A Salient Ensemble of Trees using Cascaded Linear Classifiers with Feature-Cost Constraints

Chien-Wen Huang* Chung-Kuang Chou* Ming-Syan Chen*

Abstract

In many applications the classification model needs to utilize limited resources properly while predicting an instance, e.g. the limited response time for a real-time search engine. In order to satisfy the resource constraint, many researchers try to simplify the model structure or shrink the feature subset size. Because the informative features may take too much cost for the model, a common way is to build a model by considering the trade-off between performance and cost. However, most previous works assume that the cost of a feature is independent of the cost of another feature, which is not practical in reality. In the paper, we consider two categories of the feature cost, individual cost and group cost. The former is independent of the cost of any other feature whereas the latter regards the cost dependency between the other features in the corresponding group. We propose a two-stage framework that integrates the cost-sensitive feature selection and learning a model with a cost budget constraint. First, we propose the group-cost-sensitive random forest (`GOAT`) model to consider these two costs to select a proper feature subset. Second, we propose a salient ensemble of trees each of which uses cascaded linear classifiers (`ETIC`) with the satisfaction of the feature-cost constraints using the derived features from the `GOAT` model. We conduct experiments on real-world datasets, including mobile-user preference data and object detection data. When the group cost dominates, `GOAT-ETIC` can gain a 10-30% improvement over the baselines. Even if the group cost is ignored, `GOAT-ETIC` can still get better performance than the state-of-the-arts.

1 Introduction

Feature selection has been an active research topic in data mining. To build a model on a well-selected subset of features can save computational resources and reduce overfitting. Various approaches are developed to select the subset, such as methods based on rough set theory [14, 23], decision trees [19], and Bayesian network [20]. However, the traditional feature selection techniques often neglect the effect of the feature cost[1]. To just select all the informative features with high cost becomes impractical [4]. In several applications, e.g. a real-time search engine or a self-driving car [12], there can exist a feature cost budget. For instance, applications that classify images on a mobile device may have a limited response time for every testing instance. The classification systems usually extract features from raw data as an input of a model. The objective of the system becomes more complicated because it needs to select the informative features and learn a model while holding the feature cost constraints, such as the limited prediction time of a mobile application.

However, most of the previous works are based on the simplified assumption that the feature costs are independent [5, 18, 24]. In the paper, the features costs are further divided into two categories: the individual cost and group cost[2]. The individual cost is independent to other features, which can be the time or memory usage to process a feature. The group cost is the cost it takes if the model needs to extract a group of features. More precisely, before the model selects a group of features, it needs to spend the group cost first, then it can spend the individual cost to select features. For example, when a group of multi-dimension features such as color histogram or MPEG-7 descriptors [1] is extracted, the system needs to spend the process time (group cost) first then the memory (individual cost) to store each feature.

After the proper feature subset is selected, the next problem is how to use it for learning a good classification model. Though many previous works [7, 11, 12, 21] propose classification models with cost budget, there is still a paucity of literature on how to integrate the feature selection step with the model. The feature selection step and the model-training step are often separated as two independent black-boxes, and it is hard

*Dept. of Electrical Engineering, National Taiwan University, {{cwhuang1021,ckchou}@arbor.ee., mschen@}ntu.edu.tw

[1] Although some dimension reduction techniques such as PCA can reduce the number of output features for further analysis, the output features are combinations of input features. Thus, in the prediction phase, all input features may still need to be extracted.

[2] The type of cost can be also divided into test costs [10], misclassification costs [13] and feature costs [3], which will be further discussed in Section 3.1.

to determine which feature selection method can select the best feature subset for the model. In the paper, we aim to connect these two black-boxes by using the intermediate output of cost-sensitive feature selection to learn an ensemble model with the cost constraint. The ensemble model [6, 11, 12] is popular in machine learning field because it usually has good performance. To build a unified framework on the ensemble model, the major challenge is how to produce various informative feature subsets for each base classifier and hold the cost constraint simultaneously. If the features are selected randomly for each base classifier, the model may run out of cost easily and the size of selected feature subset will be too small to perform well. In the paper, we propose a salient ensemble of trees each of which uses cascaded linear classifiers (`ETIC`) and the group-cost-sensitive random forest (`GOAT`) model for enhancement. To the best of our knowledge, this is the first work that integrates cost-sensitive feature selection and building an ensemble model with a feature-cost budget constraint, including dependent cost, i.e. group cost. The major contributions of our paper are summarized as follows:

1. We formally define the feature costs as two categories, i.e. the individual cost and group cost, to extend the independent-feature-cost assumption.
2. We propose a two-stage framework with the proposed individual cost and group cost that integrates the feature selection and model training with a cost budget constraint.
3. We conduct experiments with different feature cost settings on various real-world datasets, including mobile-user preference data [2] and object detection data [15]. The results show that our `GOAT-ETIC` framework can outperform the baseline methods in most of the settings and datasets.

The remainder of this paper is organized as follows. We next review related work in Section 2 and introduce the preliminary knowledge in Section 3. The proposed framework and models are described in Section 4. The experiment results are shown and discussed in Section 5. Finally, we conclude in Section 6.

2 Related Works

In the section, we briefly review related work on feature selection that considers not only feature information but also other properties, e.g. intrinsic structure or feature cost. We further review recent researches on learning a model with cost budget constraints.

2.1 Cost-Sensitive Feature Selection There are many kinds of desirable properties to keep during the procedure of feature selection such as the test cost, misclassification cost [10] or the intrinsic structure of data in the low dimension space [8, 9]. To tackle the cost-sensitive feature selection problem, one main strategy is to build rules about whether features should be used, e.g. by building the minimal cost tree [16]. On the other hand, Exponent weighted algorithm [5] constructs an exponent weighted function of feature importance for the problem of minimal cost feature selection. Some methods use the function to compute the feature importance, e.g. Feature Cost-Sensitive Random Forest (FCSRF) [24] is a random forest-based model which incorporates both the distinguishing ability of features and their costs into a single optimization process. However, the above methods assume that a feature cost is independent of another one, which is impractical for several kinds of applications.

2.2 Learning with Cost Budget Constraints The goal of cost-sensitive learning is to learn a model that ensures the total feature cost of any test instance will satisfy the cost budget constraints. Cost-sensitive tree of classifiers [21] is designed for this purpose that each testing instance traverses different path from root to leaf, and the node only extracts some of the features accordingly. To integrate the feature-cost concept into the model such as random forest or neural network, Nan et al. [11] propose Feature-Budgeted Random Forest to build the cost-sensitive model using their proposed impurity function to hold the constraints. Instead of using the top-down idea, the BudgetPrune [12] prunes the random forest to consider a cost constraint. These methods use only single feature to build each node in each tree whereas our `ETIC` can use multiple features in each node, which implies `ETIC` is more suitable for data containing dependent features, e.g. images, movies. Moreover, `ETIC` considers the additional group cost.

3 Preliminaries

In the section, we recap some background knowledge and define the used notation. The main notation is summarized as Table 1.

3.1 The Decision System with Costs Based on the rough set theory [5, 14], the cost-sensitive decision system S can be defined as follows:

$$S = (U, F, Y, G, V_a|a{\in}F{\cup}Y, I_a|a{\in}F{\cup}Y, mc, ic, gc),$$

where U is a nonempty finite set of objects, called the universe, F is a nonempty finite set of features, Y is a nonempty finite set of class labels, $G{\subseteq}\wp(F)$ is the subset of the power set $\wp(\cdot)$ of F, V_a is a set of values for each feature $a{\in}F{\cup}Y$, $I_a : U{\rightarrow}V_a$ is an information function for each feature $a{\in}F{\cup}Y$, $mc : |Y|{\times}|Y|{\rightarrow}R^+{\cup}\{0\}$ is

Table 1: Notation table

Symbol	Meaning
U	universe
$\boldsymbol{X}$	input dataset, $\boldsymbol{X} \subset U$
X_t	out-of-bag data of the t-th tree
Y	label set
F	feature set
G	subset of the power set of F
B	feature subset
$mc(B)$	misclassification cost of a given feature subset B
$ic(B)$	individual cost of a given feature subset B
$gc(B)$	group cost of a given feature subset B
$TC(B)$	total cost of a given feature subset B
$\mathbb{H}$	hypothesis function of an ensemble model
H_t	hypothesis function of the t-th base classifier
T	the number of base classifiers in the ensemble model
S_j	feature importance score of the j-th attribute
M	size of feature set
N	size of input dataset
$C_{t,k}$	cost budget of the k-th node of the t-th tree
K	the number of features in each node of the proposed ensemble model

the misclassification cost matrix, $ic : F \to R^+ \cup \{0\}$ is the individual cost function and gc is the group cost function.

The misclassification cost in [13] is represented as a $|Y| \times |Y|$ matrix with elements $\{mc_{i,j} | i, j \in \{1, 2, ..., |Y|\}\}$, where $|Y|$ is the number of class labels, and $mc_{i,j}$ means the cost of misclassification that predicts an instance to the i-th class but actually it belongs to the j-th class. If the classification result is correct, $mc_{i,i} = 0$. For simplicity, we simplify the misclassification cost to the classification error, which means the values of all $mc_{i,j} \forall i \neq j$ are equal. Note that the exact value of mc depends on the classification error function of the classifier.

The individual cost stands for the part of cost that is independent of other features, so we adopt the test cost independent model [18] to define it as a vector $ic = [ic(a_1), ic(a_2), \ldots, ic(a_{|F|})]$. When our model selects a feature subset $B \subseteq F$, the individual cost for this subset can be denoted as

$$ic(B) = \sum_{a \in B} ic(a). \tag{3.1}$$

Assume that given a group of features D, the group cost $gc(D)$ is accounted if and only if any feature which belongs to that group is selected. In other words, even if two or above features that are in the same group with D are selected, $gc(D)$ will still only be counted once. In other words, we can define the group cost function as a vector $[gc(D_1), gc(D_2), \ldots, gc(D_{|G|})]$, where $D_i \subseteq F$ for $i = 1, 2, \ldots, |G|$. When our model selects a feature subset $B \subseteq F$, the total group cost can be defined as

$$gc(B) = \sum_{D_i \in G, D_i \cap B \neq \phi, i=1}^{|G|} gc(D_i). \tag{3.2}$$

One of our goal is to minimize the total costs (TC) after the feature selection task is performed, and with selected feature subset B, the total costs $TC(B)$ based on the previous setting can be denoted as

$$TC(B) = mc(B) + \alpha(ic(B)) + \beta(gc(B)). \tag{3.3}$$

In the above formula, α and β denote the trade-off between the three costs. When β is set to zero, the problem is reduced to the cost-sensitive feature selection with the assumption that the feature cost is independent to the other features. Though the value of TC needs to be computed for every raw instance in both the training phase and the testing phase, our framework considers the cost budget only on testing with the assumption that there are sufficient resources in the training procedure. The problem is formulated to the following equation.

$$\begin{aligned} err(y, \mathbb{H}(x)) &= \frac{\sum_{i=1}^{N} e_{test}(y, \mathbb{H}(x))}{N} + TC(B), \\ \text{where } e_{test}(y, \mathbb{H}(x)) &= \begin{cases} 0 & if \quad \mathbb{H}(x) = y \\ 1 & if \quad \mathbb{H}(x) \neq y. \end{cases} \end{aligned} \tag{3.4}$$

Note that $err(y, \mathbb{H}(x))$ is the loss function which sums the error rate between the real label y and the predicted label $\mathbb{H}(x)$, and the total cost of the used feature subset B. In the classification problem, the error rate can be computed via the average of the $e_{test}(y, \mathbb{H}(x))$ function that outputs 1 if the predicted $\mathbb{H}(x) \neq y$ and otherwise outputs 0. To solve Eq. (3.4), a proper hypothesis function $\mathbb{H}$ needs to be found among the whole hypothesis set. In the paper, we will propose a salient ensemble model and select the best model in this hypothesis subset in the training procedure.

3.2 Random Forest for Feature Selection Random Forest (RF) [6] is a widely used model for machine learning. In addition to its good performance and high robustness in classification problem, RF can also be used to compute the feature importance with the help of the out-of-bag (OOB) training data. The core idea of RF is to build randomized decision trees with different data distribution and features to build a robust ensemble model. First, RF samples training data with replacement, i.e. bagging, for each base decision tree. Second, RF learns the nodes in each decision tree by computing an impurity function value, e.g. information gain, using the randomly-selected feature. Then

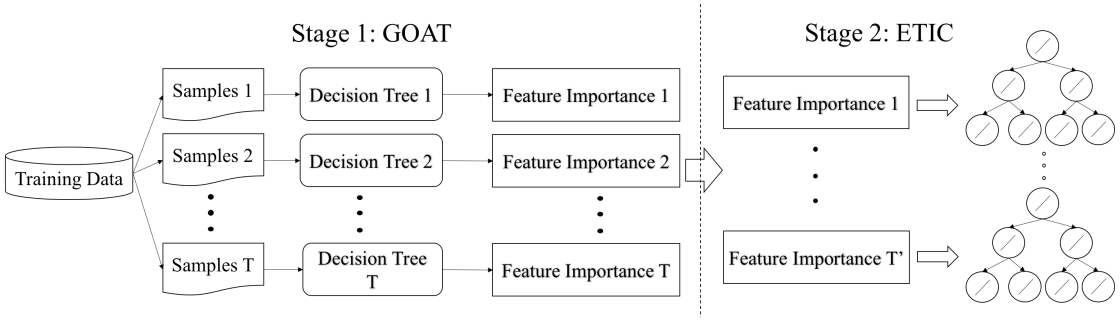


Figure 1: Flowchart

the in-bag data can traverse to left or right child node using the best-split. When the data in every leaf node is pure or satisfies the termination condition such as depth limit, the tree is well-trained.

Here we denote the OOB data for tree t as X_t and the j-th attribute of the i-th OOB instance as $x_{t,i,j}$. To learn a RF model for feature selection, the traditional way is to permute the OOB data feature value randomly. Let $\hat{X}_t$ denote the randomly-permuted OOB data for tree t. RF can compute the feature importance of the j-th attribute, i.e. S_j, as Eq. (3.5).

$$S_j = \frac{\sum_{t=1}^{T} err(y, \hat{X}_t) - err(y, X_t)}{T}. \tag{3.5}$$

4 Framework

In this section, we propose a general framework to perform the group-cost-sensitive random forest (GOAT) and learning an ensemble of trees each of which uses cascaded linear classifiers (ETIC) with cost constraints as shown in Fig. 1. In the first stage GOAT will rank the features with the consideration of both the importance and the cost. In contrast to previous cost-sensitive feature selection methods, GOAT considers additional group cost. Second, the framework will use this ranking list to learn an ensemble of trees using cascaded linear classifiers with a feature-cost constraint, i.e. the ETIC model. In the following, we use $\{x_n, y_n\}_{n=1}^{N}$ to represent the dataset and $\boldsymbol{X} = \{x_n\}_{n=1}^{N}$ to denote the data matrix. Each instance $x_n \in \mathbb{R}^M$ is characterized by features with dimension M. $y_n \in \mathbb{R}^c$ belongs to the class label $\boldsymbol{C} = \{c_l\}_{l=1}^{c}$, where $c_l = \{0, 1\}$.

4.1 Group-Cost-Sensitive Random Forest We propose a feature selection model that considers both accuracy and feature cost based on [24]. Instead of assuming the costs of each feature are independent in previous methods, our idea is to integrate the group feature cost to the FCSRF model and propose the group-cost-sensitive random forest (GOAT) model to compute the feature importance. When the traditional random forest for feature selection is trained, it selects the candidate features randomly in each node and sets the threshold with the most proper value of the impurity function such as information gain or GINI index. However, in order to take the feature cost into consideration in our model, GOAT aims to select the candidate features based on the inverse of the total feature cost. That is, the probability that the model selects feature f_i as a candidate is initialized using the following formula.

$$p(f_i) = \frac{\frac{1}{TC(f_i)}}{\sum_{j=1}^{M}(\frac{1}{TC(f_j)})} \tag{4.6}$$

The total cost of a feature f_i in the above formula is set by Eq. (3.3). After GOAT gets all the candidate features, it selects one of them with the maximum information gain of the training data in the current node and sets the threshold value. GOAT then resets the group cost of all the features that are in the same group with the selected feature to zero and updates the probability $p(f_i)$ of each feature f_i by recomputing Eq. (4.6). Because the probability is set to the inverse of the total cost, the tree will be more likely to select candidate features with less cost unlike the traditional random forest. Moreover, since the group cost is reset after each node finished training, each tree prefers the features that are in the same group. Note that the FCSRF model only considers the individual cost and uses the fixed feature probability to select candidate feature for each node. In contrast, GOAT updates the feature probability for each tree by the reset of group cost and recomputing the feature probability using 4.6.

We present the pseudo code of the proposed GOAT model in Algorithm 1 with the corresponding complexity analysis in Appendix A [3].

[3] The Appendix file is available on the https://goo.gl/EzR33L.

4.2 Ensemble of Trees using Cascaded Linear Classifiers We propose a salient model that can use the feature information obtained from GOAT to learn multiple trees as an ensemble classifier with the budget constraint. The objective function of this model is defined as follows.

$$\begin{aligned} &\min_{\mathbb{H}} E[err(y, \mathbb{H}(x))] \\ &subject\ to \sum_{x, H_t} \max TC(x, H_t) \leq C \end{aligned} \tag{4.7}$$

The goal of Eq. (4.7) is to minimize the error rate of model as shown in Eq. (3.4), while holding the cost constraint. $\mathbb{H}(x)$ is the hypothesis function of our ensemble model, which is the majority voting on all base classifiers. $\max TC(x, H_t)$ is the maximum cost for an instance x to traverse from root to leaf of tree with hypothesis function H_t and C is the feature-cost constraint in prediction time.

For each tree, our ETIC model uses bagging to sample training data and trains the nodes from root to leaf to increase the data-diversity of each base classifier. Instead of randomly selecting features and computing the impurity function value like the traditional random forest, ETIC model treats each node as a linear classifier with features selected from the top of feature importance ranking list from GOAT. By learning the nodes using multiple features, ETIC can learn a more complicated decision boundary than previous methods. Moreover, the output feature importance ranking list from GOAT helps ETIC build better top-level nodes. It's better that the training data is separated well in early stage so the child nodes can focus on the hard-to-discriminate data. The loss function of node k in the t-th tree is defined as

$$\begin{aligned} &\min_{B_{t,k}} \frac{1}{N} \sum_{i=1}^{N} p_{i,k}(y_i - x_i^T B_{t,k})^2 + \rho ||B_{t,k}||^2 \\ &subject\ to \sum_{\varepsilon : |B_{t,k}| > 0} TC(\varepsilon) \leq C_{t,k}, \end{aligned} \tag{4.8}$$

where $p_{i,k}$ is set to 1 if instance x_i has traversed to node k and set to 0 otherwise, $B_{t,k}$ is the objective vector that ETIC would like to learn and implies how many features it used in this node, ρ is the control parameter for regularization, ε is the feature subset that our ETIC model used in this node, and $C_{t,k}$ is the remaining available cost budget constraint in the k-th node of the t-th tree and it is updated by subtracting the used cost from the original cost C. Note that the constraint indicates that the costs used in this node should not exceed the remaining available cost budget $C_{t,k}$.

To solve Eq. (4.8), ETIC uses the output feature importance ranking list from GOAT to select the feature subset directly. For each node, ETIC selects K features one by one from the top of feature importance ranking list and always satisfies the cost constraint in this procedure. Note that once a feature is selected, the group costs of all the features that are in the same group with that selected feature will be set to zero. When our ETIC model runs out of the cost budget, it randomly selects the features that have been used in previous nodes because the used-features need no more cost and the model can use it with other selected features in the current node to form a different decision boundary. Using this approach, Eq. (4.8) can be reformulated to a simple convex optimization problem as

$$\min_{B'_{t,k}} \frac{1}{N} \sum_{i=1}^{N} (y_i - x_i^T B'_{t,k})^2 + \rho ||B'_{t,k}||^2, \tag{4.9}$$

where $B'_{t,k}$ is a K-dimension dense vector representing the weights of the selected features. The closed form solution of Eq. (4.9) can be derived by setting the gradient to zero, and $B'_{t,k}$ can be solved as follows.

$$B'_{t,k} = (X^T X + \rho I)^{-1} X^T Y \tag{4.10}$$

After the weight $B'_{t,k}$ is learned using Eq. (4.10), because Eq. (4.9) is a least square loss function, the threshold $\theta_{t,k}$ of this node can be set to the mean of training samples that are in the k-th node of the t-th tree. The tree will identify the node as a leaf node when the number of samples in this node is small enough or the tree is too deep. Once the leaf node is reached, our ETIC model counts the class distribution of the training data that arrived the current node and labels this node with the majority class.

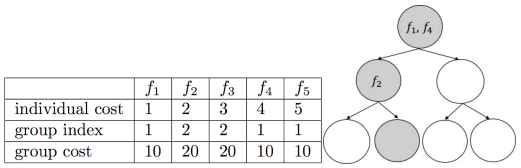
In the prediction phase, each node only extracts certain features of the testing instances then sends them to next node according the hypothesis function as defined in the following equation.

$$\begin{aligned} &If\ X^T B_{t,k} < \theta_{t,k},\ send\ X\ to\ left\ child. \\ &else,\ send\ X\ to\ right\ child. \end{aligned} \tag{4.11}$$

When the testing instance reaches the leaf node, the tree predicts the instance with the majority label that is determined in the training phase. Finally, ETIC aggregates the results of all trees to decide the final label of the testing instance by the following formula.

$$\hat{y} = \mathbb{H}(x) = Majority(\{H_1(x), H_2(x), ..., H_T(x)\}) \tag{4.12}$$

The function $Majority(\{H_1(x), H_2(x), ..., H_T(x)\})$ stands for the majority votes for the hypothesis function of all the base classifiers. In addition, the total feature cost of this instance depends on the used



	f_1	f_2	f_3	f_4	f_5
individual cost	1	2	3	4	5
group index	1	2	2	1	1
group cost	10	20	20	10	10

(a) Cost example. (b) Tree example.

Figure 2: Example of the total feature cost of an instance

Table 2: Dataset

Dataset	Dataset size	Positive label-ratio
User0	8120	4.828%
User1	8156	3.139%
DET124	72783	39.79%
DET177	64748	10.27%

features of traversed nodes. For example, consider the cost setting in Fig. 2a and the trained tree in Fig. 2b, where the nodes that this instance traversed have been colored, the total cost of a specific testing instance can be computed. Derived from Fig. 2b, the total feature cost is $TC(\{f_1, f_2, f_4\}) = 37$[4], which is smaller than the total cost of the whole available features $TC(\{f_1, f_2, f_3, f_4, f_5\}) = 45$[5]. Moreover, the different instance will traverse through a different path. Our framework guarantees that the sum of the feature cost from root to leaf of all trees is in the budget constraint. The pseudo code of ETIC is shown in Algorithm 2 with the corresponding complexity analysis in Appendix B.

5 Experiments

In the section, we first introduce four datasets, summarized as Table 2, and describe the experiment setting.

5.1 Datasets The first two datasets are from [2], which contain daily-life photos with the corresponding sensor logs of mobile phones from volunteers by a wearable camera in a two-week collection. The images are labeled to either "interesting" or "not interesting" by each user. We perform the experiments on User0 and User1 datasets[6]. The two datasets for the image-classification task are highly imbalance, as shown in Table 2. Moreover, two datasets, DET124 and DET177, from ImageNet benchmark [15] for object detection are used. DET124 is a dataset which aims to detect whether there's any human in an image or not. In DET177, an image is labeled positive if any desk is detected in the picture. There are 28959 positive images, 42945 negative images and 879 neutral images in DET124 and 6648 positive images, 58088 negative images and 12 neutral images in DET177. However, the neutral images contain only a small fraction of the target object in the scene, we set them to negative in the experiment.

Table 3: Feature Groups

Source	Abbrev.	$\|G\|$	gc
Sensor Log	SEN	25	0
RGB Color Histogram	RGB	75	1
HSV Color Histogram	HSV	75	1
Color Structure Descriptor	CSD	64	45
Scalable Color Descriptor	SCD	128	37
Color Layout Descriptor	CLD	120	27
Homogeneous Texture Descriptor	HTD	62	47
Edge Histogram Descriptor	EHD	80	22

5.2 Experiment Setting The experiments are designed to evaluate the performance of our two proposed models. First, we would like to test GOAT as a cost-sensitive feature selection model. We use the ordinary logistic classifier to evaluate the performance of GOAT and other feature selection models under different cost budget ratios. Second, we aim to test how well ETIC can classify the instances with a feature constraint. The details of the setting about extracted features, baselines and feature costs are introduced below.

5.2.1 Feature Setting We use various low-level feature extraction features such as color histogram in RGB and HSV color space and 5 kinds of MPEG-7 descriptors including color structure descriptor (CSD), scalable color descriptor (SCD), color layout descriptor (CLD), homogeneous texture descriptor (HTD) and edge histogram descriptor (EHD) [1], Although the group sets may be either joint or disjoint depending on the feature group property, we assume that all the group sets are disjoint for simplicity. The feature groups are summarized as Table 3. For User0 and User1 dataset, there are additional sensor features, which include context-aware features from the sensors of each user's mobile device, e.g. location-related features and device status features. The value of group cost (gc) is set to the proportion of the feature extraction time, which is measured based on the training data. For example, based on our computation, the ratio of the extraction time of HSV:CSD:SCD is 1:45:37. Moreover, the group cost of sensor log features is set to zero since it is simply extracted from mobile sensor data with negligible computation time. To evaluate the prediction performance, we use F1 score instead accuracy due to class imbalance in the datasets, which is computed as $\frac{2\times Precision\times Recall}{Precision+Recall}$. Each dataset is split equally for training and testing. Each experi-

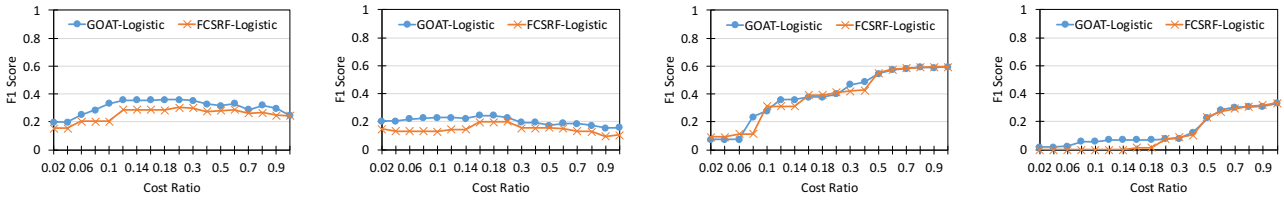
[4] $(ic) + (gc) = (1 + 2 + 4) + (10 + 20) = 37$

[5] $(ic) + (gc) = (1 + 2 + 3 + 4 + 5) + (10 + 20) = 45$

[6] The extracted features of these four datasets are available on the https://goo.gl/EzR33L.

Table 4: Feature cost setting

Setting		SEN	RGB	HSV	CSD	SCD	CLD	HTD	EHD
Balanced	individual cost	1	1	1	1	1	1	1	1
	group cost	0	1	1	45	37	27	47	22
Standard ML	individual cost	1	1	1	1	1	1	1	1
	group cost	0	0	0	0	0	0	0	0
Dominant-Group-Cost	individual cost	1	1	1	1	1	1	1	1
	group cost	0	10000	10000	450000	370000	270000	470000	220000



(a) User0 (b) User1 (c) DET124 (d) DET177

Figure 3: GOAT and FCSRF in logistic classifier on four datasets using the balanced setting

ment is conducted five times to eliminate the effect of randomness, and we compute the average F1 score of each model.

5.2.2 Baseline The first baseline is FCSRF [24] for the comparison of the proposed GOAT. FCSRF is used to produce the feature importance ranking, and we select the maximal subset of features satisfying the given cost budget by the ranking order as the input of the logistic classifier. We compute the feature probability of FCSRF based on Eq. (3.3) to consider the group cost initially. Moreover, to show that ETIC model can outperform the traditional RF model [6], we use the "FCSRF-RF" as another baseline. In addition, we construct other combinations such as "FCSRF-Logistic" and "FCSRF-ETIC" for comparison with our "GOAT-ETIC" framework. For hyperparameters, we set the number of trees to 50 and the maximum depth to infinity for GOAT, ETIC, FCSRF and RF. The number of used features K in each node of ETIC is set to 20.

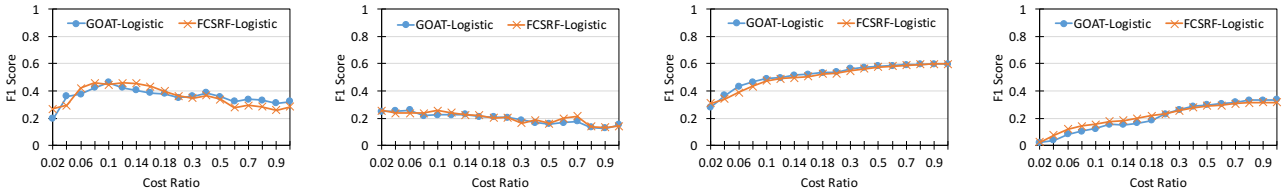
5.2.3 Cost Setting In order to evaluate the performance of our models under specified feature cost budget, first we set up three kinds of cost settings, summarized as Table 4[7]. The first setting is the *balanced setting*, which is an approximate estimation of reality based on the actual feature extraction time that computed beforehand. In this setting, the individual cost of each feature is set to 1 as a representation of the cost of the growth of dimensionality. This setting is equivalent to setting the value of gc using Table 3 with $\alpha = 1$ and $\beta = 1$. In the second setting, i.e. the *standard machine learning setting*, we would like to perform the experiment assuming the feature cost is independent. The individual cost is set to 1 and the group cost of all features is set to zero. Because the individual costs are all the same among features, this setting is equivalent to the traditional machine learning problem without the feature cost consideration, i.e. the balanced setting with $\alpha = 1$ and $\beta = 0$. In the third setting, i.e. the *dominant-group-cost setting*, we would like to test the circumstance that the group cost dominates the total cost. The individual cost is set to 1 and the group cost here is set to ten thousand times larger than the group cost in the balanced setting. This setting is equivalent to the balanced setting with $\alpha = 1$ and $\beta = 10000$.

5.3 Discussion In Fig. 3 to 8, the x-axis of each figure represents *cost ratio*, and the y-axis is F1 score. The cost ratio stands for the feature cost budget divided by the sum of all feature costs under each cost setting. If the cost ratio is set to 1, the task is equivalent to classical classification. We perform each model five times with different cost ratios[8] to stabilize results.

The first experiment is to test the performance of the proposed GOAT model for cost-sensitive feature selection, and we can see the improvement of GOAT over FCSRF in Fig. 3, 4 and 5. When the effects of group cost become larger, the superiority of GOAT expands. Note that in the dominant-group-cost setting of Fig. 5, the

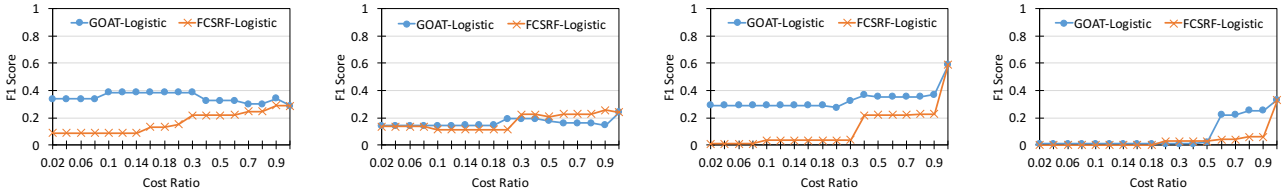
[7]The misclassification cost in Eq. (3.3) is discarded in the experiment since the misclassification cost for each class is regarded as the same. Hence, the total cost can be simplified to $TC(B) = \alpha(ic(B)) + \beta(gc(B))$.

[8]The cost ratios (x-axis) are set to $[0.02, 0.1]$ and $[0.1, 1]$ with interval 0.02 and 0.1 respectively.



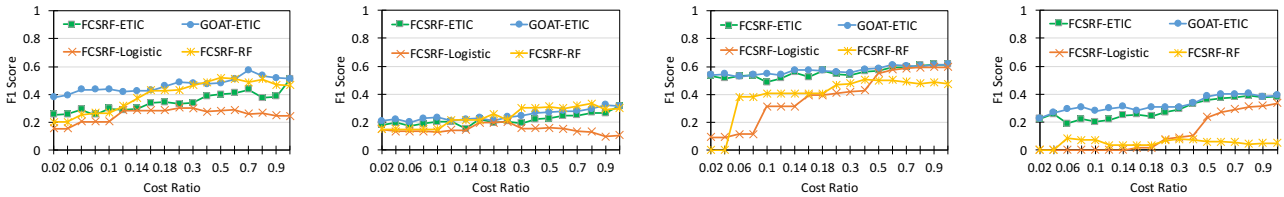
(a) User0 (b) User1 (c) DET124 (d) DET177

Figure 4: `GOAT` and FCSRF in logistic classifier on four datasets using the standard ML setting



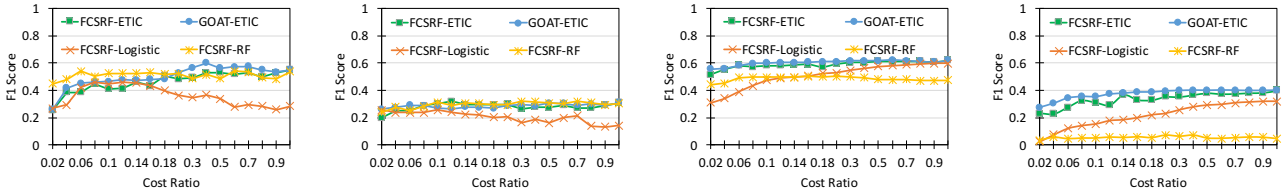
(a) User0 (b) User1 (c) DET124 (d) DET177

Figure 5: `GOAT` and FCSRF in logistic classifier on four datasets using the dominant-group-cost setting



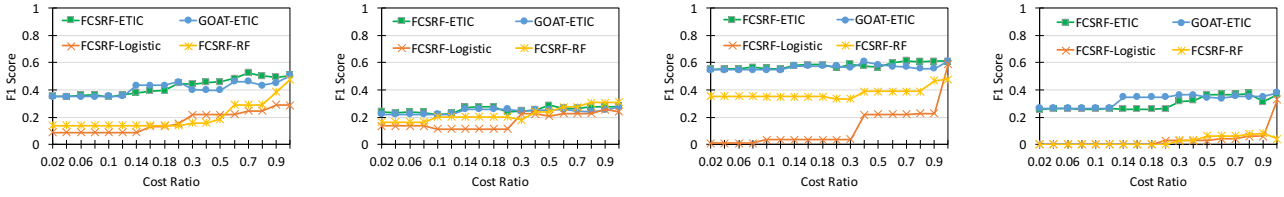
(a) User0 (b) User1 (c) DET124 (d) DET177

Figure 6: Performance of `ETIC` on four datasets using the balanced setting



(a) User0 (b) User1 (c) DET124 (d) DET177

Figure 7: Performance of `ETIC` on four datasets using the standard ML setting



(a) User0 (b) User1 (c) DET124 (d) DET177

Figure 8: Performance of `ETIC` on four datasets using the dominant-group-cost setting

results of two DET datasets degrade quickly when the cost ratio drops unlike the results in two User datasets because SEN features in the User datasets are not only cheap but also informative. Our GOAT can get at most 50% boosts over the FCSRF in three of the datasets in the dominant-group-cost setting. Even if the group cost is set to zero in the standard ML setting, our GOAT can usually get comparable performance.

As shown in Fig. 6, 7 and 8, our GOAT-ETIC beats the baselines in most situations. FCSRF-ETIC loses to the baselines when the cost ratio is pretty low in Fig. 6 and 7 because the FCSRF may select too few features with tight cost budget. Among all the three settings, our ETIC model performs especially well in the DET datasets because the image features are not suited for traditional random forest. ETIC uses multiple features to build each node, which can form a more complicated decision boundary in each node. In the User datasets, the RF model becomes competitive because the SEN features is powerful for learning the decision boundary with single feature. Note that when the cost budget is extremely tight, our GOAT-ETIC can always get higher or similar F1 score among all the settings and datasets, which shows the advantages of GOAT-ETIC.

6 Conclusions

In the paper, we propose both individual cost and group cost to formalize feature cost. We propose a two-stage framework GOAT-ETIC to integrate cost-sensitive feature selection and learning a model with feature-cost constraints. GOAT performs feature selection with the consideration of two proposed cost. ETIC learns an ensemble of trees using cascaded linear classifiers with feature-cost constraints based on the intermediate output from GOAT and guarantees that every testing instance will not run out of a given feature-cost budget.

For future work, the first we would like to do is to develop forest pruning techniques for ETIC, which can remove the rarely-used features so the feature cost of our ETIC model in prediction time can be further reduced. Second, we would like to extend the idea in [17] and [22] with our feature cost formulation to find the feature group information without specified feature groups.

References

[1] M. Bastan et al., *Bilvideo-7: an mpeg-7-compatible video indexing and retrieval system*, IEEE MultiMedia, 17 (2010), pp. 62–73.

[2] C.-K. Chou, C.-C. Lin, and M.-S. Chen, *Context-aware daily activity summarization with adaptive transmission*, in ACM/SIGAPP SAC, 2016.

[3] P. Domingos, *Metacost: A general method for making classifiers cost-sensitive*, in ACM SIGKDD, 1999.

[4] C. Elkan, *The foundations of cost-sensitive learning*, in IJCAI, 2001.

[5] X. Li, H. Zhao, and W. Zhu, *An exponent weighted algorithm for minimal cost feature selection*, IJMLC, 7 (2016), pp. 689–698.

[6] A. Liaw and M. Wiener, *Classification and regression by randomforest*, R news, 2 (2002), pp. 18–22.

[7] L.-P. Liu, Y. Yu, Y. Jiang, and Z.-H. Zhou, *Tefe: A time-efficient approach to feature extraction*, in IEEE ICDM, 2008.

[8] Q. Mao, L. Wang, S. Goodison, and Y. Sun, *Dimensionality reduction via graph structure learning*, in ACM SIGKDD, 2015.

[9] Q. Mao, L. Yang, L. Wang, S. Goodison, and Y. Sun, *Simpleppt: A simple principal tree algorithm*, in SDM, 2015.

[10] F. Min and Q. Liu, *A hierarchical model for test-cost-sensitive decision systems*, Information Sciences, 179 (2009), pp. 2442–2452.

[11] F. Nan, J. Wang, and V. Saligrama, *Feature-budgeted random forest*, in ICML, 2015.

[12] F. Nan, J. Wang, and V. Saligrama, *Pruning random forests for prediction on a budget*, in NIPS, 2016.

[13] M. Pazzani, C. Merz, P. Murphy, K. Ali, T. Hume, and C. Brunk, *Reducing misclassification costs*, in ICML, 1994.

[14] L. Polkowski, *Rough sets: Mathematical foundations*, Physica-Verlag, 2002.

[15] O. R. et al., *ImageNet Large Scale Visual Recognition Challenge*, IJCV, 115 (2015), pp. 211–252.

[16] S. Sheng and C. X. Ling, *Hybrid cost-sensitive decision tree*, in PKDD, 2005.

[17] N. Subrahmanya and Y. C. Shin, *A variational bayesian framework for group feature selection*, IJMLC, 4 (2013), pp. 609–619.

[18] P. D. Turney, *Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm*, JAIR, 2 (1995), pp. 369–409.

[19] X.-Z. Wang, L.-C. Dong, and J.-H. Yan, *Maximum ambiguity-based sample selection in fuzzy decision tree induction*, IEEE TKDE, 24 (2012), pp. 1491–1505.

[20] X.-Z. Wang, Y.-L. He, and D. D. Wang, *Non-naive bayesian classifiers for classification problems with continuous attributes*, IEEE TCYB, 44 (2014), pp. 21–39.

[21] Z. Xu, M. Kusner, K. Weinberger, and M. Chen, *Cost-sensitive tree of classifiers*, in ICML, 2013.

[22] S. Yang, L. Yuan, Y.-C. Lai, X. Shen, P. Wonka, and J. Ye, *Feature grouping and selection over an undirected graph*, in Graph Embedding for Pattern Analysis, Springer, 2013, pp. 27–43.

[23] S. Zhao, E. C. Tsang, D. Chen, and X. Wang, *Building a rule-based classifierxa fuzzy-rough set approach*, IEEE TKDE, 22 (2010), pp. 624–638.

[24] Q. Zhou, H. Zhou, and T. Li, *Cost-sensitive feature selection using random forest: Selecting low-cost subsets of informative features*, KBS, 95 (2016), pp. 1–11.