Granular Ball Twin Support Vector Machine

A. Quadir, *Graduate Student Member, IEEE*, M. Sajid, *Graduate Student Member, IEEE*, and M. Tanveer, *Senior Member, IEEE*

***Abstract*— Twin support vector machine (TSVM) is an emerging machine learning model with versatile applicability in classification and regression endeavors. Nevertheless, TSVM confronts noteworthy challenges: 1) the imperative demand for matrix inversions presents formidable obstacles to its efficiency and applicability on large-scale datasets; 2) the omission of the structural risk minimization (SRM) principle in its primal formulation heightens the vulnerability to overfitting risks; and 3) the TSVM exhibits a high susceptibility to noise and outliers and also demonstrates instability when subjected to resampling. In view of the aforementioned challenges, we propose the granular ball TSVM (GBTSVM). GBTSVM takes granular balls (GBs), rather than individual data points, as inputs to construct a classifier. These GBs, characterized by their coarser granularity, exhibit robustness to resampling and reduced susceptibility to the impact of noise and outliers. We further propose a novel large-scale GBTSVM (LS-GBTSVM). LS-GBTSVM's optimization formulation ensures two critical facets: 1) it eliminates the need for matrix inversions, streamlining the LS-GBTSVM's computational efficiency; and 2) it incorporates the SRM principle through the incorporation of regularization terms, effectively addressing the issue of overfitting. The proposed LS-GBTSVM exemplifies efficiency, scalability for large datasets, and robustness against noise and outliers. We conduct a comprehensive evaluation of the GBTSVM and LS-GBTSVM models on benchmark datasets from UCI and KEEL, both with and without the addition of label noise, and compared with existing baseline models. Furthermore, we extend our assessment to the large-scale NDC datasets to establish the practicality of the proposed models in such contexts. Our experimental findings and rigorous statistical analyses affirm the superior generalization prowess of the proposed GBTSVM and LS-GBTSVM models compared to the baseline models. The source code of the proposed GBTSVM and LS-GBTSVM models are available at https://github.com/mtanveer1/GBTSVM.**

***Index Terms*— Granular ball (GB), granular computing, large-scale dataset, structural risk minimization (SRM) principle, support vector machine (SVM), twin SVM (TSVM).**

I. Introduction

SUPPORT vector machines (SVMs) [1] are advanced kernel-based machine learning models that maximize the margin between two classes in a classification problem, aiming to find the optimal hyperplane between two parallel supporting hyperplanes. SVM has proven its remarkable utility across diverse domains such as web mining [2] and Alzheimer's disease diagnosis [3]. SVM integrates the structural risk minimization (SRM) principle within its optimization framework, thereby enhancing its generalization capabilities by minimizing an upper bound of the generalization error. SVM solves one large quadratic programming problem (QPP), resulting in escalated computational complexity, which renders it less suitable for large-scale datasets. To alleviate the computational complexity of SVM, Mangasarian and Wild [4] and Jayadeva et al. [5] proposed the generalized eigenvalue proximal SVM (GEPSVM) and twin SVM (TSVM), respectively. On the one hand, GEPSVM solves the generalized eigenvalue problem rather than dealing with a large QPP. On the other hand, TSVM solves two smaller sized QPPs instead of a single large QPP, making TSVM four times faster than the standard SVM [5], [6] and firmly establishing TSVM as a standout and superior choice. TSVM generates a pair of nonparallel hyperplanes, with each hyperplane deliberately situated in close proximity to the data points belonging to one class while ensuring a minimum separation distance of at least one unit from the data points of the other class. However, TSVM encountered two notable challenges: the necessity for matrix inversions and the absence of the SRM principle in its formulation, which presented significant obstacles to its efficacy. Shao et al. [7] and Tian et al. [8] proposed twin-bounded SVM (TBSVM) and improved TSVM (ITSVM), respectively. Both the models incorporated a regularization term in their formulation, allowing the principle of SRM to be employed. In recent years, several variants of TSVM have been proposed for both small and large datasets such as pinball TSVMs [9], [10], least-squares TSVM (LTSVM) [11], and robust energy-based LTSVM (RELTSVM) [12].

While TSVM and its variants effectively address the computational complexity challenges posed by SVM, however both SVM and TSVM encounter difficulties when faced with noisy or outlier-laden datasets. In scenarios where noise perturbs support vectors, the SVM's capacity to discern an optimal hyperplane is impeded, resulting in suboptimal outcomes. To address this issue, fuzzy SVM (FSVM) was introduced in [13], employing a degree of membership function for each training sample. Furthermore, intuitionistic fuzzy TSVM (IFTSVM) [14] is proposed and offers a substantial reduction in the adverse effects of noise and outliers by leveraging a set of membership and nonmembership values to each training sample. Several other variants, such as [15], have been proposed to mitigate the detrimental impact of noise and

Received 19 September 2023; revised 13 February 2024 and 14 July 2024; accepted 4 October 2024. This study is supported by the Indian government's Department of Science and Technology (DST) in collaboration with the Ministry of Electronics and Information Technology (MeITy) Grant No. DST/NSM/R&D_HPC_Appl/2021/03.29 and Science and Engineering Research Board under the Mathematical Research Impact-Centric Support (MATRICS) scheme Grant No. MTR/2021/000787. M. Sajid also acknowledges support from Council of Scientific and Industrial Research (CSIR), New Delhi for providing fellowship under the grants 09/1022(13847)/2022-EMR-I. *(Corresponding author: M. Tanveer.)*

The authors are with the Department of Mathematics, Indian Institute of Technology Indore, Indore 453552, India (e-mail: mscphd2207141002@iiti.ac.in; phd2101241003@iiti.ac.in; mtanveer@iiti.ac.in).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNNLS.2024.3476391, provided by the authors.

Digital Object Identifier 10.1109/TNNLS.2024.3476391

outliers; they often come at the price of increased computational complexity.

Human cognition follows the principle of "large scope first," and the visual system is especially attuned to perceiving global topological features, processing information from larger to smaller scales or from coarse grained to fine grained [16]. Unlike human cognition, most existing machine learning classifiers take inputs in the form of pixels or points because their training process consistently commences at the smallest level of granularity. This lacks the scalability and efficiency of the model. Being inspired by human cognition, Xia et al. [17] introduced granular ball (GB) classifiers, which utilize hyperballs to divide the dataset into various sizes of GBs [18]. Within the framework of granular computing, it is observed that larger granularity sizes present itself as a scalable, efficient, and robust approach that closely resembles the cognitive processes of the human brain [17], [19]. Moreover, a transition toward larger granularity entails an increased risk of reduced detail and compromised accuracy. Conversely, opting for finer granularity allows for heightened detail focus, albeit potentially at the cost of efficiency and resilience in noisy environments. Hence, achieving a judicious balance in granular size is of paramount importance. Over the past decades, scholars have continually engaged in research [20], [21], focusing on breaking down large volumes of information and knowledge into different granularities according to certain tasks. Choosing different granularities based on specific situations can enhance the effectiveness of multigranularity learning approaches and efficiently tackle real-world challenges [22], [23].

Again, to address noise and outliers, an efficient granular ball SVM (GBSVM) [24] is proposed by integrating the GB concept into SVM. GBSVM takes inputs as GBs generated from the dataset rather than the individual data points. GBSVM exhibits better resilience in contrast to standard SVM. Motivated by the merits of employing a granular approach for addressing noise and outliers alongside the efficiency exhibited by TSVM, the prospect of integrating these two concepts appears not only intriguing but also promising. Therefore, we propose granular ball TSVM (GBTSVM). GBTSVM takes GBs as inputs, resulting in improved computational efficiency and a heightened ability to withstand noise and outliers. We further propose large-scale GBTSVM (LS-GBTSVM), which efficiently manages large-scale data by incorporating a regularization term in its primal form, eliminating the need for matrix inversions and reducing overfitting risk.

The main highlights of this study are outlined as follows.

1) We propose a novel GBTSVM. The proposed GBTSVM utilizes GBs as inputs for classifier construction, offering enhanced robustness, resilience to resampling, and computational efficiency compared to SVM and GBSVM.
2) We propose a novel large-scale GBTSVM (LS-GBTSVM) to overcome the challenges associated with handling large-scale datasets encountered by the proposed GBTSVM. The LS-GBTSVM aims to reduce the structural risk inherent in its formulation by incorporating the SRM principle through the addition of a regularization term in its primal formulation. The proposed LS-GBTSVM exhibits efficiency, scalability for large datasets, proficient handling of overfitting, robustness, and improved generalization performance compared to standard SVM, TSVM, and GBSVM.
3) We provide rigorous mathematical frameworks for both GBTSVM and LS-GBTSVM, covering linear and nonlinear kernel spaces. Furthermore, we derive the violation tolerance upper bound (VTUB) for the proposed GBTSVM.
4) We performed experiments on 36 real-world UCI [25] and KEEL [26] datasets; and 10k to 5m large-scale NDC [27] datasets. Numerical experiments and statistical analyses confirm the superiority of the proposed GBTSVM and LS-GBTSVM models compared to the baseline models.
5) The proposed GBTSVM and LS-GBTSVM models are subjected to rigorous testing by adding noise to datasets. Testing under noisy conditions shows that the proposed GBTSVM and LS-GBTSVM models are insensitive to noise and stable to resampling.

The remaining structure of this article is organized as follows. We discuss related works in Section II. In Sections III and IV, we derive the mathematical formulation of the proposed GBTSVM and LS-GBTSVM models, respectively. The time complexity of the proposed GBTSVM model is given in Section V. We discuss the VTUB of the proposed GBTSVM model in Section VI. In Section VII, the discussion of the experimental results is made. Finally, the conclusions and potential future research directions are given in Section VIII.

II. Related Works

In this section, we go through the GB computing method. The mathematical formulation of GBSVM and TSVM is discussed in Section S.I of the Supplementary Material.

A. Notations

Let $D = \{(x_i, t_i), i = 1, 2, \ldots, n\}$, be the traning dataset, where $t_i \in \{+1, -1\}$ represents the label of $x_i \in \mathbb{R}^{1\times m}$. Let us consider the input matrices $A \in \mathbb{R}^{n_1\times m}$ and $B \in \mathbb{R}^{n_2\times m}$, where n_1 (n_2) is the number of data samples belonging to $+1$ (-1) class such that the total number of data samples is $n = n_1+n_2$. The set of generated GBs is denoted as $S = \{\text{GB}_i,\ i = 1, 2, \ldots, p\} = \{((c_i, r_i), y_i),\ i = 1, 2, \ldots, p\}$, where c_i signifies the center, r_i indicates the radius, and y_i is the label of the ith GB. Matrices $C_1 \in \mathbb{R}^{p_1\times m}$ ($C_2 \in \mathbb{R}^{p_2\times m}$) represent the centers associated with the class of $+1$ (-1), where $p_1+p_2 = p$. The center is computed as $c_i^{\pm} = (1/l_i^{\pm}) \sum_{s=1}^{l_i^{\pm}} x_s^{\pm}$, where $c_i^{\pm}$, $x_s^{\pm}$ ($s = 1, 2, \ldots, l_i^{\pm}$), and $l_i^{\pm}$ denotes the center, data samples, and the total number of data samples inside the GB_i, respectively. $R_1 \in \mathbb{R}^{p_1\times 1}$ ($R_2 \in \mathbb{R}^{p_2\times 1}$) denote the vector encompassing the radius of the generated GBs for $+1$ (-1) class and is calculated as $r_i^{\pm} = (1/l_i^{\pm}) \sum_{s=1}^{l_i^{\pm}} |x_s^{\pm} - c_i^{\pm}|$, where $r_i^{\pm}$ represents the radius of each GBs of $+1$ (-1) class.

B. GB Computing

GB computing is a substantial data processing technique introduced in [17] to address the scalability challenges associated with high-dimensional data. A notable advantage of

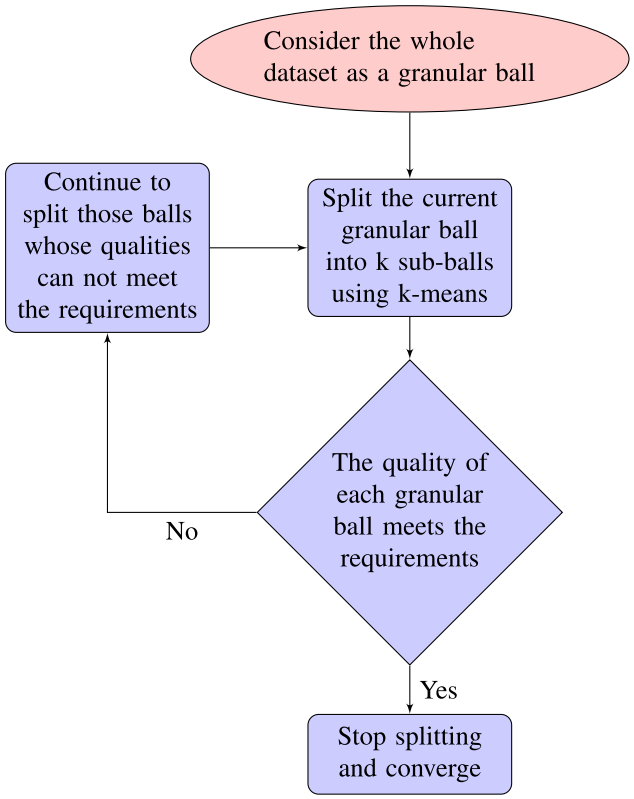

Fig. 1. Process of the GB generation.

this approach is that it requires only the radius and center to characterize a GB, rather than all the data points (samples) contained within that GB. Let "c" represent the center of gravity of the data points contained inside a GB. The radius "r" of the GB depicts the average distance between the center c and the remaining samples contained within a GB. The GB's label is determined by selecting the label of the samples that appear most frequently among the samples enclosed within the GB. To perform a quantitative assessment of the divided GB's mass, the concept of a "purity threshold" is introduced. This threshold refers to the percentage of samples within a GB that shares the same label, specifically the majority labels.

The idea of the GB generation is illustrated in Fig. 1 and the optimization objective of generation of GB_j $(j = 1, 2, \ldots, p)$ can be formulated as follows:

$$\begin{aligned} &\min\ \vartheta_1 \times \frac{n}{\sum_{i=1}^{p} |\mathrm{GB}_i|} + \vartheta_2 \times p \\ &\text{s.t. } pur(\mathrm{GB}_j) \geq T, \quad j = 1, 2, \ldots, p, \end{aligned} \tag{1}$$

where ϑ_1 and ϑ_2 represent the weight coefficients and T denotes the purity threshold. The iterative process to generate the GBs is demonstrated in Fig. 2. At the initial stage, the entire dataset can be conceptualized as a single GB, as illustrated in Fig. 2(a). The GB will undergo division, increasing its purity, as depicted in Fig. 2(b)–(d). Once the purity level of all the GBs satisfies the specified threshold, the algorithm reaches convergence, as demonstrated in Fig. 2(e). The GBs obtained are visualized in Fig. 2(f), illustrating the effectiveness of GB computing in capturing the underlying data distribution.

III. Proposed Granular Ball Twin Support Vector Machine (GBTSVM)

In this section, we provide a detailed mathematical formulation of the proposed GBTSVM model tailored for linear case. Utilizing the GB as input offers two primary advantages. First, it reduces the number of input samples, thereby enhancing training efficiency. Second, the encapsulation of data points within GBs enhances the model's resilience to noise and outliers, as the influence of individual noisy points is mitigated within the localized context of a GB, thus leading to more reliable and accurate results. The formulation of linear GBTSVM is given as follows:

$$\begin{aligned} &\min_{w_1,b_1} \frac{1}{2}\|C_1 w_1 + e_1 b_1\|^2 + d_1 e_2^T \xi_2 \\ &\text{s.t.} -(C_2 w_1 + e_2 b_1) + \xi_2 \geq e_2 + R_2, \quad \xi_2 \geq 0, \end{aligned} \tag{2}$$

and

$$\begin{aligned} &\min_{w_2,b_2} \frac{1}{2}\|C_2 w_2 + e_2 b_2\|^2 + d_2 e_1^T \xi_1 \\ &\text{s.t. } (C_1 w_2 + e_1 b_2) + \xi_1 \geq e_1 + R_1, \quad \xi_1 \geq 0, \end{aligned} \tag{3}$$

where d_1, d_2 (>0) are tunable parameters, ξ_1 and ξ_2 represent the slack variables, and e_1 and e_2 represent the vector of ones with appropriate dimensions. The first term in the objective function of (2) is the sum of squared distances between the hyperplane and the centers of GBs associated with data points belonging to the $+1$ class. Therefore, minimizing it tends to keep the hyperplane close to the GBs of one class. The constraints of the optimization problem (2) mandate that the hyperplane is at least a unit distance from the tangent of the GBs of another class. The tolerance parameter ξ_2 is introduced in (2) to quantify discrepancies arising when the hyperplane deviates from a minimum unit distance threshold. The second term of the objective function of (2) minimizes the sum of error variables, thus attempting to minimize misclassification due to points belonging to -1 class. Similarly, all the components of (3) are defined for -1 class. Fig. 3 displays the geometric representation of the proposed GBTSVM model. The Lagrangian corresponding to the problem (2) is given by

$$\begin{aligned} L = &\frac{1}{2}\|C_1 w_1 + e_1 b_1\|^2 + d_1 e_2^T \xi_2 \\ &- \alpha^T(-(C_2 w_1 + e_2 b_1) + \xi_2 - e_2 - R_2) - \beta^T \xi_2, \end{aligned} \tag{4}$$

where $\alpha \in \mathbb{R}^{p_2 \times 1}$ and $\beta \in \mathbb{R}^{p_1 \times 1}$ are the vectors of Lagrangian multipliers.

Using the Karush–Kuhn–Tucker (KKT) conditions, we have

$$C_1^T(C_1 w_1 + e_1 b_1) + C_2^T \alpha = 0, \tag{5}$$

$$e_1^T(C_1 w_1 + e_1 b_1) + e_2^T \alpha = 0, \tag{6}$$

$$e_2 d_1 - \alpha - \beta = 0, \tag{7}$$

$$-(C_2 w_1 + e_2 b_1) + \xi_2 \geq e_2 + R_2, \quad \xi_2 \geq 0, \tag{8}$$

$$\alpha^T(-(C_2 w_1 + e_2 b_1) + \xi_2 - e_2 - R_2) = 0, \quad \beta^T \xi_2 = 0, \tag{9}$$

$$\alpha \geq 0, \quad \beta \geq 0. \tag{10}$$

Combining (5) and (6) leads to

$$\begin{pmatrix} C_1^T \\ e_1^T \end{pmatrix} (C_1 \ \ e_1) \begin{pmatrix} w_1 \\ b_1 \end{pmatrix} + \begin{pmatrix} C_2^T \\ e_2^T \end{pmatrix} \alpha = 0. \tag{11}$$

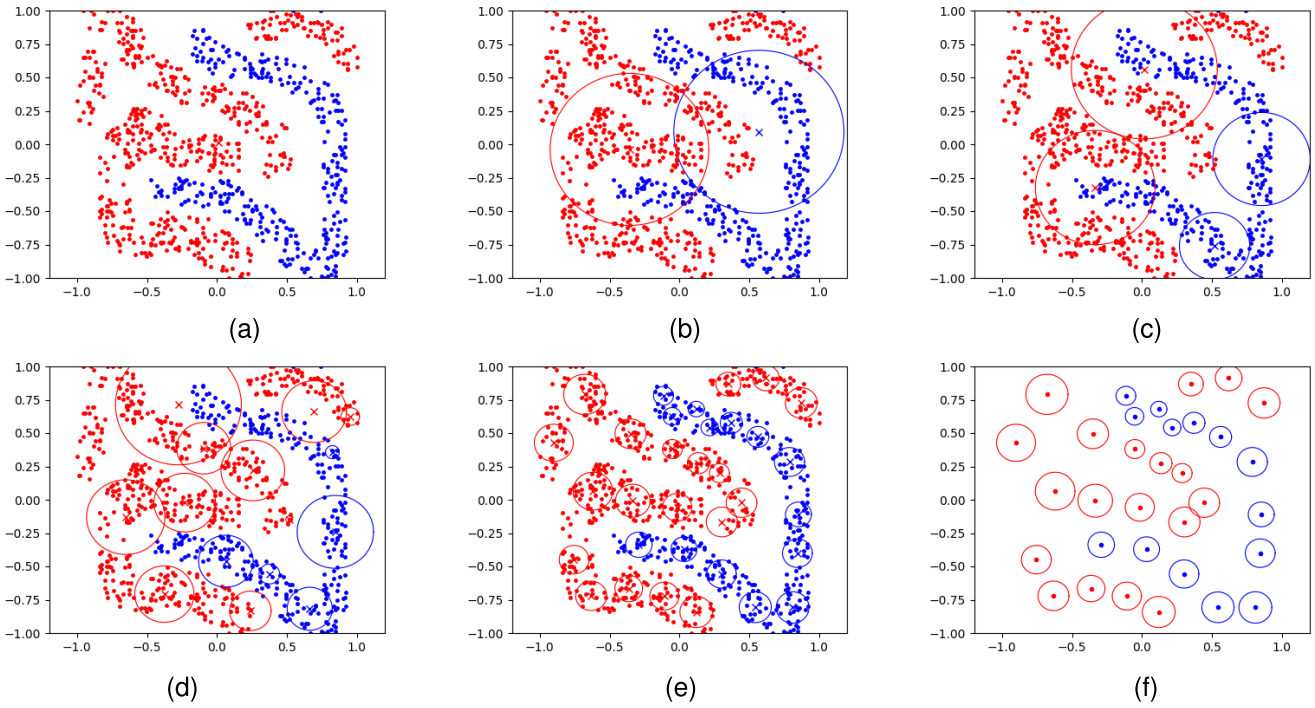

Fig. 2. Existing method for generation of GB and splitting on the "four-class" dataset. Both the red GB and the red points bear the label "+1," while both the blue GBs and the blue points are labeled as "−1." (a) Original dataset. (b) Generated GBs in the first iteration. (c) Generated GBs in the second iteration. (d) Generated GBs in the middle iteration. (e) Results after stop splitting. (f) Extracted GBs.

Let $H = (C_1 \;\; e_1)$, $G = (C_2 \;\; e_2)$, and $u_1 = \binom{w_1}{b_1}$; then, (11) can be reformulated as

$$H^T H u + G^T \alpha = 0,$$
$$\text{i.e.,} \quad u_1 = -\left(H^T H\right)^{-1} G^T \alpha. \tag{12}$$

Computing the inverse of $H^T H$ presents a formidable challenge. However, this difficulty can be effectively addressed by incorporating a regularization term denoted as δI in (12), where $\delta > 0$ is a small quantity and I denotes an identity matrix of appropriate dimensions. Thus,

$$u_1 = -\left(H^T H + \delta I\right)^{-1} G^T \alpha. \tag{13}$$

Using (13) and the above K.K.T. conditions, we can obtain the dual of (2) as follows:

$$\max_{\alpha} \; \alpha^T (e_2 + R_2) - \frac{1}{2} \alpha^T G \left(H^T H + \delta I\right)^{-1} G^T \alpha$$
$$\text{s.t. } 0 \le \alpha \le d_1 e_2. \tag{14}$$

Likewise, the Wolfe dual for (3) can be obtained as

$$\max_{\gamma} \; \gamma^T (e_1 + R_1) - \frac{1}{2} \gamma^T H \left(G^T G + \delta I\right)^{-1} H^T \gamma$$
$$\text{s.t. } 0 \le \gamma \le d_2 e_1. \tag{15}$$

Analogously, $u_2 = \binom{w_2}{b_2}$ corresponding to the -1 class can be calculated by the subsequent equation

$$u_2 = \left(G^T G + \delta I\right)^{-1} H^T \gamma. \tag{16}$$

Once the optimal values of u_1 and u_2 are calculated. The classification of a new input data point x into either the 1 (+1 class) or 2 (−1 class) class can be determined as follows:

$$\text{class}(x) = \arg\min_{i \in \{1,2\}} \frac{\left|w_i^T x + b_i\right|}{\|w_i\|}. \tag{17}$$

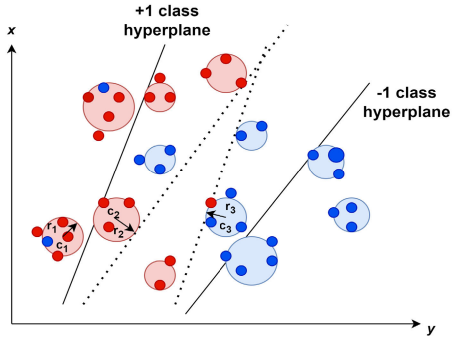

Fig. 3. Geometrical depiction of GBTSVM. Samples in the red symbolize instances of the +1 class, while those in the blue signify instances of the −1 class. In addition, c_i and r_i denote the center and radius of individual GBs.

The formulation of the proposed GBTSVM model for the nonlinear case is given in Section S.II-B in the Supplementary Material.

IV. Proposed Large-Scale Granular Ball Twin Support Vector Machine (LS-GBTSVM)

The GBTSVM formulation exhibits certain limitations: 1) performing a matrix inversion computation within the Wolfe dual formulation becomes costly when dealing with large datasets and 2) GBTSVM does not incorporate the SRM principle in its formulation, which leads to an elevated risk of overfitting. Our goal is to tailor our model for

large-scale datasets by reformulating the optimization problem of GBTSVM to remove the need for computing extensive matrix inverses in (13)–(16). To achieve this, we introduce a regularization term similar to TBSVM [7] and incorporate an additional equality constraint into the optimization problem. The optimization problem of linear LS-GBTSVM is given as

$$\begin{aligned}\min_{w_1,b_1,\eta_1,\xi_2}\ & \frac{1}{2}d_3\left(\|w_1\|^2+b_1^2\right)+\frac{1}{2}\eta_1^T\eta_1+d_1e_2^T\xi_2\\ \text{s.t. }& C_1w_1+e_1b_1=\eta_1,\\ & -(C_2w_1+e_2b_1)+\xi_2\geq e_2+R_2,\quad \xi_2\geq 0,\end{aligned}\tag{18}$$

and

$$\begin{aligned}\min_{w_2,b_2,\eta_2,\xi_1}\ & \frac{1}{2}d_4\left(\|w_2\|^2+b_2^2\right)+\frac{1}{2}\eta_2^T\eta_2+d_2e_1^T\xi_1\\ \text{s.t. }& C_2w_2+e_2b_2=\eta_2,\\ & (C_1w_2+e_1b_2)+\xi_1\geq e_1+R_1,\quad \xi_1\geq 0,\end{aligned}\tag{19}$$

where d_1, d_2, d_3, d_4 (>0) are tunable parameters, ξ_1 and ξ_2 are the slack variables, and e_1 and e_2 are the vector of ones with appropriate dimensions. The optimization problem (18) and (19) incorporates regularization terms $(1/2)d_3(\|w_1\|^2+b_1^2)$ and $(1/2)d_4(\|w_2\|^2+b_2^2)$. The inclusion of the regularization terms in the LS-GBTSVM model contributes to SRM. The modified Lagrangian is designed in a way that bypasses the need to calculate the large matrix inverse. The dual of (18) is obtained as

$$\begin{aligned}&\min_{\alpha_1,\beta_1}\ \frac{1}{2}\begin{pmatrix}\alpha_1^T & \beta_1^T\end{pmatrix}\tilde{P}\begin{pmatrix}\alpha_1\\ \beta_1\end{pmatrix}-d_3\beta_1^T(e_2+R_2)\\ &\text{s.t. } 0\leq\beta_1\leq d_1e_2,\quad \text{where}\\ &\tilde{P}=\begin{pmatrix}C_1C_1^T+d_3I & C_1C_2^T\\ C_2C_1^T & C_2C_2^T\end{pmatrix}+E,\end{aligned}\tag{20}$$

where matrix E consists entirely of ones, while I denotes the identity matrix of the suitable dimension.

The corresponding dual problems (19) can be obtained as follows:

$$\begin{aligned}&\min_{\alpha_2,\beta_2}\ \frac{1}{2}\begin{pmatrix}\alpha_2^T & \beta_2^T\end{pmatrix}\tilde{Q}\begin{pmatrix}\alpha_2\\ \beta_2\end{pmatrix}-d_4\beta_2^T(e_1+R_1)\\ &\text{s.t. } 0\leq\beta_2\leq d_2e_1,\quad \text{where}\\ &\tilde{Q}=\begin{pmatrix}C_2C_2^T+d_4I & C_2C_1^T\\ C_1C_2^T & C_1C_1^T\end{pmatrix}+E.\end{aligned}\tag{21}$$

The optimal values of $\alpha_1, \alpha_2, \beta_1$, and β_2 are used to determine using equations of nonparallel hyperplanes $w_1^Tx+b_1=0$ and $w_2^Tx+b_2=0$ and is given by:

$$\begin{aligned}w_1^*&=-\frac{1}{d_3}\left(C_1^T\alpha_1+C_2^T\beta_1\right),\quad b_1^*=-\frac{1}{d_3}\left(e_1^T\alpha_1+e_2^T\beta_1\right),\\ w_2^*&=\frac{1}{d_4}\left(C_2^T\alpha_2+C_1^T\beta_2\right),\quad b_2^*=\frac{1}{d_4}\left(e_1^T\alpha_2+e_2^T\beta_2\right),\end{aligned}$$

where $*$ represents the optimal values for the corresponding entities. Once the optimal planes are determined, the labeling of the test sample x into either the 1 ($+1$ class) or 2 (-1 class) class can be determined as follows:

$$\text{class}(x)=\arg\min_{i\in\{1,2\}}\frac{\left|w_i^{*T}x+b_i^*\right|}{\|w_i^*\|}.\tag{22}$$

The detailed mathematical formulation of the proposed LS-GBTSVM model for the linear and nonlinear case is given in Sections S.II-A and S.II-C of the Supplementary Material.

V. Time Complexity and Algorithm

Let D represents the training dataset having n samples. The time complexity of the k-means algorithm is $\mathcal{O}(nkt)$ [28], where the parameter k represents the number of clusters and t denotes the number of iterations. Also, the time complexity of standard TSVM [5] is $\mathcal{O}(n^3/4)$. We consider the training dataset D as the initial GB set. The set GB is divided into two GBs using the two-means clustering. During the initial phase of splitting, the computational complexity is $\mathcal{O}(2n)$. In the second phase, the two generated GBs are further divided into four GBs (if both the GBs are impure), with a maximum computational complexity of $\mathcal{O}(2n)$ and so on. If there is a total t number of iterations, then the overall computational complexity of generating GBs is (or less than) $\mathcal{O}(2nt)$. The complexity of the proposed GBTSVM model is (or less than) $\mathcal{O}(p^3/4)+\mathcal{O}(2nt)$, where p represents the total number of generated GBs. Also, $\mathcal{O}(p^3)\ll\mathcal{O}(n^3)$ as $p\ll n$. Therefore, we compare our proposed GBTSVM model with the standard TSVM model given as follows: $\mathcal{O}(p^3/4)+\mathcal{O}(n)\ll\mathcal{O}(n^3/4)$. The time complexity of the proposed GBTSVM model is much lower than TSVM, and hence, the proposed GBTSVM model is more efficient than the baseline models. The algorithm of the proposed GBTSVM is briefly described in Algorithm 1.

VI. Violation Tolerance Upper Bound

In this section, we discuss the violation tolerance upper bound (VTUB) for the proposed GBTSVM model. For GBTSVM, the slack variables can be interpreted as the tolerance for violations from the boundary hyperplane. A naive but reasonable perspective is that the closer two samples are, the more similar their violation tolerances will be [29]. Here, we present the statement of main Theorems 1 and 2 for $+1$ and -1 GB samples. Their proofs are given in Section S.IV of the Supplementary Material.

Let S be a set of $p=p_1+p_2$ GBs, where p_1 and p_2 represent the number of $+1$ and -1 GBs, respectively. Let $H=(C_1,e_1)$ and $G=(C_2,e_2)$, where C_1 and C_2 are matrices of $+1$ and -1 GB centers, respectively.

Theorem 1: If $(\hat{w}_1^T,\hat{b}_1^T,\hat{\xi}_2^T)$ is the optimal solution of the GBTSVM (14), then, for any two positive GBs, $((c_i,r_i),y_i)$ and $((c_j,r_j),y_j)$, the estimations of the corresponding slack variables ξ_2^i and ξ_2^j satisfy

$$\left|\hat{\xi}_2^i-\hat{\xi}_2^j\right|\leq\Delta^2(\delta+\tau_1)(\delta+\tau_1+\tau_2)\sqrt{\kappa}\|G\|_F\cdot d_{ij}^3,\tag{23}$$

where $d_{ij}=\|c_i-c_j\|$ is the distance between GB centers c_i and c_j; $\|G\|_F$ is the Frobenius norm of the matrix G; Δ (>0) and δ (>0) are very small real numbers; κ is a positive real number; and τ_1 and τ_2 correspond to the largest eigenvalues of H^TH and G^TG, respectively.

Theorem 2: If $(\hat{w}_2^T,\hat{b}_2^T,\hat{\xi}_1^T)$ is the optimal solution of the GBTSVM (15), then, for any two negative GBs $((c_i,r_i),y_i)$ and $((c_j,r_j),y_j)$, the estimations of the corresponding slack variables ξ_1^i and ξ_1^j satisfy

$$\left|\hat{\xi}_1^i-\hat{\xi}_1^j\right|\leq\Delta^2(\delta+\tau_2)(\delta+\tau_1+\tau_2)\sqrt{\kappa}\|H\|_F\cdot d_{ij}^3.\tag{24}$$

Algorithm 1 Algorithm of GBTSVM Model

Input: Traning dataset D, and the threshold purity T.
Output: GBTSVM classifier.

1: Initialize the entire dataset D as a granular ball GB and set of granular balls S to be empty set, i.e., GB $= D$ and $S = \{\}$.
2: $Dummy = \{\text{GB}\}$.
3: $for\ i = 1 : |Dummy|$
4: $if\ pur(\text{GB}_i) < T$
5: Split GB_i into GB_{i1} and GB_{i2}, using 2-means clustering algorithm.
6: $Dummy \leftarrow \text{GB}_{i1}, \text{GB}_{i2}$.
7: $end\ if$.
8: $else\ pur(\text{GB}_i) \geq T$
9: Calculate the center $c_i = \frac{1}{n_i}\sum_{j=1}^{n_i} x_j$ of GB_i, where $x_j \in \text{GB}_i$, $j = 1, 2, \ldots, n_i$, and n_i is the number of training sample in GB_i.
10: Calculate the radius $r_i = \frac{1}{n_i}\sum_{j=1}^{n_i}|x_j - c_i|$ of GB_i.
11: Calculate the label y_i of GB_i, where y_i is assigned the label of majority class samples within GB_i.
12: Put $\text{GB}_i = \{((c_i, r_i), y_i)\}$ in S.
13: $end\ else$.
14: $end\ for$.
15: $if\ Dummy \neq \{\}$
16: Go to step 3 (for further splitting).
17: $end\ if$.
18: Set $S = \{\text{GB}_i,\ i = 1, 2, \ldots, p\} = \{((c_i, r_i), y_i),\ i = 1, 2, \ldots, p\}$, where c_i signifies the center, r_i indicates the radius, y_i is the label of GB_i and p is the number of generated granular balls.
19: Solve (14) and (15) to obtain α and γ, where α and γ are the Lagrange multipliers.
20: Using (13) and (16), find optimal values of w_1, b_1, w_2, and b_2.
21: Testing sample is classified into $+1$ or -1 class using (17).

The derived VTUB, as established in Theorems 1 and 2, is directly proportional to the distance between two training samples in the GBTSVM. This means that the closer the two training samples are, the more similar their corresponding tolerances from the GBTSVM will be. Consequently, if two training samples are identical, their corresponding tolerances will also be identical. These findings align with our intuitive understanding.

VII. Experimental Results

To test the efficacy and efficiency of proposed models, i.e., GBTSVM and LS-GBTSVM, we compare them to baseline models on publicly available UCI [25] and KEEL [26] benchmark datasets under different noise levels. Furthermore, we conduct experiments on datasets generated by the NDC Data Generator [27]. In the Supplementary Material, we conduct a series of sensitivity analyses on various aspects of the proposed models. This includes investigating the impact of GB parameters num and pur on the proposed models in Section S.V-A, assessing different levels of label noise on both proposed and baseline models in Section S.V-B, and conducting sensitivity analyses on hyperparameters σ and pur in Section S.V.C. In addition, sensitivity analyses on hyperparameters d_1 and d_2 are presented in Section S.V-D, and the relationship between the number of GBs generated and the performance of the proposed GBTSVM model with different purities is analyzed in Section S.V-E.

A. Experimental Setup

The hardware environment for the experiment consists of a PC equipped with an Intel[1] Xeon[1] Gold 6226R CPU at 2.90-GHz CPU and 128-GB RAM running on Windows 11 possessing Python 3.11. The dual QPPs arising in GBTSVM, LS-GBTSVM, and baseline models are solved using the "qpsolvers" function in the CVXOPT package. Also, we use the SOR algorithm [30], [31] for solving LS-GBTSVM's dual problems, and the proposed model is named LS-GBTSVM (SOR). The dataset is randomly divided into training and testing sets in a 70:30 ratio, respectively. We use a fivefold cross validation and grid search approach to optimize the models' hyperparameters from the following ranges: $d_i = \{10^{-5}, 10^{-4}, \ldots, 10^5\}$ for $i = 1, 2, 3, 4$. For the nonlinear case, we use the Gaussian kernel and is given by $K(x_i, x_j) = e^{(-1/2\sigma^2)\|x_i - x_j\|^2}$. The Gaussian kernel parameter σ is selected from the range $\{2^{-5}, 2^{-4}, \ldots, 2^5\}$. In LS-GBTSVM, we adopt equal penalty parameters, i.e., $d_1 = d_2$ and $d_3 = d_4$, for both linear and nonlinear cases. Furthermore, we compare the results on NDC-based large-scale datasets. We assign a penalty parameter equal to 1 to the proposed and the baseline models, i.e., $(d_1 = d_2 = d_3 = d_4 = 1)$.

B. Experiments on Real World UCI and KEEL Datasets

In this section, we present an intricate analysis involving a comparison of the proposed GBTSVM, LS-GBTSVM and LS-GBTSVM (SOR) with SVM [1], GBSVM [24], and TSVM [5] models on 36 UCI and KEEL benchmark datasets. The optimization problem of GBSVM is solved by the PSO algorithm [24] and named GBSVM (PSO). The detailed experimental results are presented in Tables S.III and S.IV of the Supplementary Material. All the experimental results discussed in this section are obtained at a 0% noise level for both linear and nonlinear cases. The comparison in terms of accuracy (ACC) indicates that our proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models yield better performance than the baseline SVM, GBSVM (PSO), and TSVM models on most of the datasets. From Table I, the average ACCs of the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models are 85.30%, 85.32%, and 81.34%, respectively, whereas the average ACCs of SVM, GBSVM (PSO), and TSVM models are 82.21%, 72.46%, and 65.47%, respectively. The average ACCs of the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models surpass those of the baseline models. As the average ACC can be influenced by exceptional performance in one dataset that compensates for losses across multiple datasets, it might be a biased measure. Therefore, we employ the

[1]Registered trademark.

TABLE I

AVERAGE ACCURACY (ACC) AND AVERAGE RANK OF THE PROPOSED GBTSVM AND LS-GBTSVM ALONG WITH THE BASELINE MODELS OVER FOR UCI AND KEEL DATASETS WITH LINEAR KERNEL

	Noise	SVM [1]	GBSVM (PSO) [24]	TSVM [5]	GBTSVM†	LS-GBTSVM†	LS-GBTSVM (SOR)†
Average ACC (Rank)	0 %	82.21 (3.46)	72.46 (4.79)	65.47 (5.69)	85.30 (1.97)	**85.32** (1.94)	81.34 (3.14)
	5 %	82.11 (3.74)	75.92 (4.78)	82.27 (3.88)	**83.75** (2.83)	82.96 (2.68)	82.49 (3.10)
	10 %	83.06 (3.65)	75.36 (4.97)	84.64 (3.10)	**85.82** (2.42)	82.11 (3.28)	82.25 (3.58)
	15 %	81.25 (3.44)	74.44 (4.94)	81.11 (3.53)	**85.30** (2.26)	81.50 (3.44)	80.67 (3.38)
	20 %	82.41 (3.35)	74.89 (4.90)	82.46 (3.40)	**84.86** (2.99)	83.11 (3.19)	82.75 (3.17)

† represents the proposed models. Bold text denotes the model with the highest average ACC.

ranking method to gauge the effectiveness and appraise the performance of the models. Here, each classifier is assigned a ranking, with the superior-performing model receiving a lower rank, while the model with inferior performance is given a higher rank. For evaluation of q models across N datasets, the rank of the jth model on the ith dataset can be denoted as $\mathfrak{R}_j^i$. Then, the model's average rank is given by $\mathfrak{R}_j = (1/N)\sum_{i=1}^{N}\mathfrak{R}_j^i$. The average ranks of SVM, GBSVM (PSO), TSVM, GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) are 3.46, 4.79, 5.69, 1.97, 1.94, and 3.14, respectively. It is evident that the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) have the best average rank. Hence, the generalization ability of the proposed GBTSVM and LS-GBTSVM models are superior compared to the baseline models. Now, we conduct the Friedman test [32] to determine whether the models have significant differences. Under the null hypothesis of the Friedman test, it is presumed that all the models exhibit an equal average rank, signifying equal performance. The Friedman test adheres to the chi-squared distribution (χ_F^2) with $(q-1)$ degree of freedom (d.o.f) and its calculation involves: $\chi_F^2 = (12N/(q(q+1)))[\sum_j \mathfrak{R}_j^2 - ((q(q+1)^2)/4)]$. The F_F statistic is calculated as: $F_F = ((N-1)\chi_F^2)/(N(q-1)-\chi_F^2)$, where the F-distribution has $(q-1)$ and $(N-1)\times(q-1)$ d.o.f. For $q=6$ and $N=36$, we get $\chi_F^2 = 116.19$ and $F_F = 63.70$ at 5% level of significance. Referring to the statistical F-distribution table, $F_F(5,175) = 2.2657$. Since $63.70 > 2.2657$, we reject the null hypothesis. As a result, there exists a statistical distinction among the models being compared. Next, we employ the Nemenyi post hoc test to examine the pairwise distinctions between the models. The value of the critical difference (C.D.) is evaluated as C.D. $= q_\alpha((q(q+1))/6N)^{1/2}$, where q_α represents the critical value from the distribution table for the two-tailed Nemenyi test. According to the statistical F-distribution table, $q_\alpha = 2.850$ at 5% significance level, the C.D. is calculated to be 1.256. The average rank disparities between the proposed [GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR)] models with SVM, GBSVM (PSO), and TSVM are $(1.49, 1.52, 0.32)$, $(2.82, 2.85, 1.65)$, and $(3.72, 3.75, 2.55)$ respectively. According to the Nemenyi post hoc test, the proposed models GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) significantly differ from the baseline models except for LS-GBTSVM (SOR) with SVM. It is evident that the proposed GBTSVM and LS-GBTSVM show better performance compared to baseline models. LS-GBTSVM (SOR) is significantly different among the models (except SVM) and LS-GBTSVM (SOR) surpasses SVM in terms of average rank.

TABLE II

PAIRWISE WIN–TIE–LOSS TEST OF ALL THE COMPARED MODELS WITH LINEAR KERNEL

	SVM [1]	GBSVM (PSO) [24]	TSVM [5]	GBTSVM†	LS-GBTSVM†
GBSVM (PSO) [24]	[4, 4, 28]				
TSVM [5]	[0, 0, 36]	[9, 0, 27]			
GBTSVM†	[33, 2, 1]	[35, 0, 1]	[36, 0, 0]		
LS-GBTSVM†	[28, 2, 6]	[33, 1, 2]	[36, 0, 0]	[18, 2, 16]	
LS-GBTSVM (SOR)†	[19, 1, 16]	[29, 0, 7]	[34, 0, 2]	[12, 2, 22]	[5, 5, 26]

† represents the proposed models.

Furthermore, to analyze the models, we use a pairwise win–tie–loss sign test. As per the win–tie–loss sign test, under the null hypothesis, it is assumed that two models perform equivalently and are expected to win in $N/2$ datasets, where N represents the dataset count. If the classification model wins on approximately $(N/2) + 1.96(\sqrt{N}/2)$ datasets, then the model is significantly better. Also, if there is an even count of ties between the two models, these ties are evenly divided between them. However, if the number of ties is odd, we disregard one tie and allocate the remaining ties among the specified classifiers. In this case, when $N = 36$, if one of the models wins is at least 23.88, then there is a significant difference between the models. Table II illustrates the comparative performance of the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models along with the baseline models, presenting their outcomes in terms of pairwise wins, ties, and losses using UCI and KEEL datasets. In Table II, the entry $[x, y, z]$ indicates that the model mentioned in the row wins x times, ties y times, and loses z times in comparison to the model mentioned in the respective column. Table II clearly indicates that the proposed GBTSVM and LS-GBTSVM model exhibits significant superiority compared to the baseline models. Moreover, the proposed LS-GBTSVM (SOR) model achieves a statistically significant difference from GBSVM (PSO) and TSVM. Demonstrating a significant level of performance, the LS-GBTSVM (SOR) model succeeds in 19 out of 36 datasets. Therefore, the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models are significantly superior compared to the existing models.

For the nonlinear case, the average ACC and rank values are shown for the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) with SVM, GBSVM, and TSVM in Table III. From the table, it is evident that the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) outperform the baseline models on the majority of the datasets. The average ACCs of proposed GBTSVM, LS-GBTSVM, LS-GBTSVM (SOR), SVM, GBTSVM, and TSVM are 88.74%, 85.85%, 84.93%, 76.27%, 79.43%, and 84.83%, respectively. This demonstrates a clear performance improvement, with our proposed models securing the top position

TABLE III
AVERAGE ACCURACY (ACC) AND AVERAGE RANK OF THE PROPOSED GBTSVM AND LS-GBTSVM ALONG WITH THE BASELINE MODELS OVER FOR UCI AND KEEL DATASETS WITH NONLINEAR KERNEL

	Noise	SVM [1]	GBSVM (PSO) [24]	TSVM [5]	GBTSVM†	LS-GBTSVM†	LS-GBTSVM (SOR)†
Average ACC (Rank)	0 %	76.27 (4.42)	79.43 (4.03)	84.83 (3.5)	**88.74** (2.17)	85.85 (3.31)	84.93 (3.58)
	5 %	77.86 (4.53)	80.7 (4.38)	85.13 (3.56)	**88.61** (2.21)	86.95 (2.86)	84.57 (3.47)
	10 %	77.19 (4.43)	84.05 (3.9)	84.51 (3.81)	**89.25** (2.1)	84.79 (3.33)	84.68 (3.43)
	15 %	79.3 (4.24)	83.24 (4.5)	85.37 (3.28)	**88.49** (2.43)	85.32 (3.21)	85.48 (3.35)
	20 %	81.08 (4.18)	81.76 (4.47)	84.6 (3.44)	**86.97** (2.54)	85.39 (3.1)	85.2 (3.26)

† represents the proposed models. Bold text denotes the model with the highest average ACC.

compared to the baseline models. It can be noted that among all the models, our proposed GBTSVM holds the lowest average rank. Furthermore, we conduct the Friedman statistical test along with Nemenyi post hoc tests. We compute $\chi_F^2 = 30.94$ and $F_F = 7.26$ and $F_F(5, 175) = 2.2657$ at 5% level of significance. Since $F_F(5, 175) < F_F$, therefore we reject the null hypothesis. Moreover, the Nemenyi post hoc test is employed to identify significant differences among the pairwise comparisons. We compute C.D. $= 1.256$ and the average rankings of the models listed in Table III should have a minimum difference by 1.256. The average rank disparities between the proposed [GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR)] models with SVM, GBSVM (PSO), and TSVM are $(2.25, 1.11, 0.84)$, $(1.86, 0.72, 0.45)$, and $(1.33, 0.19, 0.08)$, respectively. Hence, the proposed GBTSVM exhibits significant superiority over the baseline models. The Friedman test did not show the statistical difference between the proposed LS-GBTSVM and LS-GBTSVM (SOR) with baseline SVM, GBTSVM, and TSVM models as well as GBTSVM with SVM. However, Table III unequivocally demonstrates that both the proposed LS-GBTSVM and LS-GBTSVM (SOR) consistently achieve the lowest average rank and higher average ACC in comparison to the baseline models. Also, the proposed GBTSVM has a lower rank when compared to SVM. As a result, the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) outperformed the baseline models. We also conduct a win–tie–loss sign test for nonlinear cases. Table IV shows the pairwise win–tie–loss of the compared models on UCI and KEEL datasets. In our case, if any of the two models wins on at least 23.88 datasets, the two models are statistically different. It is evident from Table IV that the proposed GBTSVM model statistically outperforms the baseline SVM, GBSVM (PSO), and TSVM models. The proposed LS-GBTSVM is statistically better than SVM. In general, from Table IV and the aforementioned analysis, GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models with different performance metrics and statistical tests, it becomes clear that the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models exhibit competitive or even superior performance when compared with the baseline models.

C. Experiments on UCI and KEEL Datasets With Label Noise

To validate the effectiveness and noise resilience of the proposed GBTSVM and LS-GBTSVM models, we contaminate the label noise, including 5%, 10%, 15%, and 20% on each dataset. The comparative experimental results of the proposed

TABLE IV
PAIRWISE WIN–TIE–LOSS TEST OF ALL THE COMPARED MODELS WITH THE NONLINEAR KERNEL

	SVM [1]	GBSVM (PSO) [24]	TSVM [5]	GBTSVM†	LS-GBTSVM†
GBSVM (PSO) [24]	[19, 3, 14]				
TSVM [5]	[24, 2, 10]	[18, 5, 13]			
GBTSVM†	[27, 1, 8]	[27, 1, 8]	[26, 2, 8]		
LS-GBTSVM†	[20, 10, 6]	[22, 1, 13]	[19, 2, 15]	[8, 2, 26]	
LS-GBTSVM (SOR)†	[19, 12, 5]	[22, 2, 12]	[15, 3, 18]	[6, 2, 28]	[3, 25, 8]

† represents the proposed models.

GBTSVM and LS-GBTSVM models along with the baseline models with linear and nonlinear cases are shown in Tables S.III and S.IV in Section S.VI of the Supplementary Material. The average ACC and average rank of the models with the linear kernel are shown in Table I with different percentages of noise labels. The classification ACC of the proposed GBTSVM and LS-GBTSVM models is better than the baseline models. It indicates that the GBTSVM and LS-GBTSVM show better robustness among the compared models because a GB possesses a coarser granularity, which can mitigate the impact of label noise within it. The label assigned to a GB is primarily determined by the predominant label contained within it, and the presence of label noise with minority labels does not significantly influence the determination of the GB. We observe that using the "qpsolvers" function to solve the dual problem of LS-GBTSVM gets more stable classification results than using the SOR algorithm in LS-GBTSVM on most of the dataset. The average ACC of LS-GBTSVM is higher than that of LS-GBTSVM (SOR). Furthermore, the average rank of LS-GBTSVM under 0% and 5% label noise of 1.94 and 2.68, respectively, which is the lowest rank, and the average rank of GBTSVM under 0% and 5% label noise of 2.97 and 2.83, respectively, is the second lowest. Similarly, GBTSVM has the lowest rank under 10%, 15%, and 20% label noise. In conclusion, GBTSVM and LS-GBTSVM consistently outperform the compared models across various levels of label noise. The sensitivity analysis of the proposed GBTSVM model, considering different levels of label noise, is presented in Section S.V-B of the Supplementary Material.

D. Experiment on Artificial NDC Datasets

To showcase the superiority of the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) models in terms of training speed and scalability, we perform experiments using the NDC datasets [27]. In this experiment, the NDC datasets are generated with varying sizes, ranging from 10k to 5m while keeping the number of features constant to 32. Table V presents the ACC and training time of the compared models on the NDC datasets. The results show that the

TABLE V
TESTING ACCURACY (ACC) AND TRAINING TIME OF CLASSIFIERS ON NDC DATASETS WITH LINEAR KERNEL

NDC datasets	SVM [1] ACC(%) (Time(s))	GBSVM (PSO) [24] ACC(%) (Time(s))	TSVM [5] ACC(%) (Time(s))	GBTSVM† ACC(%) (Time(s))	LS-GBTSVM† ACC(%) (Time(s))	LS-GBTSVM (SOR)† ACC(%) (Time(s))
NDC-10k	80.64 (309.0300)	52.43 (1044.2187)	**86.59** (209.606)	81.44 (0.1562)	83.89 (0.5472)	79.59 (0.2343)
NDC-50k	79.42 (809.5466)	53.41 (2478.1415)	**86.21** (715.689)	80.84 (0.578)	83.42 (2.5467)	79.9 (1.2856)
NDC-1l	b	a	b	**85.77** (0.3562)	84.17 (3.6237)	74.6 (1.5005)
NDC-3l	b	a	b	80.52 (0.906)	**82.97** (3.7371)	73.41 (4.1415)
NDC-5l	b	a	b	81.12 (1.5326)	**82.64** (5.7185)	75.65 (5.8356)
NDC-1m	b	a	b	79.42 (2.7863)	78.9 (6.9668)	**79.98** (12.3445)
NDC-3m	b	a	b	78.04 (8.1229)	**84.73** (8.1191)	79.54 (14.6126)
NDC-5m	b	a	b	**78.68** (12.8752)	77.54 (24.0667)	73.65 (32.146)

[a] Experiment is terminated because of the out of bound issue shown by PSO algorithm. [b] Terminated because of out of memory.
† represents the proposed models. Bold text denotes the model with the highest average ACC.

proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) are more efficient among the baseline models. The following issues arise in the baseline models while handling large-scale datasets: 1) training of TSVM demands substantial memory consumption to compute matrix inversion; as a result, an "out-of-memory" issue occurs when the scale reaches 1l; 2) training SVM requires solving a QPP, which requires significant computational resources; as the dataset size increases, the time required for training and prediction can become prohibitive; and 3) the PSO algorithm in the GBSVM (PSO) is halted due to the emergence of an out-of-bounds issue. The results of the experiments demonstrate that the proposed GBTSVM, LS-GBTSVM, and LS-GBTSVM (SOR) exhibit efficiency several hundreds or even thousands of times faster than the compared models. This is due to the fact that the count of generated GBs on a dataset is significantly lower compared to the total number of samples.

VIII. CONCLUSION

In this article, we proposed a novel GBTSVM as a solution to the challenges faced by TSVM. GBTSVM utilized the coarse granularity of GBs for input, leading to two nonparallel hyperplanes for sample classification. The proposed GBTSVM mitigates the impact of noise and outliers while also eliminating the overhead of higher computational costs typically associated with standard SVM, TSVM, and their variants. We again proposed a novel LS-GBTSVM by incorporating a regularization term in the primal optimization formulation to implement the SRM principle. LS-GBTSVM's key advantage lies in avoiding matrix inversions, which makes it suitable for large-scale problems, and effectively addressing overfitting concerns.

To demonstrate the effectiveness, robustness, scalability, and efficiency of the proposed GBTSVM and LS-GBTSVM models, we conducted a series of rigorous experiments and subjected them to comprehensive statistical analyses. Our experimental results, encompassing 36 UCI and KEEL datasets (with and without label noise), were subjected to a series of statistical tests. The experimental results, along with the statistical analyses, indicate that the proposed linear and nonlinear GBTSVM and LS-GBTSVM models beat baseline approaches in efficiency and generalization performance. Here are the key findings: 1) the proposed models exhibit an average ACC improvement of up to 20% in comparison to the baseline models for the linear case; 2) our models have demonstrated an up to 12% increase in average ACC compared to the baseline models in noisy conditions, showcasing exceptional resilience when contrasted with the baseline models for nonlinear cases; 3) we tested the models on large-scale NDC datasets from 10k to 5m samples. Baseline models faced memory issues beyond NDC-50k, but our proposed models excelled, demonstrating scalability and efficiency on large-scale datasets; and 4) we conducted a series of sensitivity analyses to understand the behavior of hyperparameters of the proposed models. The key hyperparameters under investigation include the GB parameters num, pur, σ, different levels of label noise, and d_1 and d_2 with different-different combinations. Our proposed models have showcased outstanding performance in binary classification problems. An essential avenue for future research would involve adapting and extending these models to address the complexities associated with multiclass classification scenarios.

REFERENCES

[1] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, Sep. 1995.
[2] D. Bollegala, Y. Matsuo, and M. Ishizuka, "A web search engine-based approach to measure semantic similarity between words," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 7, pp. 977–990, Jul. 2011.
[3] B. Richhariya, M. Tanveer, and A. H. Rashid, "Diagnosis of Alzheimer's disease using universum support vector machine based recursive feature elimination (USVM-RFE)," *Biomed. Signal Process. Control*, vol. 59, May 2020, Art. no. 101903.
[4] O. L. Mangasarian and E. W. Wild, "Multisurface proximal support vector machine classification via generalized eigenvalues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 69–74, Jan. 2006.
[5] R. Khemchandani and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007.
[6] M. Tanveer, T. Rajani, R. Rastogi, Y. H. Shao, and M. A. Ganaie, "Comprehensive review on twin support vector machines," *Ann. Oper. Res.*, pp. 1–46, Mar. 2022.
[7] Y. H. Shao, C. H. Zhang, X. B. Wang, and N. Y. Deng, "Improvements on twin support vector machines," *IEEE Trans. Neural Netw.*, vol. 22, no. 6, pp. 962–968, May 2011.
[8] Y. Tian, X. Ju, Z. Qi, and Y. Shi, "Improved twin support vector machine," *Sci. China Math.*, vol. 57, no. 2, pp. 417–432, Feb. 2014.
[9] M. Tanveer, A. Sharma, and P. N. Suganthan, "General twin support vector machine with pinball loss function," *Inf. Sci.*, vol. 494, pp. 311–327, Aug. 2019.
[10] Y. Xu, Z. Yang, and X. Pan, "A novel twin support-vector machine with pinball loss," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 359–370, Feb. 2017.
[11] M. A. Kumar and M. Gopal, "Least squares twin support vector machines for pattern classification," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7535–7543, May 2009.
[12] M. Tanveer, M. A. Khan, and S.-S. Ho, "Robust energy-based least squares twin support vector machines," *Appl. Intell.*, vol. 45, no. 1, pp. 174–186, Jul. 2016.

[13] C.-F. Lin and S.-D. Wang, "Fuzzy support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 464–471, Mar. 2002.

[14] S. Rezvani, X. Wang, and F. Pourpanah, "Intuitionistic fuzzy twin support vector machines," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 11, pp. 2140–2151, Nov. 2019.

[15] Z. Liang and L. Zhang, "Intuitionistic fuzzy twin support vector machines with the insensitive pinball loss," *Appl. Soft Comput.*, vol. 115, Jan. 2022, Art. no. 108231.

[16] S. Xia, X. Dai, G. Wang, X. Gao, and E. Giem, "An efficient and adaptive granular-ball generation method in classification problem," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 5319–5331, Apr. 2024.

[17] S. Xia, Y. Liu, X. Ding, G. Wang, H. Yu, and Y. Luo, "Granular ball computing classifiers for efficient, scalable and robust learning," *Inf. Sci.*, vol. 483, pp. 136–152, May 2019.

[18] S. Xia et al., "A fast adaptive k-means with no bounds," *IEEE Trans. Pattern Anal. Mach. Intell.*, Jan. 2020, doi: 10.1109/TPAMI.2020.3008694.

[19] S. Xia, S. Zheng, G. Wang, X. Gao, and B. Wang, "Granular ball sampling for noisy label classification or imbalanced classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 4, pp. 2144–2155, Apr. 2023.

[20] X. Zhang, H. Gou, Z. Lv, and D. Miao, "Double-quantitative distance measurement and classification learning based on the tri-level granular structure of neighborhood system," *Knowl.-Based Syst.*, vol. 217, Apr. 2021, Art. no. 106799.

[21] W. Pedrycz and K.-C. Kwak, "The development of incremental models," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 3, pp. 507–518, Jun. 2007.

[22] W. Pedrycz, "Identification in fuzzy systems," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-14, no. 2, pp. 361–366, Mar. 1984.

[23] A. Song, G. Wu, W. Pedrycz, and L. Wang, "Integrating variable reduction strategy with evolutionary algorithms for solving nonlinear equations systems," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 1, pp. 75–89, Jan. 2021.

[24] S. Xia, X. Lian, G. Wang, X. Gao, J. Chen, and X. Peng, "GBSVM: An efficient and robust support vector machine framework via granular-ball computing," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jul. 2, 2024, doi: 10.1109/TNNLS.2024.3417433.

[25] D. Dua and C. Graff. (2017). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[26] J. Derrac et al., "KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Multiple-Valued Logic Soft Comput.*, vol. 17, no. 2, pp. 255–287, 2015.

[27] D. R. Musicant. (1998). *NDC: Normally Distributed Clustered Datasets*. [Online]. Available: www.cs.wisc.edu/dmi/SVM/ndc/

[28] Y. Zhou, H. Yu, and X. Cai, "A novel k-Means algorithm for clustering and outlier detection," in *Proc. 2nd Int. Conf. Future Inf. Technol. Manage. Eng.*, Dec. 2009, pp. 476–480.

[29] K. Qi and H. Yang, "Elastic net nonparallel hyperplane support vector machine and its geometrical rationality," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 7199–7209, Dec. 2022.

[30] Z.-Q. Luo and P. Tseng, "Error bounds and convergence analysis of feasible descent methods: A general approach," *Ann. Oper. Res.*, vols. 46–47, no. 1, pp. 157–178, Mar. 1993.

[31] O. L. Mangasarian and D. R. Musicant, "Successive overrelaxation for support vector machines," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1032–1037, Sep. 1999.

[32] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006.

A. Quadir (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree with the Department of Mathematics, Indian Institute of Technology Indore, India, under the supervision of Prof. M. Tanveer. His research interests include support vector machines, support matrix machines, loss functions for classification and regression, and their application to the biomedical domain.

M. Sajid (Graduate Student Member, IEEE) received the B.Sc. degree (Hons.) in mathematics from St. Xavier's College, Ranchi, India, in 2019, and the M.Sc. degree in mathematics from IIT Delhi, Delhi, India, in 2021. He is currently pursuing the Ph.D. degree with the Department of Mathematics, IIT Indore, Indore, India, under the supervision of Prof. M. Tanveer.

His research interests include randomized neural networks, graph neural networks, and alzheimer's disease diagnosis via machine and deep learning techniques.

Mr. Sajid is the Founding Chair of the IEEE CIS Student Branch Chapter at IIT Indore and the Vice Chair of the IEEE Student Branch at IIT Indore.

M. Tanveer (Senior Member, IEEE) received the Ph.D. degree in computer science from JNU, New Delhi, India.

He is a Professor and Ramanujan Fellow in Mathematics at the IIT Indore. Previously, he was a Postdoctoral Research Fellow at the Rolls-Royce@NTU Corporate Laboratory, NTU, Singapore. He has published over 150 journal papers, with over 6400 citations and an h-index of 40 (Google Scholar, June 2024). He is listed among the world's top 2% scientists by Stanford University. His research interests span support vector machines, optimization, machine learning, deep learning, and their applications to alzheimer's disease and dementia.

Dr. Tanveer has served on review boards for over 100 scientific journals and on committees for various conferences. His awards include the 2023 INNS Aharon Katzir Young Investigator Award, 2023 IIT Indore Best Research Paper Award, 2022 Asia Pacific Neural Network Society (APNNS) Young Researcher Award, and the 2016 DST-Ramanujan Fellowship in Mathematical Sciences. He is/was an Associate/Action/Guest Editor in several prestigious journals such as IEEE (TFS, TNNLS, TETCI) and Elsevier (*Pattern Recognition, Neurocomputing, Neural Networks, Engineering Applications of AI*). He has co-edited a book on machine intelligence and signal analysis and has organized numerous conferences and sessions. He is the Principal Investigator or Co-PI of 12 major research projects funded by the Government of India. He is also an Elected Board of Governors of the APNNS, an Elected INSA Associate Fellow and IEEE CIS Distinguished Lecturer (2024–2026). Recently, he received the INSA Distinguished Lecture Award.