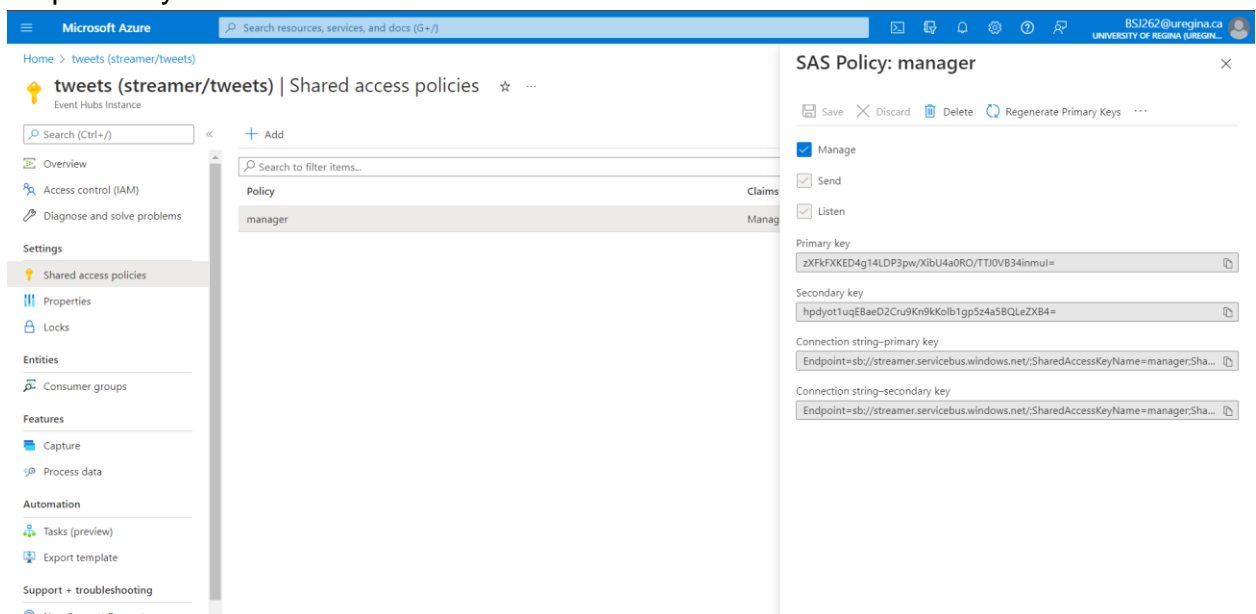


Appendices

Azure Event hub

1. Follow the steps below to create an eventhub <https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-create>
2. After creating the eventhub go to shared access policies and create a policy with manager access.
3. The primary/secondary connection string generated after creating the policy is used for both producing and consuming the data in the streamer and notebooks respectively.



Kubernetes Service

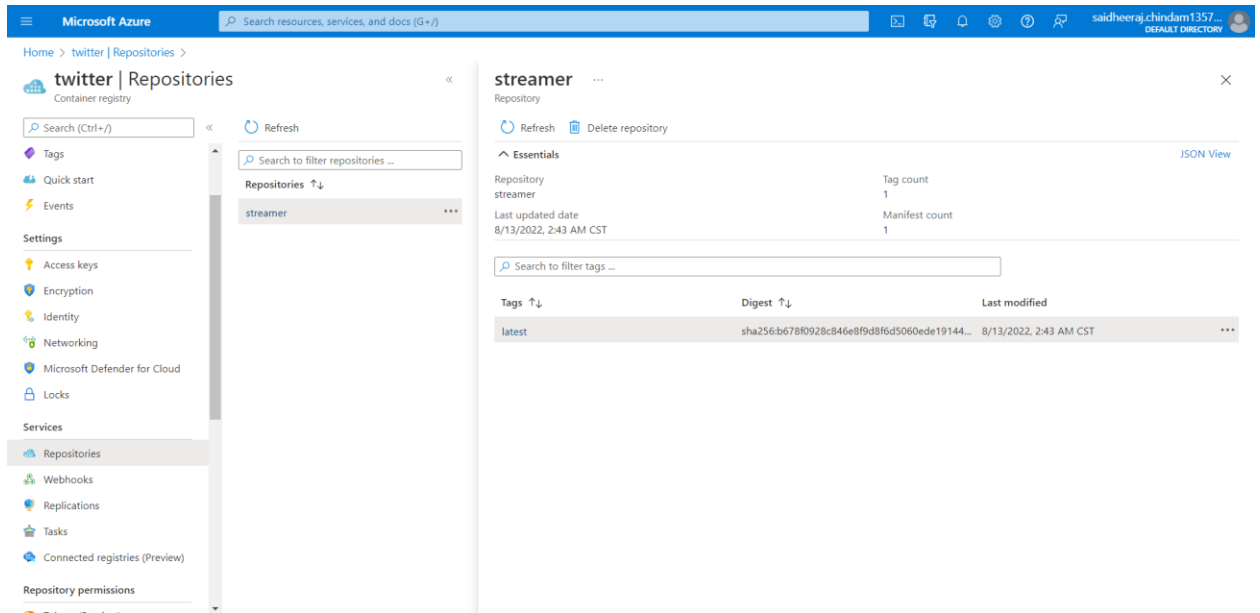
1. Follow the steps below to create an AKS cluster until the nodes are up and running with the change of adding the container registry that you created in the integrations <https://docs.microsoft.com/en-us/azure/aks/learn/quick-kubernetes-deploy-portal?tabs=azure-cli>
2. Copy the streamer.yml file to the Azure Cloud Shell and apply the file using the command, before applying change the connection string and name in the environment variable

kubectl apply -f streamer.yml

The screenshot displays the Microsoft Azure portal interface for a Kubernetes service named 'streamer'. The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, and Microsoft Defender for Cloud. The main content area is divided into 'Essentials' and 'Properties' tabs. The 'Essentials' tab shows key information: Resource group (DataBricksRG), Status (Succeeded (Running)), Location (Canada Central), Subscription (Azure for Students), and Subscription ID (12e65f67-3150-4ffe-93c9-cab9e84c5373). The 'Properties' tab is active, showing details for 'Kubernetes services' (Encryption type: Encryption at-rest with a platform-managed key, Virtual node pools: Not enabled), 'Node pools' (1 node pool, Kubernetes versions: 1.22.11, Node sizes: Standard_DS2_v2), and 'Configuration' (Kubernetes version: 1.22.11, Authentication and authorization: Local accounts with Kubernetes RBAC). A 'Networking' section on the right lists details like API server address, Network type (Kubenet), Pod CIDR, Service CIDR, DNS service IP, Docker bridge CIDR, Network Policy, Load balancer, HTTP application routing, and Private cluster status.

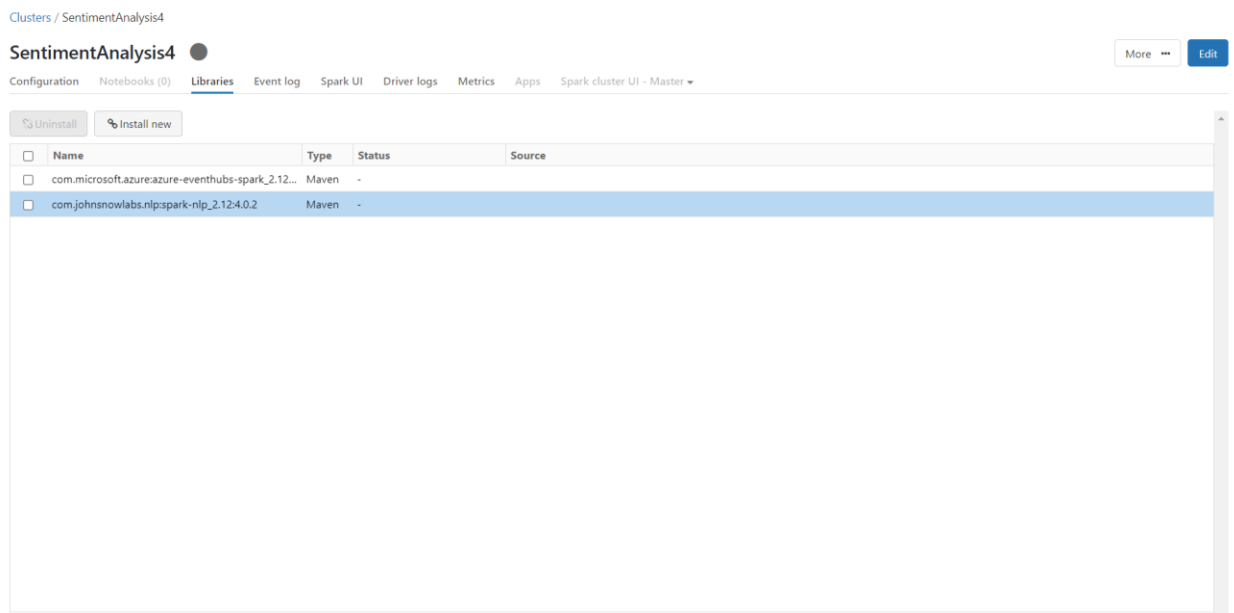
Container Registry

1. Follow the steps below just until the creation of registry
<https://docs.microsoft.com/en-us/azure/container-registry/container-registry-get-started-portal?tabs=azure-cli>
2. Go to access keys and enable admin-user
3. Use the following command with the username and password displayed on the portal to login to the registry, the first word will change depending on the name you provided
docker login twitter.azurecr.io
4. Build the streamer image using the command
docker build . -t twitter.azurecr.io/streamer:latest
5. Push the image using the command
docker push twitter.azurecr.io/streamer:latest



Databricks

1. Follow the steps below until the databricks cluster is created.
2. Go to libraries and with library source as maven install the packages on the following Maven coordinates one by one
com.microsoft.azure:azure-eventhubs-spark_2.12:2.3.22
com.johnsnowlabs.nlp:spark-nlp_2.12:4.0.2



3. The databricks should now be able to run the notebooks, make sure to change the configurations for things like storage account and event hub connection strings.

Azure Storage Account

1. Use the steps below to create a Storage Account <https://docs.microsoft.com/en-us/azure/storage/common/storage-account-create?tabs=azure-portal>
2. Use the steps below to create a container <https://docs.microsoft.com/en-us/azure/storage/blobs/storage-quickstart-blobs-portal#create-a-container>
3. Use the access key from the access keys tab to configure the notebooks

Microsoft Azure

Home > twittergenstorage | Containers >

realtimetwitterdata

Container

Search (Ctrl+J)

Upload Change access level Refresh Delete Change tier Acquire lease Break lease View snapshots Create snapshot

Authentication method: Access key (Switch to Azure AD User Account)

Location: realtimetwitterdata / models

Search blobs by prefix (case-sensitive) Show deleted blobs

Add filter

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
<input type="checkbox"/> [-]						---
<input type="checkbox"/> decision-tree						---
<input type="checkbox"/> decision-tree-2						---
<input type="checkbox"/> logistic-regression						---
<input type="checkbox"/> logistic-regression-hash						---
<input type="checkbox"/> RandomForestHash						---
<input type="checkbox"/> decision-tree	8/11/2022, 7:30:17 PM	Hot (Inferred)		Block blob	0 B	Available
<input type="checkbox"/> decision-tree-2	8/14/2022, 4:30:08 AM	Hot (Inferred)		Block blob	0 B	Available
<input type="checkbox"/> logistic-regression	8/14/2022, 2:12:44 PM	Hot (Inferred)		Block blob	0 B	Available
<input type="checkbox"/> logistic-regression-hash	8/14/2022, 2:54:25 PM	Hot (Inferred)		Block blob	0 B	Available
<input type="checkbox"/> RandomForestHash	8/14/2022, 3:28:37 PM	Hot (Inferred)		Block blob	0 B	Available