Reduction HC with Multiple Branch Selection Incorporating Contextual and Semantic Relationships (L-ER-MHC)

```
Input:
⋆ Testing instance $x$.
Initialization:
⋆ Put root node into a list $\mathcal{L}$.
⋆ Set the detected class set $\mathcal{C} = \Phi$.
Hierarchical classification
  1. Dequeue one node $c_t$ out of $\mathcal{L}$. Set $c_t$'s child nodes $\mathcal{N}(c_t)$ as the candidate nodes.
  2. Generate a committee $\mathcal{T}$ for each candidate node using its child and grandchild nodes, as well as the contextually related nodes according to Equation (5).
  3. For each candidate node, compute the responses of the nodes in its committee $\mathcal{T}$ by calling corresponding classifiers. The responses are composed into a vector $\mathbf{f}_{\mathcal{T}}(x) = [f_{c_1}(x), f_{c_2}(x), f_{c_3}(x), \ldots]$.
  4. Get the adjusted response for each candidate node by using Equation (7).
  5. Put the candidate nodes, which satisfy Equation (8), into the prediction results $\mathcal{C}$. Then, put the non-leaf nodes from the selected candidate nodes into $\mathcal{L}$ for further investigating.
  6. If $\mathcal{L}$ is not empty, return to step 1.
Output
The prediction results $\mathcal{C}$.
```

5. Experimental Setup

5.1. Datasets

The three image datasets are Caltech256 [22], ILSVRC1K [3] and NUS-WIDE [37]. Caltech256 consists of 256 labeled concepts for object annotation. In [22], a concept hierarchy is pre-defined using the 256 concepts as leaf nodes. ILSRVC1K is a subset of ImageNet, where the concepts are organized by WordNet. Starting from the 1,000 concepts in ILSRVC1K, a hierarchy is extracted from the ImageNet hierarchy. These two datasets are designed for Single-BS, and each image is only labeled with one leaf class. The third dataset used in this paper is NUS-WIDE, which is one of the largest fully annotated image datasets. NUS-WIDE is composed of 269,648

Table 1: Dataset statistics: number of leaf nodes (#Leaf) and internal nodes (#Int), depth of the hierarchy (#Dep), average number of instances of each concept for training (#Trn), validation (#Val) and testing (#Tst) respectively.

Dataset	#Leaf	#Int	#Dep	#Trn	#Val	#Tst
Caltech256	256	62	6	58	29	29
ILSRVC1K	1000	645	13	1261	50	150
NUS-WIDE	73	44	10	3722	1248	1250

Flickr images, which is divided into a training set (161,789 images) and a test set (107,859 images). It is manually labeled with 81 semantic concepts covering a wide range of topics from object to event. Starting from the 81 concepts, we extract a concept hierarchy by following the "is-a" relationship from WordNet. The constructed hierarchy supports multiple branch selection. Table 1 lists the statistics of three datasets and hierarchies. We follow the train/validation/test split in ILSRVC1K. For Caltech256, the instances of each concept are split to train/val/test by 50%-25%-25%. For NUS-WIDE, we split the original test set to a validation set and a test set by 50%-50%.

5.2. Implementation Details

Each image is represented using Locality-constrained Linear Coding (LLC) with densely sampled SIFT features [38]. We employ a visual vocabulary of 4,000 visual words, and three level spatial partitions (1×1, 2×2 and 3×1). Consequently, the dimension of feature vector is 32,000. For each node c_i, a classifier $f_{c_i}(x)$ is learnt using linear SVM [9] on training set. In addition, the logistic regression model used in our method is learnt on the validation set. The thresholds of local classifiers and LR models for Multi-BS are tuned by performing 5-fold cross validation. Following the method in [16, 5], we choose the threshold which optimizes F1 score (i.e., score-cut optimization [39]).

5.3. Evaluation Criteria

In HC with Single-BS, the test instances are all from leaf nodes which are mutually exclusive. Thus classification performance is usually measured using global accuracy among leaf nodes (*Acc*) [26], which is defined as the ratio of correctly classified instances. An instance is correctly classified when the annotated leaf node hits the ground-truth. In HC with Multi-BS, we adopt the standard *F1* measure by jointly considering the recall and precision. *F1* score is defined as:

$$F1 = \frac{2 \times Recall \times Precision}{Precision + Recall} \tag{9}$$

where

$$\begin{aligned} Precision &= \frac{\#\text{True Positives}}{\#\text{Predicted Positives}}, \\ Recall &= \frac{\#\text{True Positives}}{\#\text{Actual Positives}} \end{aligned} \tag{10}$$

Furthermore, the overall performance on several categories can be measured by using either macro-averaged or micro-averaged *F1* scores (denoted by *Macro-F1* and *Micro-F1* respectively). *Macro-F1* averages the *F1* scores computed separately for each class. *Micro-F1* is calculated using the binary predictions of all classes. For example, classifying m instances to n classes produces $m \times n$ binary predictions, from which *Micro-F1* can be computed with Equation (9) and Equation (10). *Macro-F1* and *Micro-F1* have been two standard evaluation criteria for classification [16, 5, 39]. In [40], they compare the two metrics in detail, and point out that *Macro-F1* tends to emphasize rare classes, while *Micro-F1* emphasizes common classes. In this paper, both *Micro-F1* and *Macro-F1* are used. Note that *F1* score can be used for both single-branch and multi-branch HC, while *Acc* is only suitable for single-branch scenario.

Similar to [26], where the efficiency of HC is measured using one-vs-all approach as baseline, we evaluate the efficiency using percentage of saved time cost compared to one-vs-all approach. Since we adopt linear SVM, the time cost is linear with the number of involved classifiers. Thus the efficiency is evaluated using the saved time cost TC defined in Section 3.4. In this case, TC of one-vs-all approach is 0. We further define MTC as the average saved cost over all test instances.

6. Results and Discussions

This section discusses the experimental results. The proposed methods for HC with single and multiple branch selections are verified respectively.

6.1. Performance of HC with Single-BS

We compare the following approaches for performance evaluation.

Table 2: Performance comparison of Flat and four HC methods with Single-BS on Caltech256 and ILSRVC1K. Classification performance is measured by global accuracy (*Acc*), *Macro-F1* and *Micro-F1*. The testing efficiency is measured by average saved time cost (MTC). The performance gain over baseline is shown in the parentheses.

Dataset		Flat	SHC (baseline)	SIB-ER-SHC	ER-SHC	SHC-Ref
Caltech256	*Acc*	0.376	0.267	0.270 (1.0%)	0.305 (14.0%)	0.279 (4.49%)
	Macro-F1	0.336	0.232	0.235 (1.3%)	0.275 (18.5%)	0.245 (5.60%)
	Micro-F1	0.556	0.485	0.489 (0.8%)	0.520 (7.20%)	0.501 (3.29%)
	MTC (%)	0	91.5	91.1	67.1	90.8
ILSRVC1K	*Acc*	0.201	0.094	0.096 (2.0%)	0.112 (19.0%)	0.099 (5.32%)
	Macro-F1	0.237	0.096	0.102 (6.2%)	0.122 (25.7%)	0.105 (9.37%)
	Micro-F1	0.473	0.325	0.38 (16.9%)	0.397 (22.6%)	0.352 (8.30%)
	MTC (%)	0	98.1	98.3	88.7	97.9

- **Flat**: standard multi-class SVM for single-label classification. Note that concept hierarchy is not leveraged. Flat exhibits the near-optimal performance that an HC method can achieve, considering that the classifiers of leaf nodes will be activated for testing.

- **SHC (baseline)**: the standard HC with Single-BS that employs the highest-response-first strategy.

- **ER-SHC**: our proposed error reduction HC for Single-BS.

- **SIB-ER-SHC**: a simplified ER-SHC by using only the candidate nodes to form the committee (i.e., only sibling relationship is considered).

- **SHC-Ref** [5]: local classifiers are learnt using informative negative instances. In specific, the instances, which are easily misclassified to the branch of a target concept, will be included in its negative training set, so that the error can be blocked propagating to lower levels.

The results on two datasets supporting Single-BS are summarized in Table 2. Note that *Acc* is computed among leaf nodes, and *F1* scores are averaged over all the nodes. We can see that SHC saves more than 90% of computational cost compared to Flat at the expense of classification accuracy. In contrast to Flat, which traverses all the classifiers in a brute-force manner to achieve the highest possible accuracy, SHC suffers from the problem of error propagation. This is more obvious for ILSRVC1K, where the concept hierarchy has more levels. On the other hand, our proposed method can recover the errors to certain degree, and thus the effectiveness of hierarchical classification can be improved. In specific, for Caltech256, ER-SHC improves the baseline by 14% *Acc*, 18.5% *Macro-F1* and 7.2% *Micro-F1*. By only taking the sibling relationship into account, the improvements of SIB-ER-SHC are marginal with respect to the three metrics. This result demonstrates the advantage of a larger committee, which postpones the decision making until more observations are available and it is more confident to do so. Additionally, although local classifiers are trained with some informative negatives, SHC-Ref utilizing highest-response-first local strategy only achieves slight performance improvements over baseline. In terms of efficiency, ER-SHC sacrifices more computational cost than those of other three HC methods, but the saved cost by 67.1% from the one-vs-all approach is still considered significant, an indication that ER-SHC achieves a better balance between computational cost and accuracy. Similarly, for ILSRVC1K, ER-SHC outperforms SHC and SIB-ER-SHC by 19% and 16.6% in accuracy respectively, while remaining satisfactory in efficiency by saving 88.9% computational cost. We can see that the advantage of ER-SHC on ILSRVC1K is more obvious than that on Caltech256. The observations are consistent to the analysis in Section 3.4. This is due to the fact that the hierarchy of ILSRVC1K is in a larger scale which includes 1,327 more concepts than Caltech256. Therefore, the SVM models for the nodes at higher levels of the ILSRVC1K hierarchy are with much more complex or vague decision boundaries than those in Caltech256. This is because more offspring concepts are attached to each higher level node, bringing the instances with more variances of visual appearances, and finally making the resulting classifier less reliable. In this case, the semantic relationship verification and error reduction embedded in MCR model of ER-SHC are more necessary, because the error propagation is more serious.

To grasp more insights of the methods, we plot the performance at each hierarchy level on the two datasets in Figure 4 and Figure 5 respectively. *Acc* is computed among the concepts at the same level as they are exclusive. For *Macro-F1* and *Micro-F1* at *i-th* level, we adopt the way used in [16] by averaging *F1* scores of concepts at top i levels. It shows the trend of performance when more rare concepts from lower level are included. We can see that the error propagation becomes more serious with the increase level of depth, resulting in drop of performance. ER-SHC has demonstrated consistent superiority over other methods, confirming its ability to address the issue of error propagation. Surprisingly, the performance of SIB-ER-SHC seems better on ILSRVC1K than on Caltech256. This again confirms our

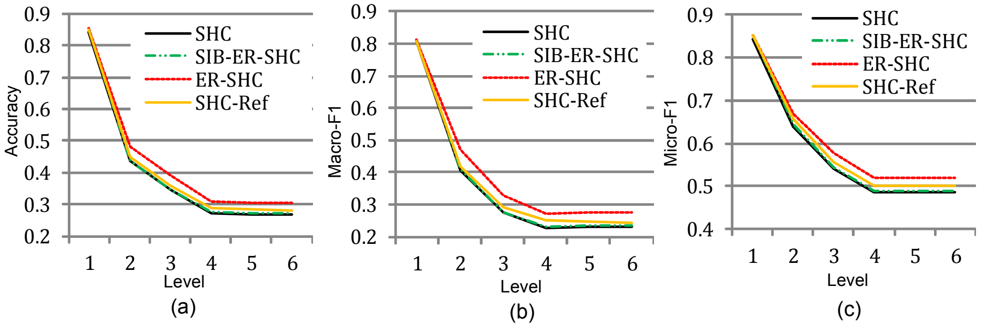


Figure 4: The performance of different HC methods with Single-BS on Caltech256 at each level measured by (a) *Accuracy*, (b) *Macro-F1* and (c) *Micro-F1*

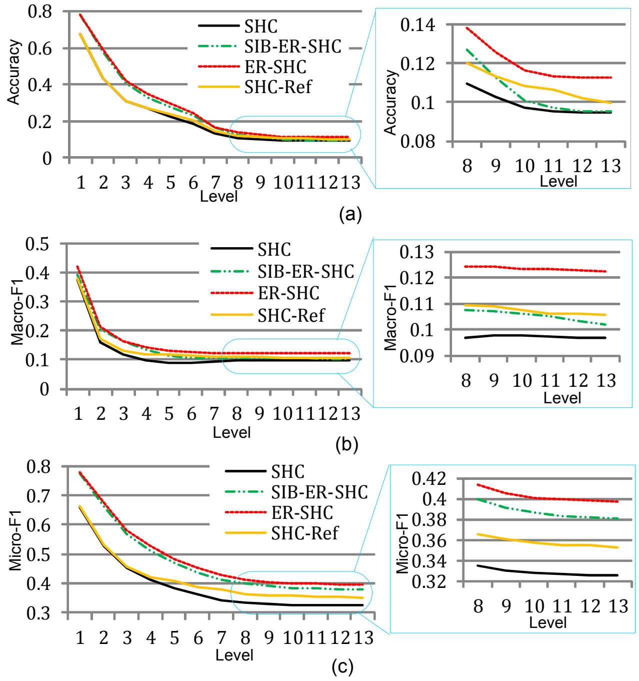


Figure 5: The performance of different HC methods with Single-BS on ILSRVC1K at each level measured by (a) *Accuracy*, (b) *Macro-F1* and (c) *Micro-F1*. The sub-figure on the right is a zoom-in from level 8 to level 13.

analysis that the advantage of employing a committee for decision making over the arbitrary highest-response-first strategy will be more obvious when the classifiers of individual nodes are weaker. It seems that involving some informative negatives is not that helpful as the decision boundary itself is

Table 3: Performance comparison of Flat and four HC methods with Multi-BS on Caltech256, ILSRVC1K and NUS-WIDE. Classification performance is measured by *Macro-F1* and *Micro-F1*. The testing efficiency is measured by average saved time cost (MTC). The performance gain over baseline is shown in the parentheses.

Dataset		Flat	MHC (baseline)	CHI-ER-MHC	ER-MHC	MHC-Ref
Caltech256	*Macro-F1*	0.248	0.171	0.187 (9.3%)	0.191 (11.6%)	0.185 (8.18%)
	Micro-F1	0.497	0.450	0.467 (3.8%)	0.478 (6.2%)	0.470 (4.44%)
	MTC (%)	0	92.9	72.3	65.7	90.6%
ILSRVC1K	*Macro-F1*	0.122	0.077	0.088 (14.2%)	0.095 (23.3%)	0.083 (7.79%)
	Micro-F1	0.277	0.191	0.229 (19.9%)	0.241 (26.1%)	0.221 (15.7%)
	MTC (%)	0	94.1	89.4	83.7	93.4%
NUS-WIDE	*Macro-F1*	0.297	0.204	0.212 (3.9%)	0.241 (18.6%)	0.220 (7.8%)
	Micro-F1	0.591	0.430	0.485 (12.7%)	0.532 (23.7%)	0.475 (10.4%)
	MTC (%)	0	86.4	70.2	63.8	85.1%

complex for a node with many offspring concepts. Thus SHC-Ref is less effective at the first few levels on ILSRVC1K. On the other hand, the improvement of SIB-ER-SHC over baseline is much more obvious on ILSRVC1K (Figure 5(c)) than on Caltech256 (Figure 4(c)) with respect to the *Micro-F1* at each level. This is because SIB-ER-SHC significantly improves the nodes with weaker classifiers at first few levels on ILSRVC1K. These nodes are all common classes, of which the performance is emphasized by *Micro-F1*. Eventually, SIB-ER-SHC improves over baseline by 16.9% *Micro-F1* at level 13 on ILSRVC1K.

6.2. Performance of HC with Multi-BS

We further evaluate our method for Multi-BS. The following methods are discussed in this section.

- **Flat**: standard multi-label SVM, where a testing instance may be assigned multiple labels by thresholding technique. Note that concept hierarchy is not leveraged.

- **MHC (baseline)**: the standard HC with Multi-BS (MHC), where a candidate node is selected by thresholding its original response.

- **ER-MHC**: our proposed error reduction HC for Multi-BS using a committee same with ER-SHC (i.e., sibling nodes and their child nodes).

- **CHI-ER-MHC**: a simplified ER-MHC by only including the candidate node and its child nodes in the committee.

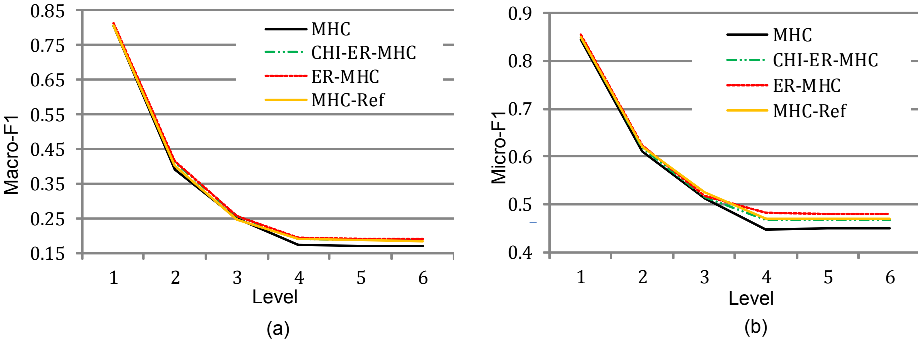


Figure 6: The performance of different HC methods with Multi-BS on Caltech256 at each level measured by (a) *Macro-F1* and (b) *Micro-F1*.

- **MHC-Ref** [5]: the refined hierarchical classification method with multi-branch selection.

- **L-ER-MHC**: the extension of our proposed method for Multi-BS by enlarging the committees using both semantic and contextual relations.

- **EXC-SIB-MHC**: our proposed exclusive-aware MHC, where the branch selection for sibling nodes with exclusive relation adopts highest-response-first strategy.

- **EXC-ER-MHC**: exclusive-aware ER-MHC, where the LR model is replaced with MCR model for adjusting responses of candidate nodes with exclusive relationship.

- **EXC-L-ER-MHC**: similar to EXC-ER-MHC, we implement exclusive-aware L-ER-MHC by replacing LR model with MCR model for adjusting responses of candidate nodes with exclusive relationship.

We first show the performance of ER-MHC and CHI-ER-MHC, where only the semantical consistency is verified in the response adjustment. Note that Multi-BS methods are applied on Caltech256 and ILSRVC1K by ignoring the prior knowledge of exclusive relationship among sibling nodes. The results on three datasets are summarized in Table 3. Similar to the observation in Table 2, compared to Flat, the effectiveness of MHC is sacrificed at the expense of efficiency. On the other hand, compared to baseline which only involves candidate node for branch selection, committee-based method CHI-ER-MHC performs better on all the three datasets with respect to both

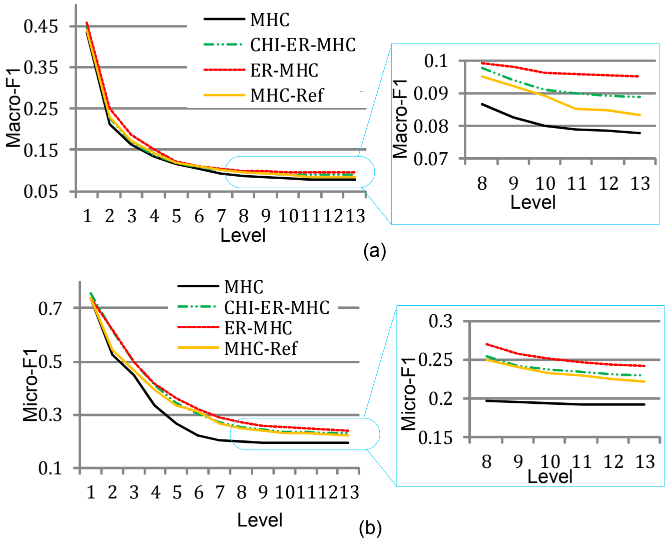


Figure 7: The performance of different HC methods with Multi-BS on ILSRVC1K at each level measured by (a) *Macro-F1* and (b) *Micro-F1*. The sub-figure on the right is the zoom-in from level 8 to level 13.

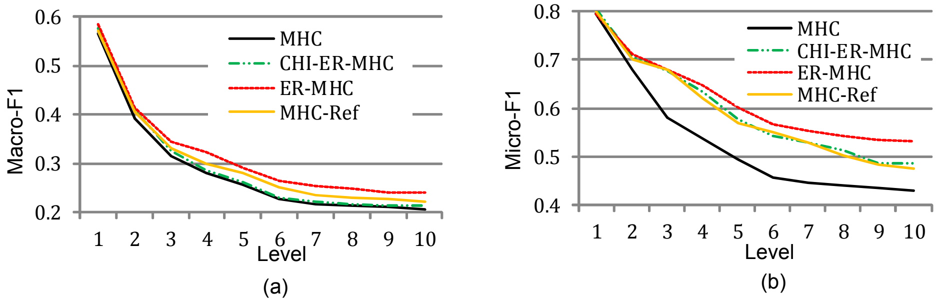


Figure 8: The performance of different HC methods with Multi-BS on NUS-WIDE at each level measured by (a) *Macro-F1* and (b) *Micro-F1*.

Macro-F1 and *Micro-F1*. With larger committee by including more relatives, ER-MHC achieves the best result. There are two benefits to involve siblings and their children in the committee. Firstly, the exclusive relationship between sibling nodes is implicitly exploited. It helps to correct the misclassified instances which actually belong to other sibling nodes, and meanwhile avoids propagating the errors to lower levels. This is particularly important for hierarchy with exclusive relationship (i.e., Caltech256 and ILSRVC1K). Secondly, compared to MHC and CHI-ER-MHC, more responses of classi-

Table 4: Performance comparison of two error reduction MHC methods on NUS-WIDE. ER-MHC only consider semantic relation, and L-ER-MHC adopts a larger committee by combining semantic and contextual relations. *F1* scores are averaged on 59 nodes with large committees. The performance gain of L-ER-MHC over ER-MHC is shown in the parentheses.

	ER-MHC	L-ER-MHC
Macro-F1	0.216	0.237 (9.7%)
Micro-F1	0.393	0.452 (15%)
MTC (%)	63.8	56.2

fiers in a larger committee provide more clues for identifying and blocking the false positives which are misclassified at higher level. Thus ER-MHC significantly improves over both MHC and CHI-ER-MHC on NUS-WIDE, where exclusive relationship may not exist. Similar to the observations in Section 6.1, the advantage of ER-MHC is more obvious for large hierarchy. It makes more improvements (23.3% *Macro-F1* and 26.1% *Micro-F1*), and saves more computational cost (83.7%) on ILSRVC1K. While MHC-Ref archives some improvements over baseline, compared with ER-MHC, there is still a performance gap. The advantage of our method is more obvious on larger hierarchy, where local branch selection strategy, which is adopted by MHC and MHC-Ref, may cause severe error propagation.

We further plot the *Macro-F1* and *Micro-F1* of each method on three datasets by level in Figure 6, Figure 7 and Figure 8 respectively. As shown in Figure 6, it seems that the improvement of ER-MHC is not obvious at higher level on Caltech256. The main reason is that Caltch256 is a relatively small dataset which only includes 7,400 test images. The common concepts at higher level can easily achieve a very high performance (e.g., 0.807 *Macro-F1* and 0.845 *Micro-F1* at first level). However, with the increase level of depth, error is amplified. With error reduction methods, improvements at lower levels are thus more noticeable than at higher levels. We can observe more improvements on ILSRVC1K (Figure 7) and NUS-WIDE (Figure 8), where the hierarchies have more levels than Caltech256. Rather than optimizing for a specific hierarchy, our method works well on different kinds of hierarchies. In general, the improvement is more obvious on larger datasets, which either include many instances (e.g., NUS-WIDE) or have many concepts organized in a larger hierarchy (e.g., ILSRVC1K). These are two common cases in many real world applications.

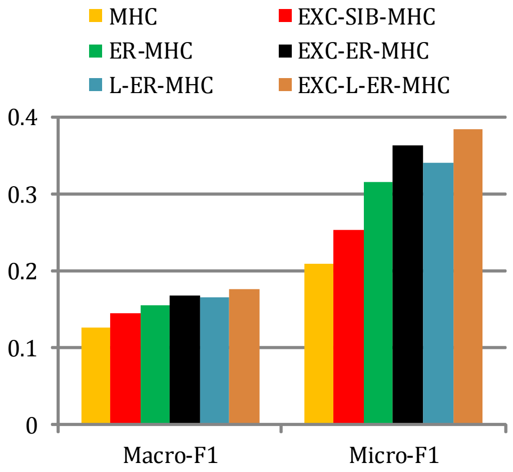


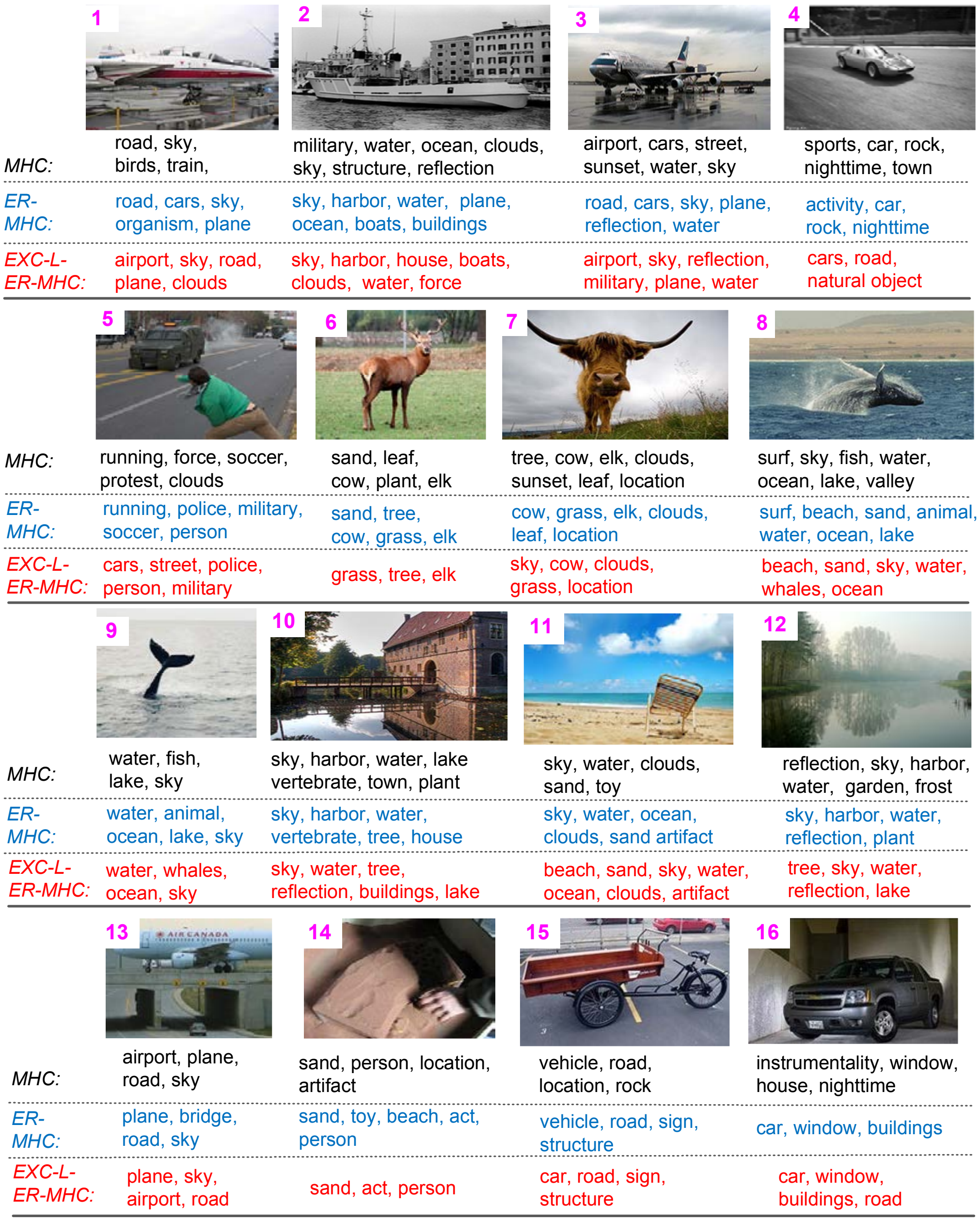
Figure 9: The results of multi-branch HC methods with and without using the prior knowledge of exclusive relationship. *F1* scores are averaged over 32 nodes which are exclusive to their siblings

We further verify the performance of L-ER-MHC combining semantic and contextual relations. The experiment is conducted on NUS-WIDE, where there are 59 nodes in the hierarchy having contextually related concepts, and thus their committees can be enlarged. For comparison, we average the *F1* scores of error reduction MHC methods on the 59 nodes. As shown in Table 4, L-ER-MHC shows improvement over ER-MHC by 9.7% *Macro-F1* and 15% *Micro-F1* respectively. This indicates that the consistency verification of responses among contextually related nodes results in a more reasonable branch selection strategy. In addition, while L-ER-MHC needs to activate more classifiers, the computational cost introduced by large committee is small (7.6% of MTC). This is because only the top most contextually related concepts (5.6 on average) are considered in the enlarged committee. Thus our method maintains the advantage of efficiency.

Comparing the results in Table 3 and Table 2, we can see that MHC performs worse than SHC on Caltech256 and ILSRVC1K with respect to *Macro-F1* and *Micro-F1*. The main reason is that the exclusive relationship is ignored in the standard MHC. While ER-MHC implicitly uses this relationship by incorporating siblings in the committee, the baseline of SHC still achieves a better result on both Caltech256 and ILSRVC1K. In other words, when the exclusive relationship is explicitly utilized by allowing only one candidate node to be selected, a better performance is achieved. To further confirm the merit of using this prior knowledge, we evaluate the proposed exclusion-aware Multi-BS methods (i.e., EXC-SIB-MHC, EXC-ER-MHC and EXC-L-ER-MHC) on NUS-WIDE. The exclusive relationship among sibling nodes is derived from the ground-truth of the training dataset. In specific, the sibling nodes are considered to be exclusive if they have no shared positive instances. Eventually, there are 32 nodes in NUS-WIDE hierarchy, which are exclusive to their sibling nodes. The *F1* scores averaging on the 32 nodes is shown in Figure 9. We can see that EXC-SIB-MHC improves over MHC by 14.3% *Macro-F1* and 21.5% *Micro-F1* respectively. In addition, EXC-ER-MHC outperforms ER-MHC by 7.9% *Macro-F1* and 14.9% *Micro-F1.* Similarly, EXC-L-ER-MHC improves over L-ER-MHC by 7.3% *Macro-F1* and 12.2% *Micro-F1.* The results confirm our suspicion that prior knowledge of mutual exclusion is indeed helpful. The less improvements of EXC-ER-MHC and EXC-L-ER-MHC compared to EXC-SIB-MHC are due to the fact that exclusive relationship has been implicitly exploited by ER-MHC and L-ER-MHC, where the sibling (exclusive) concepts are included in the committee. Furthermore, our proposed collaborative branch selection scheme (EXC-L-ER-MHC), which jointly considers multiple relationships, achieves the best result.

To verify the performance of different methods is not by chance, we further conduct significance test using randomization test [41] suggested by TRECVID [42]. The target number of iterations used in the randomization is 100,000. At 0.05 significance level, error reduction HC is significantly better than conventional method for both single-branch and multi-branch selection methods. In addition, exclusive-aware approaches are all significantly better than the corresponding methods without considering the prior knowledge of exclusion. Finally, EXC-L-ER-MHC using all the concept relationships is significantly better than all other Multi-BS methods.

To grasp the insight of how the multiple relationships affect the label assignments, we show some examples and the corresponding classification results using MHC, ER-MHC and EXC-L-ER-MHC respectively in Figure 10. Note that each method outputs multiple branches. Here we only show the most specific concept at the deepest level for each branch. We can see that ER-MHC generates more accurate labels than MHC, and the result is further improved by EXC-L-ER-MHC. For example, MHC misclassifies image 1 to “train”. This error is identified by ER-MHC, and recovered by leading to the branch “vehicle”. Then the instance is further classified to “plane”. However, we found that “cars” under “vehicle” is also assigned. This problem is ad-



	1	2	3	4
MHC:	road, sky, birds, train,	military, water, ocean, clouds, sky, structure, reflection	airport, cars, street, sunset, water, sky	sports, car, rock, nighttime, town
ER-MHC:	road, cars, sky, organism, plane	sky, harbor, water, plane, ocean, boats, buildings	road, cars, sky, plane, reflection, water	activity, car, rock, nighttime
EXC-L-ER-MHC:	airport, sky, road, plane, clouds	sky, harbor, house, boats, clouds, water, force	airport, sky, reflection, military, plane, water	cars, road, natural object

	5	6	7	8
MHC:	running, force, soccer, protest, clouds	sand, leaf, cow, plant, elk	tree, cow, elk, clouds, sunset, leaf, location	surf, sky, fish, water, ocean, lake, valley
ER-MHC:	running, police, military, soccer, person	sand, tree, cow, grass, elk	cow, grass, elk, clouds, leaf, location	surf, beach, sand, animal, water, ocean, lake
EXC-L-ER-MHC:	cars, street, police, person, military	grass, tree, elk	sky, cow, clouds, grass, location	beach, sand, sky, water, whales, ocean

	9	10	11	12
MHC:	water, fish, lake, sky	sky, harbor, water, lake vertebrate, town, plant	sky, water, clouds, sand, toy	reflection, sky, harbor, water, garden, frost
ER-MHC:	water, animal, ocean, lake, sky	sky, harbor, water, vertebrate, tree, house	sky, water, ocean, clouds, sand artifact	sky, harbor, water, reflection, plant
EXC-L-ER-MHC:	water, whales, ocean, sky	sky, water, tree, reflection, buildings, lake	beach, sand, sky, water, ocean, clouds, artifact	tree, sky, water, reflection, lake

	13	14	15	16
MHC:	airport, plane, road, sky	sand, person, location, artifact	vehicle, road, location, rock	instrumentality, window, house, nighttime
ER-MHC:	plane, bridge, road, sky	sand, toy, beach, act, person	vehicle, road, sign, structure	car, window, buildings
EXC-L-ER-MHC:	plane, sky, airport, road	sand, act, person	car, road, sign, structure	car, window, buildings, road

Figure 10: Example images in NUS-WIDE, and the classification results using different methods. Each method generates multiple branches, on which we only show the most specific concepts at the deepest level.

dressed by EXC-L-ER-MHC using the exclusive relationship between "cars" and "plane". The exclusive relationship is particularly useful for discriminating the sibling concepts which are visually similar, such as "cow" and "elk" in image 6 and image 7, as well as "fish" and "whales" in image 8 and image 9. In addition, our proposed method is able to detect the false positives from higher level. For example, image 4 is wrongly labeled with "rock" by both MHC and ER-MHC, which is caused by the wrong decision made at the parent node "nature object". In EXC-L-ER-MHC, this error is identified by exploiting the responses of concepts which are contextually related to "rock", such as "mountain", "sky". Finally, the error is blocked at node "natural object" and will not be propagated to lower level. On the other hand, we observe two representative failure cases. First, the exclusive relationship among concepts, which are derived from the ground-truth of training set, may be violated in the testing dataset. Explicitly utilizing this relation for branch selection may cause incomplete results. For example, "bridge" in image 13 and "toy" in image 14 are missed by EXC-L-ER-MHC. Second, incorrect decision may be made by incorporating contextual relationship. For example, image 15 is misclassified as "car" by EXC-L-ER-MHC, as the high response from concept "road" misleads the decision making. Similarly, "road" is also detected in image 16 by EXC-L-ER-MHC. In brief, although few exceptions may be introduced, collaborative consideration of multiple relationships in general results in a more accurate and complete label set.

7. Conclusion

We have presented a novel and effective approach, which is named as collaborative error reduction hierarchical classification, to utilize the semantic and contextual relationships encapsuled in the concept hierarchy for addressing the error propagation problem of conventional hierarchical classification. Furthermore, the approach is extended for a more general case: multiple branch selection. In particular, the semantic and contextual relations between concepts are embedded in an enriched committee, based on which the branch can be selected in a globally valid, semantically and contextually consistent view. In addition, an exclusion-aware method is proposed to explicitly integrate exclusive relationship in the branch selection, which is ignored in conventional multi-branch HC. Extensive experiments on three datasets show that the proposed methods significantly and consistently outperform conventional methods for both single-branch and multi-branch HC, while maintaining a satisfactory balance between effectiveness and efficiency.

Acknowledgement

The work described in this paper was sponsored by two grants from the National Natural Science Foundation of China (#61272290 and #61272256), and was supported by the Shenzhen Research Institute, City University of Hong Kong.

References

[1] X. Zhu, Z. Huang, J. Cui, H. T. Shen, Video-to-shot tag propagation by graph sparse group lasso, IEEE Transactions on Multimedia 15 (3) (2013) 633–646.

[2] Z. Ma, F. Nie, Y. Yang, J. R. R. Uijlings, N. Sebe, Web image annotation via subspace-sparsity collaborated feature selection, IEEE Transactions on Multimedia 14 (4) (2012) 1021–1030.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, F.-F. Li, Imagenet: A large-scale hierarchical image database, in: CVPR, 2009.

[4] G.-R. Xue, D. Xing, Q. Yang, Y. Yu, Deep classification in large-scale text hierarchies, in: SIGIR, 2008.

[5] P. N. Bennett, N. Nguyen, Refined experts: improving classification in large taxonomies, in: SIGIR, 2009.

[6] Z. Barutcuoglu, R. E. Schapire, O. G. Troyanskaya, Hierarchical multi-label prediction of gene function, Bioinformatics 22 (7) (2006) 830–836.

[7] Y. J. Lee, K. Grauman, Object-graphs for context-aware visual category discovery, IEEE Trans. on PAMI 34 (2) (2012) 346–358.

[8] S. J. Hwang, F. Sha, K. Grauman, Sharing features between objects and their attributes, in: CVPR, 2011.

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

[10] S. Zhu, C.-W. Ngo, Y.-G. Jiang, Sampling and ontologically pooling web images for visual concept learning, IEEE Trans. on MM 14 (4) (2012) 1068–1078.

[11] A. Binder, K.-R. Müller, M. Kawanabe, On taxonomies for multi-class image categorization, International Journal of Computer Vision 99 (3) (2012) 281–301.

[12] W. Bi, J. T. Kwok, Mandatory leaf node prediction in hierarchical multilabel classification, in: NIPS, 2012.

[13] S. Gopal, Y. M. Yang, B. Bai, A. Niculescu-Mizil, Bayesian models for large-scale hierarchical classification, in: NIPS, 2012.

[14] L. Xiao, D. Zhou, M. Wu, Hierarchical classification via orthogonal transfer, in: ICML, 2011.

[15] L. Xie, R. Yan, J. Tesic, A. Natsev, J. R. Smith, Probabilistic visual concept trees, in: ACM MM, 2010.

[16] T.-Y. Liu, Y. M. Yang, H. Wan, H.-J. Zeng, Z. Chen, W.-Y. Ma, Support vector machines classification with a very large-scale taxonomy, SIGKDD Explorations 7 (1) (2005) 36–43.

[17] B. Zhao, F.-F. Li, E. P. Xing, Large-scale category structure aware image categorization, in: NIPS, 2011.

[18] M. Marszalek, C. Schmid, Semantic hierarchies for visual object recognition, in: CVPR, 2007.

[19] J. Deng, J. Krause, A. C. Berg, F.-F. Li, Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition, in: CVPR, 2012.

[20] A. Barret, Nomenclature committee of the international union of biochemistry and molecular biology (nc-iubmb). enzyme nomenclature. recommendations 1992. supplement 4: corrections and additions, European Journal of Biochemistry 250 (1) (1997) 1–6.

[21] M. Ashburner, et al., Gene ontology: tool for the unification of biology, Nature Genetics 25 (2000) 25–29.

[22] G. Griffin, A. Holub, P. Perona, Caltech-256 object category dataset, Caltech Technical Report.

[23] S. Liu, H. Yi, L.-T. Chia, D. Rajan, Adaptive hierarchical multi-class svm classifier for texture-based image classification, in: ICME, 2005.

[24] Y. Chen, M. Crawford, J. Ghosh, Integrating support vector machines in a hierarchical output space decomposition framework, in: IEEE International Geoscience and Remote Sensing Symposium Melbourne, 2004.

[25] S. Bengio, J. Weston, D. Grangier, Label embedding trees for large multi-class tasks, in: NIPS, 2010.

[26] J. Deng, S. Satheesh, A. C. Berg, F.-F. Li, Fast and balanced: Efficient label tree learning for large scale object recognition, in: NIPS, 2011.

[27] M. Marszalek, C. Schmid, Constructing category hierarchies for visual recognition, in: ECCV, 2008, pp. 479–491.

[28] T. Gao, D. Koller, Discriminative learning of relaxed hierarchy for large-scale visual recognition, in: ICCV, 2011.

[29] Z. Liu, W. Shi, Q. Qin, X. Li, D. Xie, Hierarchical support vector machines, in: IEEE International Geoscience and Remote Sensing Symposium Melbourne, 2005.

[30] S. Xia, J. Li, L. Xia, C. Ju, Tree-structured support vector machines for multi-class classification, in: 4th International Symposium on Neural Networks, 2007.

[31] S. Zhu, X.-Y. Wei, C.-W. Ngo, Error recovered hierarchical classification, in: ACM MM, 2013.

[32] C.Silla, A.Freitas, A survey of hierarchical classification across different application domains, Data Mining and Knowledge Discovery 22 (1-2) (2011) 31–72.

[33] Y.-G. Jiang, Q. Dai, J. Wang, C.-W. Ngo, X. Xue, S.-F. Chang, Fast semantic diffusion for large scale context-based image and video annotation, IEEE Trans. on Image Processing 21 (6) (2012) 3080–3091.

[34] X.-Y. Wei, Y.-G. Jiang, C.-W. Ngo, Concept-driven multi-modality fusion for video search, IEEE Trans. on CSVT 21 (1) (2011) 62–73.

[35] Y.-G. Jiang, C.-W. Ngo, S.-F. Chang, Semantic context transfer across heterogeneous sources for domain adaptive video search, in: ACM MM, 2009.

[36] R. L. Cilibrasi, P. M. B. Vitányi, The google similarity distance, IEEE Trans. on KDE 19 (3) (2007) 370–383.

[37] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, Y. Zheng, NUS-WIDE: A real-world web image database from national university of singapore, in: ACM International Conference on Image and Video Retrieval, 2009.

[38] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: CVPR, 2010.

[39] Y. M. Yang, A study on thresholding strategies for text categorization, in: SIGIR, 2001.

[40] V. V. Asch, Macro- and micro-averaged evaluation measures, Technical Report.

[41] J. P. Romano, On the behavior of randomization tests without a group invarianceassumption, Journal of the American Statistical Association 85 (411) (1990) 686–692.

[42] A. F. Smeaton, P. Over, W. Kraaij, Evaluation campaigns and trecvid, in: MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, 2006.