Neural Comput & Applic
DOI 10.1007/s00521-015-1909-9

Real-time transient stability status prediction using cost-sensitive extreme learning machine

Zhen Chen[1] · Xianyong Xiao[1,2] · Changsong Li[1,2] · Yin Zhang[1] · Qingquan Hu[1]

Received: 3 December 2014 / Accepted: 6 April 2015

Abstract Real-time transient stability status prediction (RTSSP) is very important to maintain the safety and stability of electrical power systems, where any unstable contingency will be likely to cause large-scale blackout. Most of machine learning methods used for RTSSP attempt to attain a low classification error, which implies that the misclassification costs of different categories are the same. However, misclassifying an unstable case as stable one usually leads to much higher costs than misclassifying a stable case as unstable one. In this paper, a new RTSSP method based on cost-sensitive extreme learning machine (CELM) is proposed, which recognizes the RTSSP as a cost-sensitive classification problem. The CELM is constructed pursuing the minimum misclassification costs, and its detailed implementation procedures for RSSTP are also researched in this work. The proposed method is implemented on the New England 39-bus electrical power system. Compared with three cost-blind methods (ELM, SVM and DT) and two cost-sensitive methods (cost-sensitive DT, cost-sensitive SVM), the simulation results have proved that the lower total misclassification costs and false dismissal rate with low computational complexity can be achieved by the proposed method, which meets the demands for the computation speed and the reliability of RTSSP.

✉ Changsong Li
lcs_scu@163.com

[1] College of Electrical Engineering and Information Technology, Sichuan University, Chengdu 610065, Sichuan Province, China

[2] Intelligent Electric Power Grid Key Laboratory of Sichuan Province, Chengdu 610065, Sichuan Province, China

Keywords Transient stability status prediction · Cost-sensitive extreme learning machine · Electrical power system · Classification

1 Introduction

The ever-increasing load demand coupled with limited investment in generation and transmission has made the electrical power system operate close to its stability limits. Meanwhile, with the significant increase in the number of plug-in electrical vehicles and the penetration of large-scale sustainable energy sources, the dynamic behavior of the electrical power system becomes more and more uncertain and complex. In such situation, a severe contingency can make the electrical power system lose dynamic stability, resulting in large-scale blackout or even catastrophic accident [1]. As a consequence, dynamic security assessment of the electrical power system has become a very important issue. A fast and reliable tool to predict the stability status (stable/unstable) for imminent operating condition (OC) is necessary for system operators to avoid dynamic insecurity.

Transient stability is an important category of dynamic security and plays a vital role in planning and operation of the electrical power system [2, 3]. Traditional methods such as time-domain simulation and energy function method cannot meet the demand of real-time transient stability status prediction (RTSSP) very well [4]. Time-domain simulation, which employs numerical integration to solve a set of differential-algebraic equations reflecting the transient characteristics of the electrical power system, is deemed as the most accurate method to determine the transient stability status. This method can handle various dynamic models of components and give visual

Published online: 17 April 2015

information of state variables. However, the demerits of time-domain simulation are computational intensive and time-consuming [4, 5]. Energy function method can predict the transient stability status by comparing the transient energy of system against the potential energy at the unstable equilibrium point. Although this method is computational efficient, the results of it may not be accurate enough for application in real-world electrical power systems [3, 6].

Recently, machine learning methods, considered as a promising candidate for RTSSP, attract more and more attentions [7–15]. These methods can extract the mapping relationship between the input features and the corresponding stability status after training offline, and when applied online, the transient stability status can be determined immediately after feeding the input features, which makes them have the potential for real-time application [7].

Various machine learning methods, such as neural networks (NN) [8, 9], support vector machines (SVM) [10], decision trees (DT) [11, 12] and other ensemble methods [13, 15], have been applied to RTSSP, which can be considered as a binary classification problem. However, most of these methods used for RTSSP attempt to achieve a low classification error, which hides the assumption that the misclassification of different categories leads to the same cost [16]. Nevertheless, this assumption is not really reasonable for RTSSP. For example, when a false alarm happens, which means misclassifying a stable case as unstable one, the remedial action would be initiated, and then, a certain number of loads will be cut off. This kind of misclassification may cause a few number of economic losses, but does not disrupt the security and stability of the entire electrical power system; when a false dismissal occurs, which means misclassifying a unstable case as stable one, the system operators cannot realize the emergency situation of the electrical power system and do not take any control action to release its pressure. This kind of misclassification is very likely to result in system collapse and large-scale blackout ultimately. Through the analysis above, false dismissal is much more serious than false alarm, and simply taking the classification error as the performance measure is not proper for RTSSP. Therefore, the goal of machine learning methods utilized for RTSSP should minimize the total misclassification costs rather than minimizing the total classification error, which could prevent blackout accident caused by high-cost false dismissal.

Cost-sensitive learning, which is an important branch of machine learning community, provides an effective way to deal with the classification problem with different misclassification costs [16, 17]. However, to the best of our knowledge, only the cost-sensitive DT (CDT) has ever been used for stability status determination [11] and the cost-sensitive property of RTSSP has not received enough attention.

Extreme learning machine (ELM) is a new kind of feedforward neural network and has been successfully applied to RTSSP due to its fast learning speed and good generation ability [14, 15]. Unfortunately, existing ELM methods also did not recognize the RTSSP as a cost-sensitive problem. In this paper, we propose a new RTSSP method based on cost-sensitive extreme learning machine (CELM), which is constructed pursuing the minimum misclassification costs. The effectiveness and reliability of the proposed method are demonstrated on the New England 39-bus electrical power system.

The remainder of this paper is organized as follows. The problem concerned in this paper is described in Sect. 2. In Sect. 3, cost information about RTSSP and the rationale of ELM and CELM are introduced. The implementation procedures of the CELM-based RTSSP are given in Sect. 4. Following, the effectiveness of the proposed method is validated in Sect. 5. Conclusions are finally discussed in Sect. 6.

2 Problem description

Transient stability refers to the ability of the electrical power system to maintain synchronism after being subjected to a sever disturbance, such as a short circuit on a transmission line, loss of a generator or a large load [2]. The resulting system response involves large excursions of generator rotor angles and is influenced by the nonlinear power–angle relationship [2]. The initial operating state of the system and the severity of the disturbance are two key factors to affect the transient stability status of electrical power system.

As the primary indicators of transient stability status, rotor angles should be expressed referring to a common reference. However, a single generator cannot be chosen as the common reference, since any instability in the reference generator would make the relative angles nonsense [3]. To overcome this difficulty, the concept of power system center of angle (COA) is defined in Eq. 1, which can be used as the reference angle.

$$\delta_{\mathrm{COA}} = \frac{\sum_{i=1}^{N_g} T_i \delta_i}{\sum_{i=1}^{N_g} T_i} \tag{1}$$

where δ_{COA} is the COA, δ_i and T_i are the rotor angle and inertia constant of ith generator, respectively, and N_g is the number of generators.

For illustration, a three-phase short-circuit fault is applied at the bus 4 of the New England 39-bus electrical power system; by setting different fault clearing time and tripping line 4–14, the two typical generator rotor angle

trajectories, respectively, representing transient stability and transient instability can be obtained using time-domain simulation, which are shown in Figs. 1 and 2.

As can be seen from these two figures, the rotor angle curves of transient stability and transient instability are obviously different. For the stable case, the generator rotor angles increase or decrease to the maximum value after the fault occurred, and then oscillate until the steady-state recovery. For the unstable case, the generator rotor angles always increase or decrease until the system lose synchronism.

The purpose of RTSSP in the electrical power system is to determine the transient stability status as soon as possible after suffering large disturbance, so that enough time can be set aside to take better control action if necessary, such as load shedding, dynamic braking and tripping generators. Although time-domain simulation can get accurate and visual results, its calculation is intensive and time-consuming, which becomes the biggest obstacle to prevent this method for RTSSP especially when modern electrical power systems become larger and larger. Hence, it is in urgent need for the method which can predict the transient stability status rapidly and reliably.

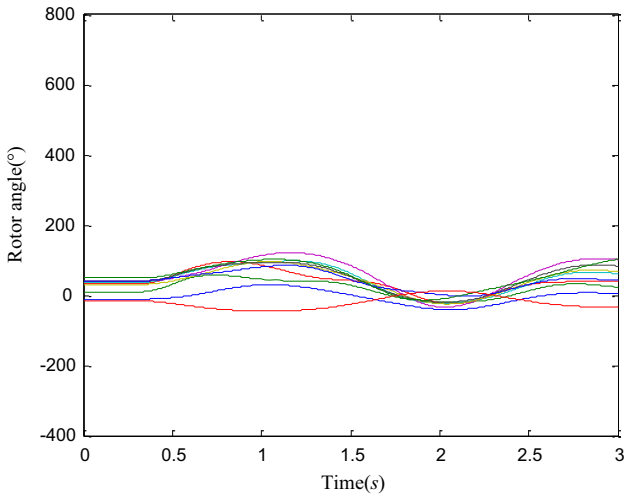

Fig. 1 Typical rotor angles for transient stability

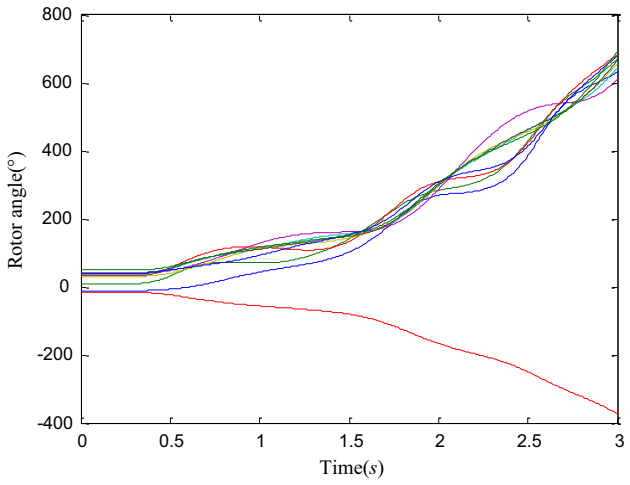

Fig. 2 Typical rotor angles for transient instability

3 Proposed approach

3.1 Cost information

There are two kinds of misclassification cost, named class-dependent cost and example-dependent cost [18]. The former means that misclassifying any example from same class has the same cost, but the example from different classes may have different cost. The latter means misclassifying different example could lead to different cost. In this paper, the former is adapted since the RTSSP is a typical class-dependent cost-sensitive problem.

Generally, the cost confusion matrix of RTSSP can be tabulated as follows

In Table 1, λ_{TS} and λ_{TU} are, respectively, the cost of classifying the stable case and the unstable case correctly; obviously, the value of them are both 0. λ_{FS} represents the cost of a false dismissal, and λ_{FU} is the cost of a false alarm. According to the theory presented by Elkan [16], we can reassign $\lambda_{FS} = \lambda_{FS}/\lambda_{FU,}$ and $\lambda_{FU} = 1$, which will lead to the same solution.

Based on above, we define the misclassification cost matrix $\boldsymbol{C}$ as an $N \times N$ diagonal matrix, which can be written as

$$\boldsymbol{C} = \begin{bmatrix} c_{11} & & & \\ & c_{22} & & \\ & & \ddots & \\ & & & c_{NN} \end{bmatrix}_{N\times N} \tag{2}$$

where N is the number of training samples. The element on the diagonal of matrix $\boldsymbol{C}$ can be represented as c_{ii}, if sample i corresponds the stable sample; the value of c_{ii} is set to 1, otherwise, the value is set to λ_{FS}. As the false dismissal is more serious than false alarm, we can easily know $\lambda_{FS} > 1$, but it is hard to determine the actual value of λ_{FS}. In general, the misclassification cost matrix can be set by the user [18].

Table 1 Cost confusion matrix of RTSSP

Predictive status	Real status	
	Stable	Unstable
Stable	λ_{TS}	λ_{FS}
Unstable	λ_{FU}	λ_{TU}

3.2 Outline of ELM and its optimization model

Different with traditional single hidden layer feed-forward neural networks (SLFN), ELM can calculate the output weights analytically after randomly choosing the input weights and biases during its training. The output function of ELM for generalized SLFN is

$$f_L(\mathbf{x}) = \sum_{i=1}^{L} \boldsymbol{\beta}_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\boldsymbol{\beta} \tag{3}$$

where $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}),\ldots, h_L(\mathbf{x})]$ is ELM nonlinear feature mapping, $\boldsymbol{\beta} = [\boldsymbol{\beta}_1,\ldots, \boldsymbol{\beta}_L]^{\mathrm{T}}$ is the output weight vector between the hidden layer of L nodes to the $m \geq 1$ output nodes [19, 20].

For N arbitrary distinct samples $(\mathbf{x}_i, \mathbf{y}_i)$, where $\mathbf{x}_i = [x_{i1}, x_{i2},\ldots, x_{in}]^{\mathrm{T}}$ and $\mathbf{y}_i = [y_{i1}, y_{i2},\ldots, y_{im}]^{\mathrm{T}}$ (n and m are the number of dimensions of input and output, respectively). If the standard SLFN can approximate the N samples with zeros error, the following equation can be given

$$\sum_{i=1}^{L} \boldsymbol{\beta}_i h_i(\mathbf{x}_j) = \mathbf{y}_j \quad j = 1,\ldots,N \tag{4}$$

Equation 4 can be compactly rewritten as

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y} \tag{5}$$

where $\mathbf{H}$ represents the hidden layer output matrix of ELM and can be represented as

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h_1(\mathbf{x}_1) & \cdots & h_L(\mathbf{x}_1) \\ \vdots & \cdots & \vdots \\ h_1(\mathbf{x}_N) & \cdots & h_L(\mathbf{x}_N) \end{bmatrix}_{N\times L} \tag{6}$$

In Eq. 4, $\boldsymbol{\beta}$ is the matrix of output weight and $\mathbf{Y}$ is the matrix of class label, which can be expressed, respectively, as

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{\beta}_L^{\mathrm{T}} \end{bmatrix}_{L\times m}$$

and

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1^{\mathrm{T}} \\ \vdots \\ \mathbf{y}_N^{\mathrm{T}} \end{bmatrix}_{N\times m} \tag{7}$$

It has been proved that after randomly generating the input weights biases, the matrix $\mathbf{H}$ can be known and needs not to be tuned [21]. Therefore, the training process of ELM is equivalent to solve the linear Eq. 4, and the output weights $\boldsymbol{\beta}$ can be estimated as

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{Y} \tag{8}$$

where $\mathbf{H}^{\dagger}$ is the Moore–Penrose generalized inverse of the matrix $\mathbf{H}$.

The optimization model of ELM is proposed in [20], which can be formulated as

$$\begin{aligned} \min \quad & f = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + \frac{M}{2}\sum_{i=1}^{N}\varepsilon_i^2 \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} = y_i - \varepsilon_i \quad i = 1,\ldots,N \end{aligned} \tag{9}$$

where ε_i is the training error of sample $\mathbf{x}_i$. M is a parameter, which provides a tradeoff between the separating margin distance and the total training error. It is worth noting that because the RTSSP is a binary classification problem, we consider there is only one node in the output layer, and the output dimension m is set to 1 in Eq. 8 and hereafter.

3.3 Rationale of CELM

Existing ELM is constructed pursuing the minimum classification error. In this subsection, the rationale of CELM whose objective is to minimize the misclassification costs is introduced. The optimization model of CELM can be described as

$$\begin{aligned} \min \quad & f = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + \frac{M}{2}\sum_{i=1}^{N}c_{ii}\varepsilon_i^2 \\ \text{s.t.} \quad & \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} = y_i - \varepsilon_i \quad i = 1,\ldots,N \end{aligned} \tag{10}$$

The Lagrangian function associated with the equality constrained problem of Eq. 9 is

$$L = \frac{1}{2}\|\boldsymbol{\beta}\|^2 + \frac{M}{2}\sum_{i=1}^{N}c_{ii}\varepsilon_i - \sum_{i=1}^{N}\alpha_i(\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - y_i + \varepsilon_i) \tag{11}$$

where α_i is the Lagrangian multiplier corresponding to the ith training sample. According to the Karush–Kuhn–Tucker (KKT) first-order optimality conditions, the following equations can be derived.

$$\frac{\partial L}{\partial \boldsymbol{\beta}} = \mathbf{0} \rightarrow \boldsymbol{\beta} = \mathbf{H}^{\mathrm{T}}\boldsymbol{\alpha} \tag{12}$$

$$\frac{\partial L}{\partial \varepsilon_i} = 0 \rightarrow \alpha_i = Mc_{ii}\varepsilon_i, \quad i = 1,\ldots,N \tag{13}$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \rightarrow \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - y_i + \varepsilon_i = 0, \quad i = 1,\ldots,N \tag{14}$$

where $\boldsymbol{\alpha} = [\alpha_1,\ldots, \alpha_N]^{\mathrm{T}}$.

The output weights $\boldsymbol{\beta}$ can be obtained by combining and solving above three equations, which including two kinds of situations:

1. If the number of training samples is small, for example, it is much smaller than the number of hidden nodes.

$$\boldsymbol{\beta} = \mathbf{H}^{\mathrm{T}}\left(\boldsymbol{C}\mathbf{H}\mathbf{H}^{\mathrm{T}} + \frac{E}{M}\right)^{-1}\boldsymbol{C}\mathbf{Y} \tag{15}$$

2. If the number of training samples is large, for example, it is much larger than the number of hidden nodes.

$$\boldsymbol{\beta} = \left(\mathbf{H}^{\mathrm{T}}\boldsymbol{C}\mathbf{H} + \frac{E}{M}\right)^{-1}\mathbf{H}^{\mathrm{T}}\boldsymbol{C}\mathbf{Y} \tag{16}$$

Although the derivation process of CELM is similar to that of weighted ELM proposed in [22], the CELM is specially designed for handling the cost-sensitive problem which weighted ELM does not refer to.

4 Implementation of the CELM for RTSSP

4.1 Input features description

Input features are the important factors affecting the classification performance of machine learning methods. In general, the input features in RTSSP can be categorized as pre-fault features [15] and post-fault features [11]. Pre-fault features are based on the steady-state information of electrical power system provided by supervisory control and data acquisition (SCADA) system, such as load, generation and voltage magnitude. Although the pre-fault features have correlation with the stability status to some extent, they cannot directly reflect the dynamic characteristics of electrical power system. Post-fault features make more sense than pre-fault features since they contain the dynamic information, especially with the advent of phase measurement units (PMU); the dispatch center can obtain the real-time and synchronous dynamic response of electrical power system, which makes the post-fault features become more accurate and effective.

As the post-fault states of the generator rotors strongly represent the transient stability status [23], the post-fault features associated with rotor such as rotor angle, rotor speed, rotor acceleration rate and kinetic energies should be the first choice for the input features of RTSSP. In addition, due to the growing scale of electrical power system, some system-level post-fault features independent of the size of system are more suitable for RTSSP to avoid the curse of dimension. Based on a comprehensive literature survey and a large number of simulations [23, 24], the 21 system-level features that are closely related to the transient stability status are chosen as the input features, which are shown in Table 2.

Here, t_0 indicates the fault occurrence time, t_{cl} indicates the fault clearing time, *and* $t_{\mathrm{cl}+3}$ and $t_{\mathrm{cl}+6}$ indicate 3 cycles and 6 cycles after the fault clearing time. In this work, the frequency is considered to be 50 HZ, which means its cycle is 0.02 s. It should be mentioned that all the features employed in this paper can be obtained with a few amount of calculation using the dynamic data from time-domain simulation.

Table 2 The input features

No	Features description
X_1	Mean value of all the generator mechanical power before t_0
X_2	Mean value of all the generator accelerating power at t_0
X_3	Maximum value of all the rotor acceleration rate at t_0
X_4	The total rotor kinetic energies at t_{cl}
X_5	Maximum value of rotor angle difference between any two generators at t_{cl}
X_6	Maximum value of rotor speed difference between any two generators at t_{cl}
X_7	Maximum value of rotor acceleration rate difference between any two generators at t_{cl}
X_8	Maximum value of rotor angle relative to the COA at t_{cl}
X_9	Kinetic energy of the generator which has largest rotor angel value at t_{cl}
X_{10}	Maximum value of rotor angle difference between any two generators at $t_{\mathrm{cl}+3}$
X_{11}	Maximum value of rotor speed difference between any two generators at $t_{\mathrm{cl}+3}$
X_{12}	Maximum value of rotor acceleration rate difference between any two generators at $t_{\mathrm{cl}+3}$
X_{13}	Maximum value of rotor angle relative to the COA at $t_{\mathrm{cl}+3}$
X_{14}	Kinetic energy of the generator which has largest rotor angel value at $t_{\mathrm{cl}+3}$
X_{15}	Maximum value of rotor angle changing rate between t_{cl} and $t_{\mathrm{cl}+3}$
X_{16}	Maximum value of rotor angle difference between any two generators at $t_{\mathrm{cl}+6}$
X_{17}	Maximum value of rotor speed difference between any two generators at $t_{\mathrm{cl}+6}$
X_{18}	Maximum value of rotor acceleration rate difference between any two generators at $t_{\mathrm{cl}+6}$
X_{19}	Maximum value of rotor angle relative to the COA at $t_{\mathrm{cl}+6}$
X_{20}	Kinetic energy of the generator which has largest rotor angel value at $t_{\mathrm{cl}+6}$
X_{21}	Maximum value of rotor angle changing rate between $t_{\mathrm{cl}+3}$ and $t_{\mathrm{cl}+6}$

4.2 Database generation

The database used for training and testing the classifier should be generated offline by time-domain simulation. In order to improve the performance of the CELM, a large number of OCs which can comprehensively represent different snapshots of electrical power system should be created. Therefore, the active power and reactive power of load buses are varied randomly between the 90–120 % of their primary value in base OC, which can be formulated as follows.

$$P_{Li}^{k} = P_{Li}^{0}(0.9 + 0.3\sigma) \tag{17}$$

$$Q_{Li}^{k} = Q_{Li}^{0}(0.9 + 0.3\sigma) \tag{18}$$

where P_{Li}^{k} and Q_{Li}^{k} respectively, denote the active power and reactive power at the ith load bus for kth OC. P_{Li}^{0} and Q_{Li}^{0} respectively, denote the base active power and base reactive power at the ith load bus. σ is a random number between 0 and 1 obeying the uniform distribution.

After determining the load level of electrical power system, the output of generator active power should be adjusted accordingly. However, in order to ensure the new generated OC is feasible, the load flow is calculated by the Newton–Raphson method. If the load flow of this OC is convergent, it is adopted. Otherwise, it is eliminated.

For each created OC, randomly choosing one line as the faulted line, time-domain simulation is executed after giving the fault information. Then, the input features can be obtained by simple calculation using the results provided by time-domain simulation, and the transient stability status can be determined by the following index [12]:

$$\eta = \frac{360 - \Delta\delta_{\max}}{360 + \Delta\delta_{\max}} \tag{19}$$

where $\Delta\delta_{\max}$ is the maximum rotor angle deviation of any two generators during the simulation time. Whenever the transient status index η becomes negative, which means that the rotor anger difference of any two generators exceeds 360°, the system is deemed as transiently unstable, and the class label of "1" is assigned to the sample; otherwise, the system remains stable, and the label of "0" is assigned to the simulation sample.

The sample in the database consists of the input features and its corresponding class label. The above process will be repeated until enough number of samples are generated, and the overall procedures for database generation are shown in Fig. 3.

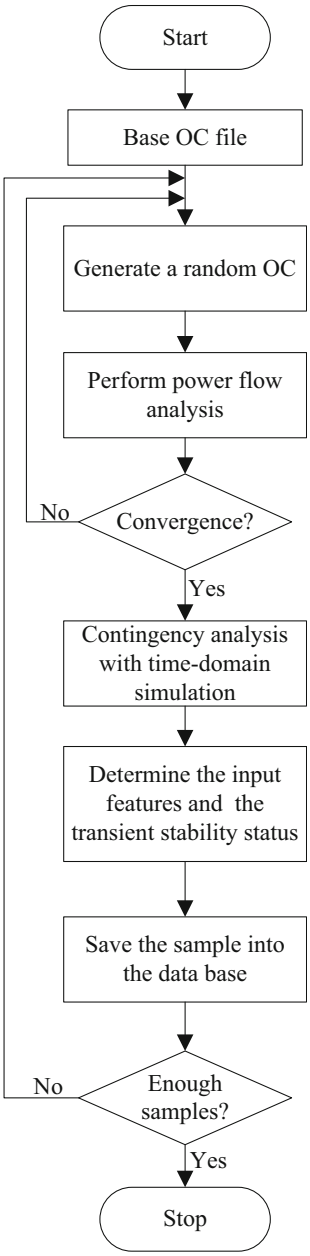

Fig. 3 Flowchart of database generation

4.3 Performance metrics

It is now well known that accuracy is not an appropriate performance metric when there are unequal misclassification costs [25]. For better assessing the performance of classifiers in cost-sensitive problem, the following metrics are defined:

1. *Costs*: the total costs of misclassifying both the stable cases and the unstable cases

$$Costs = N_{\mathrm{FU}} \times \lambda_{\mathrm{FU}} + N_{\mathrm{FS}} \times \lambda_{\mathrm{FS}} \tag{20}$$

2. False dismissal rate (*FDR*): the proportion of the number of unstable cases misclassified as stable cases in total number of unstable cases, which is also deemed as the metric of the reliability of RTSSP [24]

$$FDR = \frac{N_{FS}}{N_{FS} + N_{TU}} \tag{21}$$

3. False alarm rate (*FAR*): the proportion of the number of stable cases misclassified as unstable cases in total number of stable cases

$$FAR = \frac{N_{FU}}{N_{FU} + N_{TS}} \tag{22}$$

4. *G-mean*: the geometric mean of the stable case classification accuracy and the unstable case classification accuracy, which is used as the overall performance of classifiers in lieu of conventional accuracy

$$G\text{-mean} = \sqrt{\frac{N_{TU}}{N_{FS} + N_{TU}} \times \frac{N_{TS}}{N_{FU} + N_{TS}}} \tag{23}$$

where N_{FU} denotes the number of stable cases misclassified as unstable cases; N_{FS} denotes the number of unstable cases misclassified as stable cases; N_{TS} and N_{TU}, respectively, denote the number of stable cases and unstable cases classified correctly.

4.4 Procedures of the CELM-based RTSSP

The flowchart of the proposed CELM-based RTSSP is illustrated in Fig. 4.

The main steps of the proposed algorithm are described as follows:

1. Normalize the input features to a range of 0–1 before they are used for training and testing.
2. Determine the misclassification cost matrix by the user. As λ_{FU} is fixed to 1, the user can adjust the misclassification cost matrix by changing the value of λ_{FS}.
3. Train the CELM and then use it for classifying the test samples. In order to comprehensively test the proposed algorithm without losing generality, fivefold cross-validation methodology is utilized. During the training process, the grid search and fourfold cross-validation technique are employed to obtain the optimal parameters of CELM, which uses the *Costs* as the criterion.
4. Calculate the performance metrics.

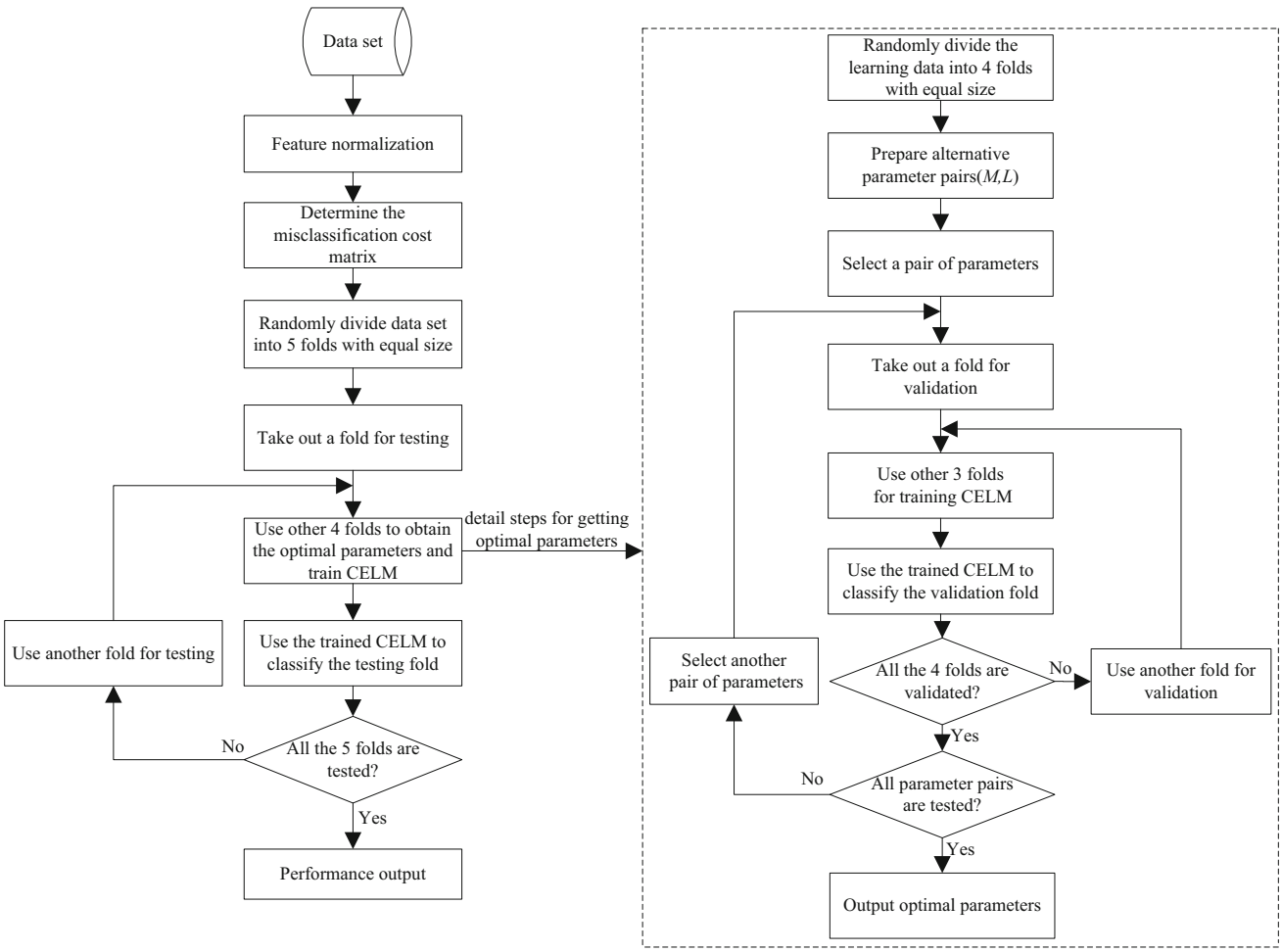

Fig. 4 Flowchart of the CELM-based RTSSP

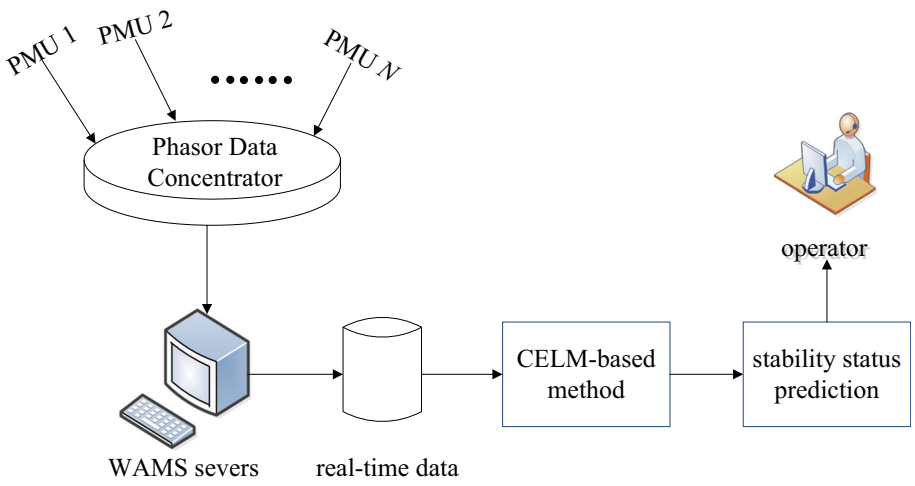

Fig. 5 Practical application framework of the CELM-based method

4.5 Practical application framework

After training offline using the data from simulation results, the CELM-based method can be used for practical RTSSP; the practical application framework is illustrated in Fig. 5.

The phasor data concentrator (PDC) are used for collecting and regulating the time of the measurement data from different PMUs, then the synchrophasor data in the PDC are transferred to the wide area measurement system (WAMS) severs, which are located at the control center. The real-time data can be used as the input features for the CELM-based method after a few amount of calculation. Once the operator is alerted by the proposed model that the power system will lose transient stability, the remedial action can be immediately initiated.

5 Simulation results

5.1 Brief introduction

The proposed CELM-based RTSSP method is tested on the New England 39-bus electrical power system, which is the most commonly used test system for RTSSP [13]. The system has 10 generators, 39 buses and 46 branches, and its single-line diagram is shown in Fig. 6. During the database generation process, Power System Toolbox, which is an open source Matlab-based software package [26], is used for solving power flow and performing time-domain simulation. In addition, the fault occurrence time is set to 0.1 s, the fault clearing time is set to 0.22 s, and the fault type is regarded as a three-phase short-circuit fault. After the faulted line is chosen randomly, the fault location is considered at the sending end of the line. Totally, 5000 samples are generated, containing 3862 stable samples and 1138 unstable samples.

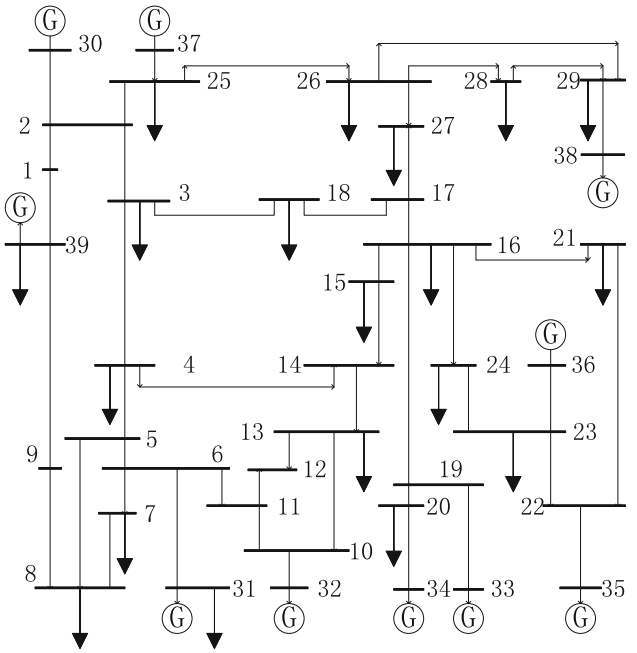

Fig. 6 Single-line diagram of the New England 39-bus electrical power system

5.2 Comparison with other methods

We compare the performance of CELM with three cost-blind methods, which are SVM, DT and ELM, and two cost-sensitive methods, which are cost-sensitive SVM (CSVM) [27] and CDT. In this subsection, the value of λ_{FS} is fixed to 6, and we also discuss the influence of different value of λ_{FS} on the simulation results later.

Sigmoid function is chosen as the activation function of ELM and CELM, and the parameters named M and L are chosen, respectively, from the range $[2^{-6}, 2^{-4},\ldots, 2^{12}]$ and $[50, 100,\ldots, 500]$, containing $10 \times 10 = 100$ alternative parameter pairs. For SVM and CSVM, the radial basis

function is chosen as the kernel function, and the optimal parameters of these two methods incorporating regularization parameter and kernel parameter are selected from 81 different combinations of $[2^{-4}, 2^{-2},\ldots, 2^{12}]$ and $[2^{-8}, 2^{-6},\ldots,2^{8}]$. The classification and regression trees (CART), which is the widely used DT learner, and corresponding cost-sensitive CART employed in [11] are utilized for comparison. Since the ELM can be deemed as the special case of CELM when $\lambda_{FS} = 1$, the procedure of applying the ELM to RTSSP is identical to that of CELM-based method except that the ELM does not take the misclassification cost matrix into consideration.

As the input weights and biases of ELM and CELM are randomly generated, the ELM and CELM are operated repeatedly for ten times with the same training and testing data to improve the statistical influence, and the experimental results of them are the average results of these 10 trials [28, 29].

It is worth noting that all the methods adapt fivefold cross-validation method for testing. The performance of the CELM-based method in each fold is shown in Table 3, and the results of these 6 methods shown in Table 4 are the average results of five folds.

From Table 3, we can find that the average of *FDR* of CELM-based method is very close to 0, especially in Fold-4, the value of *FDR* is equal to 0, which demonstrates that the proposed CELM-based RTSSP is fairly reliable. Also, we can see that the performances from different fold have not much difference between each other, which proves the robustness of the CELM-based method.

Table 3 Performance of CELM-based method

Fold number	*Costs*	*FDR* (%)	*FAR* (%)	*G-mean* (%)
Fold-1	40.9	0.09	5.12	97.4
Fold-2	38.4	0.13	4.74	97.5
Fold-3	46.1	0.37	5.26	97.2
Fold-4	38.6	0	5.03	97.5
Fold-5	43.1	0.26	5.14	97.3
Average	41.4	0.17	5.06	97.4

Table 4 Comparison of different methods

Classifier	*Costs*	*FDR* (%)	*FAR* (%)	*G-mean* (%)
DT	121.8	7.85	1.94	95.1
SVM	109.4	6.68	2.35	95.5
ELM	101.5	6.42	1.80	95.9
CDT	59.4	1.23	5.51	96.6
CSVM	49.8	0.62	5.35	97.0
CELM	41.4	0.17	5.06	97.4

The results shown in Table 4 indicate that the CELM-based method achieves better performance than other five methods in *Costs*, *FDR* and *G-mean*. For RTSSP, the occurrence of false dismissal is very dangerous and should be avoided. Instead, the occurrence of false alarm could be acceptable reluctantly. The CELM-based RTSSP could achieve smaller costs by preventing the occurrence of high-cost false dismissal while maintaining high *G-mean*.

5.3 Influence of different λ_{FS}

In this subsection, we study the performance of these six methods with different setting of λ_{FS}. The value of λ_{FS} varies from 2 to 8 with 1 as interval, and the results are shown in Fig. 7.

As can be seen in Fig. 6, cost-sensitive methods are more suitable than cost-blind methods since they achieve lower *C*osts and *FDR*. Under all different values of λ_{FS}, the CELM-based method can get better results compared with other two cost-sensitive approaches, which indicates clearly that the CELM is the best choice for RTSSP among the compared methods.

From the simulation result, we can see that the *FDR* and the *FAR* decreases and increases, respectively, along with the increase of the value of λ_{FS}. When the value of λ_{FS} is greater than 6, the *FDR* decreases very slightly while the *FAR* still increases quite fast. In addition, from the results of the other two metrics: *Costs* and *G-mean*, we can also see that the optimal performance can be obtained when λ_{FS} is 6. Therefore, in this study, the value of λ_{FS} can be set to 6 by the user, which can provide good classification performance and meet the requirements of transient stability status prediction.

5.4 Computational complexity comparison

In this subsection, the computational complexity of CELM-based method is compared with other 5 methods using the same experimental parameters in subsection 5.2. All simulations are implemented in Matlab R2010b environment running on a PC with Inter core i3 CPU @3.3 GHz and 4G RAM. DT and CDT are implemented using the Matlab function *classregtree*, and for SVM and CSVM, the LIBSVM software package is utilized [30]. Figure 8 shows the average training time and testing time of these methods.

It is noticeable that the computational complexity of CELM-based method is a little higher than that of ELM-based method, but lower than other four methods. The advantage of low computational complexity of CELM makes it very suitable for real-time application.

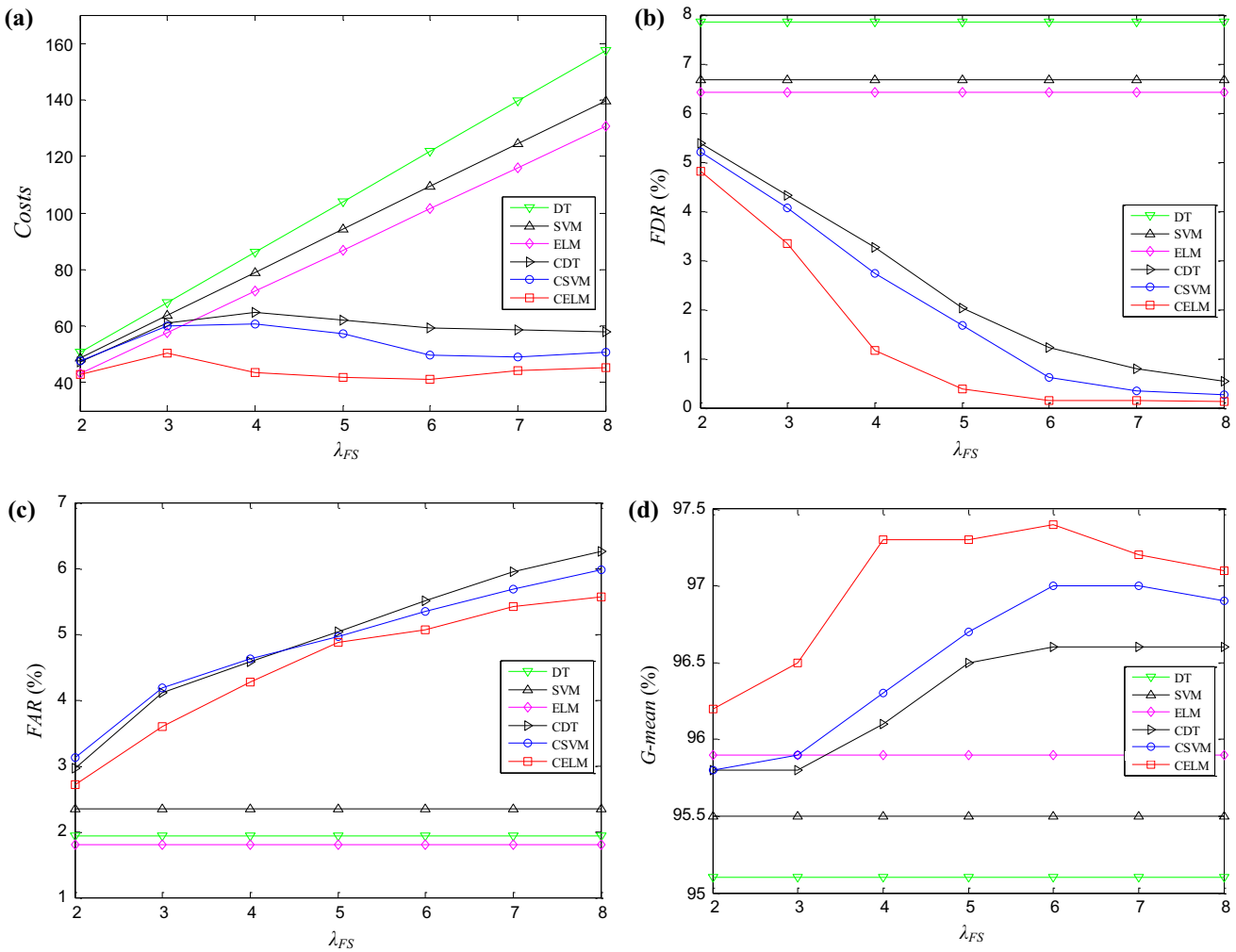

Fig. 7 Performance of these six methods with different value of λ_{FS}. **a** *Costs* **b** *FDR* **c** *FAR* **d** *G-mean*

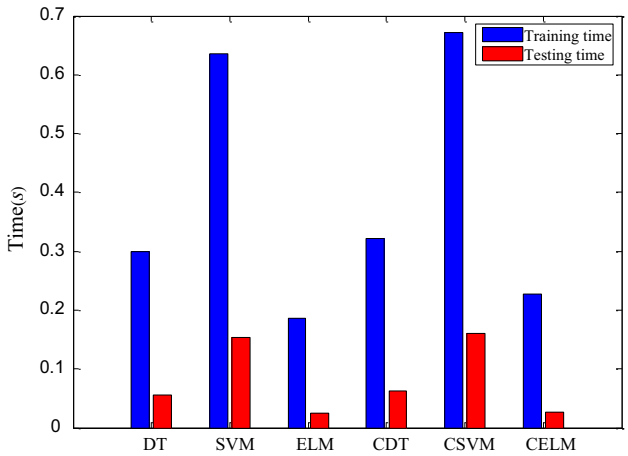

Fig. 8 Computational complexity of each method

6 Conclusions

Conventional machine learning methods used for RTSSP prefer to attain the low classification error rate. However, it is not reasonable and realistic in electrical power systems because the false dismissal will lead to much larger amount of losses than false alarm. In this paper, a novel RTSSP method based on CELM is proposed, which recognizes the RTSSP as a cost-sensitive problem. The proposed method is implemented on the New England 39-bus electrical power system, and the simulation results demonstrate that cost-sensitive methods are more appropriate than cost-blind methods when applied to RTSSP. Compared with two state-of-the-art cost-sensitive methods, the CELM-based method could achieve better performance with lower computational complexity, which meets the requirements

for the computation speed and the reliability of RTSSP. Our future work should aim to capture the best features, which are beneficial for handling the cost-sensitive problem, and propose a better RTSSP model.

References

1. Pourbeik P, Kundur PS, Taylor CW (2006) The anatomy of a power grid blackout-root cause and dynamic of recent major blackouts. IEEE Power Energy Mag 4:22–29
2. IEEE, CIGRE joint Task Force on Stability Terms and Definitions (2004) Definition and classification of power system stability. IEEE Trans Power Syst 19:1387–1401
3. Kunder P (1994) Power system stability and control. McGraw-Hill, New York
4. Li MY, Pal A, Phadke AG, Throp JS (2014) Transient stability prediction based on apparent impedance trajectory recorded by PMUs. Int J Electr Power 54:498–504
5. Tang CK, Graham CE, Eikady M, Alden RTH (1994) Transient stability index from conventional time domain simulation. IEEE Trans Power Syst 9:1524–1530
6. Athay T, Podmore R, Virmani S (1979) A practical method for the direct analysis of transient stability. IEEE Trans PAS 98:573–584
7. Amjady N, Banihashemi SA (2010) Transient stability prediction of power systems by a new synchronism status index and hybrid classifier. IET Gener Trans Distrib 4:509–518
8. Hashiesh F, Mostafa HE, Khatib AR, Helal I, Mansour MM (2012) An intelligent wide area synchrophasor based system for predicting and mitigating transient instabilities. IEEE Trans Smart Grid 3:645–652
9. Karami A, Esmaili SZ (2013) Transient stability assessment of power system described with detailed models neural networks. Int J Electr Power 45:279–292
10. Moulin LS, Silva AP, Sharkawi MA, Marks RJ (2004) Support vector machines for transient stability analysis of large-scale power systems. IEEE Trans Power Syst 19:818–825
11. Sun K, Likhate S, Vittal V, Kolluri VS, Mandal S (2007) An online dynamic security assessment scheme using phasor measurements and decision trees. IEEE Trans Power Syst 22:1935–1943
12. Liu CX, Sun K, Rather ZH, Chen Z, Bak CL, Thogersen P, Lund P (2014) A systematic approach for dynamic security assessment and the corresponding preventive control scheme based on decision trees. IEEE Trans Power Syst 29:717–730
13. Kamwa I, Samantaray SR, Joos G (2010) Catastrophe predictors from ensemble decision-tree learning of wide-area severity indices. IEEE Trans Smart Grid 1:144–158
14. Xu Y, Dong ZY, Meng K, Zhang R, Wong KP (2010) Real-time transient stability assessment model using extreme learning machine. IET Gener Trans Distr 5:314–322
15. Xu Y, Dong ZY, Zhao JH, Zhang P, Wong KP (2012) A reliable intelligent system for real-time dynamic security assessment of power systems. IEEE Trans Power Syst 27:1253–1263
16. Elkan C (2001) The foundation of cost-sensitive learning. In: proceedings of 2001 international joint conference on artificial intelligence
17. Viaene S, Dedene G (2005) Cost-sensitive learning and decision making revisited. Eur J Oper Res 166:212–220
18. Zhang Y, Zhou ZH (2010) Cost-sensitive Face recognition. IEEE Trans Pattern Anal Mach Intell 32:1758–1769
19. Huang G, Huang GB, Song SJ, You KY (2015) Trends in extreme learning machines: A review. Neural Networks 61:32–48
20. Huang GB, Zhou HM, Ding XJ, Zhang R (2012) Extreme learning machine for regression and multiclass classification. IEEE Trans Syst Man Cybern B 42:512–529
21. Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and application. Neurocomputing 70:489–501
22. Zong WW, Huang GB, Chen YQ (2013) Weighted extreme learning machine for imbalance learning. Neurocomputing 101:229–242
23. Tao SK, Gu XP, Zeng QY, Lo KL (1998) Deriving a transient stability index by neural network for power system security assessment. Eng Appl Artif Intell 11:771–779
24. Kamwa I, Samantaray SR, Joos G (2012) On the accuracy versus transparency trade-off of data-mining models for fast-response PMU-based catastrophe predictors. IEEE Trans Smart Grid 3:152–161
25. Liu XY, Wu JX, Zhou ZH (2009) Exploratory undersampling for class-imbalance learning. IEEE Trans Syst Man Cybern B 39:539–550
26. Chow JH, Cheung KW (1992) A toolbox for power system dynamics and control engineering education and research. IEEE Trans Power Syst 7:1559–1564
27. Netoff T, Park Y, Parhi K (2009) Seizure prediction using cost-sensitive support vector machine. In: Proceeding of the 31th annual international conference on engineering in medicine and biology society
28. Xu Y, Dai YY, Dong ZY, Zhang R, Meng K (2013) Extreme learning machine-based predictor for real-time frequency stability assessment of electrical power system. Neural Comput Appl 22:501–508
29. Yang JC, Xie SJ, Yoon S, Park DS, Fang ZJ, Yang SY (2013) Fingerprint matching based on extreme learning machine. Neural Comput Appl 22:435–445
30. Chang CC, Lin CJ. LIBSVM: a library for support vector machines. Available: http://www.csie.ntu.edu.yw/∼cjlin/libsvm/