Proceedings of the 2023 International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/8/20230095

Exploring the advancements and challenges of automated machine learning

Zhengyang Jin

Computer Science, University of Electronic Science and Technology of China，Zhongshan Institute, Zhongshan, Guangdong, 528400, China

1625159433@qq.com

Abstract. Automl,a rapidly growing field which is aiming to apply the machine to solve problems that human can't easily deal with . This includes tasks such as feature selection, model selection, and hyperparameter tuning. One of the many advantages about Auto Ml is that it can greatly shorten the cost of researchs and resources cost by applying machine learning to a problem. This makes it accessible to a wider range of users, including those without a background in computer science or statistics.In spite of some advantages of AutoML, many challenges are waiting to be addressed. The main challenge is that it is often challenging to ensure that the models generated by AutoML are of high quality and generalize well to new data. Another challenge is that AutoML can be computationally expensive, which can make it infeasible for some problems. Overall, AutoML has the potential to revolutionize the way we apply machine learning to real-world problems, but it is important to be aware of its limitations and challenges.

Keywords: feature selection, feature construction, model selection, hyperparameter tuning, ensemble learning, greedy algorithm, random search, grid search, organic search, model fusion.

1. Introduction

Automatic Machine Learning (AutoML) is a rapidly growing field within the machine learning community that aims to make the process of building predictive models more accessible and efficient. The purpose of AutoML is to streamline the complete process of machine learning, including data preparation, feature creation, model selection, and hyperparameter optimization, thereby freeing data scientists from repetitive and time-consuming tasks. With the rise of big data and the increasing demand for predictive models, the demand for AutoML solutions has also risen. This advancement has brought forth a vast array of AutoML tools and methods, now utilized in various industries. and academia to build high-quality predictive models with ease.

AutoML has the potential to revolutionize the way we work with machine learning by providing a user-friendly and efficient platform for building predictive models just like Alpha Go[1]. It has the potential to make machine learning more accessible to a wider audience, including those without extensive technical backgrounds. Additionally, AutoML can help data scientists to save time and focus on more important tasks, such as problem-solving and data interpretation.

Despite its many advantages, AutoML still faces obstacles that must be overcome to reach its full potential. include several key obstacles, include the complexity of the machine learning pipeline, the

DOI: 10.54254/2755-2721/8/20230095

diversity of data types, and the need for accurate performance evaluation. Nevertheless, the future of AutoML looks bright, and its significance cannot be ignored in shaping the future of machine learning and data science..

2. Techniques

There are several techniques used by auto ml (Automated Machine Learning) to automate the creation and implementation of machine learning models. Several of the primary methods:Automatic Feature Selection,Automatic Hyperparameter Tuning[2], Automatic Model Selection,Automated Model Fusion,Automated Model explanation,Automatic Feature Engineering[3,4].All these techniques are used by automl to simplify the creation and implementation of machine learning models, making it more available to more users. Additionally, automl also utilizes other methods such as evolutionary algorithms, gradient-based optimization, and Bayesian optimization to make the task of fine-tuning and optimizing ML models more efficient.

2.1 Automatic feature engineering

Automatic feature engineering is a technique used in AutoML that aims to automate the generation of novel features[5,6] from raw data. This step is pivotal in the machine learning process as it holds immense sway on the model's efficiency. Automatic feature engineering is trying to enhance the efficacy of the machine learning model by generating new features that reveal the intrinsic patterns in the data.There are several different types of automatic feature engineering methods that can vary depending on the issue at hand and the type of the data..

Statistical methods, such as correlation, chi-squared test, mutual information etc to evaluate the correlation between various features and the target variable. These methods are useful for identifying the most informative features for a given problem.

Transformations methods, on the other hand, involve applying mathematical functions to the raw data to create new features. These include logarithmic, square root, and reciprocal transformations, etc. These methods are useful for transforming the data into a more suitable format for machine learning algorithms.

Aggregation methods create new features by aggregating or summarizing the values of existing features. These methods are useful for creating new features that capture the overall trend or pattern in the data.

Synthetic feature methods create new features by combining existing features in different ways. These methods are useful for creating new features that capture more complex relationships between features.

With the combination of these methods, automatic feature engineering can be performed to extract the valuable information from raw data and improve the performance. However, it is important to note that it can be computationally intensive and may require a significant amount of data to be effective. Additionally, it should be performed with caution as the new features generated may not be always relevant or generalizable to unseen examples.

2.1.1 Automated feature selection.

A technique used in AutoML to select a selection of features from a comprehensive set of features that are most relevant to the problem at hand.. It commonly considered in feature generation, e.g.[7], The objective of automated feature selection is to enhance the machine learning model's performance by lowering the data's dimensionality and eliminating unimportant or redundant features.

Several distinct methods of automated feature selection exist, each with their own strengths and weaknesses. Filter methods, for example, use statistical tests to evaluate the relationship between each feature and the target variable. Features with a strong correlation to the target variable are retained, while others are discarded. Wrapper methods, also, by employing a designated machine learning algorithm to gauge the performance of a selection of features. The selected features that results in the best performance is selected. Embedded methods include feature selection as part of the machine

DOI: 10.54254/2755-2721/8/20230095

learning algorithm's training process. Hybrid methods combine elements from filter, wrapper, and embedded methods to leverage the advantages of each approach.

It is important to note that automated feature selection should be performed with caution as it can introduce bias, specially when the data set has a small size or the selected feature set is not generalizable to unseen examples. Additionally, it can be computationally intensive and may require a significant amount of data to be effective.

In conclusion, automated feature selection is an important step in the machine learning pipeline that can enhance the model's efficiency by selecting the most relevant features from a larger set of features. However, it should be performed with caution and understanding the limitations of the chosen method.

2.1.2. Automated feature construction

Automated Feature Construction refers to the automatic generation of new features from raw data for utilization in machine learning models. Explore Kit [8] is one of the most prominent works in automatic feature engineering, with the goal of creating new features to enhance the efficacy of machine learning algorithms.This technique is used to improve model accuracy and performance by transforming the original data into a more informative representation. The goal of automated feature construction is to extract the most relevant information from the data and use it to build a better model. This can be done through various techniques, such as feature extraction, feature transformation, and feature synthesis. Automated feature construction can help to reduce the need for manual feature engineering, permitting data scientists to concentrate on other aspects of the modeling process and potentially improve the speed and efficiency of the modeling process as well.

2.2. Automated model selection

A technique used in AutoML to determine the optimal machine learning model for a specific issue. The purpose of automated model selection is to enhance the machine learning model's performance by selecting the model that most fittingly aligns with the data and issue.There are several different types of automated model selection methods, each with their own strengths and weaknesses.

One method is Bayesian optimization[8-10], it is a probabilistic model-based approach that uses a prior over the space of possible hyperparameters[11-13], and it uses the information from previous evaluations to guide the search for the best hyperparameters.

It is important to note that automated model selection should be performed with caution as it can be computationally intensive and may require a significant amount of data to be effective. Additionally, it should be performed with caution as the selected model may not be always generalizable to unseen examples.

In conclusion, automated model selection is a technique used in AutoML aiming to find a better solution. It helps enhance the result of the model by finding a solution that best fits the problem. However, it should be performed with caution and understanding the limitations of the chosen method.

The Greedy Algorithm is a method that solves optimization problems by making the most suitable decision at each step in hopes of reaching the ideal global solution. It builds a solution by choosing the best option and feature[14] at each step, until a complete solution is found.For instance, in the Neural Architecture Search (NAS) problem, determining the architecture for each layer is required, and a greedy search is utilized for [15] multi-attribute learning problems.It can be applied to various problems and is simple to implement, but doesn't guarantee the optimal solution and may lead to suboptimal ones.

2.3. Automated hyperparameter tuning

Being known as Hyperparameter Optimization, is the process of automating the task of selecting the optimal combination of hyper-parameters for a machine learning model, also can be highly beneficial in some situations [16]. Hyper-parameters are variables set prior to training and not acquired during the training process and determine the behavior of the model.

DOI: 10.54254/2755-2721/8/20230095

The task of determining the optimal set of hyper-parameters can be labor-intensive. and requires expertise in the specific model and domain. Automated Hyperparameter Tuning aims to automate this process through utilizing methods like Grid Search, Random Search[17], and Bayesian Optimization to locate the optimal set of hyper-parameters. Another automated hyperparameter tuning method is Evolutionary Algorithm, which simulates natural selection process to find the optimal set of hyper-parameters. This method is good at handling high-dimensional, discontinuous and noisy search space.

In conclusion, Automated Hyperparameter Tuning is the automation of determining the optimal set of hyper-parameters. There are several techniques that can be used to automate this process, such as Grid Search, Random Search, Bayesian Optimization, and Evolutionary Algorithm. Each approach has its own pros and cons, and the method selection will depend on the particular model and domain.

Grid search is an approach to hyperparameter optimization in which a set of models are trained with a range of hyperparameter values specified in a grid. The objective of grid search is to identify the combination of hyper-parameters that results in optimal performance of the selected model.

The process of grid search involves by defining a range of values for each relevant hyperparameter, creating a grid of all possible combinations of these values, and training a model for each combination of hyperparameter values. The performance of each model is then evaluated using a specified metric, such as accuracy or F1-score. The combination of hyper-parameters that results in the best performance is then selected as the best set of hyper-parameters for the machine learning result.

However, one of its disadvantages is that it can be computationally expensive, especially for models with a large number of hyper-parameters. Additionally, grid search may miss the optimal solution if the grid is too coarse or if the optimal solution lies between two grid points. Despite these limitations, grid search remains a popular method for hyperparameter tuning due to its simplicity and versatility.It can help finding the optimal solution by thoroughly examining all possible combinations, making it easy to identify the best set. Grid search is a simple but powerful technique for hyperparameter optimization. It can be used to determine the optimal combination, which can result in improved performance on unseen data. But it's also has its own limitation, it's computationally expensive, and might not be able to find the optimal set of hyper-parameters in some cases.

2.4. Automated model fusion

The purpose of Automated Model Fusion is to enhance the accuracy of a machine learning system by fusing the predictions made by multiple models. By combining the predictions, the performance of the system is optimized, resulting in improved results. The idea behind this technique is that different models have different strengths and weaknesses, and by combining their predictions, a more robust and accurate model can be created.

There are various methods for combining the predictions of multiple models such as averaging, weighted averaging, and stacking. In averaging method predictions of multiple models are combined by taking the average of their predictions. In weighted averaging, different models are given different weights based on their performance. In stacking method, The models' predictions are fed into a meta-model, which then makes the final prediction.

The advantage of automated model fusion is that it can improve the system by leveraging the strengths of multiple models. However, it can be computationally expensive and require large amounts of data, particularly when the number of models being combined is high.

Ensemble learning is aiming to turn multiple individual models to create a single, more powerful model[18]. It can be achieved by using different algorithms, or with different hyperparameters, or combining multiple models that have been trained on different subsets of the data.

There are several popular techniques for ensemble learning, including bagging, boosting, and stacking. Bagging creates multiple versions of a model, the final prediction is obtained by combining the predictions of multiple models, each trained on a different random portion of the data, through averaging or voting. Boosting creates multiple versions of a model, which trained on distinct subsets of the data, are used to make the final prediction, with more emphasis on the samples that were

DOI: 10.54254/2755-2721/8/20230095

misclassified by the previous models. Ensemble learning can lead to a more accurate overall model and can be employed to restrain the sample diversity, which can lead to more robust predictions.

3. Prospect

The prospect of automl (Automated Machine Learning) is quite promising. Automl has the potential to democratize machine learning through making it available to a broader user base, including those without extensive technical expertise. This can lead to more widespread adoption of machine learning about various projects, like healthcare, manufacturing and finance.

One of the main advantages of automl is that it can save time and resources by automating many of the tedious and time-consuming tasks involved in building and deploying models. This include tasks like model selection, feature engineering, and hyperparameter tuning. Automl can also enhance the performance of models through automating the search for optimal model architectures and hyperparameters.

Another advantage of automl is its interpretability. Automl can provide insights into which features are most important for a given task, and can also help to identify potential biases in the data. This will rise transparency and accountability of models.

In addition, automl also has the potential to refine the scalability of the process. Automating the tasks involved in building and deploying machine learning models can help to reduce the need for manual intervention, which can be a bottleneck when working with large and complex datasets.

However, automl also poses some challenges, one of the key challenges is that automl requires large amounts of data and computational resources. This can make it difficult to apply automl to small or resource-constrained datasets. Additionally, automl can be difficult to interpret and explain, which can make it difficult to build trust in the models.

Overall, the prospect of automl is quite promising as it has the potential to simplify the difficulty in constructing and deploying samples, and to expand access to machine learning to a broader user base. However, there are also some challenges associated with automl that must be overcome for it to fully realize its potential.

4. Conclusion

There are several bottlenecks that can be associated with automl technology. One of the main bottlenecks is the computational cost. Automl requires a substantial quantity of data and computation power to evaluate models, which can be a significant challenge for organizations with limited resources.

Another bottleneck is the difficulty in ensuring the quality and robustness of the models generated by automl systems. Automl systems can generate a large number of models, and it can be challenging to identify which models are most suitable for a given task. Additionally, automl systems can be sensitive to the trait and diversity of the training data, which can lead to models that are not robust or generalizable.

A third bottleneck is the lack of transparency and interpretability of automl systems. Since automl systems are often highly complex, it can be difficult to understand how a given model is making its predictions, which can make it difficult to identify and correct errors.

Finally, automl systems can be vulnerable to adversarial attacks, which can lead to models that are easily fooled by carefully crafted inputs, leading to poor performance on unseen data.

In summary, the main bottlenecks of automl technology include high computational cost, difficulty in ensuring the quality and robustness of the generated models, lack of transparency and interpretability, and vulnerability to adversarial attacks.

DOI: 10.54254/2755-2721/8/20230095

References

[1] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot et al., "Mastering the game of Go with deep neural networks and tree search," Nature, vol. 529, no. 7587, p.484, 2016.

[2] M. Feurer and F. Hutter, "Hyperparameter optimization," 2018.[Online]. Available: https://www.ml4aad.org/wp-content/uploads/2018/09/chapter1-hpo.pdf.

[3] G. Katz, E. C. R. Shin, and D. Song, "Explorekit: Automatic feature generation and selection," in International Conference on Data Mining, 2016, pp. 979–984.

[4] J. M. Kanter and K. Veeramachaneni, "Deep feature synthesis: Towards automating data science endeavors," in IEEE International Conference on Data Science and Advanced Analytics, 2015, pp.1–10.

[5] G. Katz, E. C. R. Shin, and D. Song, "Explorekit: Automatic feature generation and selection," in International Conference on Data Mining, 2016, pp. 979–984.

[6] J. M. Kanter and K. Veeramachaneni, "Deep feature synthesis: Towards automating data science endeavors," in IEEE International Conference on Data Science and Advanced Analytics, 2015, pp.1–10.

[7] B. Tran, B. Xue, and M. Zhang, "Genetic programming for feature construction and selection in classification on high-dimensional data," Memetic Computing, vol. 8, no. 1, pp. 3–15, 2016

[8] G. Katz, E. C. R. Shin, and D. Song, "Explorekit: Automatic feature generation and selection," in International Conference on Data Mining, 2016, pp. 979–984.

[9] K. Swersky, J. Snoek, and R. P. Adams, "Freeze-thaw bayesian optimization," arXiv preprint arXiv:1406.3896, 2014.

[10] T. Nickson, M. A. Osborne, S. Reece, and S. J. Roberts, "Automated machine learning on big data using stochastic algorithm tuning," arXiv preprint arXiv:1407.7969, 2014.

[11] C. Thornton, F. Hutter, H. Hoos, and K. Leyton-Brown, "AutoWEKA: Combined selection and hyperparameter optimization of classification algorithms," in ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, pp.847–855.

[12] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," in Advances in Neural Information Processing Systems,2015, pp. 2962–2970.

[13] L. Kotthoff, C. Thornton, H. Hoos, F. Hutter, and K. LeytonBrown, "Auto-WEKA 2.0: Automatic model selection and hyperparameter optimization in WEKA," Journal of Machine Learning Research, vol. 18, no. 1, pp. 826–830, 2017.

[14] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," IEEE Transactions on Information theory, vol. 50,no. 10, pp. 2231–2242, 2004.

[15] S. Huang, X. Li, Z. Cheng, Z. Zhang, and A. G. Hauptmann,"GNAS: A greedy neural architecture search method for multiattribute learning," in ACM Multimedia, 2018, pp. 2049–2057.

[16] K. Eggensperger, F. Hutter, H. H. Hoos, and K. Leyton-Brown,"Efficient benchmarking of hyperparameter optimizers via surrogates," in AAAI Conference on Artificial Intelligence, 2015.

[17] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," Journal of Machine Learning Research, vol. 13, no. Feb, pp. 281–305, 2012.

[18] Guo Y, Huang J, Dong Y et al. Guoym at SemEval-2020 task 8: Ensemble-based classification of visuo-lingual metaphor in memes. In Proceedings of the Fourteenth Workshop on Semantic Evaluation. Barcelona (online): International Committee for Computational Linguistics, pp. 1120–1125. URL https://aclanthology.org/2020.semeval-1.148.