Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777399

THEORETICAL ASSESSMENT OF AUTO-ML TECHNIQUES: BENEFITS, CONSTRAINTS, AND PROSPECTS FOR FUTURE RESEARCH

[1]Dhruvi Gosai, [2]Dr. Minal Patel
[1,2]Devang Patel Institute of Advance Technology and Research (DEPSTAR)
CHARUSAT, Changa, India

Abstract
Auto Machine Learning (Auto-ML) has emerged as a promising solution to automate the traditional machine learning workflow. Auto-ML aims to reduce the manual intervention required in designing, training, and deploying machine learning models. This article theoretically assesses the existing Auto-ML methodologies, their benefits, limitations, and future research directions. We investigate the advantages of Auto-ML in terms of reducing human effort, increasing model accuracy, and democratizing machine learning.However, we also discuss the constraints such as the limited interpretability of Auto-ML models and the potential risk of over-reliance on automated techniques. Furthermore, we highlight the research gaps in Auto-ML, including the need for explainable Auto-ML models, personalized Auto-ML, and more efficient hyper-parameter optimization algorithms. Overall, this article provides a comprehensive review of Auto-ML techniques and serves as a roadmap for future research in this area.
Keywords: Auto-ML, machine learning, benefits, limitations, future research, accuracy, interpretability, democratizing, human effort, over-reliance, explainable, personalized, hyper-parameter optimization.

Introduction

Automated Machine Learning (Auto-ML) has grown in popularity as a way to automate the traditional machine learning workflow.Auto-ML aims to reduce the manual intervention required in designing, training, and deploying machine learning models. With the growing demand for machine learning in various industries, Auto-ML has emerged as a promising solution to democratize machine learning by enabling individuals with limited expertise to build accurate models. However, while Auto-ML holds great promise, it also has its limitations, which must be addressed to improve its effectiveness.

In this article, we provide a theoretical assessment of existing Auto-ML methodologies, their benefits, limitations, and future research directions. We investigate the advantages of Auto-ML in terms of reducing human effort, increasing model accuracy, and democratizing machine learning. We also discuss the potential drawbacks of Auto-ML, including the limited interpretability of Auto-ML models and the potential risk of over-reliance on automated techniques.

Furthermore, we highlight the research gaps in Auto-ML, including the need for explainable Auto-ML models, personalized Auto-ML, and more efficient hyper-parameter optimization algorithms. Explainable Auto-ML models are critical for increasing transparency and building

trust in the models developed by Auto-ML. Personalized Auto-ML, on the other hand, can tailor models to individual user requirements, while more efficient hyper-parameter optimization algorithms can reduce the computational cost of Auto-ML.

Overall, this article provides a comprehensive review of Auto-ML techniques and serves as a roadmap for future research in this area. The article emphasizes the importance of developing Auto-ML models that are accurate, transparent, and cost-effective. By addressing the limitations and gaps in the existing Auto-ML methodologies, we hope to improve the effectiveness of Auto-ML in democratizing machine learning and making it more easily available to more audiences.

Related Works

Automated Machine Learning (Auto-ML) is an emerging field that aims to automate the machine learning (ML) process. The goal of Auto-ML systems is to automate and optimize the machine learning process, from data pre-processing to model selection and tuning. In recent years, many researchers have developed Auto-ML systems that objective is to automate various stages of the ML methods. This literature study reviews some of the recent research articles on Auto-ML.

T. Nagarajahet al.provide a comprehensive review of automated machine learning (AutoML) systems, including their architecture, workflow, and limitations, as well as recent advances and future directions. They also discuss the role of AutoML in facilitating the democratization of machine learning, reducing human intervention, and improving efficiency. The review concludes that AutoML has enormous potential in various applications, including healthcare, finance, and e-commerce.

K. Chauhanet al.presents an overview of automated machine learning (AutoML) and its potential impact on the field of machine learning. The article provides a detailed description of AutoML, including its architecture, algorithms, and applications, as well as challenges and opportunities. The authors argue that AutoML represents a new wave of machine learning that can vastly enhance the efficiency and effectiveness of machine learning.They also discuss the potential of AutoML in democratizing machine learning and enabling non-experts to use machine learning techniques.

T. Dhaeneet al. propose a new approach for feature and model type selection in automated machine learning (AutoML) using multi-objective optimization. The proposed method aims to simultaneously optimize multiple objectives, such as accuracy, efficiency, and interpretability. The article presents the results of experiments performed on a variety of datasets, demonstrating that the proposed approach outperforms several existing AutoML methods in terms of accuracy and efficiency. The authors conclude that the proposed approach has a high potential for improving AutoML performance and interpretability.

Y. L.-I. J. of A. C.et al.analyse the trends in automated machine learning (AutoML) and provide a comprehensive overview of the field. They describe the challenges and opportunities in AutoML and discuss various techniques used in AutoML, including hyper-parameter optimization, feature engineering, and algorithm selection. The authors also explore the applications of AutoML in different domains, such as healthcare, finance, and transportation, and discuss the ethical and legal issues associated with AutoML. They conclude by outlining future directions in AutoML research.

M. Hanusseket al.present an AutoML benchmark study in which the authors evaluated whether AutoML systems could outperform humans in terms of accuracy and efficiency. The study found that AutoML systems outperformed humans in terms of efficiency, but not accuracy. However, the authors argue that the margin between the two is small and that AutoML systems have the potential to surpass human performance in the future. The study also highlights the need for standardized evaluation frameworks for AutoML systems.

P. Gijsberset al.propose an open-source benchmark for automated machine learning (AutoML) systems, addressing the lack of standardized evaluation metrics and datasets in the field. The article was published in 2019 and provides detailed information on the dataset used, evaluation criteria, and results of various AutoML systems tested on the benchmark.

M. Feureret al.propose a new automated machine learning (AutoML) approach that is efficient and robust. The authors address the challenges of AutoML, including high computational cost and the need for a diverse set of models, by proposing a novel evolutionary algorithm. They test their approach on various datasets and demonstrate its effectiveness and efficiency.

J. Drozdalet al.explore the role of belief in automated machine learning (AutoML) methods and the information need necessary to establish trust. The authors conducted a user study to investigate the factors that influence trust in AutoML and identify the types of information users need to establish trust. They find that transparency, interpretability, and user control are important factors for building trust in AutoML.

H. J. Escalanteet al.createa survey of the automated machine learning (AutoML) field. The authors discuss the advantages of AutoML, including its ability to reduce the time and expertise needed to build accurate models. They also highlight some of the challenges and limitations of AutoML, such as the difficulty of handling certain types of data and the need for careful consideration of model interpretability. The authors conclude with a discussion of some of the most promising directions for future research in AutoML.

B. Wang et al. present VEGA which is created AutoML pipeline with an end-to-end configuration. The authors propose a modular framework that allows users to easily build customized AutoML workflows by selecting and configuring individual components. VEGA includes several novel features, such as a tree-based search algorithm for hyper-parameter

optimization and a strategy for handling missing values in the data. The authors demonstrate the effectiveness of their approach on several real-world datasets.

Y.-W. Chen et al. provide an overview of various techniques used in AutoML. The authors discuss different types of AutoML approaches, such as model selection and hyper-parameter tuning, and describe the advantages and disadvantages of each. They additionally offer a comprehensive review of the AutoML tools and platforms, along with their key features and limitations.The authors conclude with a discussion of some of the most promising directions for future research in AutoML.

A.-I. Imbreaet al. focuses on AutoML techniques for data streams, which pose unique challenges due to their high volume and velocity. The authors review various approaches to AutoML for data streams, including online and incremental learning methods. They also discuss the importance of model interpretability in the context of data streams, where decisions must often be made in real time. The authors conclude with a discussion of some of the most promising directions for future research in AutoML for data streams.

S. Brändle et al. build AutoML tools for text classification that can evaluate different representation models. The authors compare several state-of-the-art representation models, including bag-of-words and neural network-based approaches, on a variety of text classification tasks. They also assess the performance of several popular AutoML tools, including AutoKeras and TPOT, on these tasks. The authors conclude with a discussion of the advantages and disadvantages of various representation models and also developed AutoML tools for text classification.

X. Shi et al. present a multimodal AutoML benchmarking study for tabular data with text fields. The authors evaluate the efficiency and performance of several The authors evaluate the efficiency of several cutting-edge AutoML tools on a dataset that includes both tabular and text data.They also evaluate the effectiveness of different feature engineering techniques, such as text embedding and dimensionality reduction, on the performance of AutoML models. The authors conclude with a discussion of the strengths and limitations of different AutoML tools and feature engineering techniques for multimodal data.

M. Weveret al. provides an overview and empirical evaluation of AutoML for multi-label classification. The authors discuss various approaches to multi-label classification, including problem transformation and algorithm adaptation methods, and describe the advantages and disadvantages of each. They also review several AutoML tools and platforms for multi-label classification and evaluate their performance on several benchmark datasets. The authors conclude with a discussion of some of the most promising directions for future research in AutoML for multi-label classification.

K. T. Y. Mahimaet al. evaluated sentiment analysis usingAutoML and traditional approaches. They compared the performance of two AutoML frameworks, Auto-Keras and TPOT, with

traditional machine learning techniques like Naïve Bayes (NB), Support Vector Machine (SVM), and Random Forest (RF) on a sentiment analysis task. They used Twitter data for the experiments and found that Auto-Keras outperformed TPOT and traditional approaches on most of the evaluation metrics.

V. Lopeset al. proposed an AutoML-based technique that can evaluate sentiment analysis from the multimodal image. To learn the optimal deep neural network architecture for image classification, they used a hybrid AutoML framework that combines Auto-Sklearn and Neural Architecture Search (NAS).They evaluated their approach on a benchmark dataset and discovered that the proposed method outperformed several state-of-the-art methods in terms of accuracy.

L. Vaccaroet al. conducted an empirical review of automated machine learning. They analyzed 47 AutoML frameworks based on a set of criteria like the search space, optimization algorithm, and validation strategy. They found that most of the frameworks use variations of Bayesian optimization and gradient-based optimization algorithms to search the space of models and hyper-parameters. They also highlighted the need for standardization in benchmark datasets, evaluation metrics, and experimental setups to enable fair comparisons between AutoML frameworks.

X. He et al. provided an in-depth survey of AutoML.They reviewed recent developments in the field and classified AutoML methods into four categories: automated model selection, automated feature engineering, automated hyper-parameter tuning, and automated architecture design. They also discussed AutoML's difficulties and possibilities in research directions.

S. K. K. Santudiscussed the challenges and opportunities in AutoML. They highlighted the potential benefits of AutoML, like reducing the time and cost of developing machine learning models, enabling the use of machine learning for non-experts, and accelerating the research progress of machine learning. They also discussed the challenges faced by AutoML, such as the need for standardization, reproducibility, and interpretability of AutoML models.

X. Zheng et al. proposed a novel method for developing fully automated machine learning through life-long knowledge anchors.They used a genetic programming framework to evolve a pipeline of machine learning models that can be updated continuously as new data becomes available. They evaluated their approach on several benchmark datasets and showed that it can outperform AutoML frameworks in terms of predictive performance and computational productivity and efficiency.

Literature Survey

Sr. No.	Title	Publication Details	Methods	Key Components	Advantages	Future Work/Research Gap
1.	Evolving Fully Automated Machine Learning via Life-Long Knowledge Anchors	IEEE 2021	Auto-Weka, P4ML, Auto-Sklearn, TPOT	Data Pre-processing, Feature Engineering, Model Selection, Model Optimization, and Ensemble	Maximize the model accuracy without consuming human efforts in selecting and implementing the aforementioned pipeline components	Try to reduce the computational cost
2.	AutoML for Multi-Label Classification: Overview and Empirical Evaluation	IEEE 2021	Auto-WEKA, auto-sklearn, hyperopt-sklearn,	Single-Label Classification (SLC), Multi-Label Classification (MLC)	Find a grammar-based best-first search to compare favourably to other optimizers	More work on extremely large search space and the deep hierarchical configuration structures of multi-label classifiers
3.	An AutoML-based Approach to Multimodal Image Sentiment Analysis	IEEE 2021	Perform multimodal classification, in which a method leverages more than one type of data	Pre-processing, Individual classifications and fusion stage	For Image, sentiment analysis used ResNet and DenseNet models and for the Text sentiment analysis	Try to increase high accuracy and work on more datasets

			(text, images) to perform classification		used VADER, TextBlob, FastText, LSTM, LSTM-Attn, Bi-LSTM, RNN, RCNN, TextCNN, VDCNN. Achieved 95.19% accuracy	
4.	Evaluation of Sentiment Analysis based on AutoML and Traditional Approaches	International Journal of Advanced Computer Science and Applications (IJACSA)-2021	HyperOptSkLearn, TPot, Scikit learn, Keras, Auto-Keras	Binary Classification, Multi-Class Classification	Evaluate the sentiment analysis based on AutoML and Traditional approaches. The implementation of the evaluation was done by using both machine learning and deep learning for multi-class sentiment analysis and binary sentiment analysis	Evaluate the cloud-based AutoML methods such as Google AutoML. Evaluate other AutoML libraries such as H2O.ai
5.	An Empirical Review of	Computers (2021)	AutoKeras	Neural Architectur	Highlight the strengths	Build a unique and unambiguo

	Automated Machine Learning			e Search (NAS), Reinforcement Learning (RL)	and weaknesses of inference methods to apply them to AutoML Empirical evaluation of the tested solutions are expressed in terms of Interpretability, Efficiency, Structural Invariance, Scale Invariance and Scalability	us artificial intelligence system
6.	Benchmarking Multimodal AutoML for Tabular Data with Text Fields	35th Conference on Neural Information Processing Systems (NeurIPS2021)	AutoGluon, H2O	Natural language processing (NLP)	N-Grams and word2vec provide superior text featurization than Pre-Embedding. Build multimodal networks for classification /regression	Investigate different data pre-processing pipelines. More work on more data types
7.	Evaluation of Representation	Proceedings of the Future Technologies Conference,	Auto-Sklearn, H2O AutoML, AutoKeras	Datasets, Text Pre-processing,	Investigate AutoML performance on text	Increase both the amount of considered

	Models for Text Classification with AutoML Tools	Springer, Cham, 2021		Text Representation, AutoML Tools, Evaluation	classification tasks. Simple text embedding such as Bag-of-Words performs best and is able to outperform more recent text embedding	datasets as well as the dataset's sample sizes More work on other open-source AutoML tools
8.	AutoML to Date and Beyond: Challenges and Opportunities	ACM Computing Surveys(CSUR) (2021)	Autokeras, Cloud AutoML, TPOT, IBM AutoML, Auto-Sklearn	Data Visualization, Cleaning, Feature Engineering, Hyper-parameter Tuning, Models Exploration, Evaluation	Achieved AutoML goals like particular automated task formulation, effective prediction engineering, and the recommendation of useful tasks	The AutoML tool will increase the productivity of data scientists and domain experts. Also, try to increase Human-Computer interaction
9.	Automated Machine Learning Techniques for Data Streams	arXiv preprint arXiv:2106.07317 (2021)	AutoWeka, H2O.ai, TPOT, auto-sklearn	Online Learning, Concept drift, Meta-Learning	It proposed a scalable and portable benchmarking framework based on Apache Kafka streams and Docker containers	Work on end-to-end AutoML framework in the industry, auto ml-streams may be extended with Auto Cleaning, AutoFE, and HPO components

10.	Techniques for Automated Machine Learning	ACM SIGKDD Explorations 2021	Hyperopt, Scikit-Optimize, Auto-Sklearn, TPOT, H2O, Automatic Model Tuning, Azure AutoML, Auto-Keras, AdaNet, Auto-PyTorch, Google AutoML, Neural Network Intelligence	Automated Feature Engineering (AutoFE), Automated Model and Hyper-parameter Tuning (AutoMHT), Automated Deep Learning (DL)	The system focuses on solving bi-level optimization problems. Give answerstothe following questions: (1) what category do they want to deal with, e.g. AutoFE, AutoMHT, or AutoDL (2) what search scope of the category is e.g. the range of hyper-parameters, and (3) What techniques is appropriate to perform the search	Efficiency, The design of search spaces
11.	Automated Machine Learning: The New Wave of Machine Learning	IEEE 2020	H2O-AutoML, Cloud AutoML, TPOT	Auto Pre-processing, Auto Feature Engineering, Model Selection, Auto Hyper-	Discuss various segments of AutoML with a conceptual perspective	Work on a generalized AutoML pipeline, which can accept a wide range of datasets, and establisha

				parameter tuning, Model training		central meta-learning framework that acts as a central brain for approximating the pipelines for all future problems statements
12.	Automated Machine Learning - a brief review at the end of the early years	arXiv 2020	Auto-WEKA, Auto-SkLearn, TPOT, Neural Architecture Search	Algorithm selection, Hyper-parameter optimization, Meta-learning, Full-model selection, Combined Algorithm Selection, and Hyper-parameter optimization (CASH), Neural architecture search	Categorized AutoML methodologies based on their The first wave (2006-2010), The second wave (2011-2016), The third wave (2017 and on)	AutoML for non-tabular data, Large scale AutoML
13.	Trust in AutoML: Exploring Information Needs for Establishing Trust in Automated Machine	25th International Conference on Intelligent User Interfaces, 2020	Auto-sklearn, TPOT	Data Acquisition, Data Cleaning & Labelling, Feature Engineering,	Explore trust in the relationship between human data scientists and AutoML systems	Maintaining trust between human data scientists and AutoML systems is

	Learning Systems			Model Selection, Hyper-parameter Optimization, Ensembling, Model validation, Model Deployment, Runtime Monitoring, Model Improvement		very difficult
14.	A Review on Automated Machine Learning (AutoML) Systems	IEEE 2019	Auto-WEKA Hyperopt-Sklearn, TPOT, Auto-Compete, PennAI	Pre-processing Engine, Feature Engine, Predictor Engine, Model Selection, and Ensemble Engine	By using assembling and meta-learning the problem of automated hyper-parameter tuning can be tackled efficiently	Functional end products, Accessible knowledge hub, Python-centric research Using Neural Networks
15.	Feature and Model Type Selection using Multi-Objective Optimization for AutoML	25th Belgian-Dutch Conference on Machine Learning (Benelearn), 2016	Multi-Objective method	Data Pre-processing, Feature Selection, Model type Selection, Result Analysis	The method correctly identified the appropriate features, and several model types competed to obtain	Work on more efficient methods for the multi-objective optimization step, to reduce the number of trained

					optimal accuracy	models. Also, work on real-world high-dimensional engineering applications
16.	Efficient and Robust Automated Machine Learning	Advances in neural information processing systems (2015)	Scikit-learn	Bayesian optimization methods	Systems automatically choose a good algorithm and feature pre-processing steps for a new dataset at hand, and also set their respective hyper-parameters	Work on regression or semi-supervised problems. Also, work on large datasets

Methodology For Auto-ML Software/Library

A. Auto-WEKA [1] [4]

Auto-WEKA is working on an issue called "The combined Algorithm Selection and hyper-parameter Optimization (CASH)" [1]. Auto-WEKA automatically selects an algorithm and its hyper-parameters from the WEKA Java packages for a given dataset [4]. It automatically developed models for a wide range of data sets with the help of the WEKA package.Auto-WEKA worked on both categorical and continuous hyper-parameters with the help of Sequential Model-based Optimization and Bayesian Optimization techniques [1]. With the concept of machine learning, it calculates the cross-validation loss function iteratively and selects the best candidate for hyper-parameter configuration, and updates the model with new knowledge [1].Auto-WEKA 2.0, which supports regression algorithms and Tree-based Bayesian Optimization techniques, was released in 2017 [1].

Benefits:

- Auto-WEKA is working on the 'Combined Algorithm Selection and hyper-parameter Optimization (CASH) problem.'
- Auto-WEKA is more suitable with JAVA package.

- Bayesian Optimization and Tree-Based Bayesian Optimization Techniques were used by Auto-WEKA.
- Auto-WEKA will assist non-expert users to identify the best machine-learning algorithms and hyper-parameter settings relevant to their applications, resulting in improved performance.

Constraints:

- Auto-WEKA tools can handle only small datasets and only work on JAVA Packages.

B. Hyperopt-sklearn [1]

Hyperopt-Sklearn is the same as Auto-WEKA, which is based on a Python package [1]. The HyperOpt-Sklearn library is an extended version of the HyperOpt library that can allow an automated search for classification and regression tasks in terms of data preparation methods, machine learning algorithms, and model hyper-parameters [1].The main aim of this AutoML library is to optimize the large-scale model. It also optimizes machine learning pipelines such as data preparation and model selection, among other things.

Benefits:

- Hyperopt is a large-scale AutoML open-source library.
- Hyperopt-sklearn works on Python.
- Hyperopt-sklearn has increased scalability.
- Simple to set up.
- Fast.
- The best model achieves high performance.

Constraints:

- Hyperopt-sklearn only works on Python packages.
- Various best models for the same (simple) dataset.
- Inadequate documentation.

C. Auto-sklearn [1] [4]

Auto-sklearn is a more advanced version of Auto-WEKAthat can use in scikit-learn which is one of the python libraries [4]. Auto-sklearn continued work on the 'Combined Algorithm Selection and hyper-parameter Optimization (CASH) problem, which is presented by Auto-WEKA, and better working on auto-ML concepts like Bayesian optimization [1]. Auto-sklearn is capable of solving a number of issues, such as classification, multi-label classification, regression, and clustering. Auto-sklearn provides supervised machine-learning techniques that are ready to use.For a new machine learning dataset and hyper-parameters, Auto-sklearn selects and optimizes the best learning algorithm.Automating tasks in machine learning pipelines such as data pre-processing, feature pre-processing, hyper-parameter optimization, model selection, and evaluation are referred to as automated machine learning (Auto-ML).Auto sklearn performs well with small and medium datasets, but developers run into problems when dealing with large datasets. Support vector machines (SVM), Random Forests (RF), Gradient Boosting Machines (GBM), K-means, and other ML algorithms are implemented by Auto-sklearn.Auto-

sklearn encapsulates 15 classification algorithms, and 14 feature pre-processing algorithms, and handles data scaling, categorical parameter encoding, and missing values.

Benefits:

- It works on the CASH problem.
- Auto-Sklearn makes the automated process more effective and faster.
- It learns from previous models used on similar datasets and can generate automatic ensemble models for greater accuracy.
- It automatically searches for the best Machine Learning models that fit the data using Bayesian Optimization.
- Auto-sklearn uses meta-learning to find relevant datasets and prior knowledge to accelerate optimization.
- Involves several pre-processing methods (handling missing values, normalizing data).
- Looks for optimal ML pipelines in a large search space (15 classifiers and more than 150 hyper-parameters are searched).
- Cutting-edge technology thanks to the use of meta-learning, Bayesian optimization, and ensemble techniques.

Constraints:

- Auto-Sklearn performs poorly on large datasets and lacks support for regression algorithms.
- Auto-sklearn is a completely automated and black-box learning system. It searches vast space models and builds complex ensembles with high precision, consuming a significant amount of computation and time in the process.

D. TPOT [1] [4]

Tree-based Pipeline Optimization Tool (TPOT) is a python-based open-source library that includes a scikit-learn library for data preparation and machine-learning-based AutoML models.With the help of Genetic Programming (GP), TPOT optimize the machine-learning pipeline.TPOT is capable of solving classification and regression problems [1].TPOT has three main operators: (i) Future Pre-processing Operators (ii) Feature Selection Operators (iii) Supervised Classification Operators. Each of these operators was treated as a GP primitive and these GP trees were built for each of these operators.TPOT is an adaptable library because, with the help of the TPOT library, we can easily insert or delete nodes from the pipeline [1]. TPOT intelligently explores thousands of possible pipelines to identify the most suitable one for your data, so we can say that TPOT is one of the most time-consuming parts of machine learning.After the TPOT has finished its search, you can build a pipeline by providing the Python code for the best pipeline found [4].

Benefits:

- TPOT is such a flexible tool so we can easily add and remove nodes from the pipeline.
- Provide the same best model for the same (simple) dataset every time.
- Provide well-written documentation.

- The best model achieves good results.
- TPOT is a genetic algorithm with its own base regressor and classifier methods.
- The best model is easily exported to Python code.
- Enhance the productivity of current data scientists.
- Makes machine learning more readily available to those who are non-data scientists.
- The basic models do not require domain knowledge or human input.
- All the complex tasks like data processing, model selection, and parameter tuning can be handled automatically with the help of TPOT.

Constraints:

- TPOT may take a long time to complete its search.
- As a result, it typically takes a long time to execute and is not feasible for large datasets.
- It can recommend multiple solutions for the same dataset.
- Very slow.
- Difficult to install.
- It cannot handle natural language inputs.
- It is not able to process categorical strings.
- As we increase the number of generations, the time taken to find the best model increases exponentially.
- Not good with multiclass or multi-label data.

E. Auto-Compete [1]

Auto-Compete is a semi-automated AutoML library which is released in 2016. It was developed with a specific purpose [1].The main aim of Auto-Compete is to obtain an initial statistical model in machine learning and data science challenges.The workflow for Auto-Compete included the following steps: 1. Data Splitter and Identifier of Datatypes 2. Stacker and Feature Selector 3. Hyper-parameter selector and Model [1]. If your dataset is in form of text, then Natural Language Processing (NLP) algorithms are used in this way from given input datasets, the system could even determine which type of machine learning algorithms is required. It also solves the overfitting problem [1].

Benefits:

- Auto-Compete is a semi-automated library.
- It automatically identifies machine learning types and avoids overfitting.
- It was good enough on a single use case so it is more work on specific solutions.

Constraints:

- It was overly focused on a single use case, making it difficult to apply as a generic solution.
- It faced difficulty to solve generic solutions.
- It only deals with tabular datasets.

F. Penn-AI [1]

Penn-AI, which was developed in 2017, was used in commercial industries. Penn-AI was introduced as a learning tool thathelps us to identify the best models based on given datasets. It will not replace data scientists. It is also focusing on genetic programming (GP) [1]. PennAI is used in various domains such as the healthcare, and biomedical sectors.Penn-AI developed a very systematic and defined workflow, with the help of human involvement. A user-friendly Graphical User Interface (GUI) is also built with the help of Penn-AI [1].

Benefits:

- It is more focused on the healthcare and biomedical domains.

Constraints:

- Penn-AI supported only a few techniques from the scikit-learn package.

G. H2O Auto-ML [4]

H2O Auto-ML is a leading Auto-ML library for machine learning projects. It automates various operations like algorithm selection, feature generation, hyper-parameter tuning, and iterative modeling. It assists machine learning projects in efficient training and evaluating ML models without error. H2O Auto-ML improves project performance. H2O is an open-source platform that simplifies applying various ML algorithms to a given dataset. It includes several statistical and machine learning algorithms, including deep learning. H2O Auto-ML automates model selection. It is used in data analysis platforms. A User-friendly interface is also created with the help of H2O Auto-ML.It can automate machine learning workflows such as automatic learning and optimizing multiple models within user-defined time periods [4].It can perform classification tasks on tabular data. As a result, using tools, text classification can only be performed if the text data is prepared beforehand. H2O also supports AutoML, which ranks the various algorithms based on their performance. H2O also performs well on Big Data. This allows data scientists to test different machine learning models on their datasets and choose the best one for their needs. H2O maintains familiar interfaces such as Python, R, Excel, and JSON. It allows big data enthusiasts and experts to explore models, and datasets using a variety of simple to advanced algorithms. Data collection is simple. Making decisions is difficult. H2O makes it quick and simple to gain insights from your data by using faster and more accurate predictive modeling. H2O envisions a single platform for online scoring and modelingH2O.ai is a tool that manages the entire cycle of data analysis, including:

• Data cleaning and pre-processing
• Model selection and evaluation
• Deployment

H2O Auto-AL allows you to create Web Graphical User Interface (GUI) that can select parameters by simply pointing and clicking.

Benefits:

- It provides automated model selection.
- By using H2O Auto-ML, we can design an easy-to-use interface for non-technical users.

- Highly customizable.
- Very fast and powerful.

Constraints:

- In H2O there is a lack of machine learning algorithms.
- There is no feature extraction.

H. Google Cloud Auto-ML [4]

Google Cloud Auto-ML was introduced to the market in 2018, and it has a very user-friendly interface as well as excellent performance. It provides a simple graphical interface for preparing and storing datasets as well as machine-learning tools. Google Cloud Auto-ML delivered high-quality models with minimal effort and maximum performance by using neural-structure search technology and Google's cutting-edge transfer learning.Google Cloud AutoML provides three products: vision, natural language processing, and translation [4].However, Google Cloud Auto ML is a paid platform that can only be used for commercial projects. Furthermore, this Auto ML toolkit is free to use for research purposes throughout the year. Google Cloud Auto-ML provides cloud service. It automates learning models. It also optimizes machine learning problems such as natural language processing (NLP), sentiment analysis, image classification, and so on.

Benefits:

- Google Cloud Auto-ML can produce high-quality models that meet business requirements.

Constraints:

- It is not an open-source AutoMLlibrary, and the price point is based on which package you choose.

I. Auto-Keras

Auto-Keras is one of the AutoMLlibrariesbased onKeras which is basically used in machine-learning projects.It was created by Texas A&M University's DATA Lab.Auto-Keras is one of the open-source libraries that can performAutoML for deep learning models in order to improve Bayesian optimization.Auto-Keras can automate pre-processing steps like scaling and feature extraction with the help of high-level APIs. The main goal of Auto-Keras is to make machine learning available to everyone.It can perform classification tasks using both raw text data and tabular data.It is used to perform classification tasks using text embedding generated automatically and internally. Auto-Keras includes building blocks for text, image, and structured data classification and regression. It makes machine-learning projects available to anyone who needs them.The main objective of AutoML is to create deep learning tools that are easily accessible to domain experts with limited backgrounds in data science or machine learning.Auto-Keras includes functions for searching for deep learning complex model architecture and hyper-parameters automatically.This AutoMLprovides an efficient method for automatically identifying top-performing models with predictive modelingtasks. Auto-Keras searches using Neural Architecture Search (NAS) algorithms to eventually eliminate the need for deep learning engineers.

Benefits:

- It worked on a deep learning framework.
- It can be operated both locally and in the cloud.
- It provides a straightforward and user-friendly application programming interface (API)
- Auto-Keras stands out by automating all complex machine-learning tasks such as data processing, model selection, and parameter tuning.

Constraints:

- Auto-Keras can take a long time to accomplish its task because it uses a deep learning approach.

Comparative Analysis

Table I: Auto-ML LibraryResearch till Date

Auto-ML Tool	**Release Year**	**Pre-Processing**	**Feature Selection**	**Model Selection**	**Optimization of Hyperparameter**	**Learning with Ensemble**	**Learning with Meta**	**Count of Citation**
Auto-WEKA [1] [4]	2014	Yes	Yes	Yes	Yes	No	No	1,071
Hyperopt-sklearn [1]	2014	Yes	No	Yes	Yes	No	No	1,432
Auto-sklearn [1] [4]	2015	Yes	Yes	Yes	Yes	Yes	No	5,507
TPOT [1] [4]	2016	Yes	No	Yes	Yes	Yes	No	2,668
Auto-Compete [1]	2017	No	No	Yes	Yes	Yes	Yes	5
Penn-AI [1]	2017	Yes	No	Yes	Yes	No	No	61
H2O Auto-ML- [4]	2018	Yes	No	Yes	Yes	Yes	No	2,389

Google Cloud Auto-ML [4]	2018	Yes	No	Yes	Yes	Yes	No	N/A
Auto-Keras	2019	Yes	Yes	Yes	Yes	Yes	Yes	1,757

Table II: Auto-ML Library Strengths and Weakness

Auto-ML Tool	Strengths	Weaknesses
Auto-WEKA [1] [4]	- Can automatically select and configure machine learning algorithms - Can optimize hyperparameters for better performance	- Limited to the algorithms available in WEKA - Can be computationally expensive - Limited documentation - May not perform well on datasets outside of WEKA's scope
Hyperopt-sklearn [1]	- Uses Bayesian optimization for hyperparameter tuning - Can optimize multiple models at once	- Limited to the algorithms available in scikit-learn - Can be computationally expensive - Limited documentation - May not perform well on datasets outside of scikit-learn's scope
Auto-sklearn [1] [4]	- Can automatically select and configure machine learning algorithms - Uses Bayesian optimization for hyperparameter tuning - Can handle classification and regression problems	- Can be computationally expensive - Limited to the algorithms available in scikit-learn - May not perform well on datasets outside of scikit-learn's scope - Limited documentation
TPOT [1] [4]	- Can automatically select and configure machine learning algorithms	- Can be computationally expensive - Limited to the algorithms available in scikit-learn

	- Uses genetic algorithms for hyperparameter tuning - Can handle classification and regression problems	- May not perform well on datasets outside of scikit-learn's scope - Limited documentation
Auto-Compete [1]	- Can handle a wide range of machine-learning tasks - Uses automated feature engineering -Uses Bayesian optimization for hyper parameter tuning	- Can be computationally expensive - Limited documentation - May not perform well on all types of machine learning tasks
Penn-AI [1]	- Can handle a wide range of machine-learning tasks - Uses genetic algorithms for hyper parameter tuning - Can handle large datasets	- Can be computationally expensive - Limited documentation - May not perform well on datasets outside of its scope
H2O Auto-ML [4]	- Can handle a wide range of machine-learning tasks - Uses automated feature engineering - Can handle large datasets	- Limited to the algorithms available in H2O - Can be computationally expensive - Limited documentation
Google Cloud Auto-ML [4]	- Can handle a wide range of machine-learning tasks - Uses automated feature engineering - Can handle large datasets	- Limited to the algorithms available in Google Cloud - Requires a Google Cloud account and associated costs - Limited documentation
Auto-Keras	- Can automatically select and configure neural network architectures - Uses Bayesian optimization for hyper parameter tuning - Can handle image, text, and tabular data	- Limited to neural network architectures - Can be computationally expensive - Limited documentation - May not perform well on datasets outside of its scope

Table III: Auto-ML Techniques Strengths and Weakness

Auto-ML Technique	Strengths	Weaknesses
Automated Model Selection [1] [2] [11]	- Can select the best model for a given dataset - Can lead to better performance and more accurate models	- Can be computationally expensive
Hyper parameter Optimization[2] [11] [19] [22]	- Can greatly improve the performance of machine learning models - Can find the optimal values for hyperparameters	- Can be computationally expensive
Neural Architecture Search (NAS)[18] [19] [22]	- Can lead to highly accurate models - Can find the optimal neural network architecture for a given dataset	- Can be very computationally expensive
Bayesian Optimization [15] [19]	- Can be very effective in finding the best hyperparameters - Can lead to highly accurate models	- Can be computationally expensive
Genetic Algorithms [15]	- Can be very effective in optimizing machine learning models - Can find optimal solutions even with highly complex datasets	- Can be computationally expensive - Can be sensitive to the initial population and selection criteria

In summary, each Auto-ML tool has its own strengths and weaknesses, and the choice of tool will depend on the specific problem at hand and the available computational resources. Auto-WEKA, Hyperopt-sklearn, Auto-sklearn, TPOT, Auto-Compete, Penn-AI, H2O Auto-ML, Google Cloud Auto-ML, and Auto-Keras are some of the most used Auto-ML tools. Some tools are limited to the algorithms available in their respective libraries, while others can handle a wide range of machine-learning tasks. Similarly, some tools use genetic algorithms for hyper-parameter tuning, while others use Bayesian optimization or automated feature engineering. Computational expense is a concern for most Auto-ML tools, although some are specifically designed to handle large datasets.

Conclusion and Future Scope

In conclusion, Auto-ML techniques have emerged as powerful tools that can automate many time-consuming and challenging aspects of the machine-learning process. These techniques offer several benefits, such as improving the efficiency of the model development process, democratizing access to machine learning technology, and enabling non-experts to build models.

However, Auto-ML techniques also have some constraints, such as limitations on the algorithms available, potential performance issues on datasets outside of the tool's scope, and computational expense. Additionally, while Auto-ML can automate many aspects of the machine-learning process, it is not a substitute for domain expertise, and human involvement is still necessary to ensure the quality and accuracy of the final model.

As for future directions for the theoretical assessment of Auto-ML techniques, there are several promising areas for research. These include developing better methods for evaluating the performance and efficiency of Auto-ML techniques, expanding the scope of Auto-ML to address more complex and diverse machine learning tasks, and exploring the potential ethical implications of widespread adoption of Auto-ML technology. Additionally, there is a need for more comprehensive documentation and transparency in the development and implementation of Auto-ML techniques to ensure that they are accessible and understandable to a broader range of users.

References

[1] T. Nagarajah and G. Poravi, “A Review on Automated Machine Learning (AutoML) Systems,” 2019 IEEE 5th International Conference for Convergence in Technology, I2CT 2019, pp. 1–6, 2019, doi: 10.1109/I2CT45611.2019.9033810.

[2] K. Chauhan et al., “Automated Machine Learning: The New Wave of Machine Learning,” 2nd International Conference on Innovative Mechanisms for Industry Applications, ICIMIA 2020 - Conference Proceedings, no. Icimia, pp. 205–212, 2020, doi: 10.1109/ICIMIA48430.2020.9074859.

[3] T. Dhaene, “Feature and Model Type Selection using Multi-Objective Optimization for AutoML”.

[4] Y. L.-I. J. of A. C. Technology and undefined 2018, “Analysis on trends of machine learning,” Koreascience.or.Kr, vol. 6, no. 3, pp. 303–308, 2018, [Online]. Available: https://www.koreascience.or.kr/article/JAKO201810263412472.page

[5] M. Hanussek, M. Blohm, and M. Kintz, “Can AutoML outperform humans? An evaluation on popular OpenML datasets using AutoML Benchmark,” ACM International Conference Proceeding Series, pp. 29–32, 2020, doi: 10.1145/3448326.3448353.

[6] P. Gijsbers, E. LeDell, J. Thomas, S. Poirier, B. Bischl, and J. Vanschoren, “An Open Source AutoML Benchmark,” pp. 1–8, 2019, [Online]. Available: http://arxiv.org/abs/1907.00909

[7] M. Feurer, A. Klein, K. Eggensperger, J. T. Springenberg, M. Blum, and F. Hutter, “Efficient and robust automated machine learning,” Advances in Neural Information Processing Systems, vol. 2015-January, pp. 2962–2970, 2015.

[8] J. Drozdal et al., "Trust in AutoML: Exploring Information Needs for Establishing Trust in Automated Machine Learning Systems," 2020, doi: 10.1145/3377325.3377501.
[9] H. J. Escalante, "Automated Machine Learning—A Brief Review at the End of the Early Years," Natural Computing Series, pp. 11–28, 2021, doi: 10.1007/978-3-030-72069-8_2.
[10] B. Wang et al., "VEGA: Towards an End-to-End Configurable AutoML Pipeline," 2020, [Online]. Available: http://arxiv.org/abs/2011.01507
[11] Y.-W. Chen, Q. Song, and X. Hu, "Techniques for Automated Machine Learning," ACM SIGKDD Explorations Newsletter, vol. 22, no. 2, pp. 35–50, 2021, doi: 10.1145/3447556.3447567.
[12] A.-I. Imbrea, "Automated Machine Learning Techniques for Data Streams," 2021, [Online]. Available: http://arxiv.org/abs/2106.07317
[13] S. Brändle, M. Hanussek, M. Blohm, and M. Kintz, "Evaluation of Representation Models for Text Classification with AutoML Tools," Lecture Notes in Networks and Systems, vol. 359 LNNS, pp. 310–322, 2022, doi: 10.1007/978-3-030-89880-9_24.
[14] X. Shi, J. Mueller, N. Erickson, M. Li, and A. J. Smola, "Benchmarking Multimodal AutoML for Tabular Data with Text Fields," no. NeurIPS, 2021, [Online]. Available: http://arxiv.org/abs/2111.02705
[15] M. Wever, A. Tornede, F. Mohr, and E. Hullermeier, "AutoML for Multi-Label Classification: Overview and Empirical Evaluation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 9, pp. 3037–3054, 2021, doi: 10.1109/TPAMI.2021.3051276.
[16] K. T. Y. Mahima, T. N. D. S. Ginige, and K. De Zoysa, "Evaluation of Sentiment Analysis based on AutoML and Traditional Approaches," International Journal of Advanced Computer Science and Applications, vol. 12, no. 2, pp. 612–618, 2021, doi: 10.14569/IJACSA.2021.0120277.
[17] V. Lopes, A. Gaspar, L. A. Alexandre, and J. Cordeiro, "An AutoML-based Approach to Multimodal Image Sentiment Analysis," Proceedings of the International Joint Conference on Neural Networks, vol. 2021-July, 2021, doi: 10.1109/IJCNN52387.2021.9533552.
[18] L. Vaccaro, G. Sansonetti, and A. Micarelli, "An empirical review of automated machine learning," Computers, vol. 10, no. 1, pp. 1–27, 2021, doi: 10.3390/computers10010011.
[19] X. He, K. Zhao, and X. Chu, "AutoML: A survey of the state-of-the-art," Knowledge-Based Systems, vol. 212, no. Dl, 2021, doi: 10.1016/j.knosys.2020.106622.
[20] S. K. K. Santu, M. M. Hassan, M. J. Smith, L. Xu, C. Zhai, and K. Veeramachaneni, "AutoML to Date and Beyond: Challenges and Opportunities," ACM Computing Surveys, vol. 54, no. 8, 2022, doi: 10.1145/3470918.
[21] X. Zheng et al., "Evolving Fully Automated Machine Learning via Life-Long Knowledge Anchors," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 9, pp. 3091–3107, 2021, doi: 10.1109/TPAMI.2021.3069250.
[22] Li, Yaliang, Zhen Wang, YuexiangXie, Bolin Ding, Kai Zeng, and Ce Zhang. "Automl: From methodology to application." In Proceedings of the 30th ACM

International Conference on Information & Knowledge Management, pp. 4853-4856. 2021.