Knowl Inf Syst (2016) 48:201–228
DOI 10.1007/s10115-015-0870-3

REGULAR PAPER

Transfer learning for class imbalance problems with inadequate data

Samir Al-Stouhi[1] · **Chandan K. Reddy**[2]

Received: 31 August 2014 / Revised: 30 May 2015 / Accepted: 2 August 2015 /
Published online: 25 August 2015

Abstract A fundamental problem in data mining is to effectively build robust classifiers in the presence of skewed data distributions. Class imbalance classifiers are trained specifically for skewed distribution datasets. Existing methods assume an ample supply of training examples as a fundamental prerequisite for constructing an effective classifier. However, when sufficient data are not readily available, the development of a representative classification algorithm becomes even more difficult due to the unequal distribution between classes. We provide a unified framework that will potentially take advantage of auxiliary data using a transfer learning mechanism and simultaneously build a robust classifier to tackle this imbalance issue in the presence of few training samples in a particular target domain of interest. Transfer learning methods use auxiliary data to augment learning when training examples are not sufficient and in this paper we will develop a method that is optimized to simultaneously augment the training data and induce balance into skewed datasets. We propose a novel boosting-based instance transfer classifier with a label-dependent update mechanism that simultaneously compensates for class imbalance and incorporates samples from an auxiliary domain to improve classification. We provide theoretical and empirical validation of our method and apply to healthcare and text classification applications.

Keywords Rare class · Transfer learning · Class imbalance · AdaBoost · Weighted majority algorithm · HealthCare informatics · Text mining

1 Introduction

One of the fundamental problems in machine learning is to effectively build robust classifiers in the presence of class imbalance. Imbalanced learning is a well-studied problem and many

✉ Chandan K. Reddy
reddy@cs.wayne.edu

1 Honda Automobile Technology Research, Southfield, MI, USA

2 Department of Computer Science, Wayne State University, Detroit, MI, USA

sampling techniques, cost-sensitive algorithms, kernel-based techniques, and active learning methods have been proposed in the literature [1]. Though there have been several attempts to solve this problem, most of the existing methods always assume an ample supply of training examples as a fundamental prerequisite for constructing an effective classifier tackling class imbalance problems. In other words, the existing imbalanced learning algorithms only address the problem of "Relative Imbalance" where the number of samples in one class is significantly higher compared to the other class and there is an abundant supply of training instances. However, when sufficient data for model training is not readily available, the development of a representative hypothesis becomes more difficult due to an unequal distribution between its classes.

Many datasets related to medical diagnoses, natural phenomena, or demographics are naturally imbalanced datasets and will typically have an inadequate supply of training instances. For example, datasets for cancer diagnosis in minority populations (benign or malignant), or seismic wave classification datasets (earthquake or nuclear detonation) are small and imbalanced. "Absolute Rarity" refers to a dataset where the imbalance problem is compounded by a supply of training instances that is not adequate for generalization. Many of such practical datasets have high dimensionality, small sample size and class imbalance. *The minority class within a small and imbalanced dataset is considered to be a "Rare Class"*. Classification with "Absolute Rarity" is not a well-studied problem because the lack of representative data, especially within the minority class, impedes learning.

To address this challenge, we develop an algorithm to simultaneously rectify for the skew within the label space and compensate for the overall lack of instances in the training set by borrowing from an auxiliary domain. We provide a unified framework that can potentially take advantage of the auxiliary data using a "knowledge transfer" mechanism and build a robust classifier to tackle this imbalance issue in the presence of fewer training samples in the target domain. Transfer learning algorithms [2,3] use auxiliary data to augment learning when training examples are not sufficient. In the presence of inadequate number of samples, the transfer learning algorithms will improve learning on a small dataset (referred to as target set) by including a similar and possibly larger auxiliary dataset (referred to as the source set). In this work, we will develop one such method optimized to simultaneously augment the training data and induce balance into the skewed datasets.

This paper presents the first method for rare dataset classification within a transfer learning paradigm. In this work, we propose a classification algorithm to address the problem of "Absolute Rarity" with an instance transfer method that incorporates the best-fit set of auxiliary samples that improve *balanced error minimization*. Our transfer learning framework induces balanced error optimization by simultaneously compensating for the class imbalance and the lack of training examples in "Absolute Rarity". To achieve this goal, we utilize ensemble-learning techniques that iteratively construct a classifier that is trained with the weighted source and target samples that best improve balanced classification. Our transfer learning algorithm will include label information while performing knowledge transfer.

This paper effectively combines two important machine learning concepts: the concept of compensating for the skew within the label space (which belongs to the domain of "Imbalanced Learning") and the concept of extracting knowledge from an auxiliary dataset to compensate for the overall lack of samples (which belongs to a family of methods known as "instance-based transfer learning"). We aim to construct a hypothesis and uncover the separating hyperplane with only a handful of training examples with data that is complex in both the feature and label spaces. The complexity of the data skewness and the rarity of training examples prohibit hypothesis construction by human experts or standard algorithms, and thus

we present a solution that can be applied when nothing else suffices. The main contributions of this paper are as follows:

1. Present a complete categorization of several recent works and highlight the need for a new type of specialized algorithms to solve a niche but important problem that is not addressed in the current literature.
2. Propose a novel transfer learning algorithm, Rare-Transfer, optimized for transfer within the label space to effectively handle rare class problems.
3. Provide theoretical and empirical analysis of the proposed Rare-Transfer algorithm.
4. Demonstrate the superior performance of the proposed algorithm compared to several existing methods in the literature using various real-world examples.

The rest of the paper is organized as follows: In Sect. 2, we describe the different types of datasets and briefly discuss the related methods suitable for each type. Section 3 presents the motivation for a unified balanced optimization framework. Section 4 describes our algorithm, "Rare-Transfer", which addresses the "Absolute Rarity" problem. Section 5 presents the theoretical analysis of the proposed algorithm. For further validation, Sect. 6 presents empirical analysis of our framework and is followed by experimental results on real-world data in Sect. 7. Finally, we discuss possible extensions and conclude our work.

2 Characterization of existing machine learning domains

To describe datasets in terms of both size and imbalance, we use the "Label-Dependent" view in Fig. 1. The sub-figures present a binary classification problem with normally distributed samples within each class[1] (thus we describe it as label-dependent since the distributions are normal within each label). Figure 1 illustrates the different datasets with an overview of the related machine learning fields[2] that can improve learning.

1. *Standard dataset* Figure 1a depicts a standard dataset with a relatively equal number of samples within each class (balanced class distribution) and an adequate number of samples for generalization. To learn from balanced datasets, equal importance is assigned to all classes and thus maximizing the overall arithmetic accuracy is the chosen optimization objective. A variety of standard machine learning and data mining approaches can be applied for standard datasets as such methods serve as the foundation for the algorithms that are modified for any peculiar feature set or distribution.
2. *Imbalanced dataset* The dataset in Fig. 1b is a relatively imbalanced dataset. It is relatively imbalanced because there is a between-class imbalance where one class encompass the majority of the training set. The balance is relative since both minority and majority training subsets contain adequate training samples. For example, email spam classification is a relatively imbalanced problem since 97 % (majority) of emails sent over the net are considered unwanted emails [4] and with around 200 billion messages of spam sent per day [5], the number of non-spam emails (minority) is also a large dataset. A dataset where the number samples belonging to different classes is highly disproportionate is considered to be an "imbalanced dataset" with the postulation that the imbalance is relative [1]. Because the majority class overwhelms the minority class, imbalanced learning models are biased to improve learning on the minority class (without any consideration to the availability of training examples).

[1] The terms class and label are used interchangeably in our discussion.

[2] Only concepts that are relevant for "Absolute Rarity" are discussed.

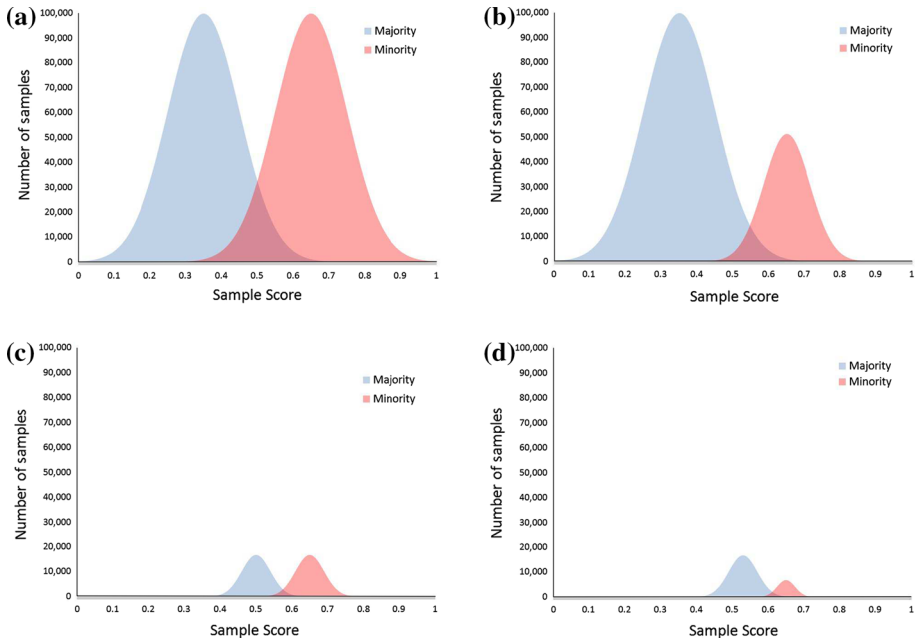

Fig. 1 Label-dependent view of different type of datasets. **a** Standard. **b** Imbalanced. **c** Small. **d** Absolute rarity

3. *Small dataset* The dataset in Fig. 1c is a balanced dataset with a training sample size that is inadequate for generalization. One method to determine the number of samples required for training is to rely on the "Probably Approximately Correct (PAC)" learning theory [6]. PAC is applied to determine whether the ratio of the dimensions of the data to the number of training samples is too high where the hypothesis space would thus be exceedingly large. If that ratio is too high, learning becomes difficult and prone to model over-fitting. PAC gives a theoretic relationship between the number of samples needed in terms of the size of hypothesis space and the number of dimensions. The simplest example is a binary dataset with binary classes and d dimensions with hypothesis space of size 2^{2^d}, requiring $O\,(2^n)$ samples [7].
4. *Rare Dataset* (*Dataset with "Absolute Rarity"*) The dataset in Fig. 1d is imbalanced as well as small and thus its imbalance is termed as "Absolute Rarity". Weiss [8] presents a good overview of the problems encountered when analyzing and evaluating such datasets. Different solutions are outlined for handling "Absolute Rarity" with a discussion of solutions for segmentation, bias and noise associated with these datasets. In [9], an end-to-end investigation of rare categories in imbalanced datasets in both the supervised and unsupervised settings is presented.

3 Learning with "Absolute Rarity"

A "Rare Dataset"[3] is a label-skewed and small dataset and presents a set of challenges that are not studied in existing literature. This section examines the parameters that are relevant for the study of "Rare Datasets".

[3] In this paper, a "Rare Dataset" refers to a dataset with "Absolute Rarity".

3.1 Effect of data size on learning

3.1.1 Balanced dataset

The first impediment to learning with "Absolute Rarity" is the fact that the small size of the training set, regardless of imbalance, impedes learning. When the number of training examples is not *adequate* to generalize to instances not present in the training data, it is not theoretically possible to use a learning model as the model will only overfit the training set. The term "adequate" is a broad term as many factors including data complexity, number of dimensions, data duplication, and overlap complexity have to be considered [1]. Computational learning theory [7] provides a general outline to estimate the difficulty of learning a model, the required number of training examples, the expected learning and generalization error and the risk of failing to learn or generalize.

A study in [10] found that the size of training set is the factor with the most significant impact on classification performance. Figure 2 depicts 4 different algorithms that are trained at different training set sizes and demonstrates that increasing the training sets' size improves the classification performance of all the algorithms. To assert that increasing the number of training examples, combined with an error minimizing classifier, yields results where the training and the generalization errors are similar is an intuitive and crucial finding as it demonstrates that the choice of a classification model is less important than the overall size of the training set.

3.1.2 Imbalanced dataset

The second impediment to learning with "Absolute Rarity" is the between-class imbalance where a majority of samples belong to an overrepresented class and a minority of samples belong to an underrepresented class [1]. The imbalanced classification study in [11] found

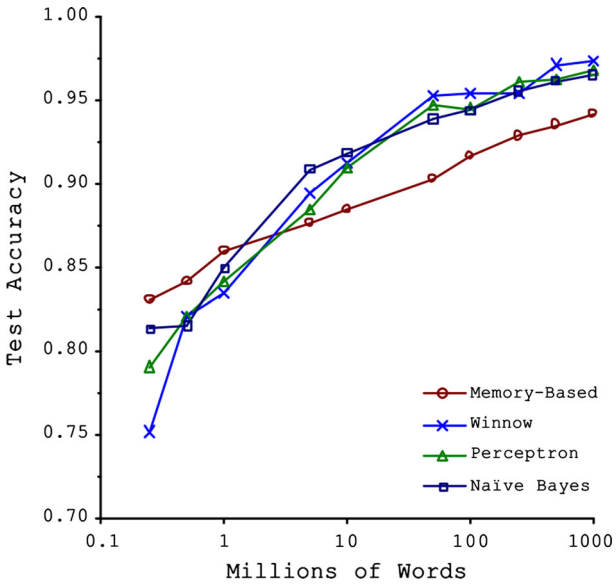

Fig. 2 Learning curves for confusion set disambiguation [10]

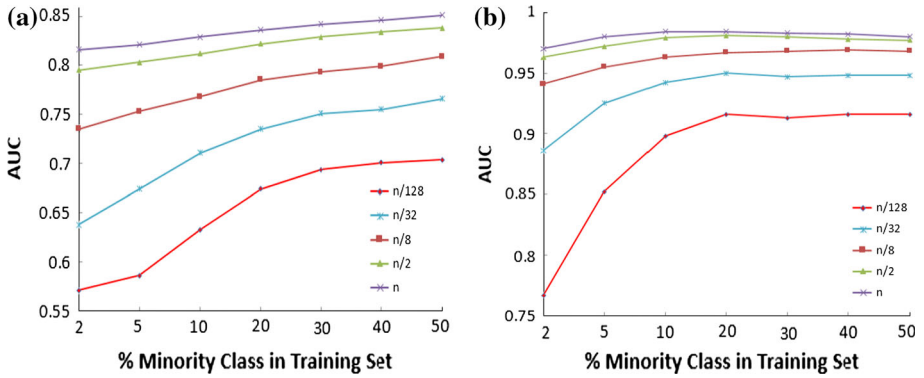

Fig. 3 AUC for imbalanced datasets at different training sample sizes [11]. **a** Adult. **b** Covertype

that the most significant effect on a classifier's performance in an imbalanced classification problem is not the ratio of imbalance, but it is the number of samples in the training set. This is an important finding as it demonstrates that the lack of data in "Absolute Rarity" intensifies the label imbalance problem. As the number of the training examples increased, the error rate caused by imbalance decreased [12] and thus increasing the number of training samples makes the classifiers less sensitive to the between-class imbalance [11].

Figure 3 demonstrates how the lack of training examples degrades learning in an imbalanced dataset [11]. The ROC curve illustrates the performance of a binary classifier where the x-axis represents the False Positive Rate (1-Specificity) and the y-axis represents the True Positive Rate and is an accepted metric in imbalanced learning problems. Figure 3 presents the Area Under the ROC curve (AUC) [13] results in [11] where a classifier was trained for two imbalanced datasets [14] with different subsets of training sets (with a total of n samples). AUC is a simple summary of the ROC performance and can be calculated by using the trapezoidal areas created between ROC points and is thus equivalent to the Wilcoxon–Mann–Whitney statistic [15]. The results demonstrate that increasing the size of the training set directly improves learning for imbalanced datasets.

4 The proposed rare-transfer algorithm

4.1 Notations

Consider a domain (D) comprised of instances $(X \in \mathbb{R}^d)$ with d features. We can specify a mapping function, F, to map the feature space to the label space as "$X \rightarrow Y$" where $Y \in \{-1, 1\}$. If no source or target instances are defined, then n will simply refer to the number of instances in a dataset; otherwise, we will denote the domain with n auxiliary instances as the source domain (D_{src}) and define (D_{tar}) as the target domain with $m \ll n$ instances. Instances that belong to the majority class will be defined as X_{majority} and those that belong to the minority class will be defined as X_{minority}. n^l is the number of source samples that belong to label l, while ε^l is the error rate for label l and denotes the misclassification rate for samples with true label l. N defines the total number of boosting iterations, w is a weight vector. A weak classifier at a given boosting iteration (t) will be defined as $\ddot{f}^t$, and its classification error is denoted by ε^t. $\mathbb{I}$ is an indicator function and is defined as:

$$\mathbb{I}\left[y \neq \ddot{f}\right] = \begin{cases} 1 & y \neq \ddot{f} \\ 0 & y = \ddot{f} \end{cases} \tag{1}$$

4.2 Boosting-based transfer learning

Boosting-based transfer learning algorithms apply ensemble methods to both source and target instances with an update mechanism that incorporates only the source instances that are useful for target instance classification. These methods perform this form of mapping by giving more weight to source instances that improve target training and decreasing the weights for instances that induce negative transfer.

TrAdaBoost [16] is the first and most popular transfer learning method that uses boosting as a best-fit inductive transfer learner. TrAdaBoost trains a base classifier on the weighted source and target set in an iterative manner. After every boosting iteration, the weights of misclassified target instances are increased and the weights of correctly classified target instances are decreased. This target update mechanism is based solely on the training error calculated on the normalized weights of the target set and uses a strategy adapted from the classical AdaBoost [17] algorithm. The weighted majority algorithm (WMA) [18] is used to adjust the weights of the source set by iteratively decreasing the weight of misclassified source instances by a constant factor and preserving the current weights of correctly classified source instances. The basic idea is that the weight of source instances that are not correctly classified on a consistent basis would converge and would not be used in the final classifier's output since that classifier only uses boosting iterations $\frac{N}{2} \rightarrow N$ for convergence [16].

TrAdaBoost has been extended to many transfer learning problems. A multi-source learning [19] approach was proposed to import knowledge from many sources. Having multiple sources increases the probability of integrating source instances that are better fit to improve target learning and thus this method can reduce negative transfer. A model-based transfer in "TaskTrAdaBoost" [19] extends this algorithm to transferring knowledge from multiple source tasks to learn a specific target task. Since closely related tasks share some common parameters, suitable parameters that induce positive transfer are integrated from multiple source tasks. Some of the prominent applications of TrAdaBoost include multi-view surveillance [19], imbalanced classification [20], head-pose estimation [21], visual tracking [22], text classification [16] and several other problems [2].

TransferBoost [23] is an AdaBoost based method for boosting when multiple source tasks are available. It boosts all source weights for instances that belong to tasks exhibiting positive transferability to the target task. TransferBoost calculates an aggregate transfer term for every source task as the difference in error between the target-only task and the target plus each additional source task. AdaBoost was extended in [24] for concept drift as a fixed cost is pre-calculated using Euclidean distance (as one of two options) as a measure of relevance between source and target distributions. This relevance ratio thus gives more weights to data that is near in the feature space and share a similar label. This ratio is finally incorporated to the update mechanism via AdaCost [25].

Many problems have been noted when using TrAdaBooost. The authors in [26] reported that there was a weight mismatch when the size of source instances is much larger than that of target instances. This required many iterations for the total weight of the target instances to approach that of the source instances. In [27], it was noted that TrAdaBoost yielded a final classifier that always predicted one label for all instances as it substantially unbalanced the weights between the different classes. Even the original implementation of TrAdaBoost in [16] re-sampled the data at each step to balance the classes. Finally, various researchers observed that beneficial source instances that are representative of the target concept tend

to have a quick and stochastic weight convergence. This quick convergence was examined by Eaton and desJardins [23] as they observed that in TrAdaBoost's reweighing scheme, the difference between the weights of the source and target instances only increased and that there was no mechanism in place to recover the weight of source instances in later boosting iterations when they become beneficial. TrAdaBoost was improved in [28] where dynamic reweighing separated the two update mechanisms of AdaBoost and WMA for better classification performance.

4.3 The proposed rare-transfer algorithm

To overcome the limitations in boosting-based transfer learning and simultaneously address imbalance in "Absolute Rarity", we present the "Rare-Transfer" algorithm (shown in Algorithm 1). The algorithm exploits transfer learning concepts to improve classification by incorporating auxiliary knowledge from a source domain to a target domain. Simultane-

Algorithm 1 Rare-Transfer Algorithm

Require:

- Source domain instances $D_{src} = \left\{\left(x_{src_i}, y_{src_i}\right)\right\}$
- Target domain instances $D_{tar} = \left\{\left(x_{tar_i}, y_{tar_i}\right)\right\}$
- Maximum number of iterations : N
- Base learner : $\ddot{f}$

Ensure: Target Classifier Output : $\left\{ \dot{f} : X \rightarrow Y \right\}$

$$\dot{f} = sign\left[\prod_{t=\frac{N}{2}}^{N}\left(\beta_{tar}^{t}{}^{-\ddot{f}^{t}}\right) - \prod_{t=\frac{N}{2}}^{N}\left(\beta_{tar}^{t}{}^{-\frac{1}{2}}\right)\right]$$

Procedure:

1: Initialize the weights w for all instances D = $\{D_{src} \cup D_{tar}\}$, where: $w_{src} = \left\{w_{src_1}, \ldots, w_{src_n}\right\}$, $w_{tar} = \left\{w_{tar_1}, \ldots, w_{tar_m}\right\}$, $w = \{w_{src} \cup w_{tar}\}$

2: Set $\beta_{src} = \frac{1}{1+\sqrt{\frac{2\ln(n)}{N}}}$

3: **for** $t = 1$ to N **do**

4: Normalize Weights: $w = \frac{w}{\sum_i^n w_{src_i} + \sum_j^m w_{tar_j}}$

5: Find the candidate weak learner $\ddot{f}^t : X \rightarrow Y$ that minimizes error for D weighted according to w

6: Calculate the error of $\ddot{f}^t$ on D_{src}: $\varepsilon_{src}^t = \sum_{j=1}^{n} \frac{\left[w_{src}^j\right] \cdot \mathbb{I}\left[y_{src_j} \neq \ddot{f}_j^t\right]}{\sum_{i=1}^{n}\left[w_{src}^i\right]}$

7: Calculate the error of $\ddot{f}^t$ on D_{tar}: $\varepsilon_{tar}^t = \sum_{j=1}^{m} \frac{\left[w_{tar}^j\right] \cdot \mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]}{\sum_{i=1}^{m}\left[w_{tar}^i\right]}$

8: Set $C^l = \left(1 - \varepsilon_{src}^l\right)$

9: Set $\beta_{tar} = \frac{1 - \varepsilon_{tar}^t}{\varepsilon_{tar}^t}$

10: $w_{src_i}^{t+1} = C^l w_{src_i}^t \beta_{src}^{\mathbb{I}\left[y_{src_i} \neq \ddot{f}_i^t\right]}$ where $i \in D_{src}$

11: $w_{tar_j}^{t+1} = w_{tar_j}^t \beta_{tar}^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]}$ where $j \in D_{tar}$

12: **end for**

ously, balanced classification is improved as the algorithm allocates higher weights to the subset of auxiliary instances that improve and balance the final classifier. The framework effectively combines the power of two boosting algorithms with AdaBoost [17] updating the target instances' weights and the weighted majority algorithm (WMA) [18], modified for balanced transfer, updating the source instances' weights to incorporate auxiliary knowledge and skew for balanced classification via transfer from the source domain.

The two algorithms operate separately and are only linked in:

1. Line 4 (normalization): Both algorithms require normalization. The combined normalization causes an anomaly that we will address in subsequent analysis.
2. Line 5: Infusing source with target for training is how transfer learning is induced from the auxiliary dataset.

The target instances are updated in lines 7, 9, and 11 as outlined by AdaBoost [17]. The weak learner in line 5 finds the separating hyperplane that forms the classification boundary and is used to calculate the target's error rate $\left(\varepsilon^t_{tar} < 0.5\right)$ in line 7. This error is used in line 9 to calculate $(\beta_{tar} > 1)$ which will then be used to update the target weights in line 11 as:

$$w^{t+1}_{tar_j} = w^t_{tar_j} \beta_{tar}^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}^t_j\right]} \tag{2}$$

Similar to AdaBoost, a misclassified target instance's weight increases after normalization and would thus gain more influence in the next iteration. Once boosting is completed, $(t = N)$, the weak classifiers $(\ddot{f}^t)$ weighted by β_{tar} are combined to construct a committee capable of nonlinear approximation.

The source instances are updated in lines 2 and 10 as done by the weighted majority algorithm [18]. The static WMA update rate $(\beta_{src} < 1)$ is calculated on line 2 and updates the source weights as:

$$w^{t+1}_{src_i} = w^t_{src_i} \beta_{src}^{\mathbb{I}\left[y_{src_i} \neq \ddot{f}^t_i\right]} \tag{3}$$

Contrary to AdaBoost, WMA decreases the influence of an instance that is misclassified and gives it a lower relative weight in the subsequent iterations. This property is beneficial for transfer learning since the source instance's contribution to the weak classifiers is dependent on its classification consistency. A consistently misclassified instance's weight converges,[4] and its influence diminishes in subsequent iterations. In Algorithm 1, the WMA update mechanism in Eq. (3) is actually modified in line 10 to incorporate the cost C^l for label-dependent transfer. This dynamic cost is calculated in line 8, and it promotes balanced transfer learning. Starting with equal initial weights and using standard weak classifiers that optimize for accuracy, these classifiers achieve low error rates for the majority and high error rate for the minority as they are overwhelmed with the majority label. The label-dependent cost, C^l, controls the rate of convergence of the source instances and hence the weights converge slower[5] for labels with high initial error rates (minority classes). As minority labels get higher normalized weights with each successive boosting iteration, the weak classifiers would subsequently construct more balanced separating hyperplanes. The $\frac{N}{2} \rightarrow N$ weak classifiers are used for the final output with the expectation that the most consistent and balanced mix of source instances would be used for learning the final classifier.

[4] All mentions of "convergence" refer to a sequence (weight) that converges to zero.

[5] Slower or decreased convergence rate means that a weight converges to zero with higher number of boosting iterations.

5 Theoretical analysis of the rare-transfer algorithm

We will refer to the cost $\left(C^l\right)$ on line 9 as the "Correction Factor" and prove in Sect. 5.1 that it prevents the source instances' weights from early convergence. This improves transfer learning and addresses the lack of training data in a rare dataset. In Sect. 5.2, we provide the motivation for balanced optimization and modify this "Correction Factor" to incorporate balanced optimization to simultaneously compensate for the lack of sufficient data and the class imbalance within a rare dataset.

5.1 Correction for transfer learning

Definition 1 Given k instances at iteration t with normalized weight w and update rate β, the sum of the weights after one boosting iteration with error rate (ε^t) is calculated as:

$$\sum_{i=1}^{k} w^{t+1} = kw^t(1-\varepsilon^t) + kw^t(\varepsilon^t)\beta \tag{4}$$

We will now explain this in more detail with the help of an example. Given $k = 10$ instances at iteration t with normalized weights $w = 0.1$, assume that weak learner $\ddot{f}$ correctly classifies 6 instances $(\varepsilon^t = 0.4)$. Sum of correctly classified instances at boosting iteration $t + 1$ is calculated as:

$$\begin{aligned}\sum_{y=\ddot{f}^t} w^{t+1} &= 0.1\beta^0 + 0.1\beta^0 + 0.1\beta^0 + 0.1\beta^0 + 0.1\beta^0 + 0.1\beta^0 \\ &= 6(w^t)\beta^0 \left\{\text{since}\left(w^t = 0.1\right)\right\} \\ &= 10(0.6)(w^t) \\ &= kw^t(1-\varepsilon^t) \ \left\{\text{since}\left(\text{k} = 10, \varepsilon^t = 0.4\right)\right\}\end{aligned} \tag{5}$$

On the other hand, the sum of misclassified instances at boosting iteration $t + 1$ is:

$$\begin{aligned}\sum_{y\neq\ddot{f}^t} w^{t+1} &= 0.1\beta^1 + 0.1\beta^1 + 0.1\beta^1 + 0.1\beta^1 \\ &= 4(w^t)\beta^1 \quad \left\{\text{since}\left(w^t = 0.1\right)\right\} \\ &= 10(0.4)(w^t)\beta \\ &= kw^t(\varepsilon^t)\beta \left\{\text{since}\left(k = 10, \varepsilon^t = 0.4\right)\right\}\end{aligned} \tag{6}$$

Thus, the sum of weights at boosting iteration "$t + 1$" is calculated as:

$$\begin{aligned}\sum_{i=1}^{k} w^{t+1} &= \sum_{y=\ddot{f}^t} w^{t+1} + \sum_{y\neq\ddot{f}^t} w^{t+1} \\ &= kw^t(1-\varepsilon^t) + kw^t(\varepsilon^t)\beta\end{aligned} \tag{7}$$

Proposition 1 *All source instances are correctly classified by the weak learner:*

$$y_{src_i} = \ddot{f}_i^t, \quad \forall i \in \{1, \ldots, n\} \tag{8}$$

Equation (8) is analogous to:

$$\sum_{i=1}^{n} w^{t+1} = nw_{src}^t\left(1-\varepsilon_{src}^t\right) + nw_{src}^t\left(\varepsilon_{src}^t\right)\beta_{src} = nw_{src}^t \tag{9}$$

This proposition is held true in subsequent analysis to theoretically demonstrate that even under ideal conditions with perfect auxiliary instances that consistently fit the classifiers, knowledge from these source instances is lost as their weights converge. A "Correction Factor" is calculated to conserve such instances, and it will be later demonstrated that this correction is inversely proportional to classifier's error and approaches unity (no correction needed) as error increases and the analysis deviates from this proposition.

Theorem 1 will examine the effect of the combined (source + target) normalization in line 4 of Algorithm 1.

Theorem 1 *If no correction is included in Algorithm 1, source weights will improperly converge even when all instances are correctly classified.*

Proof In the weighted majority algorithm, the weights are updated as:

$$w_{src}^{t+1} = \begin{cases} \frac{w_{src}^t}{\sum_{\{y_i=f_i\}} w_{src}^t + \sum_{\{y_i \neq f_i\}} \beta_{src} w_{src}^t} & y_{src} = \ddot{f}^t \\ \frac{\beta_{src} w_{src}^t}{\sum_{\{y_i=f_i\}} w_{src}^t + \sum_{\{y_i \neq f_i\}} \beta_{src} w_{src}^t} & y_{src} \neq \ddot{f}^t \end{cases} \tag{10}$$

Equation (10) demonstrates that the weights for source instances that are correctly classified should not change after normalization as:

$$w_{src}^{t+1} = \frac{w_{src}^t}{\sum_{i=1}^n w_{src_i}^t} = w_{src}^t \tag{11}$$

Without correction, the normalized source weights in Algorithm 1 are updated as:

$$w_{\text{src}}^{\text{t+1}} = \frac{w_{src}^t}{\sum_i^n w_{src_i}^t + \sum_j^m w_{tar_j}^t \beta_{tar}^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]}} \tag{12}$$

Equation (12) shows that, without correction, correctly classified source weights would still converge as:

$$\sum_j^m w_{tar_j}^t \beta_{tar}^{\left[y_{tar_j} \neq \ddot{f}_j^t\right]} = \sum_j^m w_{tar_j}^t \left(\frac{1-\varepsilon_{tar}^t}{\varepsilon_{tar}^t}\right)^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]} \tag{13}$$

Since all source weights persistently converge, all target weights would inversely increase since $(nw_{src}^{\text{t}} + mw_{tar}^t) = 1$. This will be referred to as "Weight Drift" since weight entropy drifts from source to target instances. "Weight Drift" negates transfer since the final classifier is comprised of the cascade of weak learners constructed in boosting iterations $\frac{N}{2} \rightarrow N$ (where the source instances' weights could have already converged). With converged source weights, Algorithm 1 becomes analogous to the standard AdaBoost algorithm with target instances and no transfer learning. □

Theorem 1 examined the cause of "Weight Drift" and Theorem 2 will outline the factors that control it.

Theorem 2 *For n source instances, the number of target training samples (m) affects the convergence rate and thus the "Weight Drift". "Weight Drift" is also stochastic since the rate of convergence at iteration t (without correction) is determined by that iteration's target error rate* (ε_{tar}^t).

Proof The fastest rate[6] of convergence is achieved by minimizing the weight for each subsequent boosting iteration (w_{src}^{t+1}) as:

$$\min_{m,n,\varepsilon_{tar}^{t}} \left(w_{src}^{t+1}\right) = \frac{w_{src}^{t}}{\max\limits_{m,n,\varepsilon_{tar}^{t}} \left\{ \sum_{i=1}^{n} w_{src_i}^{t} + \sum_{j=1}^{m} w_{tar_j}^{t} \left(\frac{1-\varepsilon_{tar}^{t}}{\varepsilon_{tar}^{t}} \right)^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]} \right\}} \tag{14}$$

Equation (14) shows that two factors can slow down the rate of convergence of correctly classified source instances:

1. Maximizing the weak learner's target error rate with $\varepsilon_{tar}^{t} \rightarrow 0.5$ (choosing an extremely weak learner or one that is only slightly better than random). Since the weak learner is weighted differently for each iteration, its error cannot be controlled and this factor will induce a stochastic effect.
2. Decreasing the number of target samples m, since rate of convergence accelerates when $m/n \rightarrow \infty$. Attempting to slow the improper rate of convergence by reducing the number of target instances is counterproductive as the knowledge from the removed instances would be lost. □

Theorem 2 demonstrated that a fixed cost cannot control the rate of convergence since the cumulative effect of m, n, and ε_{tar}^{t} is stochastic. A dynamic term has to be calculated to compensate for "Weight Drift" at every iteration. The calculation of a dynamic term is outlined in Theorem 3.

Theorem 3 *A correction factor of* $2\left(1-\varepsilon_{tar}^{t}\right)$ *can be applied to the source weights to prevent their "Weight Drift" and make the weights converge at a rate similar to that of the weighted majority algorithm.*

Proof Un-wrapping the WMA source update mechanism of Eq. (14), yields:

$$w_{src}^{t+1} = \frac{w_{src}^{t}}{\sum_{i=1}^{n} w_{src_i}^{t} + \sum_{j=1}^{m} w_{tar_j}^{t} \left(\frac{1-\varepsilon_{tar}^{t}}{\varepsilon_{tar}^{t}} \right)^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]}} = \frac{w_{src}^{t}}{n w_{src}^{t} + A + B} \tag{15}$$

where A and B are defined as:

$$\begin{aligned} A &= \text{Sum of correctly classified target weights at boosting iteration "}t+1\text{"} \\ &= m w_{tar}^{t} \left(1-\varepsilon_{tar}^{t}\right) \end{aligned} \tag{16}$$

$$\begin{aligned} B &= \text{Sum of misclassified target weights at boosting iteration "}t+1\text{"} \\ &= m w_{tar}^{t} \left(\varepsilon_{tar}^{t}\right) \beta_{tar}^{t} = m w_{tar}^{t} \left(\varepsilon_{tar}^{t}\right) \left(\frac{1-\varepsilon_{tar}^{t}}{\varepsilon_{tar}^{t}} \right)^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]} \\ &= m w_{tar}^{t} \left(1-\varepsilon_{tar}^{t}\right) \end{aligned} \tag{17}$$

Substituting for A and B, the source update is:

$$w_{src}^{t+1} = \frac{w_{src}^{t}}{n w_{src}^{t} + 2 m w_{tar}^{t} \left(1-\varepsilon_{tar}^{t}\right)} \tag{18}$$

[6] Faster or increased convergence rate means that a weight converges to zero with lower number of boosting iterations.

We will introduce and solve for a correction factor C^t to equate ($w_{src}^{t+1} = w_{src}^t$) for correctly classified instances (as per the WMA).

$$w_{src}^t = w_{src}^{t+1} = \frac{C^t w_{src}^t}{C^t n w_{src}^t + 2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)} \tag{19}$$

Solving for C^t:

$$C^t = \frac{2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)}{\left(1 - n w_{src}^t\right)} = \frac{2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)}{m w_{tar}^t} = 2\left(1 - \varepsilon_{tar}^t\right) \tag{20}$$

□

Adding this correction factor to line 10 of Algorithm 1 equates its normalized update mechanism to the weighted majority algorithm and subsequently prevents "Weight Drift". Theorem 4 examines the effect this factor has on the update mechanism of the target weights.

Theorem 4 *Applying a correction factor of* $2\left(1 - \varepsilon_{tar}^t\right)$ *to the source weights prevents "Weight Drift" and subsequently equates the target instances' weight update mechanism in Algorithm* 1 *to that of AdaBoost.*

Proof In AdaBoost, without any source instances ($n = 0$), target weights for correctly classified instances would be updated as:

$$\begin{aligned} w_{tar}^{t+1} &= \frac{w_{tar}^t}{\sum_{j=1}^m w_{tar_j}^t \left(\frac{1-\varepsilon_{tar}^t}{\varepsilon_{tar}^t}\right)^{\mathbb{I}\left[y_{tar_j} \neq \ddot{f}_j^t\right]}} \\ &= \frac{w_{tar}^t}{A + B} = \frac{w_{tar}^t}{2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)} = \frac{w_{tar}^t}{2(1)\left(1 - \varepsilon_{tar}^t\right)} \end{aligned} \tag{21}$$

Applying the "Correction Factor" to the source instances' weight update prevents "Weight Drift" and subsequently equates the target instances' weight update mechanism outlined in Algorithm 1 to that of AdaBoost since

$$\begin{aligned} w_{tar}^{t+1} &= \frac{w_{tar}^t}{n w_{src}^t + 2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)} = \frac{w_{tar}^t}{C^t n w_{src}^t + 2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)} \\ &= \frac{w_{tar}^t}{2\left(1 - \varepsilon_{tar}^t\right) n w_{src}^t + 2m w_{tar}^t \left(1 - \varepsilon_{tar}^t\right)} \\ &= \frac{w_{tar}^t}{2\left(1 - \varepsilon_{tar}^t\right)\left(n w_{src}^t + m w_{tar}^t\right)} = \frac{w_{tar}^t}{2\left(1 - \varepsilon_{tar}^t\right)(1)} \end{aligned} \tag{22}$$

□

It was proven that a dynamic cost can be incorporated into Algorithm 1 to correct for weight drifting from source to target instances. This factor would ultimately separate the source instance updates which rely on the WMA and β_{src}, from the target instance updates which rely on AdaBoost and ε_{tar}^t. With these two algorithms separated, they can be joined for transfer learning by infusing "best-fit" source instances to each successive weak classifier.

The "Correction Factor" introduced in this section allows for strict control of the source weights' rate of convergence, and this property will be exploited to induce balance to "Absolute Rarity". Balanced classifiers will be dynamically promoted by accelerating the rate of weight convergence of the majority label and slowing it for the minority label.

5.2 Correction for learning with "Absolute Rarity"

Instance transfer methods improve classification on a small dataset, but they also exacerbate the imbalance problem by constructing imbalanced classifiers. This outcome was even observed in generally balanced instance-transfer methods. It was noted by [27] that boosting for transfer learning sometimes yielded a final classifier that always predicted a single label. Dai et al. [16] re-sampled the data at each step to balance the class weight since they observed similar behavior. In this section, we examine the cause of this induced imbalance.

Proposition 2 *For a class imbalance problem, a standard classifier yields lower error rate for the majority label as compared to that of the minority since it optimizes:*

$$\min_{\varepsilon}(n\varepsilon) = \min_{\varepsilon^l}\left(\sum_{\forall l \in Y} n^l \varepsilon^l\right) \tag{23}$$

In a class imbalanced problem, where ($n^{l=\text{majority}} \gg n^{l=\text{minority}}$)*, a traditional classifier optimizing Eq.* (23) *can achieve high accuracy if it classifies all instances as majority instances. This proposition serves as a foundation for all imbalanced learning methods* [1,29,30].

Theorem 5 *In an imbalanced problem, the weighted majority algorithm, WMA, constructs a classifier where the minority instances' weights decrease exponentially with every boosting iteration.*

Proof A misclassified source instance at boosting iteration t is updated via the WMA update mechanism, and its $t+1$ weight is adjusted to: $w_{src}^{t+1} = \beta_{src} w_{src}^{t}$. The source update mechanism is set by β_{src} which is set to:

$$0 < \left[\beta_{src} = \frac{1}{1+\sqrt{\frac{2\ln(n)}{N}}}\right] < 1 \tag{24}$$

Since $\beta_{src} < 1$, a misclassified source instance's weight would converge after normalization. Since weak classifiers at initial boosting iterations, with equally initialized weights, yield high error rates for minority labels (Proposition 2), the minority label's weights would subsequently have less influence on the $t+1$ classifier and would accelerate the rate of convergence as

$$\begin{aligned} w_{src}^{t+1} &\geq w_{src}^{t} \qquad if\left(y_{src} = \ddot{f}^t\right) \\ w_{src}^{t+1} &< w_{src}^{t} \qquad if\left(y_{src} \neq \ddot{f}^t\right) \end{aligned} \tag{25}$$

Ignoring normalization, the minority label's weights decrease exponentially as:

$$\begin{aligned} w_{src}^{t+1} &\approx \beta_{src} w_{src}^{t} \\ w_{src}^{t+2} &\approx \beta_{src} w_{src}^{t+1} \approx \beta_{src}\beta_{src} w_{src}^{t} \\ &\vdots \\ w_{src}^{t+k} &\approx \beta_{src}^{k} w_{src}^{t} \end{aligned} \tag{26}$$

Since the final classifier in Algorithm 1 is computed from the cascade of learners constructed in iterations $\frac{N}{2} \rightarrow N$, where the minority source weights could have already converged, the final output would be extremely imbalanced as it will have added only majority weights. □

Conversely, updating the target instances via the AdaBoost update mechanism improves the performance on an imbalanced dataset particularly if the final classifier is computed using only the $\frac{N}{2} \rightarrow N$ boosting iterations. A misclassified target instance at boosting iteration t is updated via the AdaBoost update mechanism, and its $t+1$ weight is adjusted to: $w_{tar}^{t+1} = \beta_{tar} w_{tar}^{t}$. The target update for a misclassified instance's weight is dependent on β_{tar} where

$$1 < \left[\beta_{tar} = \frac{1 - \varepsilon_{tar}^{t}}{\varepsilon_{tar}^{t}} \right] < \infty \tag{27}$$

Since $\beta_{tar} > 1$, a misclassified target instance's weight would increase after normalization and the minority label's weights would in turn have more influence on the $t+1$ classifier and bias the classifier to improve learning on the minority as:

$$\begin{aligned} w_{tar}^{t+1} &< w_{tar}^{t} \quad if \quad \left(y_{tar} = \ddot{f}^{t} \right) \\ w_{tar}^{t+1} &\geq w_{tar}^{t} \quad if \quad \left(y_{tar} \neq \ddot{f}^{t} \right) \end{aligned} \tag{28}$$

Since the final classifier is computed from the cascade of learners constructed in iterations $\frac{N}{2} \rightarrow N$, where the minority label's instances have increased weights to compensate for the lack of its samples, the final output would be more balanced.

5.3 Label space optimization

Instance-transfer can improve learning with "Absolute Rarity" as it compensates for the lack of training examples with a selective set of samples from an auxiliary domain. Since instance-transfer can also induce imbalance (as proved in the previous section), intuitive results will require a balanced optimization technique to address the class imbalance in "Absolute Rarity". Figure 4 motivates for optimization with balanced measures to improve classification in imbalanced datasets. The two classifiers in Fig. 4 are optimized with different

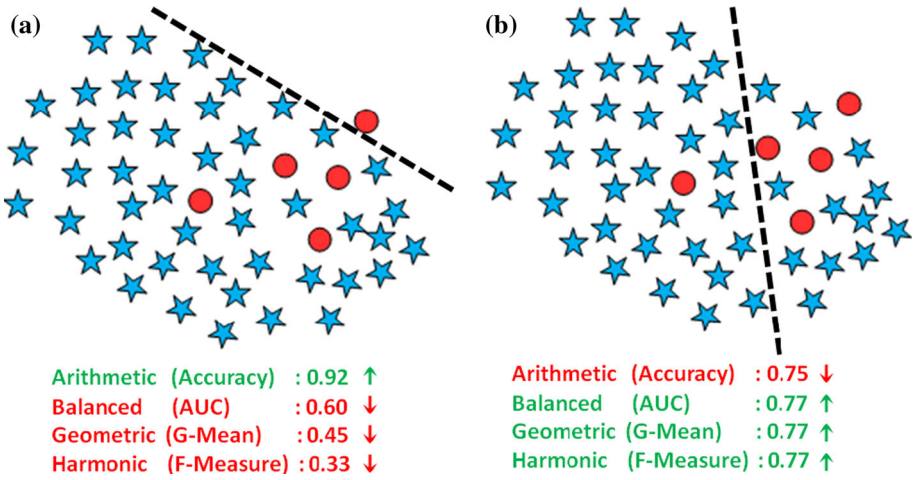

Fig. 4 **a** Algorithm minimizing arithmetic error. **b** Algorithm minimizing geometric/balanced/harmonic error

types of accuracy measures[7] where Fig. 4a minimizes the Arithmetic error while Fig. 4b minimizes the (Geometric [31]/ Balanced [32]/Harmonic [33]) errors. This example shows that given a constrained classifier (linear classifier in this example), the algorithm in Fig. 4b obtains more intuitive results with a degraded arithmetic accuracy and an improved balanced (Geometric/Balanced/Harmonic) accuracy. Theorems 6, 7, and 8 prove that minimizing a label-dependent measure improves balanced statistical measures while Theorem 9 shows that a label-dependent optimization can improve the balanced statistics for "Absolute Rarity" in Algorithm 1.

Theorem 6 *Maximizing the Balanced Accuracy (BAC) is equivalent to minimizing the sum of label-dependent errors independent of the number of samples within each class:*

$$\max_{\varepsilon^l}(BAC) = \min_{\varepsilon^l} \sum_{l \in Y} \varepsilon^l_{src}$$

Proof To prove theorem 6, we will start with a binary labeled example and extend to general form. With no optimization of the prediction threshold of a binary classifier (classifier threshold at a pre-set level), the Area under the ROC Curve (AUC) is equivalent to Balanced Accuracy (BAC) [34]. This Balanced Accuracy is the average accuracy of each class and in turn equates to the average of sensitivity and specificity. It is calculated as follows:

$$AUC = BAC = \frac{1}{2}(Sensitivity + Specificity)$$

$$= \frac{1}{2}\left[\left(\frac{TruePositive}{TruePositive + FalseNegative}\right) + \left(\frac{TrueNegative}{TrueNegative + FalsePositive}\right)\right]$$

$$= \sum_{l \in Y} \frac{0.5\left(\sum_{i=1}^{n}\left(y_i^l = f_i^l\right)\right)}{\sum_{i=1}^{n}\left(y_i^l = f_i^l\right) + \sum_{i=1}^{n}\left(y_i^l \neq f_i^l\right)} = \sum_{l \in Y} \frac{0.5\left(n^l\left(1-\varepsilon^l\right)\right)}{n^l\left(1-\varepsilon^l\right) + n^l\left(\varepsilon^l\right)} = 0.5\left(\sum_{l \in Y}\left(1-\varepsilon^l\right)\right) \tag{29}$$

Equation (29) can be maximized as follows:

$$\max_{\varepsilon^l}(BAC) = \max_{\varepsilon^l}\left(0.5\left(\sum_{l \in Y}\left(1-\varepsilon^l\right)\right)\right) = \min_{\varepsilon^l}\left(\sum_{l \in Y}\varepsilon^l\right) \tag{30}$$

□

The optimization problem in Eq. (30) is a constrained optimization problem which is minimized as follows:

$$\min_{\varepsilon^l} \sum_{l \in Y} \varepsilon^l$$
$$\text{s.t.} \sum_{\forall l \in Y} n^l \varepsilon^l = \varepsilon \tag{31}$$

Theorem 7 *Maximizing the Geometric Mean (G-Mean) is equivalent to minimizing the product of label-dependent errors and is independent of the number of samples within each class:*

$$\max_{\varepsilon^l}(G - Mean) = \min \prod_{l \in Y} \varepsilon^l_{src}$$

[7] The Up/Down arrow next to each error measure signifies that an algorithm produced better/worse results in comparison with the other algorithm.

Proof Similar to Theorem 6, we start with a binary labeled example and extend to general form:

$$\begin{aligned} G-Mean &= \sqrt{(Sensitivity)\,(Specificity)} \\ &= \sqrt{\left(\frac{TruePositive}{TruePositive+FalseNegative}\right)\left(\frac{TrueNegative}{TrueNegative+FalsePositive}\right)} \\ &= \sqrt{\prod_{l\in Y}\frac{\sum_{i=1}^{n}\left(y_i^l=f_i^l\right)}{\sum_{i=1}^{n}\left(y_i^l=f_i^l\right)+\sum_{i=1}^{n}\left(y_i^l\neq f_i^l\right)}} \\ &= \sqrt{\prod_{l\in Y}\frac{n^l\left(1-\varepsilon^l\right)}{n^l\left(1-\varepsilon^l\right)+n^l\left(\varepsilon^l\right)}} = \sqrt{\prod_{l\in Y}\left(1-\varepsilon^l\right)} \end{aligned} \tag{32}$$

Maximizing the statistic in Eq. (32), we have

$$\max_{\varepsilon^l}\left(G-Mean\right) = \max_{\varepsilon^l}\left(\sqrt{\prod_{l\in Y}\left(1-\varepsilon^l\right)}\right) = \min_{\varepsilon^l}\left(\prod_{l\in Y}\varepsilon^l\right) \tag{33}$$

□

Similar to Eq. (31), the optimization problem in Eq. (33) is a constrained optimization problem and is minimized by

$$\begin{aligned} &\min_{\varepsilon^l}\prod_{l\in Y}\varepsilon^l \\ &\text{s.t.}\sum_{\forall l\in Y}n^l\varepsilon^l=\varepsilon \end{aligned} \tag{34}$$

Since both Eqs. (31) and (34) are constrained by the classifier's error rate, modifying the weak learner to improve classification on one label can degrade classification on the other label (Fig. 4).

Theorem 8 *An improved G-Mean coupled with no degradation in the BAC will improve the F-Measure.*

Proof The harmonic mean of sensitivity and specificity is a particular realization of the F-measure [33] and is maximized as

$$\begin{aligned} f-measure &= \left[\frac{2\,(Sensitivity)\,(Specificity)}{Sensitivity+Specificity}\right] \\ &= \left[\frac{\prod_{l\in Y}\frac{n^l\left(1-\varepsilon^l\right)}{n^l\left(1-\varepsilon^l\right)+n^l\left(\varepsilon^l\right)}}{\sum_{l\in Y}\frac{0.5\left(n^l\left(1-\varepsilon^l\right)\right)}{n^l\left(1-\varepsilon^l\right)+n^l\left(\varepsilon^l\right)}}\right] = \left[\frac{\prod_{l\in Y}\left(1-\varepsilon^l\right)}{0.5\left(\sum_{l\in Y}\left(1-\varepsilon^l\right)\right)}\right] \end{aligned} \tag{35}$$

Equation (35) can be optimized as

$$\max_{\varepsilon^l}(f-measure) = \max_{\varepsilon^l}\left[\frac{\prod_{l\in Y}\left(1-\varepsilon^l\right)}{0.5\left(\sum_{l\in Y}\left(1-\varepsilon^l\right)\right)}\right] = \max_{\varepsilon^l}\left[\frac{(G-Mean)^2}{BAC}\right] = \min_{\varepsilon^l}\left[\frac{\left(\prod_{l\in Y}\varepsilon^l\right)}{\sum_{l\in Y}\varepsilon^l}\right] \tag{36}$$

□

Equation (36) proves a classifier that improves G-Mean (via balance) with no degradation in BAC (via Transfer) improves the F-measure.

Theorem 9 *In an imbalanced dataset, a label-dependent update mechanism can improve the G-Mean without degrading the BAC performance.*

Proof In a balanced learning problem, all labels have an equal effect on BAC and G-Mean but as the label space gets more imbalanced, $\frac{n^{l=\text{majority}}}{n^{l=\text{minority}}} \to \infty$, the contribution of the minority label's error rate to the classifier's overall accuracy can thus be approximated as:

$$\sum_{\forall l\in Y} n^l\varepsilon^l \approx \sum_{l\in majority} n^l\varepsilon^l \tag{37}$$

Equation (37) demonstrates that biasing the classifier to favor the minimization of the minority label, in an imbalanced dataset, has minimal effect on the overall accuracy and the balanced arithmetic mean will not be degraded since the increased error of the majority label is negated by the decreased error of the minority label. On the other hand, G-Mean is the balanced geometric mean and is significantly improved if balance is induced. □

5.3.1 Optimization for "Absolute Rarity"

Using definition 1, the sum of source instances' weight is monotonically decreasing as:

$$\begin{aligned} nw_{src}^{t+1} &= nw_{src}^t\left[1+\varepsilon_{src}^t\left(\beta_{src}-1\right)\right] \\ nw_{src}^{t+1} &\le nw_{src}^t \text{ since } \left(\beta_{src}<1,\ \varepsilon_{src}^t\ge 0\right) \end{aligned} \tag{38}$$

Similarly, the target instances' weights are monotonically increasing:

$$\begin{aligned} mw_{tar}^{t+1} &= mw_{tar}^t\left[1+\varepsilon_{tar}^t\left(\beta_{tar}-1\right)\right] \\ \text{mw}_{tar}^{t+1} &\le mw_{tar}^t \text{ since } \left(\beta_{tar}>1,\ \varepsilon_{tar}^t\ge 0\right) \end{aligned} \tag{39}$$

Line 4 of "Rare Transfer" normalizes the sum of all weights and thus, all source weights are monotonically converging. On the other hand, Theorem (5) demonstrated that the minority sources' weights converge faster than the majority sources' weights. To improve balanced classification, we include a "Label-Dependent Correction Factor" to dynamically slow the convergence of the source instances' weights while simultaneously reducing the differential in the error between the minority and majority label. It is set to:

$$C^l = \left(1-\varepsilon_{src}^l\right) \tag{40}$$

This factor dynamically slows convergence for the label with a higher error since the convergence rate is inversely correlated to the error. Biasing each label's weights allows "Rare Transfer" to steer for the construction of a final classifier that includes a best-fit set of auxiliary samples and has an equal error on all labels.

6 Empirical analysis

In this section, we provide empirical validation of our theorems. The first experiment demonstrates how a "Correction Factor" fixes the problem of "Weight Drift". The second experiment examines the effect of "Label-Dependent" optimization on imbalanced learning.

6.1 "Weight Drift" and "Correction Factor"

The first experiment demonstrates the effect of "Weight Drift" on source and target weights. In Fig. 5a, the number of instances was constant ($n = 10{,}000$, $m = 200$), the source error rate was set to zero (as per Proposition 1) and the number of boosting iterations was set to $N = 20$. According to the WMA, the weights should not change when $\varepsilon^t_{src} = 0$. The ratio of the weights (with and without correction) to the weights of the WMA are plotted at different boosting iterations and with different target error rates $\varepsilon^t_{tar} \in \{0.1, 0.2, 0.3\}$. This experiment validates the following theorems:

1. With correction, source weights converge even when correctly classified.
2. Applying our "Correction Factor" equates the weight update of Algorithm 1 to the WMA.
3. If correction is not applied, strong classifiers cause weights to converge at a faster rate than weak ones (Theorem 2).

The figure also demonstrates that for a strong learner with $\varepsilon^t_{tar} \approx 0.1$, if no correction is applied, an "un-corrected" update mechanism would not transfer knowledge from all 10,000 source instances although they were never misclassified. The final classifier uses boosting iterations $N/2 \rightarrow N$, or $10 \rightarrow 20$, where the weights of ideal source instances would have already lost over 85 % of their value. Correction conserved these instances' weights and thus helpful source instances would improve classification.

The second experiment validates the effect of the number of target instances, m, on the convergence rate (Theorem 2). The number of source instances was set ($n = 1000$), while the number of target instances was varied $\frac{m}{n} \in \{1, 2, 5\,\%\}$ and plotted for $\varepsilon^t_{tar} \in \{0.1, \ldots, 0.5\}$. The plot in Fig. 5b shows how the source weights converge after a single boosting iteration and it can be observed that the rate of convergence is affected by m/n and the error rate ε_{tar} (which is also related to m). It can also be observed that as the error rate increases ($\varepsilon \rightarrow 0.5$), less correction is required as the improper convergence rate approaches the correct WMA rate. This is expected since the "Correction Factor" $\left(C = 2\left(1 - \varepsilon^t_{tar}\right)\right)$ is inversely proportional to ε^t_{tar} and its impact reaches unity (No Correction) as the target error rate increases.

$$\lim_{\varepsilon^t_{tar} \rightarrow 0.5} \{C\} = \lim_{\varepsilon^t_{tar} \rightarrow 0.5} \left\{2\left(1 - \varepsilon^t_{tar}\right)\right\} \approx \lim_{\varepsilon^t_{src} \rightarrow 0.5} \left\{2\left(1 - \varepsilon^t_{src}\right)\right\} = 1 \tag{41}$$

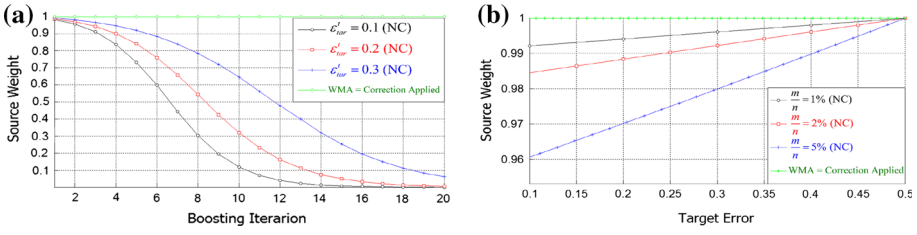

Fig. 5 The weights (relative to the WMA) for ideal source instances. **a** For 20 iterations with different error. **b** For 1 iteration with different target instances and error. (*NC* no correction)

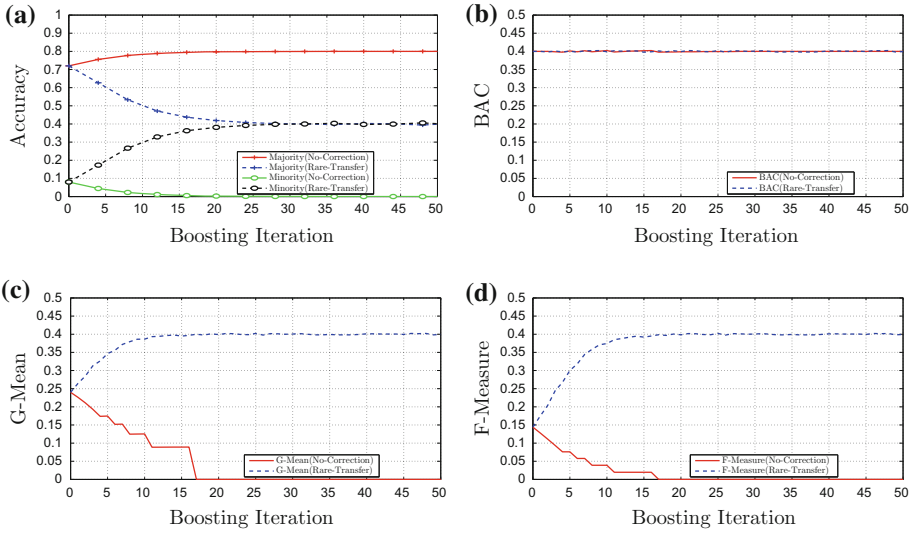

Fig. 6 The effect of "Rare Correction" using various evaluation metrics. **a** Label accuracy. **b** BAC. **c** G-mean. **d** F-measure

This is an important property because "Weight Drift" is most detrimental to learning at low error rates (where Proposition 1 was set).

It should be noted that for both plots in Fig. 5, the weight lost by the source instances is drifting to the target instances. The plots for the target weights would look inversely proportional to the plots in Fig. 5 since $\sum_{i=1}^{n} w_{src_i}^t + \sum_{j=1}^{m} w_{tar_j}^t = 1$.

6.2 "WMA Imbalanced Drift" and "Rare Correction Factor"

This section presents an empirical validation of Theorems 5, 6, 7, 8, and 9. A binary labeled classification problem was simulated with 900 majority instances, 100 minority instances and the weak classifier error rate was set to ($\varepsilon = 0.2$). Since this an imbalanced dataset and the weak classifier is weighted, error rate (ε^l) was correlated with the label's relative weight as follows: $\varepsilon^l = \frac{\varepsilon \sum_{i \in l} w^i}{\sum w}$.

In Fig. 6a, we plot the accuracy for both labels and demonstrate that applying a label-dependent correction factor to the weight update mechanism induces balance, while the un-corrected WMA update mechanism minimizes only the majority label's error and causes imbalance. This is reflected in the statistical measures as Fig. 6b shows that inducing balance causes no change in BAC while significantly improving G-Mean in Fig. 6c. The improved G-Mean coupled with no degradation in BAC is reflected in the improved F-Measure in Fig. 6d.

7 Real-world experimental results

We will now provide the details on the performance of various algorithms under different evaluation metrics using real-world datasets.

7.1 Dataset description

A Detailed description of the datasets used in our experiments is provided in Table 1.

(i) *Healthcare demographics* We collected Heart Failure (HF) patient data from the Henry Ford Health System (HFHS) in Detroit. This dataset contains records for 8913 unique patients who had their first hospitalization with primary HF diagnosis. The goal is to predict if a patient will be re-admitted within 30 days after being discharged from the

Table 1 Description of the datasets used in our experiments

Dataset	Features	Source Majority	Source Minority	Target Majority	Target Minority
		AA	AA	CA	CA
HF	Nu: 2	NReH	ReH	NReH	ReH
(Race)	No: 20	4468 (78.0 %)	1026 (17.9 %)	≈183 (3.2 %)	≈ 50 (0.9 %)
		Over 50	Over 50	Under 50	Under 50
HF	Nu: 1	NReH	ReH	NReH	ReH
(Age)	No: 21	4513 (75.4 %)	1182 (19.8 %)	≈ 241 (4.0 %)	≈ 50 (0.8 %)
		Male	Male	Female	Female
HF	Nu: 2	NReH	ReH	NReH	ReH
(Gender)	No: 20	3366 (75.7 %)	818 (18.4 %)	≈211 (4.8 %)	≈50 (1.1 %)
		Muslim	Muslim	Non-Muslim	Non-Muslim
Emp	Nu: 5	Un-employed	Employed	Un-employed	Employed
(Rel)	No:	955 (74 %)	298 (23 %)	15 (0.02 %)	7 (0.006 %)
		Male	Male	Female	Female
Park	Nu: 19	UPDRS $\geq$ 10	UPDRS $<$ 10	UPDRS $\geq$ 10	UPDRS $<$ 10
(Gender)	No: 0	3732 (89 %)	276 (8 %)	112 (0.03 %)	13(0.003 %)
REC		rec	talk	rec	talk
vs	Nu: 500	.autos	.politics.guns	.sports.baseball	.politics.mideast
TALK	No: 0	.motorcycles	.politics.misc	.sports.hockey	.religion.misc
		1009	20,50,101	472,453,393	10,23,39
			2, 5, 10 %	2, 5, 10 %	2, 5, 10 %
REC		rec	sci	rec	sci
vs	Nu: 500	.autos	.sci.crypt	.motorcycles	.sci.electronics
SCI	No: 0	.sports.baseball	.sci.space	.sports.hockey	.sci.med
		1187	24,59,119	486,518,543	10,26,55
			2, 5, 10 %	2, 5, 10 %	2, 5, 10 %
SCI		sci	talk	sci	talk
vs	Nu: 500	.sci.med	.politics.misc	.sci.crypt	.politics.guns
TALK	No: 0	.sci.electronics	.religion.misc	.sci.space	.politics.mideast
		840	17,42,84	513,529,507	10,26,51
			2, 5, 10 %	2, 5, 10 %	2, 5, 10 %

For brevity, we used several acronyms which are explained here
Nu numeric, *No* nominal, *CA* Caucasian, *AA* African American, *NReH* not re-hospitalized, *ReH* re-hospitalized, *Emp* employment, *Rel* religion, *Park* parkinson

hospital and to apply the model to rural hospitals or to demographics with less data [35]. Re-hospitalization for HF occurs in around one-in-five patients within 30 days of discharge and is disproportionately distributed across the US population with significant disparities based on gender, age, ethnicity, geographic area, and socioeconomic status [36]. Other non-demographic features included length of hospital stay, ICU stay and dichotomous variables for whether a patient was diagnosed with diabetes, hypertension, peripheral vascular disease, transient ischemic attack, heart failure, chronic kidney disease, coronary artery disease, hemodialysis treatment, cardiac catheterization, right heart catheterization, coronary angiography, balloon pump, mechanical ventilation or general intervention. The average results with 50 minority samples (patient was re-hospitalized) is reported.

(ii) *Employment dataset* This dataset is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey [37]. We used the dataset [14] to predict if a non-Muslim woman is employed based on her demographic and socio-economic characteristics. In the training set, only 22 of the 1275 were not Muslim and only 7 of them were employed.

(iii) *Parkinson dataset* This dataset [38] is composed of a range of biomedical voice measurements from people with early-stage Parkinson's disease. The goal is to predict if a female patient's score on the Unified Parkinson's Disease Rating Scale [39] is high (UPDRS $\geq$ 10) or low (UPDRS $<$ 10). In the training set, only 125 of the 3732 participants were female and only 13 of them had a low UPDRS score.

(iv) *Text dataset* 20 Newsgroups[8] is a popular text collection that is partitioned across 20 groups with 3 cross-domain tasks and a two-level hierarchy as outlined in [40]. We used Term FrequencyInverse Document Frequency (TF-IDF) [41] to maintain around 500 features and imbalanced the dataset to generate a high-dimensional, small and imbalanced dataset.

7.2 Experiment setup

AdaBoost [17] was used as the standard baseline algorithm for comparison. We applied SMOTE [42], with 5 nearest neighbors ($k = 5$), before boosting to compare with an imbalanced classification method (SMOTE-AdaBoost). TrAdaBoost [16] was used as the baseline transfer learning algorithm. The non-transfer reference algorithms were trained with the target-only set and with the combined (target+source) set. Thirty boosting iterations were experimentally proven sufficient for training.

Base learner $\left(\grave{f}\right)$ We did not use decision stumps as weak learners since most data belong to the source and it was not possible to keep the target error below 0.5 (as mandated by AdaBoost) for more than a few iterations. A strong classifier, full classification tree without pruning, is applied with a top-down approach where the tree is trimmed at the first level to achieve $\varepsilon^t_{tar} < 0.5$.

Cross validation Small datasets are prone to over-fit and terminate boosting and thus all algorithms were restarted with a new cross validation fold when any algorithm terminated before reaching 30 iterations. Random sub-sampling cross validation [43] was applied, and each statistic was tabulated with the macro-average [44] of 30 runs. Plots with two imbalance ratios across a variable size of minority samples are also presented.

[8] http://people.csail.mit.edu/jrennie/20Newsgroups/.

7.3 Experimental results

This section presents the classification results for the different balanced learning measures.

7.3.1 BAC results

The BAC results presented in Table 2 show that Rare-Transfer improved the Balanced Accuracy. The improved performance is consistent even when the addition of auxiliary data seemed to degrade the performance as evident in the 20 Newsgroups dataset. This is proof that the "transfer learning" objective in our algorithm improved learning with *only* the best set of auxiliary instances. Figure 7 demonstrates that the improved performance is consistent across different datasets, imbalance ratios and absolute number of minority samples.

7.3.2 G-mean results

The results presented in Table 3 confirm that Rare-Transfer *significantly* improved the Geometric Mean. The results on the 20 Newsgroups (2 %) dataset demonstrate improved

Table 2 Comparison of balanced accuracy values on real-world datasets

	AdaBoost (Target)	AdaBoost (Src + Tar)	SMOTE (Target)	SMOTE (Src + Tar)	TrAda Boost	Rare Transfer
HF(Gender)	0.518	0.512	0.524	0.549	0.502	0.562
HF(Race)	0.521	0.509	0.529	0.547	0.504	0.563
HF(Age)	0.526	0.511	0.538	0.535	0.503	0.556
Rec-Sci(2 %)	0.564	0.571	0.566	0.569	0.558	0.594
Sci-Talk(2 %)	0.544	0.541	0.544	0.540	0.543	0.571
Rec-Talk(2 %)	0.569	0.534	0.577	0.547	0.561	0.610
Rec-Sci(5 %)	0.635	0.622	0.635	0.645	0.608	0.664
Sci-Talk(5 %)	0.602	0.591	0.607	0.596	0.582	0.632
Rec-Talk(5 %)	0.635	0.569	0.642	0.602	0.609	0.672
Rec-Sci(10 %)	0.696	0.680	0.699	0.706	0.649	0.706
Sci-Talk(10 %)	0.662	0.639	0.672	0.647	0.620	0.679
Rec-Talk(10 %)	0.714	0.628	0.722	0.673	0.640	0.736
Employment	0.506	0.509	0.510	0.523	0.524	0.513
Parkinson	0.649	0.659	0.761	0.762	0.862	0.885

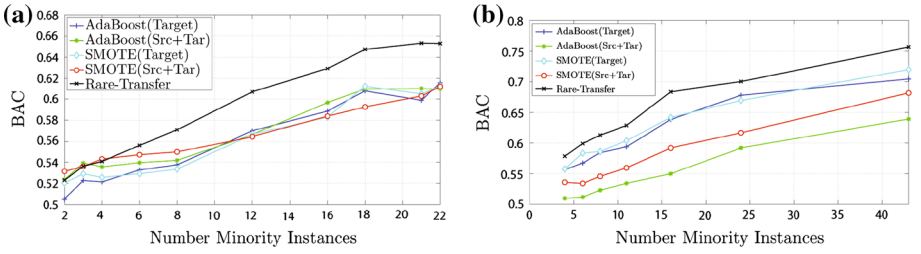

Fig. 7 BAC values when the number of minority samples is varied. **a** REC-VS-SCI (2 %). **b** REC-VS-TALK (10 %)

Table 3 Comparison of G-mean values on real-world datasets

	AdaBoost (Target)	AdaBoost (Src + Tar)	SMOTE (Target)	SMOTE (Src + Tar)	TrAda Boost	Rare Transfer
HF(Gender)	0.336	0.212	0.444	0.459	0.117	0.518
HF(Race)	0.366	0.178	0.467	0.478	0.145	0.540
HF(Age)	0.346	0.141	0.447	0.357	0.164	0.478
Rec-Sci(2 %)	0.324	0.362	0.338	0.379	0.384	0.430
Sci-Talk(2 %)	0.270	0.271	0.277	0.292	0.339	0.380
Rec-Talk(2 %)	0.343	0.221	0.378	0.293	0.387	0.460
Rec-Sci(5 %)	0.500	0.492	0.501	0.561	0.512	0.592
Sci-Talk(5 %)	0.433	0.423	0.446	0.464	0.471	0.541
Rec-Talk(5 %)	0.502	0.340	0.516	0.450	0.491	0.591
Rec-Sci(10 %)	0.615	0.605	0.623	0.671	0.581	0.674
Sci-Talk(10 %)	0.563	0.531	0.584	0.575	0.531	0.637
Rec-Talk(10 %)	0.647	0.483	0.661	0.597	0.569	0.702
Employment	0.422	0.452	0.467	0.483	0.331	0.378
Parkinson	0.518	0.570	0.715	0.741	0.841	0.874

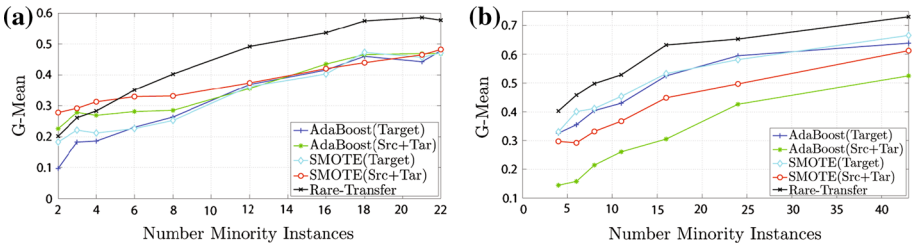

Fig. 8 G-mean values when the number of minority samples is varied. **a** REC-VS-SCI (2 %). **b** REC-VS-TALK (10 %)

performance with severe label imbalance and an extremely high features/samples ratio (10 minority samples, ≈500 majority samples, 500 features). Figure 8 shows that Rare-Transfer consistently yield superior results even after the non-transfer algorithms construct representative hypotheses with more training samples.

7.3.3 F-measure results

The F-Measure results are presented in Table 4 and demonstrate that Rare-Transfer constructs a more balanced classifier. The improvements are consistent at different imbalance ratios and sample sizes as shown in Fig. 9. The figures also demonstrate that the classification models can construct classifiers that are more balanced when the overall size of the training set increases.

7.4 Discussion and possible extensions

Traditional imbalanced modifications including SMOTEBoost [45], over or under sampling [46] followed by transfer [47] or cost-sensitive learning [25] are a straight-forward

Table 4 Comparison of F-measure values on real-world datasets

	AdaBoost (Target)	AdaBoost (Src + Tar)	SMOTE (Target)	SMOTE (Src + Tar)	TrAda Boost	Rare Transfer
HF(Gender)	0.217	0.088	0.375	0.384	0.307	0.478
HF(Race)	0.256	0.062	0.412	0.418	0.055	0.519
HF(Age)	0.228	0.039	0.372	0.238	0.063	0.411
Rec-Sci(2 %)	0.208	0.240	0.218	0.258	0.266	0.331
Sci-Talk(2 %)	0.225	0.115	0.256	0.174	0.223	0.363
Rec-Talk(2 %)	0.149	0.144	0.152	0.164	0.254	0.275
Rec-Sci(5 %)	0.405	0.393	0.408	0.490	0.434	0.534
Sci-Talk(5 %)	0.407	0.228	0.424	0.349	0.382	0.527
Rec-Talk(5 %)	0.324	0.309	0.343	0.365	0.405	0.473
Rec-Sci(10 %)	0.550	0.541	0.560	0.638	0.512	0.645
Sci-Talk(10 %)	0.590	0.389	0.609	0.536	0.472	0.671
Rec-Talk(10 %)	0.484	0.445	0.514	0.513	0.520	0.600
Employment	0.276	0.408	0.325	0.457	0.188	0.378
Parkinson	0.404	0.504	0.552	0.733	0.702	0.749

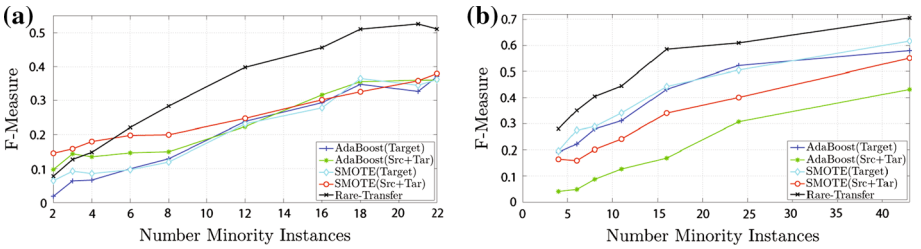

Fig. 9 F-measure values when the number of minority samples is varied. **a** REC-VS-SCI (2 %). **b** REC-VS-TALK (10 %)

extension to Algorithm 1 and can further improve classification. Improvements, even minor, from methods optimized specifically for "Absolute Rarity" can have significant practical impact within real-world domains where only human expertise are currently applicable. For example, rare diseases are a substantial public health burden as extremely low percentage of people have a rare disease at some point and currently no global registry or classification codes exist. Rare methods can improve learning and encourage data collection and warehousing. Future work will test our approach using multi-resolution methods in distributed environments with multiple source sets [48,49].

8 Conclusion

Learning with "Absolute Rarity" is an important and understudied area of research which is investigated in this paper. We discussed the impediments and proposed the first classification method optimized specifically for the problem of "Absolute Rarity". Our framework simultaneously compensated for the lack of data and the presence of class imbalance using

a transfer learning paradigm with a balanced statistics objective. We theoretically analyzed and empirically verified our work and demonstrated its effectiveness with several real-world domains. We proposed possible extensions and motivated for more research for a problem with significant social and financial impact.

Acknowledgments This work was supported in part by the National Cancer Institute of the National Institutes of Health under Award Number R21CA175974 and the US National Science Foundation grants IIS-1231742, IIS-1242304, and IIS-1527827. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH and NSF.

References

1. He H, Garcia E (2009) Learning from imbalanced data. IEEE Trans Knowl Data Eng 21(9):1263–1284
2. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
3. Li Y, Vinzamuri B, Reddy CK (2015) Constrained elastic net based knowledge transfer for healthcare information exchange. Data Min Knowl Discov 29(4):1094–1112
4. Waters D (2009) Spam overwhelms e-mail messages. BBC News. http://news.bbc.co.uk/2/hi/technology/7988579.stm
5. Halliday J (2011) Email spam level bounces back after record low. The Guardian; Retrieved 2011-01-11
6. Kearns MJ, Vazirani UV (1994) An introduction to computational learning theory. MIT Press, Cambridge
7. Mitchell T (1997) Machine learning. McGraw-Hill, New York
8. Weiss GM (2004) Mining with rarity: a unifying framework. SIGKDD Explor Newsl 6(1):7–19
9. He J (2010) Rare category analysis. Ph.D. thesis; Carnegie Mellon University
10. Banko M, Brill E (2001) Scaling to very very large corpora for natural language disambiguation. In: Proceedings of the 39th annual meeting on association for computational linguistics. Association for Computational Linguistics, pp 26–33
11. Weiss GM, Provost F (2003) Learning when training data are costly: the effect of class distribution on tree induction. J Artif Intell Res 19(1):315–354
12. Japkowicz N, Stephen S (2002) The class imbalance problem: a systematic study. Intell Data Anal 6(5):429–449
13. Bradley AP (1997) The use of the area under the roc curve in the evaluation of machine learning algorithms. Pattern Recognit 30(7):1145–1159
14. Frank A, Asuncion A (2010) UCI machine learning repository. http://archive.ics.uci.edu/ml
15. Davis J, Goadrich M (2006) The relationship between precision-recall and roc curves. In: Proceedings of the 23rd international conference on machine learning. ACM, pp 233–240
16. Dai W, Yang Q, Xue GR, Yu Y (2007a) Boosting for transfer learning. In: Proceedings of the international conference on machine learning, pp 193–200
17. Freund Y, Schapire RE (1995) A decision-theoretic generalization of on-line learning and an application to boosting. In: Proceedings of the second European conference on computational learning theory, pp 23–37
18. Littlestone N, Warmuth MK (1989) The weighted majority algorithm. In: Proceedings of the 30th annual symposium on foundations of computer science, pp 256–261
19. Yao Y, Doretto G (2010) Boosting for transfer learning with multiple sources. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 1855–1862
20. Al-Stouhi S, Reddy CK, Lanfear DE (2012) Label space transfer learning. In: IEEE 24th international conference on tools with artificial intelligence, ICTAI 2012, Athens, Greece, November 7–9, 2012, pp 727–734
21. Vieriu RL, Rajagopal A, Subramanian R, Lanz O, Ricci E, Sebe N et al (2012) Boosting-based transfer learning for multi-view head-pose classification from surveillance videos. In: Proceedings of the 20th European signal processing conference (EUSIPCO), pp 649–653
22. Luo W, Li X, Li W, Hu W (2011) Robust visual tracking via transfer learning. In: ICIP, pp 485–488
23. Eaton E, Des Jardins M (2009) Set-based boosting for instance-level transfer. In: Proceedings of the 2009 IEEE international conference on data mining workshops, pp 422 –428
24. Venkatesan A, Krishnan N, Panchanathan S (2010) Cost–sensitive boosting for concept drift. In: Proceedings of the 2010 international workshop on handling concept drift in adaptive information systems, pp 41–47

25. Sun Y, Kamel MS, Wong AK, Wang Y (2007) Cost-sensitive boosting for classification of imbalanced data. Pattern Recognit 40(12):3358–3378
26. Pardoe D, Stone P (2010) Boosting for regression transfer. In: Proceedings of the 27th international conference on machine learning, pp 863–870
27. Eaton E (2009) Selective knowledge transfer for machine learning. Ph.D. thesis. University of Maryland Baltimore County
28. Al-Stouhi S, Reddy CK (2011) Adaptive boosting for transfer learning using dynamic updates. In: Machine learning and knowledge discovery in databases. Springer, Berlin, pp 60–75
29. Provost F (2000) Machine learning from imbalanced data sets 101. In: Proceedings of the American association for artificial intelligence workshop, pp 1–3
30. Ertekin S, Huang J, Bottou L, Giles L (2007) Learning on the border: active learning in imbalanced data classification. In: Proceedings of the sixteenth ACM conference on conference on information and knowledge management, pp 127–136
31. Kubat M, Holte RC, Matwin S (1998) Machine learning for the detection of oil spills in satellite radar images. Mach Learn 30(2–3):195–215
32. Guyon I, Aliferis CF, Cooper GF, Elisseeff A, Pellet JP, Spirtes P et al (2008) Design and analysis of the causation and prediction challenge. J Mach Learn Res Proc Track 3:1–33
33. Rijsbergen CJV (1979) Information retrieval, 2nd edn. Butterworth-Heinemann, Newton, ISBN:0408709294
34. Brodersen K, Ong CS, Stephan K, Buhmann J (2010) The balanced accuracy and its posterior distribution. In: Pattern recognition (ICPR), 2010 20th international conference on, pp 3121–3124
35. Vinzamuri B, Reddy CK (2013) Cox regression with correlation based regularization for electronic health records. In: Data mining (ICDM), 2013 IEEE 13th international conference on. IEEE, pp 757–766
36. Clancy C, Munier W, Crosson K, Moy E, Ho K, Freeman W et al (2011) 2010 National healthcare quality and disparities reports. Tech. Rep, Agency for Healthcare Research and Quality (AHRQ)
37. Gertler P, Molyneaux J (1994) How economic development and family planning programs combined to reduce indonesian fertility. Demography 31(1):33–63. doi:10.2307/2061907
38. Little MA, McSharry PE, Hunter EJ, Spielman J, Ramig LO (2009) Suitability of dysphonia measurements for telemonitoring of parkinson's disease. IEEE Trans Biomed Eng 56(4):1015–1022
39. Fahn S, Elton R, Committee UD et al (1987) Unified parkinson's disease rating scale. Recent Dev Parkinson's Dis 2:153–163
40. Dai W, Xue GR, Yang Q, Yu Y (2007b) Co-clustering based classification for out-of-domain documents. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 210–219
41. Aizawa A (2003) An information-theoretic perspective of tf–idf measures. Inf Process Manag 39(1):45–65
42. Chawla N, Bowyer K, Hall L, Kegelmeyer W (2002) Smote: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
43. Kohavi R, et al (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In: International joint conference on artificial intelligence; vol. 14. Lawrence Erlbaum Associates Ltd, pp 1137–1145
44. Yang Y (1999) An evaluation of statistical approaches to text categorization. Inf Retr 1(1):69–90
45. Chawla NV, Lazarevic A, Hall LO, Bowyer KW (2003) Smoteboost: improving prediction of the minority class in boosting. In: Proceedings of the principles of knowledge discovery in databases, PKDD-2003, pp 107–119
46. Batista G, Prati R, Monard M (2004) A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explor Newsl 6(1):20–29
47. Wang Y, Xiao J (2011) Transfer ensemble model for customer churn prediction with imbalanced class distribution. In: Information technology, computer engineering and management sciences (ICM), 2011 international conference on. vol. 3. IEEE, pp 177–181
48. Palit I, Reddy CK (2012) Scalable and parallel boosting with mapreduce. IEEE Trans Knowl Data Eng 24(10):1904–1916
49. Reddy CK, Park JH (2011) Multi-resolution boosting for classification and regression problems. Knowl Inf Syst 29(2):435–456

Samir Al-Stouhi is a research engineering member of the Automobile Technology Research (ATR) group at Honda. He received his Ph.D. from the Electrical and Computer Engineering Department at Wayne State University. His primary research interests are in the area of machine learning with applications to localization via sensor fusion, transfer learning, multi-task learning and imbalanced learning.

Chandan K. Reddy is an Associate Professor in the Department of Computer Science at Wayne State University. He received his Ph.D. from Cornell University and MS from Michigan State University. His primary research interests are in the areas of data mining and machine learning with applications to healthcare, bioinformatics, and social network analysis. His research is funded by the NSF, NIH, DOT, Susan G. Komen for the Cure Foundation. He has published over 50 peer-reviewed articles in leading conferences and journals. He received the Best Application Paper Award at the ACM SIGKDD conference in 2010 and was a finalist of the INFORMS Franz Edelman Award Competition in 2011. He is a senior member of the IEEE and a life member of the ACM.