A Hybrid Simplified Swarm Optimization Method for Imbalanced Data Feature Selection

Wei Chang Yeh, Yu Tai Yang and Chyh Ming Lai
National Tsing Hua University, Taiwan

Abstract
In recent years, feature selection has become an important field in data mining and is being used heavily in numerous areas. The purpose of feature selection is to search for an optimal subset of features from existing data to maximize the accuracy. However, there are still only a few studies investigating the impact of data imbalance - the existence of underrepresented categories of data - on feature selection problem. The aim of this study is therefore to provide a feature selection method for increasing classifying high-dimensional imbalanced data accuracy. In this study, we propose a hybrid method which can spot a better optimal features subset. In the proposed method, information gain as a filter selects the most informative features from the original dataset. The imbalance of the dataset with selected features is justified by using Synthetic minority over-sampling technique. Simplified swarm optimization is then implemented as feature search engine to guide the search for an optimal feature subset. Finally, support vector machine serves as a classifier to evaluate the performance of the proposed method. To evaluate the performance of proposed algorithm, we apply our algorithm in four benchmark datasets and compare the results with existing algorithm. The results show that our algorithm has a better performance than its competitor.

Keywords: Data Mining; Feature Selection; Imbalanced Data; Soft Computing; Simplified Swarm Optimization; Support Vector Machine

1. Introduction
Feature selection has become an important field in data mining, and used frequently in numerous disciplines including text categorization, image retrieval, and genomic analysis (Liu & Yu, 2005). Feature selection is the process of selecting a subset of features (variables) from data. The reduction of features can increase the accuracy of machine learning result and reduce the time of
building model. In addition, feature selection can reduce the risk of over fitting, which is more common in high-dimensional data sets (Maldonado, Weber, & Famili, 2014).

The feature selection has three kinds of algorithms: filter, wrapper, and embedded method. Filter method uses pre-defined metric to evaluate the goodness of features before using the classifier. The common filter method includes $\chi^2$ static, Information Gain (Y. Yang & Pedersen, 1997), and Relief (Kira & Rendell, 1992). Filter method has lower computation time but has poorer performance than the other two methods.

Wrapper method generates subsets and judges the subsets by the performance of implying classifier. The embedded method finds the subset during the procedure of classifier. The wrapper and embedded method usually have better performance but are more computational incentive than the filter method (Maldonado et al., 2014).

For the wrapper method, many methods have been proposed to evaluate a better optimal features subset. In recent years, soft computing algorithm has been implied as searching for wrapper method. The randomness of these stochastic algorithm can reduce the sensitivity to the dataset (Al-Ani, Alsukker, & Khushaba, 2013). Commonly used methods such as particle swarm optimization (PSO) (Chuang, Chang, Tu, & Yang, 2008), genetic algorithm (GA) (C.-H. Yang, Chuang, & Yang, 2010), artificial bee colony (ABC) (Schiezaro & Pedrini, 2013), and simulated annealing (SA) (Lin, Lee, Chen, & Tseng, 2008). Among all these method, PSO and GA are the most commonly used methods, as well as PSO, which has a better algorithmic efficiency and is effortless to implement than GA (Al-Obeidat, Belacel, Carretero, & Mahanti, 2011). However, PSO also has its disadvantages. It may drop to the local optimal and is inadequate for discrete problems (Liang, Qin, Suganthan, & Baskar, 2006). To overcome these disadvantages, Yeh proposed the simplified swarm optimization (SSO) (Yeh, 2009). It modified the update process of variable in PSO and has few parameters to tune. The SSO also performs well in feature selection problems (Yeh, Chang, & Chiu, 2011). Thus, in this study we use SSO as the wrapper method in our hybrid algorithm.

The feature selection technique can also be implied in imbalanced data, which is also a crucial issue in recent years (He & Garcia, 2009). Imbalanced data is that one of the classes of data set that has relatively few instances, called minority class, is underrepresented. In many cases, the minority class is the major target of the data, like cancer diagnosis (Mazurowski et al., 2008), the patient with malignant tumor is way less than the patient with benign tumor. Other applications, such as fraud detection (Anil Kumar & Ravi, 2008), helicopter fault monitoring (Japkowicz, Myers, & Gluck, 1995), have a valuable minority class as well. The issue of imbalanced data is that the normal classification method may misjudge the minority class. For example, if we have 95 instances majority
class, the class holds the most instances and 5 instances of minority class in the dataset. If the classifier identifies all class as the majority class, it will have 95% accuracy, which is normally considered good. However, this result cannot reflect that the minority class is 100% misclassified, and also that the minority class is more important than the majority class. Therefore, handling the biased dataset and classifying minority class as accurately as possible are the main challenges in class imbalanced problem.

The class imbalanced feature selection problem has been studied in many fields (Villar, Fernández, Carrasco, & Herrera, 2012). However, there are still relatively few studies investigating the impact of data imbalance on high-dimensional feature selection problem (Maldonado et al., 2014). Hence, we proposed a hybrid algorithm for high-dimensional class imbalanced feature selection problem. The algorithm combines IG and simplified swarm optimization (SSO) (Yeh, 2013), a soft computing algorithm, as filter and wrapper method to explore the optimal subset. The re-sampling technique is implied to reduce class imbalanced problem. We use the support vector machine (SVM) as the classifier.

The next parts of the paper are organized as follows: in section 2 we provide a brief review of related work associate to our study. In section 3 we describe the algorithm proposed. The experiment result is showed in section 4, and a conclusion is given in section 5.

2. Related Work 1000-2000

2.1 Imbalanced Data

To overcome the underrepresented minority class, we can change the weight of different classes. For example, let the weight of minority class be $W_m$, weight of majority class be $W_s$, and $W_m > W_s$. This makes the classifier more sensitive to the misclassification of minority class. Another way is the re-sampling technique, including oversampling and under sampling. The former duplicates the minority class while the latter deletes the majority class from dataset, which may cause a loss of information (Van Hulse, Khoshgoftaar, Napolitano, & Wald, 2009).

The above method can help balance the data set and increase the accuracy of classifying class imbalanced dataset. However, due to the monotonicity of minority class and no new instances being added, these method still may cause over fitting (Van Hulse et al., 2009). Thus, the synthetic minority over-sampling technique (SMOTE) (Chawla, Bowyer, Hall, & Kegelmeyer, 2002) is proposed to overcome the above problem.

SMOTE can generate new instances from existed data, which makes the decision regions less specific and prevents them from over fitting (Han, Wang, & Mao, 2005). The procedure of
SMOTE is described as follows: First each minority instances \( k \) nearest the minority neighborhood (\( k \) usually set to 5) is found. Then one of \( k \) neighborhood is randomly picked, and this generates new instances randomly between them. This process repeats depending on the portion we want for the new artificial instance. In most cases, the portion is set to 200\% (Barua, Islam, Yao, & Murase, 2014).

For the class imbalanced problem, we usually discuss the binary (two-class) problem. The minority class is usually denoted as positive, while the majority class is denoted as negative. The confusion matrix is shown in Figure 1 as described in section 1, the performance of imbalanced data cannot be easily presented by accuracy:

\[
\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP}
\]  
(1)

Here, TP and TN denote the number of true positive and true negative instances, while FP and FN denote the number of false positive and false negative instances. Therefore, numerous evaluation metrics are introduced. In this study, the geometric mean (g-mean) is used as the performance measurement metric:

\[
\text{G-mean} = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}}
\]  
(2)

Other common metric includes F-measure (also called F-score) and Area under curve (AUC) (Sokolova, Japkowicz, & Szpakowicz, 2006).

**Figure 1: Confusion matrix for binary class problem**

2.2 Information Gain

Information Gain is a filter feature selection method. It scores the informational entropy
of features and determines the importance of these features. Informational entropy is theoretically the number of bits of data it would take to encode a given piece of information (Y. Yang & Pedersen, 1997). The more space a piece of information takes to encode, the more entropy it has (Van Hulse et al., 2009). An example can be used to demonstrate the above description: a sequential data can be easily transferred to a smaller archive file using compression algorithm; while a totally random data, which has maximum entropy, cannot be compressed.

For classification, the information of instances belong to which classes is the data we want to describe/compressed. If all instances belong to one class, the compress rate is huge (all instances are in the first class). However, if all instances are randomly separate to each classes, the space needed to record the information is huge, which means the entropy of this situation is high. The equation calculating entropy is described as follows:

$$\text{H}(T) = - \sum_{i=1}^{q} \left( \frac{n_i}{n} \right) \times \log \left( \frac{n_i}{n} \right)$$  \hspace{1cm} (3)

Here, training dataset T has $n=|T|$ instances and $q$ classes. The $i^{th}$ class has $n_i$ instances. Now the entropy related to each attribute is to be identified. For attribute $a$ which has $v$ distinct values, $a_j (j = 1,2,\ldots,v)$, the entropy is calculated by summing the entropy for each $a_j$:

$$\text{H}(T|a) = \sum_{j=1}^{v} \left( \frac{|a_j|}{n} \right) \times \text{H}(T|a = a_j)$$  \hspace{1cm} (4)

Here, $|a_j|$ denotes the number of instances has the same attribute $a$ value $a_j$. If attribute $a$ is highly related to the classes in which it belongs, the entropy of $H(T|a)$ will be low, otherwise the entropy will be close to $H(T)$. The reduction of entropy, the value measured information gain for each attribute (IG value), is shown in Eq. 5:

$$\text{IG}(a) = \text{H}(T) - \text{H}(T|a)$$  \hspace{1cm} (5)

The more the entropy decrease, the more significant feature the $x$ is for prediction (Van Hulse et al., 2009). The IG value we get in Eq. 5 is between 0 and 1. The features can be filtered by a pre-determined number of features to keep, or by setting a threshold value and reserve the features that has higher IG value.
2.3 Simplified Swarm Optimization (SSO)

Simplified swarm optimization is a soft computing algorithm proposed by Yeh (Yeh, 2009). It evolved from the PSO algorithm and is developed to overcome the drawback of PSO in discrete problem. SSO has been implied in many fields, including network intrusion detection (Chung & Wahid, 2012) and disassemble sequencing problem (Yeh, 2012). The main idea of SSO is to generate a set of solutions and randomly update them by each solution’s best historical solution \(\text{pbest} \) and global best solution \(\text{gbest} \).

Let \(X_i^t = \{x_{i1}^t, x_{i2}^t, ..., x_{im}^t\} \) denotes the \(i\)th solution in \(t\)th iteration, where each solution has \(m\) variables and \(x_{ij}^t\) is the \(j\)th variable of \(i\)th solution in \(t\)th iteration. The \(x_{ij}^t\) is updated by generating a random variable \(\rho_{ij}^t \in [0,1]\), and then updating the variable:

\[
\begin{align*}
    x_{ij}^t &= \begin{cases} 
    x_{ij}^{t-1}, & \text{if } \rho_{ij}^t \in [0, c_w) \\
    p_{ij}^{t-1}, & \text{if } \rho_{ij}^t \in [c_w, c_p) \\
    g_j^{t-1}, & \text{if } \rho_{ij}^t \in [c_p, c_g) \\
    x_{new}, & \text{if } \rho_{ij}^t \in [c_g, 1] 
    \end{cases} 
\end{align*}
\]

(6)

Here, \(c_w, c_p, c_g\) are the pre-defined parameters for update that \(0 < c_w < c_p < c_g < 1\).

The \(p_{ij}^{t-1}\) and \(g_j^{t-1}\) denote the \(\text{pbest}\) of solution \(i\) and \(\text{gbest} \) of variable \(j\) in \((t-1)\)th iteration. The \(x_{new} \) is a new random value in the feasible field of variable \(j\).

The procedure of SSO is as follows: First a set of solutions is randomly generated and the fitness function of each solution is calculated. The \(\text{pbest}\) is initially set as these solutions and \(\text{gbest}\) is the solution of \(\text{pbest}\) that has the best fitness value. For each iteration the entire variable in each solution is updated by Eq. 6. Then the fitness function of updated solutions is calculated. If the fitness value of solution is higher than \(\text{pbest}\), the \(\text{pbest}\) is replaced by present solution. If the new \(\text{pbest}\) has higher fitness value than \(\text{gbest}\), the \(\text{gbest}\) is replaced by \(\text{pbest}\). The procedure is repeated until the threshold or the maximum iterations are reached.

2.4 Support Vector Machine (SVM)

Support vector machine is a supervised learning method and can be used as classifier in machine learning. SVM determines an optimal hyperplane or a set of hyperplanes \(f(A) = w^T \times A + b\) that separates different classes as widely as possible. In this study, only the binary linear
SVM is discussed.

For binary problem, \( A_i \in \mathbb{R}^m \) is denoted as the vector of attributes of instance \( i (i = 1, ..., n) \) and \( y_i \) as the class label of instance \( i \) where \( y_i = \{-1, 1\}, i = 1, ..., n \). The goal is to find the optimal hyperplane that separate different classes. The formulation is stated as follows:

\[
\begin{align*}
\min_{w, b, \xi} & \quad \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \\
\text{s.t.} & \quad y_i \times \left( w^T \times A_i + b \right) \geq 1 - \xi_i, \quad i = 1, ..., n
\end{align*}
\]

where \( \xi_i \) is the slake variable and \( C \) is the parameter for the penalty function of training error (Vapnik & Vapnik, 1998).

There is no wrapper method investigating imbalanced data problem in high-dimensional dataset using SVM. Villar et al. (Villar et al., 2012) proposed a genetic algorithm based feature selection method for class imbalanced problem but with low-dimensional datasets. Maldonado et al. (Maldonado et al., 2014) also proposed an embedded method using backward elimination. The method is for class imbalanced problem in high-dimensional dataset, which is same as the target in this paper, and got good prediction result with fewer features.

3. Proposed Method

3.1 Encoding

The encoding technique used in the SSO wrapper method is elementary. The main idea is whether it picks the feature or not. There are \( m \) variables, the number of total features, in each solution. Each variable is a binary number (0, 1) that represent the features. Fig. 2 represents an example of solution, which pick the 1, 2, 4 features and the dataset has 5 features in total.

Figure 2: The example of encoding

```
1 1 0 1 0
```

3.2 Fitness Function

The fitness function is calculated depending on the solution subset classification result using SVM. In feature selection problem, if there exists a set of subsets with similar accuracy, then the subset that contains fewer features is considered a better subset. As mentioned in section 2.1, the accuracy cannot represent the performance of imbalanced data completely. Therefore,
referring to the previous study (C.-H. Yang et al., 2010) the fitness function is set by g-mean:

\[ fit(X_i) = \delta \times g\text{-mean} + (1 - \delta) \times \frac{T - S}{T} \]  

(8)

Here, \( T \) and \( S \) denote the number of total features and selected features. The \( \delta \) is a pre-defined parameter between 0 and 1. The fitness function is a bi-objective function which considers both the number of selected features and the g-mean.

### 3.3 The Proposed IG-SSO

The proposed hybrid method for high-dimensional class imbalanced feature selection problem has three phases. First, the SMOTE is used for pre-processing the data. It can generate artificial minority class in order to balance the dataset. Next the information gain is implied to the dataset. The elimination of insignificant features can help reduce the computation complexity for the following stage. Finally the SSO is sued as the wrapper method and SVM as the classifier to investigate the optimal features subset.

In the second phase of information gain, the selection of threshold is a challenging work. If the threshold is too strict, the valuable features are eliminated. If the threshold is too loose, the uninformative features will increase the computational complexity for the next phase. Therefore, instead of setting a threshold, the top 60 features with the highest IG values are selected for the next step (C.-H. Yang et al., 2010). The procedure of the proposed method is presented as follows:

**Step 1**: Input the dataset \( D \). Set parameters \( c_w, c_p, c_g, rnk, pop, iter \).

**Step 2**: Use SMOTE to add artificial instances set \( SM \). \( D \leftarrow D \cup SM \)

**Step 3**: Calculate the IG value and hold the top \( rnk \) features. \( D \leftarrow D_{IG(rnk)} \)

**Step 4**: Generate \( X_i (i = 1, \ldots, rnk) \), let \( t = 1 \).

**Step 5**: Let \( i = 1 \).

**Step 6**: Let \( j = 1 \).

**Step 7**: Update \( x_{ij} \) by Eq. 6.

**Step 8**: If \( j < rnk \) then \( j = j + 1 \) and go to Step 7.

**Step 9**: Calculate \( fit(X_i) \) and update \( pbest \) and \( gbest \).

**Step 10**: If \( i < pop \) then \( i = i + 1 \) and go to Step 6.

**Step 11**: If \( t < iter \) then \( t = t + 1 \) and go to Step 5.
Step 12: Return the g-mean of gbest.

4. Experiment Result

To evaluate the performance of the proposed method (IG-SSO), four benchmark problems are tested, which have been used for feature selection problem (K. Yang, Cai, Li, & Lin, 2006), and the results are compared with the method proposed by Maldonado (Maldonado et al., 2014). They are a family of class imbalanced high-dimensional method using SVM. The performance the feature selection method is evaluated by the g-mean.

The information of benchmark datasets are shown in Table 1. Since the binary problem is being studied, these datasets are modified to binary problem. In the CAR dataset the *kidney* (11 instances) class is set as the minority class, and the rest classes as the majority class. The other datasets are set as above, where GLIOMA dataset uses *cancer oligodendrogliomas* (7 instances) class, LUNG2 datasets use *small-cell lung carcinomas* (20 instances) class, and SRBCT datasets use *Burkitt lymphoma* (11 instances) class as the minority class (Maldonado et al., 2014).

Considering all these datasets have relatively few instances, the evaluation model is built using leave-one-out cross validation (LOOCV). The percentage of SMOTE to generate is 200%. For the information gain, the top 60 features ($r_{nk} = 60$) are selected for the succeeding phase. The penalty function parameter $C$ of SVM is set to 1. And for the SSO algorithm, $c_w = 0.1$, $c_p = 0.4$, $c_g = 0.9$, $\delta=0.95$. The number of solutions ($pop$) is set to 30, the number of iterations ($iter$) is set to 100, and the IG-SSO is repeated 10 times.

<table>
<thead>
<tr>
<th>Name</th>
<th>Features</th>
<th>Instances</th>
<th>Minority</th>
<th>%Minority</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAR</td>
<td>9182</td>
<td>174</td>
<td>11</td>
<td>6.3</td>
</tr>
<tr>
<td>GLIOMA</td>
<td>4433</td>
<td>50</td>
<td>7</td>
<td>14.0</td>
</tr>
<tr>
<td>LUNG2</td>
<td>3312</td>
<td>203</td>
<td>20</td>
<td>9.8</td>
</tr>
<tr>
<td>SRBCT</td>
<td>2308</td>
<td>83</td>
<td>11</td>
<td>13.3</td>
</tr>
</tbody>
</table>

The results are shown in Table 2 and the best result is marked in bold. The HO-BFE$_{bl}$ and BFE-SVM$_{bl}$ are two method proposed in Maldonado’s research. Since the number of features is not the main part of Maldonado’s study, some of the results are not shown in this table. As seen in Table 2, the IG-SSO has the best g-mean value among all result. The
classification results reach 100% accuracy in LUNG2 and SRBCT datasets. In LUNG2 datasets, the \textsc{HO-BFE} has lower number of selected features, but other three datasets the \textsc{IG-SSO} pick fewer features for classification and has better g-mean.

For the re-sampling technique, the result shows that SMOTE does increase the performance of classification in \textsc{IG-SSO}, all datasets performed better with SMOTE except LUNG2 dataset which already reach 100% g-mean. However, it seems the re-sampling technique has low influence on the \textsc{HO-BFE} and \textsc{BFE-SVM} method and causes a decrease in g-mean.

Table 2: Average g-mean, in percentage, and the number of features selected for the dataset

<table>
<thead>
<tr>
<th></th>
<th>CAR</th>
<th>GLIOMA</th>
<th>LUNG2</th>
<th>SRBCT</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>no SMOTE</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\textsc{HO-BFE}</td>
<td>93.1</td>
<td>81.7</td>
<td>100.0</td>
<td>96.5</td>
</tr>
<tr>
<td>\textsc{BFE-SVM}</td>
<td>92.1</td>
<td>80.8</td>
<td>100.0</td>
<td>99.9</td>
</tr>
<tr>
<td>\textsc{IG-SSO}</td>
<td>93.7</td>
<td>98.0</td>
<td>100.0</td>
<td>26.7</td>
</tr>
<tr>
<td><strong>SMOTE</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\textsc{HO-BFE}</td>
<td>92.6</td>
<td>74.8</td>
<td>98.3</td>
<td>95.8</td>
</tr>
<tr>
<td>\textsc{BFE-SVM}</td>
<td>92.6</td>
<td>72.7</td>
<td>99.2</td>
<td>99.2</td>
</tr>
<tr>
<td>\textsc{IG-SSO}</td>
<td>95.3</td>
<td>99.2</td>
<td>100.0</td>
<td>31.0</td>
</tr>
</tbody>
</table>

5. Conclusion

The imbalance data seriously affect the outcome of feature selection and classification. In this study, a hybrid algorithm \textsc{IG-SSO} is proposed for the high-dimensional feature selection problem with imbalanced dataset. The proposed method uses re-sampling technique SMOTE to deal with class imbalanced problem, and combined filter method, information gain, and wrapper method, SSO, for the feature selection. It is the first SVM based wrapper method facing class imbalanced high-dimensional problem. The performance of proposed method is compared with previous research (Maldonado et al., 2014) and shows that the proposed method can spot a better optimal features subset and thus, increases the accuracy of classifying high-dimensional imbalanced data problem.

Although the experiment shows a promising result for the proposed method, there is work left for future researchers. The number of features selected by information gain is determined by researchers and may vary by case. For the future investigation, researchers can
focus on the filter method phase and construct a threshold that vary the number of features selected, or use other filter method for the pre-selected phase. It is expected that this method can be applied to more scenario and help improve feature selection problems.

References
new algorithm. Paper presented at the AAAI.


