Heterogeneous Cross-Company Effort Estimation through Transfer Learning

Shensi Tong*, Qing He*, Yuting Chen*, Ye Yang†, Beijun Shen*‡
*School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China.
†School of Systems and Enterprises, Stevens Institute of Technology, Hoboken, USA
‡bjshen@sjtu.edu.cn

***Abstract*—Software effort estimation is vital but challenging activity during software development. In many small or medium-sized companies, such challenges are stemmed from historical data shortage. The problem can be solved by leveraging cross-company data for effort estimation. While in practice, cross-company effort estimation may not be easy to take because the cross-company data for effort estimation can be heterogenous. In this paper, we propose a novel approach named Mixture of Canonical Correlation Analysis and Restricted Boltzmann Machines (MCR) to address data heterogeneity issue in cross-company effort estimation. The essential ideas in MCR are (1) to present a unified metric representing heterogenous effort estimation data; and (2) to combine Canonical Correlation Analysis and Restricted Boltzmann Machines method to estimate effort in heterogenous cross-company effort estimation. The MCR approach is evaluated on 5 public datasets in PROMISE repository. The evaluation results show that: (1) for estimations with partially different metrics, the MCR approach outperforms within-company effort estimator KNN with a decrease in MMRE by 0.60, an increase in PRED(25) by 0.16, and a decrease in MdMRE by 0.19; (2) for estimations with totally different metrics, the MCR approach outperforms within-company effort estimator KNN with a decrease in MMRE by 0.49, an increase in PRED(25) by 0.08, and a decrease in MdMRE by 0.10.**

I. Introduction

Accurate software effort estimation is vital for the success of software products. Without effective methods, inaccurate effort estimation may lead to many problems: overestimation hinders the acceptance of promising ideas, threatening organizational competitiveness; on the contrary, underestimation may result in schedule and budget overruns, even project cancellation.

Over the last four decades, tremendous software effort estimation methods have been proposed, such as COCOMO [1], Story Points analysis [2] and Function Points analysis [3]. Among them, the most used methods are regression-based, expert-based, and analogy-based methods [4], while none of them are perfect. For example, regression-based methods such as COCOMO [1] need a complicated local calibration procedure [5]. Expert-based methods such as Delphi method [6][7] and planning poker method [8][9] are expensive and time-consuming, especially for small or medium-sized companies, and their performance is heavily depending on the capability of individual experts. Analogy-based methods require a lot of historical data, which can either be collected within company, or be obtained from historical software engineering repository. For small or medium-sized companies, it is very expensive and almost impractical to build their own historical data. On the other hand, software engineering repository data is highly heterogeneous, which makes the sharing of software estimation data across companies very difficult.

TABLE I. Number of common metrics between projects of different companies

Comany A ∩ Company B	Albrecht ∩ China	Albrecht ∩ Kemerer
Number	4	2
Albrecht ∩ Kichenham	Albrecht ∩ Nasa93	China ∩ Kemerer
1	0	2
China ∩ Kichenham	China ∩ Nasa93	Kemerer ∩ Kichenham
2	0	2
Kemerer ∩ Nasa93	Kichenham ∩ Nasa93	
0	0	

A. Motivation

To address the data shortage issue and promote the data shareability and usability across company boundaries, a viable option is to transform heterogeneous data in software engineering repository into homogeneous data. As a result, the heterogeneous data can be used for improving the performance of effort estimation.

Fig. 1 tabulates the number of metrics in effort estimation data of companies including Albrecht [10], China [11], Kemerer [12], Kichenham [13] and Nasa93 [14] and illustrates the detailed metrics used by the five companies. In the figure, the common metrics shared by companies are marked using rectangles in different colors. Specifically, the red rectangle corresponds to the common metrics for Albrecht and China, the blue one for Albrecht and Kemerer, the green one for Albrecht and Kichenham, the orange one for China and Kemerer, the brown one for China and Kichenham, and the purple one for Kemerer and Kitchenham.

Table I shows the number of common metrics for the projects of these five companies.

Obviously, the types of metrics and the size of metric sets in Fig. 1 and Table I vary in these companies. That is, for small or medium-sized companies, the data obtained from software engineering repository is heterogenous.

To address the heterogenous data issues in software effort estimation, this paper proposes a novel, Canonical Correlation Analysis- and Restricted Boltzmann Machines-based transfer learning approach named MCR to heterogenous cross-company effort estimation. MCR advocates an idea of (1) using z-score technology to preprocess the estimation data, (2) taking a unified metric representation for comparing heterogenous effort estimation data among companies, and (3) employing

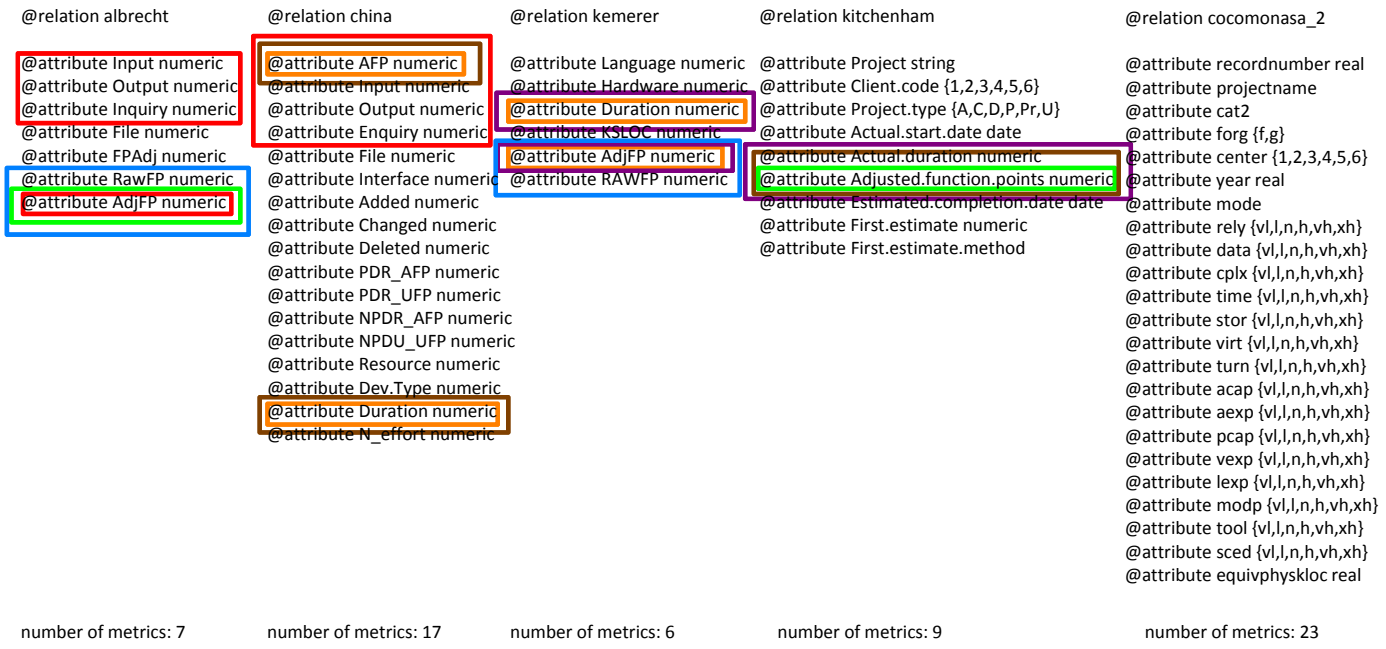

Fig. 1. List of Metrics Used in Effort Estimation Data from Five Companies

Canonical Correlation Analysis and Restricted Boltzmann Machines method to perform heterogenous cross-company effort estimation. MCR also combines Canonical Correlation Analysis and Restricted Boltzmann Machines method to optimize the estimation results.

B. Contributions and Paper Organization

This paper makes three contributions:

1) A novel approach solving the heterogenous cross-company effort estimation problem. Many existing methods for cross-company effort estimation are based on an assumption that the data from the source and the target companies are homogeneous. However, for cross-company effort estimation, the source company data is mostly heterogenous from the target company data. MCR provides with a solution to the heterogenous cross-company effort estimation problem such that effort estimation for small or medium-sized companies can be performed.
2) A combination of the Canonical Correlation Analysis and Restricted Boltzmann Machines methods. MCR combines the Canonical Correlation Analysis and Restricted Boltzmann Machines methods in order to improve performance of heterogenous cross-company effort estimation. The experimental results show that the MCR approach is the best among the within-company effort estimation methods, where the combination of Canonical Correlation Analysis and Restricted Boltzmann Machines can indeed improve the performance of heterogenous cross-company effort estimation.
3) The evaluation of MCR on 5 public datasets. The results reveal that the MCR approach is effective for solving the heterogenous cross-company effort estimation problem. For estimations with partially different metrics, the MCR approach outperforms within-company effort estimator KNN with a decrease in MMRE by 0.60, an increase in PRED(25) by 0.16, and a decrease in MdMRE by 0.19. For estimations with totally different metrics, the MCR approach outperforms within-company effort estimator KNN with a decrease in MMRE by 0.49, an increase in PRED(25) by 0.08, and a decrease in MdMRE by 0.10.

The rest of this paper is organized as follows: Section II surveys related work, section III describes the proposed Canonical Correlation Analysis- and Restricted Boltzmann Machines-based transfer learning approach, section IV shows experimental results, section V concludes.

II. Related Work

We discuss three strands of related work: (1) transfer learning in effort estimation, (2) solutions to heterogenous data in software engineering, (3) Canonical Correlation Analysis and Restricted Boltzmann Machines.

A. Transfer Learning in Effort Estimation

Transfer learning is a machine learning method, which is used to solve the problem in a target domain that is different from but strongly related to a source domain, using the knowledge from the source domain [15]. Based on different situations between the source and target domains and the tasks, transfer learning is divided into three sub-settings: inductive transfer learning, transductive transfer learning and unsupervised transfer learning. In an inductive transfer learning, the target task is different from the source task, no matter whether the source and the target domains are the same. In a transductive transfer learning, the source and the target tasks are the

same, while their domains are different. In an unsupervised transfer learning, similar to the inductive transfer learning, the target task is different from but strongly-related to the source task.

Minku et al. used the transfer learning method in effort estimation, and their results showed that making use of cross-company data can improve performance for effort estimation tasks [16]. In 2014, Minku et al. proposed a new framework to learn the relationship between cross-company and within-company project explicitly, allowing cross-company models to be mapped to the within-company context [17]. Kocaguneli et al. used data on large datasets to verify whether transfer learning is useful in effort estimation, and found that it is effective in solving both the cross-company learning problem and the cross-time learning problem [18]. However, these methods only solve homogenous data instead of heterogenous data, which is usually occurred in cross-company effort estimation.

B. Solutions to Heterogeneous Data in Software Engineering

Solutions do exist to leverage heterogenous data for defect prediction and for missing data imputation for software effort estimation. Jing et al. proposed a Canonical Correlation Analysis based transfer learning method to solve the cross-company defect prediction problem for the first time [19]. Experiments on 14 public heterogenous datasets showed that their method performed well both on partially different metrics and on totally different metrics. Jing et al. also proposed a low-rank recovery and semi-supervised regression method to the missing data imputation problem for software effort estimation [20]. Their experiments on 7 widely used software effort datasets showed that their method can solve several problems such as drive factors missing, effort labels missing, and both. However, when the missing rate increases to 40%, their results is not acceptable, indicating that the method is not suitable to leverage heterogenous data in effort estimation. To the best of our knowledge, there is no effective means of using heterogeneous data in software effort estimation.

C. Canonical Correlation Analysis and Restricted Bolzmann Machines

In statistics, Canonical Correlation Analysis is a way of making sense of cross-covariance matrices. Let two vectors of random variables, X = (X_1, ..., X_n) and Y = (Y_1, ..., Y_m), have correlations among the variables. Canonical Correlation Analysis can find linear combinations of X_i and Y_j having maximum correlation with each other [21]. Canonical Correlation Analysis has been applied in many areas. For example, Du et al. used Canonical Correlation Analysis to identify imaging genetic associations [22], Xing et al. used Canonical Correlation Analysis for multi-view gait recognition [23], Li et al. used Canonical Correlation Analysis in signal processing [24], and Jing et al. used Canonical Correlation Analysis to solve cross-company defeat prediction problem [19]. Because Canonical Correlation Analysis is used to make the distributions of two vectors similar, we believe it can also solve the heterogenous cross-company effort estimation problem.

A Restricted Boltzmann Machines is a generative stochastic artificial neural network that can learn a probability distribution over its set of inputs. Restricted Boltzmann Machines is initially invented under the name of Harmonium by Paul Smolensky in 1986 [25], but only rose to prominence after Geoffrey Hinton and collaborators invented fast learning algorithms for them in the mid-2000s. Nowadays, Restricted Boltzmann Machines has found applications in dimensionality reduction [26], classification [27], collaborative filtering [28], feature learning [29] and topic modelling [30]. They can be either supervised or unsupervised trained, depending on the tasks. Salakhutdinov et al. and Georgiev et al. used Restricted Boltzmann Machines for collaborative filtering in recommendation system to predict the missing value [28][31]. We believe Restricted Boltzmann Machines is also useful for solving heterogenous cross-company effort estimation problem because we can take zero values as missing values to predict them. However, neither Canonical Correlation Analysis nor Restricted Boltzmann Machines has been applied to software effort estimation so far.

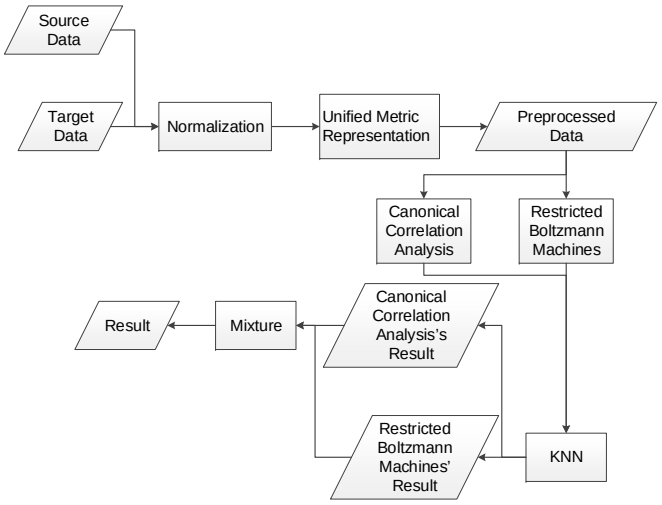

Fig. 2. Framework of Our Approach

III. APPROACH

We have proposed the MCR approach to heterogenous cross-company effort estimation. Fig. 2 shows the framework of our approach. It consists of five steps: normalization, unified metric representation, Canonical Correlation Analysis, Restricted Boltzmann Machines and the mixture. Firstly, we get data from source company and target company. Generally, source company represents small or medium-sized companies who lack historical data, and target company represents the company who is willing to share software projects' data to software engineering repository. Secondly, because the distribution of source company data and target company data is different, we need to normalize them. After normalization, we need to transform heterogenous data into homogenous data using our proposed unified metric representation. Then we use Canonical Correlation Analysis technology and Restricted Boltzmann Machines technology to process the data. Further, we use k-nearest neighbors algorithm to derive the effort estimate. At last, to optimize the results, we mix the results of Canonical Correlation Analysis and Restricted Boltzmann Machines.

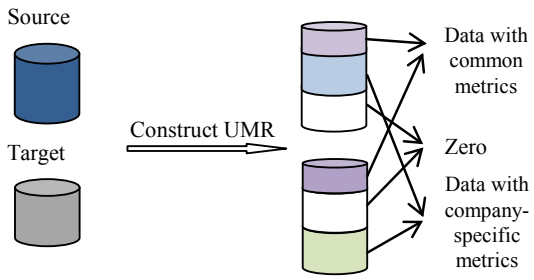

Fig. 3. Illustration of Unified Metric Representation Construction for Heterogenous Source and Target Data

A. Normalization

There are many normalization methods such as z-score, Student's t-statistic and Studentized residual [32]. In our approach, we use z-score technology to normalize the data. The equation is shown in formula (1):

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where z is the z-score value, x is the origin value, μ is the mean of the x value, σ is the standard deviation of the x.

B. Unified Metric Representation

To effectively represent the heterogenous data from two domains, Jing et al. [19] introduced a common subspace to compare source data with target data. Fig. 3. is the illustration of unified metric representation construction for heterogenous source and target data. The unified metric representation is shown in formula (2):

$$\overline{X}_S = \begin{bmatrix} X_S^C \\ X_S^s \\ 0_{(d_t - d_c)\times N} \end{bmatrix} \text{ and } \overline{X}_T = \begin{bmatrix} X_T^C \\ 0_{(d_s - d_c)\times M} \\ X_T^s \end{bmatrix} \tag{2}$$

For source data X_S and target data X_T, X_S^C and X_T^C represent the common metrics, X_S^s and $0_{(d_s-d_c)\times M}$ represent the source-company specific metrics, $0_{(d_t-d_c)\times N}$ and X_T^s represent the target-company specific metrics. It is noted that when there exist no common metrics in the data from two companies, the unified metric representation can be defined as:

$$\overline{X}_S = \begin{bmatrix} X_S \\ 0_{d_t \times N} \end{bmatrix} \text{ and } \overline{X}_T = \begin{bmatrix} 0_{d_s \times M} \\ X_T \end{bmatrix} \tag{3}$$

Using unified metric representation technology, we can compare source data with target data.

C. Canonical Correlation Analysis

Based on unified metric representation technology, we employ the effective transfer learning method, Canonical Correlation Analysis, to make the distributions of source and target company data similar. Canonical Correlation Analysis is used to find a common space for data from two domains so that the correlation between the projected data in the space is maximized.

For example, if we want to investigate the relationship between a person's ability to solve problems X_1 (speed X_{11}, the correct rate of problem solving X_{12}) and his / her reading ability X_2 (reading speed X_{21}, understanding degree X_{22}), we can represent them in Formula (4):

$$\begin{aligned} u &= a_1 x_{11} + a_2 x_{12} \\ v &= b_1 x_{21} + b_2 x_{22} \end{aligned} \tag{4}$$

$$\begin{aligned} \rho_{X_1,X_2} &= corr(X_1, X_2) \\ &= \frac{cov(X_1,X_2)}{\sigma_{X_1}\sigma_{X_2}} \\ &= \frac{E[(X_1-\mu_{X_1})(X_2-\mu_{X_2})]}{\sigma_{X_1}\sigma_{X_2}} \end{aligned} \tag{5}$$

And then we use the Pearson correlation coefficient (Formula (5)) to measure the relationship between u and v, in expectation of finding a set of optimal solutions A and B, which maximize Corr (u, v). Let Σ donates the covariance matrix of X:

$$\Sigma = Var(x) = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \tag{6}$$

where Σ_{11} is cov(X_1, X_1), Σ_{12} is cov(X_1, X_2), Σ_{21} is cov(X_2, X_1), Σ_{22} is cov(X_2, X_2).

After Corr(u,v) is simplified, we can get that:

$$Corr(u, v) = \frac{a^T \Sigma_{12} b}{\sqrt{a^T \Sigma_{11} a}\sqrt{b^T \Sigma_{22} b}} \tag{7}$$

And this is the same to the problem

$$\begin{aligned} &\text{Maximize } a^T \Sigma_{12} b \\ &\text{Subject to: } a^T \Sigma_{11} a = 1, b^T \Sigma_{22} b = 1 \end{aligned} \tag{8}$$

Formula (8) can be solved by generalized eigenvalue problem as:

$$\begin{bmatrix} & \Sigma_{12} \\ \Sigma_{21} & \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \lambda \begin{bmatrix} \Sigma_{11} & \\ & \Sigma_{22} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \tag{9}$$

λ is the generalized eigenvalue corresponding to the generalized eigenvector $\begin{bmatrix} a \\ b \end{bmatrix}$. Suppose that we get p pairs of projective vectors (a,b) corresponding to the largest eigenvalues, we can construct the projective transformation = A [a_1,...,a_p] and B = [b_1,...,b_p].

After obtaining the projected samples $A^T X_1$ and $B^T X_2$, we use the k-nearest neighbor classifier with the Euclidean distance for effort estimation. In our experiments, k is set to 1 for better performance compared with other values.

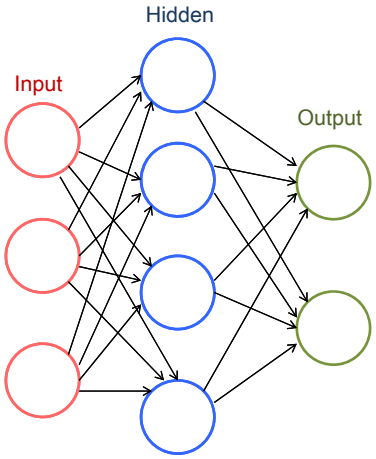

Fig. 4. Artificial Neural Network

D. Restricted Boltzmann Machines

Restricted Boltzmann Machines (Fig. 4) [25] is an Artificial Neural Network (Fig. 5) [33]. In a word, Artificial Neural Network is used to learn an input to output mapping, typically consisting of three layers: the input layer, the hidden layer and the output layer. Restricted Boltzmann Machines is used to learn a mapping from visible units to visible units, which expects the result of the output as close as possible to the input. As shown in Fig. 5, the Restricted Boltzmann Machines' visible units are both input layer and output layer, and the hidden units represents hidden layer.

As shown in Fig. 5, visible units are divided into three parts: common metrics, source-company specific metrics and target-company specific metrics. For source-company specific metrics and target-company specific metrics, there may exist zero values. But after the mapping from visible units to hidden units and the mapping from hidden units to visible units, the results of output may be non-zero values. This is the reason that why we can use Restricted Boltzmann Machines to predict missing values. In our experiments, the number of hidden units are set to half the number of visible units. The seeking of the best number of hidden units is left in our future work.

It is worth noting that we only use Restricted Boltzmann Machines-based method in partially different metrics because Restricted Boltzmann Machines does not perform well for totally different metrics.

E. Combining Canonical Correlation Analysis and Restricted Boltzmann Machines

Prior work suggests that it may be impossible to access which effort estimator is the best. Shepperd et al. warned that when we compare M estimation methods, the ranking of any method may change if the conditions are changed [4]. Kocaguneli et al. revisited the Shepperd et al. results and offered a more optimistic conclusion [11]. They found that if we combined the estimates with multiple estimators, those combined methods performed better than any single estimator. According to their experiments, solo-methods themselves which form top-ranked multimethods were also top-ranked.

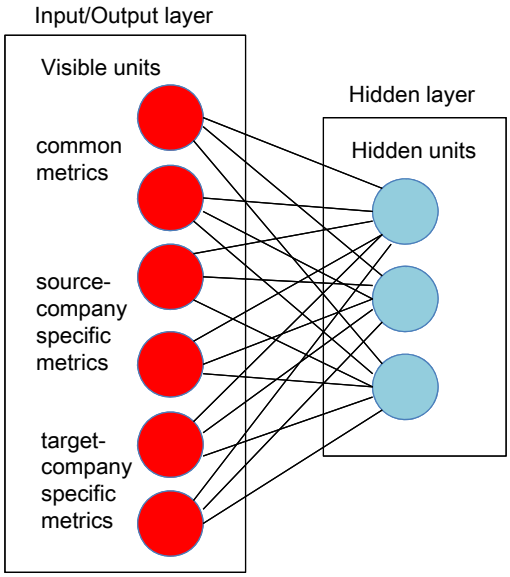

Fig. 5. Restricted Boltzmann Machines

Further, top-ranked multimethods had the greatest stability among any of the 102 methods explored in their study.

Algorithm 1 MCR

Input:
Source company data X_S;
Target company data X_T;
Actual effort of X_S;

Output:
Effort estimation of X_T;

1: Use the z-score normalization to preprocess X_S and X_T
2: For heterogenous data X_S and X_T, search the common metrics from them and construct the unified metric representation as Formula (2) and (3) to obtain $\overline{X}_S$ and $\overline{X}_T$
3: Calculate the covariance metrics Σ_{11}, Σ_{12} and Σ_{22}
4: Obtain the projective transformations A and B by using Formula (9)
5: Based on the obtained $A^T\overline{X}_S$ and $B^T\overline{X}_T$, use the k-nearest neighbor classifier with the Euclidean distance to get the effort estimation results $\hat{X}_{T1}$
6: Using $\overline{X}_S$ and $\overline{X}_T$ as the input of Restricted Boltzmann Machines, obtain the output result $\overline{X}_{S2}$ and $\overline{X}_{T2}$
7: Based on the obtained $\overline{X}_{S2}$ and $\overline{X}_{T2}$, use the k-nearest neighbor classifier with the Euclidean distance to get the effort estimation results $\hat{X}_{T2}$
8: Obtain the final result $\hat{X}_T = Average(\hat{X}_{T1}, \hat{X}_{T2})$
9: **return** $\hat{X}_T$;

Inspired by their findings, we try to combine the Canonical Correlation Analysis and Restricted Boltzmann Machines methods in order to optimize our approach. According to our experiments, Canonical Correlation Analysis and Restricted Boltzmann Machines method are both better than within-company effort estimator. In addition, the mixture of Canonical Correlation Analysis and Restricted Boltzmann Machines method is even better than these two methods. So we finally choose the mixture as our optimal approach.

Algorithm 1 implements our approach. It is worth noting that for data with totally different metrics, $\hat{X}_{T1}$ is the final results.

IV. Experiments

In this section, we first introduce our research questions. Second, we present the data sets and evaluation measures. Last, we conduct experiments of heterogenous cross-company effort estimation with partially different metrics and totally different metrics.

A. Research Questions

To verify that our approach MCR is effective for heterogenous cross-company effort estimation, we propose the following research questions:

RQ1: Is the heterogenous cross-company effort data helpful for cross-company effort estimation? If yes, to what extent?

RQ2: Can multimethod also improve the performance in heterogenous cross-company effort estimation? If yes, to what extent?

B. Data Sets

In our experiments, we employ 5 public available and commonly used datasets as the test data, which includes Albrecht [10], China [11], Kemerer [12], Kichenham [13] and Nasa93 [14]. The Albrecht dataset consists of projects completed in IBM in the 1970s [10]. It contains 24 software projects that are developed by using the third generation languages such as COBOL, PL1, etc. The China dataset includes various software projects from multiple companies developed in China. The Nasa93 dataset is collected with the COCOMO approach [14]. Although the COCOMO dataset has been established for several years, it is still frequently used for validating various effort estimation methods. The Kemerer dataset is a relatively small dataset with 15 software projects described by 6 drive factors and 1 effort label [12].The Kitchenham dataset contains effort data from 145 maintenance and development projects managed by a single outsourcing company. All these 5 datasets are available in the Promise Repository [34].

C. Evaluation Measures

We use three measures, namely Mean Magnitude of Relative Error (MMRE), PRED(25) and Median Magnitude of Relative Error (MdMRE), which are commonly used for evaluating effort estimation accuracy of estimators.

Magnitude of Relative Error (MRE) is defined as:

$$MRE_i = \frac{|e_i - \hat{e}_i|}{e_i} \tag{10}$$

where e_i is the actual effort of project i, $\hat{e}_i$ is the estimate effort of project i.

MMRE is defined as:

$$MMRE = \frac{1}{T}\sum_{i=1}^{T} MRE_i \tag{11}$$

where T is the number of total projects.

PRED(25) is defined as:

$$PRED(25) = \frac{1}{T}\sum_{i=1}^{T} \begin{cases} 1 & \text{if } MRE_i \leq 0.25 \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

MdMRE is defined as:

$$MdMRE = median(MRE_1, MRE_2, ..., MRE_T) \tag{13}$$

For MMRE and MdMRE, the lower the value represent, the better the estimation method perform. And for PRED(25), the higher the value represents, the better the estimation method performs.

D. Experiments with Partially Different Metrics

To validate the effectiveness of the proposed MCR approach when source and target data are partially different, we compare the MCR and within-company estimator. Therefore, a one-to-one heterogenous cross-company effort estimation experiment is carried out. We conduct cross-company effort estimation by using all projects in the source company as the source data, and randomly select 15 percent of all projects from the source data to form the target data. To reduce the randomization error, we repeat this randomization process for one hundred times and derive the mean value as the performance.

Table II shows the MMRE, PRED(25) and MdMRE value of one-to-one heterogenous cross-company effort estimation when partially different metrics exist. M denotes the measurement, CCA refers to Canonical Correlation Analysis, RBM refers to Restricted Boltzmann Machines and KNN refers to within-company effort estimator. In the table, the number presented with boldface denotes the best results in the corresponding estimation scenes.

E. Experiments with Totally Different Metrics

To validate the effectiveness of the proposed MCR approach when source and target data are totally different, we compare MCR with within-company effort estimator. Therefore, we conduct a one-to-one heterogenous cross-company effort estimation experiment here. We conduct cross-company effort estimation by using all projects in the source company as the source data, and randomly select 15 percent of source data to form the target data. To reduce the randomization error, we repeat this randomization process for one hundred times and derive the mean value as the performance.

Table III shows the MMRE, PRED(25) and MdMRE value to one-to-one heterogenous cross-company effort estimation when source and target companies have totally different metrics. M denotes the measurement, CCA refers to Canonical Correlation Analysis, RBM refers to Restricted Boltzmann Machines and KNN refers to within-company effort estimator. In the table, the number presented with boldface denotes the best results in the corresponding estimation scenes.

TABLE II. MMRE, PRED(25) AND MDMRE VALUE OF ONE-TO-ONE HETEROGENOUS CROSS-COMPANY EFFORT ESTIMATION WITH PARTIALLY DIFFERENT METRICS

Source	Target	M	KNN	CCA	RBM	MCR
kemerer	albrecht	MMRE	1.50	**0.29**	0.31	0.30
		PRED(25)	0.15	**0.67**	**0.67**	**0.67**
		MdMRE	0.67	0.19	0.18	**0.14**
albrecht	kemerer	MMRE	1.00	**0.33**	0.48	0.41
		PRED(25)	0.23	0.34	0.28	**0.37**
		MdMRE	0.62	0.46	0.50	**0.41**
kichenham	albrecht	MMRE	0.95	**0.56**	0.92	0.67
		PRED(25)	0.32	0.29	0.25	**0.42**
		MdMRE	0.45	0.39	0.48	**0.34**
albrecht	kichenham	MMRE	1.03	0.96	0.92	**0.77**
		PRED(25)	0.18	0.19	0.21	**0.28**
		MdMRE	0.59	0.53	0.51	**0.41**
china	albrecht	MMRE	1.07	**0.94**	1.21	0.99
		PRED(25)	0.28	0.29	0.25	**0.33**
		MdMRE	0.45	0.39	0.51	**0.38**
albrecht	china	MMRE	1.21	1.42	**0.40**	0.42
		PRED(25)	0.15	0.14	**0.46**	0.42
		MdMRE	0.63	0.66	**0.27**	0.31
kichenham	kemerer	MMRE	0.84	0.77	**0.48**	0.53
		PRED(25)	0.27	0.2	0.13	**0.33**
		MdMRE	0.55	0.70	0.53	**0.49**
kemerer	kichenham	MMRE	1.14	1.01	0.86	**0.71**
		PRED(25)	0.15	0.2	0.21	**0.26**
		MdMRE	0.65	0.52	0.66	**0.51**
china	kemerer	MMRE	0.84	0.98	0.74	**0.69**
		PRED(25)	0.27	0.07	**0.33**	**0.33**
		MdMRE	0.55	0.67	0.70	**0.51**
kemerer	china	MMRE	1.86	1.18	**0.39**	0.71
		PRED(25)	0.15	0.18	**0.48**	0.32
		MdMRE	0.69	0.50	**0.26**	0.40
china	kichenham	MMRE	0.90	0.86	0.82	**0.80**
		PRED(25)	0.24	0.28	0.28	**0.31**
		MdMRE	0.52	0.51	0.65	**0.50**
kichenham	china	MMRE	1.08	1.35	**0.40**	0.49
		PRED(25)	0.19	0.21	**0.51**	0.46
		MdMRE	0.60	0.58	**0.25**	0.28
Average		MMRE	1.12	0.89	0.66	**0.62**
		PRED(25)	0.22	0.26	0.34	**0.38**
		MdMRE	0.58	0.51	0.46	**0.39**

TABLE III. MMRE, PRED(25) AND MDMRE VALUE OF ONE-TO-ONE HETEROGENOUS CROSS-COMPANY EFFORT ESTIMATION WITH TOTALLY DIFFERENT METRICS

Source	Target	M	KNN	MCR
nasa	albrecht	MMRE	1.06	**0.84**
		PRED(25)	0.25	**0.33**
		MdMRE	0.51	**0.34**
albrecht	nasa	MMRE	**4.25**	4.48
		PRED(25)	0.12	**0.13**
		MdMRE	0.95	**0.88**
nasa	china	MMRE	1.16	**0.42**
		PRED(25)	0.16	**0.40**
		MdMRE	0.62	**0.30**
china	nasa	MMRE	1.66	**1.48**
		PRED(25)	0.21	**0.23**
		MdMRE	0.62	**0.58**
nasa	kemerer	MMRE	0.83	**0.59**
		PRED(25)	0.25	**0.40**
		MdMRE	0.56	**0.47**
kemerer	nasa	MMRE	3.40	**2.48**
		PRED(25)	0.14	**0.23**
		MdMRE	1.07	**0.88**
nasa	kichenham	MMRE	**0.98**	1.06
		PRED(25)	0.21	**0.27**
		MdMRE	0.57	**0.55**
kichenham	nasa	MMRE	2.44	**1.38**
		PRED(25)	0.14	**0.25**
		MdMRE	0.77	**0.58**
Average		MMRE	1.71	**1.22**
		PRED(25)	0.18	**0.26**
		MdMRE	0.67	**0.57**

F. Answers to Research Questions

RQ1: Is the heterogenous cross-company effort data helpful for cross-company effort estimation? If yes, to what extent?

From the table above, we can conclude that the heterogenous cross-company effort data is helpful for cross-company effort estimation. For heterogenous cross-company effort estimation with partially different metrics, the MCR approach is better than within-company effort estimator in that its MMRE is decreased by 0.60, PRED(25) increased by 0.16, MdMRE decreased by 0.19 compared with within-company effort estimator on average. For heterogenous cross-company effort estimation with totally different metrics, the MCR approach obtains better effort estimation performance compared with within-project estimator in that its MMRE is decreased by 0.49, PRED(25) increased by 0.08, MdMRE decreased by 0.10 compared with within-company effort estimator on average.

RQ2: Can multimethod also improve the performance in heterogenous cross-company effort estimation? If yes, to what extent?

From the table above, we can conclude that multimethod can also improve the performance in heterogenous cross-company effort estimation. According to the effort estimation results in Table II, we can find that the MCR approach is the best approach in that its MMRE is decreased by 0.27, PRED(25) increased by 0.12, MdMRE decreased by 0.12 compared with Canonical Correlation Analysis-based method and in that its MMRE is decreased 0.04, PRED(25) increased by 0.04, MdMRE decreased by 0.07 compared with Restricted Boltzmann Machines-based method.

V. CONCLUSION

In this paper, we propose an effective approach, MCR to address heterogenous cross-company effort estimation problem, which refers to the cross-company effort estimation scenario where source and target company data have different metrics. We use unified metric representation technology to effectively combine the common metrics, company-specific metrics and an appropriate number of zeros, so we can obtain a unified metric representation for data from two different companies. Based on the unified metric representation, mixture of Canonical Correlation Analysis, an effective transfer learning method to make the data distributions of source and target companies similar, and Restricted Boltzmann Machines, a valid Artificial Neutral Network to predict the missing value, have been raised.

We conduct heterogenous cross-company effort estimation experiments on the 5 widely used open source projects from Promise Repository [34]. We design the one-to-one experiments on partially and totally different metrics to evaluate the performance of the proposed approach MCR. The experimental results indicate that the MCR approach is an effective solution for heterogenous cross-company effort estimation. Especially, for partially different metrics, Canonical Correlation Analysis-based and Restricted Boltzmann Machines-based method are both superior than within-company effort estimator, and the MCR approach is the best among these methods. For totally different metrics, the MCR approach also shows desirable effort estimation effects that is better than within-company effort estimator.

In our future work, we would like to optimize our Restricted Boltzmann Machines-based approach in seeking the best number of hidden units and in questing whether increasing the

number of hidden layers can raise the performance of our approach; to conduct many-to-one heterogenous cross-company effort estimation experiments; to employ more company data that contains both open source and commercial proprietary closed projects to validate the generalization of our approach.

ACKNOWLEDGEMENT

This research is supported by 973 Program in China (Grant No. 2015CB352203) and National Natural Science Foundation of China (Grant No. 61472242).

REFERENCES

[1] Barry W. Boehm. Software engineering economics. *Pioneers and Their Contributions to Software Engineering*, se-10(1):641–686, 1984.

[2] Evita Coelho and Anirban Basu. Effort estimation in agile software development using story points. *International Journal of Applied Information Systems*, 2012.

[3] Jack E. Matson, Bruce E. Barrett, and Joseph M. Mellichamp. Software development estimation using function points. *IEEE Transactions on Software Engineering*, 20(4):275–287, 1994.

[4] Magne Jorgensen and Martin Shepperd. A systematic review of software development cost estimation studies. *IEEE Transactions on Software Engineering*, 33(1):33–53, 2007.

[5] Anandi Hira, Shreya Sharma, and Barry Boehm. Calibrating cocomo® ii for projects with high personnel turnover. In *Proceedings of the International Workshop on Software and Systems Process*, pages 51–55. ACM, 2016.

[6] Norman Dalkey and Olaf Helmer. An experimental application of the delphi method to the use of experts. *Management Science*, 9(3):458–467, 1963.

[7] Bernice B Brown. Delphi process: a methodology used for the elicitation of opinions of experts. Technical report, DTIC Document, 1968.

[8] Roger Buehler, Deanna Messervey, and Dale Griffin. Collaborative planning and prediction: Does group discussion affect optimistic biases in time estimation? *Organizational Behavior and Human Decision Processes*, 97(1):47–63, 2005.

[9] Kjetil Moløkken-Østvold, Nils Christian Haugen, and Hans Christian Benestad. Using planning poker for combining expert estimates in software projects. *Journal of Systems and Software*, 81(12):2106–2117, 2008.

[10] Allan J. Albrecht and John E Gaffney. Software function, source lines of code, and development effort prediction: a software science validation. *IEEE transactions on software engineering*, (6):639–648, 1983.

[11] Ekrem Kocaguneli, Tim Menzies, and Jacky W. Keung. On the value of ensemble effort estimation. *IEEE Transactions on Software Engineering*, 38(6):1403–1416, 2012.

[12] Chris F Kemerer. An empirical validation of software cost estimation models. *Communications of the Acm*, 30(5):416–429, 1987.

[13] Barbara Kitchenham, Shari Lawrence Pfleeger, Beth Mccoll, and Suzanne Eagan. An empirical study of maintenance and development estimation accuracy. *Journal of Systems and Software*, 64(1):57–77, 2002.

[14] A. Dempster. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

[15] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[16] Leandro L. Minku and Xin Yao. Can cross-company data improve performance in software effort estimation? In *International Conference on Predictive MODELS in Software Engineering*, pages 69–78, 2012.

[17] Leandro L. Minku and Xin Yao. How to make best use of cross-company data in software effort estimation? In *International Conference on Software Engineering*, pages 446–456, 2014.

[18] Ekrem Kocaguneli, Tim Menzies, and Emilia Mendes. Transfer learning in effort estimation. *Empirical Software Engineering*, 20(3):813–843, 2014.

[19] Xiaoyuan Jing, Fei Wu, Xiwei Dong, Fumin Qi, and Baowen Xu. Heterogeneous cross-company defect prediction by unified metric representation and cca-based transfer learning. In *Joint Meeting*, pages 496–507, 2015.

[20] Xiao-Yuan Jing, Fumin Qi, Fei Wu, and Baowen Xu. Missing data imputation based on low-rank recovery and semi-supervised regression for software effort estimation. In *International Conference on Software Engineering*, 2016.

[21] David R. Hardoon, Sandor R. Szedmak, and John R. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664, 2004.

[22] L. Du, H. Huang, J. Yan, S Kim, S. L. Risacher, M Inlow, J. H. Moore, A. J. Saykin, and L. Shen. Structured sparse canonical correlation analysis for brain imaging genetics: An improved graphnet method. *Bioinformatics*, 2016.

[23] Xianglei Xing, Kejun Wang, Tao Yan, and Zhuowen Lv. Complete canonical correlation analysis with application to multi-view gait recognition. *Pattern Recognition*, 50(C):107–117, 2015.

[24] Yi Ou Li, Student Member, IEEE, Tulay Adali, Fellow, IEEE, Wei Wang, Student Member, IEEE, Calhoun, and Vince D. Joint blind source separation by multiset canonical correlation analysis. *IEEE Transactions on Signal Processing*, 57(10):3918–3929, 2009.

[25] David E. Rumelhart, James L. Mcclelland, and Corporate Pdp Group. Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations. *Language*, 63(4), 1986.

[26] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[27] Hugo Larochelle and Yoshua Bengio. Classification using discriminative restricted boltzmann machines. In *ICML 08: International Conference on Machine Learning ACM*, pages 536–543, 2008.

[28] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*, pages 791–798, 2015.

[29] Adam Coates, Andrew Y. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. *Journal of Machine Learning Research*, 15:215–223, 2011.

[30] Ruslan Salakhutdinov and Geoffrey E. Hinton. Replicated softmax: an undirected topic model. In *Advances in Neural Information Processing Systems 22: Conference on Neural Information Processing Systems 2009. Proceedings of A Meeting Held 7-10 December 2009, Vancouver, British Columbia, Canada*, pages 1607–1614, 2010.

[31] K. Georgiev and P. Nakov. A non-iid framework for collaborative filtering with restricted boltzmann machines. In *International Conference on Machine Learning*, pages 1148–1156, 2013.

[32] Mariano Ruiz Espejo. *The Oxford Dictionary of Statistical Terms*. Oxford University Press,, 2003.

[33] W. S. Mcculloch and W. H. Pitts. A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1942.

[34] G Boetticher, Tim Menzies, and T Ostrand. Promise repository of empirical software engineering data. *West Virginia University, Department of Computer Science*, 2007.