Information Sciences 568 (2021) 448–462

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Hierarchical feature selection with multi-granularity clustering structure

Shunxin Guo [a,c], Hong Zhao [a,b,*], Wenyuan Yang [c]

[a] School of Computer Science in Minnan Normal University, Zhangzhou, Fujian 363000, China
[b] Key Laboratory of Data Science and Intelligence Application, Fujian Province University, Zhangzhou, Fujian, 363000, China
[c] Fujian Key Laboratory of Granular Computing and Application (Minnan Normal University), Zhangzhou, Fujian 363000, China

ARTICLE INFO

Article history:
Received 29 April 2020
Received in revised form 6 January 2021
Accepted 4 April 2021
Available online 20 April 2021

Keywords:
Granular computing
Hierarchical feature selection
Multi-granularity clustering
Semantic gap

ABSTRACT

Hierarchical feature selection addresses the issues caused by the presence of high-dimensional features in multi-category classification systems with hierarchical structures. Granular calculations are made to analyze the hierarchical relationships among categories when selecting the optimal feature subset. However, semantic hierarchy-based feature selection methods are prone to the semantic gap problem, which affects classification accuracy. In this paper, we propose a hierarchical feature selection method with a multi-granularity clustering structure that can effectively alleviate the semantic gap problem. Firstly, a hierarchical structure is constructed via bottom-up multi-granularity clustering based on feature similarities rather than semantic categories. This clustering hierarchy is conducive to solving semantic gap problems in the existing hierarchy. Secondly, the optimal feature subset is selected using the $\ell_{1,2}$-norms in each hierarchy's granularity layer. This joint minimization approach can retain both the granularity layers' shared features and granularity-specific features. Finally, we execute hierarchical classification according to the granular structure in a coarse to fine sequence. Extensive experiments demonstrate that the proposed method outperforms several state-of-the-art hierarchical feature selection approaches.

1. Introduction

High-dimensional features present challenges in large-scale classification tasks. Feature selection is an essential preprocessing step for high-dimensional features [43] and has received wide attention in recent years [39]. Feature selection methods based on granular computing [30,42,31,38] are effective dimension-reduction methods that use data granularity. Liang et al. [23] presented a feature selection algorithm based on multi-granularity for large-scale datasets. The strategy is to divide the dataset into different fine-grained subsets and select features form each one. Dong et al. [8] proposed a feature selection model that considers granular information. The model improves the sample granularity neighborhood rough set under different granularity conditions and feature subset qualities. Liao et al. [24] considered using multi-granularity to evaluate and selected feature granularity based on the error confidence of feature values. The feature selection methods

* Corresponding author at: School of Computer Science in Minnan Normal University, Zhangzhou, Fujian 363000, China; Key Laboratory of Data Science and Intelligence Application, Fujian Province University, Zhangzhou, Fujian, 363000, China.
E-mail address: hongzhaocn@163.com (H. Zhao).

https://doi.org/10.1016/j.ins.2021.04.046
0020-0255/

described above consider that granular information can improve calculation speed and provide models that are more compact with better generalizability.

Besides high-dimensional problems, continuous increases in category number are another crucial challenge for large-scale classification tasks. Many application areas use semantic hierarchies to organize categories as their number increases [33], such as in ImageNet [5], Wikipedia [17], and text data [12]. The semantic hierarchy is crucial prior knowledge that reflects the semantic granularity from coarse categories to fine ones. Recently, the solving of practical application problems based on semantic hierarchy methods has achieved significant progress. Ma et al. [26] proposed an algorithm to deal with missing labels according to categories with semantic feature relevance and a semantic layer. Zhang et al. [47] presented an image retrieval method based on semantic hierarchy that eliminates the semantic and intention gaps in content-based image retrieval. Mittelman et al. [27] proposed a Bayesian generation model to learn the semantic layer, which can handle fine-grained visual recognition problems with thousands of categories. Such methods using a semantic hierarchy as prior knowledge have become very influential in addressing problems in various fields.

A large-scale classification task can be divided into a group of small sub-classification tasks using the semantic hierarchy [46,37,14]. The hierarchical feature selection method selects a feature subset for each sub-class to eliminate feature redundancy in layers of each granularity [36]. The hierarchical feature selection method has been applied in various fields. Ruvolo et al. [32] proposed a hierarchical feature selection method to obtain fixed low-level features for feature aggregation in audio classification. Freeman et al. [13] proposed a method to improve classifier accuracy by combining feature selection with a hierarchical classifier based on a genetic algorithm. Cheng et al. [3] proposed an effective hierarchical feature selection and learning fusion strategy for computer vision tasks based on image segmentation and super-pixel extraction. In these studies, the selected feature subsets make it easier to distinguish features of different granularities at each semantic level. The semantic hierarchy of categories plays a crucial role in hierarchical feature selection because this process is overwhelmingly dependent on the semantic tree structure. However, there is a semantic gap in the semantic hierarchy in many applications. For example, whales belong to the class mammals in terms of semantics, as shown in Fig. 1. In terms of computer image recognition, however, whales are more likely to be classified as *Fish*, which affects learning based on the semantic hierarchy.

In this paper, we propose a hierarchical feature selection method based on multi-granularity clustering of sample feature similarity. This method considers the importance of special granular features and shared granularity layer features. Firstly, we perform bottom-up information granulation clustering according to the fundamental data feature matrix to obtain a clustering hierarchy suitable for sample classification. The strategy is to construct a hierarchy based on the distance among sample features. This helps alleviate the semantic gap problem [35], assuming that there is a hierarchical structure based on semantic knowledge in the classification task. Secondly, we separately apply $\ell_{1,2}$-norms regularization constraints on the decomposed model's two components. The purpose is to capture the shared features and special features among the classes on each granularity layer. Each granularity layer describes the inherent information of the classes embedded in the feature space. The features in each granularity layer are partially redundant, which represent the same data features. Each granularity has some special features that are different from other granularities. Finally, the test sample is classified into the correct category using the classifier to assist the classification process at each granularity layer. This hierarchical classification method is independent and is not affected by the classification relationship between the upper and lower granularity layers. Extensive experiments on several real datasets and a synthetic dataset support our analysis and conclusions. The results show that our model alleviates the semantic gap in the semantic hierarchy and improves feature subset selection performance.

The remainder of this paper is organized as follows. In Section 2, we describe the details of our proposed approach. Details of our experimental setup are presented in Section 3. In Section 4, we present the experimental results to demonstrate the performance of our approach with several real-world and synthetic dataset. Finally, conclusions are drawn and future work suggested in Section 5.

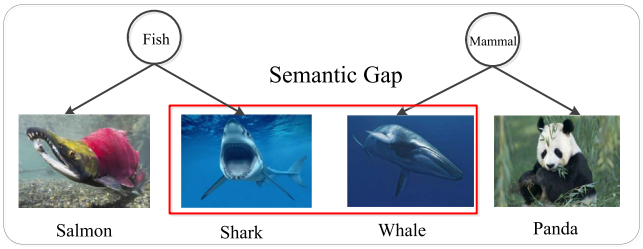

Fig. 1. An example diagram describing the semantic gap.

2. Hierarchical feature selection method with clustering hierarchy

In this section, we introduce the framework of the proposed model and its three main parts.

2.1. Framework overview

The basic flowchart of the HCFS is shown in Fig. 2. This model is composed of three main parts:

(1) Multi-granularity clustering from fine- to coarse-grained. The hierarchical clustering approach is to construct a clustering tree to manage clusters of different granularities. The coarse-grained category is clustered based on the similarities among the sample features of different fine-grained categories.
(2) Hierarchical feature selection from coarse to fine-grained. The feature matrix of each coarse-grained layer in the hierarchy is decomposed into two components. Also, joint $\ell_{1,2}$-norms regularization is performed on each component, respectively.
(3) Hierarchical classification from coarse to fine-grained. The tested samples are classified from top to bottom to verify the effectiveness of the proposed feature selection method.

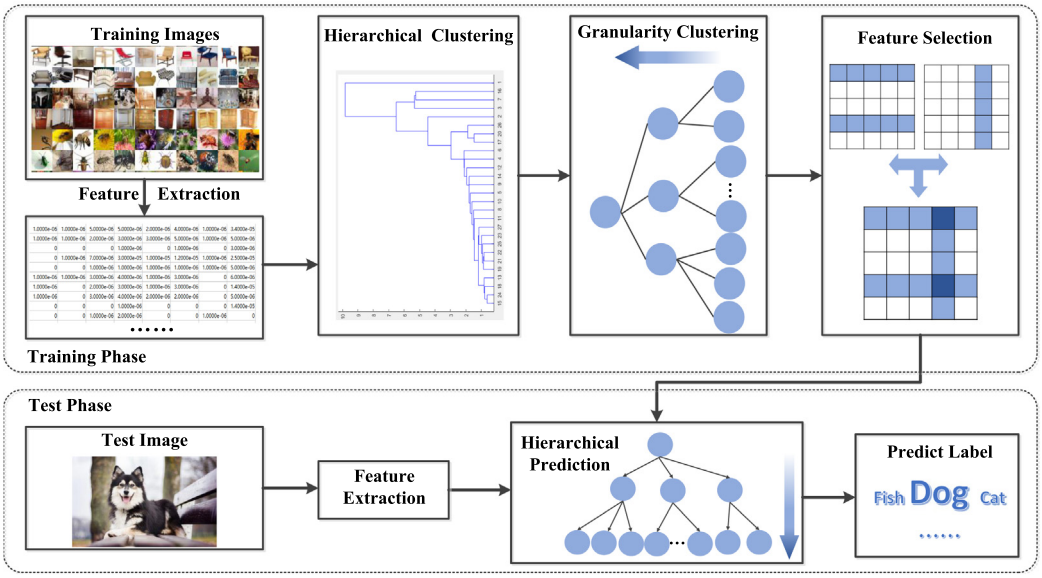

Fig. 2. Diagram of the HCFS flowchart.

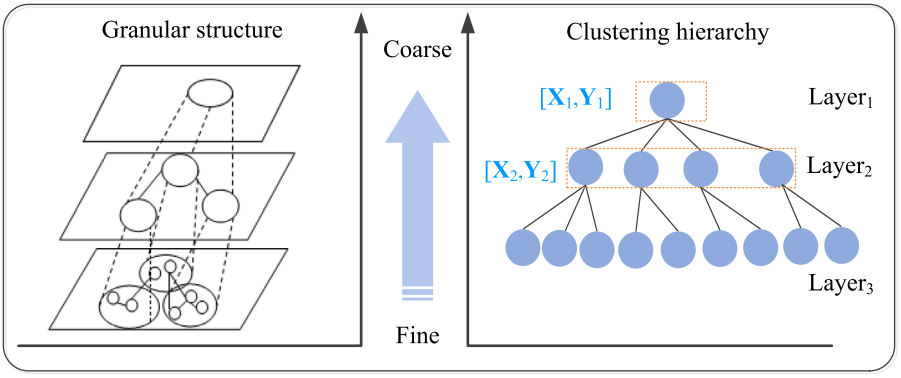

Fig. 3. Diagram of hierarchical clustering from fine to coarse ($l = 2$).

2.2. Multi-granularity clustering from fine- to coarse-grained

We build a hierarchy from bottom to the top based on the similarity of data features, as shown in Fig. 3. The clustering results at different layers reflect the granular structure of the data from concrete to abstract, which also conforms to the computer information processing mechanism from fine to coarse [1]. The layer of lowest granularity in the hierarchy is the data layer. Moreover, the data feature space description is the most detailed and specific. There are two coarse granularity layers in the clustering hierarchy ($l = 2$). The clustering hierarchy established based on feature similarity is more conducive to distinguishing between shared and unique features in the feature space. Let $\mathbf{X} \in \mathbb{R}^{N\times d}$ be a data matrix, where N and d are the numbers of samples and features, respectively. The vector $\mathbf{x}_i$ denotes the i-th sample of the matrix $\mathbf{X}$, and $x_{i\epsilon}$ denotes the ϵ-th feature value of the $\mathbf{x}_i$. We use the Euclidean distance [41] to calculate the distance (similarity) D_{ij} between data points i and j of different categories. The formula for calculating the distance is as follows:

$$D_{ij} = \sqrt{\sum_{\epsilon=1}^{d} \left(x_{i\epsilon} - x_{j\epsilon}\right)^2}. \tag{1}$$

Starting from the original data, we calculate the correlation distances of the sample's features to cluster them from fine to coarse.

Algorithm 1. Multi-granularity clustering hierarchy

Input: Input data matrix $\mathbf{X} \in \mathbb{R}^{N\times d}$ and labels $\mathbf{y} \in \mathbb{R}^{N\times 1}$. We let the number of clusters on a coarse-grained layer be *num*.
Output: The category clusters of a coarse-grained layer.
1: Aggregate the data matrix into m categories $\mathbf{C}_m = \{C_1, \cdots, C_j, \cdots, C_m\} \in \mathbb{R}^{m\times d}$, where m is the number of classes.
// All sample points are treated as an independent cluster.
2: $n = num + 1$;
3: **for** $\gamma = m : n$ **do**
4: Select the two samples $\{C_i, C_j\}$ from $\mathbf{C}_\gamma$ with the minimal D_{ij} ;
5: Obtain the corresponding category cluster $\{C_i, C_j\}$;
6: $C_T = C_i \bigcup C_j$ and $\mathbf{C}_{\gamma-1} = (\mathbf{C}_\gamma - \{C_i, C_j\}) \bigcup C_T$;
7:**end for**
8: **return** $\mathbf{C}_{num}$;

Generally, there are three methods of calculating the distance between data points: single linkage, complete linkage, and average linkage. We choose single linkage in HCFS when multiple fine-grained (combined data points) continue to cluster upwards. The single linkage defines the distance between two fine granularities of maximum similarity as the distance between two coarse granularities. Algorithm 1 mainly introduces the process of multi-granularity clustering.

Let $\mathbf{Y} \in \mathbb{R}^{m\times 1}$ be a class matrix, where m is the number of all classes. A hierarchical tree obtained by granular clustering is defined as a node-branch pair $(\mathbf{Y}, \prec)$. Each element of the $\mathbf{Y}$ is represented as a class in the hierarchical tree structure and a granularity in the granularity structure. Symbol $\prec$ stands for the relationship between the classes and nodes of the tree structure [20]. If $u \prec v$, it means that u is a child node of v and also represents a branch of the hierarchical tree structure. The parameters u and v belong to two upper and lower granularity layers, respectively. The properties of the relationships contained in this hierarchy are as follows:

(1) Asymmetry: If $u \prec v$, then $v \not\prec u$ for each $u, v \in \mathbf{Y}$.
(2) Anti-reflexivity: $u \not\prec u$ for each $u \in \mathbf{Y}$.
(3) Transitivity: If $u \prec v$ and $v \prec w$, then $u \prec w$ for each $u, v, w \in \mathbf{Y}$.

The dataset categories are equivalent to the nodes in the tree structure and the granularity in the granularity structure. The root node contains all categories, and the coarse layer contains all fine grains. Each leaf node (the finest granularity) of the hierarchical tree has a unique path starting from the root (the coarsest granularity). In the established clustering hierarchy, we use $\left\{\left(\mathbf{X}_i^1, \mathbf{y}_i^1\right), \cdots, \left(\mathbf{X}_i^j, \mathbf{y}_i^j\right), \cdots, \left(\mathbf{X}_i^{m_i}, \mathbf{y}_i^{m_i}\right)\right\}$ to represent the data sample of the m_i coarser classes of the i-th coarser granularity layer, where $\mathbf{X}_i^j \in \mathbb{R}^{n_i^j\times d}$ is the data matrix of the j-th class with each column as a sample and $\mathbf{X}_i = \left\{\mathbf{X}_i^1, \cdots, \mathbf{X}_i^j, \cdots, \mathbf{X}_i^{m_i}\right\}$. Let $\mathbf{y}_i^j \in \mathbb{R}^{n_i^j}$ be a vector, where n_i^j is the number of samples for the j-th class at the i-th granularity layer and $\mathbf{Y}_i = \left\{\mathbf{y}_i^1, \cdots, \mathbf{y}_i^j, \cdots, \mathbf{y}^{\mathbf{m}_i}\right\}$. As shown in Fig. 3, the second granularity layer's samples set is $\mathbf{X}_2 = \left\{\mathbf{X}_2^1, \mathbf{X}_2^2, \mathbf{X}_2^3, \mathbf{X}_2^4\right\}$. The corresponding class set is $\mathbf{Y}_2 = \{\mathbf{y}_2^1, \mathbf{y}_2^2, \mathbf{y}_2^3, \mathbf{y}_2^4\}$. Table 1 describes the most commonly used symbols in this paper.

Table 1
Description of the symbols used in this article.

Symbol	Meaning
l	Number of coarser layers in the hierarchy
d	Sample feature dimensionality
i	i-th coarser granularity layer of the granularity hierarchy, $i \in \{1, 2, \cdots, l\}$
m_i	Number of grained classes at i-th coarser granularity layer
k	k-th iteration of gradient descent algorithm
$\mathbf{X}_i$	Sample data matrix at the i-th granularity layer
$\mathbf{Y}_i$	Class matrix at the i-th granularity layer

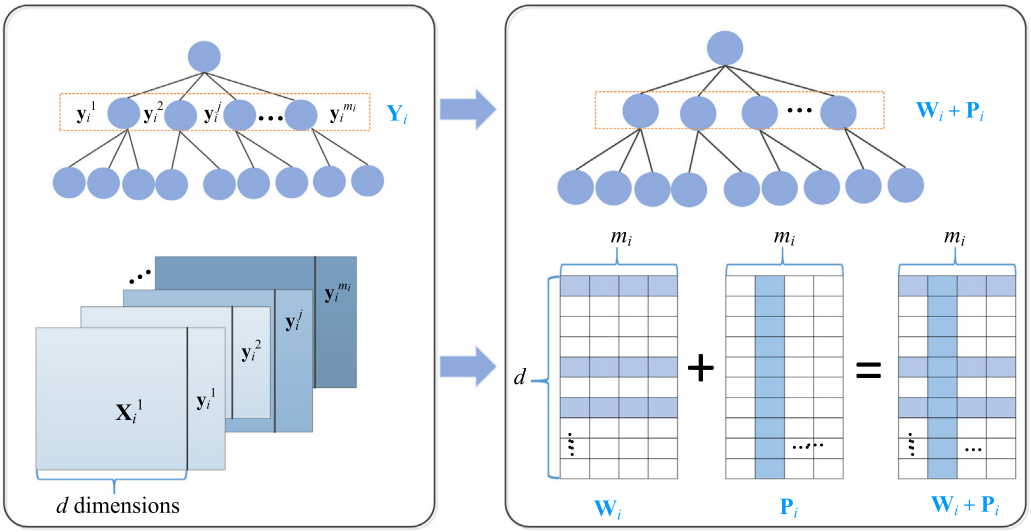

Fig. 4. Model of hierarchical feature selection at the i-th granularity layer.

2.3. Hierarchical feature selection from coarse to fine-grained

We decompose the weight matrix of each coarse-grained layer into the sum of two components $\mathbf{W}_i$ and $\mathbf{P}_i$ (as shown in Fig. 4), where $\mathbf{W}_i = [\mathbf{w}_i^1, \cdots, \mathbf{w}_i^{m_i}] \in \mathbb{R}^{d \times m_i}$ and $\mathbf{P}_i = [\mathbf{p}_i^1, \cdots, \mathbf{p}_i^{m_i}] \in \mathbb{R}^{d \times m_i}$ are the feature weight matrixes. We explore the use of $\ell_{1,2}$ regularization, which is an alternative to the ℓ_1 counterpart. Xu et al. [45] proved that $\ell_{1,2}$ regularization is an unbiased estimation, which imposes strong sparsity on the minimization problem. In addition, $\ell_{1,2}$ regularization not only provides sparse solutions close to ℓ_0 but also has high computational efficiency. In the context of a granularity hierarchy, the i-th granularity layer optimization problem is to minimize $J(\mathbf{W}_i, \mathbf{P}_i)$:

$$J(\mathbf{W}_i, \mathbf{P}_i) = \sum_{j=1}^{m_i} \frac{1}{n_i^j} \left\| \mathbf{X}_i^j \left(\mathbf{w}_i^j + \mathbf{p}_i^j \right) - \mathbf{y}_i^j \right\|^2 + \lambda_1 \|\mathbf{W}_i\|_{1,2} + \lambda_2 \left\| \mathbf{P}_i^T \right\|_{1,2}. \tag{2}$$

The first regularization term $\mathbf{W}_i$ can capture the shared features and the second regularization term $\mathbf{P}_i$ finds the specific features at each granularity layer. Parameters λ_1 and λ_2 are nonnegative parameters that control these two components. The first regularization term is based on the $\ell_{1,2}$-norms joint optimization on the row group of $\mathbf{W}_i$. It limits the rows of the optimal solution $\mathbf{W}_i^{(*)}$ to be composed of zero or non-zero elements. Each level of granularity in the hierarchy should select a set of common features. However, each granularity feature space has its own special features in practical applications. We introduce a second regularization term $\mathbf{P}_i$ based on the same $\ell_{1,2}$-norms to find the specific features in the granularity feature space. Similarly, the column of the optimal solution $\mathbf{P}_i^{(*)}$ is composed of all zero or non-zero elements, and the non-zero column corresponds to a special granularity. Intuitively, if the i-th column of $\mathbf{P}_i^{(*)}$ is non-zero, then the i-th column of $\mathbf{W}_i^{(*)}$+$\mathbf{P}_i^{(*)}$ is also non-zero. Therefore, the j-th granularity does not have a common feature set with other granularities, and it is considered to have a special feature space. At the same time, for other granularities corresponding to the zero columns of $\mathbf{P}_i^{(*)}$, they share the common feature set captured by the non-zero row of $\mathbf{W}_i^{(*)}$. Fig. 4 shows the feature selection process of the i-th granularity layer based on the shared features and the special features of the granularity itself.

We describe how to resolve the HCFS formulation in Eq. (2) and denote

$$\begin{aligned} t(\mathbf{W}_i,\mathbf{P}_i) &= \sum_{j=1}^{m_i}\frac{1}{n_i^j}\left\|\mathbf{X}_i^j\left(\mathbf{w}_i^j+\mathbf{p}_i^j\right)-\mathbf{y}_i^j\right\|^2, \\ r(\mathbf{W}_i,\mathbf{P}_i) &= \lambda_1\|\mathbf{W}_i\|_{1,2}+\lambda_2\left\|\mathbf{P}_i^T\right\|_{1,2}, \end{aligned} \tag{3}$$

where $t(\mathbf{W}_i,\mathbf{P}_i)$ is the empirical loss function and $r(\mathbf{W}_i,\mathbf{P}_i)$ is the regularization term. The objective function in Eq. (3) is a composite function of the differential term $t(\mathbf{W}_i,\mathbf{P}_i)$ and the non-differential term $r(\mathbf{W}_i,\mathbf{P}_i)$. Denote

$$T_{R,S,\eta}(\mathbf{W}_i,\mathbf{P}_i)=t(\mathbf{R}_i,\mathbf{S}_i)+\left\langle\frac{\partial t(\mathbf{R}_i,\mathbf{S}_i)}{\partial\mathbf{R}_i},\mathbf{W}_i-\mathbf{R}_i\right\rangle+\frac{\eta}{2}\|\mathbf{W}_i-\mathbf{R}_i\|_F^2+\left\langle\frac{\partial t(\mathbf{R}_i,\mathbf{S}_i)}{\partial\mathbf{S}_i},\mathbf{P}_i-\mathbf{S}_i\right\rangle+\frac{\eta}{2}\|\mathbf{P}_i-\mathbf{S}_i\|_F^2, \tag{4}$$

which is the first-order Taylor expansion of $t(\mathbf{W}_i,\mathbf{P}_i)$ at $(\mathbf{R}_i,\mathbf{S}_i)$, with the squared Euclidean distance between $(\mathbf{W}_i,\mathbf{P}_i)$ and $(\mathbf{R}_i,\mathbf{S}_i)$ as the regularization term. We use the accelerated gradient descent method to solve the optimization convergence problem, which generates the solution at the k-th iteration ($k\geqslant 1$) by calculating the following approximate operator [25]. The expression is as follows:

$$\left(\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)}\right)=\arg\min_{\mathbf{W}_i,\mathbf{P}_i}T_{\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)},\eta^{(k)}}(\mathbf{W}_i,\mathbf{P}_i)+r(\mathbf{W}_i,\mathbf{P}_i) \tag{5}$$

where $\mathbf{R}_i^{(1)}=\mathbf{W}_i^{(0)},\mathbf{S}_i^{(1)}=\mathbf{P}_i^{(0)}$ and $\mathbf{R}_i^{(k+1)}=\mathbf{W}_i^{(k)}+\alpha_k\left(\mathbf{W}_i^{(k)}-\mathbf{W}_i^{(k-1)}\right),\mathbf{S}_i^{(k+1)}=\mathbf{P}_i^{(k)}+\alpha_k\left(\mathbf{P}_i^{(k)}-\mathbf{P}_i^{(k-1)}\right)$ with $k\geqslant 1$.

Due to the decomposability of Eq. (5), we can transform it into the following two separate proximal operator problems:

$$\begin{aligned} \mathbf{W}_i^{(k)} &= \arg\min_{\mathbf{W}_i}\tfrac{1}{2}\left\|\mathbf{W}_i-\left(\mathbf{R}_i^{(k)}-\tfrac{1}{\eta_k}\nabla_{\mathbf{R}_i}l\left(\mathbf{R}_i^{(k)},\mathbf{S}_i^{(k)}\right)\right)\right\|_F^2+\tfrac{\lambda_1}{\eta_k}\|\mathbf{W}_i\|_{1,2} \\ \mathbf{P}_i^{(k)} &= \arg\min_{\mathbf{P}_i}\tfrac{1}{2}\left\|\mathbf{P}_i-\left(\mathbf{S}_i^{(k)}-\tfrac{1}{\eta_k}\nabla_{\mathbf{S}_i}l\left(\mathbf{R}_i^{(k)},\mathbf{S}_i^{(k)}\right)\right)\right\|_F^2+\tfrac{\lambda_2}{\eta_k}\left\|\mathbf{P}_i^T\right\|_{1,2}, \end{aligned} \tag{6}$$

where $\nabla_{\mathbf{R}_i}l\left(\mathbf{R}_i^{(k)},\mathbf{S}_i^{(k)}\right)$ and $\nabla_{\mathbf{S}_i}l\left(\mathbf{R}_i^{(k)},\mathbf{S}_i^{(k)}\right)$ are the partial derivatives of $l(\mathbf{R}_i,\mathbf{S}_i)$ with respect to $\mathbf{S}_i$ and $\mathbf{R}_i$ at $\left(\mathbf{R}_i^{(k)},\mathbf{S}_i^{(k)}\right)$.

The proper step size $\eta^{(k)}(k\geqslant 1)$ is set by finding the smallest nonnegative integer m_ik, such that with $\eta^{(k)}=2^{m_ik}\eta^{(k-1)}$. Denote

$$t\left(\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)}\right)\leqslant T_{\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)},\eta^{(k)}}\left(\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)}\right). \tag{7}$$

We set $\alpha_k=(t_{k-1}-1)/t_k$, where $t_0=1$ as suggested by reference [2] and $t_k=\left(1+\sqrt{t_{k-1}^2+1}\right)/2$ for $k\geqslant 1$.

The coefficient α_k has an essential influence on the convergence of the algorithm. Then, let $\left(\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)}\right)$ be generated by Eq. (5) and use a properly chosen $\eta^{(k)}$ satisfying Eq. (7). For any $k\geqslant 1$, we can obtain $f\left(\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)}\right)-f\left(\mathbf{W}_i^{(*)},\mathbf{P}_i^{(*)}\right)=O\left(\frac{1}{k^2}\right)$, where $f(\cdot,\cdot)$ refers to the objective function and $f\left(\mathbf{W}_i^{(*)},\mathbf{P}_i^{(*)}\right)$ indicates the optimal solution in Eq. (2). The detailed proof of convergence analysis used to prove the HCFS is similar to that in [16]. Algorithm 2 mainly describes the process of the hierarchical feature selection.

Algorithm 2. Hierarchical feature selection

Input: Input a l-level clustering tree structure.
Output: The feature sorting set of a coarse-grained layer.
1: **for** i = 1: l **do**
2: $t_0=1,k=1$;
3: **repeat**
4: $\alpha_k=(t_{k-1}-1)/t_k$;
5: $t_k=\left(1+\sqrt{t_{k-1}^2+1}\right)/2$;
6: $k=k+1$;
7: Update $\mathbf{W}_i^{(k)}$ and $\mathbf{P}_i^{(k)}$ through Eq. (6);
8: **until** $f\left(\mathbf{W}_i^{(k)},\mathbf{P}_i^{(k)}\right)-f\left(\mathbf{W}_i^{(*)},\mathbf{P}_i^{(*)}\right)=O\left(\frac{1}{k^2}\right)$;
9: Obtain optimal solution $\mathbf{W}_i^{(*)}$ and $\mathbf{P}_i^{(*)}$ at each granularity layer;
10: Obtain the feature sorting set at i-th granularity layer $\mathbf{B}_i$;
11: **end for**
12: **return** $\mathbf{B}=[\mathbf{B}_1;\mathbf{B}_2;\cdots;\mathbf{B}_l]$;

2.4. Hierarchical classification from coarse- to fine-grained

We predict the category of a test sample according to the established granularity level. The hierarchical classification method is based on using the model of the upper layer to predict the category of the lower granularity layer. We select the feature subsets from the training sets and test them on the test sets using the classifier. In the experiment, we used the 10-fold cross-validation method in the classification process. Specifically, 90% of the dataset is used for model training and 10% is used to test the effectiveness of the selected feature subset.

The test sample is predicted from coarse- to fine-grained. Fig. 5 gives an intuitive example to explain the hierarchical classification process. The test sample gradually predicts from the coarse-grained *Object* to the fine-grained *Fish* class and, ultimately, stops at the finest-grained *Salmon* class.

3. Experimental settings

In this section, we first describe the actual datasets used for experiments. Then, we summarize the performance evaluation indicators. Finally, the methods used to compare baseline layer feature selection results are discussed. All experiments were performed on a Windows 10 desktop computer with 24.0 GB memory and a 3.40 GHz Intel Core i7-3770 CPU.

3.1. Experimental datasets

Six real-world datasets were used in the experiments, including two protein datasets and four image datasets. The feature distribution was extracted from [40] for the two protein datasets. For the image datasets, we extracted features with the VGG19 model [34] pre-trained with the ImageNet dataset. The VGG19 model and its parameters can be downloaded from https://github.com/SnailTyan/caffemodel-zoo/tree/master/VGG19.

The categories of all these datasets belong to a category hierarchy, where the correct class label of each sample is the grain of the finest granularity. We briefly describe the size of the dataset in Table 2. The "Node" column is the number of all nodes in the semantic hierarchical tree structure, including the real class nodes in the leaf level and their ancestor class nodes.

(1) DD [7] and F194 [49] datasets: The two protein datasets are the fold-recognition datasets provided by Ding and Dubchak [7], which are based on the SCOP [28] structure. They are similar in structure, with 473 features (five groups: amino acid, 1-gram, 2-gram, global, local) and a three-level hierarchy. The DD dataset includes 3020 training samples and 605 test samples, of which there are 27 true fine-grained classes. The F194 dataset includes 7105 training samples and 1420 test samples,and there are 194 true fine-grained classes.
(2) CLEF [6] dataset: This is a medical X-ray image dataset, which was used in the experiment as a small image dataset. Its granularity structure includes four granularity layers, 80 feature dimensions, and 80 fine-grained categories.
(3) Car196 [21] dataset: This is a large-scale dataset containing 196 car types for fine-grained classification tasks. There are 196 categories with 8144 training images and 7541 test images. Each category is distinguished by year, manufacturer, and model.
(4) Cifar100 [22] dataset: This dataset was collected by Alex et al. It has 100 fine-grained classes, each of which contains 500 training images and 100 test images. The 100 fine-grained categories in Cifar100 are divided into 20 coarse-grained categories. Each image comes with a fine-grained category and a coarse-grained category.
(5) SUN324 [44] dataset: This is a Scene UNderstanding image dataset, which includes indoor pictures, outdoor natural scenery, outdoor buildings, and other pictures. This dataset contains 67,665 samples and 4096-dimensional features obtained through deep learning. Besides, there are 324 granularities in the fine-grained layer. The multi-label categories of some samples needed to be processed. Multi-label samples need to be deleted first because the single-label data was mainly considered in these experiments.

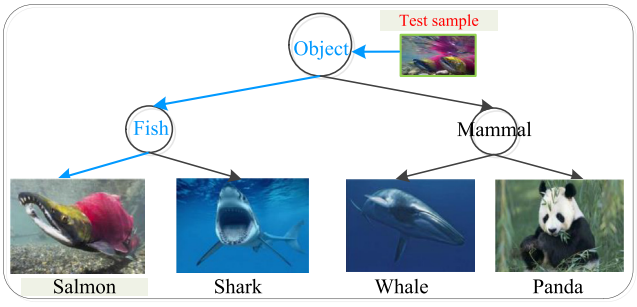

Fig. 5. Example providing an intuitive understanding of coarse- to fine-grained classification.

Table 2
Dataset description.

Dataset	Training	Testing	Features	Labels	Nodes	Depth
DD	3020	605	473	27	32	3
F194	7105	1420	473	194	202	3
CLEF	8368	939	80	80	88	4
Car196	8144	7541	4096	196	206	3
Cifar100	50,000	10,000	4096	100	121	3
SUN324	45,109	22,556	4096	324	343	4

3.2. Evaluation measures

Traditional multi-class task evaluation indicators, such as *F*-scores, can measure a classifier's ability to classify different categories. However, they cannot correctly describe the degree of misclassification in the category hierarchy. Therefore, the evaluation index of hierarchical feature selection should be different from the traditional classification task. Kosmopoulos et al. [20] introduced some measures for hierarchical classification.

The *tree induced error* (TIE) is a measure of the distance between a sample's predicted category and the correct category in the hierarchical structure [11]. It measures the number of edges that need to pass from the predicted label to the true label. The TIE can reflect the degree to which the sample is misclassified in the hierarchical structure, and its formal representation is as follows:

$$TIE(y,\hat{y}) = |E_h(y,\hat{y})|,$$

where y is the true label of a sample $\mathbf{x}$ and $\hat{y}$ are the predicted labels of $\mathbf{x}$. $E_h(y,\hat{y})$ represents the set of edges from y to $\hat{y}$ and $|\cdot|$ is expressed as the number of elements in the set. Hierarchical precision and recall can reflect the affiliation of the categories in the hierarchy, thus measuring the degree of error, and are widely used. The hierarchical precision and recall of the set are determined as:

$$P_h = \frac{|y_{all} \cap \widehat{y_{all}}|}{|\widehat{y_{all}}|}, \quad R_h = \frac{|y_{all} \cap \widehat{y_{all}}|}{|y_{all}|}.$$

The hierarchical F1-measure evaluation based on the extended set is defined as follows:
$\mathrm{F1} = \frac{2 \cdot R_h \cdot P_h}{R_h + P_h}$.

There is also an evaluation standard (accuracy) that can be applied to hierarchical and flat structures. Accuracy can evaluate the relationship between predicted labels and true labels to reflect the pros and cons of the classifier. The definition of *acc* is as follows:

$$acc = \frac{\sum_{j=1}^{n} a(\hat{y},y)}{n},$$

where n is the number of test samples and

$$a(\hat{y},y) = \begin{cases} 1, & \text{if} \quad \hat{y} = y; \\ 0, & \text{if} \quad \hat{y} \neq y. \end{cases}$$

3.3. Comparison methods

The proposed HCFS is compared with the following five hierarchical feature selection methods to verify the selected feature subset's performance.

(1) HRelif: Relief is a statistical-based feature selection method that has advantages in terms of the learning time and accuracy of the learned concepts, and can select statistics-related features [19]. The hierarchical feature selection method HRelif is an extension of the classical Relief algorithm based on the data's hierarchical structure. We record the experimental results when *relief* = 5 in this approach.
(2) HFisher: Fisher scores rely totally on labeled training data to select features with the best recognition capabilities [9]. HFisher method is an improved hierarchical feature selection method based on Fisher score and hierarchical structure.
(3) HFSNM: The feature selection method emphasizes joint $\ell_{2,1}$-norms minimization in both the loss function and regularization. It selects features with joint sparsity in all data points. HFSNM is a hierarchical feature selection method that can be applied to hierarchical structures by modifying FSNM [29].
(4) HmRMR: The minimum redundancy and maximum correlation (mRMR) dependence criterion based on mutual information is a scheme used to select the best features. It can directly avoid the difficulty in realizing the maximum depen-

dency condition using an equivalent form. HmRMR [18] is a hierarchical feature selection method that can select features in each layer of the hierarchy to obtain the feature subset with minimum redundancy and maximum correlation.
(5) HiFSRR: The hierarchical feature selection algorithm fully considers the relationships among the hierarchy's granularities. It selects different feature subsets for each node's granularity in the hierarchy [48], mainly focusing on extracting common features in the classification tasks. We record the experimental results when $\alpha = 1, \beta = 1$, and $\lambda = 10$ in this approach.

4. Experimental results and analysis

In this section, we first verify the problem-solving effectiveness of the hierarchical structure constructed by multi-granularity clustering. Secondly, we show the performance of the feature subsets selected by HCFS. Finally, we evaluate the validity of our algorithm on a synthetic dataset. In all experiments, we employ a top-down Support Vector Machine classifier to test the effectiveness of our method. For the SVM classifier, 10-fold cross-validation is performed using a linear kernel and $c = 1$.

4.1. Comparison of clustering hierarchy and existing semantic hierarchy

We make a performance comparison between the hierarchical structure already existing in the data and that established from bottom-to-top by clustering. To better facilitate the comparison, the hierarchical structure obtained by clustering and granulation is roughly similar to the existing semantic hierarchy. The number of granularities in the second coarser-grained layer (GMSCL) is the same as the number of parent classes of the labels of the semantic hierarchy. The structural parameters of the clustering hierarchy are listed in Table 3. The "CH Node" column is the total number of nodes in the clustering hierarchical tree structure, including the real class nodes at the leaf level and their ancestor class nodes.

The hierarchical F1-measure results of selecting the different numbers of features from different datasets are shown in Fig. 6. For the two protein datasets, using 10% of the selected features achieves the same hierarchical F1-measure as using all features. For the image datasets, using 20% of the features achieves the same hierarchical F1-measure as the baseline algorithm. Therefore, we used 10% of the features for the two protein datasets and 20% of the features for the image datasets.

Table 4 compares the performance of using the clustering hierarchy and the existing semantic hierarchy on the different datasets. Feature selection performance is stable. It can be seen that the performance of the hierarchical structure obtained from data granulation is better than that of the existing hierarchical structure. On different image datasets, the F1-measure of the clustering hierarchy is, on average, approximately 3% higher than that of the existing hierarchy.

The performance of the clustering hierarchy with protein datasets is slightly lower than that of existing sematic hierarchies due to the sparse feature distribution of those datasets. Hence, the granular hierarchical structure of clustering the data from the bottom-up is suitable for problem-solving.

To further verify that the HCFS method solves the semantic gap problem in the semantic hierarchy, we artificially synthesized a toy dataset for verification testing, which consists of four categories: panda, whale, shark, and salmon. The numbers of image samples in each category are the same. The hierarchical structure established by semantic knowledge is shown in Fig. 1. The *panda* and *whale* categories are both mammals. The *shark* and *salmon* categories are both fish. We extract features from the original picture through the deep-learning pre-trained network VGG19 and execute the HCFS method. Fig. 7 gives an intuitive understanding of how to solve the semantic gap problem using multi-granularity clustering based on sample similarity. The *whale*, *shark*, and *salmon* categories are clustered into one coarse-grained category when there are two coarse-grained classes. The *Salmon* and *shark* categories are clustered into a single category when there are three coarse-grained classes. Using visual features instead of semantic features for clustering can effectively alleviate the semantic gap.

4.2. Analysis of the contribution of component **P** in HCFS

In the next few experiments, the number of granular clusters is set to three to compromise between the efficiency and effectiveness of the HCFS feature selection method. We consider the component **P** in the model coefficient matrix and regularize it with $\ell_{1,2}$-norms in the proposed method's model. To determine the importance of the different aspects of the gran-

Table 3
Clustering hierarchy description.

Dataset	Features	Fine-grained labels	Coarse-grained labels	CH Nodes	Depth
DD	473	27	4	32	3
F194	473	194	7	202	3
CLEF	80	80	17	88	3
Car196	4096	196	9	206	3
Cifar100	4096	100	20	121	3
SUN324	4096	324	15	340	3

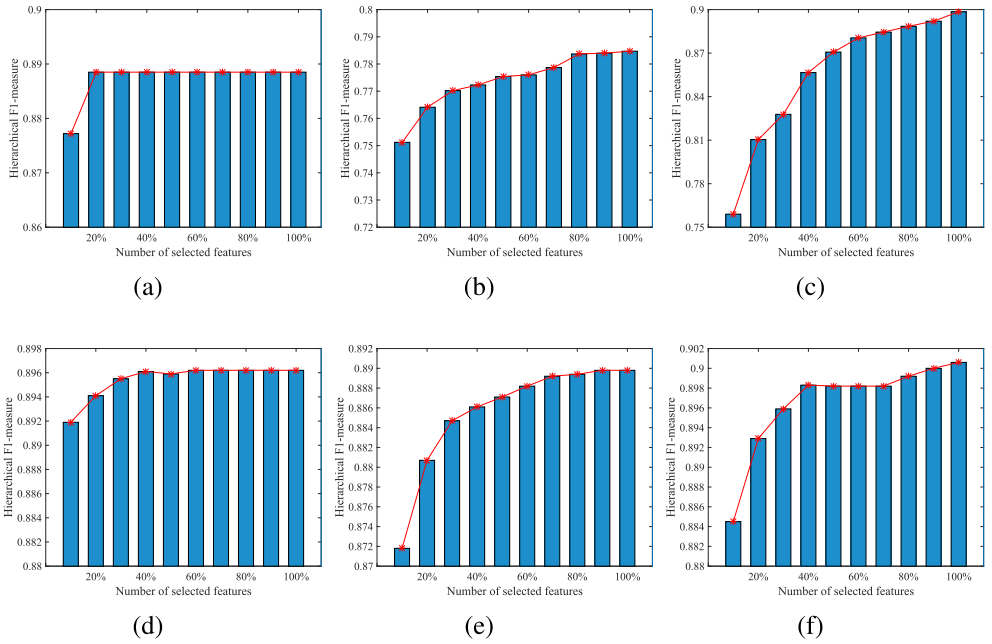

Fig. 6. Hierarchical F1-measure results using different numbers of features with datasets: (a) DD; (b) F194; (c) CLEF; (d) Car196; (e) Cifar100; (f) SUN324.

Table 4
Performance comparison of clustering hierarchy (CH) and existing hierarchy (EH).

Dataset	Structure	Selected features	GMSCL	F1-measure	*acc*
DD	EH	47 (10%)	4	0.8535	0.6795
	CH	47 (10%)	4	0.8507	0.6895
F194	EH	47 (10%)	7	0.6864	0.3113
	CH	47 (10%)	7	0.6829	0.3275
CLEF	EH	16 (20%)	17	0.7095	0.4590
	CH	16 (20%)	17	0.7274	0.5134
Car196	EH	819 (20%)	9	0.8367	0.6859
	CH	819 (20%)	9	0.8877	0.6866
Cifar100	EH	819 (20%)	20	0.8161	0.6616
	CH	819 (20%)	20	0.8513	0.6613
SUN324	EH	819 (20%)	15	0.8709	0.6827
	CH	819 (20%)	15	0.8876	0.6829

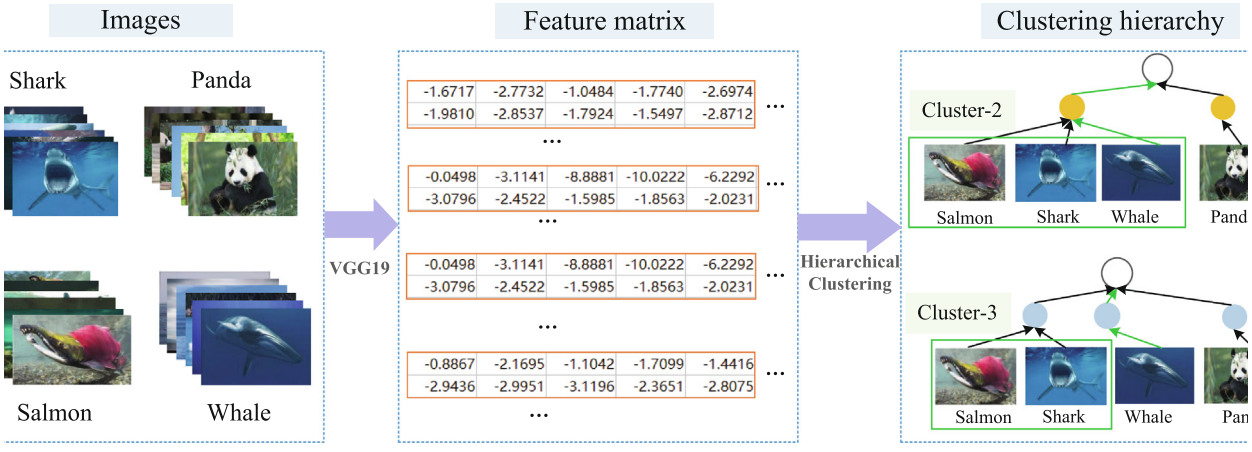

Fig. 7. An example of an intuitive understanding of solving the semantic gap problem with the HCFS.

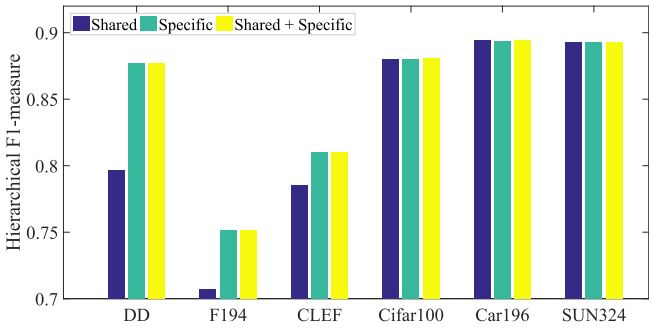

Fig. 8. Contributions of component **P** to the results on six datasets.

ularity features in the model, we use three models for learning performance, having 1) granularity-shared features, 2) granularity-specific features, and 3) shared and specific features of the granularity feature space.

The experimental results of the three type methods with different datasets are shown in Fig. 8. It seems that the model with granularity-specific features achieves similar results to the model with both shared and specific features. These results show that we can obtain better performance than when using shared features by paying attention to the specific granularity features in the classification settings. The experimental results for the two protein datasets and the medical dataset show the importance of considering the special features of granularity. The results show that the characteristics due to the complexity of these datasets' granular structure are essential. In addition, the granularity space in the hierarchy contains a large number of shared features while retaining a small number of granularity features for high-dimensional data (e.g., the SUN324 dataset). Therefore, focusing on the specific features of granularity, especially with complex granularity structure and high-dimensional features, can improve the generalization performance.

4.3. Performance comparison of different methods

We now compare HCFS with the baseline feature selection methods in the evaluation of hierarchical measures. We set the experiment's feature proportions to 10% for the two protein datasets and 20% for the four image datasets.

In terms of the hierarchical F1-measure results of the six datasets in Fig. 9, we observe the following:

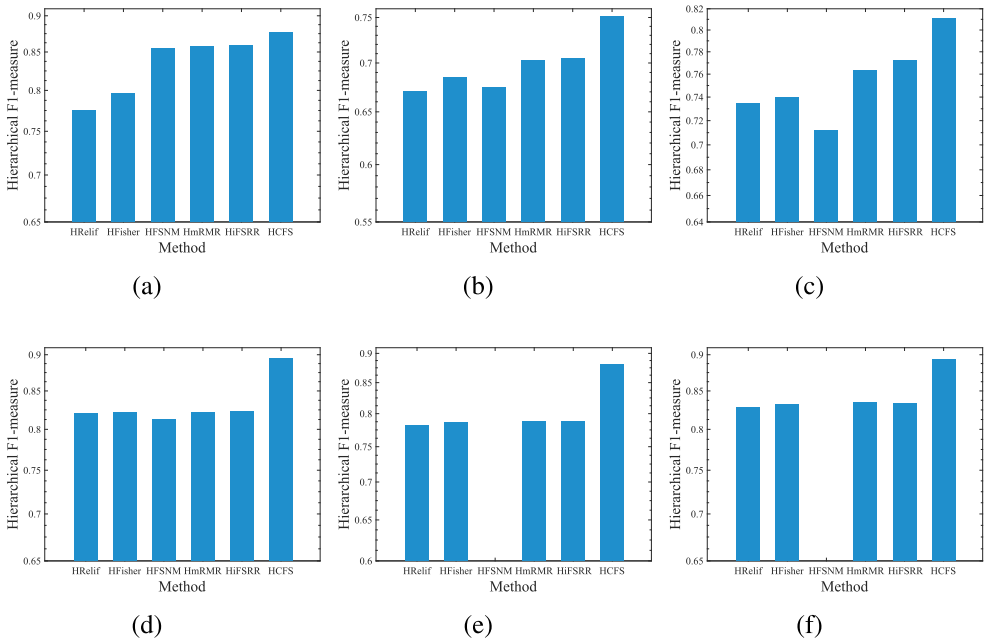

Fig. 9. Hierarchical F1-measure results for the hierarchical feature selection methods with different datasets: (a) DD; (b) F194; (c) CLEF; (d) Car196; (e) Cifar100; (f) SUN324.

(1) The results of the two protein datasets are shown in Fig. 9(a) and (b). The two datasets have the same feature sequence, and their main difference between them is in the classes. The hierarchical F1-measure of HCFS is 87.72% on the DD dataset, which is 1.82% better than on the second-placed HiFSRR method, at 85.9%. The hierarchical F1-measure on the F194 dataset is 4.60% better than the second-placed HiFSRR. The advantages of the F194 dataset prove the effectiveness of the selected features on multiple categories.
(2) The results in Fig. 9(c) and (d) show that HCFS has a definite advantage over the other five methods on the CLEF dataset and Car196 datasets, respectively. The other five algorithms have similar performance on the Car196 dataset. However, the hierarchical F1-measure of HCFS is 89.41%, which is 7.06% higher than that of the second-placed HiFSRR. These results further show that HCFS selects the features that can distinguish fine-grained categories.
(3) Fig. 9(e) and (f) show the hierarchical F1-measure results on the Cifar100 and SUN324 datasets. The HFSNM method produced no results with these two datasets due to insufficient memory. The best feature subset of HCFS is obtained by selecting shared and specific features in each layer's granular feature space. Therefore, HCFS has a very stable advantage with these two large image datasets.

The TIE evaluation criterion indicates the degree of prediction error in the hierarchy. The results in Table 5 show that the degree of error in the HCFS prediction labels is relatively low compared with the other feature selection methods. Finally, we conducted the nonparametric Friedman test [15] followed by the Bonferroni-Dunn test [10] to explore the statistical significance of the compared methods. These tests are generally accepted for statistically comparing multiple methods used on many datasets. Given k comparison methods and N datasets, $R_j = \frac{1}{N}\sum_{i=1}^{N} r_i^j$ is the average rank of the j-th method on all datasets, where r_i^j is the rank of j-th method on the i-th dataset. To rank all the methods, the best-performing method is designated as 1, the next method as 2, and so on. The average rank of each method on all datasets is then calculated. Table 5 shows the TIE ranks of the different methods. Under the null-hypothesis (i.e., the ranks of all hierarchical feature selection methods are equivalent), the Friedman statistics can be defined as:

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2}, \tag{8}$$

where

$$\chi_F^2 = \frac{12N}{k(k-1)}\left(\sum_{i=1}^{k} R_j^2 - \frac{k(k+1)^2}{4}\right) \tag{9}$$

and F_F is distributed according to the F-distribution with $(k-1)$ and $(k-1)(N-1)$ degrees of freedom. We calculate the value F_F = 52.5658 by Eq. (8). In the case of the six methods and six datasets, $F(6-1),(6-1)\times(6-1)) = F(5,25)$ for α = 0.05 is 2.6030, so we reject the null-hypothesis. Therefore, these six methods have significant differences in performance.

We next employ the Bonferroni-Dunn test to compare the performance of the methods. The performance difference between HCFS and a comparison method is significant when the distance of the average rank exceeds the *critical difference* (CD):

$$CD_\alpha = q_\alpha\sqrt{\frac{k(k+1)}{6N}}. \tag{10}$$

For the Bonferroni-Dunn test, we have $q_\alpha = 2.576$ at significance level $\alpha = 0.05$ in Table 5 in [4], and thus CD = 2.7824 ($k = 6, N = 6$). Fig. 10 depicts the statistical results of the Bonferroni-Dunn test on six datasets at $\alpha = 0.05$. The TIE measure of HCFS is statistically superior to those of HFSNM, HRelif, and HFisher. However, there is no consistent evidence that the performance of HCFS is significantly different from those of HmRMR and HiFSRR.

Table 5
Normalized TIE results of different feature selection methods on different datasets.

Dataset	HRelif	HFisher	HFSNM	HmRMR	HiFSRR	HCFS
DD	0.1352(6)	0.1226(5)	0.0873(4)	0.0853(2)	0.0860(3)	0.0017(1)
F194	0.1977(6)	0.1894(4)	0.1951(5)	0.1785(3)	0.1769(2)	0.1493(1)
CLEF	0.2037(5)	0.2011(4)	0.2194(6)	0.1825(3)	0.1768(2)	0.1137(1)
Car196	0.1081(5)	0.1072(4)	0.1132(6)	0.1072(3)	0.1059(2)	0.0635(1)
Cifar100	0.1309(5)	0.1285(4)	–(6)	0.1273(2)	0.1274(3)	0.0716(1)
SUN324	0.1379(5)	0.1341(4)	–(6)	0.1322(2)	0.1333(3)	0.0642(1)
Ave. Rank	5.3	4.2	5.5	2.5	2.5	1.0

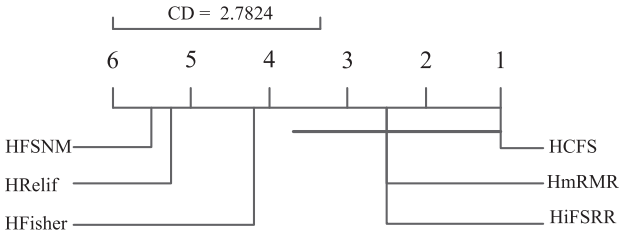

Fig. 10. Comparison of HCFS against other comparison methods with the Bonferroni-Dunn test on TIE.

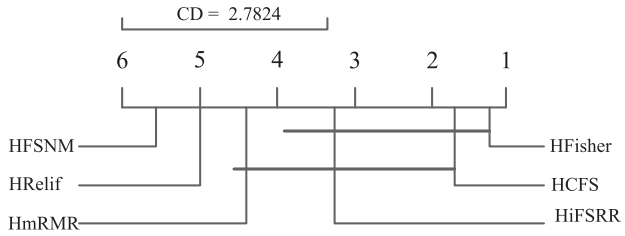

Fig. 11. Comparison of HCFS against other comparison methods with the Bonferroni-Dunn test on running time.

4.4. Efficiency comparison of different methods

The hierarchical evaluation standard for running time is to see how long it takes to select discriminative features in the hierarchical structure. A comparison of the proposed method and other hierarchical feature selection methods is presented in Table 6.

The proposed HCFS method is generally inferior to HFisher, because HFisher selects a suitable set of feature subsets from all the features. In contrast, HCFS selects different feature subsets on different granularity layers, from coarse- to fine-grained, for hierarchical classification. In contrast, HCFS selects different feature subsets of different granularity layers, from coarse- to fine-grained, for hierarchical classification. Although the efficiency of our method is not always the best, its performance on different datasets is quite robust while the performance of some others is very unpredictable and highly variable.

We perform the Friedman test to further explore whether there is a significant difference in the running times of the six hierarchical feature selection methods. In the null hypothesis, $F_F = 32.3932$ is calculated according to the results in Table 6. Therefore, we reject the null hypothesis and execute the Bonferroni-Dunn test. As shown in Fig. 11, the running time of HCFS is statistically inferior to that of HFisher because it needs time to construct a hierarchical structure with feature correlations. There is no consistent evidence of a statistical difference in running time between HCFS and HFisher.

4.5. Performance evaluation on the synthetic dataset

Finally, we evaluate the feature selection effect of HCFS on synthetic data. In our experiment, the DD dataset was noise-processed by adding a randomly generated matrix, which was generated by random numbers from a normal distribution or a matrix function.

Table 7 compares the F1-measure results of different hierarchical feature selection methods on the synthetic dataset. It shows that, compared with most methods, HCFS achieves competitive performance when noise is present in the dataset.

Table 6
Running time of different feature selection methods on different datasets (s).

Dataset	HRelif	HFisher	HFSNM	HmRMR	HiFSRR	HCFS
DD	55.3(5)	0.4(1)	80.1(6)	26.4(4)	1.8(3)	0.7(2)
F194	237.6(5)	2.2(2)	827.2(6)	62.3(4)	5.4(3)	1.7(1)
CLEF	47.7(5)	0.2(1)	1221.8(6)	3.7(4)	0.4(3)	0.3(2)
Car196	2378.5(5)	19.8(1)	411.1(3)	6638.2(6)	1950.6(4)	46.6(2)
Cifar100	73323.1(5)	18.4(1)	–(6)	24593.8(4)	8868.4(3)	395.9(2)
SUN324	82694.9(5)	40.8(1)	–(6)	31494.3(4)	2212.2(3)	327.9(2)
Ave. Rank	5.0	1.2	5.5	4.3	3.2	1.8

Table 7
F1-measure results of different feature selection methods on the synthetic dataset.

Features	Percentage	HRelif	HFisher	HFSNM	HmRMR	HiFSRR	EH	HCFS
47	10%	0.4590(7)	0.5195(3)	0.5058(6)	0.5185(4)	0.5444(2)	0.5179(5)	0.5828(1)
95	20%	0.4854(7)	0.5162(4)	0.4866(6)	0.5229(2)	0.5167(3)	0.5135(5)	0.5884(1)
142	30%	0.4816(7)	0.5130(3)	0.4865(6)	0.5003(5)	0.5085(4)	0.5223(2)	0.5813(1)
189	40%	0.4815(7)	0.5086(4)	0.4859(6)	0.5008(5)	0.5113(3)	0.5205(2)	0.5790(1)
237	50%	0.4773(7)	0.5185(2)	0.4970(6)	0.5030(4)	0.5151(3)	0.5014(5)	0.5569(1)
473	100%	0.5092(4)	0.5092(4)	0.5092(4)	0.5092(4)	0.5092(4)	0.4926(7)	0.5449(1)
Ave. Rank		6.5	3.3	5.7	4.0	3.2	4.3	1.0

However, some other hierarchical feature selection methods are slightly less effective after selecting smaller feature subsets; for example, with the HRelif and HmRMR hierarchical feature selection methods. According to the results of the Friedman test, HCFS is approximately 5.4% better, on average, than the second-ranked HiFSRR hierarchical feature method. The proposed method is more robust and effective than other methods, especially when selecting a small number of features.

5. Conclusions and future work

In this paper, we proposed a hierarchical feature selection method based on a hierarchical structure obtained by clustering data from fine to coarse granularities. The hierarchical structure obtained according to feature similarity effectively solves the semantic gap problem in the semantic hierarchy. This method effectively reduces redundancy because it shares the granularity layer's common features and captures the granularity-specific features. The experimental results demonstrate that HCFS can select the optimal feature subset and achieves better performance than other hierarchical feature selection methods in real-world problems. In future work, we will focus on building models for different granularity layers to cope with their different distributions. Also, the number of layers to be considered in hierarchical clustering should be optimized.

CRediT authorship contribution statement

Shunxin Guo: Methodology, Software, Writing - original draft. **Hong Zhao:** Conceptualization, Data curation, Supervision, Writing - review & editing. **Wenyuan Yang:** Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No. 61703196, the Natural Science Foundation of Fujian Province under Grant No. 2018J01549, and the President's Fund of Minnan Normal University under Grant No. KJ19021.

References

[1] R. Aliev, W. Pedrycz, B. Guirimov, O. Huseynov, Clustering method for production of z-number based if-then rules, Inf. Sci. 520 (2020) 155–176.
[2] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, Siam J. Imag. Sci. 2 (1) (2009) 183–202.
[3] M.M. Cheng, Y. Liu, Q. Hou, J. Bian, Z. Tu, HFS: hierarchical feature selection for efficient image segmentation, in: European Conference on Computer Vision, 2016, pp. 1–16.
[4] J. Demsar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.
[5] J. Deng, W. Dong, R. Socher, L.J. Li, F.F. Li, ImageNet: a large-scale hierarchical image database, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
[6] I. Dimitrovski, D. Kocev, S. Loskovska, S. Džeroski, Hierarchical annotation of medical images, Pattern Recogn. 44 (10–11) (2011) 2436–2449.
[7] C.H. Ding, I. Dubchak, Multi-class protein fold recognition using support vector machines and neural networks, Bioinformatics 17 (4) (2001) 349–358.
[8] H. Dong, T. Li, R. Ding, J. Sun, A novel hybrid genetic algorithm with granular information for feature selection and optimization, Appl. Soft Comput. 65 (2018) 33–46.
[9] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, Wiley, 2001.
[10] O.J. Dunn, Multiple comparisons among means, J. Am. Stat. Assoc. 56 (293) (1961) 52–64.
[11] F. Esposito, D. Malerba, V. Tamma, H.H. Bock, Classical resemblance measures, Anal. Symbol. Data (2000) 139–152..
[12] A. Esuli, T. Fagni, F. Sebastiani, Boosting multi-label hierarchical text categorization, Inf. Retrieval 11 (4) (2008) 287–313.
[13] C. Freeman, D. Kulić, O. Basir, Joint feature selection and hierarchical classifier design, in: IEEE International Conference on Systems, 2011, pp. 1728–1734.
[14] C. Freeman, D. Kulić, O. Basir, Feature-selected tree-based classification, IEEE Trans. Cybern. 43 (6) (2013) 1990–2004.
[15] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, Ann. Math. Stat. 11 (1) (1940) 86–92.
[16] P. Gong, J. Ye, C. Zhang, Robust multi-task feature learning, Knowl. Discovery Data Min. 2012 (2012), 895–903.

[17] S. Gopal, Y. Yang, Hierarchical Bayesian inference and recursive regularization for large-scale classification, ACM Trans. Knowl. Discovery Data 9 (3) (2015) 1–23.
[18] L. Grimaudo, M. Mellia, E. Baralis, Hierarchical learning for fine grained internet traffic classification, in: International Wireless Communications and Mobile Computing Conference, 2012, pp. 1–7.
[19] K. Kira, L.A. Rendell, A practical approach to feature selection, in: International Workshop on Machine Learning, 1992, pp. 249–256..
[20] A. Kosmopoulos, I. Partalas, E. Gaussier, G. Paliouras, I. Androutsopoulos, Evaluation measures for hierarchical classification: a unified view and novel approaches, Data Min. Knowl. Disc. 29 (3) (2015) 820–865.
[21] J. Krause, M. Stark, J. Deng, F.F. Li, 3D object representations for fine-grained categorization, in: International IEEE Workshop on 3D Representation and Recognition, 2013, pp. 554–561.
[22] A. Krizhevsky, Learning multiple layers of features from tiny images, Tech Report, Department of Computer Science, University of Toronto, 2009..
[23] J. Liang, F. Wang, C. Dang, Y. Qian, An efficient rough feature selection algorithm with a multi-granulation view, Int. J. Approx. Reason. 53 (6) (2012) 912–926.
[24] S. Liao, Q. Zhu, Y. Qian, G. Lin, Multi-granularity feature selection on cost-sensitive data with measurement errors and variable costs, Knowl. Based Syst. 158 (2018) 25–42.
[25] J. Liu, S. Ji, J. Ye, SLEP: sparse learning with efficient projections, 2013, pp. 1–41..
[26] J. Ma, T.W.S. Chow, Topic-based algorithm for multilabel learning with missing labels, IEEE Trans. Neural Networks Learn. Syst. PP (99) (2018) 1–15..
[27] R. Mittelman, M. Sun, B. Kuipers, S. Savarese, A Bayesian generative model for learning semantic hierarchies, Front. Psychol. 5 (5) (2014) 1–9.
[28] A.G. Murzin, S.E. Brenner, T.J.P. Hubbard, C. Chothia, SCOP: a structural classification of proteins database for the investigation of sequences and structures, J. Mol. Biol. 247 (4) (1995) 536–540.
[29] F.P. Nie, H. Huang, X. Cai, C.H. Ding, Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization, in: Advances in Neural Information Processing Systems, 2010, pp. 1813–1821..
[30] W. Pedrycz, Granular computing as a framework of system modeling, J. Control Autom. Electr. Syst. 24 (1–2) (1997) 81–86.
[31] W. Pedrycz, An introduction to computing with fuzzy sets – analysis, design, and applications, Intell. Syst. Ref. Lib. 190 (2021) 1–283.
[32] P. Ruvolo, I. Fasel, J.R. Movellan, A learning approach to hierarchical feature selection and aggregation for audio classification, Pattern Recogn. Lett. 31 (12) (2010) 1535–1542.
[33] C.N. Silla, A.A. Freitas, A survey of hierarchical classification across different application domains, Data Min. Knowl. Disc. 22 (1) (2011) 31–72.
[34] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Comput. Sci. (2014) 1–12.
[35] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, R.C. Jain, Content-based image retrieval at the end of the early years, IEEE Trans. Pattern Anal. Mach. Intell. 22 (12) (2000) 1349–1380.
[36] C. Wan, A.A. Freitas, An empirical evaluation of hierarchical feature selection methods for classification in bioinformatics datasets with gene ontology-based features, Artif. Intell. Rev. 50 (2) (2018) 201–240.
[37] H. Wang, P. Zhang, X. Zhu, I. Tsang, L. Chen, C. Zhang, X. Wu, Incremental subgraph feature selection for graph classification, IEEE Trans. Knowl. Data Eng. 29 (1) (2017) 128–142.
[38] L. Wang, Z. Han, W. Pedrycz, J. Zhao, W. Wang, A granular computing-based hybrid hierarchical method for construction of long-term prediction intervals for gaseous system of steel industry, IEEE Access 8 (2020) 63538–63550.
[39] S. Wang, F. Nie, X. Chang, L. Yao, X. Li, Q.Z. Sheng, Unsupervised feature analysis with class margin optimization, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2015, pp. 1–17.
[40] L. Wei, M. Liao, X. Gao, Q. Zou, An improved protein structural classes prediction method by incorporating both sequence and structure information, IEEE Trans. Nanobioence 14 (4) (2015) 339–349.
[41] S. Xia, Z. Xiong, Y. Luo, Wei-xu, G. Zhang, Effectiveness of the euclidean distance in high dimensional spaces, Optik - Int. J. Light Electron Opt. 126 (24) (2015) 5614–5619..
[42] S. Xia, Z. Zhang, W. Li, G. Wang, E. Giem, Z. Chen, GBNRS: a novel rough set algorithm for fast adaptive attribute reduction in classification, IEEE Trans. Knowl. Data Eng. PP (99) (2020) 1–12..
[43] C. Xiao, F. Nie, H. Huang, C. Ding, Multi-class $\ell_{1,2}$-norm support vector machine, in: IEEE International Conference on Data Mining, 2011, pp. 91–100.
[44] J. Xiao, K.A. Ehinger, J. Hays, A. Torralba, A. Oliva, Sun database: exploring a large collection of scene categories, Int. J. Comput. Vision 119 (1) (2016) 3–22.
[45] Z.B. Xu, Z. Hai, W. Yao, X.Y. Chang, L. Yong, L1/2 regularization, Sci. China Inf. Sci. 53 (6) (2010) 1159–1169.
[46] Y. Yang, H.T. Shen, Z. Ma, Z. Huang, X. Zhou, $\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning, International Joint Conference on Artificial Intelligence 22 (2011) 1589–1594.
[47] H. Zhang, Z.J. Zha, Y. Yang, S. Yan, Y. Gao, T.S. Chua, Attribute-augmented semantic hierarchy: towards a unified framework for content-based image retrieval, ACM Trans. Multimedia Comput. Commun. Appl. 11 (1s) (2014) 1–21.
[48] H. Zhao, Q. Hu, P. Zhu, P. Wang, Hierarchical feature selection with recursive regularization, in: International Joint Conference on Artificial Intelligence, 2017, pp. 3483–3489.
[49] Q. Zou, Y. Ju, D. Li, Protein folds prediction with hierarchical structured svm, Curr. Proteom. 13 (2) (2016) 79–85.