Comparative Study of Google Cloud AutoML vs Google Cloud Custom Training

Gayatri Samal, 11037887

Ketan Darekar, 11037367

Cris Kohn, 11014674

Vipul Zope, 11037893

2024-02-09

SRH University Heidelberg

School of Information, Media and Design

Applied Computer Science (ACS)

Degree course "ACS"

Under the Supervision of

Prof. Dr-Ing Kamellia Reshadi .

Mr. Paul Tanzer.

Certificate

This is to certify that the work contained in this thesis entitled **"Google Cloud AutoML vs. Google Cloud Custom Training "** is a Bonafide work of **Gayatri, Ketan, Cris & Vipul** (11037887, 11037367, 11014674, 11037893), carried out in the Applied Computer Science, SRH Heidelberg University under my supervision and that it has not been submitted elsewhere for a Degree.

Heidelberg, Germany
February 9th, 2024

Paul Tanzer & Prof. Dr.-Ing Kamellia Reshadi
Applied Computer Science
SRH University Heidelberg

Acknowledgement

We would like to express our deepest gratitude towards our Supervisor **Paul Tanzer & Prof. Dr.-Ing Kamellia Reshadi** for the constant help and encouragement from the start of the thesis work.
We have been fortunate to have Supervisors who gave us the freedom to explore on our own and at the same time helped us plan the future work-plan with timely reviews and suggestions wherever required. Thank you for giving us enough extension of time whenever required and having faith in our team.

Heidelberg, Germany

February 9th,2024

Abstract

German

Diese Studie vergleicht Google AutoML und Custom Training in Verwendung mit Tensor Flow, zwei beliebte Methoden, um Bilder zu klassifizieren. Jeder Vorgang wird in Ressourcen, Hyperparamter Tuning, Modellerstellung und -training, sowie Dataset Vorbereitung untersucht. Diese Arbeit bietet ebenfalls eine Empfehlung von AutoML im Vergleich zu Custom Training, basierend auf den Anforderungen für dieses Projekt. Zusätzlich werden in dieser Arbeit Wachstumsmöglichkeiten für beide Methoden für die Zukunft aufgelistet. Diese Studie ist nützlich für jede*r Nutzer*in, welche*r TensorFlow verwenden möchte und zwischen AutoML und Custom Training wählen muss.

English

This study compares Google AutoML and Custom Training with TensorFlow, two well-liked methods for automating machine learning processes, in detail. Each approach is examined from several angles in this study, including resource requirements, deployment, assessment and validation, hyperparameter tuning, model creation and training, and dataset preparation. The article also provides advice for selecting AutoML vs Custom Training depending on the particular requirements of the project. In addition, the study highlights possible areas for growth and improvement and explores future directions and advances for both solutions. All things considered, this study is a useful tool for anyone who

wants to use TensorFlow to automate their machine learning processes and choose between AutoML and Custom Training.

Table of contents

1 Introduction

Google Cloud Platform (GCP) is a cloud computing platform and infrastructure developed by Google to develop, deliver, and administer web-based applications and services.

1.1 GCP Cloud Computing Architecture

The foundation of GCP's architecture is a global network of highly secure and available data centers connected by a dependable and quick network. The fundamental elements of GCP's architecture are as follows.

1. Virtual Machines: GCP provides a range of virtual machine (VM) options, such as custom and pre-configured images, for running applications.

2. Containers: GCP offers options for Docker containers in addition to managed Kubernetes clusters.

3. Storage: GCP provides a variety of storage options, such as file, object, and block storage.

4. Databases: GCP offers managed database options, including NoSQL and relational databases.

5. Networking: GCP provides a worldwide network with load balancing, VPN, and firewall rule options for connecting resources.

6. Big Data and Analytics: Google Cloud Platform offers services for processing and analyzing massive amounts of data, such as big data processing, data warehousing, and machine learning.

1.2 Why GCP?

GCP is a desirable choice for businesses and organizations looking to use cloud services because of its infrastructure, security, affordability, innovation, and openness. However, a company's specific requirements and goals will determine the best cloud option for it.

1. Advanced Infrastructure: As part of Google's world-class infrastructure, GCP is built upon the company's excellent network of data centers and quick fiber-optic connections. This architecture provides GCP with high performance, scalability, and reliability characteristics.

2. Security and compliance: GCP prioritizes security. Google has a dedicated team of security professionals who work around the clock to protect the platform from threats and makes large investments in security.

3. Cost-Effective: GCP offers both adjustable and reasonably priced cost options. Additionally, GCP offers per-second charging, so users only pay for the resources they utilize.

4. Innovation: GCP is only the latest example of Google's long history of innovation. GCP gives users access to cutting-edge technologies including serverless computing, big data analytics, and machine learning.

5. Openness: Since GCP is based on open-source technologies, users can access a variety of tools and resources. Additionally, GCP supports open standards like Kubernetes, making it simple for users to connect to other platforms and services.

6.worldwide Network and Edge Locations: For worldwide applications, GCP's vast network and edge locations lower latency and enhance user experience.

1.3 Introduction Of AutoML and Custom Training

AutoML and Custom Training are two popular approaches for developing machine learning models in the Google Cloud AI platform.

Automated Machine Learning, or AutoML, is a cutting-edge methodology intended to democratize AI by making it understandable to laypeople. With minimal ML knowledge, customers can train high-quality models customized to their data using GCP's user-friendly AutoML interface. Complex procedures like feature engineering, model selection, and hyperparameter tweaking are automated by it. This automation greatly cuts down on the time and resources usually needed, while also simplifying the model-building process. For common applications like data classification, image recognition, and natural language processing, AutoML is especially helpful since it can provide reliable models without requiring a lot of customization.

Custom Training, on the other hand, gives developers more customization options and total control over the machine learning model. Although this method necessitates a greater level of coding expertise, it enables more refined models that may be customized to the project's unique requirements.

GCP's AutoML and Custom Training both help close the gap between cutting-edge AI research and useful, real-world applications. While custom training gives specialists a platform to push the limits of machine learning, AutoML enables enterprises and developers with little to no ML knowledge to take advantage of AI. Taken as a whole, they reflect the wide range of tools that GCP offers, guaranteeing that the platform will support your journey regardless of whether you're wanting to innovate at the cutting edge or are just starting out with AI.

1.4 Purpose of comparison

In the context of machine learning, the goal of contrasting AutoML with Custom Training is to assist users and organizations in selecting the best course of action depending on their unique requirements, level of experience, and the nature of the ML project.

Two well-liked methods for accomplishing this objective—AutoML and Custom Training (using TensorFlow)—have emerged as choices as the need for automating machine learning workflows grows. Both methods are appealing choices for automating machine learning operations because of their distinctive features.

Google's open-source TensorFlow machine learning framework provides a full range of tools for developing, refining, and implementing machine learning models. Conversely, Google created the AutoML set of automated machine learning tools to streamline the process of creating, honing, and implementing machine learning models.

The comparison that follows will concentrate on the distinctions and parallels between TensorFlow-based custom training and AutoML. Developers and data scientists will benefit from this.

2 AutoML For Image Classification

The process of automating the complete pipeline of creating, Training, testing, and implementing an image classification model is known as AutoML (automated machine learning).

In Order to find the best-performing model for a given dataset, AutoML algorithms automatically search through a wide range of potential combinations of model architectures, hyperparameters, and data pre-processing approaches using machine learning techniques. Convolutional Neural Networks (CNNs), a type of deep learning model that is intended to identify patterns in images, are commonly used in AutoML for image categorization.

The process of creating accurate Picture classification models can be greatly shortened by using AutoML, and it can also be easier for individuals without specialist machine learning knowledge to complete.

2.1 Definition of AutoML

The Term AutoML, which stands for Automated Machine Learning, describes the application of automated tools and procedures to streamline and speed up developing, constructing, and implementing machine learning models. The main objective of AutoML is to make artificial intelligence accessible to people with different degrees of machine learning experience, without requiring them to become intimately involved in the intricacies of model architecture creation, hyperparameter tuning, and algorithm selection.

Machine learning Tasks that have historically required a great deal of manual labor and knowledge are automated in AutoML to optimize the workflow. This covers activities like feature engineering, model selection, hyperparameter.

optimization, data pretreatment, and even model deployment. AutoML democratizes machine learning by automating these procedures, opening the field to a wider range of users, including domain experts, business analysts, and engineers with less experience with machine learning.

It streamlines the process of choosing, building, and customizing machine learning models. In comparison to hand-coded methods, machine learning is more user-friendly and frequently yields faster, more accurate results when it is automated.

- The best model architecture is automatically chosen by AutoML depending on the job type and the data that is supplied.
- AutoML optimizes hyperparameters, such as learning rates and regularization terms, to enhance model performance.
- AutoML is improving fraud detection and risk assessment, which is revolutionizing the financial services sector.
- Facial recognition can be aided by image recognition.
- Risk assessment and management in banking, finance, and insurance.
- AutoML is enhancing supply chain management and logistics by optimizing routes and forecasting demand.

- The energy industry is benefiting from automated machine learning by increasing energy efficiency, predicting equipment problems, and optimizing power generation.
- In agriculture, it can be applied to speed up the process of quality testing.

2.2 Advantages and Disadvantages of AutoML

2.1.1 Advantages

Scalability:

Organizations may quickly scale their machine learning efforts with the help of AutoML. AutoML technologies can handle the increasing workload without requiring a corresponding increase in human effort as the need for machine learning models rises.

Diminished Human Error:

The possibility of human Error in the machine learning process is decreased via automation. By handling repetitive processes like model selection and hyperparameter tweaking, AutoML solutions reduce the possibility of errors and inconsistencies during the model building process.

Cost Saving:

Developing machine learning Models may be made faster and cheaper by automating the machine learning pipeline, which includes features engineering, model selection, hyperparameter tuning, and deployment.

This is especially helpful for companies who have tight schedules or few resources.

Constant Enhancement:

Continuous Model improvement strategies are present in many AutoML frameworks. They are appropriate for dynamic and changing datasets since they can adjust to variations in the data distribution.

2.1.2 Disadvantages

Data Quality and Preprocessing:

The quality of the input data must be sufficient for AutoML to function. Problems like noise, outlier, or missing data can have a detrimental effect on how well AutoML model's function. Prior to using AutoML, users should make sure that the data is of high quality and attend to preprocessing activities.

Excessive Fit:

Overfitting is a potential problem with AutoML tools, particularly when improperly designed. When a model works well on training data but is unable to generalize to fresh, untried data, this is known as overfitting. Users should employ strategies to reduce the risk of overfitting and exercise caution when it arises.

Restricted Algorithm Choice:

For model selection, AutoML programs usually provide a predefined set of algorithms. Although this works well for a lot of tasks, there may be some circumstances in which a unique or specialized approach is preferable.

Users that have certain algorithmic needs could find AutoML's options to be restricted.

2.3 AutoML workflows for image classification

Here's an overview of how Google's AutoML solution works:

Data Preparation:

Gather and organize your image dataset. Ensure that the dataset is labeled with appropriate class labels for supervised learning. Split the dataset into training, validation, and test sets.

Data pre-processing:

Utilize preprocessing methods to get the photos ready for model training. Resizing photos to a consistent size, standardizing pixel values, and enhancing the dataset with methods like rotation, flipping, and zooming are typical preprocessing procedures.

Feature extraction

Feature Extraction in the context of picture classification often uses convolutional neural networks (CNNs) trained as feature extractors. Using AutoML tools, pre-trained models such as VGG, ResNet, or Inception can be automatically selected and fine-tuned to extract relevant features from the photos.

Data modelling:

Model architectures that are appropriate for picture classification are automatically explored and chosen by AutoML frameworks. To find the best model for the given dataset, this entails testing several deep learning architectures, such as various CNN architectures.

Hyperparameter tuning:

To get the best performance from your ML model, you will need to tune its hyperparameters, which are the settings that control the behavior of the model. AutoML includes tools for hyperparameter tuning, which can help you find the optimal hyperparameter settings for your model.

Prediction:

Your machine learning model can be trained and then used to forecast fresh data. To facilitate seamless integration of your model into your systems or applications, AutoML comes with tools for assessing your model's performance and deploying it as an API or web service.

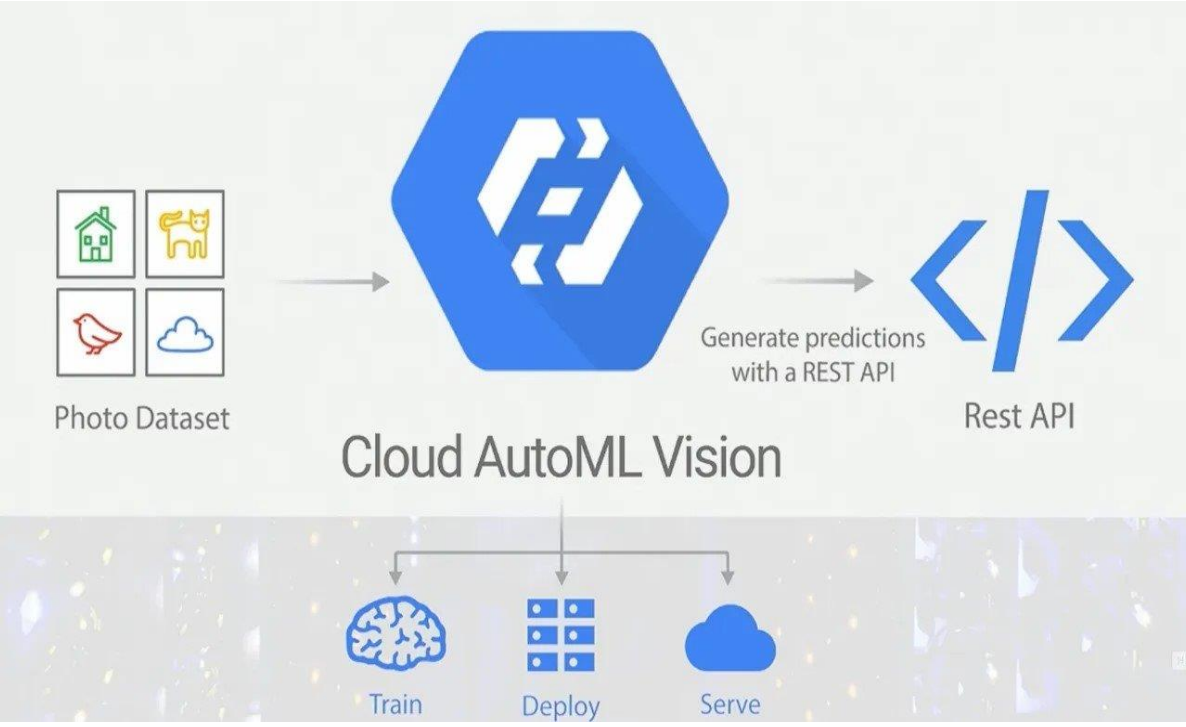

Figure 1 - AutoML Workflow

3 Custom Training (Using Tensor Flow)

Instead of utilizing AutoML, custom training is the process of training machine learning models with your own training codes or software. There are several approaches to individualized cloud computing training. The functionality of the training application is entirely in your control. We can utilize several algorithms, target any purpose, and perform any additional customisation. Popular machine learning (ML) frameworks are linked with Vertex AI, a cloud computing service. (1)

For training the model, it lets users pick from a variety of pre-built containers already in existence or create their own bespoke containers. One well-known ML framework that Vertex AI offers is TensorFlow, which gives developers the freedom to tailor the machine learning process to the requirements of their projects. (1)

Defining the model architecture, choosing and implementing metrics, loss functions, and custom layers are all part of the custom training process. Additionally, the training process is configured by choosing an optimizer, deciding on a batch size, and establishing the number of training epochs. Users can incorporate complicated deep learning architectures that might not be available in pre-built models or libraries and develop models that are well-suited to their specific use case by modifying the model architecture. (1)

3.1 Advantages and Disadvantages of Custom training

Advantages

- Community support
- Cost savings
- High efficiency at scale
- Optimized Performance
- Flexibility in Data Processing
- Integration with other tools

Disadvantages

- Time consuming
- Lack of pre-built models
- Maintenance and support
- Steep learning curve
- Resource-intensive

3.2 Custom Training workflows

Building a model that can classify photos into predetermined categories, enhancing the model's performance, and processing image data are all part of the custom training approach for image classification. Since convolutional neural networks (CNNs) are widely used to solve image classification tasks because of their efficacy in capturing spatial hierarchies in image data, this process combines general machine learning workflows with techniques specific to image processing and deep learning.

Gathering and Preparing Dataset:

Assemble a diverse and sizable enough image collection to cover every category the machine must be able to recognize. This could entail creating and labeling a new data set or using one that already exists. Eliminate duplicates, irrelevant samples, and corrupted photos. Make sure the dataset is evenly distributed among the classes or choose a method to address the imbalance. Use data augmentation methods including rotation, scaling, cropping, flipping, and color variation to broaden the training set's variability and minimize overfitting.

Model Design and Selection:

Select whether to apply transfer learning, utilize a pre-trained model for fine-tuning, or create a CNN from scratch. Due to their efficacy on various image classification tasks, pre-trained models (such as VGG, ResNet, and Inception) are often used as beginning points. Convolutional layers, pooling layers, fully connected layers, and activation functions (ReLU is common) need all be designed when creating a CNN from scratch or altering a current architecture.

Custom Training Process:

Use data loaders to load and prepare images for training and validation quickly and effectively. This involves resizing photos, translating them into the appropriate format for the model, and normalizing pixel values. Select the optimization technique (SGD, Adam, etc.), the loss function (cross-entropy is a common choice for classification) and build the model while indicating the metrics to track. Train the model on the training set, then use the validation set to fine-tune the hyperparameters and apply dropout or modify the learning rate to avoid overfitting.

Model Evaluation and Validation

Examine how well the model performs on the validation set. Examine the kinds of mistakes it commits, the confusion matrix, and the performance of each class individually to find areas that need work. To increase accuracy and decrease overfitting, adjust the model architecture, hyperparameters, and data augmentation techniques based on validation performance.

Deployment:

Install the trained model in a real-world setting so it can identify fresh photos. This could entail incorporating the model into a service or application.

Track the model's performance over time to identify any decline or changes in the distribution of data that may require retraining.

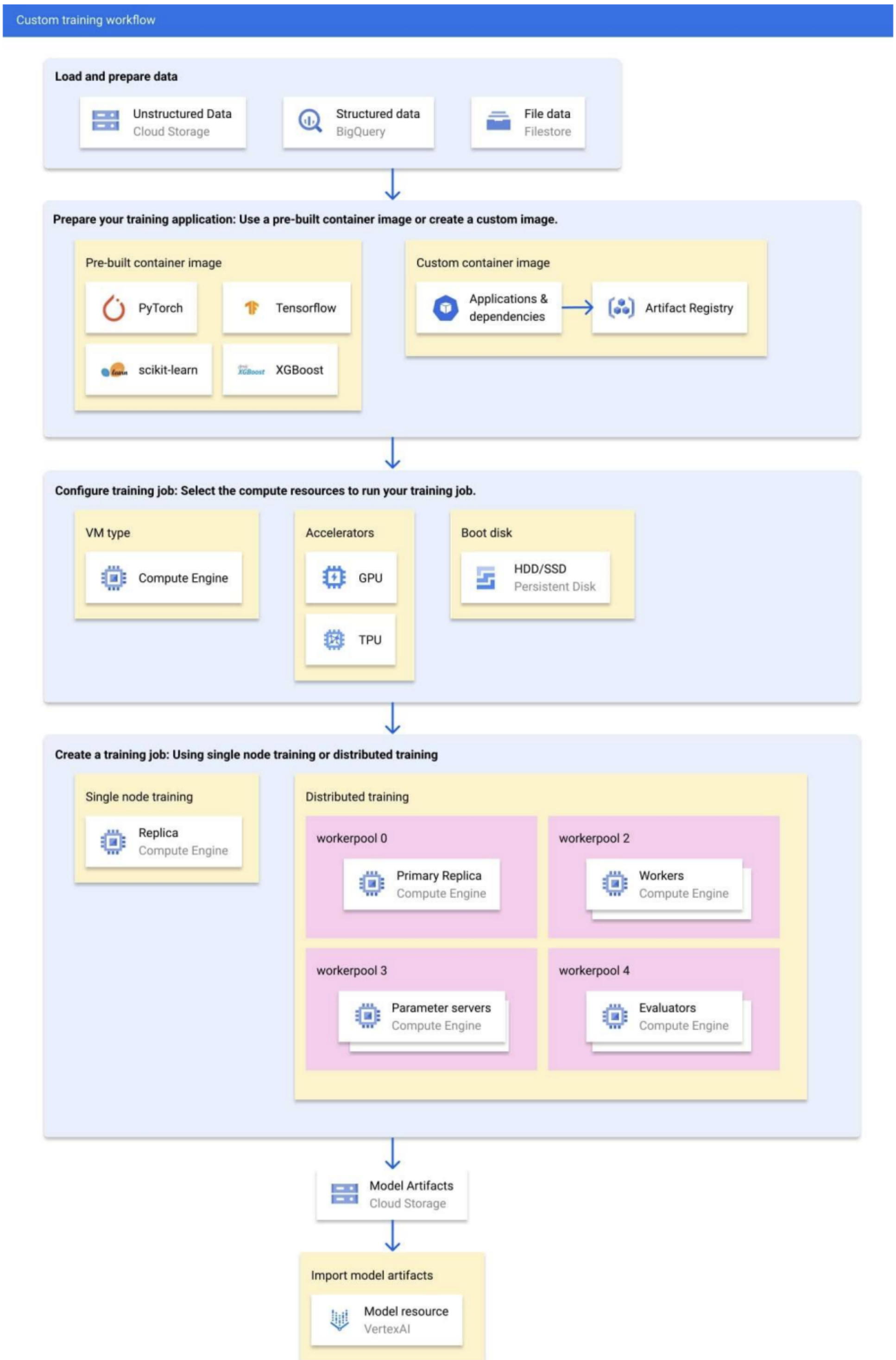

Figure 2 - Custom training workflow

4 Comparison of AutoML and Custom Training (Using TensorFlow)

Two well-liked methods for automating machine learning workflows are AutoML and custom training (using TensorFlow). While Custom Training gives developers more flexibility and customization to control the machine learning process, AutoML delivers a collection of automated machine learning tools that make it easier for developers to create, train, and deploy machine learning models. Although each strategy has benefits, the decision between AutoML and Custom Training ultimately comes down to the particular requirements and project objectives. (2)

4.1 Accuracy and performance

AUTOML

Platforms for AutoML are efficient in that they automate tasks like feature engineering, hyperparameter tweaking, and model selection. This automation can save a great deal of time and processing power when developing models, especially for users who are not very experienced with machine learning.

These systems provide a reliable solution for a variety of applications since they can scale across many datasets and problem kinds with little modification. However, depending on the dataset's complexity and the particular AutoML tool being used, performance can differ in terms of processing time and resource usage.

Model training to production can be streamlined by using AutoML technologies, which frequently come with deployment solutions or are integrated with them. Nevertheless, it is possible that the models produced are not always best suited for deployment limitations such as inference speed or memory consumption.

AutoML systems typically perform well across a broad range of standard tasks and datasets, leveraging large model repositories and extensive hyperparameter search spaces to find effective solutions. For many applications, AutoML can achieve accuracy levels close to those of custom-trained models with far less effort.

CUSTOM TRAINING

The selection of pre-processing techniques is a crucial component that can influence the performance of customized TensorFlow models for image categorization. Tokenization, stemming, and stop-word removal, for instance, can enhance the quality of the input data, which can increase the accuracy of the model. (3)

Custom training enables deep model optimization for tasks, which may improve resource efficiency and accelerate inference times. This strategy may be especially crucial for applications that have stringent performance requirements, such low-power or real-time computing.

Practitioners can experiment with state-of-the-art models and customize the architecture to meet performance parameters other than accuracy, including lowering latency or memory footprint, via bespoke training.

There exist multiple optimal methodologies that can enhance the efficacy and precision of TensorFlow models in text categorization. Pre-trained word embeddings, adjusting the pre-trained model on the target dataset, and early halting to avoid overfitting are a few of these. (3)

4.2 Flexibility and customization

AutoML from Google provides pre-trained models that can be used with minimal customization. Users only need to provide their dataset and let the models do the work. While this approach offers ease of use and high accuracy, it may not be suitable for all use cases as the models are not customizable and users may be limited in their choice of algorithms.

Custom Training, on the other hand, offers greater flexibility and customization. Users can train their own models and fine-tune the algorithms based on their specific needs. Using TensorFlow also provides a wide range of pre-built models, which can be customized to suit different use cases. This allows users to achieve higher accuracy and better performance than with AutoML, especially when dealing with complex datasets.

Overall, the choice between AutoML and TensorFlow depends on the specific needs of the user. If ease of use and high accuracy are the main requirements, then AutoML may be the better choice. However, if customization and flexibility are needed, then TensorFlow provides a more powerful platform for developing and deploying machine learning models.

4.3 Cost and resource requirements

AUTOML

Depending on the needs of the project, there are different costs and resource requirements for Google AutoML and Custom Training using TensorFlow in Vertex AI of Google Cloud Platform. Because automated machine learning operations make AutoML more convenient, it may be more expensive overall. In contrast, custom training using TensorFlow may need more resources up front but may end up being more economical over time.

For AutoML, the pricing starts at €0.18 per hour for training and €1.32 per hour for prediction, with additional charges for storage and data processing. You pay for three primary actions when using Vertex AI AutoML models:

Educating the model

implementing the model at a destination

Utilizing the model for forecasting

For Vertex AutoML models, Vertex AI uses preconfigured machine configurations; the resource utilization is reflected in the hourly charge for these operations.

The volume and complexity of your training data will determine how long it takes to train your model. Before models can offer online explanations or forecasts, they need to be installed.

If no forecast is produced, you are still charged for each model that is deployed to an endpoint. To cease paying more, you have to undeployed your model. There are no charges associated with models that are not deployed or have not been deployed.

If training is unsuccessful, you only pay for the computer hours used.

Image data Video data Tabular data Text data

Operation	Price per node hour (classification)	Price per node hour (object detection)
Training	$3.465	$3.465
Training (Edge on-device model)	$18.00	$18.00
Deployment and online prediction	$1.375	$2.002
Batch prediction	$2.222	$2.222

Figure 3 - AutoML costs

CUSTOM TRAINING

For custom training, a notebook, a bucket and training job is required on google cloud. Furthermore, The model needs to be deployed to an endpoint.

- Notebooks start at $ 0.273 hourly.
- Custom training with the cheapest machine cost also $ 0.273 hourly
- Cloud storage costs $ 0.020 for one GB for one month.
- A machine is needed to deploy the model to an endpoint. This costs again $ 0.273 hourly.

Belgium (europe-west1)

Machine type	Price per hour (USD)
n1-standard-4	$0.240580
n1-standard-8	$0.481160
n1-standard-16	$0.962320
n1-standard-32	$1.924640
n1-standard-64	$3.849280
n1-standard-96	$5.782200
n1-highmem-2	$0.149730

Figure 4 – Custom training costs

Data storage

Click on a geographic area to view the at-rest costs for associated locations:

Regions | Dual-regions | Multi-regions

North America | South America | Europe | Middle East | Asia | Africa | Australia

Location	Standard storage (per GB per Month)	Nearline storage (per GB per Month)	Coldline storage (per GB per Month)	Archive storage (per GB per Month)
Warsaw (`europe-central2`)	$0.023	$0.013	$0.006	$0.0025
Finland (`europe-north1`)	$0.020	$0.010	$0.004	$0.0012
Belgium (`europe-west1`)	$0.020	$0.010	$0.004	$0.0012
London (`europe-west2`)	$0.023	$0.013	$0.007	$0.0025

Figure 5 - Data storage costs

4.4 Integration and compatibility

AUTOML

With Google AutoML, users can easily create and implement machine learning models on a cloud-based platform. Data management and resource access are made simple with AutoML's high degree of compatibility with other Google Cloud services, such as Big Query and Cloud Storage.

For creating and implementing machine learning models, AutoML is a robust platform that offers broad support for various activities and use cases. With its pre-built models and customization possibilities, it's simple to get started with machine learning, even for inexperienced users, and its connectivity with other Google Cloud services facilitates data management and resource access. (6)

CUSTOM TRAINING

Numerous tools and functionalities offered by Google Cloud Vertex AI provide simple integration and interoperability with TensorFlow. TensorFlow 1.x and 2.x are supported by Vertex AI, enabling customers to include the most recent features and functionalities of TensorFlow into their machine learning models. Additionally, Vertex AI offers a selection of pre-built TensorFlow models and libraries that are readily deployable and adaptable to various use cases. (7)

Vertex AI not only supports TensorFlow but also integrates with Big Query and Cloud Storage from Google Cloud to facilitate effective data processing and administration. This facilitates the easy access to and analysis of big datasets, which in turn makes the development of high-performance machine learning models easier. (7)

5 Use cases and examples.

5.1 AutoML for image classification in practice

- **Prepared our dataset**: Arranged our photos into a directory structure and got the dataset ready for upload.
- **Created a Google Cloud Storage bucket**: This is where we will store our trained model artifacts and training data.
- **Created an AutoML Vision dataset**: We created a new dataset and uploaded our training data to it using the AutoML Vision panel.
- **Trained the AutoML model**: We trained a new AutoML model using the dataset that was recently created. In addition, we defined various training variables and designed the model to use a particular type of machine.
- **Evaluated the trained model**: After training was finished, we used the test dataset to assess our model's performance.
- **Exported the trained model**: We exported our trained model to a Saved Model format that can be deployed to a production environment.
- **Deployed the model**: We deployed our trained model to a Vertex AI endpoint so that it can serve predictions in response to requests.

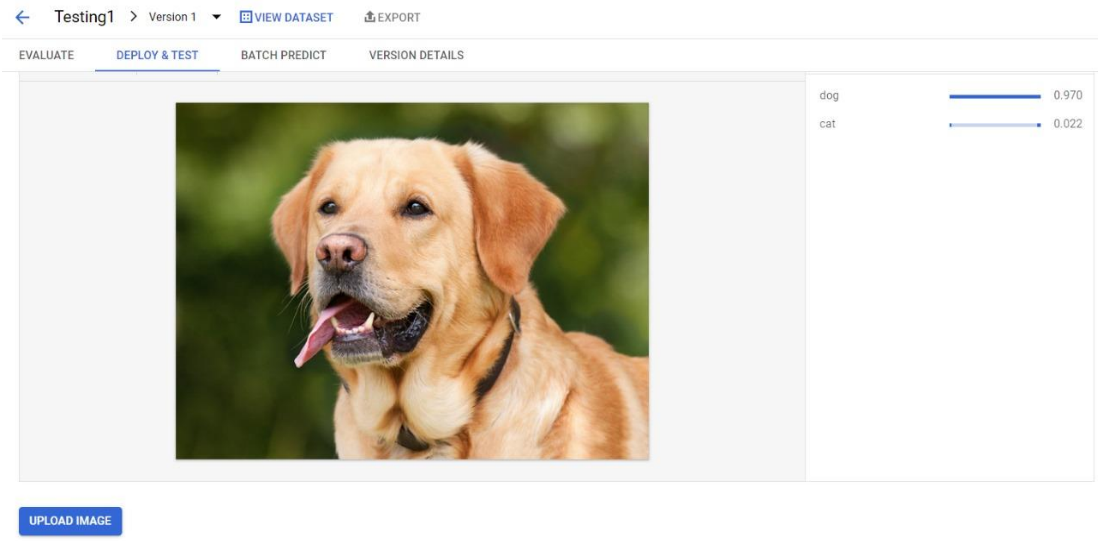

Figure 6 6 - Dog testing result (AutoML)

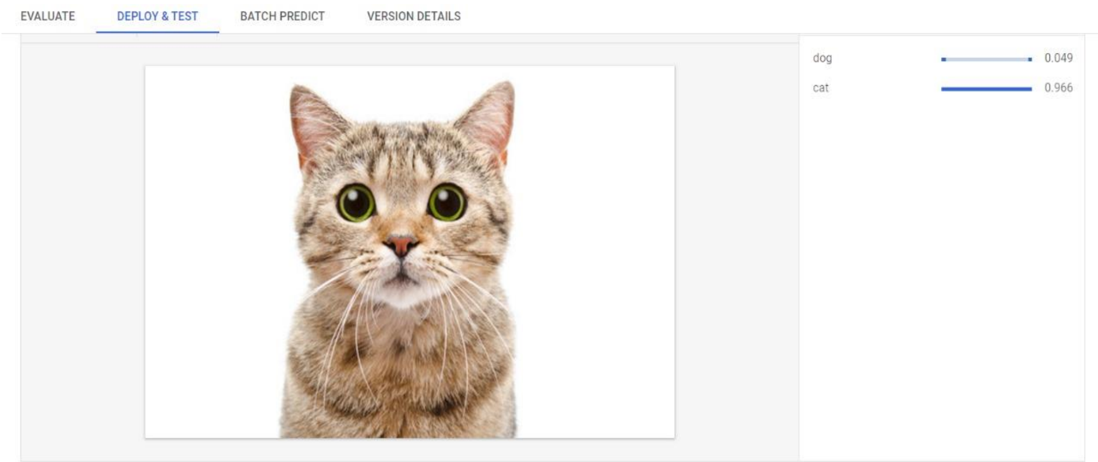

Figure 7 7 - cat testing result (AutoML)

5.2 TensorFlow in image classification in practice

Prepared our dataset: arranged our photos into a directory and was ready to pour them into a bucket.

Created a Google Cloud Storage bucket: Our training data and trained model artifacts were kept in this location. Comparative Analysis of Vertex AI's Google AutoML and Custom Training (using TensorFlow)

Created a Vertex AI custom job: Here, we used a YAML configuration file to set up our training job. The location of our training data, the kind of machine to be used, and the model's hyperparameters were all given.

Started the training job: We gave Vertex AI the task, and we used the console or the command line to track its progress.

Evaluated the trained model: Once the training job is complete, we evaluated the performance of your model using the test dataset.

Exported the trained model: Our trained model was exported to a format called Saved Model, which allows it to be used in a production setting.

Deployed the model: We deployed our trained model to a Vertex AI endpoint so that it can serve predictions in response to requests.

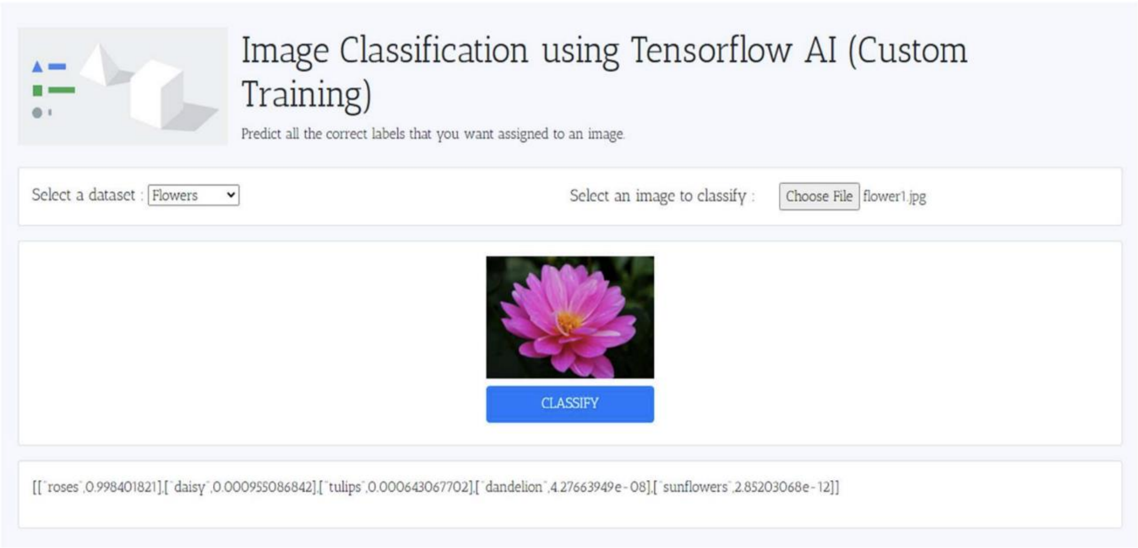

Figure 8 8 - Flower testing result (Custom training)

6 Conclusion

In summary, both Google AutoML and Custom Training (using TensorFlow) have their advantages and disadvantages. AutoML is a more user-friendly option but lacks flexibility and customization. Custom Training is more complex and requires more expertise but offers greater flexibility and control over the model.

Custom Training, however, needs some pre knowledge and coding. It is a complex process which fails if not done 100% correctly. Therefore, Custom Training is recommended for professionals or people who want precise results in cost of time consumption.

The first trainings were successful, then the following trainings, done the same way, failed after different durations. Therefore, a comparison could only be made for the first trainings.

6.1 Recommendations for choosing AutoML and custom training

The choice between AutoML and Custom Training ultimately depends on the specific needs and resources of the project. For those with limited resources and expertise, AutoML may be the best option. For those with more resources and a need for greater customization, Custom Training may be the better choice.

6.2 Future directions and development

As machine learning continues to evolve, it is likely that both AutoML and Custom Training will continue to develop and improve. AutoML may become more customizable, while Custom Training may become more user-friendly. Additionally, new tools and technologies may emerge that offer even more efficient and effective ways of automating machine learning workflows.

7 Index of figures

8 Literature Cited

1. TechTarget. What is TensorFlow? | Definition from TechTarget; 2018 [cited 2024 Feb 7]. Available from: URL: https://www.techtarget.com/searchdatamanagement/definition/TensorFlow.

2. Tan M, V. Le Q. Improving Accuracy and Efficiency through AutoML and Model Scaling; 2019 [cited 2024 Feb 7]. Available from: URL: https://blog.research.google/2019/05/efficientnet-improving-accuracy-and.html.

3. Lakshmanan L. Choosing between TensorFlow/Keras, BigQuery ML, and AutoML Natural Language for text classification. Towards Data Science 2019 Jan 4 [cited 2024 Feb 7]. Available from: URL: https://towardsdatascience.com/choosing-between-tensorflow-keras-bigquery-ml-and-automl-natural-language-for-text-classification-6b1c9fc21013.

4. Run:ai. Google AutoML: Quick Solution Overview; 2023 [cited 07-Feb-24]. Available from: URL: https://www.run.ai/guides/automl/google-automl.

5. Corbo A. What Is TensorFlow? Built In 2022 Dec 29 [cited 2024 Feb 7]. Available from: URL: https://builtin.com/data-science/tensorflow.

6. Google Cloud. TensorFlow-Integration 2024 Feb 6 [cited 07-Feb-24]. Available from: URL: https://cloud.google.com/vertex-ai/docs/start/tensorflow?hl=de.

7. Google Cloud. AutoML beginner's guide 2024 Jan 3 [cited 07-Feb-24]. Available from: URL: https://cloud.google.com/vertex-ai/docs/beginner/beginners-guide?hl=en.