Transfer Learning for Education Data[*]

Xin J. Hunt
SAS Institute Inc.
100 SAS Campus Dr
Cary, North Carolina 27513
xin.hunt@sas.com

Ilknur Kaynar Kabul
SAS Institute Inc.
100 SAS Campus Dr
Cary, North Carolina 27513
ilknur.kaynarkabul@sas.com

Jorge Silva
SAS Institute Inc.
100 SAS Campus Dr
Cary, North Carolina 27513
jorge.silva@sas.com

ABSTRACT

Predicting the future performance and graduation rate of students based on their academic records is of crucial importance. Such capabilities allow accurate estimates of on-time graduation and, when necessary, enable effective student interventions or adjustments to the degree programs. Predicting performance in an entire degree program poses many challenges: similarly-named courses can be very different across degree programs; students differ in their background; they may select different combinations of courses; and, critically, the number of students in some degree programs is very small, which hinders predictive modeling. In this paper, we propose an approach for predicting graduation rates in degree programs by leveraging data across multiple degree programs to address these challenges. The proposed method is based on the burgeoning machine learning discipline of *transfer learning*. Transfer learning pools information from multiple degree programs, thereby increasing the effective sample size. At the same time, transfer learning takes into account the differences across different degree programs and automatically down-weighs less-relevant data. We demonstrate our approach using anonymized real data from North Carolina State University, where the proposed method achieves highly promising results.

KEYWORDS

Transfer learning; Gradient boosting; Education; Degree program graduation prediction

ACM Reference format:
Xin J. Hunt, Ilknur Kaynar Kabul, and Jorge Silva. 2017. Transfer Learning for Education Data. In *Proceedings of ACM SIGKDD Conference, El Halifax, Nova Scotia Canada, August 2017 (KDD'17),* 6 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Enrollment to a degree program is a substantial investment for students and their families. The number of students that stay throughout the duration of a degree program and successfully leave with a degree is one of the major indicators of the program's success. Such indicators are considered by prospective students and their parents when they select colleges to attend, since they help to establish whether the college is a good fit, and whether the monetary investment is likely to succeed.

While there is substantial literature pertaining to the problem of predicting graduation within a course of study, there has been less research on predicting graduation for an entire degree program. This is due to multiple challenges, such as: variation in the content of similarly-named courses across different degrees; heterogeneity in student backgrounds; the ability of students to pick multiple combinations of courses within a degree; and the fact that some degree programs enroll very few students, which is of particular concern because it leads to small sample sizes for training predictive models.

Graduation rates reported to the federal government consist of the percentage of students who remain at the same college and finish a degree in six years. This information may provide many insights about the degree program and the university. If graduation rates are low, it may mean that students do not get the academic support that they need to succeed (e.g., support from faculty and staff, quality of the courses offered, cost of living). While the underlying reasons may be complex, low graduation rates can at least provide advance warning that something may be wrong and may need to be investigated. On the other hand, high graduation rates are often associated with: a positive environment, in which graduation is highly valued and encouraged by other students and faculty members; high standards for admission, meaning that high-performing students are selected and therefore more likely to graduate; and/or an academic program which supports the needs of the students. Graduation rates are also important for the federal officials and college leaders, in order to assess budgetary needs and potential improvements to the degree programs in the upcoming years. In short, graduation rates are an essential tool for any institution [7].

Predicting the performance of each student is also very important in order to put together an intervention plan for the student. The success of an intervention program depends on the college's ability to accurately predict the need for it as early in the program as possible. One of the main challenges in predicting successful graduation is the lack of information for each student in each department. We can treat all of the students from different departments in a similar fashion, but this may not always work due to the differences in the courses in each department. Thus, we are forced to choose between using training data from each single degree program in isolation, which may be too small, or gathering a larger dataset which includes multiple degree programs but does not take into account the significant heterogeneity between those programs.

[*] Produces the permission block, and copyright information

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1.1 Proposed approach

Transfer learning allows us to reach a beneficial balance, by using all available data from multiple degree programs (or different academic status) but automatically assigning appropriate weights to each observation. Hence, data which come from a less-relevant distribution (different degree program, different students, etc.) are not discarded but will have lower weight in the training process. In the remainder of this paper, we provide a short review of relevant background and prior work, followed by a brief explanation of transfer learning in general, and the TrAdaBoost algorithm in particular, which we employ in our analysis. We then describe the data and experimental setup, and present results obtained with real data from the North Carolina State University (NCSU). Final thoughts and future research directions are then proposed in the conclusion.

2 PRIOR WORK

In [14] a method is proposed to solve some of the challenges in predicting student success, namely tackling the issues of different student backgrounds, course content, courses taken and the students' evolving progress during the program. In the proposed approach, they first use a bi-layered structure comprising multiple base predictors, and a cascade of ensemble predictors. They also determine course relevance using a data-driven approach based on latent factor models and probabilistic matrix factorization. The algorithm is used on data collected from the Mechanical and Aerospace Engineering department at the University of California in Los Angeles (UCLA) for students graduating during three years (2013, 2014, 2015).

A framework to identify students who are at risk of not graduating high school on time is presented in [10]. This framework includes a feature extraction process, different classifiers and evaluation criteria. They use modeling algorithms such as random forest and support vector machines (SVM). This framework is applied within two school districts with a combined enrollment of around 200,000 students.

In [6], results are presented for predicting the graduation rates based on the characteristics of incoming first-year students (*e.g.,* high school grade point average (HSGPA), standardized assessment test (SAT) scores). In their initial experiments, they use different regression models for each degree completion measure. They also investigate the inclusion of information from freshman surveys and the corresponding impact on the results. Adding such information substantially improves the prediction accuracy. Additionally, they conclude that early admittance decision, overall cost of attending, and the size of the college are the three factors that have the largest positive impact on degree completion. Overall, the difference in degree completion rates between institutions is attributable to variation in the characteristics and profiles of the incoming students.

It should be noted that the aforementioned approaches use multiple models to address the fact that the data is drawn from different populations with significantly different distributions. In contrast, our approach utilizes a single model which explicitly accounts for population heterogeneity with transfer learning, thereby significantly simplifying the modeling task.

Transfer learning on education data has been discussed in [3]. In [3], the authors predict student dropouts in online courses by first training ensemble models from the data of four online courses, and then transfer the model to a new course (new task). However, the setup is different from ours. In [3], the new target course is "unseen", *i.e.,* no data is available from the new task, which means the predictive model is transferred without being modified or tailored to the new task. As a result, the ultimate goal of [3] is to build a universal predictive model for different online courses from available data. In this work, we assume a small amount of labeled data from the new task is available. The data from the new task is then used to guide the transferring process. Our approach yields a model specifically tailored to the new task, as opposed to a universal-fitting model.

3 TRANSFER LEARNING

In traditional machine learning applications, the assumption is that the training data and testing data are taken from the same domain (e.g. the input feature space and data distribution are same). This is not always the case in the real world. Sometimes collecting training data can be expensive or difficult due to different constraints. In such cases, we may want to take advantage of other data sets in related domains that are already available. Transfer learning algorithms enable us to transfer the knowledge from a related (source) task that has already been learned, to a new (target) task. This transfer can take place in multiple ways, such as the reuse of some or all of the training data sets, or features extracted from those data sets. The transfer can also consist of reusing some model-specific settings (e.g., neural-node layering).

Transfer learning has crucial importance when we have insufficient amounts of labeled training data of high quality in the new application domain. It is also essential in applications when the machine learning models become outdated due to changes in the underlying data.

Transfer learning can be most successful when the target and source domains have some similarities. For instance, a natural language algorithm that is used to classify English documents in a particular discipline should be adaptable to classify Spanish documents in a related field.

See [12] for a survey about transfer learning for classification, regression and clustering problems. This survey also describes the relationship between transfer learning and other related machine learning techniques such as domain adaptation, multitask learning and sample selection bias. Recently, there has been considerable interest on transfer learning for deep learning applications that involve image data. [11] presents a method for transferring features extracted by a trained convolutional neural network to another task. This eliminates the need for retraining a large neural network for feature learning.

In this paper, we use the TrAdaBoost algorithm which is described below.

3.1 Boosting for transfer learning

Boosting algorithms have long been used in machine learning for converting weak learners to strong learners [4, 15]. AdaBoost, or "Adaptive Boosting" [8], one of the most popular and widely used boosting algorithms, has been successfully applied to various areas

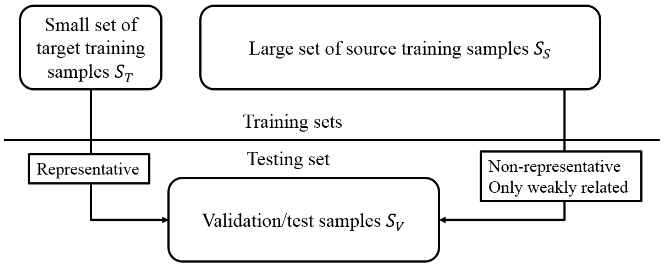

Small set of target training samples S_T
Large set of source training samples S_S
Training sets
Testing set
Representative
Non-representative Only weakly related
Validation/test samples S_V

Figure 1: Relationship between target training set $\mathcal{S}_T$, source training set $\mathcal{S}_S$, and validation set $\mathcal{S}_V$.

like multimedia [9], computer vision [2, 13] and financial forecasting [1]. The basic idea behind AdaBoost is to iteratively evaluate how “difficult” it is to classify each sample in the training data, and increase weights for the difficult-to-classify samples (*i.e.,* samples that got classified incorrectly by the learners) to produce learners that better fits the data. AdaBoost can be used with a variety of weak learners, though decision tree is often a good choice [15].

TrAdaBoost [5] was developed in 2007 by Dai et al. to extend AdaBoost for transfer learning. Instead of assuming that all training data come from the same distribution as the target task, TrAdaBoost assumes that we have two sets of training data: target training set $\mathcal{S}_T$, and source training set $\mathcal{S}_S$. The target set $\mathcal{S}_T = \{x_1, \cdots, x_m\} \subset \mathcal{X}$ is the high quality data that comes from the same distribution as the validation/testing set $\mathcal{S}_V$ which we want to train a good model for. The source set $\mathcal{S}_S = \{x_{m+1}, \cdots, x_{m+n}\} \subset \mathcal{X}$ are of less quality, where the data can be outdated or just weakly-related to the task, *i.e.,* the samples in $\mathcal{S}_S$ come from a different distribution from $\mathcal{S}_V$. Typically $n \gg m$. For classical machine learning algorithms, $\mathcal{S}_S$ would not be of much use since the underlying distribution can be very different from the $\mathcal{S}_V$ and cause strong bias in the model. However, TrAdaBoost improves performance by transferring these weakly-related or outdated data to the task of interest. Like AdaBoost, TrAdaBoost weighs samples from both $\mathcal{S}_T$ and $\mathcal{S}_S$ to train a new weak learner $h_t(x)$ at each iteration t. However, TrAdaBoost treats $\mathcal{S}_T$ and $\mathcal{S}_S$ differently:

- For $x_i \in \mathcal{S}_T$, increase weights for incorrectly classified samples;
- For $x_i \in \mathcal{S}_S$, reduce weights for incorrectly classified samples.

The strategy of TrAdaBoost can be seen as using $\mathcal{S}_T$ as a guide to filter out data from $\mathcal{S}_S$ that are most relevant to the target task. By doing so, TrAdaBoost reduces variance by including more data from $\mathcal{S}_S$, yet avoiding overwhelming bias by strategically weighing down less-related data from $\mathcal{S}_S$. The relationship between $\mathcal{S}_T, \mathcal{S}_S$, and $\mathcal{S}_V$ is illustrated in Figure 1. For completeness, the TrAdaBoost algorithm is summarized in Algorithm 1[1].

Algorithm 1 TrAdaBoost

```
Input:
    $\{(x_i, y_i)\}_{i=1}^{m+n}, x_i \in \mathcal{X}, y_i \in \{0, 1\}$
  where $x_1, \cdots, x_m \in \mathcal{S}_T$, and $x_{m+1}, \cdots, x_{m+n} \in \mathcal{S}_S$
Initialize weights
    $w_1(i) = \frac{1}{m+n},\ i = 1, \ldots, m+n$
    $\widetilde{w}_1(i) = \frac{1}{m},\ i = 1, \ldots, m$
for $t = 1, \ldots, T$ do
    Learn weak learner $h_t : \mathcal{X} \to \{0, 1\}$ s.t.
      $h_t = \arg\min_{h \in \mathcal{H}} \sum_{i=1}^{m+n} w_t(i) \mathbb{1}_{y_i \neq h_t(x_i)}$
    Compute
      $r_t = \sum_{i=1}^{m} \widetilde{w}_t(i) |h_t(x_i) - y_i|$ (needs to be $< 0.5$)
      $\beta_t = \frac{r_t}{1-r_t}$
      $\beta_0 = \frac{1}{(1+\sqrt{2\log(n/T)})}$
    Update
      $v_t(i) = \begin{cases} w_t(i)\beta_t^{-|y_i - h_t(x_i)|}, & 1 \leq i \leq m \\ w_t(i)\beta_0^{|y_i - h_t(x_i)|}, & m+1 \leq i \leq m+n \end{cases}$
      $w_{t+1}(i) = \frac{v_t(i)}{\sum_{j=1}^{m+n} v_t(j)}$
      $\widetilde{w}_{t+1}(i) = \frac{v_t(i)}{\sum_{j=1}^{m} v_t(j)}$
end for
Output:
  Strong learner    $H(x) = \mathbb{1}_{\left\{\prod_{t=1}^{T} \beta_t^{1/2 - h_t(x)} \geq 1\right\}}$
```

4 EXPERIMENTAL RESULTS WITH THE NCSU STUDENT DATASET

In this section, we show experimental results on the NCSU student dataset with TrAdaBoost in Algorithm 1 and AdaBoost to illustrate how transfer learning can be used to improve the graduation prediction. We also discuss a few scenarios when transfer learning techniques are not as effective as classical non-transfer methods.

4.1 The NCSU student dataset

The NC State University (NCSU) student dataset collects academic and demographic information (a total of 211 different variables) for undergraduate students during one academic year (with follow-up information of whether the student graduated in six years or not). The dataset has been anonymized, and no personal-identifiable information is included.

In this section we use the student outcome data from a total number of 7637 students. We select 152 variable related to the students’ general academic information, including students’ current academic standing, high-school performance, SAT and/or ACT scores, courses (and corresponding scores) taken during college, and other basic academic enrollment information. We exclude most demographic information like race and gender, but keep in-state/out-of-state tuition status. A few variables are also removed due to significant percentage of missing values. The goal is to predict whether a student will graduate within six years after enrollment based on the academic information.

[1] The original TrAdaBoost combines only the second half of the trained weak learners. In our experiments, we combine all trained weak learners as this provides better empirical results.

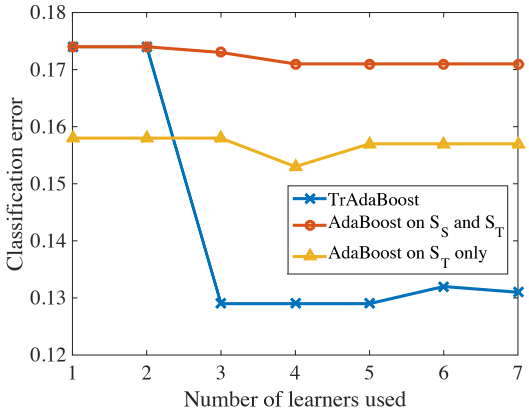


Figure 2: Comparison between TrAdaBoost and AdaBoost on three engineering departments from the NCSU student dataset.

4.2 Prediction for students from three engineering departments

In this experiment, we learn discipline-specific predictive models for students. Specifically, we take students in three engineering departments: biomedical engineering, computer engineering, and electrical engineering (a total of 465 students), as our target population. All students from other departments are treated as the source training set ($\mathcal{S}_S$, total of 6834 students). The data of students from the three departments is randomly divided into the target training set ($\mathcal{S}_T$, 140 students, 30% of the target population) and validation set ($\mathcal{S}_V$, 325 students, 70% of the target population). Because of the small number of students per department (from ten to about a hundred and fifty students per department), we combine three departments where students take many similar classes in math, physics, and engineering.

Figure 2 shows the result of TrAdaBoost and AdaBoost on the students from three engineering departments. For all the algorithms, we use binary decision trees as weak learners. For AdaBoost, we train two models with different training sets: (a) $\mathcal{S}_S \bigcup \mathcal{S}_T$, and (b) $\mathcal{S}_T$ only. As seen in Figure 2, TrAdaBoost has the smallest error among the three models trained (when the number of weak learners used is larger than two). Between the two AdaBoost-trained models, AdaBoost trained on $\mathcal{S}_T \bigcup \mathcal{S}_S$ yields higher prediction error than AdaBoost trained on $\mathcal{S}_T$ alone, despite having a larger number of training samples. The performance difference between the two AdaBoost-trained models is a direct result of the inherent heterogeneity between disciplines. Although the source training set $\mathcal{S}_S$ has a larger number of students, the underlying distribution is different from the target student population in $\mathcal{S}_V$. Thus using $\mathcal{S}_S$ without selection increases, not decreases prediction error. TrAdaBoost, on the other hand, uses $\mathcal{S}_S$ selectively by strategically weighing down less-related samples, providing better performance than both AdaBoost-trained models.

Qualitatively speaking, the most predictive variables in the ensemble models are

- End-of-semester academic standing
  Students with good academic standing are more likely to graduate than students suspended or on academic warning.
- Grade point average (GPA)
  Students with higher GPA are more likely to graduate in six years.
- Highest level of courses completed in physics and computer science
  For students that are not in good academic standing, completing higher level of courses in physics and computer science indicates better chance of graduating in six years.

4.3 Prediction for suspended students and students on academic warnings

In the second experiment, we learn predictive models based on the academic standing of students. Specifically, models are trained for students who are suspended or on academic warning. In general, student that are suspended or on academic warning have higher drop-out rate than students in good academic standing, and academic standing is, as seen in the first experiment, often a good predictor for graduation. However, if we want to focus on the students who are suspended or on academic warning, and understand what specifically impacts their likelihood to graduate, the academic standing itself is no longer useful.

The distribution of data from students that are suspended or on academic warning can be very different from students with good academic standing, where the suspended or on-warning students tend to have lower GPA scores and take less advanced classes. As a result (and we will see further in the experiments), the predictive models learned from the entire student population do not work very well for those suspended or on academic warning. In this experiment, we use the data from students that are suspended or on academic warning as our target population (total of 550 students), and use the data from students with good academic standing as our source training set ($\mathcal{S}_S$, total of 6749 students). The target set is randomly split in half, where one half is used as the target training set ($\mathcal{S}_T$, total of 275 students), and the other ($\mathcal{S}_V$, the other 275 students) is used as the validation set.

Figure 3 shows the result of TrAdaBoost and AdaBoost on the students that are suspended or on academic warning. Similar to the first experiment, we use binary decision trees as weak learners. For AdaBoost, we trained two models with different training sets: (a) $\mathcal{S}_S \bigcup \mathcal{S}_T$, and (b) $\mathcal{S}_T$ only. As seen in Figure 3, TrAdaBoost has the smallest error among the three models trained (when the number of weak learners used is larger than three). Furthermore, the error of the AdaBoost model trained on $\mathcal{S}_T$ fluctuates with the number of learners. This is because the number of students in $\mathcal{S}_T$ alone is small, and over-fitting can lead to unstable predictive models. The number of students in $\mathcal{S}_S \bigcup \mathcal{S}_T$ is large enough to estimate a stable model, but because the underlying distribution of the majority of students differs significantly from the students suspended or on academic warning, not differentiating the two populations results in the model biasing towards the general population and causes lower performance, compared to models focusing on the suspended or on-warning students.

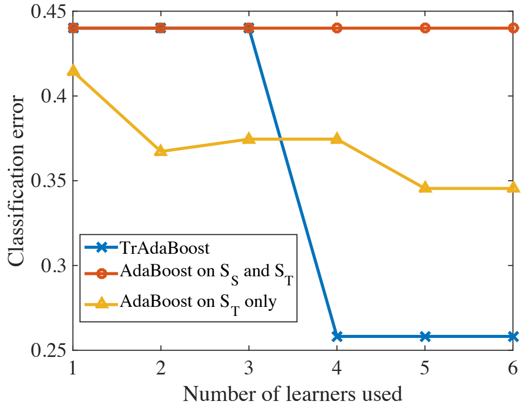


Figure 3: Comparison between TrAdaBoost and AdaBoost on suspended and on-academic-warning students from the NCSU student dataset.

Qualitatively speaking, the most predictive variables in the ensemble model for suspended students or students on academic warnings are

- Grade point average (GPA)
  Students with higher total GPA or higher GPA from the current semester are more likely to graduate in six years.
- Course load
  Students that are enrolled full-time are more likely to graduate than students enrolled three-quarter time, half-time or less than half-time.
- Number of courses withdrawn, number of "F" grades achieved, and high school GPA
  These are much less predictive than the student's GPA. However, in general, students with less "F" grades and withdrawn courses are more likely to graduate than students with more "F" grades and withdrawn courses. Interestingly, for students suspended or on academic warning, high school GPA is actually negatively correlated with the chance of graduation. In fact, the suspended or on-warning students are more likely to graduate if their high school GPA is lower than 4/5.

4.4 When TrAdaBoost does not improve performance

In the previous experiments, we have seen that when the target set has a different underlying distribution from the source set, TrAdaBoost helps improve the accuracy of predictive models by using the target set as a guide to select related data from the source set. However, there are also cases when such transfer learning techniques do not improve the performance of classical models.

- When the target and source distributions are identical or independent
  When the target and source samples come from the same distribution, TrAdaBoost's selective process reduces the amount of useful training samples (which increases error variance), and bias towards the (usually significantly) smaller target training set. In this case, it is more effective to train a classical model on the combined training set $\mathcal{S}_T \bigcup \mathcal{S}_S$ without transfer learning.
  On the other hand, when the target and source samples come from independent distributions, the use of the source set simply does not provide additional information. In this case, training directly over the target set is more efficient and accurate.
- When the target training set too small to guide the selection in the source set
  TrAdaBoost relies on the target set as a distribution guide to select good samples from the source set. If the target set is too small to be representative, the selection process will be severely biased by the target samples and cause over-fitting to the target set even when the source training set is large. In this case, even though the source set brings bias to the model, training over the combined training set $\mathcal{S}_T \bigcup \mathcal{S}_S$ is usually more robust and reliable than fitting a model based on an extremely small number of target samples.

Table 1 shows two examples of when TrAdaBoost is less effective than the classical AdaBoost algorithm. Specifically, Table 1a shows the results when the target and source samples all come from the same distribution. In this experiment, we randomly partition the entire student population into $\mathcal{S}_T$, $\mathcal{S}_S$, and $\mathcal{S}_V$ (with equal proportions). In this case, data from all three sets come from a single distribution, and AdaBoost trained on the combined training set achieves the lowest classification error. In fact, TrAdaBoost performs the worst in this case, since the selection process artificially increases the bias by favoring $\mathcal{S}_T$ over $\mathcal{S}_S$.

Table 1b shows the results when the target training set $\mathcal{S}_T$ is too small. In this experiment, we set the target population as the students from the paper science department, where we only have 24 samples in the NCSU student outcome dataset. The extremely small number of samples results in lower accuracy with TrAdaBoost and AdaBoost trained on $\mathcal{S}_T$ only due to overfitting. The model trained on the larger general population by AdaBoost has the best predictive performance.

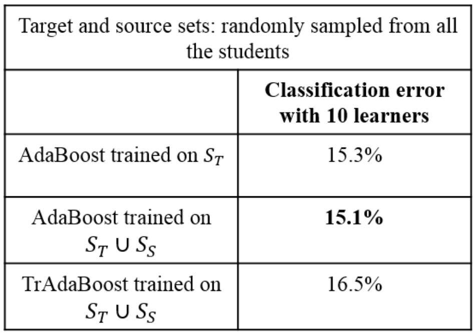
5 CONCLUSION

Graduation rates are an important measure of degree program success, which is highly informative and helps in the early detection and correction of problems with students and academic institutions. Predicting graduation rates is a very challenging problem due to the variation in student backgrounds, degree content, schools and departments, as well as enrollment numbers. By using transfer learning we are able to mitigate many of these issues and improve the predictive accuracy of our models, as demonstrated using real data.

The TrAdaBoost algorithm, a transfer learning variant of gradient boosting, allows us to decrease the classification error in graduation rate prediction using data from multiple engineering departments in the NCSU. This is the case for both the general student population and for the subset of students given suspensions or academic warnings. On the other hand, transfer learning

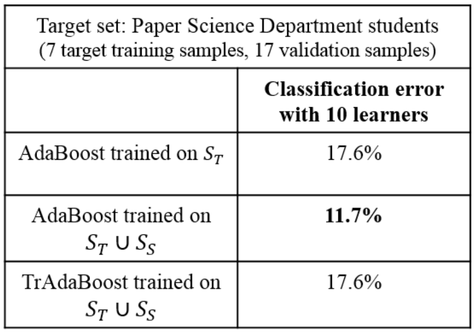
Table 1: Examples of when TrAdaBoost does not improve the prediction accuracy

(a) Identical source and target distribution



Target and source sets: randomly sampled from all the students	
	Classification error with 10 learners
AdaBoost trained on S_T	15.3%
AdaBoost trained on $S_T \cup S_S$	**15.1%**
TrAdaBoost trained on $S_T \cup S_S$	16.5%

(b) Target set too small to be representative



Target set: Paper Science Department students (7 target training samples, 17 validation samples)	
	Classification error with 10 learners
AdaBoost trained on S_T	17.6%
AdaBoost trained on $S_T \cup S_S$	**11.7%**
TrAdaBoost trained on $S_T \cup S_S$	17.6%

is not advantageous when either the source and target distributions are identical, or when the target sample size is too small to be representative.

Further research is necessary, involving more departments and schools, so that the ability to transfer knowledge from one state to another, or from one department to another, can be assessed. Using all this additional data can yield more accurate predictions and better academic outcomes. Future research should also investigate the performance of different transfer and multi-task learning algorithms besides TrAdaBoost. Ultimately, we wish to also focus on finding the features within the data that are most informative regarding graduation success for all types of students, in multiple department and schools at the national level.

REFERENCES

[1] Esteban Alfaro, Noelia García, Matías Gámez, and David Elizondo. 2008. Bankruptcy forecasting: An empirical comparison of AdaBoost and neural networks. *Decision Support Systems* 45, 1 (2008), 110–122.

[2] Marian Stewart Bartlett, Gwen Littlewort, Ian Fasel, and Javier R Movellan. 2003. Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction.. In *Computer Vision and Pattern Recognition Workshop, 2003. CVPRW'03. Conference on*, Vol. 5. IEEE, 53–53.

[3] Sebastien Boyer and Kalyan Veeramachaneni. 2015. Transfer Learning for Predictive Models in Massive Open Online Courses.. In *AIED*. 54–63.

[4] Leo Breiman et al. 1998. Arcing classifier (with discussion and a rejoinder by the author). *The annals of statistics* 26, 3 (1998), 801–849.

[5] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. 2007. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*. ACM, 193–200.

[6] Linda DeAngelo, Ray Franke, Sylvia Hurtado, John H Pryor, and Serge Tran. 2011. Completing college: Assessing graduation rates at four-year institutions. (2011).

[7] Ray Franke. 2012. Towards the Education Nation: Revisiting the impact of financial aid, college experience, and institutional context on baccalaureate degree attainment using a propensity score matching, multilevel modeling approach. (2012).

[8] Yoav Freund and Robert E Schapire. 1995. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*. Springer, 23–37.

[9] Wei Hao and Jiebo Luo. 2006. Generalized multiclass adaboost and its applications to multimedia classification. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*. IEEE, 113–113.

[10] Himabindu Lakkaraju, Everaldo Aguiar, Carl Shan, David Miller, Nasir Bhanpuri, Rayid Ghani, and Kecia L Addison. 2015. A Machine Learning Framework to Identify Students at Risk of Adverse Academic Outcomes. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*.

[11] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. 2014. Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 1717–1724. https://doi.org/10.1109/CVPR.2014.222

[12] S. J. Pan and Q. Yang. 2010. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10 (Oct 2010), 1345–1359. https://doi.org/10.1109/TKDE.2009.191

[13] Yubo Wang, Haizhou Ai, Bo Wu, and Chang Huang. 2004. Real time facial expression recognition with adaboost. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, Vol. 3. IEEE, 926–929.

[14] J. Xu, K. H. Moon, and M. van der Schaar. 2017. A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs. *IEEE Journal of Selected Topics in Signal Processing* PP, 99 (2017), 1–1. https://doi.org/10.1109/JSTSP.2017.2692560

[15] Zhi-Hua Zhou. 2012. *Ensemble methods: foundations and algorithms*. CRC press.