TRBO: Transfer Learning Accelerated Bayesian Optimization

Haoyang Hu,a Runzhe Liang,a Zhihong Yuana\*

aThe State Key Laboratory of Chemical Engineering, Department of Chemical Engineering, Tsinghua University, Beijing 100084, China

\*Corresponding author’s E-mail: zhihongyuan@mail.tsinghua.edu.cn

Abstract

Despite its success in delivering high-quality solutions for numerous black-box design problems, Bayesian optimization encounters limitations when experimental resources are strictly constrained, known as the small-sample problem. To overcome this hurdle, we propose a novel optimization method named TRBO (TransfeR learning accelerated Bayesian Optimization). TRBO innovatively replaces the conventional Gaussian process surrogate model in Bayesian optimization with modified TrAdaBoost.R2 algorithm, a variation of AdaBoost.R2 specifically designed for transfer learning, in order to leverage knowledge from historical data. Through comprehensive evaluations and ablation studies, we demonstrate that TRBO significantly elevates optimization performance, particularly when confronted with noisy prior knowledge.

**Keywords**: Bayesian optimization, surrogate model, machine learning, transfer learning.

* 1. Introduction

Black-box design problems are pervasive in research and development endeavors, such as the condition design for carrying out chemical reactions and the hyperparameter design for training machine learning models. As the de facto standard solution for gradient-free black-box optimization, Bayesian optimization (BO) attempts to converge within a minimal number of target function evaluations by combining probabilistic surrogate models and acquisition functions (Shahriari et al., 2016).

Despite the high efficiency of BO in utilizing experimental data, the small-sample problem persists in most reaction optimization scenarios, where limited experimental cost constrains the ability of surrogate models to acquire sufficient knowledge within the allowed number of evaluations. On the contrary, in practice, researchers often possess preliminary empirical data from pre-experiments or literature research, which typically follows a different distribution from formal experiments and cannot be directly integrated.

Transfer learning presents a promising resolution to alleviate the small-sample problem, thus introducing novel approaches to accelerating BO (Bai et al., 2023). Several BO variations utilizing self-adaptive transfer learning have been proposed for hyperparameter tuning, such as products of Gaussian process (GP) experts (Schilling et al., 2016) and two-stage training strategy (Li et al., 2022). While these methods focus on the adjustment of GP surrogate models, the possibility of other structures remain unexplored.

In this work, we formulate a novel transfer learning accelerated BO variation named TRBO with ensemble learning-powered surrogate model, which self-adaptively extracts the valuable part of historical data that follows similar distribution to improve its performance. By mitigating the small-sample problem through self-adaptive transfer learning, TRBO represents a valuable contribution to accelerating black-box optimization.

* 1. Transfer learning accelerated Bayesian optimization (TRBO)

The basic idea of TRBO is to adopt modified TrAdaBoost.R2 (Pardoe and Stone, 2010) algorithm instead of popular GP algorithm to construct the surrogate model in BO. TrAdaBoost (Dai et al., 2007), a transfer learning variation of AdaBoost (Freund and Schapire, 1997), concurrently trains a classifier using data from both the source and target domains. TrAdaBoost.R2 is the regressor version of TrAdaBoost. Moreover, since there is some critical difference between general regressors and surrogate models in BO, we modify TrAdaBoost.R2 in TRBO to improve its stability during early iterations. The rest of this section describes the implementation of TRBO.

* + 1. Standard Bayesian optimization framework

The standard BO framework comprises two components: the surrogate model and the acquisition function. The surrogate model depicts an estimation of the input-output mapping relationship of the black-box problem, commonly denoted as the target space. The acquisition function determines the sample point to evaluate next, correspondingly. A typical single iteration of BO can be described as:

First, the surrogate model is built or updated on the currently available dataset . From a Bayesian perspective, this is equivalent to calculating the posterior distribution of the model parameters after observing :

|  |  |
| --- | --- |
|  | (1) |

where and denotes parameter distributions of the updated and the original surrogate model , and can be regarded as a normalizing constant.

Then, the acquisition function is calculated with and optimized to determine the sample next point to evaluate next. For example, the Upper Confidence Bound (UCB) function comprehensively considers the expectation and uncertainty of . It can be written as:

|  |  |
| --- | --- |
|  | (2) |

where and denotes prediction and standard deviation, respectively. Hyperparameter balances exploration and exploitation. Bigger means stronger tendency to exploration.

The selected sample point is merged into after its evaluation on the black-box problem. Such iteration is repeated until the number of iterations reaches a preset value .

* + 1. TrAdaBoost.R2 algorithm

As mentioned above, the surrogate model in BO should estimate both the expectation and uncertainty of a specific sample point in the target space. As an ensemble learning algorithm, TrAdaBoost.R2 implement this through building multiple different weak learners. Mathematically, the training process can be expressed as:

Given the maximum number of iterations, source domain including samples and target domain including samples, the initial sample weight vector is set to:

|  |  |
| --- | --- |
|  | (3) |

The following steps are repeated until the number of weak learners reaches :

The sample weight vector is normalized to first. Then, the -th weak learner is built using on both and , but only its prediction error on is calculated:

|  |  |
| --- | --- |
|  | (4) |
|  | (5) |

The new sample weight vector is updated according to by:

|  |  |
| --- | --- |
|  | (6) |
|  | (7) |

After all weak learners are built, the expectation and uncertainty of the final learner can be estimated by the weighted median and standard deviation of all weak learners respectively, where the weight of the -th weak learner is .

During the training process, inaccurately predicted sample points in are treated as the focus of subsequent learning, leading to an increase in their weights. Conversely, inaccurately predicted sample points in are treated as interference stemming from different data distributions, prompting a reduction of their weights. In this manner, TRBO identifies the transferable subset in , and thus provides valuable knowledge to the surrogate model to expedite exploration in .

* + 1. Implementation modification

In application, we observed instability in the behavior of TRBO due to the scarcity of evaluated sample points in the target domain during early iterations. Two reasons were identified, and corresponding modifications were made to the original implementation.

First, an issue arose from the excessively high average weight assigned to sample points in the target domain during early iterations. To rectify this imbalance between the weights of the source and target domains, an additional hyperparameter *target\_init\_weight\_ratio* (TIWR) was introduced into the weight initialization step. Consequently, Eq. (1) was replaced with Eq. (8):

|  |  |
| --- | --- |
|  | (8) |

Furthermore, we also found that solely factoring in the prediction error on the target domain in Eq. (6) led to an overemphasis on the target domain, and thus hurt the stability due to the inner randomness of acquisition function optimization during early iterations. To give rational weight to prior knowledge, another hyperparameter *target\_error\_ratio* (TER) was introduced to this step by replacing Eq. (6) with Eq. (9):

|  |  |
| --- | --- |
|  | (9) |

* + 1. Flowchart of TRBO

Figure 1 summarizes the overall flow chart of TRBO.

日程表

描述已自动生成

Figure 1 Flowchart of TRBO

* 1. Case studies

For case study, we conducted detailed evaluations and ablation studies on well-known benchmark black-box functions.

* + 1. Benchmark Functions

To comprehensively evaluate the performance of TRBO, we selected three representative benchmark functions: the Rosenbrock function (2 dimensions, 1 global minimum point), the Branin function (2 dimensions, 3 global minimum points), and the Hartman6 function (6 dimensions, 1 global minimum point) All these functions are converted into maximize problems with their global maximum values moved to zero.

* + 1. Source training datasets

As mentioned in the Introduction section, unlike the target domain, which is expensive to explore, the source domain is usually derived from related empirical data following a distribution that may differ slightly or even substantially from formal experiments. Hence, source training datasets comprised more samples with strong noise added to their target function values. Specifically, each source training dataset consisted of 200 points randomly sampled from the target space, with their target function values multiplied by a random interference factor ranging from 0.5 to 2.0 as noise.

* + 1. Control groups for ablation study

Traditional method to leverage prior knowledge is to directly pretrain the GP model with known sample points. So, standard BO with GP surrogate model pretrained on the source training dataset was used as a control group, hereinafter referred to as pretrained BO.

To evaluate the difference between GP and boosting models, standard BO using normal AdaBoost.R2 algorithm for building surrogate model was also used as a control group, hereinafter simply referred to as AdaBoost.R2. Here, AdaBoost was also pretrained on the source training dataset for fair comparison with TRBO.

Apart from BO, heuristic algorithms are also proved to be powerful solutions for black-box optimization problems, such as genetic algorithm and evolution strategy. We used Covariance Matrix Adaptation Evolution Strategy (CMA-ES) (Hansen, 2006), a popular variation of evolution strategy, as a control group to validate the effectiveness on selected benchmark functions of BO.

* + 1. Hyperparameters

For standard BO and pretrained BO, Matern 2.5 kernel with =1 was adopted. For AdaBoost.R2 and TRBO, decision tree regressor with *max\_depth*=6 was adopted as the weak learner, and the number of weak learners was set to 25 to avoid overfitting on the source domain. UCB acquisition function with decay from =10 to 0.1 was applied to all BO-based algorithms. For CMA-ES, population size was set to 3, the minimum allowed value, to simulate limited parallel evaluation capability in most applications.

Up to 50 target function evaluations, 1/4 of the source training dataset capacity, were allowed for each algorithm to simulate the expensive evaluation cost. To eliminate the influence of randomness, each experiment was repeated 10 times with different random seeds independently. All experiments were performed on an Intel Core i7-8700 CPU.

* 1. Results and discussion

The means and standard deviations of maximum target function values found by all algorithms on benchmark functions are presented in Table 1.

Table 1 Means and standard deviations of maximum target function values found

|  |  |  |  |
| --- | --- | --- | --- |
|  | Rosenbrock | Branin | Hartman6 |
| Standard BO | -(1.98±0.227)×10-2 | -(1.93±1.19)×10-3 | -0.389±0.0635 |
| Pretrained BO | -(501±1.93)×10-2 | -(201±8.74) ×10-3 | -0.351±0.0980 |
| CMA-ES | -(56.5±0.228) ×10-2 | -(329±1.37) ×10-3 | -0.615±0.119 |
| AdaBoost.R2 | -(611±2.86)×10-2 | -(64.9±3.87)×10-3 | -0.637±0.100 |
| TRBO | **-(1.72±0.389)×10-2** | **-(1.69±1.34) ×10-3** | **-0.345±0.0800** |

For clarity, the maximum target function value curve corresponding to each run with the best performance among 10 independent runs is depicted in Figure 2.

图示, 示意图

描述已自动生成

Figure 2 Maximum target function value curves of the best performing runs

Both Table 1 and Figure 2 demonstrate that TRBO consistently outperformed all control groups across various benchmark functions, underscoring its universality in tackling diverse problems. Notably, TRBO exhibited significantly accelerated speed on the Branin function and Hartman6 function, indicating its adept utilization of prior knowledge.

For ablation study, taking the two-dimensional Rosenbrock function as a visible example, detailed optimization processes of the best performing runs are presented in Figure 3.

图示

描述已自动生成

Figure 3 Detailed optimization processes of the best performing runs on the Rosenbrock function

Figure 3 shows that, following the introduction of simulated noisy prior knowledge, pretrained BO was disturbed by differences in data distribution, resulting in diminished performance. In practice, although the quantity of historical data may be substantial, its relevance to formal experiments is often not assured. Therefore, in such cases, the source training dataset cannot be directly employed for target space modeling, and standard BO using GP surrogate model cannot be directly adopted by transfer learning.

Figure 3c shows that, CMA-ES could rapidly identify the approximate orientation of the global maximum, but hardly the precise location. So, under stringent limitations on target function evaluations, the performance of evolution strategy is usually inferior to BO.

Figure 3d shows that, AdaBoost.R2 was also disturbed by differences in data distribution like standard BO, with its performance even worse than that of BO. This implies that, GP indeed outperforms AdaBoost.R2 in normal optimization situations, justifying GP as the default choice for BO. However, Figure 3e underscores that TRBO, bolstered by prior knowledge, not only mitigated the drawbacks of AdaBoost.R2 but also navigated interference from the source domain.

In summary, the above ablation study validates that, the modified TrAdaBoost.R2-based surrogate model is the key of TRBO to enable self-adaptive transfer learning.

* 1. Conclusion

A transfer learning accelerated black-box optimization algorithm TRBO is proposed under the BO framework to effectively harness historical data with unguaranteed data distribution. The key innovation of TRBO is the modified TrAdaBoost.R2 algorithm which self-adaptively identifies the transferable subset of the historical data to enhance the learning process of surrogate models, thereby elevating overall performance. Given the ubiquity of similar transferable datasets in various design problems, TRBO holds promise for a broad spectrum of applications. Furthermore, since boosting models are open due to their ensemble structure, TRBO can be easily extended to other types of data-driven learners, showcasing its potential for continuous refinement and expansion.

References

Bai, T., Li, Y., Shen, Y., Zhang, X., Zhang, W., Cui, B., 2023. Transfer Learning for Bayesian Optimization: A Survey. arXiv preprint.

Dai, W., Yang, Q., Xue, G.R., Yu, Y., 2007. Boosting for transfer learning, in: ACM International Conference Proceeding Series. pp. 193–200.

Freund, Y., Schapire, R.E., 1997. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. J Comput Syst Sci 55, 119–139.

Hansen, N., 2006. The CMA evolution strategy: A comparing review. Studies in Fuzziness and Soft Computing 192, 75–102.

Li, Y., Shen, Y., Jiang, H., Zhang, W., Yang, Z., Zhang, C., Cui, B., 2022. TransBO: Hyperparameter Optimization via Two-Phase Transfer Learning, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 956–966.

Pardoe, D., Stone, P., 2010. Boosting for regression transfer, in: ICML 2010 - Proceedings, 27th International Conference on Machine Learning. pp. 863–870.

Schilling, N., Wistuba, M., Schmidt-Thieme, L., 2016. Scalable hyperparameter optimization with products of gaussian process experts, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 33–48.

Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., De Freitas, N., 2016. Taking the human out of the loop: A review of Bayesian optimization. Proceedings of the IEEE 104, 148–175. https://doi.org/10.1109/JPROC.2015.2494218