Auto-ML (Automated Machine Learning)
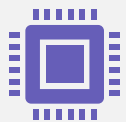
Presented by

M.Mohana

What is AutoML?

Process of **automating the end-to-end process** of applying machine learning algorithms to real-world problems.

Automation includes tasks such as **data preprocessing, feature engineering, model selection, hyperparameter tuning, and model evaluation**

Aims to make machine learning **more accessible to non-experts** and accelerate the development of machine learning applications by automating repetitive and time-consuming tasks

M.Mohana, Research Scholar

Components of AutoML

Data Preprocessing: asks like handling missing values, encoding categorical variables, scaling features, and splitting data into training and validation sets.

Feature Engineering: Automatic generation or selection of relevant features to improve model performance.

Model Selection: Choosing the most appropriate machine learning algorithms based on the dataset and problem type.

Hyperparameter Tuning: Optimizing model hyperparameters to achieve better performance.

Model Evaluation: Assessing model performance using metrics like accuracy, precision, recall, F1-score, etc.

M.Mohana, Research Scholar

AutoML for Different Data Types

- Used to perform a wide range of machine learning tasks, including classification, regression, clustering, deep learning, and even forecasting, Computer Vision

Tabular Data: Classification and Regression

- Can be used also be used for Time Series Forecasting, by automatically exploring and evaluating multiple time series forecasting algorithms including traditional methods like ARIMA (AutoRegressive Integrated Moving Average).

Image Data: Computer Vision

- Process of selecting the most suitable model architectures for image recognition tasks.

Text Data: Natural Language Processing (NLP)

- Process of extracting meaningful insights from text data, eliminating the need for manual feature engineering analyzing linguistic patterns, relationships, and structures within the text, and facilitating the extraction of relevant information.
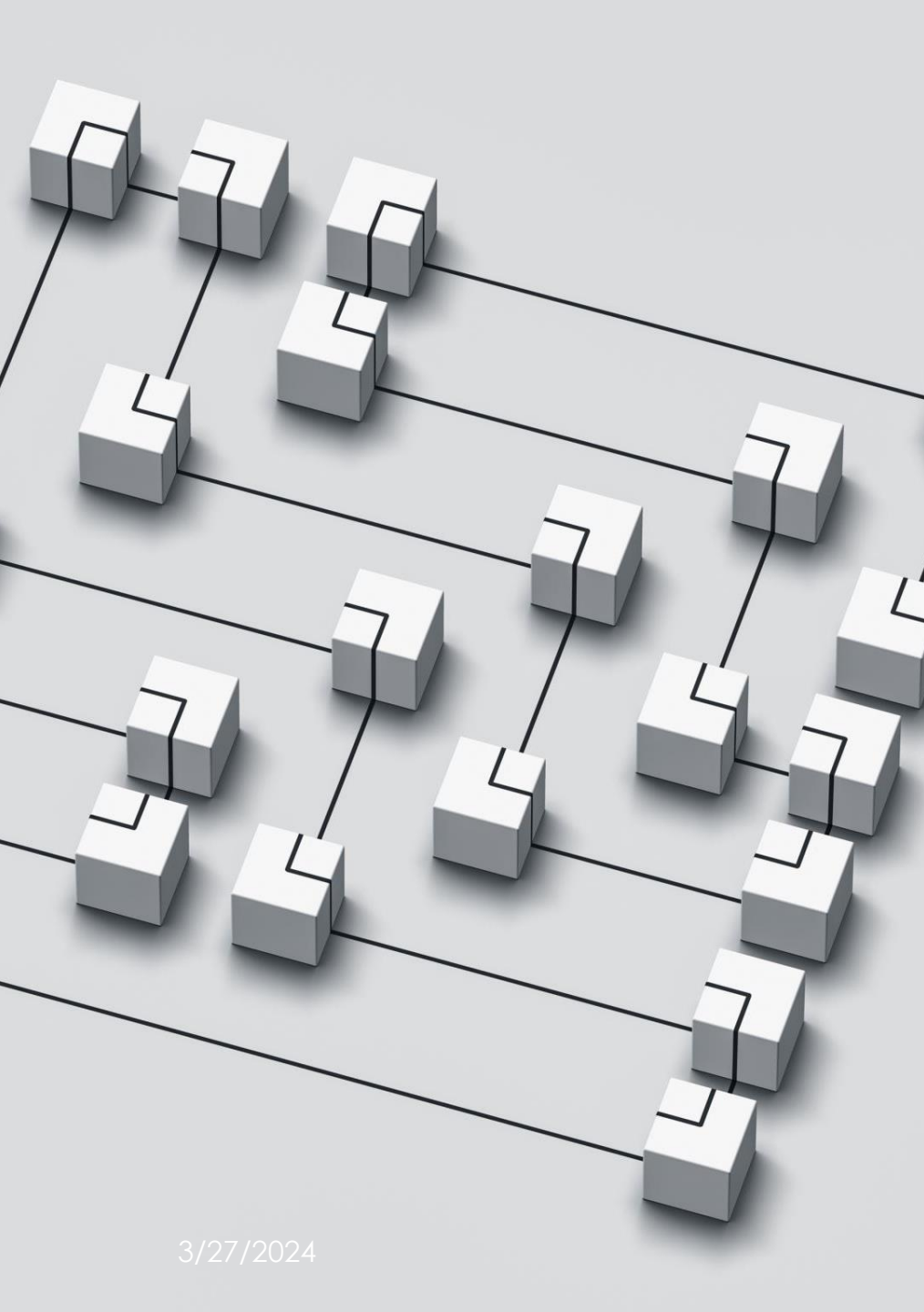
Benefits of AutoML

Time Efficiency

- Reduces the time required for manual model development and optimization.

Accessibility

- Enables non-experts to leverage machine learning techniques without extensive knowledge.

Consistency

- Ensures consistent application of best practices across different projects.

Scalability

- Facilitates the handling of large datasets and complex models.

Optimization

- Automatically tunes hyperparameters to improve model performance.

https://www.geeksforgeeks.org/what-is-automl-in-machine-learning/

Top AutoML tools and platforms

- Google AutoML
- H2O.ai
- Auto-sklearn
- TPOT
- Microsoft Azure AutoML
- Databricks AutoML
- TIBCO Data Science
- AutoKeras
- Auto-PyTorch
- Amazon Lex
- AutoGluon
- AutoWEKA

https://www.run.ai/guides/automl

Challenges of AutoML

Black Box Models

- Automated processes may result in complex models that are challenging to interpret.

Limited Customization

- Some AutoML tools may have limited flexibility compared to manual model development.
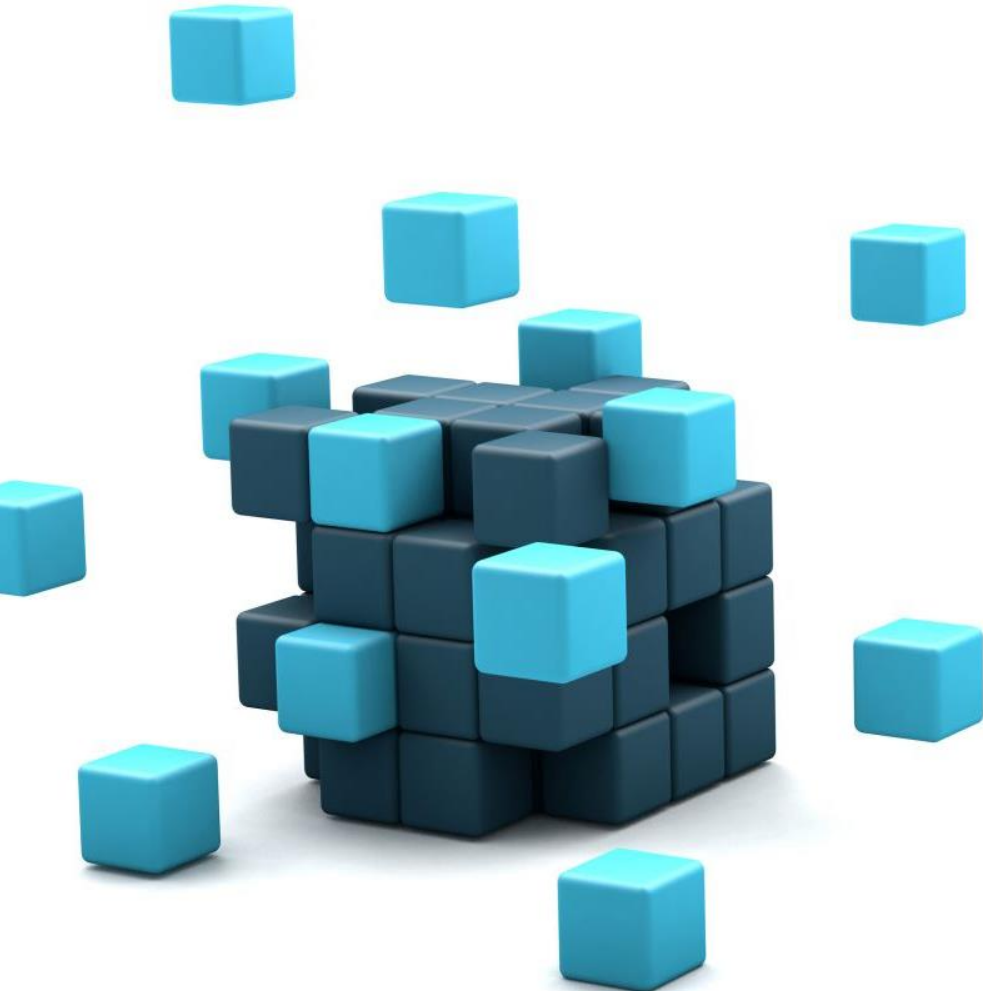
Domain Knowledge

- Understanding domain-specific requirements and nuances is still crucial.

Data Quality

- AutoML performance heavily depends on the quality and cleanliness of the input data.

Overfitting

- Automated processes may lead to overfitting if not properly controlled.

Auto Keras

- AutoKeras is an open-source AutoML (Automated Machine Learning) library built on TensorFlow and developed by the Data Analytics and Decision Support Lab (DADS) at Texas A&M University.

- Simplify the process of building and training ML models by automating several key steps such as architecture search, hyperparameter tuning, and model selection.

Some key features of AutoKeras include:

- AutoKeras uses neural architecture search (NAS) techniques to automatically search for the best neural network architecture for a given dataset and task.

- Automates the process of hyperparameter tuning, such as learning rates, batch sizes, and activation functions, to improve model performance.

- AutoKeras provides a user-friendly API that allows developers and data scientists to quickly build and train machine learning models without extensive manual configuration.

- Supports various machine learning tasks, including classification, regression, image classification, text classification, and structured data prediction.

- Since AutoKeras is built on top of TensorFlow, it seamlessly integrates with other TensorFlow tools and libraries, making it easy to incorporate into existing TensorFlow workflows.

Install autokeras

```python
1 !pip install autokeras
```

```python
1 import pandas as pd
2 import numpy as np
3 from autokeras import StructuredDataClassifier
4 from sklearn.model_selection import train_test_split
```

```python
1 diabetes = pd.read_csv("diabetes.xls")
2 diabetes.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

```
1 target_col = "Outcome"
2 X = diabetes.loc[:, diabetes.columns != target_col]
3 y = diabetes.loc[:, target_col]
```

```
1 X_train, X_test, y_train, y_test = train_test_split(X, y,
2                                                     test_size=0.20,
3                                                     random_state=2021)
```

```
1 search = StructuredDataClassifier(max_trials = 20)
```

```
1 search.fit(x=X_train, y=y_train)
```

```
Trial 20 Complete [00h 00m 04s]
val_accuracy: 0.7058823704719543

Best val_accuracy So Far: 0.7647058963775635
Total elapsed time: 00h 01m 43s
20/20 [==============================] - 1s 2ms/step - loss: 0.7140 - accuracy: 0.5065
<keras.callbacks.History at 0x7831a6958e90>
```

```
1 loss, acc = search.evaluate(X_train, y_train, verbose=0)
2 print('Accuracy is {}'.format(acc))
```

```
Accuracy is 0.6856677532196045
```

Thank you..!

Keep Learning..!