



NetApp のファイルオブジェクト二重性と
AWS SageMaker
を使用したクラウドデータ管理
NetApp artificial intelligence solutions

NetApp
August 18, 2025

目次

NetApp のファイルオブジェクト二重性と AWS SageMaker を使用したクラウドデータ管理	1
TR-4967: NetAppファイル・オブジェクト・デュアリティと AWS SageMaker によるクラウドデータ管理	1
ソリューション技術	1
ユースケースの概要	2
データサイエンティストやその他のアプリケーションのためのデータの二重性	2
技術要件	2
展開手順	3
Jupyter Notebook から機械学習を検証する	17
まとめ	29
詳細情報の入手方法	30

NetApp のファイルオブジェクト二重性と AWS SageMaker を使用したクラウドデータ管理

TR-4967: NetApp ファイル・オブジェクト・デュアリティと AWS SageMaker によるクラウドデータ管理

カーティケヤン ナガリングム、NetApp

データサイエンティストやエンジニアは、NFS 形式で保存されたデータにアクセスする必要がありますが、AWS は S3 バケット アクセスのみをサポートしているため、AWS SageMaker の S3 プロトコルからこのデータに直接アクセスするのは難しい場合があります。ただし、NetApp ONTAP は、NFS と S3 のデュアルプロトコルアクセスを有効にすることでソリューションを提供します。このソリューションにより、データサイエンティストやエンジニアは、NetApp Cloud Volumes ONTAP の S3 バケットを介して AWS SageMaker ノートブックの NFS データにアクセスできるようになります。このアプローチにより、追加のソフトウェアを必要とせずに、NFS と S3 の両方から同じデータに簡単にアクセスして共有できるようになります。

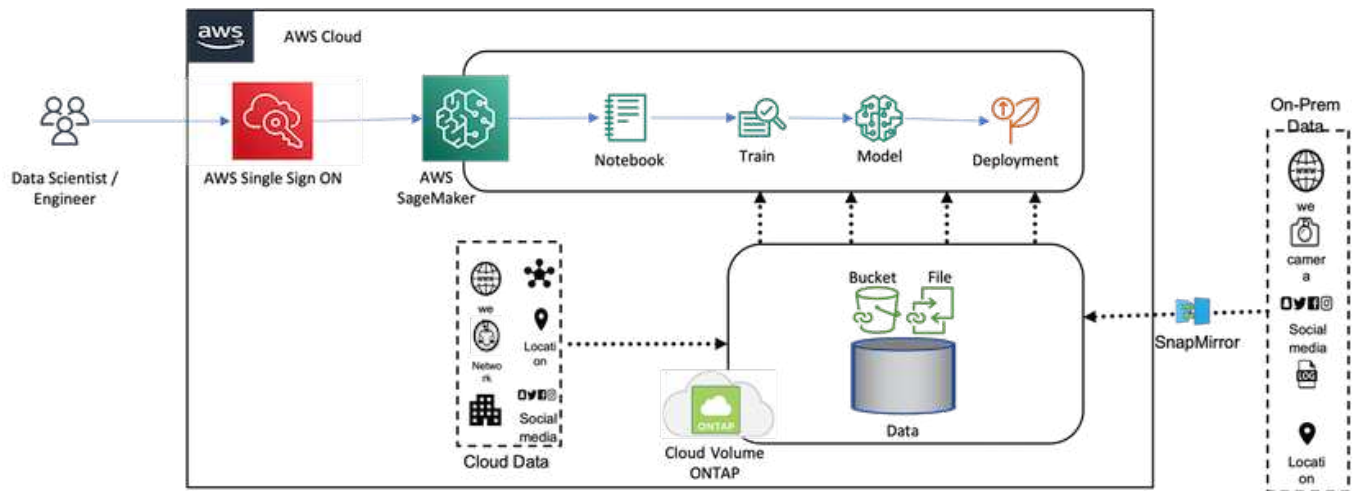
ソリューション技術

このソリューションでは、次のテクノロジーが活用されています。

- *AWS SageMaker ノートブック* 開発者やデータサイエンティストに機械学習機能を提供し、高品質の ML モデルを効率的に作成、トレーニング、デプロイできるようにします。
- *NetApp BlueXP。* オンプレミスだけでなく、AWS、Azure、Google Cloud 上のストレージの検出、展開、操作を可能にします。データ損失、サイバー脅威、予期しない停止に対するデータ保護を提供し、データストレージとインフラストラクチャを最適化します。
- *NetApp Cloud Volumes ONTAP* AWS、Azure、Google Cloud 上で NFS、SMB/CIFS、iSCSI、S3 プロトコルを使用したエンタープライズグレードのストレージボリュームを提供し、ユーザーにクラウド内のデータへのアクセスと管理の柔軟性を提供します。

ML データを保存するために BlueXP から作成された NetApp Cloud Volumes ONTAP。

次の図は、ソリューションの技術コンポーネントを示しています。



ユースケースの概要

NFS と S3 のデュアル プロトコル アクセスの潜在的な使用例としては、機械学習とデータサイエンスの分野が挙げられます。たとえば、データサイエンティストのチームが AWS SageMaker を使用して機械学習プロジェクトに取り組んでいる場合、NFS 形式で保存されたデータにアクセスする必要があります。ただし、他のチームメンバーと共同作業したり、S3 を使用する他のアプリケーションと統合したりするには、S3 バケットを介してデータにアクセスして共有する必要がある場合もあります。

NetApp Cloud Volumes ONTAP を利用することで、チームはデータを 1 つの場所に保存し、NFS プロトコルと S3 プロトコルの両方でアクセスできるようになります。データサイエンティストは AWS SageMaker から NFS 形式のデータに直接アクセスでき、他のチームメンバーやアプリケーションは S3 バケットを介して同じデータにアクセスできます。

このアプローチにより、追加のソフトウェアや異なるストレージソリューション間のデータ移行を必要とせずに、データに簡単かつ効率的にアクセスして共有できるようになります。また、より合理化されたワークフローとチームメンバー間のコラボレーションが可能になり、機械学習モデルの開発がより迅速かつ効果的になります。

データサイエンティストやその他のアプリケーションのためのデータの二重性

データは NFS で利用可能であり、AWS SageMaker の S3 からアクセスできます。

技術要件

データ二重性のユースケースには、NetApp BlueXP、NetApp Cloud Volumes ONTAP、AWS SageMaker Notebooks が必要です。

ソフトウェア要件

次の表は、ユースケースを実装するために必要なソフトウェアコンポーネントを示しています。

ソフトウェア	数量
BlueXP	1

ソフトウェア	数量
NetAppCloud Volumes ONTAP	1
AWS SageMaker ノートブック	1

展開手順

データ二重性ソリューションの展開には、次のタスクが含まれます。

- BlueXPコネクタ
- NetAppCloud Volumes ONTAP
- 機械学習用データ
- AWS セージメーカー
- Jupyter Notebook からの検証済み機械学習

BlueXPコネクタ

今回の検証ではAWSを使用しました。Azure および Google Cloud にも適用できます。AWS でBlueXPコネクタを作成するには、次の手順を実行します。

1. BlueXPの mcarl-marketplace-subscription に基づく資格情報を使用しました。
2. 環境に適したリージョン (例: us-east-1 [N. Virginia]) を選択し、認証方法 (例: Assume Role または AWS キー) を選択します。この検証では、AWS キーを使用します。
3. コネクタの名前を指定してロールを作成します。
4. パブリック IP が必要かどうかに応じて、VPC、サブネット、キーペアなどのネットワークの詳細を指定します。
5. セキュリティ グループの詳細 (ソース タイプからの HTTP、HTTPS、SSH アクセスなど) と、任意の場所や IP 範囲情報などを指定します。
6. BlueXPコネクタを確認して作成します。
7. AWS コンソールでBlueXP EC2 インスタンスの状態が実行中であることを確認し、[ネットワーク] タブから IP アドレスを確認します。
8. BlueXPポータルからコネクタ ユーザー インターフェイスにログインするか、IP アドレスを使用してブラウザからアクセスすることもできます。

NetAppCloud Volumes ONTAP

BlueXPでCloud Volumes ONTAPインスタンスを作成するには、次の手順を実行します。

1. 新しい作業環境を作成し、クラウドプロバイダーを選択して、Cloud Volumes ONTAPインスタンスのタイプ (シングル CVO、HA、Amazon FSx ONTAP for ONTAPなど) を選択します。
2. Cloud Volumes ONTAPクラスター名や資格情報などの詳細を提供します。この検証では、Cloud Volumes ONTAPインスタンスを作成しました。svm_sagemaker_cvo_sn1。
3. Cloud Volumes ONTAPに必要なサービスを選択します。この検証では監視のみを選択したため、*データセンサとコンプライアンス*と*クラウド サービスへのバックアップ*を無効にしました。

4. *場所と接続*セクションで、AWS リージョン、VPC、サブネット、セキュリティグループ、SSH 認証方法、およびパスワードまたはキーペアのいずれかを選択します。
5. 充電方法を選択してください。この検証には **Professional** を使用しました。
6. **POC** および小規模ワークロード、データベースおよびアプリケーション データの運用ワークロード、コスト効率の高い **DR**、最高パフォーマンスの運用ワークロード などの事前構成済みパッケージを選択できます。この検証では、*Poc と小規模ワークロード*を選択します。
7. 特定のサイズ、許可されたプロトコル、およびエクスポート オプションを使用してボリュームを作成します。この検証では、vol11。
8. プロファイル ディスク タイプと階層化ポリシーを選択します。この検証では、*ストレージ効率*と*汎用 SSD – 動的パフォーマンス*を無効にしました。
9. 最後に、Cloud Volumes ONTAPインスタンスを確認して作成します。次に、BlueXPがCloud Volumes ONTAP作業環境を作成するまで 15 ～ 20 分待ちます。
10. Duality プロトコルを有効にするには、次のパラメータを設定します。Duality プロトコル (NFS/S3) はONTAP 9 からサポートされます。 12.1 以降。
 - a. この検証では、SVMと呼ばれるものを作成しました。svm_sagemaker_cvo_sn1`とボリューム`vol11。
 - b. SVM が NFS および S3 のプロトコルをサポートしていることを確認します。そうでない場合は、それらをサポートするように SVM を変更します。

```

sagemaker_cvo_sn1::> vservers show -vservers svm_sagemaker_cvo_sn1
                                Vserver: svm_sagemaker_cvo_sn1
                                Vserver Type: data
                                Vserver Subtype: default
                                Vserver UUID: 911065dd-a8bc-11ed-bc24-
e1c0f00ad86b
                                Root Volume:
svm_sagemaker_cvo_sn1_root
                                Aggregate: aggr1
                                NIS Domain: -
                                Root Volume Security Style: unix
                                LDAP Client: -
                                Default Volume Language Code: C.UTF-8
                                Snapshot Policy: default
                                Data Services: data-cifs, data-
flexcache,
                                data-iscsi, data-nfs,
                                data-nvme-tcp
                                Comment:
                                Quota Policy: default
                                List of Aggregates Assigned: aggr1
                                Limit on Maximum Number of Volumes allowed: unlimited
                                Vserver Admin State: running
                                Vserver Operational State: running
                                Vserver Operational State Stopped Reason: -
                                Allowed Protocols: nfs, cifs, fcp, iscsi,
ndmp, s3
                                Disallowed Protocols: nvme
                                Is Vserver with Infinite Volume: false
                                QoS Policy Group: -
                                Caching Policy Name: -
                                Config Lock: false
                                IPspace Name: Default
                                Foreground Process: -
                                Logical Space Reporting: true
                                Logical Space Enforcement: false
                                Default Anti_ransomware State of the Vserver's Volumes: disabled
                                Enable Analytics on New Volumes: false
                                Enable Activity Tracking on New Volumes: false

sagemaker_cvo_sn1::>

```

11. 必要に応じて CA 証明書を作成してインストールします。

12. サービス データ ポリシーを作成します。

```
sagemaker_cvo_sn1::*> network interface service-policy create -vserver
svm_sagemaker_cvo_sn1 -policy sagemaker_s3_nfs_policy -services data-
core,data-s3-server,data-nfs,data-flexcache
sagemaker_cvo_sn1::*> network interface create -vserver
svm_sagemaker_cvo_sn1 -lif svm_sagemaker_cvo_sn1_s3_lif -service-policy
sagemaker_s3_nfs_policy -home-node sagemaker_cvo_sn1-01 -address
172.30.10.41 -netmask 255.255.255.192
```

Warning: The configured failover-group has no valid failover targets for the LIF's failover-policy. To view the failover targets for a LIF, use the "network interface show -failover" command.

```
sagemaker_cvo_sn1::*>
```

```
sagemaker_cvo_sn1::*> network interface show
```

Logical Vserver Home	Status Interface	Network Admin/Oper	Current Address/Mask	Current Is Node	Is Port

sagemaker_cvo_sn1	cluster-mgmt	up/up	172.30.10.40/26	sagemaker_cvo_sn1-	
01					e0a
true					
	intercluster	up/up	172.30.10.48/26	sagemaker_cvo_sn1-	
01					e0a
true					
	sagemaker_cvo_sn1-01_mgmt1	up/up	172.30.10.58/26	sagemaker_cvo_sn1-	
01					e0a
true					
svm_sagemaker_cvo_sn1	svm_sagemaker_cvo_sn1_data_lif	up/up	172.30.10.23/26	sagemaker_cvo_sn1-	
01					e0a
true					
	svm_sagemaker_cvo_sn1_mgmt_lif	up/up	172.30.10.32/26	sagemaker_cvo_sn1-	
01					e0a
true					
	svm_sagemaker_cvo_sn1_s3_lif	up/up	172.30.10.41/26	sagemaker_cvo_sn1-	

01

e0a

true

6 entries were displayed.

sagemaker_cvo_sn1::*>

```
sagemaker_cvo_sn1::*> vsserver object-store-server create -vsserver  
svm_sagemaker_cvo_sn1 -is-http-enabled true -object-store-server  
svm_sagemaker_cvo_s3_sn1 -is-https-enabled false  
sagemaker_cvo_sn1::*> vsserver object-store-server show
```

Vserver: svm_sagemaker_cvo_sn1

Object Store Server Name: svm_sagemaker_cvo_s3_sn1

Administrative State: up

HTTP Enabled: true

Listener Port For HTTP: 80

HTTPS Enabled: false

Secure Listener Port For HTTPS: 443

Certificate for HTTPS Connections: -

Default UNIX User: pcuser

Default Windows User: -

Comment:

sagemaker_cvo_sn1::*>

13. 集計の詳細を確認します。

```
sagemaker_cvo_sn1::*> aggr show
```

Aggregate Status	Size	Available	Used%	State	#Vols	Nodes	RAID
-----	-----	-----	-----	-----	-----	-----	-----
aggr0_sagemaker_cvo_sn1_01 raid0,	124.0GB	50.88GB	59%	online	1	sagemaker_cvo_ sn1-01	
normal							
aggr1 raid0,	907.1GB	904.9GB	0%	online	2	sagemaker_cvo_ sn1-01	
normal							

2 entries were displayed.

```
sagemaker_cvo_sn1::*>
```

14. ユーザーとグループを作成します。

```
sagemaker_cvo_sn1:*> vservers object-store-server user create -vservers
svm_sagemaker_cvo_sn1 -user s3user

sagemaker_cvo_sn1:*> vservers object-store-server user show
Vserver      User      ID      Access Key      Secret Key
-----
svm_sagemaker_cvo_sn1
      root      0      -      -
      Comment: Root User
svm_sagemaker_cvo_sn1
      s3user      1      0ZNAX21JW5Q8AP80CQ2E
PpLs4gA9K0_2gPhuykkp014gBjcC9Rbi3QDX_6rr
2 entries were displayed.

sagemaker_cvo_sn1:*>

sagemaker_cvo_sn1:*> vservers object-store-server group create -name
s3group -users s3user -comment ""

sagemaker_cvo_sn1:*>
sagemaker_cvo_sn1:*> vservers object-store-server group delete -gid 1
-vservers svm_sagemaker_cvo_sn1

sagemaker_cvo_sn1:*> vservers object-store-server group create -name
s3group -users s3user -comment "" -policies FullAccess

sagemaker_cvo_sn1:*>
```

15. NFS ボリューム上にバケットを作成します。

```
sagemaker_cvo_sn1::~*> vsriver object-store-server bucket create -bucket
ontapbucket1 -type nas -comment "" -vsriver svm_sagemaker_cvo_sn1 -nas
-path /vol1
sagemaker_cvo_sn1::~*> vsriver object-store-server bucket show
Vserver      Bucket      Type      Volume      Size
Encryption Role      NAS Path
-----
svm_sagemaker_cvo_sn1
ontapbucket1 nas      vol1      -      false
-      /vol1
sagemaker_cvo_sn1::~*>
```

AWS セージメーカー

AWS SageMaker から AWS ノートブックを作成するには、次の手順を実行します。

1. ノートブックインスタンスを作成するユーザーに AmazonSageMakerFullAccess IAM ポリシーがあること、または AmazonSageMakerFullAccess 権限を持つ既存のグループに属していることを確認します。この検証では、ユーザーは既存のグループの一部です。
2. 次の情報を入力します。
 - ノートブックインスタンス名。
 - インスタンスタイプ。
 - プラットフォーム識別子。
 - AmazonSageMakerFullAccess 権限を持つ IAM ロールを選択します。
 - ルートアクセス – 有効。
 - 暗号化キー - カスタム暗号化を選択しません。
 - 残りのデフォルトオプションはそのままにしておきます。
3. この検証では、SageMaker インスタンスの詳細は次のとおりです。

Amazon SageMaker > Notebook instances > nkarthiksagemaker

nkarthiksagemaker

Delete

Stop

Open Jupyter

Open JupyterLab

Notebook instance settings

Edit

Name	Status	Notebook instance type	Platform identifier
nkarthiksagemaker	✔ InService	ml.t2.medium	Amazon Linux 2, Jupyter Lab 3 (notebook-ai2-v2)
ARN	Creation time	Elastic Inference	Minimum IMDS Version
arn:aws:sagemaker:us-east-1:210811600188:notebook-instance/nkarthiksagemaker	Feb 16, 2023 18:55 UTC	-	2
Lifecycle configuration	Last updated	Volume Size	
-	Mar 22, 2023 20:59 UTC	5GB EBS	

Permissions and encryption

IAM role ARN

[arn:aws:iam::210811600188:role/SageMakerFullRole](#)

Root access

Enabled

Encryption key

Network

Subnet(s)

[subnet-00f94558](#)

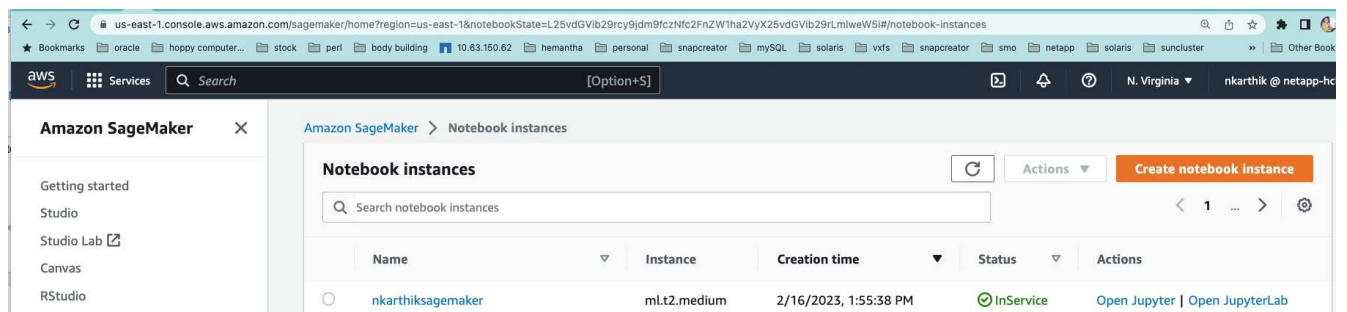
Security Group(s)

[sg-07111a8c16d67c81d](#)

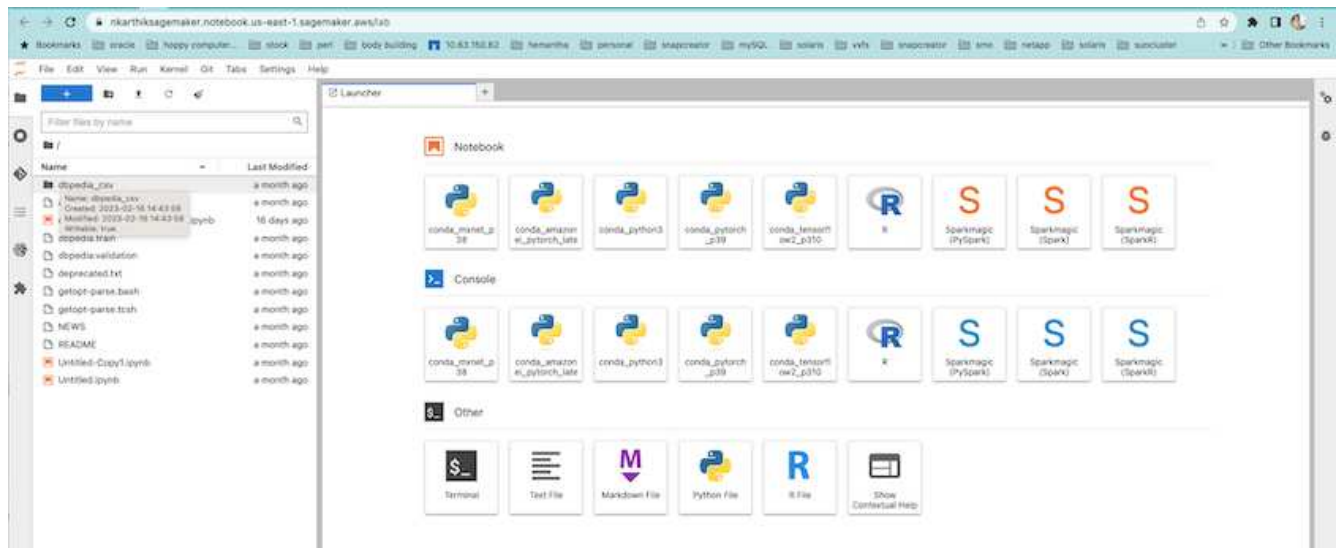
Direct internet access

Enabled: [Learn more](#)

4. AWS ノートブックを起動します。



5. Jupyter ラボを開きます。



- ターミナルにログインし、Cloud Volumes ONTAPボリュームをマウントします。

```
sh-4.2$ sudo mkdir /vol1; sudo mount -t nfs 172.30.10.41:/vol1 /vol1
sh-4.2$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	2.0G	0	2.0G	0%	/dev
tmpfs	2.0G	0	2.0G	0%	/dev/shm
tmpfs	2.0G	624K	2.0G	1%	/run
tmpfs	2.0G	0	2.0G	0%	/sys/fs/cgroup
/dev/xvda1	140G	114G	27G	82%	/
/dev/xvdf	4.8G	72K	4.6G	1%	/home/ec2-user/SageMaker
tmpfs	393M	0	393M	0%	/run/user/1001
tmpfs	393M	0	393M	0%	/run/user/1002
tmpfs	393M	0	393M	0%	/run/user/1000
172.30.10.41:/vol1	973M	189M	785M	20%	/vol1

```
sh-4.2$
```

- AWS CLI コマンドを使用して、Cloud Volumes ONTAPボリュームに作成されたバケットを確認します。

```
sh-4.2$ aws configure --profile netapp
AWS Access Key ID [None]: 0ZNAX21JW5Q8AP80CQ2E
AWS Secret Access Key [None]: PpLs4gA9K0_2gPhuykkp014gBjcC9Rbi3QDX_6rr
Default region name [None]: us-east-1
Default output format [None]:
sh-4.2$

sh-4.2$ aws s3 ls --profile netapp --endpoint-url
2023-02-10 17:59:48 ontapbucket1

sh-4.2$ aws s3 ls --profile netapp --endpoint-url s3://ontapbucket1/

2023-02-10 18:46:44          4747 1
2023-02-10 18:48:32          96 setup.cfg

sh-4.2$
```

機械学習用データ

この検証では、クラウドソーシングのコミュニティ活動である DBpedia のデータセットを使用して、さまざまなウィキメディア プロジェクトで作成された情報から構造化されたコンテンツを抽出しました。

1. DBpedia GitHub の場所からデータをダウンロードして抽出します。前のセクションで使ったのと同じ端末を使用します。

```

sh-4.2$ wget
--2023-02-14 23:12:11--
Resolving github.com (github.com)... 140.82.113.3
Connecting to github.com (github.com)|140.82.113.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: [following]
--2023-02-14 23:12:11--
Resolving raw.githubusercontent.com (raw.githubusercontent.com)...
185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connecting to raw.githubusercontent.com
(raw.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 68431223 (65M) [application/octet-stream]
Saving to: 'dbpedia_csv.tar.gz'

100%[=====
=====
=====>] 68,431,223  56.2MB/s   in 1.2s

2023-02-14 23:12:13 (56.2 MB/s) - 'dbpedia_csv.tar.gz' saved
[68431223/68431223]

sh-4.2$ tar -zxvf dbpedia_csv.tar.gz
dbpedia_csv/
dbpedia_csv/test.csv
dbpedia_csv/classes.txt
dbpedia_csv/train.csv
dbpedia_csv/readme.txt
sh-4.2$

```

2. データをCloud Volumes ONTAP の場所にコピーし、AWS CLI を使用して S3 バケットから確認します。


```

sh-4.2$ df -h
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                  2.0G         0  2.0G   0% /dev
tmpfs                     2.0G         0  2.0G   0% /dev/shm
tmpfs                     2.0G    628K  2.0G   1% /run
tmpfs                     2.0G         0  2.0G   0% /sys/fs/cgroup
/dev/xvda1                140G    114G   27G  82% /
/dev/xvdf                 4.8G     52K  4.6G   1% /home/ec2-user/SageMaker
tmpfs                     393M         0  393M   0% /run/user/1002
tmpfs                     393M         0  393M   0% /run/user/1001
tmpfs                     393M         0  393M   0% /run/user/1000
172.30.10.41:/vol1        973M    384K  973M   1% /vol1
sh-4.2$ pwd
/home/ec2-user
sh-4.2$ cp -ra dbpedia_csv /vol1
sh-4.2$ aws s3 ls --profile netapp --endpoint-url s3://ontapbucket1/
PRE dbpedia_csv/
2023-02-10 18:46:44          4747 1
2023-02-10 18:48:32           96 setup.cfg
sh-4.2$

```

3. 基本的な検証を実行して、S3 バケットで読み取り/書き込み機能が動作することを確認します。

```

sh-4.2$ aws s3 cp --profile netapp --endpoint-url /usr/share/doc/util-
linux-2.30.2 s3://ontapbucket1/ --recursive
upload: ../../usr/share/doc/util-linux-2.30.2/deprecated.txt to
s3://ontapbucket1/deprecated.txt
upload: ../../usr/share/doc/util-linux-2.30.2/getopt-parse.bash to
s3://ontapbucket1/getopt-parse.bash
upload: ../../usr/share/doc/util-linux-2.30.2/README to
s3://ontapbucket1/README
upload: ../../usr/share/doc/util-linux-2.30.2/getopt-parse.tcsh to
s3://ontapbucket1/getopt-parse.tcsh
upload: ../../usr/share/doc/util-linux-2.30.2/AUTHORS to
s3://ontapbucket1/AUTHORS
upload: ../../usr/share/doc/util-linux-2.30.2/NEWS to
s3://ontapbucket1/NEWS
sh-4.2$ aws s3 ls --profile netapp --endpoint-url
s3://ontapbucket1/s3://ontapbucket1/

An error occurred (InternalError) when calling the ListObjectsV2
operation: We encountered an internal error. Please try again.
sh-4.2$ aws s3 ls --profile netapp --endpoint-url s3://ontapbucket1/
PRE dbpedia_csv/

```

```

2023-02-16 19:19:27      26774 AUTHORS
2023-02-16 19:19:27      72727 NEWS
2023-02-16 19:19:27      4493 README
2023-02-16 19:19:27      2825 deprecated.txt
2023-02-16 19:19:27      1590 getopt-parse.bash
2023-02-16 19:19:27      2245 getopt-parse.tcsh
sh-4.2$ ls -ltr /vol1
total 132
drwxrwxr-x 2 ec2-user ec2-user 4096 Mar 29 2015 dbpedia_csv
-rw-r--r-- 1 nobody  nobody  2245 Apr 10 17:37 getopt-parse.tcsh
-rw-r--r-- 1 nobody  nobody  2825 Apr 10 17:37 deprecated.txt
-rw-r--r-- 1 nobody  nobody  4493 Apr 10 17:37 README
-rw-r--r-- 1 nobody  nobody  1590 Apr 10 17:37 getopt-parse.bash
-rw-r--r-- 1 nobody  nobody  26774 Apr 10 17:37 AUTHORS
-rw-r--r-- 1 nobody  nobody  72727 Apr 10 17:37 NEWS
sh-4.2$ ls -ltr /vol1/dbpedia_csv/
total 192104
-rw----- 1 ec2-user ec2-user 174148970 Mar 28 2015 train.csv
-rw----- 1 ec2-user ec2-user 21775285 Mar 28 2015 test.csv
-rw----- 1 ec2-user ec2-user 146 Mar 28 2015 classes.txt
-rw-rw-r-- 1 ec2-user ec2-user 1758 Mar 29 2015 readme.txt
sh-4.2$ chmod -R 777 /vol1/dbpedia_csv
sh-4.2$ ls -ltr /vol1/dbpedia_csv/
total 192104
-rwxrwxrwx 1 ec2-user ec2-user 174148970 Mar 28 2015 train.csv
-rwxrwxrwx 1 ec2-user ec2-user 21775285 Mar 28 2015 test.csv
-rwxrwxrwx 1 ec2-user ec2-user 146 Mar 28 2015 classes.txt
-rwxrwxrwx 1 ec2-user ec2-user 1758 Mar 29 2015 readme.txt
sh-4.2$ aws s3 cp --profile netapp --endpoint-url http://172.30.2.248/
s3://ontapbucket1/ /tmp --recursive
download: s3://ontapbucket1/AUTHORS to ../../tmp/AUTHORS
download: s3://ontapbucket1/README to ../../tmp/README
download: s3://ontapbucket1/NEWS to ../../tmp/NEWS
download: s3://ontapbucket1/dbpedia_csv/classes.txt to
../../tmp/dbpedia_csv/classes.txt
download: s3://ontapbucket1/dbpedia_csv/readme.txt to
../../tmp/dbpedia_csv/readme.txt
download: s3://ontapbucket1/deprecated.txt to ../../tmp/deprecated.txt
download: s3://ontapbucket1/getopt-parse.bash to ../../tmp/getopt-
parse.bash
download: s3://ontapbucket1/getopt-parse.tcsh to ../../tmp/getopt-
parse.tcsh
download: s3://ontapbucket1/dbpedia_csv/test.csv to
../../tmp/dbpedia_csv/test.csv
download: s3://ontapbucket1/dbpedia_csv/train.csv to
../../tmp/dbpedia_csv/train.csv

```

```
sh-4.2$  
sh-4.2$ aws s3 ls --profile netapp --endpoint-url s3://ontapbucket1/  
PRE dbpedia_csv/  
2023-02-16 19:19:27 26774 AUTHORS  
2023-02-16 19:19:27 72727 NEWS  
2023-02-16 19:19:27 4493 README  
2023-02-16 19:19:27 2825 deprecated.txt  
2023-02-16 19:19:27 1590 getopt-parse.bash  
2023-02-16 19:19:27 2245 getopt-parse.tcsh  
sh-4.2$
```

Jupyter Notebook から機械学習を検証する

次の検証では、以下の SageMaker BlazingText の例を使用して、テキスト分類を通じて機械学習モデルを構築、トレーニング、およびデプロイする方法を示します。

1. boto3 および SageMaker パッケージをインストールします。

```
In [1]: pip install --upgrade boto3 sagemaker
```

出力：

```
Looking in indexes: https://pypi.org/simple,  
https://pip.repos.neuron.amazonaws.com  
Requirement already satisfied: boto3 in /home/ec2-  
user/anaconda3/envs/python3/lib/python3.10/site-packages (1.26.44)  
Collecting boto3  
  Downloading boto3-1.26.72-py3-none-any.whl (132 kB)  
132.7/132.7 kB 14.6 MB/s eta 0: 00:00  
Requirement already satisfied: sagemaker in /home/ec2-  
user/anaconda3/envs/python3/lib/python3.10/site-packages (2.127.0)  
Collecting sagemaker  
  Downloading sagemaker-2.132.0.tar.gz (668 kB)  
668.0/668.0 kB 12.3 MB/s eta 0:  
00:0000:01  
  Preparing metadata (setup.py) ... done  
Collecting botocore<1.30.0,>=1.29.72  
  Downloading botocore-1.29.72-py3-none-any.whl (10.4 MB)  
10.4/10.4 MB 44.3 MB/s eta 0: 00:0000:010:01  
Requirement already satisfied: s3transfer<0.7.0,>=0.6.0 in /home/ec2-  
user/anaconda3/envs/python3/lib/python3.10/site-packages (from boto3)
```

```

(0.6.0)
Requirement already satisfied: jmespath<2.0.0,>=0.7.1 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from boto3)
(0.10.0)
Requirement already satisfied: attrs<23,>=20.3.0 in /home/ec2-
user/anaconda
3/envs/python3/lib/python3.10/site-packages (from sagemaker) (22.1.0)
Requirement already satisfied: google-pasta in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (0.2.0)
Requirement already satisfied: numpy<2.0,>=1.9.0 in /home/ec2-
user/anaconda
3/envs/python3/lib/python3.10/site-packages (from sagemaker) (1.22.4)
Requirement already satisfied: protobuf<4.0,>=3.1 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (3.20.3)
Requirement already satisfied: protobuf3-to-dict<1.0,>=0.1.5 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from sagemaker)
(0.1.5)
Requirement already satisfied: smdebug_rulesconfig==1.0.1 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (1.
0.1) Requirement already satisfied: importlib-metadata<5.0,>=1.4.0 in
/home/ec2user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker)
(4.13.0)
Requirement already satisfied: packaging>=20.0 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (21.3)
Requirement already satisfied: pandas in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (1.5.1)
Requirement already satisfied: pathos in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (0.3.0)
Requirement already satisfied: schema in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
sagemaker) (0.7.5) Requirement already satisfied: python-
dateutil<3.0.0,>=2.1 in /home/ec2-use
r/anaconda3/envs/python3/lib/python3.10/site-packages (from
botocore<1.30.
0,>=1.29.72->boto3) (2.8.2)
Requirement already satisfied: urllib3<1.27,>=1.25.4 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
botocore<1.30.0,>=1.2

```

```

9.72->boto3) (1.26.8) Requirement already satisfied: zipp>=0.5 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from importlib-metadata<5.0,>=1.4.0->sagemaker) (3.10.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
packaging>=20.0->sagemaker) (3.0.9)
Requirement already satisfied: six in /home/ec2-
user/anaconda3/envs/python
3/lib/python3.10/site-packages (from protobuf3-to-dict<1.0,>=0.1.5-
>sagemaker) (1.16.0)
Requirement already satisfied: pytz>=2020.1 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from pandas-
>sagemaker) (2022.5)
Requirement already satisfied: ppft>=1.7.6.6 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from pathos-
>sagemaker) (1.7.6.6) Requirement already satisfied:
multiprocess>=0.70.14 in /home/ec2-user/anac
onda3/envs/python3/lib/python3.10/site-packages (from pathos->sagemaker)
(0.70.14)
Requirement already satisfied: dill>=0.3.6 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from pathos-
>sagemaker) (0.3.6)
Requirement already satisfied: pox>=0.3.2 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from pathos-
>sagemaker) (0.3.2) Requirement already satisfied: contextlib2>=0.5.5 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from schema->sagemaker) (21.
6.0) Building wheels for collected packages: sagemaker
  Building wheel for sagemaker (setup.py) ... done
  Created wheel for sagemaker: filename=sagemaker-2.132.0-py2.py3-none-
any.whl size=905449
sha256=f6100a5dc95627f2e2a49824e38f0481459a27805ee19b5a06ec
83db0252fd41
  Stored in directory: /home/ec2-
user/.cache/pip/wheels/60/41/b6/482e7ab096
520df034fbf2dddd244a1d7ba0681b27ef45aa61
Successfully built sagemaker
Installing collected packages: botocore, boto3, sagemaker
  Attempting uninstall: botocore      Found existing installation:
botocore 1.24.19
    Uninstalling botocore-1.24.19:      Successfully uninstalled
botocore-1.24.19
  Attempting uninstall: boto3      Found existing installation: boto3
1.26.44
    Uninstalling boto3-1.26.44:
      Successfully uninstalled boto3-1.26.44

```

```
Attempting uninstall: sagemaker      Found existing installation:
sagemaker 2.127.0
  Uninstalling sagemaker-2.127.0:
    Successfully uninstalled sagemaker-2.127.0
ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
awscli 1.27.44 requires botocore==1.29.44, but you have botocore 1.29.72
which is incompatible.
aiobotocore 2.0.1 requires botocore<1.22.9,>=1.22.8, but you have
botocore 1.29.72 which is incompatible. Successfully installed boto3-
1.26.72 botocore-1.29.72 sagemaker-2.132.0 Note: you may need to restart
the kernel to use updated packages.
```

2. 次のステップでは、データが(dbpedia_csv) はs3バケットからダウンロードされます`ontapbucket1`機械学習で使用される Jupyter Notebook インスタンスに。

```

In [2]: import sagemaker
In [3]: from sagemaker import get_execution_role
In [4]:
import json
import boto3
sess = sagemaker.Session()
role = get_execution_role()
print(role)
bucket = "ontapbucket1"
print(bucket)
sess.s3_client = boto3.client('s3',region_name='',aws_access_key_id =
'0ZNAX21JW5Q8AP80CQ2E', aws_secret_access_key =
'PpLs4gA9K0_2gPhuykkp014gBjcC9Rbi3QDX_6rr',
                                use_ssl = False, endpoint_url =
'http://172.30.10.41',

config=boto3.session.Config(signature_version='s3v4',
s3={'addressing_style':'path'}) )
sess.s3_resource = boto3.resource('s3',region_name='',aws_access_key_id
= '0ZNAX21JW5Q8AP80CQ2E', aws_secret_access_key =
'PpLs4gA9K0_2gPhuykkp014gBjcC9Rbi3QDX_6rr',
                                use_ssl = False, endpoint_url =
'http://172.30.10.41',

config=boto3.session.Config(signature_version='s3v4',
s3={'addressing_style':'path'}) )
prefix = "blazingtext/supervised"
import os
my_bucket = sess.s3_resource.Bucket(bucket)
my_bucket = sess.s3_resource.Bucket(bucket)
#os.mkdir('dbpedia_csv')
for s3_object in my_bucket.objects.all():
    filename = s3_object.key
#    print(filename)
#    print(s3_object.key)
    my_bucket.download_file(s3_object.key, filename)

```

3. 次のコードは、推論中に実際のクラス名を取得するために使用される整数インデックスからクラス ラベル へのマッピングを作成します。

```

index_to_label = {}
with open("dbpedia_csv/classes.txt") as f:
    for i,label in enumerate(f.readlines()):
        index_to_label[str(i + 1)] = label.strip()

```

出力には、 ontapbucket1 AWS SageMaker 機械学習検証のデータとして使用されるバケット。

```
arn:aws:iam::210811600188:role/SageMakerFullRole ontapbucket1
AUTHORS
AUTHORS
NEWS
NEWS
README README
dbpedia_csv/classes.txt dbpedia_csv/classes.txt dbpedia_csv/readme.txt
dbpedia_csv/readme.txt dbpedia_csv/test.csv dbpedia_csv/test.csv
dbpedia_csv/train.csv dbpedia_csv/train.csv deprecated.txt
deprecated.txt getopt-parse.bash getopt-parse.bash getopt-parse.tcsh
getopt-parse.tcsh
In [5]: ls
AUTHORS          deprecated.txt      getopt-parse.tcsh  NEWS
Untitled.ipynb dbpedia_csv/      getopt-parse.bash  lost+found/
README
In [6]: ls -l dbpedia_csv
total 191344
-rw-rw-r-- 1 ec2-user ec2-user      146 Feb 16 19:43 classes.txt
-rw-rw-r-- 1 ec2-user ec2-user     1758 Feb 16 19:43 readme.txt
-rw-rw-r-- 1 ec2-user ec2-user  21775285 Feb 16 19:43 test.csv
-rw-rw-r-- 1 ec2-user ec2-user 174148970 Feb 16 19:43 train.csv
```

4. データ前処理フェーズを開始して、トレーニング データをスペースで区切られたトークン化されたテキスト形式に前処理します。この形式は、BlazingText アルゴリズムと nltk ライブラリで使用でき、DBPedia データセットからの入力文をトークン化します。nltk トークナイザーとその他のライブラリをダウンロードします。その `transform\_instance` 各データ インスタンスに並列に適用するには、Python マルチプロセッシング モジュールを使用します。

```
In [7]: from random import shuffle
import multiprocessing
from multiprocessing import Pool
import csv
import nltk
nltk.download("punkt")
def transform_instance(row):
    cur_row = []
    label = "__label__" + index_to_label [row[0]] # Prefix the index-ed
label with __label__
    cur_row.append (label)
    cur_row.extend(nltk.word_tokenize(row[1].lower ()))
    cur_row.extend(nltk.word_tokenize(row[2].lower ()))
    return cur_row
def preprocess(input_file, output_file, keep=1):
```



```

all_rows = []
with open(input_file,"r") as csvinfile:
    csv_reader = csv.reader(csvinfile, delimiter=",")
    for row in csv_reader:
        all_rows.append(row)
shuffle(all_rows)
all_rows = all_rows[: int(keep * len(all_rows))]
pool = Pool(processes=multiprocessing.cpu_count())
transformed_rows = pool.map(transform_instance, all_rows)
pool.close()
pool.join()
with open(output_file, "w") as csvoutfile:
    csv_writer = csv.writer (csvoutfile, delimiter=" ",
lineterminator="\n")
    csv_writer.writerows (transformed_rows)

# Preparing the training dataset
# since preprocessing the whole dataset might take a couple of minutes,
# we keep 20% of the training dataset for this demo.
# Set keep to 1 if you want to use the complete dataset
preprocess("dbpedia_csv/train.csv","dbpedia.train", keep=0.2)
# Preparing the validation dataset
preprocess("dbpedia_csv/test.csv","dbpedia.validation")
sess = sagemaker.Session()
role = get_execution_role()
print (role) # This is the role that sageMaker would use to leverage Aws
resources (S3, Cloudwatch) on your behalf
bucket = sess.default_bucket() # Replace with your own bucket name if
needed
print("default Bucket::: ")
print(bucket)

```

出力：

```

[nltk_data] Downloading package punkt to /home/ec2-user/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
arn:aws:iam::210811600188:role/SageMakerFullRole default Bucket:::
sagemaker-us-east-1-210811600188

```

5. フォーマットされたトレーニング データセットを S3 にアップロードして、SageMaker がトレーニング ジョブを実行するために使用できるようにします。次に、Python SDK を使用して、バケットとプレフィックスの場所に 2 つのファイルをアップロードします。

```

In [8]: %%time
train_channel = prefix + "/train"
validation_channel = prefix + "/validation"
sess.upload_data(path="dbpedia.train", bucket=bucket,
key_prefix=train_channel)
sess.upload_data(path="dbpedia.validation", bucket=bucket,
key_prefix=validation_channel)
s3_train_data = "s3://{}/{}".format(bucket, train_channel)
s3_validation_data = "s3://{}/{}".format(bucket, validation_channel)

```

出力：

```

CPU times: user 546 ms, sys: 163 ms, total: 709 ms
Wall time: 1.32 s

```

6. モデル成果物がロードされる S3 に出力場所を設定し、成果物をアルゴリズムのトレーニング ジョブの出力にできるようにします。作成する `sageMaker.estimator.Estimator` トレーニング ジョブを起動するオブジェクト。

```

In [9]: s3_output_location = "s3://{}/{}/output".format(bucket, prefix)
In [10]: region_name = boto3.Session().region_name
In [11]: container =
sagemaker.amazon.amazon_estimator.get_image_uri(region_name,
"blazingtext", "latest")
print("Using SageMaker BlazingText container: {} ({}).format(container,
region_name))

```

出力：

```

The method get_image_uri has been renamed in sagemaker>=2.
See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.
Defaulting to the only supported framework/algorithm version: 1.
Ignoring framework/algorithm version: latest.
Using SageMaker BlazingText container: 811284229777.dkr.ecr.us-east-1.
amazonaws.com/blazingtext:1 (us-east-1)

```

7. SageMakerを定義する `Estimator` c4.4xlarge インスタンスの教師ありモードを使用して、DBPedia データセットでテキスト分類をトレーニングするためのリソース構成とハイパーパラメータ。

```

In [12]: bt_model = sagemaker.estimator.Estimator(
    container,
    role,
    instance_count=1,
    instance_type="ml.c4.4xlarge",
    volume_size=30,
    max_run=360000,
    input_mode="File",
    output_path=s3_output_location,
    hyperparameters={
        "mode": "supervised",
        "epochs": 1,
        "min_count": 2,
        "learning_rate": 0.05,
        "vector_dim": 10,
        "early_stopping": True,
        "patience": 4,
        "min_epochs": 5,
        "word_ngrams": 2,
    },
)

```

8. データ チャンネルとアルゴリズム間のハンドシェイクを準備します。これを行うには、`sagemaker.session.s3\_input` データ チャンネルからオブジェクトを取得し、アルゴリズムが使用できるように辞書に保存します。

```

In [13]: train_data = sagemaker.inputs.TrainingInput(
    s3_train_data,
    distribution="FullyReplicated",
    content_type="text/plain",
    s3_data_type="S3Prefix",
)
validation_data = sagemaker.inputs.TrainingInput(
    s3_validation_data,
    distribution="FullyReplicated",
    content_type="text/plain",
    s3_data_type="S3Prefix",
)
data_channels = {"train": train_data, "validation": validation_data}

```

9. ジョブが完了すると、「ジョブ完了」メッセージが表示されます。トレーニング済みのモデルは、S3バケットに保存されています。`output\_path` 推定値で。

```
ln [14]: bt_model.fit(inputs=data_channels, logs=True)
```

出力：

```
INFO:sagemaker:Creating training-job with name: blazingtext-2023-02-16-
20-3
7-30-748
2023-02-16 20:37:30 Starting - Starting the training job.....
2023-02-16 20:38:09 Starting - Preparing the instances for
training.....
2023-02-16 20:39:24 Downloading - Downloading input data
2023-02-16 20:39:24 Training - Training image download completed.
Training in progress... Arguments: train
[02/16/2023 20:39:41 WARNING 140279908747072] Loggers have already been
set up. [02/16/2023 20:39:41 WARNING 140279908747072] Loggers have
already been set up.
[02/16/2023 20:39:41 INFO 140279908747072] nvidia-smi took:
0.0251793861389
16016 secs to identify 0 gpus
[02/16/2023 20:39:41 INFO 140279908747072] Running single machine CPU
Blazi ngText training using supervised mode.
Number of CPU sockets found in instance is 1
[02/16/2023 20:39:41 INFO 140279908747072] Processing
/opt/ml/input/data/tr ain/dbpedia.train . File size: 35.0693244934082 MB
[02/16/2023 20:39:41 INFO 140279908747072] Processing
/opt/ml/input/data/va lidation/dbpedia.validation . File size:
21.887572288513184 MB
Read 6M words
Number of words: 149301
Loading validation data from
/opt/ml/input/data/validation/dbpedia.validati on
Loaded validation data.
----- End of epoch: 1 ##### Alpha: 0.0000 Progress: 100.00%
Million Words/sec: 10.39 ##### Training finished.
Average throughput in Million words/sec: 10.39
Total training time in seconds: 0.60
#train_accuracy: 0.7223
Number of train examples: 112000
#validation_accuracy: 0.7205
Number of validation examples: 70000
2023-02-16 20:39:55 Uploading - Uploading generated training model
2023-02-16 20:40:11 Completed - Training job completed
Training seconds: 68
Billable seconds: 68
```

10. トレーニングが完了したら、トレーニング済みのモデルを Amazon SageMaker リアルタイムホストエンドポイントとしてデプロイし、予測を行います。

```
In [15]: from sagemaker.serializers import JSONSerializer
text_classifier = bt_model.deploy(
    initial_instance_count=1, instance_type="ml.m4.xlarge",
    serializer=JSONS
)
```

出力：

```
INFO:sagemaker:Creating model with name: blazingtext-2023-02-16-20-41-33-10
0
INFO:sagemaker:Creating endpoint-config with name blazingtext-2023-02-16-20-41-33-100
INFO:sagemaker:Creating endpoint with name blazingtext-2023-02-16-20-41-33-100
-----!
```

```
In [16]: sentences = [
    "Convair was an american aircraft manufacturing company which later expanded into rockets and spacecraft.",
    "Berwick secondary college is situated in the outer melbourne metropolitan suburb of berwick .",
]
# using the same nltk tokenizer that we used during data preparation for training
tokenized_sentences = [" ".join(nltk.word_tokenize(sent)) for sent in sentences]
payload = {"instances": tokenized_sentences} response = text_classifier.predict(payload)
predictions = json.loads(response)
print(json.dumps(predictions, indent=2))
```

```
[
  {
    "label": [
      "__label__Artist"
    ],
    "prob": [
      0.4090951681137085
    ]
  },
  {
    "label": [
      "__label__EducationalInstitution"
    ],
    "prob": [
      0.49466073513031006
    ]
  }
]
```

11. デフォルトでは、モデルは最も高い確率を持つ 1 つの予測を返します。上部を取得するには `k` 予測、設定 `k` 設定ファイル内。

```
In [17]: payload = {"instances": tokenized_sentences, "configuration":
{"k": 2}}
response = text_classifier.predict(payload)

predictions = json.loads(response)
print(json.dumps(predictions, indent=2))
```

```
[
  {
    "label": [
      "__label__Artist",
      "__label__MeanOfTransportation"
    ],
    "prob": [
      0.4090951681137085,
      0.26930734515190125
    ]
  },
  {
    "label": [
      "__label__EducationalInstitution",
      "__label__Building"
    ],
    "prob": [
      0.49466073513031006,
      0.15817692875862122
    ]
  }
]
```

12. ノートブックを閉じる前にエンドポイントを削除します。

```
In [18]: sess.delete_endpoint(text_classifier.endpoint)
WARNING:sagemaker.deprecations:The endpoint attribute has been renamed
in sagemaker>=2.
See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.
INFO:sagemaker:Deleting endpoint with name: blazingtext-2023-02-16-20-
41-33
-100
```

まとめ

この検証に基づいて、データサイエンティストとエンジニアは、NetApp Cloud Volumes ONTAPの S3 バケットを介して AWS SageMaker Jupyter Notebook の NFS データにアクセスできるようになります。このアプローチにより、追加のソフトウェアを必要とせずに、NFS と S3 の両方から同じデータに簡単にアクセスして共有できるようになります。

詳細情報の入手方法

このドキュメントに記載されている情報の詳細については、次のドキュメントや Web サイトを参照してください。

- SageMaker BlazingText を使用したテキスト分類
- ONTAPバージョンとS3オブジェクト ストレージのサポート

["https://docs.netapp.com/us-en/ontap/s3-config/ontap-version-support-s3-concept.html"](https://docs.netapp.com/us-en/ontap/s3-config/ontap-version-support-s3-concept.html)

著作権に関する情報

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。