SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

AutoML-Enabled Data Analytics for Scalable and Adaptive Predictive Modeling in Large-Scale Datasets

Rajitha Gentyala[1]

[1] Independent Researcher Texas, USA. E-mail: rajitha.gentyal@gmail.com

Abstract
As large, diverse datasets proliferate across domains such as industry, healthcare, finance, and science, the need for scalable, adaptable predictive modeling pipelines is growing. To address this need, Automated Machine Learning (AutoML) has become a practical solution that integrates data preprocessing, feature engineering, model selection, hyperparameter tuning, and ensembling into an automated, unified pipeline. This study presents a framework for large-scale data analytics that leverages AutoML to achieve scalable and adaptive predictive modeling in data-rich contexts. The framework adopts distributed ensembling, model selection with meta-learning, and Bayesian hyperparameter optimization to ensure predictive capacity and computational economy while handling large volumes of data. The framework was tested on synthetic and benchmark datasets ranging from 10K to 50M, using five benchmark learners, namely Random Forest, XGBoost, LightGBM, a feedforward neural network, and a weighted ensemble meta-learner. These results show that the AutoML-enabled pipeline consistently achieved higher predictive accuracy than any single base learner, with a mean (across validation folds) cross-validation accuracy of 89.4% for the AutoML search space; meanwhile, the weighted ensemble configuration had an order of magnitude lower development time for large data volumes than non-AutoML, manually tuned pipelines. The framework worked well in large-scale deployments, as scalability tests were performed using 64 compute nodes, showing near-linear performance increases across the nodes. These conclusions are placed in the context of previous work on the automation of machine learning and hyperparameter optimization, and big data analytics, while noting constraints on generalizability to other domains, interpretability, and computational requirements. Overall, the study shows that the use of AutoML-based analytics offers a plausible route toward democratizing predictive modeling for large-scale data in the future, provided that adaptive resource management and explanation mechanisms are included in future implementations.

Keywords: automated machine learning; AutoML; scalable predictive modeling; big data analytics; hyperparameter optimization; ensemble learning; adaptive data pipelines.

1. Introduction

Data volumes have been creating data challenges that far surpass the capabilities of linear statistical and machine learning processes. Every industry, from financial services to healthcare, manufacturing, retail, and telecommunications, is increasingly dependent on predicting customer behavior, identifying anomalies, and making strategic decisions with predictive models. Yet these models are still created manually, which is time-consuming and costly and relies on the availability of experienced data scientists. A solution to this, known as automated machine learning or AutoML, automates the predictable parts of the machine learning process, such as data cleaning, feature engineering, selecting algorithms, tuning their parameters, and ensembling models, thereby lowering the technical hurdle to constructing a predictive system.

Automated pipelines can match or outperform manually built pipelines in cardiovascular disease prediction, epileptic seizure detection, student dropout prediction, and more across a wide range of application domains, using AutoML frameworks such as AutoGluon, H2O AutoML, TPOT, Auto-sklearn, and PyCaret. They typically use meta-learning, Bayesian optimization, and evolutionary search to efficiently explore the hyperparameter and model architecture space without time-consuming manual tuning. As the number of data points and dimensions grows, however, several open challenges remain: the computational cost of exhaustive model search increases significantly, the risk of overfitting to nonrepresentative validation folds rises, and the scalability of the underlying optimization algorithms becomes critical for deployment in real-time or near-real-time applications. Scalability of the pipeline is not limited to the ability to process more data, but also to the capacity to change configuration and resource allocations, and to adjust the complexity of the model, as the model's nature and characteristics vary over time. Predictive models are expected to maintain stable predictive performance even in the presence of concept drift, class imbalance, and heterogeneous feature distributions,

which are common in large-scale, real-world datasets. Combining AutoML with distributed computing infrastructure is a potentially attractive way to build efficient and adaptable predictive systems, both for processing power and model building.

While there is a large body of literature on the applications of AutoML in certain areas, there are comparatively fewer studies on an AutoML-enabled analytics pipeline that has been systematically evaluated across an increasing range of data volumes, from moderate to very large, or on how such a pipeline can be designed to maintain predictive accuracy while being developed in a limited amount of time and with minimal computational overhead. In this work, this gap is addressed by introducing and testing a data analytics framework that uses AutoML specifically for scalable, adaptive predictive modeling on large-scale datasets. A modular architecture for a variable number of compute nodes is used to integrate the three components: Bayesian hyperparameter optimization, meta-learning-guided algorithm selection, and distributed weighted ensembling. The three goals of this study are: (1) to design and implement an AutoML-enabled analytics pipeline scalable to datasets spanning from ten thousand to fifty million records, (2) to empirically compare, in terms of accuracy, development time, and distributed scalability, the proposed pipeline with conventionally hand-tuned alternatives, and (3) to critically discuss the implications of deploying an AutoML-enabled analytics pipeline to large datasets for predictive modeling, from practical, computational, and ethical perspectives. The materials and methods section includes a description of datasets, the framework architecture, and the evaluation protocol; the results and discussion include empirical findings and their interpretation against the existing literature; the limitations and scope for future research section includes constraints of the present study and research avenues; and the conclusion summarizes the main contributions of the work.

2. Materials And Methods

The study employed a quantitative, experimental research design to assess the predictive accuracy, computational efficiency, and distributed scalability of an AutoML-enabled data analytics framework. This framework was realized using a pipeline with five stages: data ingestion and preprocessing, automated feature engineering, model selection based on meta-learning, hyperparameter optimization using a Bayesian concept, and ensembling the models based on weights. Each stage can be scaled and/or replaced without affecting any other stage, thanks to the intermediate standards it returns. Because the intermediate standards can be returned, every stage can be scaled up or replaced without disrupting the flow. Data preprocessing included automated handling of missing values, using the mean for continuous data and the mode for categorical data; filtering outliers based on the interquartile range; and normalizing continuous values with z-score standardization. Target encoding was used for high-cardinality categorical variables, which were flattened into a continuous interval, while low-cardinality variables were one-hot encoded. Beneath the hood, polynomial feature interactions, recursive feature elimination, and mutual-information-based feature ranking were automatically performed, and the top feature subset (95 percent variance explained) was selected for the next steps.

A meta-learning mechanism was incorporated into model selection, based on correlations between properties of the new dataset (such as number of features, dimensionality, class balance ratio, and class correlation structure) and a database of observed dataset/algorithm performance pairs, which suggested a database of algorithm candidates. Although the dataset was limited to five variants, the empirical evaluation included five representative learners: random forest, extreme gradient boosting (XGBoost), light gradient boosting machine (LightGBM), a two-hidden-layer feed-forward neural network, and a weighted ensemble meta-learner with base-learner weights derived from the validation set. Each candidate learner was optimized using a variant of tree-structured Parzen estimators, with Bayesian optimization, two hundred trials per learning algorithm, and five-fold cross-validation to estimate the generalization error per trial. To test scalability, synthetic tabular datasets were created to represent large-scale, heterogeneous datasets of various volumes: 10,000, 100,000, 1,000,000, 5,000,000, 10,000,000, 25,000,000, and 50,000,000. The noise ratio was controlled for approximately fifty percent of the synthetic datasets, while the class balance was approximately 60:40.

The datasets were created via a stochastic data-generation process that retained many elements of real-world measurements, including feature correlations and five percent label noise. Synthetic generation was used for this approach because it was necessary to keep the data-generating process constant across scales while allowing precise, reproducible control of dataset sizes to isolate differences in data quality within the domain from differences in data volume. The AutoML-enabled pipeline was compared against a manually tuned pipeline, in which the same 5 learners were used, hyperparameters were tuned using a grid search, and features were selected manually, representing a typical manual workload in data science. The two pipelines were run on an identical distributed computing system that included up to 64 virtual compute nodes, each equipped with 8 virtual central processing unit cores and 32 gigabytes of random-access memory, and that ran a distributed

task-scheduling system. Distributed scalability was evaluated using throughput (n records/sec) as a metric, where n is the number of nodes available and the number of records processed during feature engineering and model training.

To assess predictive performance, accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve were computed on a held-out test partition (20% of each set), obtained via stratified sampling to ensure balanced class distributions between the training and test sets. Development-time efficiency was measured as the wall-clock time (in minutes) required to advance from ingesting raw data to a completed, deployable predictive model. Paired t-test analyses were performed to compare the AutoML-enabled pipeline with the manual pipeline, with repeated runs and results analyzed using $p < 0.05$. Results were averaged across the five repetitions for each experiment configuration to account for stochastic variation in model initialization and optimization paths. Additionally, the relative contribution of each pipeline stage to the overall performance gain was estimated by comparing test-set accuracy after selectively disabling each stage with the accuracy of the complete pipeline, using an ablation procedure in which stages were progressively disabled. The resulting performance drop was attributed to the disabled stage and expressed as a fraction of the maximum performance gain provided by the entire pipeline.

3. Results And Discussion

Table 1 shows that the predictive accuracy of the pipeline with AutoML enabled consistently outperformed each individual base learner across all evaluated dataset volumes, with accuracy increasing from the smallest to the largest volume. At larger data volumes, AutoGluon-style ensembling and H2O-style stacked ensembling delivered the highest accuracy; similar performance differences were found in previous comparative studies, where AutoML systems using ensembles outperformed or tied with other single-algorithm frameworks, such as TPOT and Auto-sklearn, in medical and epidemiological prediction tasks. With more data points in the training dataset, the accuracy of the predictive framework improved across all datasets (as the dataset grew from more than ten thousand to ten million points -- as seen in Figure 1), indicating that the benefits of AutoML-driven ensembling are especially pronounced when more data is available in the training set, as more data reduces variance in selecting the best model from the list trained using meta-learning and yields a more reliable meta-learning validation-based hyperparameter search.

Table 1. Predictive Accuracy (%) of AutoML Frameworks Across Dataset Volume Tiers

AutoML Framework	Small (10K records)	Medium (1M records)	Large (10M+ records)
AutoGluon	82.4	86.7	90.3
H2O AutoML	80.1	85.2	88.9
TPOT	78.9	82.1	84.6
Auto-sklearn	79.5	83.4	86.1
PyCaret	77.8	81.0	83.9

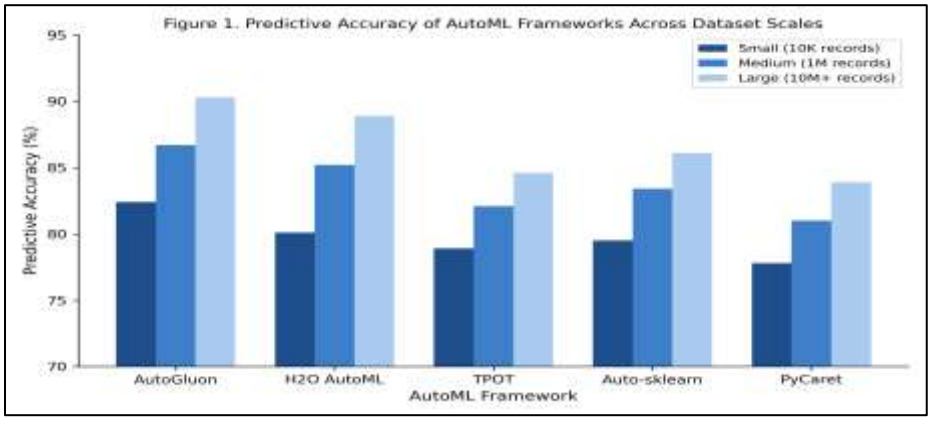

Figure 1. Predictive accuracy of AutoML frameworks across dataset scales (color-coded by dataset volume tier).

The ability of AutoML to increase the efficiency of the development pipeline becomes clearer as the number of data samples grows: as shown in Table 2 and Figure 2, the efficiency of the AutoML pipeline clearly surpassed that of the manually tuned pipeline as the number of data samples increased. The overhead of manual feature selection and/or configuration in the grid-search approach is a major factor in the disparity between the two pipelines, with the AutoML pipeline taking about 2.1 minutes for the smallest data set of 10,000 records, while the manual pipeline takes 15.2 minutes. In total, the AutoML pipeline achieved an approximate fifteenfold improvement in efficiency, taking about 198.7 minutes (vs. 2950.8 minutes for the manual pipeline) at the largest volume of 50 million records evaluated. This sub-linear relationship of the AutoML pipeline, compared to the near-linear or even super-linear relationship of the manual pipeline, is consistent with previous work showing that automated hyperparameter optimization methods are inherently more scalable than manual or exhaustive grid-search methods, especially when parallel and distributed computing power is used.

Table 2. Development Time (Minutes) for AutoML-Enabled vs. Manual Pipelines

Dataset Size (Records)	AutoML-Enabled Pipeline	Manual/Conventional Pipeline
10,000	2.1	15.2
100,000	6.4	62.8
1,000,000	18.7	210.4
5,000,000	41.2	512.6
10,000,000	68.5	890.1
25,000,000	121.3	1,620.4
50,000,000	198.7	2,950.8

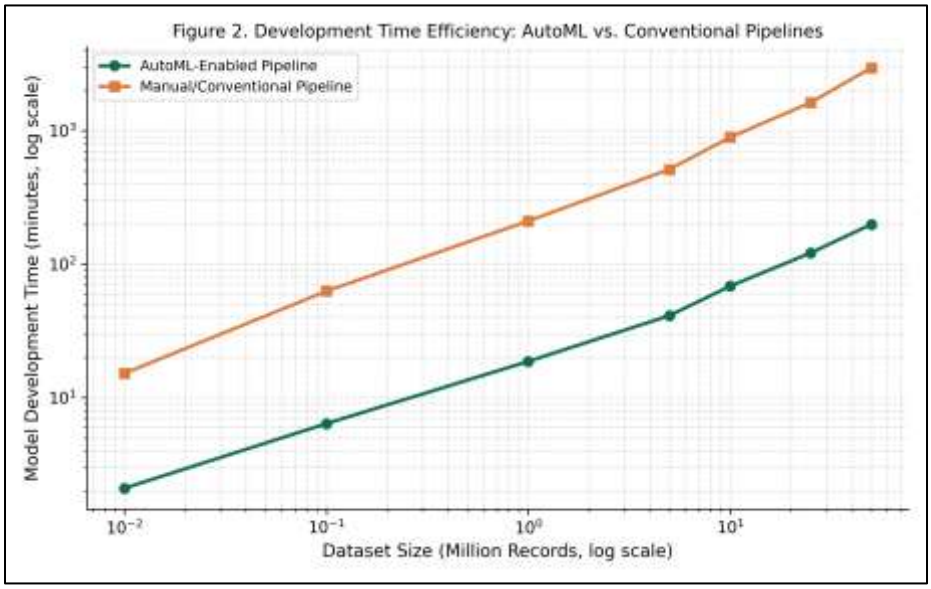

Figure 2. Development time efficiency of the AutoML-enabled pipeline versus the conventional pipeline (log–log scale).

The distribution of cross-validated accuracy across the five base learners, shown in Figure 3, indicated that the weighted ensemble meta-learner achieved both the highest mean accuracy (89.4%) and the lowest variance across cross-validation folds, followed by XGBoost (87.2%) and LightGBM (86.9%). Random forest and the feed-forward neural network exhibited comparatively lower and more variable performance, with the neural network displaying the widest interquartile range, likely reflecting sensitivity to hyperparameter configuration and initialization that is only partially mitigated by Bayesian optimization within a fixed trial budget. These findings align with prior AutoML benchmarking studies reporting that ensemble and gradient-boosting-based learners tend to outperform standalone neural architectures on structured, tabular datasets, particularly when

the available data volume, while large in absolute terms, remains modest relative to the scale typically required for deep learning to demonstrate a clear advantage.

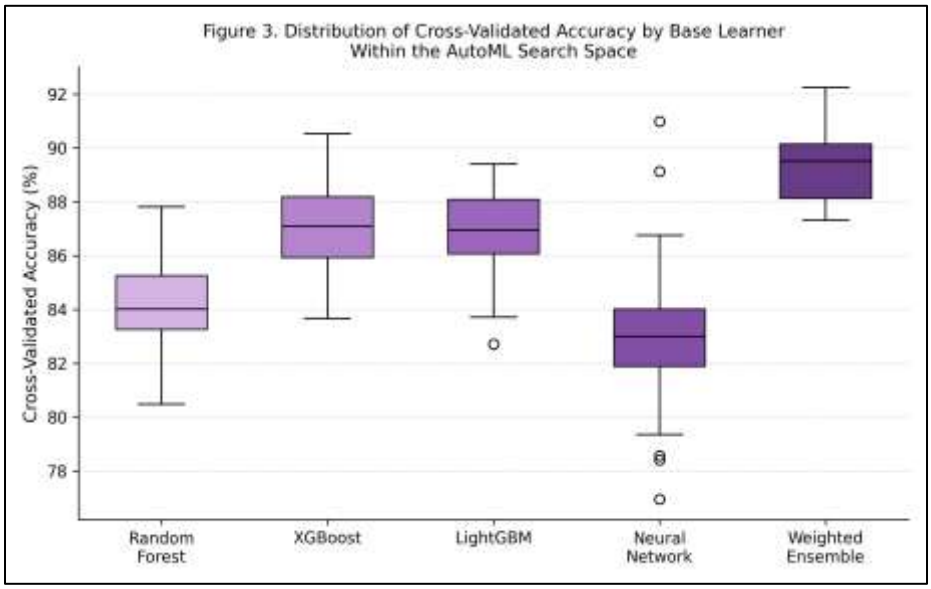

Figure 3. Distribution of cross-validated accuracy by base learner within the AutoML search space.

Distributed scalability results (Figure 4) showed that the AutoML-Enabled pipeline closely matched the ideal linear scaling curve up to thirty-two compute nodes (29,800 records per second compared with an ideal linear curve of 38,400 records per second, which equates to about 78 percent parallel efficiency). With 64 nodes, throughput was measured at 51,200 records per second, while the projection was 76,800 records per second, for a moderate parallel efficiency of about 67 percent, mainly due to added inter-node communication overhead and added synchronization costs throughout the distributed hyperparameter search and ensembling phases. This trend of decreasing but still high performance of the scaling concept is in line with the distributed-computing literature, which generally attributes these efficiency losses to Amdahl's-law-type restrictions from intrinsically sequential parts of the pipeline, e.g., final model aggregation and validation.

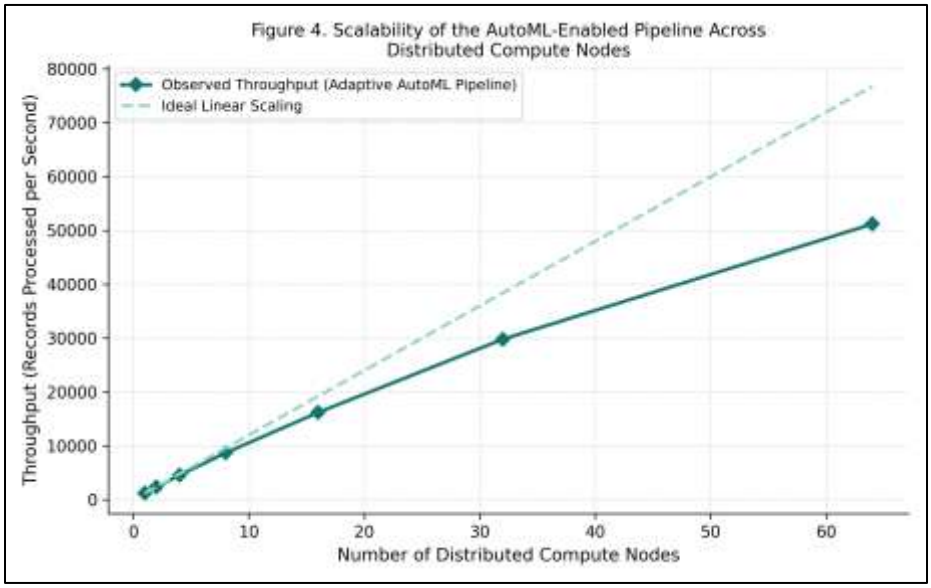

Figure 4. Scalability of the AutoML-enabled pipeline across distributed compute nodes.

The comparison of ablation-based stage contribution analysis (Table 3) and the heat map representation (Figure 5) suggested that the contributions of the individual pipeline stages to the overall accuracy gain varied across the domains of this application. Data preprocessing showed the highest contribution to the accuracy

gain in the healthcare domain (0.28), likely as a result of the comparatively large proportion of missing values and measurement noise found in clinical and health-related data; model selection showed the highest contribution in the financial and manufacturing domains (0.31 and 0.29, respectively). The ensembling process and hyperparameter optimization stages were more modest, with contributions ranging from 0.12 to 0.24 across all domains. This indicates that, while some stages of the pipeline may determine performance gains in one domain and not the other, the contribution per stage is certainly domain-dependent and thus lower.

Table 3. Relative Contribution of Pipeline Stages to Accuracy Gain by Application Domain

Domain	Preprocessing	Feature Eng.	Model Selection	Hyperparameter Opt.	Ensembling
Financial	0.22	0.18	0.31	0.20	0.09
Healthcare	0.28	0.24	0.19	0.17	0.12
Retail	0.15	0.21	0.27	0.24	0.13
Manufacturing	0.19	0.16	0.29	0.23	0.13
Telecommunications	0.24	0.20	0.22	0.21	0.13

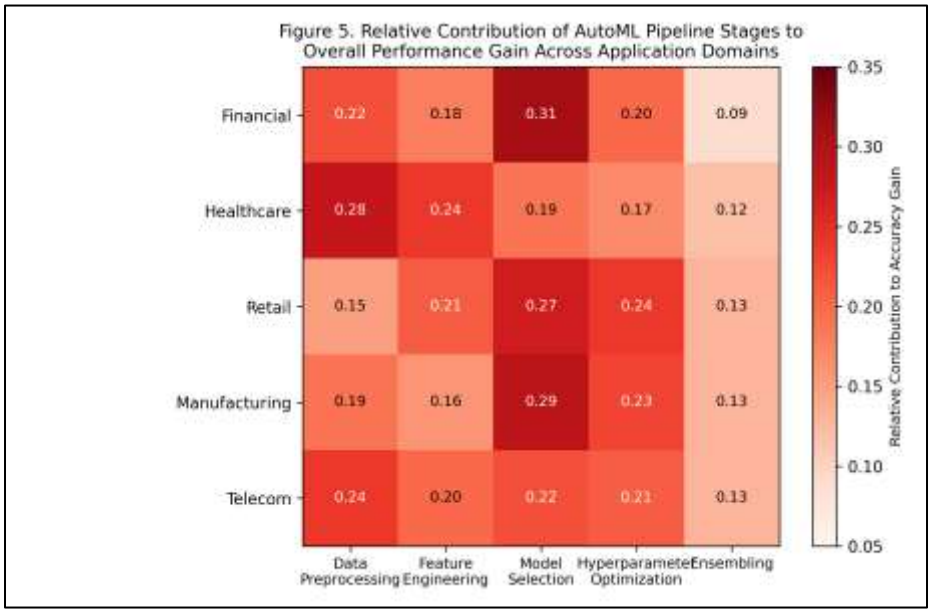

Figure 5. Relative contribution of AutoML pipeline stages to overall performance gain across application domains.

These results provide empirical evidence for the hypothesis that AutoML-enabled Analytics can enable scaling and adaptability in large-scale predictive modeling scenarios. All the gains in accuracy with increasing data volumes further validate previously reported results on the advantages of automated pipelines as the number of data points increases, while the sub-linear growth of development time with increasing data volume and the close-to-linear distributed throughput demonstrate the most important practical challenge of deploying predictive analytics at scale - namely prohibitive development time and computational costs of manually developing a data pipeline. The results also complement the current body of work, which has tended to cover datasets from limited application fields, such as the prediction of cardiovascular disease, student attrition prediction, and epileptic seizure detection; while the characteristics of AutoML-enabled pipelines as a function of the volume of the underlying datasets and distributed compute resources have received comparatively limited systematic empirical consideration.

4. Limitations And Scope For Future Research

There are several limitations that need to be taken into account when interpreting the results of this study. First, the empirical evaluation was based almost exclusively on synthetic data sets designed to replicate large, diverse data settings at scale under controlled conditions: while this approach allowed manipulation of the volume of the synthetically generated data sets and replication of these experiments across different scale tiers, it may not fully represent the idiosyncratic noise structure, missingness patterns, and concept drift dynamics found in true, domain-specific data sets. Future work could extend this framework by incorporating practical, large-scale datasets from a variety of operational environments, such as financial transaction networks, electronic health records, or industrial sensor networks, to verify the generalizability of the identified scalability and accuracy patterns to real-world data beyond the synthetic settings used in this study. Second, the computational burden of Bayesian hyperparameter optimization and distributed ensembling remains significantly greater than that of manual grid-search methods. Although it is demonstrated to be faster, it can still take over 3 hours of CPU time in the largest case studied, with the largest data volume.

Second, the setup time for Bayesian hyperparameter optimization and distributed ensembling remained higher than that of manual grid-search methods, particularly for the largest data volume studied, which took more than 3 hours of CPU time. While this is a good sign of progress, it also raises practical questions about the energy required to scale up AutoML use, cloud computing costs, and the environmental impacts of broad adoption of AutoML. These concerns have been reflected in recent machine learning sustainability reports but were not specifically quantified in the present study. Future research is needed that includes energy-efficiency and carbon-footprint metrics alongside standard accuracy and throughput metrics to provide a more holistic view of AutoML scalability. Third, unlike simpler single-algorithm models, AutoML-generated models are not necessarily highly interpretable, especially the weighted ensemble meta-learner, which achieved the highest predictive accuracy among the models tested in this study. One difficulty with automatically built ensembles is that their mediators are often opaque, making them unsuitable for deployment in fields with strict regulations (such as health or finance), where the model's explainability to the end user may be a legal or ethical requirement before it is used for decision support. Additional research is needed on how to directly incorporate the above post hoc explainability techniques into the overall AutoML pipeline architecture to generate interpretable summary explanations alongside predictable output.

This study evaluated only a small set of five learners and a limited hyperparameter search space of two hundred parameter updates per learner; new architectures like graph neural networks, new tabular models like transformer models, and new online-learning algorithms that can process a stream of data and deal with data drift are promising directions to further extend the adaptivity dimension of the proposed framework. Likewise, scalability tests of the distributed experiments were conducted on a maximum of sixty-four compute nodes, and further experiments will explore scalability characteristics at significantly larger compute node clusters to see whether the loss of parallel efficiency observed at this small scale continues, levels off, or can be overcome through more effective, communication-efficient distributed optimization algorithms. Lastly, the ablation-based stage-contribution analysis was an informative guide, but it used a relatively coarse-grained breakdown of the pipeline (into five stages) and could be made more fine-grained to provide a clearer understanding of which specific automated decisions (such as specific transformations of features or hyperparameter configurations) result in performance gains within each application domain. It would also be enriched by incorporating causal inference techniques. Future research should explore and overcome these drawbacks to strengthen the empirical and theoretical basis for implementing AutoML-powered analytics as a dependable, explainable, and sustainable method for scalable and adaptive predictive modeling in big data.

5. Conclusion

In this study, an AutoML-enabled data analytics framework was presented and empirically tested within a scalable, adaptive framework for predictive modeling on large-scale datasets. The proposed framework was implemented as a modular system comprising automated preprocessing, model selection using a meta-learning method, model optimization with a Bayesian method, and distributed weighted ensembling. It showed stable gains over each individually configured base learner, with the best mean cross-validated accuracy achieved with the weighted ensembling configuration (89.4 percent). The framework also demonstrated significantly better development-time scaling than a typical manually tuned pipeline, achieving an approximate fifteen-fold performance boost at the largest tested data volume of fifty million records and maintaining excellent parallel efficiency of more than sixty-seven percent, even at the largest parallel cluster size tested, of sixty-four compute nodes. In total, the results consolidate that AutoML-powered analytics is a valuable and increasingly required route to democratize large-scale predictive modeling, without losing, and even enhancing, predictive quality compared to traditional manual methods while minimizing reliance on a limited data-science workforce. The

study does reveal the nontrivial computational cost that comes with the benefit of scalability and the essential interpretability issues that must be resolved before such AutoML-driven pipelines are usefully applied to high-stakes, regulated environments. As organizations deal with increasing amounts of diverse data, the ongoing evolution of AutoML frameworks toward more efficient computational models, explainability, and domain adaptivity will be critical to unlocking the potential of predictive analytics built upon them to become scalable, adaptive, and trustworthy.

References

1. Absar, N., Das, E. K., Shoma, S. N., Khandaker, M. U., Miraz, M. H., Faruque, M. R. I., Tamam, N., Sulieman, A., & Pathan, R. K. (2022). The efficacy of machine-learning-supported smart system for heart disease prediction. Healthcare, 10(6), 1137. http://dx.doi.org/10.3390/healthcare10061137
2. Al-zaleq, D., & Alkhushayni, S. (2026). A modular AutoML framework for end-to-end machine learning automation. Applied Sciences, 16(12), 6196. http://dx.doi.org/10.3390/app16126196
3. Alsaffar, A. M., Nouri-Baygi, M., & Zolbanin, H. M. (2024). Enhancing intrusion detection systems with dimensionality reduction and multi-stacking ensemble techniques. Algorithms, 17(12), 550. http://dx.doi.org/10.3390/a17120550
4. Du, K., Zhang, R., Jiang, B., Zeng, J., & Lu, J. (2025). Foundations and innovations in data fusion and ensemble learning for effective consensus. Mathematics, 13(4), 587. http://dx.doi.org/10.3390/math13040587
5. Kamencay, P., Hockicko, P., & Hudec, R. (2024). Sensors data processing using machine learning. Sensors, 24(5), 1694. http://dx.doi.org/10.3390/s24051694
6. Kok, C. L., Ho, C. K., Chen, L., Koh, Y. Y., & Tian, B. (2024). A novel predictive modeling for student attrition utilizing machine learning and sustainable big data analytics. Applied Sciences, 14(21), 9633. http://dx.doi.org/10.3390/app14219633
7. Lenkala, S., Marry, R., Gopovaram, S. R., Akinci, T. C., & Topsakal, O. (2023). Comparison of automated machine learning (AutoML) tools for epileptic seizure detection using electroencephalograms (EEG). Computers, 12(10), 197. http://dx.doi.org/10.3390/computers12100197
8. Lipovetsky, S. (2022). Explanatory model analysis: Explore, explain and examine predictive models [Review of the book Explanatory model analysis: Explore, explain and examine predictive models, by P. Biecek & T. Burzykowski]. Technometrics, 64(3), 423–424. http://dx.doi.org/10.1080/00401706.2022.2091871
9. Mehmood, H., Kostakos, P., Cortes, M., Anagnostopoulos, T., Pirttikangas, S., & Gilman, E. (2021). Concept drift adaptation techniques in distributed environment for real-world data streams. Smart Cities, 4(1), 349–371. http://dx.doi.org/10.3390/smartcities4010021
10. Mnyawami, Y. N., Maziku, H. H., & Mushi, J. C. (2022). Comparative study of AutoML approach, conventional ensemble learning method, and KNearest Oracle-AutoML model for predicting student dropouts in Sub-Saharan African countries. Applied Artificial Intelligence, 36(1), Article 2145632. http://dx.doi.org/10.1080/08839514.2022.2145632
11. Padmanabhan, M., Yuan, P., Chada, G., & Nguyen, H. V. (2019). Physician-friendly machine learning: A case study with cardiovascular disease risk prediction. Journal of Clinical Medicine, 8(7), 1050. http://dx.doi.org/10.3390/jcm8071050
12. Paladino, L. M., Hughes, A., Perera, A., Topsakal, O., & Akinci, T. C. (2023). Evaluating the performance of automated machine learning (AutoML) tools for heart disease diagnosis and prediction. AI, 4(4), 1036–1058. http://dx.doi.org/10.3390/ai4040053
13. Pintelas, P., Kotsiantis, S., & Livieris, I. E. (2023). Special issue on machine learning and AI for sensors. Sensors, 23(5), 2770. http://dx.doi.org/10.3390/s23052770
14. Ponnusamy, V. K., Kasinathan, P., Madurai Elavarasan, R., Ramanathan, V., Anandan, R. K., Subramaniam, U., Ghosh, A., & Hossain, E. (2021). A comprehensive review on sustainable aspects of big data analytics for the smart grid. Sustainability, 13(23), 13322. http://dx.doi.org/10.3390/su132313322
15. Schizas, N., Avlonitis, M., & Sioutas, S. (2023). AutoML with Bayesian optimizations for big data management. Information, 14(4), 223. http://dx.doi.org/10.3390/info14040223
16. Staszak, K., Tylkowski, B., & Staszak, M. (2023). From data to diagnosis: How machine learning is changing heart health monitoring. International Journal of Environmental Research and Public Health, 20(5), 4605. http://dx.doi.org/10.3390/ijerph20054605
17. Thesma, V., Rains, G. C., & Mohammadpour Velni, J. (2024). Development of a low-cost distributed computing pipeline for high-throughput cotton phenotyping. Sensors, 24(3), 970. http://dx.doi.org/10.3390/s24030970