Learning Complex Rare Categories with Dual Heterogeneity

Pei Yang∗ Jingrui He† Jia-Yu Pan‡

Abstract
In the era of big data, it is often the case that the self-similar rare categories in a large data set are of great importance, such as the malicious insiders in big organizations, and the IC devices with defects in semiconductor manufacturing. Furthermore, such rare categories often exhibit multiple types of heterogeneity, such as the task heterogeneity, which originates from data collected in multiple domains, and the view heterogeneity, which originates from multiple information sources. Existing methods for learning rare categories mainly focus on the homogeneous settings, i.e., a single task and a single view. In this paper, for the first time, we study complex rare categories with both task and view heterogeneity, and propose a novel optimization framework named $M^2LID$. It introduces a boundary characterization metric to capture the sharp changes in density near the boundary of the rare categories in the feature space, and constructs a graph-based model to leverage both task and view heterogeneity. Furthermore, $M^2LID$ integrates them in a way of mutual benefit. We also present an effective algorithm to solve this framework, analyze its performance from various aspects, and demonstrate its effectiveness on both synthetic and real datasets.

1 Introduction
Many real world applications involve large amount of data. However, it is often the case that the target examples of great interest to us are both rare and self-similar. These target examples are collectively called rare categories, such as the malicious insiders in big organizations consisting of tens of thousands of employees, and the IC devices with defects in semiconductor manufacturing. Furthermore, the data in such applications often exhibit multiple types of heterogeneity. Take malicious insiders as an example. The data collected from multiple related organizations (e.g., multiple financial institutes) correspond to the task heterogeneity, and the multiple information sources for characterizing the behaviors of employees (e.g., emails, social networks, website browsing history) correspond to the view heterogeneity. Existing methods for analyzing rare categories mainly focus on the homogeneous settings. In other words, they assume that the data come from a single domain and described based on a single information source [14, 17, 25, 15]. Therefore, they are not best suited for analyzing the complex rare categories with multiple types of heterogeneity.

To address this problem, in this paper, for the first time, we study complex rare categories with both task and view heterogeneity. Here we are facing two major challenges. The first one is the rarity challenge: given the small percentage of examples from the rare categories, how can we effectively detect and characterize them? The second one is the dual heterogeneity challenge: how can we leverage both task and view heterogeneity to maximally boost the performance of rare category analysis?

To answer these questions, we propose a novel framework named $M^2LID$ for Multi-task Multi-view Learning on Imbalanced Data. First, to address the rarity challenge, in $M^2LID$, we propose to use a novel metric named border-degree to capture the sharp changes in density near the boundary of rare categories in the feature space. It is based on the different properties between $k$-nearest neighbors (KNN) and reverse $k$-nearest neighbors (RKNN). Second, to address the dual heterogeneity challenge, in $M^2LID$, we construct a graph-based model to leverage both task and view heterogeneity. To be specific, we model task relatedness by requiring the task-specific learners to behave similarly on the features, and we model view consistency by requiring the view-based learners to behave similarly on the examples. Finally, we integrate them into an optimization framework in a way of mutual benefit based on the following intuition: dual heterogeneity helps characterize the boundary of rare categories more accurately, which in turn helps model the dual heterogeneity and improve the learning performance of multiple tasks. Based on this framework, we propose an iterative algorithm using block coordinate descent to repeatedly update the boundary characteristics of rare categories and multiple classification functions for different tasks. We also analyze its performance in terms of the convergence property, the error bound, and the algorithm complexity.

The major contributions of this paper can be summarized as follows.

∗The City University of New York.
†Arizona State University.
‡Google Inc.
1. An effective metric for characterizing boundaries of rare categories;
2. A novel optimization framework $M^2LID$ for modeling the rarity and dual heterogeneity properties in a way of mutual benefit;
3. Performance analysis with respect to the convergence property, the error bound, and the algorithm complexity of the proposed algorithm;
4. Experimental results on a variety of data sets demonstrating the effectiveness of the proposed algorithm.

The rest of the paper is organized as follows. After reviewing the related work in Section 2, we present the proposed $M^2LID$ framework in Section 3 and analyze its performance in Section 4, followed by experimental results in Section 5. Finally, we conclude in Section 6.

2 Related Work

We review the related work on imbalanced classification, rare category analysis, and dual-heterogeneity learning.

2.1 Imbalanced Classification and Rare Category Analysis

Imbalanced data correspond to data sets exhibiting significant, and in some cases extreme, imbalances. Researchers have proposed many methods to address this problem [5], such as sampling methods [20, 6], ensemble based methods [30]. However, these methods might not take full advantage of the minority class properties (e.g., the compactness property) [17], as well as multiple types of heterogeneity.

Outlier detection aims to find patterns that do not conform to expected behavior [4]. According to [4], outlier detection techniques can be categorized into classification based, nearest neighbor based, clustering based, information theoretic, spectral, and statistical techniques. They usually assume that the outliers are separable from normal data points.

In contrast, in rare category detection, each rare category forms a compact cluster in the feature space and are self-similar. [15] developed an unsupervised local-density-differential sampling method for detecting examples of each minority class. [10] presented an active learning scheme to exploit the cluster structure in the data. In order to identify anomalies at different scales, [25] created a hierarchy by repeatedly applying mean shift with an increasing bandwidth on the data. [17] proposed the RACH algorithm to address the problem of rare category characterization by exploring the compactness property of the minority class. Similar to [17], we assume that the majority classes have a smooth distribution, and the minority classes exhibit a compactness property.

2.2 Dual-heterogeneity Learning


Multi-task learning assumes that multiple tasks can benefit from certain common structures. To name a few, feature selection based methods model the task relatedness by constraining all models to share a common set of features [2]; clustered-based methods aim to discover the clustered structure from multiple task [28]; alternating structure optimization (ASO) [1] decomposes the model into the task-specific and task-shared feature mapping. Some recent multi-task learning methods dealt with irrelevant tasks [8, 13].

Most recently, dual-heterogeneity learning began to receive more attention, but most of the existing work focus on addressing the heterogeneity challenge but ignoring the rarity issue. A graph-based learning framework [16] was proposed to address the problems with both feature heterogeneity and task heterogeneity. In contrast, an inductive multi-view learning algorithm [27] was proposed to learn from multiple related tasks via co-regularization. To the best of our knowledge, there are no previous work focused on rare category analysis in the dual-heterogeneity setting.

3 The Proposed $M^2LID$ Model

We first introduce an effective metric for boundary characterization. Then, a novel $M^2LID$ framework is proposed to model the rarity and task/view heterogeneity properties simultaneously and complementarily.

3.1 Boundary Characterization

First, the different properties between KNN and RKNN is introduced. Then, the Hub/Authority [19] concepts are used to quantitatively measure their difference. Based on the Hub/Authority values, a new metric called border-degree is defined for boundary characterization.

According to [26], reverse $k$-nearest neighbor (RKNN) is defined as follows:

**Definition 1. (RKNN)** Given a dataset $D$, a query point $p$, and a positive integer $k$, reverse $k$ nearest neighbor of $p$, denoted as $RKNN_p(k)$, is a set of points $p_i$ that $p_i \in D$ and $\forall p_i, p \in KNN_p(k)$, where $KNN_p(k)$ are the $k$-nearest neighbors of point $p_i$.

Since the nearest neighbor relation is asymmetric, the set of points that are closest to a query point (i.e., nearest neighbors) differs from the set of points that have the query point as their nearest neighbors (i.e., reverse nearest neighbors). A property of RKNN is
that it examines the neighborhood of an object with the view of entire dataset instead of the object itself. Hence, it can capture the distribution property of the underlying data and allow the identification of boundary points [26]. Thus, we make use of the difference properties between KNN and RKNN to capture the sharp changes in density near the boundary of minority classes. It is based on the following observation: for the minority class, the boundary instance points to a few of instances of minority class, as well as being pointed by a few of instances in majority class, which is different from the interior instances in minority class. In other words, the difference between KNN and RKNN of the boundary instances is more significant than that of interior instances. Such a distinctive feature help to capture the density change around the border.

Intuitively, if two instances have more common k-nearest neighbors, they will have more similar Hub values; if two instances have more common reverse k-nearest neighbors, they will have more similar Authority values. Therefore, we use Hub/Authority [19] concepts to model the difference between KNN/RKNN and measure the density changes quantitatively. Based on the Hub/Authority values, we define a novel metric called border-degree to measure the probability of a point near the border.

Definition 2. (Border-degree) Given an instance \( x \), its border-degree is defined as:

\[
b(x) = h(x) - \sigma a(x)
\]

where \( \sigma \) is a positive parameter, \( h(x) \) and \( a(x) \) denote its Hub and Authority values, respectively.

The larger border-degree value an instance has, the more probably it is near the boundary. A property of border-degree metric is that it is skewed around the border while flat in the regions far from border. The border-degree encodes the boundary information between majority and minority and captures the sharp changes in density near the boundary, which can be used to improve the prediction performance on imbalanced data. An illustrating example will be provided in the experiment section.

3.2 The \( M^2LID \) Framework

We first construct both the directed and undirected KNN graph. Then, a graph-based framework \( M^2LID \) is proposed.

Directed and Undirected KNN Graph: Suppose we have \( T \) tasks and \( V \) views. For the \( i \)-th task and the \( j \)-th view, we build a directed KNN graph \( G_{ij}^w \) where the vertex set \( X_{ij} \) consists of two subsets, i.e., instance set and feature set. Let \( m_{ij} \) be the number of instances, \( n_{ij} \) the number of features. Each directed edge is an ordered pair of nodes \((u, v)\) representing that \( v \) is the \( k \)-nearest neighbor of \( u \). Let \( W_{ij} \) be the similarity matrix for the graph \( G_{ij}^w \). \( W_{ij} \) comprises the instance-feature similarity denoted by the feature values, the instance-instance and feature-feature similarities which can be estimated by using the Gaussian similarity function. Similar to [16], we assume that the feature values are non-negative. Note that we can also build the directed RKNN graph by reversing the direction of the edges in the directed KNN graph [26].

We can further build an undirected KNN graph \( G_{ij}^s \), where we connect \( u \) and \( v \) with an undirected edge if \( v \) is among either the \( k \)-nearest or reverse \( k \)-nearest neighbors of \( u \). Let \( S_{ij} \) be the similarity matrix for \( G_{ij}^s \). Note that \( S_{ij} \) is symmetric, while \( W_{ij} \) is asymmetric.

Consistency on Undirected Graphs: From the random walks point of view, given the undirected KNN graph \( G_{ij}^s \), we hope to observe the smoothness consistency among the nodes. We assume that each feature is associated with a label which is analogous to the instances [16].

Specifically, suppose \( f_{ij} \) is the prediction function whose first \( n_{ij} \) elements are the predictions for the instances and the remainders for features (i.e., \( f_{ij} = [f_{ij}, f_{ij}^T] \)), \( y_{ij} \) the labels for \( i \)-th task and the \( j \)-th view. Denote the prediction vector as \( f = [f_{11}^T, \ldots, f_{1V}^T, \ldots, f_{T1}^T, \ldots, f_{TV}^T]^T \), and Hub vector \( h \), Authority vector \( a \), border-degree vector \( b \), label vector \( y \), correspondingly. We first derive the smoothness consistency function for the prediction \( f \). Define the Laplace matrix \( L(S) = D^{-\frac{1}{2}}(D - S)D^{-\frac{1}{2}} \), where \( D \) is a diagonal matrix whose element \( D_{uu} = \sum_v S_{uv} \). The objective is to maximize the smooth consistency among nearest neighbors of both instances and features (i.e., the first term), consistency with the label information from the training instances (i.e., the second term), view consistency in terms of instances (i.e., the third term), and task consistency in terms of features (i.e., the last term), which is equivalent to minimizing:

\[
J_C(f) = \sum_{i=1}^{T} \sum_{j=1}^{V} f_{ij}^T L_{ij} f_{ij} + \gamma \sum_{i=1}^{T} \sum_{j=1}^{V} ||f_{ij} - y_{ij}||^2 \\
+ \alpha \sum_{i=1}^{T} \sum_{k=1}^{V} ||f_{ij} - f_{ik}||^2 + \beta \sum_{i=1}^{T} \sum_{j=1}^{V} \sum_{k=1}^{V} ||f_{ij} - f_{kl}||^2
\]

where the Laplace matrix \( L_{f_{ij}} = L(S_{ij}) \), and \( \gamma, \alpha, \beta \) are non-negative parameters.

Consistency on Directed Graphs: Meanwhile, we hope to observe the smoothness consistency on the directed KNN graph \( G_{ij}^w \). Accoring to [19], we have

\[
h_{ij}^{t+1} = W_{ij} W_{ij}^T h_{ij}^t \quad \text{and} \quad a_{ij}^{t+1} = W_{ij}^T W_{ij} a_{ij}^t
\]

where \( t \) is the iteration. So we can derive the Laplace matrix \( L_{h_{ij}} = L(W_{ij} W_{ij}^T) \) and \( L_{a_{ij}} = L(W_{ij}^T W_{ij}) \) for Hub and Authority respectively.

Likewise, to further take advantage of the dual heterogeneity, we define the smoothness consistency functions for regularized Hub as...
\[ J_C (h) = \frac{\gamma}{V} \sum_{i=1}^{V} h_i^T L_{h_i} h_i + \alpha \sum_{i=1}^{V} \sum_{j,k=1}^{T} ||h_{ij}^T - h_{ik}^T||^2 + \beta \sum_{i=1}^{V} \sum_{j,k=1}^{T} ||h_{P}^T - h_{P}^T||^2 \]

and regularized Authority \( J_C (a) \) correspondingly. Note that different from \( J_C (f) \), the term controlling the consistency with label information does not appear in \( J_C (h) \) (or \( J_C (a) \)) because there is no correspondence between the Hub (or Authority) values and the labels.

**Consistency between Prediction and Border-degree:** Without loss of generality, let \( y(x) = 1 \) for positive instance (minority), \( y(x) = -1 \) for negative instance (majority), and \( y(x) = 0 \) for unlabeled instance initially. Intuitively, the boundary instance will have large border-degree and small absolute value of prediction, while the instance far away from boundary will have small border-degree and large absolute value of prediction. In other words, the absolute value of prediction and border-degree is negatively correlated. Various metrics can be used to model the correlation between the prediction \( f \) and border-degree \( b \), such as Pearson correlation coefficient and Kullback-Leibler (KL) divergence. Alternately, we borrow the idea from Pearson correlation coefficient to model the correlation between prediction and border-degree:

\[
J_P (f, b) = \left[ \frac{f - \bar{r}_f}{\sigma_f} \right]^T \left[ \frac{b - \bar{r}_b}{\sigma_b} \right]^2
\]

where \( \mu_f \) and \( \sigma_f \) (or \( \mu_b \) and \( \sigma_b \)) denote the mean and standard deviation of \( f \) (or \( b \)), respectively, which can be estimated empirically. \( \bar{r}_f \) (or \( \bar{r}_b \)) is a vector where each element is \( \mu_f \) (or \( \mu_b \)). For simplicity, denote \( v^2 = [v_1^2, \ldots, v_n^2]^T \) for any n-by-1 vector \( v \) here.

**Overall Objective:** In summary, our overall goal is to maximize the smoothness consistency objective for all of predictions, Hub, and Authority, i.e., \( f, h, a \), and simultaneously minimize the correlation between the prediction and the border-degree. Hence, the objective is to minimize,

\[
J(f, h, a) = J_C (f) + J_C (h) + J_C (a) + \lambda J_P (f, b)
\]

where \( \lambda \) is the trade-off coefficient.

\( M^2LID \) tackles the rarity by using border-degree for boundary characterization, enhances the view consistency by requiring the view-based learners to behave similarly on the instances, and models the task relatedness by requiring the task-specific learners to behave similarly on the features. Furthermore, \( M^2LID \) addresses both the rarity and heterogeneity challenges in a way of mutual benefit. Intuitively, heterogeneity helps characterize the rarity and detect the border more accurately, which in turn helps mine the heterogeneity and improve the learning performance. Specifically, it is expected that one can obtain more accurate regularized Hub/Authority values in a multi-view multi-task regularization framework than in a single-view single-task framework by borrowing strength from related tasks/views. A more accurate Hub/Authority values help characterize the rarity and detect the border more accurately. On the other hand, a refined border-degree values can guide the model to tackle the imbalanced data and improve the prediction performance of the multi-view multi-task framework.

**Algorithm 1** Algorithm for \( M^2LID \)

**Input:**
- T-task data with V-view \( \{X_{ij} | 1 \leq i \leq T, 1 \leq j \leq V \} \)
- free parameters: \( \alpha, \beta, \gamma, \lambda, \sigma \)

1. Initialize the label vectors \( y \);
2. Initialize the Hub/Authority \( h^{(0)}/a^{(0)} \), and border-degree \( b^{(0)} \) using Eq. 3.1;
3. for \( t = 0 : n_{iter} \) do
   4. Fix \( \{ h^{(t)}, a^{(t)} \} \) and update the predicted labels \( f^{(t)} \);
   5. Fix \( \{ f^{(t)}, h^{(t)} \} \) and update the Hub \( h^{(t+1)} \);
   6. Fix \( \{ f^{(t)}, h^{(t+1)} \} \) and update the Authority \( a^{(t+1)} \);
   7. end for;
8. return predicted labels for the test data by using Eq. 3.6.

**Optimization:** The overall objective Eq. 3.5 is an unconstrained quadratic optimization, which can be solved very efficiently. For example, Eq. 3.2 can be transformed into the compact matrix form:

\[ J_C (f) = f^T H_f f - 2p^T f \]

where \( p = \gamma y \) and the block matrices \( H_f \) are defined as:

\[
[H_f]_{(f_i, f_{st})} = \begin{cases} \frac{L_{f_{ij}} + \gamma I_{n_{ij} + m_{ij}} + 2\alpha (V - 1) A + 2\beta (T - 1) B}{2} & i = s \land j = t \\ -\alpha A & i = s \land j \neq t \\ -\beta B & i \neq s \land j = t \\ 0_{(n_{ij} + m_{ij}) \times (n_{st} + m_{st})} & \text{otherwise} \end{cases}
\]

where \( 1 \leq i, s \leq T, 1 \leq j, t \leq V \), and

\[
A = \begin{bmatrix} I_{n_{ij}} & 0_{n_{ij} \times m_{ij}} \\ 0_{m_{ij} \times n_{ij}} & 0_{m_{ij} \times m_{ij}} \end{bmatrix}, \quad B = \begin{bmatrix} 0_{n_{ij} \times n_{st}} & 0_{n_{ij} \times m_{ij}} \\ 0_{m_{ij} \times n_{st}} & I_{m_{ij}} \end{bmatrix}
\]

By setting the derivatives of Eq. 3.5 with respect to each block of \( \{ f, h, a \} \) to zero, we can obtain the analytical solutions.

Intuitively, the smaller the border-degree is, the more confident the view-based classifier is with its prediction. Thus, for an unlabeled instance, its final prediction takes the weighted sum of the predictions resulting from multiple view-based classifiers:

\[
f_i^* (x) = \sum_{j=1}^{V} \left[ 1 - \tilde{b}_{ij} (x) \right] f_{ij} (x)
\]

where \( \tilde{b}_{ij} \) is the normalized border-degree, i.e., \( \tilde{b}_{ij} (x) = b_{ij} (x) / \sum_{k=1}^{V} b_{ik} (x) \).

We propose an iterative \( M^2LID \) Algorithm as shown in Algorithm 1. It is based on block coordinate descent algorithm [21], and updates each block of \( \{ f, h, a \} \) iteratively.
4 Performance Analysis

In this section, we present the performance analysis on the important properties regarding the convergence, error bound, and algorithm complexity of the proposed approach.

4.1 Convergence

**Theorem 4.1. (Convergence)** The proposed M^2LID algorithm converges to the local optimum.

**Proof.** For Eq. 3.5, since the Laplace matrices are positive semi-definite, it is easily to prove that \( H_f \) is also positive semi-definite. By taking second-order derivative of \( J_C(f) \) with respect to \( f \), we have \( \nabla^2 J_C(f) = 2H_f \). Since \( H_f \) is positive semi-definite, \( J_C(f) \) is convex with respect to \( f \). Moreover, we can prove that \( J_P(f,b) \) is convex with respect to \( f \). Therefore, given \((h,a)\), Eq. 3.5 is convex with respect to \( f \). Likewise, we can prove that Eq. 3.5 is also convex with respect to \( h \) and \( a \).

Based on the theoretical results from block coordinate descent methods [21], the proposed M^2LID algorithm in Algorithm 1 converges to local optimum. □

4.2 Error Bound Analysis

Analogous to [12], we derive the error bound for our proposed method. Here, we focus on multi-task learning from imbalanced data with multi-view, which distinguishes us from [12].

For \( j \)-th \((1 \leq j \leq V)\) view, denote the conditional probabilities \( P[f_j = -1 | y = 1] = p_j \) and \( P[f_j = 1 | y = -1] = q_j \). Denote \( P[y = 1] = r \).

**Theorem 4.2. (False Negative Error Bound)**

Given the error bound \( \rho \geq \frac{rE[p_j(1-b_j)]}{rE[p_j(1-b_j)] + (1-r)E[(1-b_j)(1-q_j)]} \), the probability of making a false negative error by M^2LID can be bounded as follows:

\[
P[f = -1 | y = 1] \geq \exp \left( -2\mu^2 \frac{\rho C}{V} \right)
\]

where \( \mu = E \left[ (1 - b_j) [rp_j(1 - \rho) - \rho(1 - q_j)(1 - r)] \right] \), and \( C \) is a constant.

**Proof.** According to Eq. 3.6, since the final prediction is the weighted sum of output from view-based classifiers, we have \( P[f = -1 | y = 1] = \sum_{j=1}^{V} (1 - b_j) P[f_j = -1 | y = 1] = \sum_{j=1}^{V} (1 - b_j) p_j \), and \( P[f = 1 | y = -1] = \sum_{j=1}^{V} (1 - b_j) \) likewise. Then, the probability of making a false negative prediction can be estimated by using Bayes theorem:

\[
P[y = 1 | f = -1] = \frac{P[f = -1 | y = 1]P[y = 1]}{P[f = -1 | y = 1]P[y = 1] + P[f = 1 | y = -1]P[y = -1]}
\]

\[
= \frac{\sum_{j=1}^{V} (1 - b_j)p_j + (1-r)\sum_{j=1}^{V} (1-b_j)(1-q_j)}{\sum_{j=1}^{V} (1 - b_j)p_j + (1-r)\sum_{j=1}^{V} (1-b_j)(1-q_j)}
\]

Given the error bound \( \rho \), we have:

\[
P[y = 1 | f = -1] \geq \rho \iff \sum_{j=1}^{V} Z_j \geq 0,
\]

where \( Z_j = (1 - b_j) [rp_j(1 - \rho) - \rho(1 - q_j)(1 - r)] \) is a random variable. Denote the expectation of \( Z_j \) as \( E[Z_j] = \mu \). If \( \rho \geq \frac{rE[p_j(1-b_j)]}{rE[p_j(1-b_j)] + (1-r)E[(1-b_j)(1-q_j)]} \), we have \( \mu \leq 0 \). It is easy to see that \( Z_j \) is bounded, which is denoted by \([\omega_1, \omega_2]\). The Hoeffding inequality [18] shows that when \( \tau \geq 0 \):

\[
P \left[ \sum_{j=1}^{V} Z_j - E \left[ \sum_{j=1}^{V} Z_j \right] \geq \tau \right] \leq \exp \left( \frac{-2\tau^2}{V(\omega_1 - \omega_2)^2} \right)
\]

By letting \( \tau = -V \mu \) and \( C = (\omega_1 - \omega_2)^2 \), we have

\[
P \left[ \sum_{j=1}^{V} Z_j \geq 0 \right] \leq \exp \left( \frac{-2V\mu^2}{C} \right)
\]

which completes the proof. □

Likewise, we can also derive the false positive error bound as given in Theorem 4.3:

**Theorem 4.3. (False Positive Error Bound)**

Given the error bound \( \rho \geq \frac{rE[(1-p_j)(1-b_j)]}{rE[(1-p_j)(1-b_j)] + (1-r)E[(1-b_j)(1-q_j)]} \), the probability of making a false positive error by M^2LID can be bounded as follows:

\[
P \{ P[y = -1 | f = 1] \geq \rho \} \leq \exp \left( \frac{-2V\mu^2}{C} \right)
\]

where \( \mu = E \left[ (1 - b_j) [(1 - r)q_j] \right] \), \( r \) is a constant.

The theorems indicate that the error bound \( \rho \) is directly related to the probability of errors made by view-based classifier and the border-degree. Furthermore, the probability that the algorithm making a false negative (positive) error greater than a given error bound \( \rho \) will be exponentially decreased when the number of views increases.

4.3 Algorithm Complexity

The time-consuming steps in the algorithm of M^2LID are the computations of the inverse matrices in Steps 4-6. Denote the number of rows in the square matrix \( H_f \) by \( N = \sum_{i=1}^{T} \sum_{j=1}^{V} (n_{ij} + m_{ij}) \). The time complexity of the proposed algorithm is given in Lemma 4.1.

**Lemma 4.1. (Time Complexity)** The time complexity of M^2LID is \( O(\tau_{iter}N^2) \).

**Proof.** The time complexity of computing the inverse matrix is \( O(N^2) \) by using Coppersmith-Winograd algorithm [9]. Since M^2LID algorithm repeats to run \( \tau_{iter} \) times, the total time complexity is \( O(\tau_{iter}N^2) \).

Likewise, the space-consuming steps in the M^2LID algorithm are to store the matrices \( H_f \) and the corresponding inverse matrices. It is easy to obtain the space complexity of the M^2LID algorithm, which is given in Lemma 4.2.

**Lemma 4.2. (Space Complexity)** The space complexity of M^2LID is \( O(N^2) \).
Figure 1: Boundary characterization on three synthetic datasets: a) Circle; b) Half-moon; c) Plus, where the blue (or green, yellow) stars representing the instances with large border-degree values are located around the boundary regions.

Table 1: Task description for Cora datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>DA-NT</td>
<td>/datastructures/algorithms/mltheory/computationalcomplexity/ (711)</td>
</tr>
<tr>
<td>DA-ML</td>
<td>/datastructures/algorithms/mltheory/computationalcomplexity/ (459)</td>
</tr>
<tr>
<td>NT-ML</td>
<td>/networking/routing/ (477)</td>
</tr>
</tbody>
</table>

5 Experiments

In this section, we present the experimental results on both synthetic and real-world datasets.

5.1 Synthetic Datasets

We generate synthetic datasets and visualize the boundary characterization in order to verify the effectiveness of the proposed border-degree metric. Figure 1 shows three 2-dimensional synthetic datasets. In each dataset, the major class has 2000 instances (black pluses) with a Gaussian distribution. The minority class has 100 instances (red circles) with uniform distribution within the regions with different shapes, i.e., 'circle', 'moon', 'plus', respectively.

Then, we also visualize those instances with large border-degree values. The blue (green, yellow) stars represent the instances with top-10 (20, 40) largest border-degree values, respectively. From these figures, we can see that most boundary instances in either majority or minority class have large border-degree value, which verifies our intuition. Therefore, border-degree can be used as an effective metric for boundary characterization.

5.2 Real Datasets

The Spam Email data set was released by ECML/PKDD 2006 discovery challenge (www.ecmlpkdd2006.org/challenge.html). The inboxes differ in the distribution of emails. The goal is to construct a spam filter for each single user that correctly classifies its emails as spam or non-spam. In problem A, there are emails from 3 different users (2500 emails per user) corresponding to different tasks. The emails are described from two views: the first view corresponds to the TF-IDF features; the second view corresponds to the latent topics obtained by applying Probabilistic Latent Semantic Analysis on the term counts.

Cora [22] is an online archive which contains approximately 37,000 computer science research papers.

The documents are categorized into a hierarchical structure. Different sub-categories under the same top category are drawn from different distributions. Table 1 shows the data sets and task description, where the number in the parenthesis is the number of instances. We create two views for each dataset as we done for the Spam Email dataset.

The F-score (harmonic mean of precision and recall) of minority class is used as the evaluation criterion. We repeat all the comparison algorithms ten times for each dataset and report the average F-score. For $M^2LID$, we tune $\alpha, \beta, \gamma$, and $\lambda$ on the grid $4^{-5:1:5}$, and $\sigma$ on $2^{1:5:1:5}$ by cross-validation on the training data.
5.2.1 Comparison with Heterogeneous Learning

We compare $M^2LID$ with a variety of heterogeneous learning approaches including: 1) multi-task multi-view learning algorithm $IteM^2$ [16]; 2) multi-view learning method CoEM which is a variant of Co-Training [3]; 3) multi-task algorithms implemented in MALSAR [29] toolbox including CASO [7], CMTL [28], rMTFL [13], RMTL [8].

All of $M^2LID$, $IteM^2$, and CoEM are fed with the multi-view data, whereas the multi-task approaches are input with the concatenated features from all the views. The parameters are tuned for each algorithm by cross-validation on training data.

For Spam Email datasets, Figures 2-4 show the comparison results for each task, respectively. The average performance and standard deviations are shown in Figure 5 by aggregating all the tasks. In each figure, $x$-axis represents the ratio between the number of instances in minority and majority classes, and $y$-axis denotes the F-score for the minority class. We randomly sample the instances to generate the imbalanced subset according to the specified ratio value. Each dataset is randomly splitted into training set (50%) and test set (50%), where they hold the same ratio between the number of instances in minority and majority classes.

From these figures, we can observe a common trend that the performance of all the algorithms usually become worse significantly when the ratio decreases, which indicates that the rarity property of data would greatly influence the learning system. $M^2LID$ performs the best among all the algorithms when the ratio is less than 0.2, which demonstrates the effectiveness of our proposed model. In comparison with the other methods, the key competency of $M^2LID$ is that it takes the rarity into consideration and models the rarity and task/view heterogeneity properties in a way of mutual benefit. In contrast, though $IteM^2$ makes full advantage of task and view heterogeneity, its discrimination power on imbalanced data is limited since it does not consider the rarity issue. Among all the algorithms CoEM performs the worst, which shows that the traditional multi-view learning approaches would not be effective in the situations where the data in different tasks follow different distributions. This may due to the fact that they treat the multiple tasks indiscriminately. On the contrary, the multi-task learning methods significantly outperform CoEM by modeling the task relatedness in different way. It is interesting that both of two approaches dealing with outlier tasks, i.e., RMTL and rMTFL, perform better than the other two multi-task learning methods, i.e., CASO and CMTL. It suggests that the multi-task learning algorithms dealing with outlier tasks may be more robust to the imbalanced data comparing to those assuming that there are no outliers.

Figures 6-8 show the comparison results for Cora datasets. For each dataset, the results are averaged by the tasks. From these figures, we have the similar conclusions with those on Spam Email datasets. Note that the result for CoEM is not shown due to its relatively worse performance.
5.2.2 Comparison with Imbalanced Learning

We also compare $M^2LID$ with various imbalanced learning methods including: 1) OverSampling; 2) UnderSampling; 3) SMOTE [6]; 4) ensemble learning methods for imbalanced data, which include HardEnsemble and SoftEnsemble [30]. An online imbalanced learning package named CSNN (http://lamda.nju.edu.cn/Data.ashx) which implemented all above algorithms is used for comparison. All the imbalanced learning algorithms are input with the concatenated features from all the views. The parameters are tuned for each algorithm by using cross-validation on the training data.

Figure 9 shows the average performance on Spam Email datasets by aggregating all the tasks, and Figures 10-12 show the results for Cora datasets, respectively. First of all, the performance of all the methods usually worsen along with the decrease of ratio, which is similar to the trend we got from the last subsection. It indicates that a higher degree of class imbalance may result in greater difficulty in imbalanced learning. Both OverSampling and SMOTE basically perform better than UnderSampling which is worst among all the methods. It seems that it would pay more attention to characterize the minority instances than majority instances on our datasets. Ensemble methods including HardEnsemble and SoftEnsemble can further improve generalization ability by combining the predictions from different learners. However, though the above approaches are effective for homogeneous data, they might not be the best choices for heterogeneous data, which is demonstrated by the experimental results that all the comparison methods basically perform worse than $M^2LID$. First, it would due to the fact that traditional imbalanced learning algorithms cannot take fully advantage of task heterogeneity and/or view heterogeneity to improve the generalization performance. Second, these methods might not make full use of the the compactness property of minority class [17].

5.2.3 Parameter Sensitivity and Convergence

We study the performance sensitivity with $k$, which is used to select the nearest neighbors and construct the KNN/RKNN graph. Figure 13 plots the performance curves varying with different values of $k$ on the DA-NT training data. It shows that the performance of $M^2LID$ is quite robust over a wide range of values for $k$. As a result, we set $k = 40$ in the other experiments.

We empirically study the convergence property of $M^2LID$ on the Spam Email dataset. The result is shown in Figure 14. From this figure, we can see that the $M^2LID$ converges fast and its performance becomes stable after 5 iterations. Thus, we terminate the algorithm after a maximum of 10 iterations.

6 Conclusion

In this paper, we first introduce an effective metric for boundary characterization. Then, a novel $M^2LID$ framework is proposed to fully make use of both rarity and heterogeneity simultaneously to improve learning performance on imbalanced data. An effective algorithm based on block coordinate descent method is also
Figure 12: F-score of different imbalanced learning methods on Cora DA-ML (average).

Figure 13: F-score varies with k.

presented. Then the theoretical analysis regarding the convergence, error bound, and time/space complexity, are presented for the $M^2\text{LID}$ approach. Experimental results on synthetic and real data sets demonstrate the effectiveness of the $M^2\text{LID}$ model.

References


