Transfer Learning in Dynamic Business Environments: An Application in Earnings Forecast for Public Firms

Jiaxu Peng
School of Computing, National University of Singapore, j.peng@u.nus.edu

Learning in Dynamic Business Environments: An Application in Earnings Forecast for Public Firms

Short Paper

Jiaxu Peng
National University of Singapore
13 Computing Drive
Singapore 117417
j.peng@u.nus.edu

Abstract

In dynamic business environments, the underlying true data pattern changes rapidly. Machine learning models built upon historical data may not be responsive to the changes. A simple solution is to re-train a machine learning model using the re-collected current data. However, current data are often scarce. Therefore, it would be optimal to adapt the machine learning model built on historical data to the current period. In this study, we propose a two-step transfer learning method for enhancing machine learning in dynamic data environments. Our insight is that, by comparing current data and historical data, we gain information on the change of data environments, which guides the training of machine learning using historical and current data sets simultaneously. In this research-in-progress, we evaluate our method and an existing state-of-art algorithm in the earnings prediction tasks. Preliminary results show the effectiveness of transfer learning in dynamic business environments.

Keywords: Business analytics, transfer learning, dynamic data environments, machine learning, earnings forecast

Introduction

Machine learning has been successfully applied in a variety of business applications such as in predicting defaults in consumer credit loans (Khandani et al. 2010) and bankruptcy of public firms (Barboza et al. 2017). However, applications of machine learning into the business world create not only opportunities but also challenges. One of the challenges arises from dynamically changing data environments (Yang and Wu 2006; Saboo et al. 2016; Grover et al. 2018). In practice, empirical evidence shows that forecasting changes is difficult even for experts. For instance, on the brink of the economic recession of 2008, economists projected, on average, that the economy will grow 2.1% from the fourth quarter of 2007 to the end of 2008 (published in Business Week, December 20, 2007). Obviously, they did not successfully forecast the many credit defaults and bankruptcies that happened in the next few months (Makridakis et al. 2009). With traditional supervised learning approaches, there is no standard method to update existing forecasting models to predict any financial variables for 2009 during the early financial crisis in 2008. Under such a significant change in the external environment, supervised classification approaches to predict credit defaults or bankruptcies may result in a considerable loss of accuracy because similar training cases rarely occurred in the past.

Supervised machine learning methods use historical data as a training set to construct a prediction model, and then applies the built model to current test data to make predictions of future events or of variables of interest. One important assumption is that the historical training data and current test data exhibit the same underlying pattern (Pan and Yang 2010). In dynamic data environments, this assumption may not hold. For instance, in predicting firms' future earnings during recession periods, both the distribution of predictors and the function between predictors and dependent variable may change, which is referred to as dynamically changing data pattern in the literature and also in this study (Li and Mohanram 2014).

This study aims at improving forecasting in *dynamic data environments* where the historical data and the current data may exhibit different patterns. More specifically, we approach this problem from a *transfer learning perspective* and explore *how machine learning models built upon historical data can be transferred when the data environment undergoes significant changes*. In this study, transfer learning is defined as extracting knowledge from a *source* data set and applying this knowledge to a *target* task (Pan and Yang 2010).[1] Within this framework, researchers can observe the predictor variables (denoted with $\boldsymbol{x}$) in both the source and target data, while the variable to be predicted (denoted with y) is only observed in the source data and is to be estimated for the target data.

The related transfer learning literature has mainly focused on the changes in the distribution of $\boldsymbol{x}$ across the source and target data (Sugiyama et al. 2008). However, in real-world applications, the distributional change of y may not be fully explained by the change in probability distribution of $\boldsymbol{x}$. In other words, the conditional probability distribution of y conditional on $\boldsymbol{x}$ (or almost equivalently, $y = f(\boldsymbol{x})$) may become different in a new macro-economic environment. For instance, in the earnings prediction task, our domain knowledge can intuitively anticipate that firms' earnings would systematically decrease during the 2018-19 USA-China trade war. However, existing machine learning methods cannot embed our intuitions into quantitative predictions, neither do they explicitly allow the underlying pattern between $\boldsymbol{x}$ and y to change. Without modeling the joint distributional changes of $\boldsymbol{x}$ and y completely, the generalizability of the prediction model built upon the source data may be jeopardized.

A key technical challenge is that if y is arbitrarily changing, we have no clue of its distribution in the target data. In this case, the analyst must rely on some "good" and "current" source data which exhibit the same data pattern as the target data to induce a better estimation of the relationship between $\boldsymbol{x}$ and y. This is essentially the idea of *inductive transfer learning* (Pan and Yang 2010). To exploit "good" source data, the closest method from the literature is TrAdaBoost, which is proposed by Dai et al. (2007). TrAdaBoost is built on the Adaboost algorithm. The innovative aspect of TrAdaBoost is that it assigns greater weights to "good" source data and decreases the weights of the remaining "bad" source data. However, since TrAdaBoost employs the adjusted weighting formula from the original Adaboost algorithm, we still face the obstacle of generalizing inductive learning to other better-performing modern machine learning algorithms, such as XGBoost for ensemble of decision trees and stochastic gradient descent for training deep neural network. Moreover, Dai et al. (2007) point out that TrAdaBoost does not guarantee to always outperform AdaBoost since the quality of the "bad" source data records is unknown.

In this paper, we conduct preliminary experimentation of the earnings prediction task based on two methods: the two-stage TrAdaBoost.R2 (Dai et al. 2007; Pardoe and Stone 2010) and a preliminary version of our newly proposed method. For earnings prediction application, our purpose is to adapt the machine learning model trained with data from historical periods (source data) for making predictions in a new time period (target data). Both methods rely on changing the weights of the source data to reflect changes in the underlying data pattern. Particularly, our method, motivated by empirical risk minimization (ERM, (Vapnik 1995)), depicts the changes of the data environments through a probability model which in turn generates the weights of the source data. Results show that both methods improve prediction performance. Moreover, our method demonstrates outstanding performance in predicting earnings during the great recession period when the data environment underwent significant changes.

This study contributes to the dynamic learning challenge in two important aspects. First, we evaluate the effectiveness of inductive learning strategy from the literature. Without any inductive clues, forecasting the future based on the historical data is difficult when data environment undergoes significant changes. Inductive learning sheds light on this problem by highlighting the role of "good" source data in inducing valuable historical data. Second, we propose and evaluate a method that explicitly generates the weights

[1] Similar to typical transfer learning studies, we study the case of one source task and focus on one target task. A complete description of the goals of transfer learning is provided by Broad Agency Announcement 05-29 of DARPA's Information Processing Technology Office.

of source data records, thus making it applicable to a wide range of machine learning algorithms, which can potentially improve prediction performance in many practical applications.

Related Literature

In this section, we review studies on dynamic data environments from the data mining literature and the information systems literature, respectively. We then present how a specific type of transfer learning, inductive transfer learning, can be applied to the earnings prediction context.

A Brief Review of Transfer Learning

In a supervised machine learning setting, the task is to learn the relationship between $\boldsymbol{x}$, the predictors, and y, the variable to be predicted. In a probabilistic view, it is the conditional probability as follows,

$$f(y|\boldsymbol{x};\boldsymbol{\omega}) \tag{1}$$

where $\boldsymbol{\omega}$ is a vector of the unknown parameters in the conditional probability density function. In this brief overview, we simplify the typology of transfer learning defined by Pan and Yang (2010).

The two broad types of transfer learning are *inductive transfer learning* and *transductive transfer learning* (see Table 1). Transductive transfer learning assumes that the conditional probability $f(y|\boldsymbol{x};\boldsymbol{\omega})$ is fixed while the change of data environments is represented by the different distributions of $\boldsymbol{x}$ across the source and target data. Within machine learning literature, this sub-problem is also known as covariate shift (Sugiyama et al. 2008) or sample selection bias (Huang et al. 2007).

Inductive transfer learning has a more general research scope and aims to investigate not only the possible change in $\boldsymbol{x}$, but also the change in the conditional probability $f(y|\boldsymbol{x};\boldsymbol{\omega})$. However, if $f(y|\boldsymbol{x};\boldsymbol{\omega})$ is changing arbitrarily, there is no way to infer a good estimator only based on the source data (Huang et al. 2007). Therefore, to induce the change of $f(y|\boldsymbol{x};\boldsymbol{\omega})$, we must identify some "good" source data that exhibit the same data pattern as the target data (called *same-distribution source data*). Dai et al. (2007) develop a TrAdaBoost classifier to exploit the "good" source data. They adjust the iterative process of the original AdaBoost, by increasing the weights of "good" source data and decreasing the weights of the remaining "bad" source data (called *diff-distribution source data*). A recent study sought to learn the change of $f(y|\boldsymbol{x};\boldsymbol{\omega})$ without relying on same-distribution source data. For instance, Kumagai and Iwata (2018) modeled the change of $f(y|\boldsymbol{x};\boldsymbol{\omega})$ by assuming that $\boldsymbol{\omega}$ in $f(y|\boldsymbol{x};\boldsymbol{\omega})$ follows a Gaussian Process (GP). Compared to this line of work, TrAdaBoost does not require a GP-type view of the dynamic changes, but rather adopts a data-driven approach and adjusts the training of the machine learning algorithm in a desirable direction implied by the same-distribution source data records.

Table 1. Categories of Transfer Learning in Supervised Machine Learning

		Distribution of $\boldsymbol{x}$ across source and target data	Conditional probability $f(y|\boldsymbol{x})$ across source and target data
Traditional machine learning		same	same
Transfer learning	Transductive transfer	different but related	same
	Inductive transfer	same, or different but related	different but related

However, although the weighting scheme of TrAdaBoost is intuitively reasonable, TrAdaBoost does not guarantee to always improve AdaBoost, since the quality of diff-distribution source data is not certain. Moreover, since the weighting scheme of TrAdaBoost is embedded into the iterative process of Adaboost, it may not be directly generalizable to other machine learning algorithms such as XGBoost.

Although the transfer learning literature is continuing growing rapidly, the recent methods are mainly designed for pattern recognition tasks such as image recognition (Yosinski et al. 2014) and text analysis (Ganin et al. 2016). For instance, a deep neural network for classifying one set of animals (e.g. tabby cat versus tiger cat) can be adapted to classify another similar set of animals (e.g. lynx versus leopard). However, given the gap between analyzing unstructured data (such as image and text) and structured data (such as firms' financial statement), these methods are not directly applicable.

IS Literature on Data Analytics in Dynamic Environments

In the IS literature, there exists scarce research that addresses similar problems. Two recent examples are conducted by Meyer et al. (2014) and Saboo et al. (2016). Meyer et al. (2014) proposed a systematic approach called PROCEDO, which employs data mining techniques to iteratively determine conditions under which a dynamic decision making may fail so that users can modify and improve decision making under these conditions. Saboo et al. (2016) employ a time-varying effect analysis to model the regression coefficients as a smooth function of time. Their method provides comprehensive insights into understanding the temporal variation of marketing effectiveness.

Our research is distinguished from existing studies in several aspects. First, our study aims at developing solutions for adapting machine learning algorithms for dynamic data environments. In this sense, Meyer et al. (2014) focus more on the strategic identification of decision failure while we focus on the subsequent improvement of decision models, particularly machine learning models. Moreover, our study focuses on predictive performance while Saboo et al. (2016) provide explanatory insights. As pointed out by Shmueli and Koppius (2011), predictive studies aim at minimizing out-of-sample prediction errors while explanatory studies aim at identifying causal effects. The time effects estimated by explanatory studies are within-sample while the future time effect is unknown. Although it is possible to extrapolate the future time effect according to the previous trend, the extrapolation may lose accuracy when previous trends do not persist.

Application of Inductive Transfer Learning in Quarterly Earnings Prediction

Firms' operating environment varies over time, which could make prediction models built upon historical data obsolete. To alleviate this problem, one strategy is to use a rolling training set of past ten quarters' records (Hou et al. 2012; Li and Mohanram 2014). To predict earnings of time *t*, the training data are from *t-1* to *t-10*. To predict earnings of *t+1*, the training data are from *t* to *t-9*, and so forth.

However, even with the rolling of the training set, it is uncertain whether the past ten quarters' pattern persists in the current quarter. To implement the inductive transfer learning strategy, we need a set of "good" source data to induce the pattern of the target data. We illustrate the formation of the target data, the same-distribution source data and the diff-distribution source data in Figure 1.

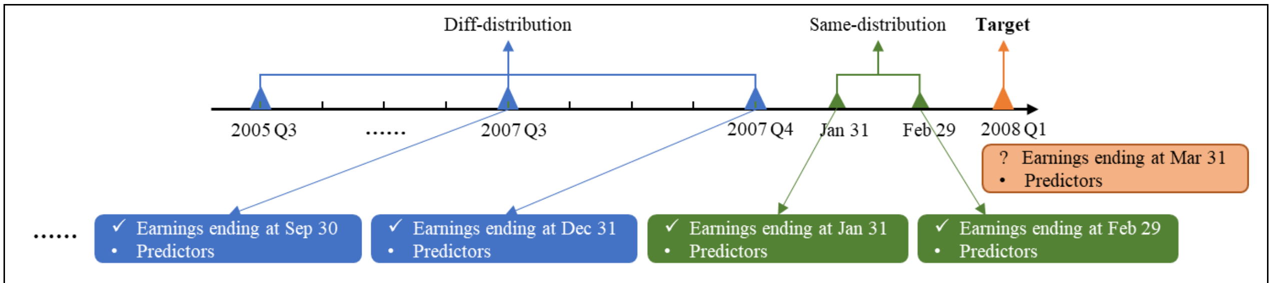

Figure 1. Illustration of Source Data and Target Data

To be specific, in the quarterly earnings prediction task, consider that we were at the end of the first calendar quarter of 2008 (2008 March 31) and were trying to predict earnings for firms with March 31 being a fiscal-quarter-end. The predictor variables are constructed using each focal firm's financial statement in previous quarters. Detailed descriptions of predictors are postponed to the section of data experimentations. In Figure 1, the target data is denoted with the orange block with unknown earnings of March 31.

To train a machine learning model, we need a set of source data. Two small groups of source data are from firms with February 29 and January 31 being the fiscal-quarter-end, respectively. Since these two groups of data contain earnings information of the current quarter, they tend to exhibit the same data pattern as the target data. In Figure 1, they are viewed as the same-distribution source data denoted with the green blocks. The historical data from the previous ten quarters are used as the diff-distribution source data, following the rolling strategy. In Figure 1, diff-distribution source data are represented with blue blocks.

The earnings prediction for other quarters proceeds in a similar manner with all the blocks representing different quarters/months sliding correspondingly. The inductive learning strategy can also be extended to other business applications such as credit default prediction and bankruptcy prediction as long as continuous data can be collected. Moreover, an important feature of inductive transfer learning is that it enables the embedding of domain knowledge to decide the appropriate same-distribution source data.

Proposed Method

In the traditional machine learning setting where the source data and target data are assumed to exhibit the same distribution, we can relatively safely minimize the expected risk/loss function by searching for the parameters of $f(y|\boldsymbol{x};\boldsymbol{\omega})$ that minimize the empirical risk (ERM) of the source data:

$$\boldsymbol{\omega}^* = \arg_{\boldsymbol{\omega}} \min \frac{1}{m} \sum_{(x,y)\in D} [l(\boldsymbol{x}, y; \boldsymbol{\omega})] \,, \tag{2}$$

where D indicates the source data set with sample size m, and $l(\boldsymbol{x}, y; \boldsymbol{\omega})$ indicates the risk/loss function (e.g., mean squared error). However, given that the source data and target data could exhibit different distributions, the weights of the source data records need to be adjusted. Theoretically, the weight of each source data record should be the ratio of its probability following the target distribution to its probability following the source distribution (Pan and Yang 2010), namely:

$$\boldsymbol{\omega}^* = \arg_{\boldsymbol{\omega}} \min \frac{1}{m} \sum_{(x,y)\in D} \left[\frac{\Pr^T(x,y)}{\Pr^S(x,y)} l(\boldsymbol{x}, y; \boldsymbol{\omega})\right] , \tag{3}$$

where $\Pr^T(\cdot)$ and $\Pr^S(\cdot)$ denote the probability distribution of target and source data, respectively. Estimating each of $\Pr^T(\cdot)$ and $\Pr^S(\cdot)$ separately is challenging and an inaccurate estimation could even make this weighting strategy worse off. Thus, researchers focus on approximating this ratio as a whole. Along this stream of literature, existing studies mainly focus on the transductive transfer learning where the conditional probability does not change over time (Sugiyama et al. 2008). In this study, we propose a two-step method for handling inductive transfer learning case.

Given a set of source data records $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots , (\boldsymbol{x}_m, y_m)\}$, where m denotes the sample size of the source data, we divide D to two components, the same-distribution part and the diff-distribution part. The same-distribution data record is denoted with $(\boldsymbol{x}^{SD}, y^{SD})$ and the diff-distribution data record is denoted with $(\boldsymbol{x}^{DD}, y^{DD})$.

Step 1: The first step of our method aims at providing a fundamental description of the change of data distribution. We introduce a selection indicator variable s for each source data record. *Let s=1 if the data record is from the same-distribution subset, and s=0 if the record is from the diff-distribution subset*. In dynamic data environments, the distributional change of data could involve changes in both $\boldsymbol{x}$ and y. We use the following probabilistic model to describe how likely a data record is from the same-distribution subset.

$$\Pr(s = 1|\boldsymbol{x}, y) = g(\boldsymbol{x}, y; \boldsymbol{\varphi}) \,, \tag{4}$$

where $\boldsymbol{\varphi}$ is the vector of unknown parameters. To estimate parameter $\boldsymbol{\varphi}$, we combine the diff-distribution and same-distribution data records. Model (4) then can be estimated with any classification model (e.g., a logistic model or a more advanced machine learning classifier).

Step 2: The second step uses the output from Step 1 to weight all (i.e., same- and diff-distribution) source data records during the training of a machine learning model. Unlike the traditional training of a machine learning model where all the source data records are equally weighted, we set the weight for a same-distribution data record $(\boldsymbol{x}^{SD}, y^{SD})$ as 1 while the weight for a diff-distribution training data record $(\boldsymbol{x}^{DD}, y^{DD})$ as $\Pr(s = 1|\boldsymbol{x}^{DD}, y^{DD})$.[2]

[2] Our weighting scheme is built upon the non-parametric regression with sample correction method (Kim and Yu 2011). Particularly, Kim and Yu (2011) show that one important component of the probability ratio in Equation (3) is the inverse of $\Pr(s = 0|\boldsymbol{x}, y)$. In our implementation, we use $\Pr(s = 1|\boldsymbol{x}, y)$ as a linear approximation. In future work, we plan to implement the model of Kim and Yu (2011) in a more rigorous manner.

A similar weighting method also appears in the transductive transfer learning literature (Huang et al. 2007). However, unlike the existing methods that only model the distributional change of $\boldsymbol{x}$, we exploit the same-distribution source data to provide a more complete description on the change of $\boldsymbol{x}$ and y. Compared to the exiting inductive transfer learning method TrAdaBoost, an advantage of our two-step approach is that we explicitly provide the weights of source data through probability model (4). Conceptually, this design enables the generalizability of our weighting scheme to different machine learning algorithms. Moreover, our method is motivated by the related literature on ERM. This theoretical guideline lays the foundation for further improving our method.

Data Experimentations in Quarterly Earnings Forecast

Data Description

The data set for earnings prediction is obtained from the merged data set from Compustat and CRSP databases. Compustat provides firms' financial statement and CRSP provides information on firms' stock price, common shares outstanding, etc. The sample period of our data set is from Q1 of 1979 to Q2 of 2014. We follow the sample selection process of annual earnings prediction by Li and Mohanram (2014). On average, there are 2991 firms in each year of the sample. However, we exclude firms from finance and utility industries following the common practice of accounting research on financial statements. In addition, we only make predictions for firms with fiscal-quarter end being consistent with calendar-quarter end so that in each quarter we hypothesize that we make prediction on the calendar-quarter end date. Finally, after the sample selection process, the number of firms in each quarter is 2,257 on average.

Since we need ten past quarters for the training set, and certain predictor variables are severely missing in the early time periods, the earliest prediction is made for Q3 of 1983 and the latest prediction is made for Q1 of 2014. In total, the prediction data set consists of 277,611 firm-quarter observations.

Predictor Variables

As a starting point for constructing the earnings prediction model, we refer to the predictors used by Li and Mohanram (2014). More specifically, the prediction model is specified as below:

$$E_{i,t+1} = f\left(E_{i,t}, E_{i,t-1}, E_{i,t-2}, E_{i,t-3}, NegE_{i,t}, NegE_{i,t} \times E_{i,t}, B_{i,t}, TACC_{i,t}, EG_{t+1}, EG_{t+1} \times SIC_i\right)$$

Table 2. Descriptions of Predictor Variables

Notation	Description
$E_{i,t}$	Earnings of firm *i* at quarter *t*
$NegE_{i,t}$	Indicator of negative earnings of firm *i* at quarter *t*
$E_{i,t-1}$, $E_{i,t-2}$, $E_{i,t-3}$	Earnings of firm *i* at quarter *t-1*, *t-2*, and *t-3*, respectively
$B_{i,t}$	Book value of equity of firm *i* at quarter *t*
$TACC_{i,t}$	Total accounting accruals of firm *i* at quarter *t*
EG_{t+1}	Average earnings growth of the newly-reported firms that release earnings in the first or the second month of calendar quarter *t+1*. Since we make prediction on the calendar quarter end, this variable does not result in look-ahead bias.
$EG_{t+1} \times SIC_i$	Interaction term between EG_{t+1} and firm *i*'s Standard Industrial Classification (SIC) codes (12 dummies are created using the first two digits of SIC codes).

Table 2 provides descriptions for the predictor variables. Among the predictors, five ($E_{i,t}$, $NegE_{i,t}$, $NegE_{i,t} \times E_{i,t}$, $B_{i,t}$, $TACC_{i,t}$) are from the annual earnings prediction model by Li and Mohanram (2014). We introduce three additional lagged earnings to account for the seasonality of quarterly earnings. In addition, we introduce EG_{t+1} to let the new information in the first two months of quarter *t+1* be reflected directly in the prediction. Finally, the interaction term $EG_{t+1} \times SIC_i$ allows for the heterogenous effect of newly-reported firms' average earnings growth on firms from different industries.

Prediction Model Construction

We evaluate two inductive learning methods – our proposed method and Two-stage TrAdaBoost.R2 (Pardoe and Stone 2010) which extends the classification oriented TrAdaBoost (Dai et al. 2007) to a regression framework. To implement inductive learning, we use the first two months' records in quarter *t+1* as the same-distribution source data, as mentioned in the previous section. However, due to the extremely scarce same-distribution source data set (sample size<500) compared to the diff-distribution source data set (sample size>20,000), estimation of Model (4) faces an unbalanced-classification issue. We alleviate this issue by setting an additional recent quarter's data (from quarter *t*) as the same-distribution source data. Therefore, the diff-distribution source data are from the earlier nine quarters.

The machine learning algorithm for building the prediction model is Adaboost (from python package sklearn) tuned with default parameters. Future work will also experiment on its extensions, such as XGBoost and LightGBM. As a benchmark of inductive learning, we let all the same- and diff-distribution source data sets weighted equally during the training of Adaboost.

Preliminary Results

Following the accounting literature on earnings forecast, the prediction error for each test record is scaled by stock price (Li and Mohanram 2014). We report two metrics of the scaled error, mean absolute error (MAE) and mean squared error (MSE).

Table 3 presents the earnings prediction performance of the original AdaBoost algorithm (benchmark) and the two inductive learning methods. Particularly, we compare the prediction performance during the great recession period (defined by NBER) and during the whole sample period.

Table 3. Earnings Prediction Performance

Time Periods	Prediction for Quarters in Great Recession (2008 Q1 ~ 2009 Q2)		Prediction for Quarters in Whole Sample Period (1983 Q3 ~ 2014 Q1)	
	MAE	MSE	MAE	MSE
Original Adaboost	0.0567	0.2237	0.0224	0.0362
Proposed Method	0.0532 (6.2%)	0.1992 (11.0%)	0.0223 (0.4%)	0.0347 (4.1%)
TrAdaBoost.R2	0.0544 (4.1%)	0.2405 (-7.5%)	0.0222 (0.9%)	0.0347 (4.1%)

During the great recession period, the data environment could be significantly different compared to the prior periods. Using the original AdaBoost, MAE during the great recession (0.0567) is more than twice the MAE during the whole sample period (0.0224). Comparing the two inductive learning methods with the original Adaboost, we can see that inductive learning is generally beneficial in enhancing the adaptiveness of the prediction model during the great recession period. Our proposed method reduces MAE by 6.2% and MSE by 11.0%, whereas TrAdaBoost.R2 reduces MAE by 4.1% while increases MSE by 7.5%.[3] Compared to TrAdaBoost, our method reduces MAE by 2.2% and MSE by 17.2%.[4]

---

[3] The discrepancy of TrAdaboost.R2 on MAE and MSE could be because it improves the prediction of small-error test records while somehow worsening the prediction of some large-error test records.

[4] In Table 3, we set as additional benchmark a state-of-art transfer learning method – i.e., TrAdaBoost.R2. However, given the times series feature of the earning prediction task, it could be worthwhile to evaluate the performance of time series models. We experimented with different time series models such as AR(*p*) and ARIMA(*p, d, q*). We find that AR(4) achieves the highest prediction accuracy and prediction coverage (estimating a time series model requires enough time series observations of a particular firm thus a more complex time series model tends to exclude newly publicly listed firms to a larger extent). Results show that AR(4)'s prediction error (MAE being 0.0884 for the recession period and 0.0392 for the entire period) is much higher than original AdaBoost.

Comparing the performance during the whole sample period, we can see that inductive learning is still able to improve the prediction. However, the improvement tends to be limited (<1% for MAE and 4.1% for MSE). The performance of inductive learning is within expectation. As the data environment undergoes significant change, it would be especially necessary to adjust the machine learning model built upon historical data set.

The changes in data environments around the economic recession period are illustrated in Figure 2. Figure 2(a) depicts the skewness of three key predictor variables (Earnings, Book value, and Total accounting accruals being scaled to [-1, 1]) over time. Figure 2(b) shows these predictors' importance scores. The red dashed line indicates Q1 of 2008. Figure 2(a) shows large decrease in skewness of earnings and TACC since Q1 of 2008, which indicates longer tail on the negative direction. Untabulated analysis of kurtosis shows its increase during the recession period, which represents increasing probability of obtaining an extreme value during economic recession. Figure 2(b) shows mainly the decrease of earnings' importance and increase of the importance of Book value and TACC for the next quarter's earnings forecast during the recession period. Combining Figure 2(a) and Figure 2(b), our method which incorporates the dynamics of variables, both predictors and the variable to be predicted, through Equation (4) is well motivated.

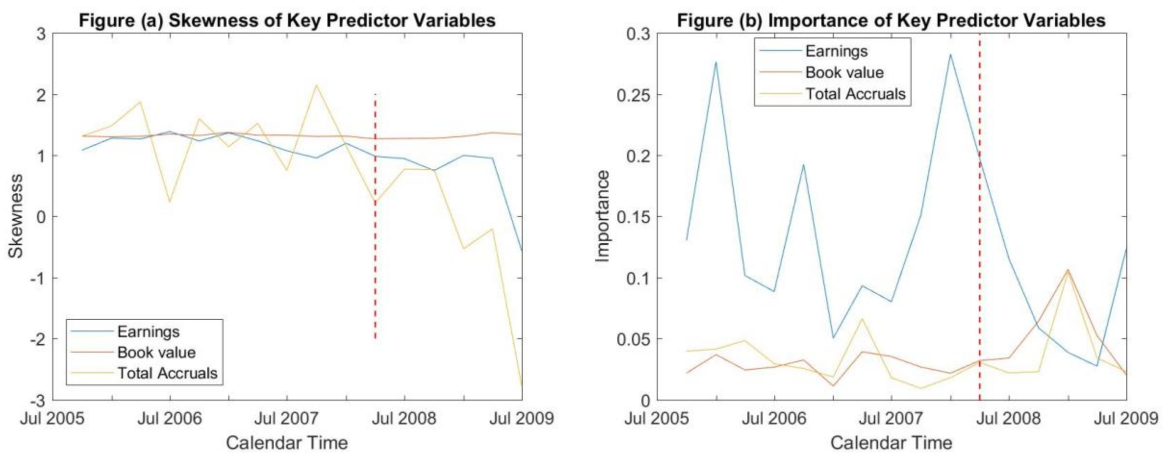

Figure 2. Illusion of Dynamic Business Environments

Conclusions and Future Directions

This study proposes an inductive transfer learning perspective to deal with the dynamic data problem. Through our preliminary experimentation in earnings forecast, we show that our proposed method and the state-of-the-art Two-stage TrAdaBoost.R2 improve the performance of machine learning algorithms when the data environment undergoes significant changes.

To further complete this study, we plan to explore along three directions. First, we aim at conducting a more comprehensive evaluation of inductive learning based on simulations and empirical investigations. With simulations, we plan to evaluate the performance of inductive learning under different extents of dynamic changes, different sample size and different quality levels of the "good" source data. In addition, we aim at applying inductive learning to other business contexts such as credit default prediction of consumer loans. Second, we plan to experiment with additional inductive learning methods, such as the recent one proposed by Kumagai and Iwata (2018). However, an implementation challenge is that this method is for classification. We aim at referring to Pardoe and Stone (2010) to overcome the possible issues in modifying a classifier to a regression model. Third, we plan to further improve the proposed method. As aforementioned, in step 2, we employ a simplified weighting scheme of Kim and Yu (2011). Future improvements include refined implementation of the model of Kim and Yu (2011) and other potentially related statistical models such as inverse propensity weighting (Shao and Wang 2016). Also, our framework is to be implemented on additional machine learning algorithms such as neural networks and XGBoost.

References

Barboza, F., Kimura, H., and Altman, E. 2017. "Machine Learning Models and Bankruptcy Prediction," *Expert Systems with Applications* (83), pp. 405-417.

Dai, W., Yang, Q., Xue, G.-R., and Yu, Y. 2007. "Boosting for Transfer Learning," in *Proceedings of the 24th International Conference on Machine Learning*, Z. Ghahramani (ed.), Corvallis, Oregon, US, June 20-24: ACM, pp. 193-200.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. 2016. "Domain-Adversarial Training of Neural Networks," *Journal of Machine Learning Research* (17:1), pp. 2096-2030.

Grover, V., Chiang, R. H., Liang, T.-P., and Zhang, D. 2018. "Creating Strategic Business Value from Big Data Analytics: A Research Framework," *Journal of Management Information Systems* (35:2), pp. 388-423.

Hou, K., Van Dijk, M. A., and Zhang, Y. 2012. "The Implied Cost of Capital: A New Approach," *Journal of Accounting and Economics* (53:3), pp. 504-526.

Huang, J., Gretton, A., Borgwardt, K. M., Schölkopf, B., and Smola, A. J. 2007. "Correcting Sample Selection Bias by Unlabeled Data," in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, J.C. Platt, D. Koller, Y. Singer and S.T. Roweis (eds.), Vancouver, BC, Canada, 3-6 December, pp. 601-608.

Khandani, A. E., Kim, A. J., and Lo, A. W. 2010. "Consumer Credit-Risk Models Via Machine-Learning Algorithms," *Journal of Banking & Finance* (34:11), pp. 2767-2787.

Kim, J. K., and Yu, C. L. 2011. "A Semiparametric Estimation of Mean Functionals with Nonignorable Missing Data," *Journal of the American Statistical Association* (106:493), pp. 157-165.

Kumagai, A., and Iwata, T. 2018. "Learning Dynamics of Decision Boundaries without Additional Labeled Data," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Y. Guo and F. Farooq (eds.), London, United Kingdom: ACM, pp. 1627-1636.

Li, K. K., and Mohanram, P. 2014. "Evaluating Cross-Sectional Forecasting Models for Implied Cost of Capital," *Review of Accounting Studies* (19:3), pp. 1152-1185.

Makridakis, S., Hogarth, R. M., and Gaba, A. 2009. "Forecasting and Uncertainty in the Economic and Business World," *International Journal of Forecasting* (25:4), pp. 794-812.

Meyer, G., Adomavicius, G., Johnson, P. E., Elidrisi, M., Rush, W. A., Sperl-Hillen, J. M., and O'Connor, P. J. 2014. "A Machine Learning Approach to Improving Dynamic Decision Making," *Information Systems Research* (25:2), pp. 239-263.

Pan, S. J., and Yang, Q. 2010. "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering* (22:10), pp. 1345-1359.

Pardoe, D., and Stone, P. 2010. "Boosting for Regression Transfer," in *Proceedings of the 27th International Conference on Machine Learning*, J. Fürnkranz and T. Joachims (eds.), Omnipress, pp. 863-870.

Saboo, A. R., Kumar, V., and Park, I. 2016. "Using Big Data to Model Time-Varying Effects for Marketing Resource (Re)Allocation," *MIS Quarterly* (40:4), pp. 911-939.

Shao, J., and Wang, L. 2016. "Semiparametric Inverse Propensity Weighting for Nonignorable Missing Data," *Biometrika* (103:1), pp. 175-187.

Shmueli, G., and Koppius, O. R. 2011. "Predictive Analytics in Information Systems Research," *MIS Quarterly* (35:3), pp. 553-572.

Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P. V., and Kawanabe, M. 2008. "Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation," in *Proceedings of the 21st International Conference on Neural Information Processing Systems*, D. Koller, D. Schuurmans, Y. Bengio and L. Bottou (eds.), Vancouver, BC, Canada, December 8-11, pp. 1433-1440.

Vapnik, V. 1995. *The Nature of Statistical Learning Theory*. New York: Springer science & business media.

Yang, Q., and Wu, X. 2006. "10 Challenging Problems in Data Mining Research," *International Journal of Information Technology & Decision Making* (5:4), pp. 597-604.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. 2014. "How Transferable Are Features in Deep Neural Networks?," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence and K.Q. Weinberger (eds.), Montreal, Quebec, Canada, December 8-13, pp. 3320-3328.