



MLRun-Pipeline mit Iguazio

NetApp Solutions

NetApp
May 14, 2024

Inhalt

- MLRun-Pipeline mit Iguazio 1
 - TR-4834: NetApp und Iguazio für MLRun Pipeline 1
 - Technologischer Überblick 3
 - Software- und Hardware-Anforderungen erfüllt 7
 - Übersicht Über Die Fehlervorhersage Für Netzwerkgeräte: Anwendungsbeispiel 8
 - Setup-Übersicht 8
 - Bereitstellen der Anwendung 13
 - Schlussfolgerung 28

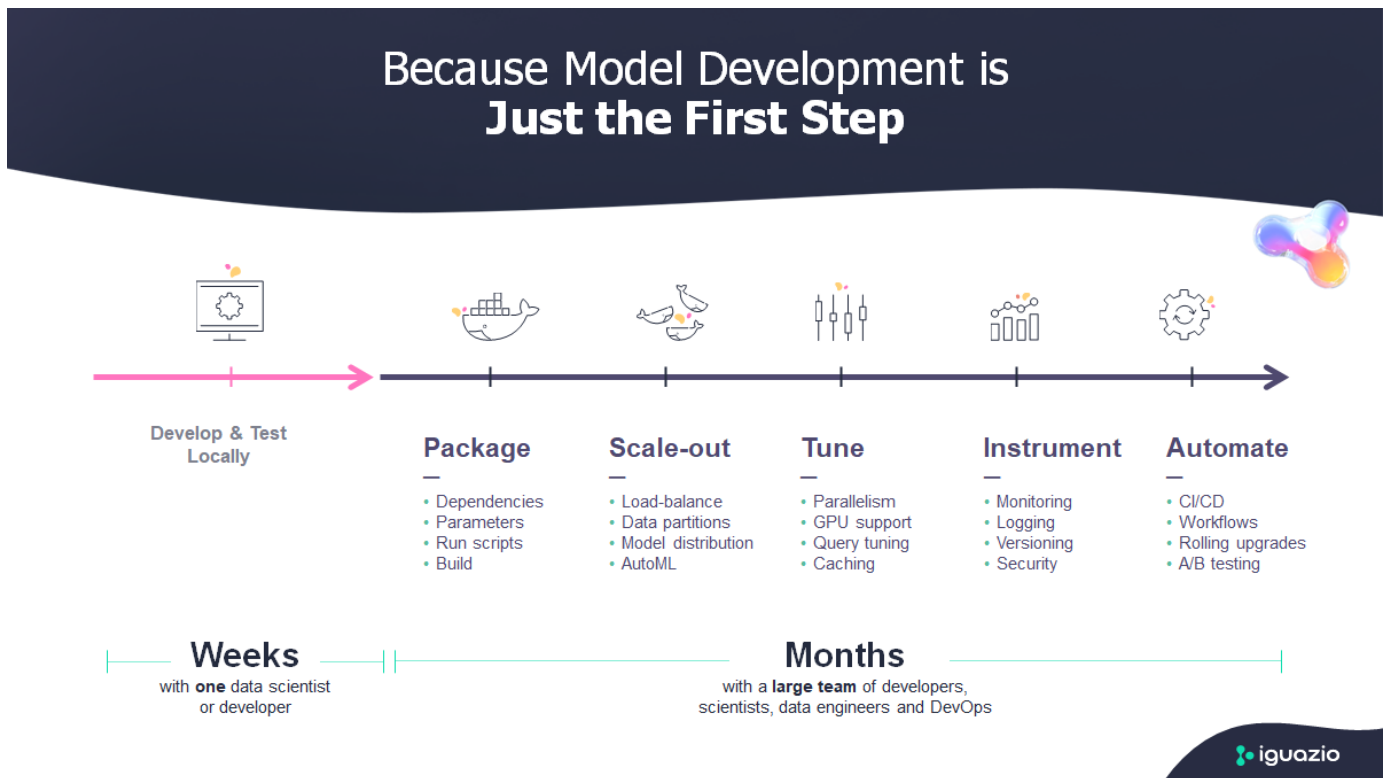
MLRun-Pipeline mit Iguazio

TR-4834: NetApp und Iguazio für MLRun Pipeline

Rick Huang, David Arnette, NetApp Marcelo Litovsky, Iguazio

In diesem Dokument werden die Details der MLRun-Pipeline unter Verwendung von NetApp ONTAP AI, der NetApp AI Control Plane, der NetApp Cloud Volumes Software und der Iguazio Data Science Platform behandelt. Wir verwendeten Nuclio serverlose Funktion, Kubernetes Persistent Volumes, NetApp Cloud Volumes, NetApp Snapshot Kopien, Grafana Dashboard, Und andere Dienste auf der Iguazio-Plattform, um eine End-to-End-Daten-Pipeline für die Simulation von Netzwerkausfall Erkennung zu bauen. Die Technologien von Iguazio und NetApp wurden integriert, um schnelle Modellbereitstellung, Datenreplizierung und Produktionsüberwachungsfunktionen vor Ort und in der Cloud zu ermöglichen.

Der Fokus der Arbeit eines Datenwissenschaftlers sollte auf das Training und Tuning von ml- (Machine Learning) und künstlicher Intelligenz (KI)-Modellen liegen. Allerdings verbringen Data Scientists nach Untersuchungen von Google ~80 % ihrer Zeit damit, herauszufinden, wie ihre Modelle mit Enterprise-Applikationen funktionieren und im Maßstab laufen lassen können, wie in der folgenden Abbildung dargestellt, die Modellentwicklung im AI/ML-Workflow.



Für das Management von End-to-End-KI/ML-Projekten ist ein umfassenderes Verständnis von Unternehmenskomponenten erforderlich. Obwohl DevOps die Definition, Integration und Implementierung dieser Typen von Komponenten übernommen hat, zielen Machine-Learning-Vorgänge auf einen ähnlichen Flow ab, der KI/ML-Projekte umfasst. Eine Vorstellung der Merkmale einer End-to-End-KI/ML-Pipeline im Unternehmen erhalten Sie in der folgenden Liste der erforderlichen Komponenten:

- Storage
- Netzwerkbetrieb
- Datenbanken
- File-Systeme
- Container
- CI/CD-Pipeline (Continuous Integration und Continuous Deployment)
- Integrierte Entwicklungsumgebung (IDE)
- Sicherheit
- Datenzugriffsrichtlinien
- Trennt
- Cloud
- Einheitliche
- Toolsets und Bibliotheken für Data Science

In diesem Whitepaper zeigen wir, wie die Partnerschaft zwischen NetApp und Iguazio die Entwicklung einer End-to-End-KI/ML-Pipeline deutlich vereinfacht. Dadurch wird die Markteinführungszeit für alle Ihre KI/ML-Applikationen verkürzt.

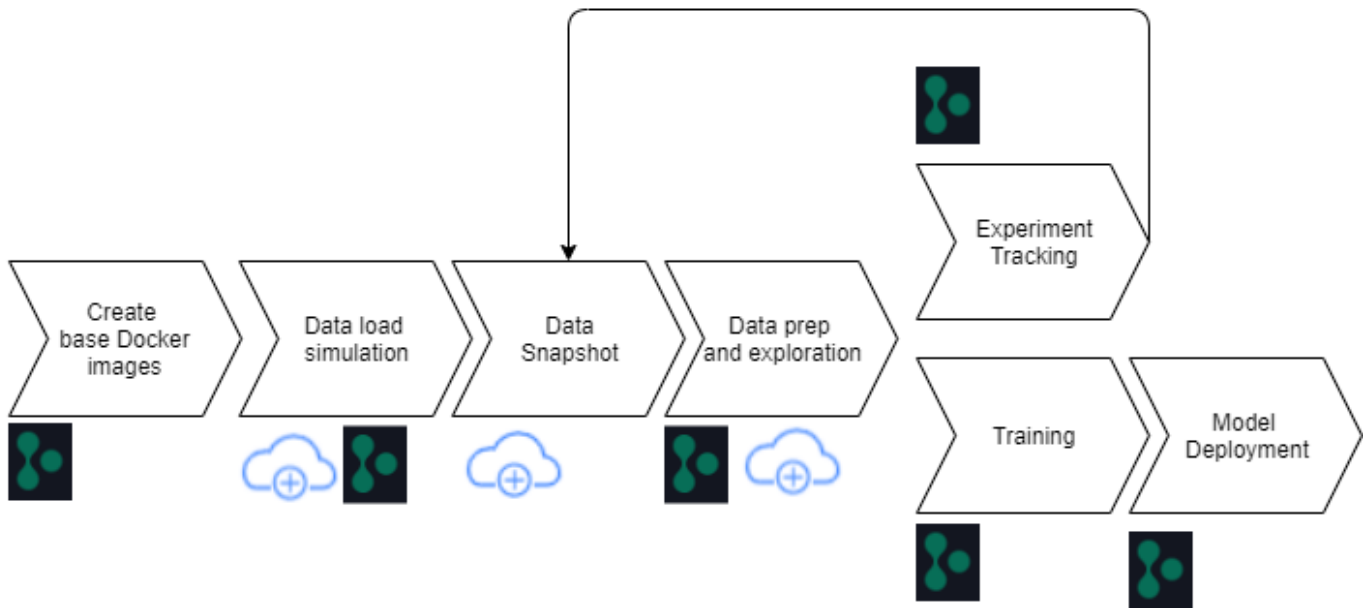
Zielgruppe

Die Welt der Datenwissenschaft berührt mehrere Disziplinen in der Informationstechnologie und im Business.

- Der Data Scientist benötigt die Flexibilität, ihre Tools und Bibliotheken einzusetzen.
- Der Data Engineer muss wissen, wie die Daten fließen und wo sie sich befinden.
- DevOps-Engineers benötigen die Tools, um neue KI/ML-Applikationen in ihre CI/CD-Pipelines zu integrieren.
- Geschäftsanwender möchten auf KI-/ML-Applikationen zugreifen können. Wir beschreiben, wie NetApp und Iguazio jede dieser Rollen unterstützen, mit unseren Plattformen einen Mehrwert für das Unternehmen zu schaffen.

Lösungsüberblick

Diese Lösung folgt dem Lebenszyklus einer KI/ML-Applikation. Zunächst legen wir die Arbeit der Datenanalysten fest, welche Schritte für die Datenvorbereitung und das Training und die Implementierung der Modelle erforderlich sind. Wir befolgen die erforderlichen Arbeiten, um eine vollständige Pipeline zu erstellen mit der Möglichkeit, Artefakte zu verfolgen, mit der Durchführung zu experimentieren und sich nach Kubeflow zu implementieren. Um den gesamten Zyklus abzuschließen, integrieren wir die Pipeline mit NetApp Cloud Volumes, um die Datenversionierung zu ermöglichen, wie im folgenden Bild zu sehen ist.



Technologischer Überblick

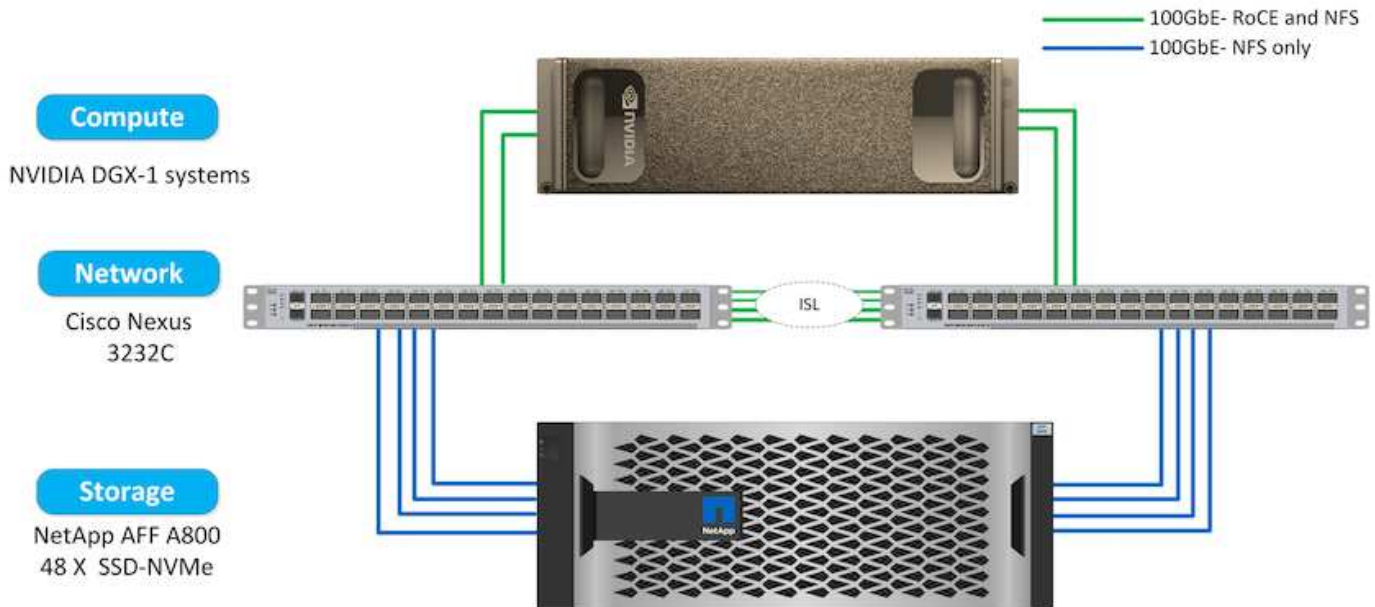
Übersicht über NetApp

NetApp ist die Instanz für Daten in der Hybrid Cloud. Mit einem Portfolio an Hybrid Cloud Data Services, die das Management von Applikationen und Daten über Cloud- und On-Premises-Umgebungen hinweg vereinfachen, beschleunigt NetApp den digitalen Wandel. Gemeinsam mit Partnern hilft NetApp Unternehmen weltweit, das volle Potenzial ihrer Daten auszuschöpfen und so Touchpoints zu Kunden auszubauen, Innovationen voranzutreiben und Betriebsabläufe zu optimieren.

NetApp ONTAP AI

Mit NetApp ONTAP AI mit Unterstützung von NVIDIA DGX Systemen und NetApp All-Flash-Storage mit Cloud-Integration optimieren Sie den Datenfluss zuverlässig und beschleunigen Analysen, Training und Inferenz über Ihre Data-Fabric-Architektur im gesamten Datacenter (Edge-, Core- und Cloud-Bereich). Sie bietet IT-Abteilungen eine Architektur mit folgenden Vorteilen:






- Sie vereinfacht das Design
- Computing und Storage können unabhängig voneinander skaliert werden
- Kunden können mit einer kleinen Konfiguration starten und nahtlos skalieren
- Bietet eine Reihe von Storage-Optionen für verschiedene Performance- und Kostennutzung
NetApp ONTAP AI bietet konvergente Infrastruktur-Stacks mit NVIDIA DGX-1, einem KI-System im Petaflop-Bereich und NVIDIA Mellanox High-Performance-Ethernet-Switches, um KI-Workloads zu vereinheitlichen, die Implementierung zu vereinfachen und den ROI zu beschleunigen. Für diesen technischen Bericht haben wir ONTAP AI mit einem DGX-1 und einem NetApp AFF A800 Storage-System genutzt. Das folgende Bild zeigt die Topologie von ONTAP AI mit dem in dieser Validierung verwendeten DGX-1-System.








NetApp AI Control Plane

Die NetApp AI Control Plane ermöglicht Ihnen die optimale Nutzung von KI- und ML-Applikationen mit einer Lösung, die höchste Skalierbarkeit, optimierte Implementierung und ununterbrochene Datenverfügbarkeit bietet. Die AI Control Plane Lösung integriert Kubernetes und Kubeflow in die Data-Fabric-Strategie von NetApp. Kubernetes, die standardmäßige Container-Orchestrierungsplattform der Branche für Cloud-native Bereitstellungen, bietet Skalierbarkeit und Portabilität von Workloads. Kubeflow ist eine Open-Source Machine Learning-Plattform, die Management und Bereitstellung vereinfacht und Entwicklern ermöglicht, mehr Datenwissenschaft in weniger Zeit zu leisten. Eine Data-Fabric-Strategie von NetApp bietet Datenverfügbarkeit und -Portabilität ohne Kompromisse. So wird sichergestellt, dass Ihre Daten in der gesamten Pipeline (Edge, Core, Cloud) aufgerufen werden können. Dieser technische Bericht verwendet die NetApp AI Control Plane in einer MLRun-Pipeline. Das folgende Bild zeigt die Kubernetes-Cluster-Managementseite, auf der Sie unterschiedliche Endpunkte für jedes Cluster haben können. Wir haben persistente NFS-Volumes mit dem Kubernetes-Cluster verbunden. Die folgenden Images zeigen ein persistentes Volume, das mit dem Cluster verbunden ist "NetApp Trident" Unterstützung für persistenten Storage und Datenmanagementfunktionen

4 Kubernetes Clusters

 kubernetes			
 https://3.20.111.39:6443 Cluster Endpoint	 v1.15.5 Cluster Version	 19.07.1 Trident Version	 0 Working Environments

 kubernetes			
 https://172.31.14.31:6443 Cluster Endpoint	 v1.15.5 Cluster Version	 19.07.1 Trident Version	 1 Working Environments

Persistent Volumes for Kubernetes

Connected with Kubernetes Cluster

Cloud Volumes ONTAP is connected to 1 Kubernetes cluster. [View Cluster](#)

You can connect another Kubernetes cluster to this Cloud Volumes ONTAP system. If the Kubernetes cluster is in a different network than Cloud Volumes ONTAP, specify a custom export policy to provide access to clients.

Kubernetes Cluster

Custom Export Policy *(Optional)*

Select Kubernetes Cluster

Custom Export Policy

Set as default storage class

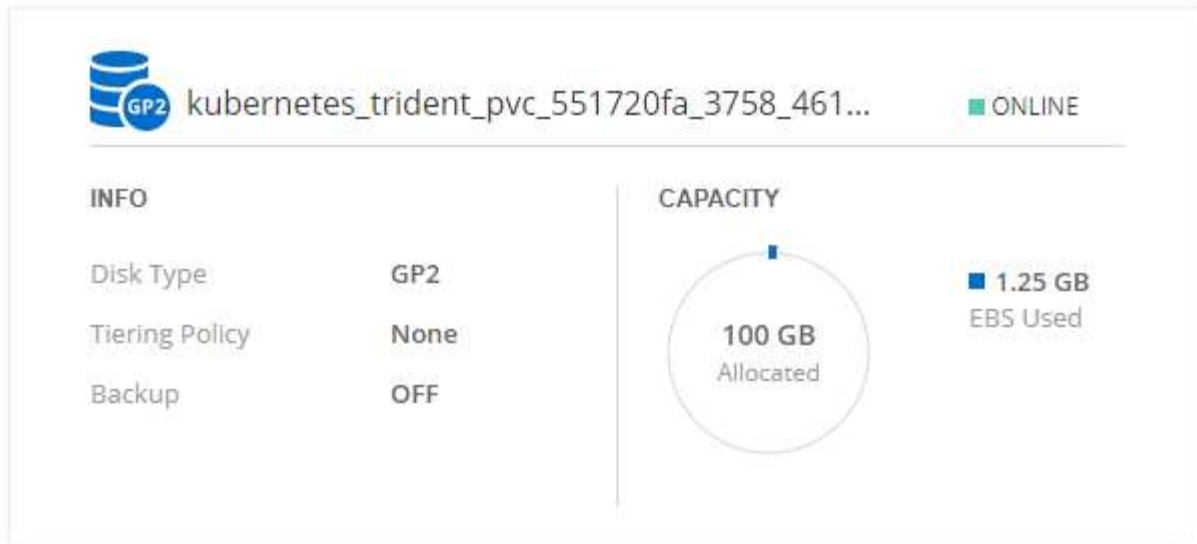
- NFS
 iSCSI

Connect

Cancel

Volumes

4 Volumes | 300 GB Allocated | 1.43 GB Total Used



The screenshot shows the details of an Amazon EBS volume. At the top, there is a blue icon representing a database and the text 'kubernetes_trident_pvc_551720fa_3758_461...' followed by a green 'ONLINE' status indicator. Below this, there are two main sections: 'INFO' and 'CAPACITY'. The 'INFO' section contains a table with the following data:

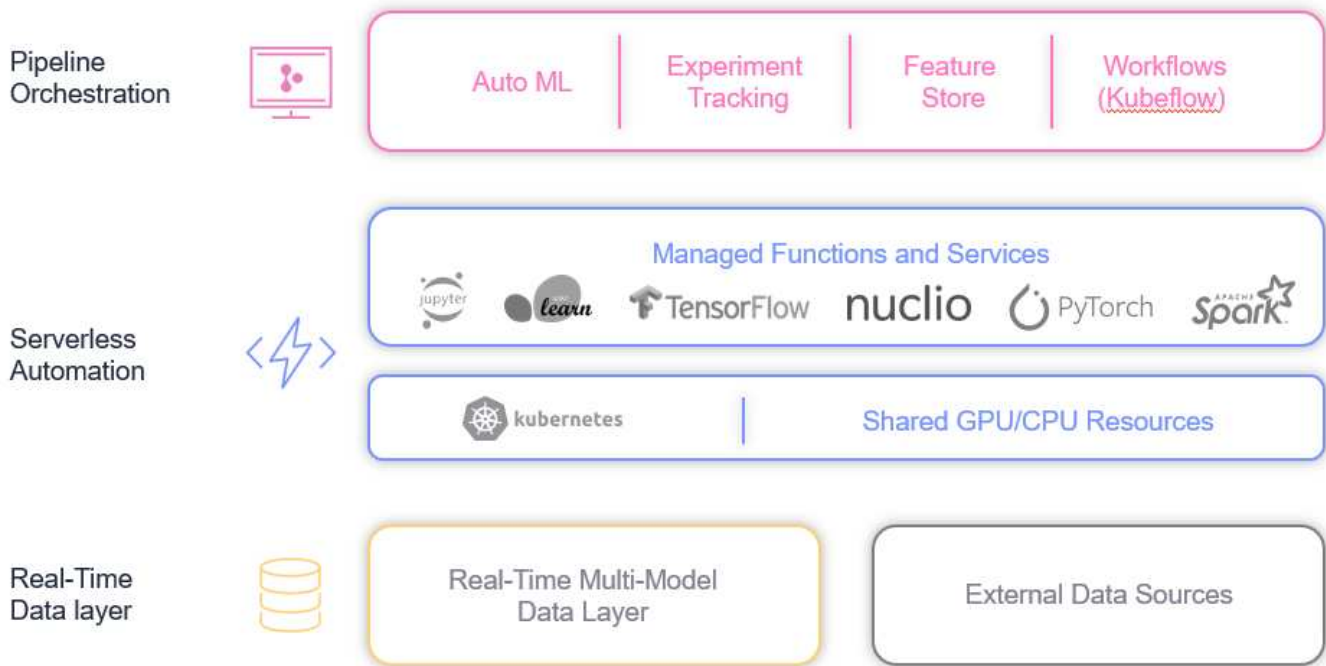
INFO	
Disk Type	GP2
Tiering Policy	None
Backup	OFF

The 'CAPACITY' section features a circular progress indicator showing '100 GB Allocated' and '1.25 GB EBS Used'.

Iguazio Übersicht

Die Iguazio Data Science Platform ist eine vollständig integrierte und sichere Plattform für Data-Science-Dienste (PaaS), die die Entwicklung vereinfacht, die Performance beschleunigt, die Zusammenarbeit erleichtert und betriebliche Herausforderungen bewältigt. Diese Plattform umfasst die folgenden Komponenten, und die Iguazio Data Science Platform wird im folgenden Bild dargestellt:

- Eine Data-Science-Umgebung mit Jupyter Notebooks, integrierten Analyse-Engines und Python-Paketen
- Modellmanagement mit Experimentierverfolgung und automatisierten Pipeline-Funktionen
- Gemanagt von Daten und ML-Services über ein skalierbares Kubernetes-Cluster
- Nuclio, ein Framework ohne Server in Echtzeit
- Eine extrem schnelle und sichere Datenschicht, die SQL, NoSQL, Zeitreihen-Datenbanken, Dateien (einfache Objekte) und Streaming unterstützt
- Integration mit Datenquellen von Drittanbietern wie NetApp, Amazon S3, HDFS, SQL Datenbanken sowie Streaming- und Messaging-Protokollen
- Echtzeit-Dashboards auf Basis von Grafana



Software- und Hardware-Anforderungen erfüllt

Netzwerkconfiguration

Die folgende Netzwerkkonfigurationsanforderung für die Einrichtung in der Cloud:

- Das Iguazio Cluster und NetApp Cloud Volumes müssen sich in der gleichen virtuellen Private Cloud befinden.
- Der Cloud-Manager muss Zugriff auf Port 6443 auf den Iguazio-App-Nodes haben.
- In diesem technischen Bericht haben wir Amazon Web Services genutzt. Anwender haben jedoch die Möglichkeit, die Lösung bei jedem Cloud-Provider zu implementieren. für lokale Tests in ONTAP AI mit NVIDIA DGX-1 haben wir den gehosteten DNS-Service von Iguazio verwendet.

Clients müssen auf dynamisch erstellte DNS-Domänen zugreifen können. Auf Wunsch können Kunden ihr eigenes DNS verwenden.

Hardwareanforderungen

Sie können Iguazio vor Ort in Ihrem eigenen Cluster installieren. Wir haben die Lösung in NetApp ONTAP AI mit einem NVIDIA DGX-1-System verifiziert. In der folgenden Tabelle ist die Hardware aufgeführt, die zum Testen dieser Lösung verwendet wird.

Trennt	Menge
DGX-1-Systeme	1
NetApp AFF A800 System	1 Hochverfügbarkeitspaar (HA), umfasst 2 Controller und 48 NVMe-SSDs (3,8 TB oder höher)
Netzwerk-Switches der Cisco Nexus 3232C-Serie	2

In der folgenden Tabelle sind die erforderlichen Softwarekomponenten für lokale Tests aufgeführt:

Software	Version oder sonstige Informationen
NetApp ONTAP Datenmanagement-Software	9.7
Cisco NX-OS Switch-Firmware	7.0(3)I6(1)
NVIDIA DGX-BETRIEBSSYSTEM	4.4 – Ubuntu 18.04 LTS
Docker-Container-Plattform	19.03.5
Container-Version	20.01-tf1-py2
Framework für Machine Learning	TensorFlow 1.15.0
Iguazio	Version 2.8 Oder Höher
ESX Server	6.5

Die Lösung wurde mit Iguazio Version 2.5 und NetApp Cloud Volumes ONTAP für AWS vollständig getestet. Das Iguazio Cluster und die NetApp Software werden beide auf AWS ausgeführt.

Software	Version oder Typ
Iguazio	Version 2.8 Oder Höher
App-Node	M5.4xlarge
Daten-Node	I3.4xlarge

Übersicht Über Die Fehlervorhersage Für Netzwerkgeräte: Anwendungsbeispiel

Dieser Anwendungsfall basiert auf einem Iguazio-Kunden im Telekommunikationsbereich in Asien. Da 100.000 Enterprise-Kunden und jedes Jahr 125 Netzwerkausfälle verursacht haben, war es entscheidend, dass sie proaktiv Maßnahmen ergreifen und verhindern müssen, dass Netzwerkausfälle die Kunden beeinträchtigen. Diese Lösung bot den Kunden folgende Vorteile:

- Prädiktive Analysen bei Netzwerkausfällen
- Integration in ein Ticketsystem
- Proaktive Maßnahmen zur Verhinderung von Netzwerkfehlern infolge dieser Implementierung von Iguazio wurden 60 % der Ausfälle proaktiv verhindert.

Setup-Übersicht

Iguazio kann lokal oder bei einem Cloud-Provider installiert werden.

Iguazio-Installation

Die Bereitstellung kann als Service ausgeführt und von Iguazio oder vom Kunden gemanagt werden. In beiden Fällen bietet Iguazio eine Implementierungsapplikation (Provazio) für die Bereitstellung und das Management von Clustern.

Informationen zur vor-Ort-Installation finden Sie unter "[NVA-1121](#)" Für Computing-, Netzwerk- und Storage-

Einrichtung. Der vor-Ort-Einsatz von Iguazio wird von Iguazio ohne zusätzliche Kosten für den Kunden bereitgestellt. Siehe "[Auf dieser Seite](#)" Für DNS- und SMTP-Serverkonfigurationen. Die Seite für die Installation von Provazio wird wie folgt angezeigt.

✕ New System (dev)

● Installation Scenario — ● General — ● Clusters — ● Cloud

- Bare metal / virtual machines
Installs the system on bare-metal or virtual-machine instances, pre-provisioned with prerequ...
- AWS
Creates applicable compute/networking resources in AWS and installs the system on the in...
- Azure
Creates applicable compute/networking resources in Azure and installs the system on the i...
- AWS (pre-provisioned)
Installs the system on Amazon Web Services instances, manually provisioned beforehand
- Azure (pre-provisioned)
Installs the system on Microsoft Azure instances, manually provisioned beforehand

Advanced
Show advanced options in the next steps

BACK NEXT

Kubernetes-Cluster Konfigurieren

Dieser Abschnitt ist in zwei Teile für die Cloud und lokale Implementierung unterteilt.

Konfiguration Von Cloud Deployment Kubernetes

Über NetApp Cloud Manager können Sie die Verbindung zum Iguazio Kubernetes Cluster definieren. Für Trident ist Zugriff auf mehrere Ressourcen im Cluster erforderlich, damit das Volume verfügbar wird.

1. Um den Zugriff zu aktivieren, rufen Sie die Kubernetes-Konfigurationsdatei von einem der Iguazio-Knoten ab. Die Datei befindet sich unter `/home/Iguazio/.kube/config`. Laden Sie diese Datei auf Ihren Desktop herunter.
2. Gehen Sie zu Discover Cluster, um zu konfigurieren.

4 Kubernetes Clusters

The screenshot displays two Kubernetes clusters in a list. Each cluster entry includes a Kubernetes icon, the name 'kubernetes', and four key metrics: Cluster Endpoint, Cluster Version, Trident Version, and Working Environments.

Cluster Name	Cluster Endpoint	Cluster Version	Trident Version	Working Environments
kubernetes	https://3.20.111.39:6443	v1.15.5	19.07.1	0
kubernetes	https://172.31.14.31:6443	v1.15.5	19.07.1	1

3. Laden Sie die Kubernetes-Konfigurationsdatei hoch. Siehe folgendes Bild.

Upload Kubernetes Configuration File

Upload the Kubernetes configuration file (kubeconfig) so Cloud Manager can install Trident on the Kubernetes cluster.

Connecting Cloud Volumes ONTAP with a Kubernetes cluster enables users to request and manage persistent volumes using native Kubernetes interfaces and constructs. Users can take advantage of ONTAP's advanced data management features without having to know anything about it. Storage provisioning is enabled by using NetApp Trident.

Learn more about [Trident for Kubernetes](#).

Upload File

4. Trident implementieren und ein Volume dem Cluster zuweisen. Sehen Sie das folgende Bild zum Definieren und Zuweisen eines Persistent Volumes zum Iguazio-Cluster. Dieser Prozess erstellt ein Persistent Volume (PV) im Iguazio-Cluster. Bevor Sie ihn verwenden können, müssen Sie einen Persistent Volume Claim (PVC) definieren.

Persistent Volumes for Kubernetes

Connected with Kubernetes Cluster

Cloud Volumes ONTAP is connected to 1 Kubernetes cluster. [View Cluster](#) ⓘ

You can connect another Kubernetes cluster to this Cloud Volumes ONTAP system. If the Kubernetes cluster is in a different network than Cloud Volumes ONTAP, specify a custom export policy to provide access to clients.

Kubernetes Cluster

Select Kubernetes Cluster

kubernetes

Custom Export Policy *(Optional)* ⓘ

Custom Export Policy

172.31.0.0/16

Set as default storage class

NFS iSCSI

Connect

Cancel

On-Premises-Implementierung Der Kubernetes-Konfiguration

Eine lokale Installation von NetApp Trident finden Sie unter "[TR-4798](#)" Entsprechende Details. Nach der Konfiguration des Kubernetes-Clusters und der Installation von NetApp Trident können Sie Trident mit dem Iguazio Cluster verbinden, um NetApp Datenmanagementfunktionen zu aktivieren, wie beispielsweise die Erstellung von Snapshot Kopien Ihrer Daten und Modelle.

Persistent Volume Claim Definieren

1. Speichern Sie die folgende YAML in einer Datei, um ein PVC vom Typ Basic zu erstellen.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 100Gi
  storageClassName: netapp-file
```

2. Wenden Sie die YAML-Datei auf Ihr Iguazio Kubernetes-Cluster an.

```
Kubectl -n default-tenant apply -f <your yaml file>
```

Weisen Sie NetApp Volume dem Jupyter Notebook zu

Iguazio bietet verschiedene Managed Services, um Data Scientists einen vollständigen End-to-End-Stack für die Entwicklung und Implementierung von KI/ML-Applikationen bereitzustellen. Weitere Informationen zu diesen Komponenten finden Sie unter "[Iguazio Übersicht über Anwendungsdienste und -Tools](#)".

Einer der Managed Services ist Jupyter Notebook. Jeder Entwickler erhält seinen eigenen Einsatz eines Notebook-Containers mit den nötigen Ressourcen für die Entwicklung. Um ihnen Zugriff auf das NetApp Cloud Volume zu geben, können Sie das Volume ihrem Container zuweisen und die Ressourcenzuweisung, den laufenden Benutzer sowie die Einstellungen für Umgebungsvariablen für Persistent Volume Claims nutzen. Das folgende Bild ist dargestellt:

Informationen zu einer On-Premises-Konfiguration finden Sie unter "[TR-4798](#)". Bei dem Setup von Trident sind die Datenmanagementfunktionen von NetApp ONTAP aktiviert, wie z. B. die Erstellung von Snapshot Kopien Ihrer Daten oder des Modells zur Versionierung. Fügen Sie in Ihrer Trident Back-End-Konfigurationsdatei die folgende Zeile ein, um Snapshot Verzeichnisse sichtbar zu machen:

```
{
  ...
  "defaults": {
    "snapshotDir": "true"
  }
}
```

Sie müssen eine Back-End-Konfigurationsdatei von Trident im JSON-Format erstellen und dann Folgendes ausführen "[Trident-Befehl](#)" So verwenden Sie sie:

```
tridentctl create backend -f <backend-file>
```

Enabled

Inactivity window: 5m, 10m, 1h, 2h, 4h

Resources

For more information about the resource parameters, see [Kubernetes documentation](#).

The memory and CPU configurations are applied to each replica.

Memory: Request [] GB, Limit [] GB

CPU: Request [Example: 1500] millicpu, Limit [Example: 1500] millicpu

Running User *

admin Username

Flavor: Full stack without GPU

Spark: spark Create new...

Environment Variables

Create a new environment variable

Persistent Volume Claims (PVCs)

Name	Mount Path
basic	/nctapp

Add PVC

Bereitstellen der Anwendung

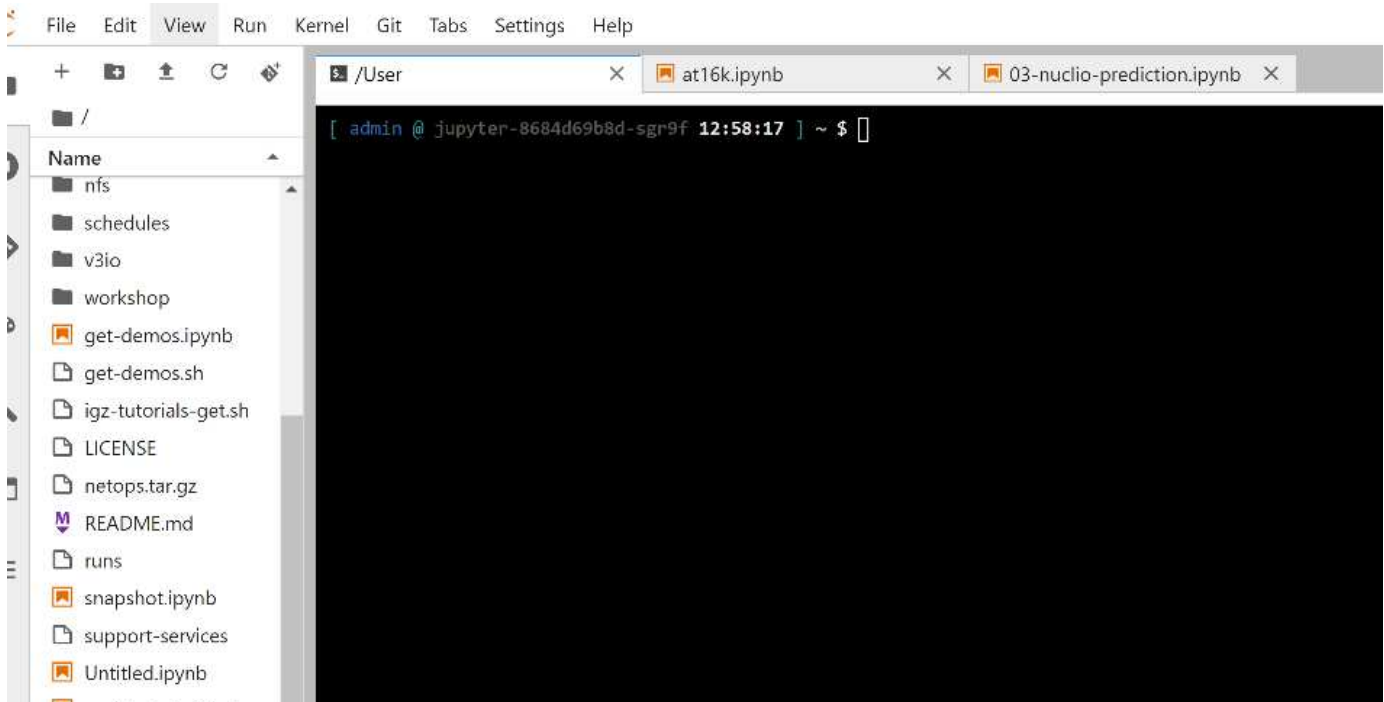
In den folgenden Abschnitten wird die Installation und Bereitstellung der Anwendung beschrieben.

Code von GitHub

Nachdem das NetApp Cloud Volume oder NetApp Trident Volume nun für den Iguazio Cluster und die Entwicklungsumgebung zur Verfügung steht, können Sie nun mit der Überprüfung der Applikation beginnen.

Benutzer haben ihren eigenen Arbeitsbereich (Verzeichnis). Auf jedem Notebook befindet sich der Pfad zum Benutzerverzeichnis `/User`. Die Iguazio-Plattform verwaltet das Verzeichnis. Wenn Sie die obigen Anweisungen befolgen, steht das NetApp Cloud Volume im zur Verfügung `/netapp` Verzeichnis.

Holen Sie sich den Code von GitHub mit einem Jupyter Terminal.



Klont das Projekt in der Jupyter-Terminalaufforderung.

```
cd /User
git clone .
```

Sie sollten jetzt sehen, die `netops-` `netapp` Ordner in der Dateistruktur im Jupyter-Arbeitsbereich.

Konfigurieren Sie Die Arbeitsumgebung

Kopieren Sie die Notebook `set_env-Example.ipynb` Als `set_env.ipynb`. Öffnen und bearbeiten `set_env.ipynb`. Dieses Notizbuch legt Variablen für

Anmeldeinformationen, Dateipositionen und Ausführungstreiber fest.

Wenn Sie die obigen Anweisungen befolgen, sind die folgenden Schritte die einzigen Änderungen, die vorgenommen werden müssen:

1. Erhalten Sie diesen Wert über das Iguazio Services Dashboard: `docker_registry`

Beispiel: `docker-registry.default-tenant.app.clusterq.iguaziodev.com:80`

2. Ändern `admin` An Ihren Iguazio-Benutzernamen:

```
IGZ_CONTAINER_PATH = '/users/admin'
```

Nachstehend finden Sie Einzelheiten zur Systemverbindung mit ONTAP. Geben Sie den Volume-Namen an, der bei der Installation von Trident generiert wurde. Die folgende Einstellung gilt für ein ONTAP Cluster vor Ort:

```
ontapClusterMgmtHostname = '0.0.0.0'
ontapClusterAdminUsername = 'USER'
ontapClusterAdminPassword = 'PASSWORD'
sourceVolumeName = 'SOURCE VOLUME'
```

Die folgende Einstellung gilt für Cloud Volumes ONTAP:

```
MANAGER=ontapClusterMgmtHostname
svm='svm'
email='email'
password=ontapClusterAdminPassword
weid="weid"
volume=sourceVolumeName
```

Basis-Docker-Images Erstellen

Alles, was Sie zum Aufbau einer ML-Pipeline benötigen, ist in der Iguazio-Plattform enthalten. Der Entwickler kann die Spezifikationen der Docker-Images festlegen, die für die Ausführung der Pipeline erforderlich sind, und die Image-Erstellung aus Jupyter Notebook ausführen. Öffnen Sie das Notebook `create-images.ipynb` Und alle Zellen ausführen.

Dieses Notizbuch erstellt zwei Bilder, die wir in der Pipeline verwenden.

- `iguazio/netapp`. Zum behandeln VON ML-Aufgaben.

Create image for training pipeline

```
[4]: fn.build_config(image=docker_registry+'/iguazio/netapp', commands=['pip install \
v3io_frames fsspec>=0.3.3 PyYAML==5.1.2 pyarrow==0.15.1 pandas==0.25.3 matplotlib seaborn yellowb
fn.deploy()'
```

- `netapp/pipeline`. Enthält Dienstprogramme zur Bearbeitung von NetApp Snapshot Kopien.

Create image for Ontap utilities

```
[0]: fn.build_config(image=docker_registry + '/netapp/pipeline:latest', commands=['apt -y update', 'pip install v3io_fraaes netapp_ontap']  
fn.deploy()
```

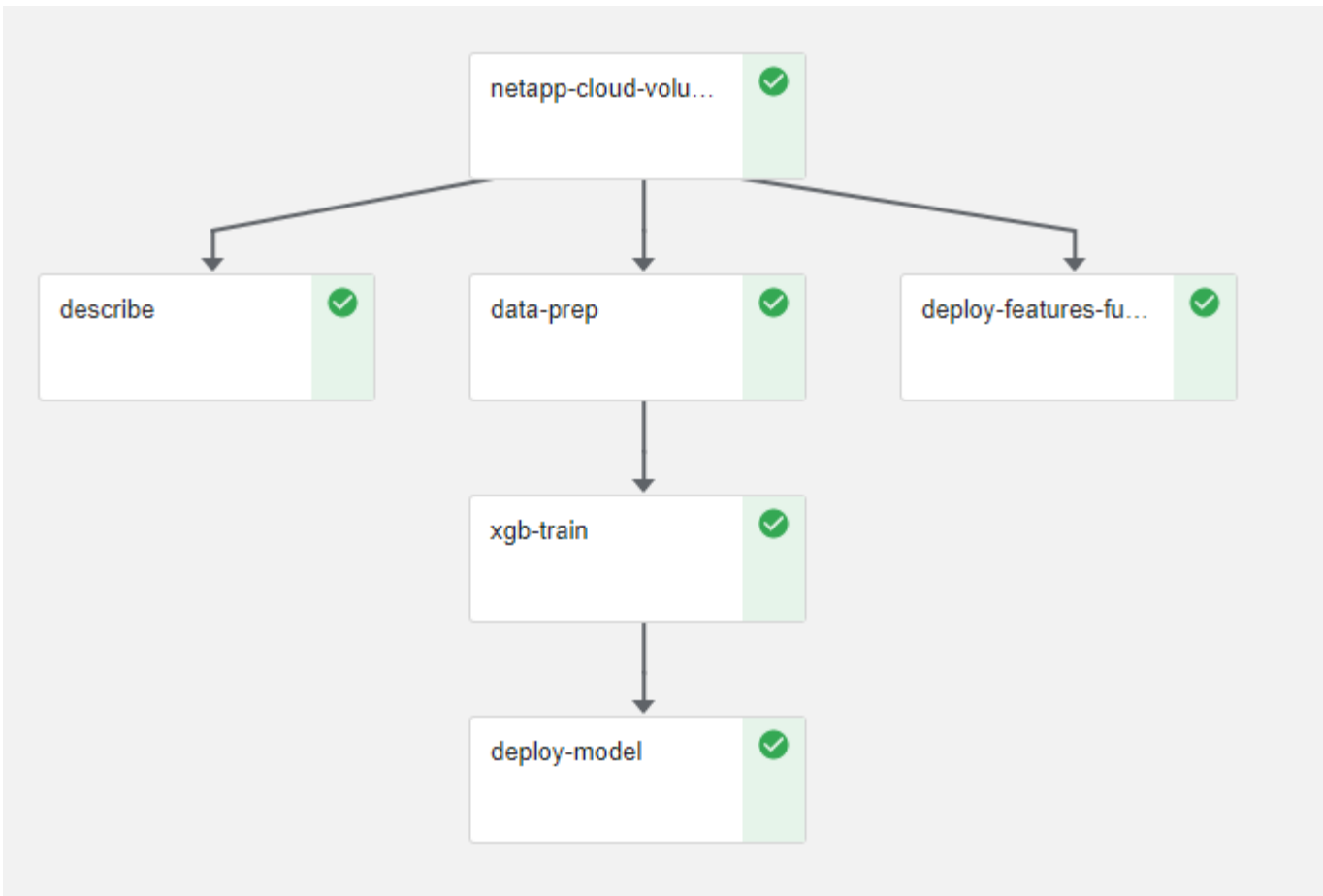
Überprüfen Sie Einzelne Jupyter Notebooks

In der folgenden Tabelle sind die Bibliotheken und Frameworks aufgeführt, mit denen wir diese Aufgabe erstellt haben. All diese Komponenten wurden vollständig in die rollenbasierten Zugriffs- und Sicherheitskontrollen von Iguazio integriert.

Bibliotheken/Framework	Beschreibung
MLRun	Ein von Iguazio verwaltet, um die Montage, Ausführung und Überwachung einer ML/AI-Pipeline zu ermöglichen.
Nuclio	Ein serverloses Funktionsgerüst, das mit Iguazio integriert ist. Auch als Open-Source-Projekt von Iguazio verfügbar.
Kubeflow	Kubernetes-basiertes Framework für die Implementierung der Pipeline Dies ist auch ein Open-Source-Projekt, zu dem Iguazio beiträgt. Es ist in Iguazio integriert für zusätzliche Sicherheit und Integration in den Rest der Infrastruktur.
Docker	Eine Docker-Registry wird als Service in der Iguazio-Plattform ausgeführt. Sie können dies auch ändern, um eine Verbindung zu Ihrer Registrierung herzustellen.
NetApp Cloud Volumes	Cloud Volumes auf AWS ermöglichen uns Zugriff auf große Datenmengen und die Möglichkeit, Snapshot Kopien mit den für das Training verwendeten Datensätzen zu erstellen.
Trident	Trident ist ein Open-Source-Projekt, das von NetApp gemanagt wird. Sie vereinfacht die Integration in Storage- und Computing-Ressourcen in Kubernetes.

Für die Erstellung der ML-Pipeline haben wir mehrere Notebooks verwendet. Jedes Notebook kann einzeln getestet werden, bevor es in die Pipeline gebracht wird. Wir decken jedes Notebook individuell nach dem Bereitstellungsablauf dieser Demonstrationsanwendung ab.

Das gewünschte Ergebnis ist eine Pipeline, mit der ein Modell auf Basis einer Snapshot-Kopie der Daten trainiert und das Modell zur Inferenz implementiert wird. Im folgenden Bild ist ein Blockdiagramm einer abgeschlossenen MLRun-Pipeline dargestellt.



Implementierung Der Funktion Zur Datengenerierung

In diesem Abschnitt wird beschrieben, wie wir Nuclio serverlose Funktionen zur Generierung von Netzwerkgerätedaten verwendet haben. Der Anwendungsfall wird von einem Iguazio-Client aus angepasst, der die Pipeline implementierte und Iguazio-Services zur Überwachung und Vorhersage von Fehlern bei Netzwerkgeräten verwendete.

Wir haben Daten von Netzwerkgeräten simuliert. Ausführen des Jupyter-Notebooks `data-generator.ipynb` Erstellt eine serverlose Funktion, die alle 10 Minuten läuft und eine Parkett-Datei mit neuen Daten erzeugt. Um die Funktion bereitzustellen, führen Sie alle Zellen in diesem Notebook aus. Siehe "[Nuclio Website](#)" Um unbekannte Komponenten in diesem Notebook zu überprüfen.

Beim Generieren der Funktion wird eine Zelle mit dem folgenden Kommentar ignoriert. Jede Zelle im Notebook wird als Teil der Funktion angenommen. Importieren Sie das Nuclio-Modul, um es zu aktivieren `%nuclio magic`.

```
# nuclio: ignore
import nuclio
```

In der Spezifikation für die Funktion definieren wir die Umgebung, in der die Funktion ausgeführt wird, wie sie ausgelöst wird und welche Ressourcen sie verbraucht.

```
spec = nuclio.ConfigSpec(config={"spec.triggers.inference.kind":"cron",
"spec.triggers.inference.attributes.interval" : "10m",
                                "spec.readinessTimeoutSeconds" : 60,
                                "spec.minReplicas" : 1},.....
```

Der `init_context` Die Funktion wird beim Initialisieren der Funktion vom Nuclio Framework aufgerufen.

```
def init_context(context):
    ...
```

Jeder Code, der nicht in einer Funktion enthalten ist, wird aufgerufen, wenn die Funktion initialisiert wird. Wenn Sie ihn aufrufen, wird eine Handler-Funktion ausgeführt. Sie können den Namen des Handlers ändern und in der Funktionsspezifikation angeben.

```
def handler(context, event):
    ...
```

Sie können die Funktion vor der Bereitstellung vom Notebook aus testen.

```
%%time
# nuclio: ignore
init_context(context)
event = nuclio.Event(body='')
output = handler(context, event)
output
```

Die Funktion kann über das Notebook bereitgestellt oder über eine CI/CD-Pipeline (Anpassung dieses Codes) bereitgestellt werden.

```
addr = nuclio.deploy_file(name='generator',project='netops',spec=spec,
tag='v1.1')
```

Pipeline-Notebooks

Diese Notizbücher sollen für dieses Setup nicht einzeln ausgeführt werden. Dies ist nur ein Bericht über jedes Notizbuch. Wir haben sie als Teil der Pipeline aufgerufen. Um sie einzeln auszuführen, lesen Sie die MLRun Dokumentation durch, um sie als Kubernetes Jobs auszuführen.

Snap_cv.ipynb

Dieses Notebook wickelt die Snapshot-Kopien des Cloud Volumes am Anfang der Pipeline ab. Er übergibt den Namen des Volumes an den Pipeline-Kontext. Dieses Notebook stößt ein Shell-Skript an, um die Snapshot

Kopie zu bearbeiten. Während der Ausführung in der Pipeline enthält der Ausführungskontext Variablen, um alle Dateien zu finden, die für die Ausführung benötigt werden. Beim Schreiben dieses Codes muss sich der Entwickler nicht um den Speicherort der Datei im Container kümmern, der ihn ausführt. Wie später beschrieben, wird diese Applikation mit allen Abhängigkeiten implementiert und ist die Definition der Pipeline-Parameter, die den Kontext der Ausführung liefern.

```
command = os.path.join(context.get_param('APP_DIR'), "snap_cv.sh")
```

Der Speicherort der erstellten Snapshot-Kopie wird im MLRun-Kontext platziert, der von Schritten in der Pipeline verwendet werden soll.

```
context.log_result('snapVolumeDetails', snap_path)
```

Die nächsten drei Notebooks werden parallel ausgeführt.

Data-prep.ipynb

RAW-Metriken müssen in Funktionen für das Modelltraining umgewandelt werden. Dieses Notebook liest die RAW-Kennzahlen aus dem Snapshot Verzeichnis und schreibt die Funktionen für das Modelltraining auf das NetApp Volume.

Bei Ausführung im Kontext der Pipeline ist der Input `DATA_DIR` Enthält den Speicherort der Snapshot Kopie.

```
metrics_table = os.path.join(str(mlruncontext.get_input('DATA_DIR',
os.getenv('DATA_DIR', '/netpp'))),
                             mlruncontext.get_param('metrics_table',
os.getenv('metrics_table', 'netops_metrics_parquet')))
```

Beschreiben.ipynb

Um die eingehenden Kennzahlen zu visualisieren, stellen wir einen Pipeline-Schritt bereit, der über die Kubeflow und MLRun Uls verfügbare Plots und Diagramme bereitstellt. Jede Ausführung hat eine eigene Version dieses Visualisierungstools.

```
ax.set_title("features correlation")
plt.savefig(os.path.join(base_path, "plots/corr.png"))
context.log_artifact(PlotArtifact("correlation", body=plt.gcf()),
local_path="plots/corr.html")
```

Deploy-Feature-function.ipynb

Wir überwachen kontinuierlich die Metriken, die nach Anomalien suchen. Dieses Notizbuch erstellt eine serverlose Funktion, die die Funktionen generiert, die für die Vorhersage von eingehenden Metriken erforderlich sind. Dieses Notizbuch ruft die Erstellung der Funktion auf. Der Funktionscode befindet sich im Notebook `data- prep.ipynb`. Beachten Sie, dass wir für diesen Zweck dasselbe Notebook wie einen Schritt in der Pipeline verwenden.

Training.ipynb

Nachdem wir die Funktionen erstellt haben, lösen wir das Modelltraining aus. Das Ergebnis dieses Schritts ist das Modell, das zur Inferenz verwendet wird. Wir sammeln auch Statistiken, um jede Ausführung (Experiment) im Auge zu behalten.

Beispielsweise gibt der folgende Befehl die Genauigkeitsbewertung in den Kontext des entsprechenden Experiments ein. Dieser Wert ist in Kubeflow und MLRun sichtbar.

```
context.log_result('accuracy', score)
```

Bereitstellen-Inferenz-Funktion.ipynb

Der letzte Schritt in der Pipeline ist die Implementierung des Modells als serverlose Funktion für kontinuierliche Inferenz. Dieses Notebook ruft die Erstellung der serverlosen Funktion auf, die in definiert ist `nuclio-inference-function.ipynb`.

Pipeline prüfen und aufbauen

Die Kombination der Ausführung aller Notebooks in einer Pipeline ermöglicht es, kontinuierlich Experimente durchzuführen, um die Genauigkeit des Modells anhand neuer Metriken neu zu bewerten. Öffnen Sie zuerst das `pipeline.ipynb` Notebook. Wir erläutern Ihnen die Details, die zeigen, wie NetApp und Iguazio die Implementierung dieser ML-Pipeline vereinfachen.

Wir verwenden MLRun, um jedem Schritt der Pipeline Kontext bereitzustellen und die Ressourcenzuordnung zu bearbeiten. Der MLRun API-Service wird auf der Iguazio-Plattform ausgeführt und dient als Interaktionsstelle mit Kubernetes-Ressourcen. Jeder Entwickler kann keine Ressourcen direkt anfordern. Die API verarbeitet die Anforderungen und ermöglicht Zugriffskontrollen.

```
# MLRun API connection definition
mlconf.dbpath = 'http://mlrun-api:8080'
```

Diese Pipeline kann mit NetApp Cloud Volumes und On-Premises-Volumes eingesetzt werden. Wir haben diese Demo für die Verwendung von Cloud Volumes erstellt. Im Code sehen Sie jedoch die Option zur Ausführung vor Ort.

```

# Initialize the NetApp snap function once for all functions in a notebook
if [ NETAPP_CLOUD_VOLUME ]:
    snapfn =
code_to_function('snap',project='NetApp',kind='job',filename="snap_cv.ipyn
b").apply(mount_v3io())
    snap_params = {
    "metrics_table" : metrics_table,
    "NETAPP_MOUNT_PATH" : NETAPP_MOUNT_PATH,
    'MANAGER' : MANAGER,
    'svm' : svm,
    'email': email,
    'password': password ,
    'weid': weid,
    'volume': volume,
    "APP_DIR" : APP_DIR
    }
else:
    snapfn =
code_to_function('snap',project='NetApp',kind='job',filename="snapshot.ipy
nb").apply(mount_v3io())
...
snapfn.spec.image = docker_registry + '/netapp/pipeline:latest'
snapfn.spec.volume_mounts =
[snapfn.spec.volume_mounts[0],netapp_volume_mounts]
    snapfn.spec.volumes = [ snapfn.spec.volumes[0],netapp_volumes]

```

Die erste Aktion, die erforderlich ist, um ein Jupyter-Notebook in einen Kubeflow-Schritt zu verwandeln, ist, den Code in eine Funktion zu verwandeln. Eine Funktion verfügt über alle Spezifikationen, die zum Ausführen dieses Notebooks erforderlich sind. Wenn Sie das Notebook nach unten scrollen, sehen Sie, dass wir für jeden Schritt in der Pipeline eine Funktion definieren.

Teil des Notebooks	Beschreibung
<Code_to_Function> (Teil des MLRun-Moduls)	Name der Funktion: Projektname. Dient zur Organisation aller Projektartefakte. Dies ist in der MLRun UI sichtbar. Freundlich. In diesem Fall ist Kubernetes-Job. Dies könnte DASK, mpi, funkk8s und mehr sein. Weitere Informationen finden Sie in der MLRun-Dokumentation. Datei: Der Name des Notebooks. Dies kann auch ein Speicherort in Git (HTTP) sein.
Bild	Der Name des Docker Images, das wir für diesen Schritt verwenden. Das haben wir früher mit dem Notebook create-image.ipynb erstellt.
Volume_Mounts und Volumes	Einzelheiten zum Mounten des NetApp Cloud Volume zur Laufzeit.

Außerdem definieren wir Parameter für die Schritte.

```
params={
    "FEATURES_TABLE":FEATURES_TABLE,
    "SAVE_TO" : SAVE_TO,
    "metrics_table" : metrics_table,
    'FROM_TSDB': 0,
    'PREDICTIONS_TABLE': PREDICTIONS_TABLE,
    'TRAIN_ON_LAST': '1d',
    'TRAIN_SIZE':0.7,
    'NUMBER_OF_SHARDS' : 4,
    'MODEL_FILENAME' : 'netops.v3.model.pickle',
    'APP_DIR' : APP_DIR,
    'FUNCTION_NAME' : 'netops-inference',
    'PROJECT_NAME' : 'netops',
    'NETAPP_SIM' : NETAPP_SIM,
    'NETAPP_MOUNT_PATH': NETAPP_MOUNT_PATH,
    'NETAPP_PVC_CLAIM' : NETAPP_PVC_CLAIM,
    'IGZ_CONTAINER_PATH' : IGZ_CONTAINER_PATH,
    'IGZ_MOUNT_PATH' : IGZ_MOUNT_PATH
}
```

Nachdem Sie die Funktionsdefinition für alle Schritte erstellt haben, können Sie die Pipeline erstellen. Wir verwenden den `kfp` Modul, um diese Definition zu erstellen. Der Unterschied zwischen der Verwendung von `MLRun` und dem Selbstaufbau besteht in der Vereinfachung und Verkürzung der Codierung.

Die von uns definierten Funktionen werden mit dem in Schrittkomponenten umgewandelt `as_step` Funktion von `MLRun`.

Definition Von Snapshot-Schritten

Initiieren einer Snapshot-Funktion, -Ausgabe und Mounten von `v3io` als Quelle:

```
snap = snapfn.as_step(NewTask(handler='handler',params=snap_params),
name='NetApp_Cloud_Volume_Snapshot',outputs=['snapVolumeDetails','training_
_parquet_file']).apply(mount_v3io())
```

Parameter	Details
NewTask	NewTask ist die Definition der Funktion Run.
(MLRun-Modul)	Handler. Name der zu aufrufe Python-Funktion. Wir haben den Namen Handler im Notebook verwendet, aber er ist nicht erforderlich. Param. Die Parameter, die wir an die Ausführung übergeben haben. Innerhalb unseres Codes verwenden wir <code>context.get_param ('PARAMETER')</code> , um die Werte zu erhalten.

Parameter	Details
As_Step	Name: Name des Kubeflow-Pipeline-Schritts. Ausgänge. Dies sind die Werte, die der Schritt dem Wörterbuch nach Abschluss hinzugefügt hat. Werfen Sie einen Blick auf das Snap_cv.ipynb Notebook. Mount_v3io(). Hiermit wird der Schritt zum Mounten von /Benutzer für den Benutzer konfiguriert, der die Pipeline ausführt.

```

prep = data_prep.as_step(name='data-prep',
handler='handler',params=params,
                        inputs = {'DATA_DIR':
snap.outputs['snapVolumeDetails']}) ,

out_path=artifacts_path).apply(mount_v3io()).after(snap)

```

Parameter	Details
Eingänge	Sie können die Ausgänge eines vorherigen Schritts an einen Schritt übergeben. In diesem Fall ist Snap.Outputs ['snapVolumeDetails'] der Name der Snapshot Kopie, die wir im Snap Schritt erstellt haben.
Out_PATH	Ein Speicherort für Artefakte, die mithilfe des MLRun-Moduls log_Artefakte erzeugt werden.

Sie können laufen `pipeline.ipynb` Von oben nach unten. Anschließend können Sie im Iguazio-Dashboard zur Registerkarte Pipelines wechseln, um den Fortschritt zu überwachen, wie auf der Registerkarte Iguazio-Dashboard-Pipelines zu sehen ist.

The screenshot shows the 'Pipelines' section of a software interface. On the left is a dark sidebar with icons for Pipelines, Projects, and Services. The main area is titled 'Pipelines' and shows a breadcrumb 'Experiments > NetAppXGB'. Below this is a navigation bar with 'Graph', 'Run output', and 'Config' tabs. The 'Graph' tab is active, displaying a pipeline graph. A box labeled 'netapp-cloud-volu...' has arrows pointing to two steps: 'describe' and 'data-prep'. The 'describe' step has a green checkmark icon next to it, indicating it is completed or successful.

Da wir die Genauigkeit des Trainingschritts in jedem Lauf protokolliert haben, haben wir für jedes Experiment eine Aufzeichnung der Genauigkeit, wie in der Aufzeichnung der Trainingsgenauigkeit zu sehen ist.

<input type="checkbox"/>	Run name	Status	Duration	Pipeline Version	Recurring ...	Start time	accuracy
<input type="checkbox"/>	xgb_pipeline 2020-03-24 18-51-...	✓	0:08:43	[View pipeline]	-	3/24/2020, 2:51:09 PM	0.985
<input type="checkbox"/>	xgb_pipeline 2020-03-19 13-31-...	✓	0:08:14	[View pipeline]	-	3/19/2020, 9:31:19 AM	0.980
<input type="checkbox"/>	xgb_pipeline 2020-03-18 12-56-...	✓	0:08:11	[View pipeline]	-	3/18/2020, 8:56:08 AM	0.990
<input type="checkbox"/>	xgb_pipeline 2020-03-17 19-49-...	✓	0:08:03	[View pipeline]	-	3/17/2020, 3:49:31 PM	0.985
<input type="checkbox"/>	xgb_pipeline 2020-03-17 18-34-...	✓	0:05:54	[View pipeline]	-	3/17/2020, 2:34:56 PM	0.980
<input type="checkbox"/>	xgb_pipeline 2020-03-17 17-34-...	✓	0:04:48	[View pipeline]	-	3/17/2020, 1:34:16 PM	0.982
<input type="checkbox"/>	xgb_pipeline 2020-03-17 17-01-...	✓	0:05:25	[View pipeline]	-	3/17/2020, 1:01:58 PM	0.987
<input type="checkbox"/>	xgb_pipeline 2020-03-16 16-47-...	✓	0:06:08	[View pipeline]	-	3/16/2020, 12:47:19 ...	0.983
<input type="checkbox"/>	xgb_pipeline 2020-03-16 13-57-...	✓	0:05:18	[View pipeline]	-	3/16/2020, 9:57:03 AM	0.980

Wenn Sie den Schritt Snapshot auswählen, wird der Name der Snapshot Kopie angezeigt, die zur Ausführung dieses Experiments verwendet wurde.

netops-trainign-pipeline-with-netapp-volume-cloning-rtxdl-2910983943

Artifacts **Input/Output** Volumes Manifest Logs

input artifacts

Output parameters

netapp-cloud-volume-snapshot-snapVolumeDetails	/netapp/snapshot/kfp_20200324_185122
netapp-cloud-volume-snapshot-training_parquet_file	/netapp/snapshot/kfp_20200324_18512...

Output artifacts

Der beschriebene Schritt hat visuelle Artefakte, um die von uns verwendeten Metriken zu untersuchen. Sie können erweitern, um die vollständige Darstellung wie im folgenden Bild zu sehen.

netops-trainign-pipeline-with-netapp-volume-cloning-rtxdl-2

Artifacts **Input/Output** Volumes Manifest Logs

Static HTML

Class Balance for 48,008

40000

Die MLRun API-Datenbank verfolgt auch Eingänge, Ausgänge und Artefakte für jeden nach Projekt organisierten Durchlauf. Ein Beispiel für Eingänge, Ausgänge und Artefakte für jeden Durchlauf ist im folgenden Bild zu sehen.

Projects

The screenshot shows the 'Projects' section of the MLRun UI. It features a sidebar with a home icon and a main area with three project cards: 'NetApp', 'default', and 'describe'. Each card displays a green checkmark icon for 'Jobs' and a document icon for 'Artifacts'.

Für jede Aufgabe werden zusätzliche Details gespeichert.

Name	
deploy-model ● 24 Mar, 14:56:03 ...bcbe38e	
xgb_train ● 24 Mar, 14:53:18 ...5c85949	
data-prep ● 24 Mar, 14:52:46 ...126dc73	
describe ● 24 Mar, 14:52:45 ...c2a460e	<h2>describe</h2> <p>24 Mar, 14:52:45 ●</p> <p>Info Inputs Artifacts Results Logs</p> <hr/> <p>UID 66ef22187efb4ad89e8da8433c2a460e</p> <hr/> <p>Start time 24 Mar, 14:52:45</p> <hr/> <p>Parameters Completed ●</p> <hr/> <p>Results class_label... <input type="text"/> key: summary label_colu... <input type="text"/></p>
deploy-features-function ● 24 Mar, 14:52:43 ...50d8b83	
NetApp_Cloud_Volume_Sna ● 24 Mar, 14:51:22 ...3108eb2	

Es gibt mehr Informationen über MLRun, als wir in diesem Dokument abdecken können. AI Artefakte, einschließlich der Definition der Schritte und Funktionen, können in der API-Datenbank gespeichert, versioniert und einzeln oder als volles Projekt aufgerufen werden. Projekte können auch gespeichert und zur späteren Verwendung an Git geschoben werden. Wir empfehlen Ihnen, weitere Informationen im zu erhalten "[MLRun GitHub-Website](#)".

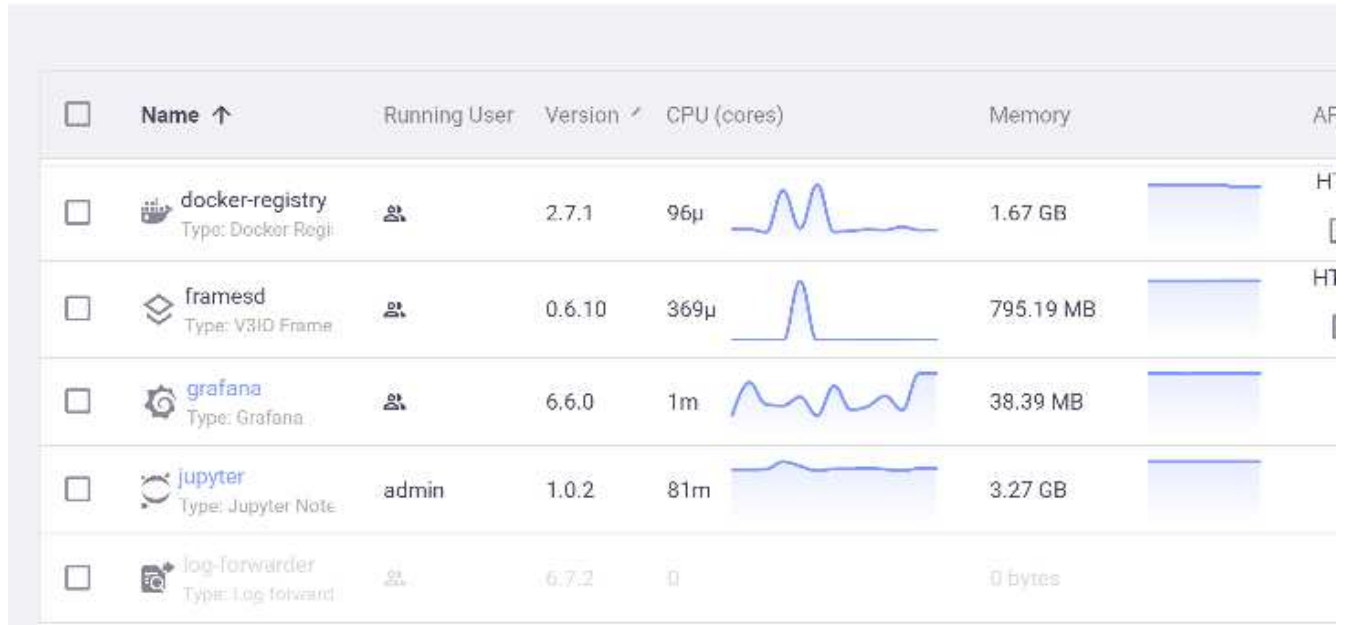
Implementierung Von Grafana Dashboard








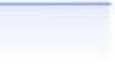











Wenn alles bereitgestellt ist, kommen wir Rückschlüsse auf neue Daten. Die Modelle prognostizieren den Ausfall der Netzwerkgeräte. Die Ergebnisse der Vorhersage werden in einer Iguazio-Timeseries-Tabelle gespeichert. Sie können die Ergebnisse mit Grafana in der in die Sicherheits- und Datenzugriffsrichtlinie von Iguazio integrierten Plattform visualisieren.

Sie können das Dashboard bereitstellen, indem Sie die bereitgestellte JSON-Datei in die Grafana-Schnittstellen im Cluster importieren.

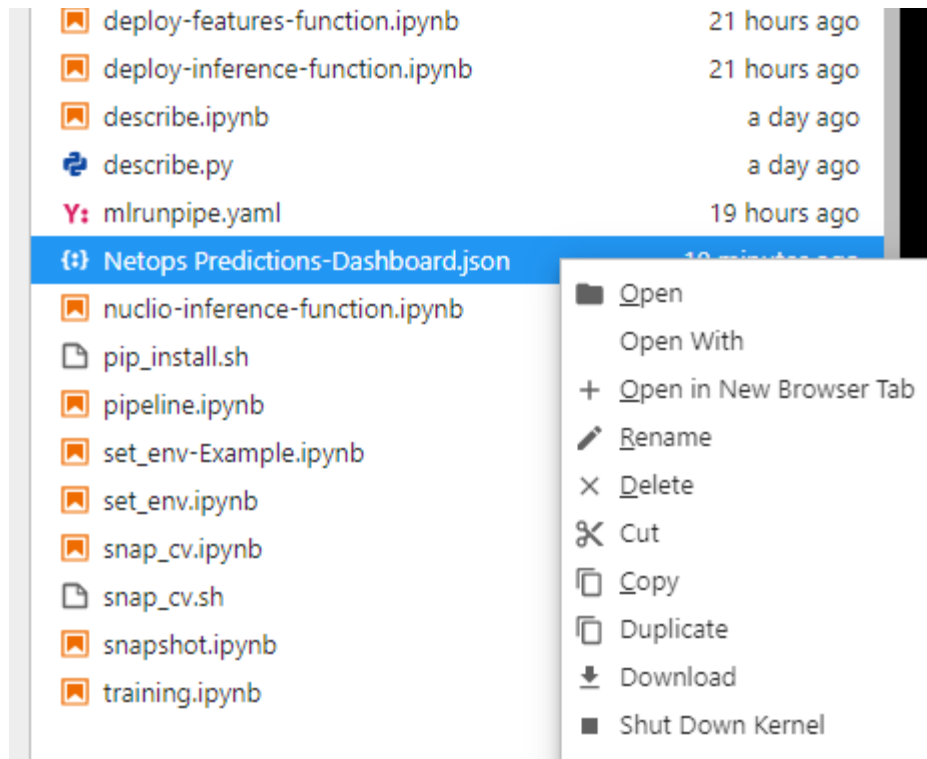
1. Um zu überprüfen, ob der Grafana-Service ausgeführt wird, suchen Sie unter Services.

Services

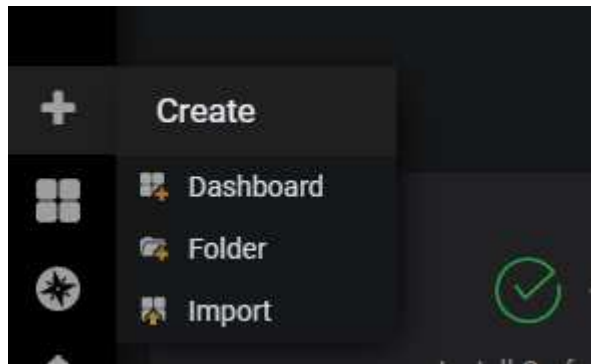


<input type="checkbox"/>	Name ↑	Running User	Version ↕	CPU (cores)	Memory	AF
<input type="checkbox"/>	 docker-registry Type: Docker Regi		2.7.1	96μ 	1.67 GB 	HT
<input type="checkbox"/>	 framesd Type: V3ID Frame		0.6.10	369μ 	795.19 MB 	HT
<input type="checkbox"/>	 grafana Type: Grafana		6.6.0	1m 	38.39 MB 	
<input type="checkbox"/>	 jupyter Type: Jupyter Note	admin	1.0.2	81m 	3.27 GB 	
<input type="checkbox"/>	 log-forwarder Type: Log forward		6.7.2	0 	0 bytes 	

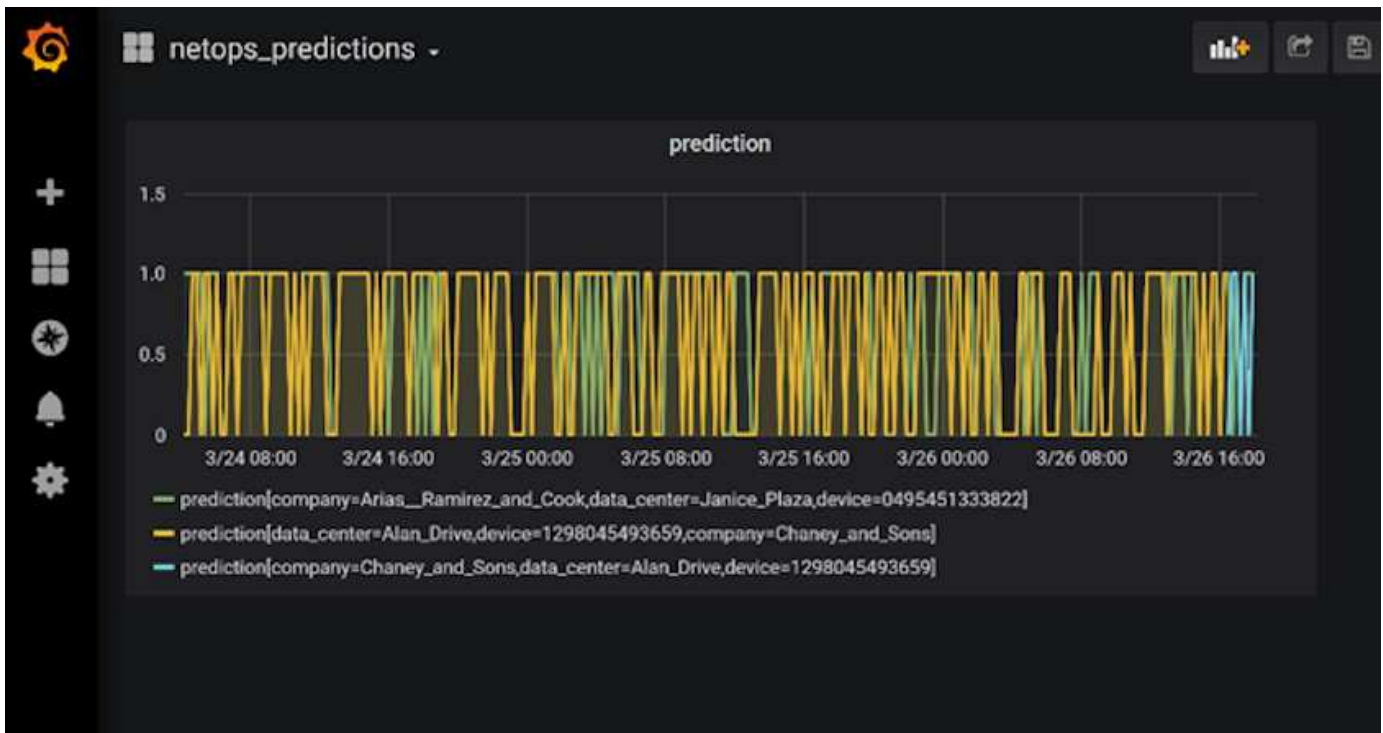
2. Wenn diese nicht vorhanden ist, stellen Sie eine Instanz im Abschnitt „Services“ bereit:
 - a. Klicken Sie Auf Neuer Dienst.
 - b. Wählen Sie Grafana aus der Liste aus.
 - c. Akzeptieren Sie die Standardeinstellungen.
 - d. Klicken Sie Auf Weiter.
 - e. Geben Sie Ihre Benutzer-ID ein.
 - f. Klicken Sie Auf Dienst Speichern.
 - g. Klicken Sie oben auf Änderungen übernehmen.
3. Laden Sie die Datei herunter, um das Dashboard bereitzustellen NetopsPredictions-Dashboard.json Über die Jupyter-Schnittstelle.



4. Öffnen Sie Grafana im Abschnitt Services, und importieren Sie das Dashboard.



5. Klicken Sie Auf Hochladen *.json Datei und wählen Sie die Datei aus, die Sie zuvor heruntergeladen haben (NetopsPredictions-Dashboard.json). Das Dashboard wird nach Abschluss des Uploads angezeigt.



Funktion „Bereinigen“ Bereitstellen

Wenn Sie eine Menge Daten generieren, ist es wichtig, die Dinge sauber und organisiert zu halten. Stellen Sie dazu die Bereinigungsfunktion mit dem bereit `cleanup.ipynb` Notebook.

Vorteile

NetApp und Iguazio beschleunigen und vereinfachen die Implementierung von KI- und ML-Applikationen. Dazu bauen sie in wichtigen Frameworks wie Kubeflow, Apache Spark und TensorFlow sowie Orchestrierungs-Tools wie Docker und Kubernetes auf. Durch die Vereinheitlichung der End-to-End-Datenpipeline reduzieren NetApp und Iguazio die Latenz und Komplexität moderner Computing-Workloads, wodurch die Lücke zwischen Entwicklung und Betrieb geschlossen wird. Data Scientists können Abfragen zu großen Datensätzen durchführen und Daten und algorithmische Modelle während der Trainingsphase sicher mit autorisierten Benutzern teilen. Sobald die Container-Modelle für die Produktion bereit sind, können Sie sie ganz einfach von Entwicklungsumgebungen in betriebliche Umgebungen verschieben.

Schlussfolgerung

Bei der Erstellung eigener KI/ML-Pipelines ist die Konfiguration der Integration, Verwaltung, Sicherheit und Zugänglichkeit der Komponenten einer Architektur eine schwierige Aufgabe. Der Zugriff auf und die Kontrolle über ihre Umgebung für Entwickler stellt eine weitere Herausforderung dar.

Die Kombination aus NetApp und Iguazio vereint diese Technologien als Managed Services, um die Technologieeinführung zu beschleunigen und die Markteinführungszeit für neue KI/ML-Applikationen zu verkürzen.

Copyright-Informationen

Copyright © 2024 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFT SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.