AUTOMATED MACHINE LEARNING THE NEW WAVE OF MACHINE LEARNING

[1]N. SRAVANI, [2]Dr. P.VELAYUTHAM
[1]MCA Student, [2]Assistant Professor
DEPARTMENT OF MCA
SREE CHAITANYA COLLEGE OF ENGINEERING, KARIMNAGAR

ABSTRACT

With the explosion in the use of machine learning in various domains, the need for an efficient pipeline for the development of machine learning models has never been more critical. However, the task of forming and training models largely remains traditional with a dependency on domain experts and time-consuming data manipulation operations, which impedes the development of machine learning models in both academia as well as industry. This demand advocates the new research era concerned with fitting machine learning models fully automatically i.e., AutoML. Automated Machine Learning(AutoML) is an end-to-end process that aims at automating this model development pipeline without any external assistance. First, we provide an insights of AutoML. Second, we delve into the individual segments in the AutoML pipeline and cover their approaches in brief. We also provide a case study on the industrial use and impact of AutoML with a focus on practical applicability in a business context. At last, we conclude with the open research issues, and future research directions.

I. INTRODUCTION

Data analysis is a powerful tool for learning insights on how to improve the decision making, business model and even products. This involves the construction and training of a machine learning model which faces several challenges due to lack of expert knowledge. This challenges can be overcomed by using automated machine learning(AutoML) field. AutoML refers to the process of studying a traditional machine learning model development pipeline to segment it into modules and automate each of those to accelerate workflow. With the advent of deeper models, such as the ones used in image processing, Natural Language Processing, etc., there is an increasing need for tailored models that can be crafted for specific workloads. However, such specific models require immense resources such as high capacity memory, strong GPUs, domain experts to help during the development and long wait times during training. The task gets critical as there is not much work done for creating a formal framework for deciding model parameters without the need for trial and error. These nuances emphasized the need for AutoML where automation can reduce turnaround times and also increase the accuracy of the derived models by removing human errors. In recent years, several tools and models have been proposed in the domain of AutoML. Some of these focus on particular segments of AutoML such as feature engineering or model selection, whereas some models attempt to optimize the complete pipeline. These tools have matured enough to be able to compare with human experts on Kaggle competitions and at times have beat them as well, showcasing their veracity.There are wide variety of applications based on AutoML such as autonomic cloud computing, Intelligent Vehicular networks, Block Chain,Software Defined Networking, among others. This paper aims at providing an overview of the advances seen in the realm of AutoML in recent years. We focus on individual aspects of AutoML and summarize the improvements achieved in recent years. The motivation of this paper stems from the unavailability of a compact study of the current state of AutoML. While we acknowledge the existence of other surveys, their motive is to either provide an in-depth understanding of a

particular segment of AutoML, provide just an experimental comparison of various tools used or are fixated towards deep learning models.

There is a lot of buzz for machine learning algorithms as well as a requirement for its experts. We all know that there is a significant gap in the skill requirement. The motive of H2O is to provide a platform which made easy for the non-experts to do experiments with machine learning.

H2O architecture can be divided into different layers in which the top layer will be different APIs, and the bottom layer will be H2O JVM.

H2O's core code is written in Java that enables the whole framework for multi-threading. Although it is written in Java, it provides interfaces for R, Python and few others shown in the architecture, thus enabling it to be used efficiently.

In crux, we can say that H2O is an open source, in memory, distributed, fast and scalable machine learning and predictive analytics that allow building machine learning models to be an ease.

If you are using python the same method is applied in it too, from command line pip install -U h2o and h2o will be installed for your python environment. The same process will go on for Initializing h2o.

The h2o.init() command is pretty smart and does a lot of work. At first, it looks for any active h2o instance before starting a new one and then starts a new one when instance are not present.

It does have arguments which helps to accommodate resources to the h2o instance frequently used are:

nthreads: By default, the value of nthreads will be -1 which means the instance can use all the cores of the CPU, we can set the number of cores utilized by passing the value to the argument.

max_mem_size: By passing a value to this argument you can restrict the maximum memory allocated to the instance. Its od string type can pass an argument as '2g' or '2G' for 2 GBs of memory, same when you want to allocate in MBs.

Once instance is created, you can access the flow by typing http://localhost:54321 in your browser. Flow is the name of the web interface that is part of h2o which does not require any extra installations which is written in CoffeeScript(a JavaScript like language). You can use it for doing the following things:

II. LITERATURE SURVEY

1) Automated Machine Learning In Practice: State Of The Art And Recent Results

AUTHORS: Lukas Tuggener, Mohammadreza Amirian, Katharina Rombach, Stefan L¨orwald, Anastasia Varlet, Christian Westermann, and Thilo Stadelmann

A main driver behind the digitization of industry and society is the belief that data-driven model building and decision making can contribute to higher degrees of automation and more informed decisions. Building such models from data often involves the application of some form of machine learning. Thus, there is an ever growing demand in work force with the necessary skill set to do so. This demand has given rise to a new research topic concerned with fitting machine learning models fully automatically-AutoML. This paper gives an overview of the state of the art in AutoML with a focus on practical applicability in a business context, and provides recent benchmark results of the most important AutoML algorithms.

2) Bert: Pre-training of deep bidirectional transformers for language understanding

AUTHORS: Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova

We introduce a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from

unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, including pushing the GLUE score to 80.5% (7.7% point absolute improvement), MultiNLI accuracy to 86.7% (4.6% absolute improvement), SQuAD v1.1 question answering Test F1 to 93.2 (1.5 point absolute improvement) and SQuAD v2.0 Test F1 to 83.1 (5.1 point absolute improvement).

III. SYSTEM ANALYSIS

EXISTING SYSTEM:

In the existing system the data preprocess has dine with structured data. Even though data pre-processing consumes a large chunk of time in an ML pipeline, it is astonishing to see the inadequate amount of work done to automate it. For data preprocessing, it can be noted that while the data pre process approaches are adequate for structured data, work still needs to be done to assimilate on Structured data. We suggest the incorporation of data-mining methods as they can deal with such unformed data. This can allow AutoML pipelines to create models capable of learning from Internet sources. In feature engineering, it should be noted that most methods used until now adhere to supervised learning. However, dataset specificity is high, and therefore, AutoML pipelines should be as generic as possible to accommodate the diverse datasets. Therefore, a gradual paradigm shift towards unsupervised.

DISADVANTAGES OF EXISTING SYSTEM:

- Feature Generation is not up to the mark where domain experts excepted results.
- Most AutoML tools emphasize the performance but in the real world, that's just one aspect being covered in machine learning projects. So the companies can't compromise the computing plus storage specification sheet.
- CASH(Combined Algorithm Selection and Hyperparameter) problem considers model selection and hyperparameters optimization as a single hierarchical parameter.
- **Algorithm**: SmartML,J48,C50

PROPOSED SYSTEM:

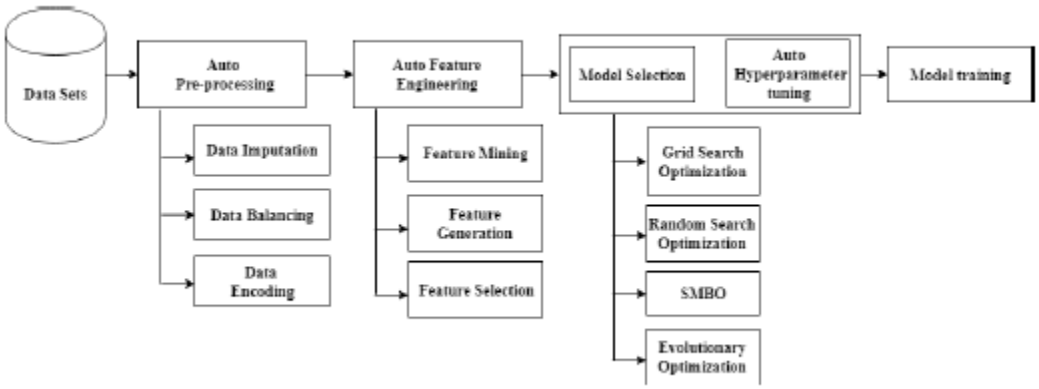
The proposed System aims at providing an overview of the advances seen in the realm of AutoML in recent years. We focus on individual aspects of AutoML and summarize the improvements achieved in recent years. The motivation of proposed system stems from the unavailability of a compact study of the current state of AutoML. While we acknowledge the existence of other surveys, their motive is to either provide an in-depth understanding of a particular segment of AutoML, provide just an experimental comparison of various tools used or are fixated towards deep learning models.

ADVANTAGES OF PROPOSED SYSTEM:

- We segment the AutoML pipeline into parts and review the contributions in each of these segments.
- We explore the various state-of-the-art tools currently available for AutoML and evaluate them.
- We also incorporate the advancements seen in machine learning which seems to be overshadowed by deep learning in recent years.

Algorithm:H2O-AutoML, LinearRegression, Gradient Boosting Regressor

SYSTEM ARCHITECTURE



IV. IMPLEMENTATION:

MODULES:

- **User**
- **Admin**
- **Data Preprocess**
- **AutoML**

MODULES DESCRIPTION:

User:
The User can register the first. While registering he required a valid user email and mobile for further communications. Once the user register then admin can activate the customer. Once admin activated the customer then user can login into our system. User can do the data preprocess. At the time of data preprocess the h2o auto Ml server will start automatically and initiate the data from dataset of the adult's data. This data publically available at shap server in the central repository. In the h2o models will load automatically in the project and will split the data our requirements. The files will be pickled and stored in file path locations. Later user can test the salary vs experience dataset. Here user can give the dynamically test split size. Based this size the salary dataset can split and will train to our model and fetch the predicted results. Use can compare the original and predicted results.
.

Admin:
Admin can login with his credentials. Once he login he can activate the users. The activated user only login in our applications. The admin can set the training and testing data for the project dynamically to the code. When h2o servers starts we can see the all models loading process. Admin can also view the user perfumed results. The test size and acquired scores also displayed in the admin page.

Data Preprocess:
Data pre-processing guarantees the delivery of quality data derived from the original dataset. It is an important step due to the unavailability of quality data as a large portion of information generated and stored is usually semi-structural or even non-structured in form. However, even though it is a crucial part of any machine learning pipeline, it is reported to be the least enjoyable part, with authors stating that 60-80% of data scientists finding it to be the most mundane and tedious job. In AutoML, certain data-preprocessing operations are hardcoded, which are then applied to a given dataset in certain combinations such that the overall clarity and usability of the data increases. We have largely classified these operations into the following categories based on our surveys of recent papers.

AutoML:

H2O is a fully open-source, distributed in-memory machine learning platform with linear scalability. H2O supports the most widely used statistical & machine learning algorithms, including gradient boosted machines, generalized linear models, deep learning, and many more. We suggest The incorporation of data-mining methods as they can deal with such unformed data. This can allow AutoML pipelines to create models capable of learning from Internet sources. In feature engineering, it should be noted that most methods used until now adhere to supervised learning. However, dataset specificity is high, and therefore, AutoML pipelines should be as generic as possible to accommodate the diverse datasets.

V. CONCLUSION

In this paper, we provide insights to the readers about the various segments of AutoML with a conceptual perspective. Each of these segments has various approaches that have been briefly explained to provide a concise overview. We also discuss the various trends seen in recent years including suggestions of thirsty research areas which need attention.

REFERENCES

[1] Lukas Tuggener, Mohammadreza Amirian, Katharina Rombach, Stefan L¨orwald, Anastasia Varlet, Christian Westermann, and Thilo Stadelmann. Automated machine learning in practice: state of the art and recent results. In 2019 6th Swiss Conference on Data Science (SDS), pages 31–36. IEEE, 2019.

[2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[4] Avatar Jaykrushna, Pathik Patel, Harshal Trivedi, and Jitendra Bhatia. Linear regression assisted prediction based load balancer for cloud computing. In 2018 IEEE Punecon, pages 1–3. IEEE.

[5] Jitendra Bhatia, Ruchi Mehta, and Madhuri Bhavsar. Variants of software defined network (sdn) based load balancing in cloud computing: A quick review. In International Conference on Future Internet Technologies and Trends, pages 164–173. Springer, 2017.

[6] Ishan Mistry, Sudeep Tanwar, Sudhanshu Tyagi, and Neeraj Kumar. Blockchain for 5g-enabled iot for industrial automation: A systematic review, solutions, and challenges. Mechanical Systems and Signal Processing, 135:106382, 2020.

[7] Jitendra Bhatia, Yash Modi, Sudeep Tanwar, and Madhuri Bhavsar. Software defined vehicular networks: A comprehensive review. International Journal of Communication Systems, 32(12):e4005, 2019.

[8] Jitendra Bhatia, Ridham Dave, Heta Bhayani, Sudeep Tanwar, and Anand Nayyar. Sdn-based real-time urban traffic analysis in vanet environment. Computer Communications, 149:162 – 175, 2020.

[9] Xin He, Kaiyong Zhao, and Xiaowen Chu. Automl: A survey of the state-of-the-art. arXiv preprint arXiv:1908.00709, 2019.

[10] Radwa Elshawi, Mohamed Maher, and Sherif Sakr. Automated machine learning: State-of-the-art and open challenges. arXiv preprint arXiv:1906.02287, 2019.

[11] Anh Truong, Austin Walters, Jeremy Goodsitt, Keegan Hines, Bayan Bruss, and Reza Farivar. Towards automated machine learning: Evaluation and comparison of automl approaches and tools. arXiv preprint arXiv:1908.05557, 2019.

[12] Shichao Zhang, Chengqi Zhang, and Qiang Yang. Data preparation for data mining. Applied artificial intelligence, 17(5-6):375–381, 2003.

[13] Erhard Rahm and Hong Hai Do. Data cleaning: Problems and current approaches. IEEE Data Eng. Bull., 23(4):3–13, 2000.

[14] Dipali Shete and Sachin Bojewar. Auto approach for extracting relevant data using machine learning. International Journal of Electronics, 6:0, 2019.