

IBM Maximo PQI SaaS Visual Insights

Guide d'utilisation



Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 125.

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.ibm.com/ca/fr> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France
Direction Qualité
17, avenue de l'Europe
92275 Bois-Colombes Cedex*

© Copyright IBM France 2019. Tous droits réservés.

Table des matières

Avis aux lecteurs canadiens.....	ix
Chapitre 1. Présentation du produit.....	1
Rôles.....	1
Sauvegarde et restauration des données.....	1
Nouveautés de cette édition.....	2
Fonctions d'accessibilité.....	4
Chapitre 2. Création de systèmes de périphérie.....	5
Exigences des systèmes de périphérie.....	5
Ouverture des ports de périphérie.....	6
Installation des packs d'unité de traitement graphique NVIDIA pour Ubuntu.....	6
Installation des packs d'unité de traitement graphique NVIDIA pour Linux on NVIDIA Tesla K80 Power Systems Servers.....	7
Installation des packs d'unité de traitement graphique NVIDIA pour Linux on NVIDIA Tesla V100 Power Systems Servers.....	8
Installation de Caffe pour Ubuntu.....	9
Installation de TensorFlow pour Ubuntu.....	10
Installation de Caffe pour Linux on Power Systems Servers.....	10
Traitement des incidents liés à l'installation de Caffe.....	12
Installation d'Open CV.....	14
Installation de bibliothèques de détection d'objet.....	15
Traitement des incidents liés à l'installation de la bibliothèque Python Faster-RCNN.....	17
Configuration du serveur d'images.....	17
Configuration du magasin de modèles.....	18
Installation des modules Python.....	19
Installation de Paramiko pour Linux on Power Systems Servers.....	19
Enregistrement du serveur de périphérie dans l'application centrale.....	20
Enregistrement d'un serveur de périphérie connecté dans l'application centrale.....	20
Enregistrement d'un serveur de périphérie autonome dans l'application centrale.....	20
Installation d'un serveur de périphérie autonome.....	21
Vérification du statut des services sur les systèmes de périphérie.....	24
Mise à niveau des systèmes de périphérie.....	26
Chapitre 3. Création et utilisation de modèles.....	29
Structure des fichiers image compressés.....	29
Ajout d'images d'historique pour les groupes d'images	30
Outil d'affectation de libellés d'image.....	31
Création d'un groupe d'images sans libellé.....	31
Affectation manuelle des noms d'échantillon d'image.....	32
Affectation automatique des noms d'échantillon d'image.....	32
Création de modèle.....	32
Création de modèles.....	33
Utilisation du catalogue de modèles.....	34
Test des modèles dans le catalogue de modèles.....	35
Création de modèles à l'aide du catalogue de modèles.....	35
Entraînement des modèles.....	35
Modèles entraînés.....	36
Structure des fichiers de modèle.....	36
Modèles validés.....	43

Distribution de modèles entraînés sur des serveurs de périphérie.....	43
Nouvel entraînement des modèles	43
Utilisation de modèles avec un serveur de périphérie autonome.....	44
Publication des modèles.....	44
Déploiement d'une instance de modèle.....	44
Annulation du déploiement des modèles.....	45
Chapitre 4. Consultation des résultats d'inspection.....	47
Images.....	47
Filtrage des défauts.....	47
Vérification des défauts.....	47
Transfert d'images à l'aide du simulateur.....	48
Chapitre 5. Tableau de bord des indicateurs clés de performance.....	49
Chapitre 6. Intégration à Prescriptive Quality.....	51
Chapitre 7. Interface de programmation.....	53
Flux de travaux d'API.....	53
Préparation pour l'utilisation d'appels API.....	54
Réponses de service.....	54
Services de groupe de données.....	55
Obtenir tous les groupes de données.....	55
Obtention de tous les groupes de données avec des fichiers de données.....	56
Obtenir un groupe de données particulier.....	58
Créer un groupe de données.....	59
Mettre à jour un groupe de données.....	60
Supprimer des groupes de données.....	61
Services de fichier de données.....	62
Obtenir tous les fichiers de données appartenant à un groupe de données.....	62
Obtenir un fichier de données particulier.....	64
Télécharger le contenu binaire du fichier de données.....	65
Transférer des fichiers de données dans un groupe de données.....	65
Supprimer un fichier de données particulier.....	67
Services de groupe de données sans libellé.....	68
Transférer un fichier image compressé sans libellé.....	68
Créer un groupe de données sans libellé.....	69
Obtenir un groupe de données sans libellé.....	71
Services de modèle.....	72
Obtenir tous les modèles.....	72
Obtention d'un modèle particulier.....	74
Créer un modèle.....	75
Mettre à jour un modèle.....	78
Supprimer un modèle particulier.....	79
Obtenir tous les modèles partagés.....	80
Services d'instance de modèle.....	81
Créer une instance de modèle.....	81
Obtenir l'instance de modèle appartenant au modèle.....	83
Obtenir une instance de modèle particulière.....	85
Obtenir le résultat de la validation de l'instance de modèle.....	86
Services de résultat d'inspection.....	88
Obtenir la liste de résultats d'inspection.....	88
Obtenir les détails du résultat d'inspection.....	89
Obtenir la vue générale d'une cellule de résultat d'inspection.....	91
Confirmer les résultats d'inspection.....	92
Supprimer les résultats d'inspection.....	93
Services d'action d'instance de modèle.....	94

Entraîner une instance de modèle.....	94
Télécharger un fichier journal d'entraînement.....	95
Valider l'instance de modèle.....	96
Rejeter une instance de modèle.....	97
Déployer une instance de modèle.....	98
Entraîner à nouveau l'instance de modèle.....	99
Annuler le déploiement d'une instance de modèle.....	100
Services de serveur de périphérie.....	101
Créer un serveur de périphérie.....	101
Obtenir un serveur de périphérie.....	102
Supprimer un serveur de périphérie.....	103
Mettre à niveau un serveur de périphérie.....	104
Service d'évaluation.....	105
Evaluer une image.....	105
Service d'intégration QEWS.....	107
Obtenir un fichier de taux d'images de défaut.....	107
Service composite.....	108
Enregistrer un modèle.....	108
Services de serveurs de périphérie autonomes.....	109
Obtenir les modèles disponibles.....	109
Déployer un modèle.....	110
Annuler le déploiement du modèle.....	111
Transférer et évaluer une image sur le serveur de périphérie.....	112
Synchroniser le résultat de l'inspection du serveur de périphérie dans l'application centrale.....	113
Nettoyer un résultat d'inspection que vous avez synchronisé avec l'application centrale.....	114

Chapitre 8. Enregistrement, déploiement et test d'un modèle à l'aide de l'API..... 115

Chapitre 9. Traitement des incidents..... 117

Messages.....	117
Marques.....	126
Dispositions relatives à la documentation du produit.....	127
Déclaration IBM de confidentialité sur Internet.....	127

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
⌂ (Pos1)	⌂	Home
Fin	Fin	End
⬆️ (PgAr)	⬆️	PgUp
⬇️ (PgAv)	⬇️	PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
🔒 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

Chapitre 1. Présentation du produit

IBM® Maximo PQI SaaS Visual Insights est une solution de diffusion d'alertes et de surveillance de la qualité qui peut recueillir des images de produits et d'assemblages en cours et finis et les classer dans des catégories de défauts.

Rôles

Pour comprendre Maximo PQI SaaS Visual Insights, il est utile de comprendre la façon dont les différents rôles interagissent avec le produit.

Tableau 1. Rôles de Maximo PQI SaaS Visual Insights

Rôle	Description
Gestionnaire de modèles	Gère les types de défaut et les modèles, transfère des ensembles d'images pour des types de défaut spécifiques, entraîne des modèles, et distribue des modèles exécutables à des serveurs de périphérie.
Inspecteur	Vérifie les résultats d'inspection qui sont générés par le produit, change les types de défaut si nécessaire, signale les types de défaut "inconnu" et les transmet à son superviseur pour une évaluation approfondie.
Superviseur d'inspecteur	Revérifie les résultats d'inspection de l'inspecteur. Révise et classe les types de défaut "inconnu". Consulte le tableau de bord des indicateurs clés de performance, qui inclut les défauts par unité et le taux de défaut.

Sauvegarde et restauration des données

La redondance IBM Open Platform est utilisée pour protéger les données des clients dans un environnement big data. De plus, Tivoli Storage Manager est utilisé pour sauvegarder les données dans l'environnement de production, notamment des fichiers Linux. Les fichiers Linux peuvent être des fichiers transférés par le client et des fichiers de configuration/journaux de middleware/d'application.

Le tableau ci-dessous contient le planning de sauvegarde pour différents aspects de la solution.

Tableau 2. Planning de sauvegarde des données de la solution

Données	Type de sauvegarde	Fréquence	Heure (heure normale du Centre)	Durée de conservation
Fichiers	Complète	Deux fois par semaine	00 h 00 à 03 h 00 le 1er et le 16 de chaque mois	5 semaines
Fichiers	Incrémentielle	Deux fois par jour	00 h 00 à 03 h 00 et 12 h 00 à 15 h 00	14 jours

Pendant la fenêtre de sauvegarde, la solution est accessible. Toutefois, les performances peuvent être réduites.

Dans le cas d'une défaillance du système entraînant des dommages ou la perte des données, IBM vous aidera à restaurer les données au niveau des points de récupération, en fonction de sa stratégie de sauvegarde.

Nouveautés de cette édition

Les nouvelles fonctions ci-après sont disponibles dans IBM Maximo PQI SaaS Visual Insights.

Nouveautés de cette édition

Mars 2019

- Prise en charge du transfert de modèle TensorFlow et de l'évaluation d'image

Nouveautés dans les éditions précédentes

Janvier 2019

- Désormais, le nom du produit est IBM Maximo PQI SaaS Visual Insights.
- Résolution d'un problème de courbe d'entraînement survenant lors de l'entraînement de modèles de classification en parallèle.
- Résolution d'un problème d'entraînement de modèle survenant lors du nouvel entraînement de modèles YOLO.
- Résolution d'un problème survenant lors de la validation et de l'évaluation des modèles FRCNN-VGG16.
- Mise à jour des propriétés de connexion pour améliorer les accès concurrents à l'évaluation.

Novembre 2018

- Prise en charge de plusieurs clusters de serveurs de périphérie dans un titulaire. Les modèles peuvent être déployés dans plusieurs clusters de serveurs de périphérie.
- Prise en charge des clusters d'entraînement pour le modèle de classification, dans lesquels vous pouvez ajouter plus ou moins de serveurs, pour traiter plusieurs soumissions de travaux d'entraînement en parallèle afin de réduire le temps d'attente.
- Prise en charge de la gestion des versions de serveur de périphérie. Les informations de version sont gérées lorsque vous enregistrez ou mettez à niveau un serveur de périphérie. Vous pouvez vérifier la version du serveur de périphérie lorsque vous déployez un modèle.
- Ajout d'une règle d'archivage des images pour spécifier les types d'images à archiver et à utiliser pour le nouvel entraînement d'un modèle.
- Renvoi des détails de l'erreur par le produit en cas d'échec de l'évaluation.
- Résolution d'un problème de sécurité de type détournement de session.
- Mise à jour des règles de tarification pour la comptabilisation du nombre de fichiers d'entraînement, du nombre de fichiers d'évaluation, et du stockage utilisé.

Septembre 2018

- Prise en charge de l'époque en tant qu'unité lors de l'entraînement des modèles de classification. Elle facilite la définition des hyperparamètres.
- Clusters de serveurs d'entraînement activés lors de l'entraînement des modèles de détection d'objet. Ce regroupement prend en charge la mise à l'échelle si un serveur d'entraînement ne peut pas traiter plusieurs demandes parallèles.
- Prise en charge de l'exécution du serveur de périphérie sur la plateforme Linux on Power Systems Servers.

- Prise en charge des fichiers zip conditionnés sur MacOS lors du transfert des fichiers zip d'images et des fichiers zip de modèles.
- Amélioration de l'outil d'affectation de libellés automatique.

Juin 2018

- Capacité à nommer automatiquement les images d'entraînement par l'intermédiaire d'un petit fichier annoté, ce qui réduit le coût et le temps nécessaires pour annoter les données.
- Améliorations des serveurs de périphérie à partir des commentaires sur le terrain et notamment suppression de l'accès NFS depuis le centre, prise en charge de la haute disponibilité du serveur de périphérie maître et prise en charge de l'exécution d'un modèle sur un serveur de périphérie spécifique et une unité de traitement graphique spécifique.
- Amélioration de l'entraînement des modèles par nos spécialistes des données en fonction d'un réseau pré-entraîné pour prendre en charge les petites itérations avec une grande précision.

Mars 2018

- Prise en charge du parcours des modèles dans un catalogue de modèles et du test de vos propres images avec des modèles partagés.
- Prise en charge de l'utilisation de modèles partagés dans un catalogue de modèles et de l'entraînement de votre propre modèle avec des images spécifiques.
- Prise en charge de différents choix de types de modèle, d'algorithmes, de réseaux et d'hyperparamètres lorsque vous créez un modèle.
- Prise en charge de l'entraînement de modèle en ligne. Des fichiers de modèle sont générés automatiquement sans qu'il soit nécessaire d'utiliser d'autres outils d'entraînement.
- Tableau de bord d'entraînement qui présente un graphique de production des valeurs de perte et de précision du modèle avec des itérations et des fichiers journaux détaillés.
- Liste des instantanés disponibles au cours du processus d'entraînement. Prise en charge de l'utilisation d'un instantané en tant que modèle entraîné.
- Conseils pour les API en trois étapes en vue de l'intégration de service pour l'enregistrement, le déploiement et le test d'un modèle.
- Simulateur pouvant être utilisé pour envoyer des images au serveur et consulter les résultats d'inspection.

Décembre 2017

- Interface utilisateur permettant à un spécialiste des données de télécharger des ensembles de données et de joindre des modèles entraînés.
- Prise en charge du nouvel entraînement d'un modèle de détection d'objet.
- Interface utilisateur permettant à un gestionnaire de modèles de surveiller la file d'attente pour le nouvel entraînement.
- Prise en charge du mode hors ligne pour les serveurs de périphérie. En mode hors ligne, les serveurs de périphérie stockent les résultats des inspections localement et les envoient à l'application centrale par lots.
- Prise en charge d'un cluster de serveurs de périphérie pour le traitement des demandes d'évaluation avec équilibrage de charge.
- Support pour l'échantillonnage d'ensembles de données pour le nouvel entraînement des modèles de classification.
- Possibilité d'exporter des fichiers au format CSV pour l'intégration à IBM Prescriptive Quality.
- API REST publique permettant d'exécuter des fonctions pour obtenir tous les groupes de données existants, tous les fichiers de données appartenant à un groupe de données, et tous les modèles existants.

Septembre 2017

- Désormais, vous pouvez utiliser des groupes d'images pour représenter un même type d'images en utilisant un ou plusieurs fichiers image compressés.
- Ajout de la prise en charge de plusieurs versions de modèle qui partagent les mêmes groupes d'images mais utilisent des fichiers image différents pour entraîner le modèle.
- Ajout du processus de nouvel entraînement d'un modèle. Vous pouvez procéder automatiquement ou manuellement au nouvel entraînement d'une nouvelle version de modèle en utilisant des fichiers image différents.
- Ajout de la prise en charge de la validation de modèle. Le processus de validation calcule et affiche un rapport sur la précision du modèle en s'appuyant sur des fichiers image de validation.
- Désormais, vous pouvez utiliser des cadres de défaut et des types de défaut dans une image pour signaler l'emplacement du défaut à l'inspecteur.
- Désormais, vous pouvez afficher plusieurs emplacements de défaut dans une image. Vous pouvez aussi ajouter, ajuster et supprimer des cadres de défaut dans une image.
- Mise à jour du tableau de bord des indicateurs clés de performance pour inclure le nombre de défauts par unité et le taux de défaut pour le superviseur d'inspecteur.
- Ajout de la prise en charge du modèle de détection d'objet permettant de détecter plusieurs défauts dans une image avec le modèle CNN (réseau neuronal convolutif).

Fonctions d'accessibilité

Les fonctions d'accessibilité permettent aux utilisateurs souffrant d'un handicap physique, tel qu'une mobilité réduite ou une vision limitée, d'utiliser les produits informatiques.

Pour plus d'informations sur l'engagement d'IBM envers l'accessibilité, voir [IBM Accessibility Center](http://www.ibm.com/able) (www.ibm.com/able).

La documentation au format HTML propose des fonctions d'accessibilité. Les documents au format PDF sont considérés comme des documents d'appoint et, en tant que tel, n'en sont pas dotés.

Chapitre 2. Création de systèmes de périphérie

Maximo PQI SaaS Visual Insights comprend l'application centrale et les serveurs de périphérie. Les serveurs de périphérie sont des systèmes Linux utilisés pour détecter les défauts d'exécution.

Les systèmes de périphérie utilisent l'infrastructure d'apprentissage en profondeur Caffe. Caffe est un environnement d'apprentissage de type réseau de neurones artificiel dédié. L'apprentissage en profondeur requiert un nombre significatif de ressources de traitement. Il peut être effectué efficacement à l'aide d'une unité de traitement graphique (GPU). La plupart des infrastructures d'apprentissage en profondeur prennent également en charge le traitement à l'aide de l'unité centrale, mais le traitement à l'aide d'une unité de traitement graphique fournit des performances raisonnables pour les environnements de production.

Les serveurs de périphérie sont regroupés en cluster pour l'équilibrage de charge. Un cluster se compose d'un serveur de périphérie maître et de plusieurs serveurs de périphérie esclave. Lorsqu'une cellule envoie une image en vue de son traitement, le serveur de périphérie maître reçoit l'image, puis l'envoie à un serveur de périphérie esclave disposant de ressources d'unité de traitement graphique (GPU).

L'architecture de regroupement en cluster des serveurs de périphérie est évolutive. Si vous avez besoin de davantage de ressources de traitement, vous pouvez ajouter d'autres serveurs de périphérie.

Lorsque vous créez un serveur de périphérie, vous indiquez s'il s'agit d'un serveur de périphérie maître ou esclave. Le premier serveur de périphérie que vous créez doit être le serveur de périphérie maître. Une fois qu'un serveur de périphérie est créé, vous ne pouvez pas changer son type. Vous ne pouvez créer qu'un seul serveur de périphérie maître par titulaire. Le serveur de périphérie maître ne peut pas être supprimé si tous les serveurs de périphérie esclave n'ont pas été supprimés.

Les serveurs de périphérie maîtres peuvent être connectés ou autonomes. Connecté signifie que lorsqu'une image est évaluée sur un serveur de périphérie, le résultat d'inspection est envoyé à l'application centrale immédiatement. Autonome signifie que le résultat d'inspection est stocké sur le serveur de périphérie jusqu'à ce que vous cliquiez sur **Extraire les résultats** dans la boîte de dialogue **Aperçu du serveur de périphérie** ou appelez le service qui est déployé sur le serveur de périphérie.

Exigences des systèmes de périphérie

Avant de créer un système de périphérie, assurez-vous que votre système satisfait la configuration requise.

- L'un des systèmes d'exploitation suivants :
 - Ubuntu 16.04 sur x86_64
 - Red Hat Enterprise Linux 7.5 sur x86_64
 - Red Hat Enterprise Linux 7.5 sur IBM Power System
- Architecture de l'unité centrale : x86_64 ou IBM Power System
- Un processeur 4 coeurs
- 64 Go de mémoire
- Une unité de disque dur de 2 To
- Une ou plusieurs cartes d'unité de traitement graphique (GPU) NVIDIA

Ouverture des ports de périphérie

Avant d'utiliser des serveurs de périphérie, vous devez ouvrir les ports de pare-feu qui sont utilisés par les systèmes de périphérie.

Pourquoi et quand exécuter cette tâche

Si le pare-feu UFW (Uncomplicated Firewall) est activé et actif sur les serveurs de périphérie, utilisez les commandes suivantes afin d'ouvrir UFW pour le service NFS (système de fichiers réseau) et afin d'ouvrir les ports de pare-feu suivants sur les serveurs de périphérie :

```
sudo ufw enable
sudo ufw allow nfs
sudo ufw allow 22
sudo ufw allow 5005
sudo ufw allow 5070:5090/tcp
sudo ufw allow 6005
sudo ufw allow 8449
```

Installation des packs d'unité de traitement graphique NVIDIA pour Ubuntu

Cette tâche permet d'installer les packs d'unité de traitement graphique NVIDIA pour les systèmes Ubuntu. Pour activer le traitement à l'aide d'une unité de traitement graphique (GPU), vous devez installer les packs d'unité de traitement graphique NVIDIA requis.

Procédure

1. Téléchargez et installez les pilotes pour votre unité de traitement graphique (GPU) NVIDIA. La liste des pilotes NVIDIA pour Ubuntu est disponible à l'adresse suivante : [Binary Driver How to - Nvidia](#). Voici un exemple de commande :

```
sudo apt-get install ubuntu-drivers-common
sudo ubuntu-drivers devices
sudo apt-get install nvidia-384
```
2. Utilisez la commande suivante pour vérifier si votre pilote NVIDIA a été installé correctement :

```
sudo nvidia-smi
```
3. Téléchargez et installez le kit d'outils CUDA de NVIDIA et la bibliothèque CUDNN correspondante. CUDA 8.0, CUDA 9.0 et CUDA 10.0 sont pris en charge. La commande suivante est un exemple pour CUDA 8.0 :

```
wget
http://developer.download.nvidia.com/compute/cuda/repos/ubuntu1604/x86_64/
cuda-repo-ubuntu1604_8.0.61-1_amd64.deb
```
4. Installez le fichier CUDA sur le serveur cible avec les commandes suivantes :

```
sudo dpkg -i cuda-repo-ubuntu1604_8.0.61-1_amd64.deb
sudo apt-get update
sudo apt-get install cuda
```

Etant donné que NVIDIA met à niveau le kit d'outils, il se peut que vous obteniez une version plus récente du kit d'outils en utilisant cette commande. Il est recommandé d'installer CUDA 8.0 car cette version a été testée. Si vous découvrez qu'une version plus récente a été installée, vous pouvez revenir à CUDA 8.0 avec les commandes suivantes :

```
sudo apt-get remove cuda
sudo apt-get install cuda-8-0
sudo ln -s /usr/local/cuda-8.0 /usr/local/cuda
```
5. Téléchargez la bibliothèque NVIDIA CUDA Deep Neural Network `cuda-8.0-linux-x64-v6.0.tgz` depuis le lien suivant : https://developer.nvidia.com/compute/machine-learning/cudnn/secure/v6/prod/8.0_20170307/cudnn-8.0-linux-x64-v6.0.tgz. Vous devrez peut-être créer un compte et vous connecter pour pouvoir télécharger le fichier.

- Décompressez le fichier `cuda-8.0-linux-x64-v6.0.tgz` dans le répertoire d'installation de cuda avec la commande suivante :

```
sudo tar -xvf cuda-8.0-linux-x64-v6.0.tgz -C /usr/local
```
- Définissez la variable d'environnement avec les commandes suivantes :

```
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH
export PATH=/usr/local/cuda/bin:$PATH
```

Ajoutez également ces commandes au script `~/.bashrc`.
- Installez le pack NVIDIA NCCL avec les commandes suivantes :

```
git clone https://github.com/NVIDIA/nccl.git
cd nccl
sudo make install -j4
```

Installation des packs d'unité de traitement graphique NVIDIA pour Linux on NVIDIA Tesla K80 Power Systems Servers

Cette tâche permet d'installer les packs d'unité de traitement graphique NVIDIA pour Linux on NVIDIA Tesla K80 Power Systems Servers. Pour activer le traitement à l'aide d'une unité de traitement graphique (GPU), vous devez installer les packs d'unité de traitement graphique NVIDIA requis.

Pourquoi et quand exécuter cette tâche

Pour les systèmes Power8, utilisez CUDA 8 comme décrit dans la tâche ci-dessous. Pour les systèmes Power9, remplacez CUDA 8 par CUDA 10 dans la tâche ci-dessous.

Procédure

- Téléchargez et installez les pilotes pour votre unité de traitement graphique (GPU) NVIDIA. La liste des pilotes NVIDIA pour Linux on Power Systems Servers est disponible à l'adresse suivante : [NVIDIA Driver Downloads](#). Suivez les instructions d'installation qui figurent sur la page de téléchargement.
- Téléchargez le fichier référentiel CUDA et installez CUDA 8 à l'aide des commandes suivantes :

```
wget https://developer.download.nvidia.com/compute/cuda/repos/rhel7/ppc64le/cuda-repo-rhel7-8.0.61-1.ppc64le.rpm
rpm -i cuda-repo-rhel7-8.0.61-1.ppc64le.rpm
yum clean all
yum install cuda
```
- Pour CUDA 8, téléchargez la bibliothèque NVIDIA CUDA Deep Neural Network `cuda-8.0-linux-ppc64le-v6.0.tgz` depuis l'URL suivante : https://developer.nvidia.com/compute/machine-learning/cudnn/secure/v6/prod/8.0_20170307/cudnn-8.0-linux-ppc64le-v6.0.tgz. Pour CUDA 10, téléchargez la bibliothèque NVIDIA CUDA Deep Neural Network `cuda-10.0-linux-ppc64le-v7.3.1.20.tgz` depuis l'URL suivante : <https://developer.download.nvidia.com/compute/cuda/repos/rhel7/ppc64le/cuda-10.0.130-1.ppc64le.rpm>. Vous devrez peut-être créer un compte et vous connecter pour pouvoir télécharger le fichier.
- Décompressez le fichier `cuda-8.0-linux-ppc64le-v6.0.tgz` dans le répertoire d'installation de cuda à l'aide de la commande suivante :

```
sudo tar -xvf cuda-8.0-linux-ppc64le-v6.0.tgz -C /usr/local
```
- Définissez la variable d'environnement à l'aide de la commande suivante :

```
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH
```

Ajoutez également cette commande au script `~/.bashrc`.
- Installez le pack NVIDIA NCCL à l'aide des commandes suivantes :

```
git clone https://github.com/NVIDIA/nccl.git
cd nccl
sudo make install -j4
```

Installation des packs d'unité de traitement graphique NVIDIA pour Linux on NVIDIA Tesla V100 Power Systems Servers

Cette tâche permet d'installer les packs d'unité de traitement graphique NVIDIA pour Linux on NVIDIA Tesla V100 Power Systems Servers. Pour activer le traitement à l'aide d'une unité de traitement graphique (GPU), vous devez installer les packs d'unité de traitement graphique NVIDIA requis.

Procédure

1. Téléchargez et installez NVIDIA CUDA 9.2.148 depuis developer.nvidia.com/cuda-92-download-archive.
 - a) Sélectionnez le système d'exploitation Linux.
 - b) Sélectionnez l'architecture ppc64le.
 - c) Sélectionnez la distribution RHEL.
 - d) Sélectionnez la version 7.
 - e) Sélectionnez le type de programme d'installation rpm (local). Le programme d'installation rpm local est préférable au programme rpm réseau car il garantit que la version qui est installée est celle qui a été téléchargée. Avec le programme d'installation rpm réseau, la commande `yum install cuda` installe toujours la version la plus récente du kit d'outils CUDA.
 - f) Cliquez sur **Download** pour télécharger le programme d'installation de base.
 - g) Cliquez sur **Download** pour télécharger le correctif 1.
 - h) Suivez les instructions d'installation pour Linux on POWER présentées dans le document [CUDA Quick Start Guide](#), notamment les étapes qui décrivent comment configurer l'environnement de développement CUDA en mettant à jour `PATH` et `LD_LIBRARY_PATH`.
2. Téléchargez le pilote NVIDIA 410.104 depuis <http://www.nvidia.com/Download/index.aspx>.
 - a) Sélectionnez le type de produit Tesla.
 - b) Sélectionnez la série de produits V-Series.
 - c) Sélectionnez le produit Tesla V100.
 - d) Sélectionnez le système d'exploitation Linux POWER LE RHEL 7.
 - e) Sélectionnez le kit d'outils CUDA 10.0.
 - f) Cliquez sur **Search** pour accéder au lien de téléchargement, puis cliquez sur **Download**.
3. **Remarque :** Pour les systèmes IBM Power System AC922, les mises à jour du système d'exploitation et du microprogramme du système doivent être appliquées pour que vous puissiez installer le pilote d'unité de traitement graphique le plus récent.

Installez CUDA et le pilote d'unité de traitement graphique :

 - a) Installez le rpm de référentiel pour la base de CUDA.
 - b) Installez le rpm de référentiel pour le correctif 1 de CUDA.
 - c) Installez le rpm de référentiel pour le pilote d'unité de traitement graphique.
 - d) Exécutez la commande suivante pour installer CUDA, le correctif et le pilote d'unité de traitement graphique :

```
sudo yum install cuda
```
 - e) Redémarrez le système pour activer le pilote.
4. Activez les services de persistance du système NVIDIA avec la commande shell suivante :

```
systemctl enable nvidia-persistenced
```
5. Vérifiez les pilotes NVIDIA avec la commande shell suivante :

```
nvidia-smi
```

6. Téléchargez NVIDIA cuDNN version 7.4.2 pour CUDA 10.0 (bibliothèque cuDNN version 7.4.2 pour Linux (Power8/Power9)) depuis developer.nvidia.com/cudnn. L'enregistrement dans NVIDIA Accelerated Computing Developer Program est requis.
7. Téléchargez NVIDIA NCCL version 2.3.7 pour CUDA 10.0 (NCCL 2.3.7 indépendant de la plateforme, pour CUDA 10.0 et IBM Power) depuis developer.nvidia.com/nccl. L'enregistrement dans NVIDIA Accelerated Computing Developer Program est requis.
8. Installez les packs cuDNN version 7.4.2 et NCCL version 2.3.7, puis actualisez le cache des bibliothèques partagées avec les commandes suivantes :


```
sudo tar -C /usr/local --no-same-owner -xzf cudnn-9.2-linux-ppc64le-v7.4.2.tgz
sudo tar -C /usr/local --no-same-owner -xzf nccl_2.3.7+cuda10.0_ppc64le.tgz
sudo ldconfig
```

Installation de Caffe pour Ubuntu

Vous devez installer l'infrastructure d'apprentissage en profondeur Caffe et les packs associés. Caffe est utilisé pour l'entraînement des modèles et la classification des défauts.

Procédure

1. Installez les packs requis pour Caffe avec les commandes suivantes :


```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install -y build-essential cmake git pkg-config
sudo apt-get install -y libprotobuf-dev libleveldb-dev libsnpappy-dev
libhdf5-serial-dev protobuf-compiler
sudo apt-get install -y libatlas-base-dev libjasper-dev
sudo apt-get install -y --no-install-recommends libboost-all-dev
sudo apt-get install -y libgflags-dev libgoogle-glog-dev liblmdb-dev
sudo apt-get install -y python-pip
sudo apt-get install -y python-dev
sudo apt-get install -y python-numpy python-scipy
sudo apt-get install -y libopencv-dev
sudo pip install opencv-python
sudo pip install flask_httpauth
sudo pip install gevent
sudo pip install pyinotify
sudo pip install tornado
```
2. Téléchargez le code source de Caffe avec la commande suivante :


```
wget https://github.com/BVLC/caffe/archive/1.0.zip
```
3. Décompressez le pack et placez-vous dans le répertoire du pack avec les commandes suivantes :


```
unzip 1.0.zip
cd ./caffe-1.0
```
4. Effectuez une copie du fichier de configuration make avec la commande suivante :


```
cp Makefile.config.example Makefile.config
```
5. Ajoutez les variables suivantes dans le fichier Makefile.config :


```
USE_CUDNN := 1
CUDA_DIR := /usr/local/cuda
PYTHON_INCLUDE := /usr/include/python2.7 \
/usr/lib/python2.7/dist-packages/numpy/core/include
PYTHON_LIB := /usr/lib/x86_64-linux-gnu
WITH_PYTHON_LAYER := 1
INCLUDE_DIRS := $(PYTHON_INCLUDE) /usr/local/include \
/usr/include/hdf5/serial
LIBRARY_DIRS := $(PYTHON_LIB) /usr/local/lib /usr/lib \
```

- ```

/usr/lib/x86_64-linux-gnu /usr/lib/x86_64-linux-gnu/hdf5/serial

```
6. Dans le répertoire `caffe-1.0`, exécutez la commande suivante :

```

find . -type f -exec sed -i -e 's^"hdf5.h"^"hdf5/serial/hdf5.h"^g' -e
's^"hdf5_h1.h"^"hdf5/serial/hdf5_h1.h"^g' '{} ' \;

```
  7. Exécutez les commandes suivantes :

```

cd /usr/lib/x86_64-linux-gnu
sudo ln -s libhdf5_serial.so.10.1.0 libhdf5.so
sudo ln -s libhdf5_serial_h1.so.10.0.2 libhdf5_h1.so

```
  8. Installez les packages Python requis dans le répertoire `caffe-1.0/python` avec les commandes suivantes :

```

cd {chemin-installation-caffe}/caffe-1.0/python
for req in $(cat requirements.txt); do sudo -H pip install $req --upgrade;
done

```

où `{chemin-installation-caffe}` est le chemin de déploiement de Caffe.
  9. Ouvrez le fichier `makefile` dans le répertoire `{chemin-installation-caffe}` et remplacez le paramètre `NVCCFLAGS` comme suit :

```

NVCCFLAGS += -D_FORCE_INLINES -ccbin=$(CXX) -Xcompiler -fPIC $
(COMMON_FLAGS)

```
  10. Dans le répertoire Caffe principal `caffe-1.0`, commencez la génération et l'installation de Caffe avec les commandes suivantes :

```

make all
make test
make runtest
make pycaffe
make distribute

```
  11. Ajoutez la ligne suivante dans le script `~/ .bashrc` :

```

export PYTHONPATH="/usr/lib/python2.7:{chemin-installation-caffe}/
caffe-1.0/python:$PYTHONPATH"

```

où `{chemin-installation-caffe}` est le chemin de déploiement de Caffe.

## Installation de TensorFlow pour Ubuntu

---

Si vous voulez évaluer une image en fonction d'un modèle TensorFlow, vous devez préparer l'environnement TensorFlow sur le serveur de périphérie.

### Procédure

1. Installez `tensorflow-gpu 1.4.0` avec la commande ci-dessous. Assurez-vous qu'aucune erreur n'est signalée au cours de l'installation et que les versions de tous les packages de dépendance sont correctes.

```

pip install tensorflow-gpu==1.4.0

```
2. Importez TensorFlow dans Python pour vérifier que TensorFlow est installé correctement. Si aucune erreur n'est signalée, cela signifie que TensorFlow est installé correctement.

```

import tensorflow

```

## Installation de Caffe pour Linux on Power Systems Servers

---

Cette tâche permet d'installer Caffe pour les systèmes Linux on Power Systems Servers. Vous devez installer l'infrastructure d'apprentissage en profondeur Caffe et les packs associés. Caffe est utilisé pour l'entraînement des modèles et la classification des défauts.

### Procédure

1. Installez les packs requis pour Caffe avec les commandes suivantes :

```

sudo yum clean all
sudo yum update
sudo yum install upgrade
sudo yum install -y libboost-*
sudo yum install -y gflags-devel glog-devel lmdb-devel
sudo yum install -y python-pip
sudo yum install -y python-devel
sudo yum install -y opencv-devel
sudo yum makecache
sudo yum install -y protobuf-devel leveldb-devel lmdb-devel snappy-devel
opencv-devel boost-devel hdf5-devel atlas-devel glog-devel gflags-devel
sudo yum install libpng-devel
sudo yum install freetype-devel
sudo yum install libjpeg-turbo-devel
sudo yum install opencv-python
sudo rpm -e --nodeps numpy
sudo pip install numpy
pip install --upgrade pip
sudo pip install flask_httppauth
sudo pip install gevent
sudo pip install pyinotify
ln -s /usr/local/cuda-10.0 /usr/local/cuda
pip install scikit-image
sudo pip install tornado

```

2. Liez la bibliothèque Atlas à l'aide des commandes suivantes :

```

ln -fs /usr/lib64/atlas/libsatlas.so /usr/lib64/libatlas.so
ln -fs /usr/lib64/atlas/libsatlas.so /usr/lib64/libcblas.so

```

3. Téléchargez le code source de Caffe avec la commande suivante :

```
wget https://github.com/BVLC/caffe/archive/1.0.zip
```

4. Décompressez le pack et placez-vous dans le répertoire du pack avec les commandes suivantes :

```

unzip 1.0.zip
cd ./caffe-1.0

```

5. Remplacez les fichiers Caffe suivants :

- Dans le répertoire `/include/caffe/util/cudnn.hpp`, remplacez le fichier `cudnn.hpp` par le fichier `cudnn.hpp` le plus récent se trouvant dans le référentiel Caffe sur GitHub : [github.com/BVLC/caffe.git](https://github.com/BVLC/caffe.git).
- Remplacez tous les fichiers `cudnn` qui se trouvent dans le dossier `/src/caffe/layers` par les fichiers `cudnn` les plus récents qui se trouvent dans le référentiel Caffe sur GitHub : [github.com/BVLC/caffe.git](https://github.com/BVLC/caffe.git).

Par exemple, exécutez les commandes suivantes :

```

cp -rf /root/source/caffe-git/caffe-master/include/caffe/util/
cudnn.hpp /usr/local/caffe-1.0/include/caffe/util/
cp -rf /root/source/caffe-git/caffe-master/src/caffe/layers/cudnn_* /usr/
local/caffe-1.0/src/caffe/layers/
cp -rf /root/source/caffe-git/caffe-master/include/caffe/layers/
cudnn_* /usr/local/p/usr/local/caffe-1.0/include/caffe/layers/

```

6. Effectuez une copie du fichier de configuration `make` avec la commande suivante :

```
cp Makefile.config.example Makefile.config
```

7. Ajoutez les variables suivantes dans le fichier `Makefile.config` :

```

USE_CUDNN := 1
CUDA_DIR := /usr/local/cuda
PYTHON_INCLUDE := /usr/include/python2.7 \
 /usr/lib64/python2.7/site-packages/numpy/core/include/
PYTHON_LIB := /usr/lib/gcc/ppc64le-redhat-linux/4.8.5/
WITH_PYTHON_LAYER := 1
INCLUDE_DIRS :=

```

```
$(PYTHON_INCLUDE) /usr/local/include
/usr/local/cuda-10.0/targets/ppc64le-linux/include/
LIBRARY_DIRS
:= $(PYTHON_LIB) /usr/local/lib /usr/lib \
 /usr/lib64 /usr/local/lib64
```

Modifiez CUDA\_ARCH en entrant le texte suivant :

```
CUDA_ARCH := -gencode arch=compute_30,code=sm_30
-gencode arch=compute_35,code=sm_35
-gencode arch=compute_50,code=sm_50
-gencode arch=compute_52,code=sm_52
-gencode arch=compute_60,code=sm_60
-gencode arch=compute_61,code=sm_61
-gencode arch=compute_61,code=compute_61
```

8. Installez les packages Python requis dans le répertoire `caffe-1.0/python` à l'aide des commandes suivantes :  

```
cd chemin-installation-caffe/caffe-1.0/python
for req in $(cat requirements.txt); do sudo -H pip install $req --upgrade;
done
```

où *chemin-installation-caffe* est le chemin de déploiement de Caffe.
9. Ouvrez le fichier Makefile dans le répertoire *chemin-installation-caffe* et modifiez le paramètre NVCCFLAGS comme suit :  
NVCCFLAGS += -D\_FORCE\_INLINES -ccbin=\$(CXX) -Xcompiler -fPIC \$(COMMON\_FLAGS)
10. Dans le répertoire Caffe principal `caffe-1.0`, commencez la génération et l'installation de Caffe avec les commandes suivantes :  

```
make all
make test
make runtest
make pycaffe
make distribute
```
11. Ajoutez la ligne suivante dans le script `~/.bashrc` :  

```
export PYTHONPATH="/usr/lib/python2.7:chemin-installation-caffe/caffe-1.0/python:$PYTHONPATH"
```

où *chemin-installation-caffe* est le chemin de déploiement de Caffe.
12. Exécutez les tests de post-installation suivants :
  - a) `make runtest | tee -a runtest.out`
  - b) `grep -i OK runtest.out | wc -l`  
La sortie du test Caffe doit être 2101.
  - c) `python -c "import caffe"`  
pour tester l'installation de Pycaffe.
  - d) `tail -n 2 runtest.out`  
runtest.out doit contenir le texte suivant :  
[=====] 2101 tests from 277 test cases ran. (291548 ms total)  
[ PASSED ] 2101 tests

## Traitement des incidents liés à l'installation de Caffe

Si un message d'erreur est consigné dans le journal lorsque vous commencez la génération et l'installation de Caffe, vous pouvez prendre certaines mesures pour essayer de résoudre le problème.

### Symptômes 1

Lorsque vous commencez la génération et l'installation de Caffe, le message suivant s'affiche :

```
1. In file included from ./include/caffe/util/device_alternate.hpp:40:0,
```

```

2. from ./include/caffe/common.hpp:19,
3. from src/caffe/common.cpp:7:
4. ./include/caffe/util/cudnn.hpp: In function 'void
caffe::cudnn::createPoolingDesc(cudnnPoolingStruct**,
 caffe::PoolingParameter_PoolMethod, cudnnPoolingMode_t*, int, int, int, int, int,
int)':
5. ./include/caffe/util/cudnn.hpp:127:41: error: too few arguments to function
'cudnnStatus_t
 cudnnSetPooling2dDescriptor(cudnnPoolingDescriptor_t, cudnnPoolingMode_t,
cudnnNanPropagation_t, int,
 int, int, int, int, int)'
6. pad_h, pad_w, stride_h, stride_w));
7.
8. ./include/caffe/util/cudnn.hpp:15:28: note: in definition of macro 'CUDNN_CHECK'
9. cudnnStatus_t status = condition; \
10. ^
11. In file included from ./include/caffe/util/cudnn.hpp:5:0,
12. from ./include/caffe/util/device_alternate.hpp:40,
13. from ./include/caffe/common.hpp:19,
14. from src/caffe/common.cpp:7:
15. /usr/local/cuda-7.5/include/cudnn.h:803:27: note: declared here
16. cudnnStatus_t CUDNNWAPI cudnnSetPooling2dDescriptor(
17. ^
18. make: *** [.build_release/src/caffe/common.o] Error 1
19.

```

## Résolution du problème 1

Pour corriger l'erreur, procédez comme suit :

1. Dans le répertoire `/include/caffe/util/cudnn.hpp`, remplacez le fichier `cudnn.hpp` par le fichier `cudnn.hpp` le plus récent se trouvant dans le référentiel Caffe sur GitHub.
2. Dans le dossier `/src/caffe/layers`, remplacez tous les fichiers `cudnn` par les fichiers `cudnn` les plus récents se trouvant dans le référentiel Caffe sur GitHub.

## Symptômes 2

Lorsque vous installez les packages Python requis dans le répertoire `caffe-1.0/python`, le message suivant apparaît :

```

Traceback (most recent call last):
 File "/usr/bin/pip", line 11, in <module>
 sys.exit(main())
 File "/usr/lib/python2.7/dist-packages/pip/__init__.py", line 215, in main
 locale.setlocale(locale.LC_ALL, '')
 File "/usr/lib/python2.7/locale.py", line 581, in setlocale
 return _setlocale(category, locale)
locale.Error: unsupported locale setting
Traceback (most recent call last):
 File "/usr/bin/pip", line 11, in <module>
 sys.exit(main())
 File "/usr/lib/python2.7/dist-packages/pip/__init__.py", line 215, in main
 locale.setlocale(locale.LC_ALL, '')
 File "/usr/lib/python2.7/locale.py", line 581, in setlocale
 return _setlocale(category, locale)
locale.Error: unsupported locale setting

```

## Résolution du problème 2

Pour corriger cette erreur, exécutez la commande suivante :

```
export LC_ALL=C
```

## Symptômes 3

Lorsque vous commencez la génération et l'installation de Caffe, le message suivant s'affiche :

```

nvcc fatal : Unsupported gpu architecture 'compute_20'
Makefile:595: recipe for target '.build_release/cuda/src/caffe/layers/prelu_layer.o' failed
make: *** [.build_release/cuda/src/caffe/layers/prelu_layer.o] Error 1

```

### Résolution du problème 3

Mettez en commentaire `-gencode arch=compute_20` dans `Makefile.config`.

### Symptômes 4

Lorsque vous commencez la génération et l'installation de Caffe, le message suivant s'affiche :

```
PROTOC src/caffe/proto/caffe.proto
make: protoc: Command not found
Makefile:639: recipe for target '.build_release/src/caffe/proto/caffe.pb.cc' failed
make: *** [.build_release/src/caffe/proto/caffe.pb.cc] Error 127
```

### Résolution du problème 4

Exécutez la commande suivante pour installer le programme protoc :

```
sudo apt install protobuf-compiler
```

### Symptômes 5

Lorsque vous commencez la génération et l'installation de Caffe, le message suivant s'affiche :

```
src/caffe/layers/hdf5_data_layer.cpp:13:30: fatal error: hdf5/serial/hdf5.h: No such file or
directory
compilation terminated.
Makefile:582: recipe for target '.build_release/src/caffe/layers/hdf5_data_layer.o' failed
make: *** [.build_release/src/caffe/layers/hdf5_data_layer.o] Error 1
```

### Résolution du problème 5

Il peut être nécessaire d'installer un package de dépendance Caffe avec les commandes suivantes :

```
sudo apt-get install libprotobuf-dev libleveldb-dev libsnpappy-dev libopencv-dev
libhdf5-serial-dev protobuf-compiler
```

```
sudo apt-get install --no-install-recommends libboost-all-dev
```

```
sudo apt-get install libopenblas-dev liblapack-dev libatlas-base-dev
```

```
sudo apt-get install libgflags-dev libgoogle-glog-dev liblmdb-dev
```

## Installation d'Open CV

---

Vous pouvez installer la bibliothèque Open Source Computer Vision (OpenCV) 3.2 si vous avez besoin de personnaliser des modèles avec les fonctions d'OpenCV 3.2.

### Procédure

1. Extrayez le code source d'OpenCV depuis Github :  
`wget https://github.com/opencv/opencv/archive/3.2.0.zip`
2. Décompressez le pack téléchargé, puis accédez au répertoire du pack :  
`unzip 3.2.0.zip`  
`cd opencv-3.2.0`
3. Créez un sous-répertoire de génération, puis accédez-y :  
`mkdir build`  
`cd build`
4. Préparez et générez la configuration de génération :  
`cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D WITH_TBB=ON -D WITH_V4L=ON ..`
5. Compilez et générez le pack :  
`make -j $(( $(nproc) + 1 ))`

6. Installez le pack :  
`sudo make install`
7. Enregistrez les bibliothèques et les modules sur le système :  
`sudo /bin/bash -c 'echo "/usr/local/lib" > /etc/ld.so.conf.d/opencv.conf'`  
`sudo ldconfig`
8. Si nécessaire, désinstallez l'ancienne version d'opencv pour éviter les conflits de version :  
`sudo apt-get autoremove libopencv-dev`

## Installation de bibliothèques de détection d'objet

Vous installez une bibliothèque de détection d'objet pour pouvoir exécuter le modèle de détection d'objet sur un serveur de périphérie.

### Pourquoi et quand exécuter cette tâche

IBM Maximo PQI SaaS Visual Insights prend en charge les bibliothèques de détection d'objet suivantes : YOLO (You Only Look Once), Faster R-CNN et SSD (Single Shot MultiBox Detector).

### Procédure

1. Installez les packs Python liés avec les commandes suivantes :

```

sudo apt-get install python-numpy
sudo apt-get install python-scipy
sudo pip install cython
sudo pip install easydict
sudo pip install uuid
sudo pip install multiprocessing

```

2. Installez toutes les bibliothèques suivantes :

| Bibliothèque                            | Instructions d'installation                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Bibliothèque YOLO version 2</b>      | <ol style="list-style-type: none"> <li>a. Exécutez les commandes suivantes pour obtenir le code source YOLO :<br/> <code>git clone --recursive https://github.com/pjreddie/darknet.git</code><br/> <code>cd darknet</code><br/> <code>git checkout 691debd</code></li> <li>b. Editez le fichier makefile en activant le paramètre <b>GPU</b>, et sélectionnez le paramètre <b>GPU ARCH</b> correct en fonction de la configuration de votre machine :<br/> <code>vi Makefile</code><br/> <code>GPU=1</code></li> <li>c. Exécutez la commande suivante pour compiler YOLO :<br/> <code>make</code></li> </ol>                                                        |
| <b>Bibliothèque Python Faster R-CNN</b> | <ol style="list-style-type: none"> <li>a. Exécutez la commande suivante pour obtenir le code source de Faster R-CNN :<br/> <code>git clone --recursive https://github.com/rbgirshick/py-faster-rcnn.git</code></li> <li>b. Dans le répertoire <code>py-faster-rcnn</code>, dans le dossier <code>lib</code>, exécutez la commande suivante pour compiler Cython :<br/> <code>make</code></li> <li>c. Placez-vous dans le répertoire <code>caffe-fast-rcnn</code> sous le répertoire <code>py-faster-rcnn</code> et effectuez une copie du fichier de configuration <code>make</code> avec les commandes suivantes :<br/> <code>cd caffe-fast-rcnn</code></li> </ol> |

| Bibliothèque     | Instructions d'installation                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                  | <p><b>cp Makefile.config.example Makefile.config</b></p> <p>d. Ajoutez les variables suivantes dans le fichier Makefile.config :</p> <pre data-bbox="586 285 1476 625"> USE_CUDNN := 1 CUDA_DIR := /usr/local/cuda PYTHON_INCLUDE := /usr/include/python2.7 \ /usr/lib/python2.7/dist-packages/numpy/core/include PYTHON_LIB := /usr/lib/x86_64-linux-gnu WITH_PYTHON_LAYER := 1 INCLUDE_DIRS := \$(PYTHON_INCLUDE) /usr/local/include \ /usr/include/hdf5/serial LIBRARY_DIRS := \$(PYTHON_LIB) /usr/local/lib /usr/lib \ /usr/lib/x86_64-linux-gnu /usr/lib/x86_64-linux-gnu/hdf5/serial </pre> <p>e. Exécutez la commande suivante pour compiler Caffe :</p> <p><b>make</b></p> <p>f. Exécutez la commande suivante pour compiler pycaffe à l'aide de la couche Python :</p> <p><b>make pycaffe</b></p>                                                                                                                                                                                                                                                               |
| Bibliothèque SSD | <p>a. Exécutez la commande suivante pour obtenir le code source de SSD :</p> <pre data-bbox="586 867 1476 999"> git clone --recursive https://github.com/weiliu89/caffe.git ~/ssd-caffe cd ~/ssd-caffe git checkout ssd </pre> <p>b. Effectuez une copie du fichier de configuration makefile avec la commande suivante :</p> <p><b>cp Makefile.config.example Makefile.config</b></p> <p>c. Editez le fichier de configuration makefile et changez les paramètres <b>CUDA_ARCH</b>, <b>BLAS</b> et <b>PYTHON_INCLUDE</b> en fonction de la configuration de votre machine.</p> <p>d. Exécutez la commande suivante sur une seule ligne :</p> <pre data-bbox="586 1230 1476 1350"> find . -type f -exec sed -i -e 's^"hdf5.h"^"hdf5/serial/hdf5.h"^g' -e 's^"hdf5_h1.h"^"hdf5/serial/hdf5_h1.h"^g' '{}' \; </pre> <p>e. Compilez le code avec la commande suivante :</p> <p><b>make -j8</b></p> <p>f. Compilez la couche Python avec la commande suivante :</p> <p><b>make py</b></p> <p>g. Compilez le test avec la commande suivante :</p> <p><b>make test -j8</b></p> |

3. Ajoutez les variables d'environnement suivantes au fichier ~/.bashrc : **YOLO\_HOME**, **FRCNN\_HOME** et **SSD\_HOME**. Voici un exemple d'ajout de variables d'environnement : **YOLO\_HOME=~/darknet/** **FRCNN\_HOME=~/py-faster-rcnn/** **SSD\_HOME=~/ssd-caffe/**.

## Traitement des incidents liés à l'installation de la bibliothèque Python Faster-RCNN

Si un message d'erreur s'affiche lorsque vous installez la bibliothèque Python Faster-RCNN, vous pouvez prendre des mesures pour tenter de résoudre le problème.

### Symptômes

Lorsque vous installez la bibliothèque Python Faster-RCNN, l'erreur de compilation cudnn suivante s'affiche :

```
CXX src/caffe/layers/hdf5_data_layer.cpp
./include/caffe/util/cudnn.hpp: In function 'const char* cudnnGetErrorString(cudnnStatus_t)':
./include/caffe/util/cudnn.hpp:21:10: warning: enumeration value
'CUDNN_STATUS_RUNTIME_PREREQUISITE_MISSING' not handled in switch
./include/caffe/util/cudnn.hpp:15:28: note: in definition of macro 'CUDNN_CHECK'
 cudnnStatus_t status = condition;
Makefile:563: recipe for target '.build_release/src/caffe/layers/hdf5_data_layer.o' failed
make: *** [.build_release/src/caffe/layers/hdf5_data_layer.o] Error 1
```

Sur les systèmes Power9 avec CUDA 10, l'erreur de compilation cudnn suivante s'affiche :

```
./include/caffe/util/cudnn.hpp:127:41: error: too few arguments to function
'cudnnStatus_t cudnnSetPooling2dDescriptor(cudnnPoolingDescriptor_t, cudnnPoolingMode_t,
cudnnNanPropagation_t, int, int, int, int, int, int, int)' pad_h, pad_w, stride_h, stride_w));
^./include/caffe/util/cudnn.hpp:15:28: note: in definition of macro 'CUDNN_CHECK'
cudnnStatus_t status = condition; \ ^In file included from ./include/caffe/util/
cudnn.hpp:5:0, from ./include/caffe/util/device_alternate.hpp:40, from ./include/
caffe/common.hpp:19, from ./include/caffe/blob.hpp:8, from src/caffe/blob.cpp:4:/
usr/local/cuda/include/cudnn.h:991:1: note: declared here cudnnSetPooling2dDescriptor
(cudnnPoolingDescriptor_t poolingDesc, ^
```

### Résolution du problème

Pour corriger l'erreur, localisez votre installation Caffe et exécutez les commandes suivantes :

```
cp -rf ~/caffe-1.0/include/caffe/util/cudnn.hpp ~/py-faster-rcnn/caffe-fast-rcnn/include/caffe/util/
cp -rf ~/caffe-1.0/src/caffe/layers/cudnn_* ~/py-faster-rcnn/caffe-fast-rcnn/src/caffe/layers/
cp -rf ~/caffe-1.0/include/caffe/layers/cudnn_* ~/py-faster-rcnn/caffe-fast-rcnn/include/caffe/layers/
```

Pour les systèmes Power9 avec CUDA 10, éditez le fichier Makefile.config avec la commande suivante :

```
vim ~/py-faster-rcnn/caffe-fast-rcnn/Makefile.config
```

Changez la valeur de CUDA\_ARCH comme suit :

```
CUDA_ARCH := -gencode arch=compute_30,code=sm_30 \
-gencode arch=compute_35,code=sm_35 \
-gencode arch=compute_50,code=sm_50 \
-gencode arch=compute_50,code=compute_50
```

## Configuration du serveur d'images

Vous configurez le serveur d'images sur la machine de périphérie maître pour que la machine de périphérie maître puisse stocker les images qui sont capturées par une caméra industrielle. Le serveur d'images est surveillé par le contrôleur de périphérie. Lorsqu'une nouvelle image est ajoutée, elle est évaluée et le résultat d'inspection est envoyé à l'application centrale pour que l'inspecteur puisse évaluer l'image et les résultats d'inspection.

### Procédure

1. Installez et démarrez le service NFS sur le système de périphérie maître avec la commande suivante :  
`sudo apt-get install nfs-kernel-server`  
Pour Linux on Power Systems Servers, utilisez les commandes suivantes :

- ```
sudo yum install -y nfs-utils
sudo yum install -y rpcbind
systemctl start nfs
systemctl start nfslock
systemctl start rpcbind
```
2. Créez un répertoire nommé `imageserver` et changez le propriétaire du dossier avec les commandes suivantes :

```
sudo mkdir /imageserver
sudo chown utilisateur /imageserver
```

où *utilisateur* est l'utilisateur que vous spécifiez lorsque vous enregistrez le serveur de périphérie dans l'application centrale.
 3. Facultatif : Procédez comme suit uniquement si vous disposez à la fois d'un système de périphérie maître et d'un système de périphérie esclave.
 - a. Pour exporter le répertoire `imageserver`, éditez le fichier `/etc/exports` avec la commande suivante :

```
vi /etc/exports
```
 - b. Ajoutez la ligne ci-dessous dans votre fichier d'exportation. Le réseau d'adresse IP est le sous-réseau IP pouvant accéder au dossier partagé. Le format est le suivant :
adresse IP/masque de sous-réseau. Tous les systèmes de périphérie esclave doivent pouvoir accéder au dossier partagé :

```
/imageserver adresse IP/réseau(rw, sync, no_root_squash, no_all_squash)
```

Exemple :

```
/imageserver 10.173.0.0/29(rw, sync, no_root_squash, no_all_squash)
```
 - c. Redémarrez le service NFS avec la commande suivante :

```
sudo service nfs-server restart
```
 4. Sur chaque machine de périphérie esclave, montez le répertoire `/imageserver` dans le dossier partagé du serveur de périphérie maître :
 - a) Exécutez la commande suivante sur la machine de périphérie esclave :

```
mkdir /imageserver
```
 - b) Editez le fichier `/etc/fstab` avec la commande suivante :

```
vi /etc/fstab
```
 - c) Ajoutez la ligne suivante au fichier en spécifiant l'adresse IP du serveur de périphérie maître :
adresse_IP:/imageserver /imageserver nfs defaults 0 0
 - d) Exécutez la commande suivante sur la machine de périphérie esclave pour que le montage prenne effet :

```
mount -a
```

Configuration du magasin de modèles

Vous devez configurer le magasin de modèles sur la machine de périphérie maître. Le magasin de modèles est un référentiel pour les modèles exécutables qui sont distribués depuis l'application centrale.

Procédure

1. Créez un répertoire `modelstore` et changez le propriétaire du dossier sur le système de périphérie maître avec les commandes suivantes :

```
sudo mkdir /modelstore
sudo chown utilisateur /modelstore
```

où *utilisateur* est l'utilisateur que vous spécifiez lorsque vous enregistrez le serveur de périphérie dans l'application centrale.
2. Facultatif : Procédez comme suit uniquement si vous disposez à la fois d'un système de périphérie maître et d'un système de périphérie esclave.

- a. Pour exporter le répertoire modelstore, éditez le fichier /etc/exports avec la commande suivante :
`vi /etc/exports`
 - b. Ajoutez la ligne ci-dessous dans votre fichier d'exportation. Le réseau d'adresse IP est le sous-réseau IP pouvant accéder au dossier partagé. Le format est le suivant :
adresse IP/masque de sous-réseau. Tous les systèmes de périphérie esclave doivent pouvoir accéder au dossier partagé :
`/modelstore adresse IP/réseau(rw, sync, no_root_squash, no_all_squash)`
 Exemple :
`/modelstore 10.173.0.0/29(rw, sync, no_root_squash, no_all_squash)`
 - c. Redémarrez le service NFS avec la commande suivante :
`sudo service nfs-server restart`
3. Sur chaque machine de périphérie esclave, montez le répertoire /modelstore dans le dossier partagé du serveur de périphérie maître :
 - a) Exécutez la commande suivante sur la machine de périphérie esclave :
`mkdir /modelstore`
 - b) Editez le fichier /etc/fstab avec la commande suivante :
`vi /etc/fstab`
 - c) Ajoutez la ligne suivante au fichier en spécifiant l'adresse IP du serveur de périphérie maître :
adresse_IP:/modelstore /modelstore nfs defaults 0 0
 - d) Exécutez la commande suivante sur la machine de périphérie esclave pour que le montage prenne effet :
`mount -a`

Installation des modules Python

Effectuez cette tâche pour installer les modules Python prérequis.

Procédure

Exécutez les commandes suivantes sur le système de périphérie maître et sur chaque système de périphérie esclave pour installer les packages prérequis :

```
sudo pip install flask
sudo pip install gevent
sudo pip install requests
sudo pip install pyinotify
sudo pip install opencv-python
sudo pip install lmbd
Pour Ubuntu: sudo apt-get install dos2unix
```

```
Pour Linux on Power Systems Servers: sudo yum install dos2unix
sudo apt install curl
sudo pip install flask_httpauth
sudo pip install paramiko
sudo pip install tornado
```

Installation de Paramiko pour Linux on Power Systems Servers

Si vous rencontrez des problèmes lors de l'installation de Paramiko sur Linux on Power Systems Servers, reportez-vous à cette tâche pour installer Paramiko manuellement.

Procédure

1. Définissez `rhel-75-server.repo` sur `/etc/yum.repos.d`
2. Exécutez la commande suivante :

```
sudo yum install redhat-rpm-config gcc libffi-devel python-devel openssl-  
devel
```

3. Exécutez la commande suivante :

```
pip install cryptography --no-binary cryptography
```
4. Exécutez la commande suivante :

```
sudo pip install bcrypt
```
5. Exécutez la commande suivante :

```
sudo pip install pynacl
```
6. Exécutez la commande suivante :

```
sudo pip install paramiko
```

Enregistrement du serveur de périphérie dans l'application centrale

Vous devez enregistrer des serveurs de périphérie dans l'application centrale. La procédure est différente pour les serveurs de périphérie connectés et les serveurs de périphérie autonomes.

Il existe deux types de serveur de périphérie : les serveurs de périphérie connectés et les serveurs de périphérie autonomes. Un serveur de périphérie connecté possède une adresse IP publique qui permet à l'application centrale de se connecter au serveur de périphérie directement. Le serveur de périphérie envoie les résultats de l'inspection à l'application centrale en temps réel. Un serveur de périphérie autonome ne possède pas d'adresse IP publique et parfois, ne peut pas se connecter à l'application centrale. Le résultat de l'inspection peut être stocké localement sur le serveur de périphérie autonome, et l'administrateur du serveur de périphérie doit appeler le service de serveur de périphérie explicitement pour communiquer avec l'application centrale lorsque le serveur de périphérie se connecte à Internet.

Enregistrement d'un serveur de périphérie connecté dans l'application centrale

Une fois que vous avez configuré un système de périphérie connecté, vous devez l'enregistrer dans l'application centrale. Vous pouvez créer un serveur de périphérie ou en éditer un existant. Les serveurs de périphérie sont utilisés pour exécuter un modèle d'évaluation.

Procédure

1. Dans **Gestionnaire de modèles**, sélectionnez **Données > Serveurs de périphérie**.
2. Sélectionnez **Créer un serveur de périphérie** et entrez le nom du serveur de périphérie.
3. Entrez l'adresse IP, le nom d'utilisateur SSH et le mot de passe du système de périphérie, et spécifiez **Maître** ou **Esclave** comme type de serveur de périphérie. L'adresse IP doit être accessible depuis l'application centrale. Le nom d'utilisateur et le mot de passe SSH sont utilisés pour la connexion du système de périphérie en vue du déploiement du contrôleur de périphérie.
4. Si vous créez un serveur de périphérie maître, spécifiez **Connecté** comme mode de connexion. Si vous créez un serveur de périphérie esclave, spécifiez le serveur de périphérie maître correspondant.
5. Si vous avez sélectionné le mode de connexion, sélectionnez **Suivant** et spécifiez la règle de suppression.
6. Sélectionnez **Créer** pour créer le serveur de périphérie. Le contrôleur de périphérie et le moteur d'évaluation sont déployés sur le système de périphérie, et le serveur de périphérie est ajouté à la liste des serveurs de périphérie enregistrés.

Enregistrement d'un serveur de périphérie autonome dans l'application centrale

Une fois que vous avez configuré un système de périphérie autonome, vous devez l'enregistrer dans l'application centrale. Vous pouvez créer un serveur de périphérie ou en éditer un existant. Les serveurs de périphérie sont utilisés pour exécuter un modèle d'évaluation.

Pourquoi et quand exécuter cette tâche

Un serveur de périphérie autonome vous permet d'exécuter certaines fonctions du produit lorsque vous êtes déconnecté de l'application centrale, comme l'évaluation d'images dans la ligne de production, la

synchronisation des données avec le centre, et le nettoyage des données transférées. Pour installer un serveur de périphérie autonome, vous devez installer PostgreSQL et les modules Python requis. Vous devez aussi installer le serveur de périphérie sur le système de périphérie manuellement.

Procédure

1. Dans **Gestionnaire de modèles**, sélectionnez **Données > Serveurs de périphérie**.
2. Sélectionnez **Créer un serveur de périphérie** et entrez le nom du serveur de périphérie.
3. Entrez l'adresse IP, le nom d'utilisateur SSH et le mot de passe du système de périphérie, et spécifiez **Maître** ou **Esclave** comme type de serveur de périphérie. L'adresse IP doit être accessible depuis l'application centrale. Le nom d'utilisateur et le mot de passe SSH sont utilisés pour la connexion du système de périphérie en vue du déploiement du contrôleur de périphérie.
4. Si vous créez un serveur de périphérie maître, spécifiez **Autonome** comme mode de connexion.
5. Si vous avez sélectionné le mode de connexion, sélectionnez **Suivant** et spécifiez la règle de suppression.
6. Sélectionnez **Créer** pour créer le serveur de périphérie. Pour un serveur de périphérie autonome, l'administrateur du serveur de périphérie doit établir une connexion SSH au système de périphérie et installer le serveur de périphérie manuellement. Pour plus d'informations, voir la rubrique Installation d'un serveur de périphérie autonome.

Installation d'un serveur de périphérie autonome

Un serveur de périphérie autonome vous permet d'exécuter certaines fonctions du produit lorsque vous êtes déconnecté du centre, comme l'évaluation d'images dans la ligne de production, la synchronisation des données avec le centre, et le nettoyage des données transférées. Pour installer un serveur de périphérie autonome, vous devez installer PostgreSQL et le module python requis. Vous devez aussi installer le serveur de périphérie sur la machine de périphérie manuellement.

Pour créer un serveur de périphérie autonome, vous devez choisir le mode de connexion autonome lorsque vous créez le serveur de périphérie.

Vous devez créer le serveur de périphérie autonome sur le même système que le serveur de périphérie maître.

Installation de PostgreSQL pour Ubuntu

Cette tâche permet d'installer PostgreSQL sur les systèmes Ubuntu. Le serveur d'entraînement et le serveur de périphérie autonome utilisent une base de données PostgreSQL.

Procédure

1. Installez PostgreSQL, PHPPgadmin et Apache2 avec la commande suivante :
`sudo apt-get -y install postgresql postgresql-contrib phppgadmin`
2. Configurez l'utilisateur PostgreSQL.
 - a) Connectez-vous en tant qu'utilisateur PostgreSQL avec les commandes suivantes :
`sudo su`
`su - postgres`
`psql`
 - b) Configurez le mot de passe de l'utilisateur postgres avec les commandes suivantes :
`password postgres`
`mot_de_passe`
`\q`
3. Configurez Apache2 en éditant le fichier `nano phppgadmin.conf` :
`cd /etc/apache2/conf-available/`
`nano phppgadmin.conf`
Supprimez la ligne suivante : `Require local`. Ajoutez la ligne suivante dans le fichier :
`Require all granted`
4. Configurez PHPPgadmin en éditant le fichier `config.inc.php` :

```
cd /etc/phppgadmin/  
nano config.inc.php  
Recherchez la ligne suivante dans le fichier :  
$conf['extra_login_security'] = true  
Remplacez true par false.
```

5. Redémarrez PostgreSQL et Apache2 avec les commandes suivantes :

```
systemctl restart postgresql  
systemctl restart apache2
```
6. Vérifiez que vous pouvez accéder à l'interface utilisateur sur le serveur de périphérie autonome en accédant à l'URL suivante :
`http://IP_serveur_périphérie_autonome/phppgadmin`
où `IP_serveur_périphérie_autonome` est l'adresse IP du serveur de périphérie autonome.
7. Créez le schéma de base de données dans PostgreSQL.
 - a) Exécutez la commande suivante dans la console SQL dans PHPPgadmin :

```
create database edge with owner postgres encoding='UTF-8'  
lc_collate='en_US.utf8' lc_ctype='en_US.utf8' template template0;
```
 - b) Dans la base de données, créez les tables suivantes :

```
CREATE TABLE vi_titulaire_inspectionresult(id text, info jsonb);  
CREATE TABLE vi_titulaire_notification(id text, info jsonb);  
CREATE TABLE vi_titulaire_defectsummary(id text, info jsonb);  
CREATE TABLE vi_titulaire_uploaddataset(id text, info jsonb);  
CREATE TABLE vi_titulaire_syncprocess(id text, info jsonb);  
CREATE TABLE vi_titulaire_model(id text, info jsonb);  
CREATE TABLE vi_titulaire_datagroup(id text, info jsonb);
```

où *titulaire* est le titulaire de l'utilisateur connecté au centre Maximo PQI SaaS Visual Insights. Vous pouvez obtenir la valeur du titulaire depuis le profil utilisateur dans l'interface utilisateur de l'application centrale.

Installation de PostgreSQL pour Linux on Power Systems Servers

Cette tâche permet d'installer PostgreSQL pour Linux on Power Systems Servers. Le serveur d'entraînement et le serveur de périphérie autonome utilisent une base de données PostgreSQL.

Procédure

1. Connectez-vous en tant qu'utilisateur root :

```
sudo su
```
2. Téléchargez le source de PostgreSQL :

```
wget https://ftp.postgresql.org/pub/source/v9.5.13/postgresql-9.5.13.tar.gz
```
3. Installez PostgreSQL à l'aide des commandes suivantes :

```
tar -zxvf postgresql-9.5.13.tar.gz  
cd postgresql-9.5.13/  
yum -y install readline-devel  
./configure --prefix=/usr/local/postgresql  
make  
make install
```
4. Créez l'utilisateur postgres et modifiez le propriétaire du répertoire postgres :

```
useradd postgres  
chown -R postgres:postgres /usr/local/postgresql/
```
5. Connectez-vous en tant qu'utilisateur postgres :

```
su postgres
```
6. Configurez le chemin du système pour postgres :

```
vi ~/.bashrc  
PGHOME=/usr/local/postgresql  
export PGHOME  
PGDATA=/usr/local/postgresql/data
```

```
export PGDATA
PATH=$PATH:$HOME/.local/bin:$HOME/bin:$PGHOME/bin
export PATH
```

7. Rechargez la configuration à l'aide de la commande source :
`source ~/.bashrc`
8. Initialisez la base de données PostgreSQL :
`initdb`
9. Configurez la base de données. Ouvrez `postgresql.conf` dans `vi` :
`vi /usr/local/postgresql/data/postgresql.conf`
Remplacez :

```
#listen_address='localhost'
#port = 5432
```

par :

```
listen_address='*'
port = 5432
```

Ouvrez le fichier `pg_hba.conf` dans `vi` :
`vi /usr/local/postgresql/data/pg_hba.conf`
Ajoutez la ligne suivante dans le fichier :
`host all all 0.0.0.0/0 trust`

10. Redémarrez `postgresql` :
`pg_ctl -D /usr/local/postgresql/data -l logfile restart`
11. Modifiez le mot de passe de l'utilisateur `postgres` dans la base de données PostgreSQL :
`psql`
`ALTER USER postgres WITH PASSWORD 'mot_de_passe';`
`\q`
Si le service `postgresql` n'est pas démarré, exécutez les commandes suivantes :
`su postgres`
`vi ~/.bashrc`
Ajoutez `/usr/local/pgsql/bin/` au fichier :
`export PATH=/usr/local/cuda-8.0/bin:$PATH:/usr/local/pgsql/bin/`
Exécutez la commande suivante :
`source ~/.bashrc`
12. Créez le schéma de base de données dans PostgreSQL. Exécutez la commande suivante sur la console `psql` :
`create database edge with owner postgres encoding='UTF-8'`
`lc_collate='en_US.utf8' lc_ctype='en_US.utf8' template template0;`
Dans la base de données, créez les tables suivantes :

```
CREATE TABLE vi_titulaire_inspectionresult(id text, info jsonb);
CREATE TABLE vi_titulaire_notification(id text, info jsonb);
CREATE TABLE vi_titulaire_defectsummary(id text, info jsonb);
CREATE TABLE vi_titulaire_uploaddataset(id text, info jsonb);
CREATE TABLE vi_titulaire_syncprocess(id text, info jsonb);
CREATE TABLE vi_titulaire_model(id text, info jsonb);
CREATE TABLE vi_titulaire_datagroup(id text, info jsonb);
```

où *titulaire* est le titulaire de l'utilisateur connecté au centre Maximo PQI SaaS Visual Insights. Vous pouvez obtenir la valeur du titulaire depuis le profil utilisateur dans l'interface utilisateur de l'application centrale.

Installation des modules Python

Pour créer un serveur d'entraînement ou un serveur de périphérie autonome, vous devez installer le module Python requis.

Procédure

Installez le module Python avec la commande suivante :

```
sudo apt-get update
sudo apt -y install postgresql
sudo apt -y install libpq-dev
sudo pip install PyGreSQL
sudo pip install DBUtils
```

Installation du serveur de périphérie autonome

Vous devez installer le serveur de périphérie autonome sur la machine de périphérie manuellement.

Téléchargement du fichier de génération et du script d'installation

Vous devez télécharger le fichier de génération et le script d'installation du serveur de périphérie autonome pour installer un serveur de périphérie autonome.

Procédure

1. Dans l'application Maximo PQI SaaS Visual Insights, sélectionnez **Données > Serveurs de périphérie**. Sélectionnez le serveur de périphérie maître et affichez ses détails.
2. Sélectionnez et téléchargez le fichier de génération pour le système d'exploitation requis. Les systèmes d'exploitation suivants sont pris en charge : Ubuntu, Red Hat Enterprise Linux et IBM Power System.
3. Sélectionnez et téléchargez le fichier script shell d'installation.

Installation du fichier de génération sur le serveur de périphérie autonome

Pour installer le système de périphérie sur la machine de périphérie, vous devez utiliser un nom d'utilisateur SSH et établir une connexion SSH à la machine de périphérie.

Procédure

1. Une fois la connexion SSH établie, utilisez le chemin de déploiement de la boîte de dialogue de création d'un serveur de périphérie comme répertoire de travail.
2. Ajoutez le droit d'exécution pour le fichier script shell avec la commande suivante :
`chmod +x ./edgeDeployed.sh`
3. Exécutez le fichier script shell pour installer le serveur de périphérie autonome. Utilisez la commande suivante :
`./edgeDeployed.sh`
4. Mettez à jour le fichier `chemin_déploiement/vi_edge-bin_vi/vi_edge/postgres/DBConfig.json` avec vos informations PostgreSQL afin de corriger la connexion de base de données. Ensuite, exécutez les commandes suivantes pour redémarrer le serveur de périphérie :
`cd chemin_déploiement/vi_edge-bin_vi/vi_edge/`
`./restartController.sh`

Vérification du statut des services sur les systèmes de périphérie

Une fois que vous avez enregistré le serveur de périphérie dans l'application centrale, vérifiez que les processus Python appropriés sont exécutés sur les systèmes de périphérie.

Procédure

1. Une fois le serveur de périphérie installé sur le système de périphérie, établissez une connexion SSH au système de périphérie.

2. Sur le système de périphérie maître, exécutez la commande suivante afin de rechercher le processus Python pour le moteur principal :


```
ps aux | grep python
```

 Le résultat doit inclure le processus Python suivant :


```
python dossier_déploiement/vi_edge-bin_vi/vi_task_manager/run.py.
```

 Si vous ne trouvez pas le processus Python sur le système de périphérie maître, consultez les fichiers journaux dans `dossier_déploiement/vi_edge-bin_vi/vi_task_manager/master.log`. Essayez de démarrer le moteur principal en exécutant la commande suivante :


```
dossier_déploiement/vi_edge-bin_vi/vi_task_manager/restartMaster.sh
```

```
IP_noeud_maître
```
3. Sur le système de périphérie maître, exécutez la commande suivante afin de rechercher le processus Python pour le contrôleur de périphérie :


```
ps aux | grep python
```

 Le résultat doit inclure les processus Python suivants :


```
python dossier_déploiement/vi_edge-bin_vi/vi_edge/runMonitor.py
```

```
python dossier_déploiement/vi_edge-bin_vi/vi_edge/runService.py.
```

 Si vous ne trouvez pas le processus Python `runMonitor` sur le système de périphérie maître, consultez les fichiers journaux dans `dossier_déploiement/vi_edge-bin_vi/vi_edge/Event.log`. Si vous ne trouvez pas le processus Python `runService` sur le système de périphérie maître, consultez les fichiers journaux dans `dossier_déploiement/vi_edge-bin_vi/vi_edge/Service.log`. Essayez de démarrer le contrôleur de périphérie en exécutant la commande suivante :


```
dossier_déploiement/vi_edge-bin_vi/vi_edge/restartController.sh
```
4. Exécutez la commande suivante sur le système de périphérie maître et les systèmes de périphérie esclave afin de rechercher les processus Python pour le moteur d'évaluation du modèle de classification :


```
ps aux | grep python
```

 Le résultat doit inclure les processus Python suivants :


```
python dossier_déploiement/vi_edge-bin_vi/vi_score_engine_restful/
```

```
front_run.py 5005 6005
```

```
python dossier_déploiement/vi_edge-bin_vi/vi_score_engine_restful/
```

```
back_run.py 6005
```

 Si vous ne trouvez pas le processus Python sur les systèmes de périphérie, consultez les fichiers journaux :


```
dossier_déploiement/vi_edge-bin_vi/vi_score_engine_restful/front_log.txt
```

```
dossier_déploiement/vi_edge-bin_vi/vi_score_engine_restful/back_log.txt
```

 Une fois les éventuels problèmes résolus, le moteur principal devrait démarrer le moteur d'évaluation automatiquement. Pour démarrer le moteur d'évaluation manuellement, exécutez les commandes suivantes :


```
source ~/.bashrc
```

```
dossier_déploiement/vi_edge-bin_vi/vi_score_engine_restful/restartEngine.sh
```
5. Exécutez la commande suivante sur le système de périphérie maître et les systèmes de périphérie esclave afin de rechercher le processus Python pour le moteur d'évaluation du modèle de détection d'objet :


```
ps aux | grep python
```

 Le résultat doit inclure les processus Python suivants :


```
python dossier_déploiement/vi_edge-bin_vi/vi_obj_detection/RESTAPI/model/
```

```
run.py port gpuid FRCNN
```

```
python dossier_déploiement/vi_edge-bin_vi/vi_obj_detection/RESTAPI/model/
```

```
run.py port gpuid SSD
```

 Si vous ne trouvez pas les processus Python sur les systèmes de périphérie, consultez les fichiers journaux suivants :


```
dossier_déploiement/vi_edge-bin_vi/vi_obj_detection/RESTAPI/model/
```

```
FRCNN_id_gpu.log
```

```
dossier_déploiement/vi_edge-bin_vi/vi_obj_detection/RESTAPI/model/
```

```
SSD_id_gpu.log
```

Si le message d'erreur "No module named iotmyolo" apparaît, copiez le fichier `iotmyolo.so` ou `iotmyolo.py` depuis le répertoire `/home/utilisateur/vi_edge-bin_vi/vi_obj_detection/model_library/yolo` dans le répertoire `YOLO_HOME`.

Une fois les éventuels problèmes résolus, le moteur principal devrait démarrer le moteur d'évaluation automatiquement.

6. Régénérez la bibliothèque YOLO avec le composant de périphérie :

- Accédez au répertoire `/home/utilisateur/vi_edge-bin_vi/vi_obj_detection/model_library/yolo`.
- Ajoutez le fichier `detectorobj.c` dans le dossier `darknet/examples`.
- Editez le fichier `darknet/Makefile` et spécifiez `EXECOBJA=detectorobj.o`. Exemple de code dans le fichier `makefile` :

```
EXECOBJA=detectorobj.o captcha.o lsd.o super.o voxel.o art.o tag.o cifar.o go.o rnn.o
rnn_vid.o compare.o segmenter.o regressor.o classifier.o coco.o dice.o yolo.o
detector.o writing.o nightmare.o swag.o darknet.o
```

- Dans le fichier `makefile`, ajoutez `$(EXECOBJ)` pour les objets `$(SLIB)` et `$(ALIB)`. Exemple de code dans le fichier `makefile` :

```
$(ALIB): $(EXECOBJ) $(OBJS)
    $(AR) $(ARFLAGS) $@ $^
$(SLIB): $(EXECOBJ) $(OBJS)
    $(CC) $(CFLAGS) -shared $^ -o $@ $(LDFLAGS)
```

- Exécutez la commande suivante :
`make`

Mise à niveau des systèmes de périphérie

Mettez à niveau vos serveurs de périphérie si leur version est ancienne. Vous pouvez télécharger le fichier de génération et le script d'installation du serveur de périphérie pour mettre à niveau le serveur de périphérie.

Pourquoi et quand exécuter cette tâche

Si vous mettez à niveau l'application centrale Maximo PQI SaaS Visual Insights mais pas le serveur de périphérie, vous créez une incohérence entre l'application centrale et le serveur de périphérie, et le déploiement des modèles peut échouer. Un message d'erreur similaire au suivant, indiquant que le serveur de périphérie doit être mis à niveau avant le déploiement d'un nouveau modèle, peut s'afficher : "The edge needs to be upgraded before deploying new model."

Procédure

- Exécutez la commande suivante pour obtenir l'ID du serveur de périphérie :
`/ibm/iotm/vi/service/edge?user=xxx&solution=vi`
- Pour le script shell d'installation, appelez le service suivant :
`/ibm/iotm/vi/service/edgeFile?edgeId=xxx&version=shell&user=xxx&solution=vi`
Remplacez `edgeId=xxx` par l'ID de serveur de périphérie que vous avez obtenu à l'étape précédente. Sauvegardez le fichier en tant que `edgeDeployed.sh`. Voici un exemple de commande :
`curl -k -H "apikey:votre_clé_api" "https://iotm.predictivesolutionsapps.ibmcloud.com/ibm/iotm/vi/service/edgeFile?edgeId=1508123458000&version=shell&user=votre_utilisateur&solution=vi" edgeDeployed.sh`
- Vérifiez le fichier `edgeDeployed.sh`. Vérifiez les valeurs pour `username`, qui correspond à l'utilisateur SSH employé pour démarrer les services de serveur de périphérie, `password`, qui correspond au mot de passe de l'utilisateur SSH, et `basefolder`, qui correspond au chemin de déploiement des services de serveur de périphérie. Entrez la valeur pour `<to edit>`.
- Pour le fichier de génération de serveur de périphérie, appelez le service suivant :
`/ibm/iotm/vi/service/edgeFile?edgeId=xxx&version=xxx&user=xxx`

&solution=vi

où `version` est `ubuntu`, `redhat` ou `power`, selon le système. Sauvegardez le fichier en tant que `vi_edge-bin_vi.zip`. Voici un exemple de commande :

```
curl -k -H "apikey:votre_clé_api" -o vi_edge-bin_vi.zip
```

```
"https://iotm.predictivesolutionsapps.ibmcloud.com/ibm/iotm/vi/service/edgeFile?
```

```
edgeId=1508123458000&version=ubuntu&user=votre_utilisateur&solution=vi"
```

5. Placez les fichiers `edgeDeployed.sh` et `vi_edge-bin_vi.zip` sur le système de périphérie.
6. Connectez-vous au serveur de périphérie en tant qu'utilisateur SSH qui a été indiqué lors de la création du serveur de périphérie dans l'application centrale.
7. Placez les fichiers `edgeDeployed.sh` et `vi_edge-bin_vi.zip` dans le répertoire `deployPath`.
8. Arrêtez le service de serveur de périphérie en cours.
9. Exécutez les commandes suivantes :

```
dos2unix edgeDeployed.sh
chmod +x edgeDeployed.sh
./edgeDeployed.sh
```
10. Assurez-vous que le service Python pour le serveur de périphérie a démarré.

Tâches associées

Vérification du statut des services sur les systèmes de périphérie

Une fois que vous avez enregistré le serveur de périphérie dans l'application centrale, vérifiez que les processus Python appropriés sont exécutés sur les systèmes de périphérie.

Chapitre 3. Création et utilisation de modèles

Vous créez des modèles pour collecter des images d'historique et des informations sur les défauts. Les informations sont utilisées pour entraîner le modèle. Une fois le modèle entraîné, il doit être validé pour pouvoir être déployé sur un serveur de périphérie. La validation du modèle fournit des informations sur sa précision. Il peut exister plusieurs versions d'un modèle. Les modèles peuvent partager des informations sur les défauts mais comporter des fichiers image différents provenant de lignes de produits différentes. Vous pouvez procéder au nouvel entraînement d'un modèle pour essayer d'obtenir une meilleure précision du modèle, afin de pouvoir remplacer la version de modèle par une version de modèle plus récente. Il existe deux types d'implémentation de modèle : le modèle de classification et le modèle de détection d'objet.

Structure des fichiers image compressés

Afin de pouvoir ajouter des images d'historique pour des groupes d'images, vous devez disposer de fichiers qui contiennent les fichiers image dont vous avez besoin pour le modèle de classification ou le modèle de détection d'objet.

Modèle de classification

Ajoutez les images dans des fichiers compressés. Un fichier compressé doit contenir toutes les images appartenant à un même groupe d'images. Vous devez placer toutes les images dans une structure à plat sans sous-dossier dans le fichier compressé. Les types d'image suivants sont pris en charge : PNG, JPEG et JPG. Les extensions des fichiers image doivent être entièrement en majuscules (PNG, JPEG, JPG) ou entièrement en minuscules (png, jpeg, jpg).

Modèle de détection d'objet

La structure du modèle de détection d'objet doit contenir deux dossiers dans un fichier compressé. L'un des dossiers doit être nommé JPEGImages et l'autre Annotations. En plus des deux dossiers, le fichier compressé doit également contenir un fichier labels.txt.

Ajoutez tous les fichiers image dans le dossier JPEGImages. Les types d'image suivants sont pris en charge : PNG, JPEG et JPG. Les extensions des fichiers image doivent être entièrement en majuscules (PNG, JPEG, JPG) ou entièrement en minuscules (png, jpeg, jpg). Ajoutez tous les fichiers d'annotation dans le dossier Annotations. Un fichier d'annotations doit avoir le même nom de fichier que son fichier image. Les fichiers doivent être au format XML. Si vous utilisez l'outil d'affectation de libellés d'image dans Maximo PQI SaaS Visual Insights, la sortie est un fichier image compressé contenant le fichier d'images et le fichier d'annotations. Si vous utilisez un outil d'affectation de libellés d'image externe, vous devez vous assurer que le fichier d'images et le fichier d'annotations se trouvent dans la structure prévue. Voici un exemple de fichier d'annotation :

```
<annotation>
  <folder>JPEGImages</folder>
  <filename>000001.jpg</filename>
  <source>
    <database>Unknown</database>
  </source>
  <size>
    <width>864</width>
    <height>1296</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>defect1</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>474</xmin>
```

```
<ymin>368</ymin>
<xmax>540</xmax>
<ymin>448</ymin>
</bndbox>
</object>
<object>
<name>defect2</name>
<pose>Unspecified</pose>
<truncated>0</truncated>
<difficult>0</difficult>
<bndbox>
<xmin>303</xmin>
<ymin>387</ymin>
<xmax>369</xmax>
<ymin>452</ymin>
</bndbox>
</object>
</annotation>
```

Le fichier `labels.txt` contient les noms de tous les types de défaut se trouvant dans le dossier d'annotations. Chaque défaut doit se trouver sur une ligne distincte, conformément à l'exemple suivant :

```
defect1
defect2
defect3
```

Ajout d'images d'historique pour les groupes d'images

Le gestionnaire de modèles utilise des images d'historique pour entraîner le modèle.

Pourquoi et quand exécuter cette tâche

Vous pouvez transférer des fichiers image compressés (.zip) dans le groupe d'images. Préparez les fichiers image compressés avec une taille de fichier correcte. Si un fichier est trop volumineux, vous pouvez le diviser en plusieurs fichiers. En fonction de la vitesse de transfert de votre réseau vers l'application, assurez-vous que chaque fichier image compressé peut être transféré en moins de 30 minutes. Par exemple, si la vitesse de transfert de votre réseau vers l'application est de 100 ko/s, la taille maximale d'un fichier image compressé est de 200 Mo.

Procédure

1. Sélectionnez **Données > Groupes d'images > Nouveau groupe d'images**. Sélectionnez **Nouveau groupe d'images sans libellé** si vous voulez utiliser l'outil d'affectation de libellés dans Maximo PQI SaaS Visual Insights pour libeller les images. Sélectionnez **Nouveau groupe d'images avec libellés** si les images ont déjà été libellées par un outil externe.
2. Ajoutez un nom de groupe d'images unique et une description, sélectionnez le type de groupe d'images, puis cliquez sur **Suivant**.

Pour le type de groupe d'images :

- L'option **Caractéristiques uniques** signifie que les images du groupe appartiennent à un type de défaut. Elle utilise le modèle de classification.
- L'option **N'est pas un défaut** signifie que les images dans le groupe présentent une caractéristique unique et ne présentent pas de défaut. Elle utilise le modèle de classification.
- L'option **Caractéristiques multiples** signifie que les images contiennent chacune un ou plusieurs défauts appartenant au même type de défaut ou à des types de défaut différents. Elle utilise le modèle de détection d'objet.

Remarque : Une fois que l'ensemble d'images a été transféré, le type de groupe d'images que vous avez sélectionné ne peut plus être changé.

3. Dans la sous-fenêtre **Ensembles d'images**, ajoutez des images et sélectionnez **Ajouter un groupe d'images**.

Résultats

Dans la sous-fenêtre **Groupes d'images**, vous pouvez sélectionner un groupe d'images, puis cliquer sur Editer. Vous pouvez ajouter ou supprimer des ensembles d'images, mettre à jour le nom ou la description d'un groupe d'images, ou changer le type du groupe d'images.

Remarque : Si l'ensemble d'images est cité dans une instance de modèle, il ne peut pas être supprimé.

Outil d'affectation de libellés d'image

L'outil d'affectation de libellés d'image peut être utilisé pour affecter aux images des libellés de manière semi-automatique. Vous pouvez transférer des fichiers d'image sans libellé, affecter manuellement un libellé à certaines images, puis déclencher l'affectation automatique de libellés pour affecter un libellé aux images qui n'en possèdent pas. Vous pouvez vérifier le résultat de l'affectation automatique de libellés. En fonction des informations récapitulatives, vous pouvez accepter le résultat de l'affectation de libellés. Les images reçoivent un libellé et sont conditionnées dans des fichiers d'image zip correspondants avant d'être automatiquement jointes aux groupes d'images pour qu'elles puissent être utilisées dans le processus d'entraînement des modèles.

Création d'un groupe d'images sans libellé

Vous pouvez transférer des fichiers image compressés (.zip) qui contiennent des images sans libellé pour créer un groupe d'images sans libellé.

Pourquoi et quand exécuter cette tâche

Lorsque vous créez un groupe d'images sans libellé, vous pouvez utiliser un fichier compressé qui ne contient aucune image avec libellé ou un fichier compressé qui contient à la fois des images avec et sans libellé. Pour les fichiers à caractéristiques uniques, les images avec libellé doivent être contenues dans un sous-dossier où le nom du dossier correspond au nom du défaut. Pour les fichiers à caractéristiques multiples, le fichier image avec libellé doit posséder un fichier XML d'annotations correspondant dans le même dossier.

Préparez les fichiers image compressés avec une taille de fichier correcte. Si un fichier est trop volumineux, vous pouvez le diviser en plusieurs fichiers. En fonction de la vitesse de transfert de votre réseau vers l'application, assurez-vous que chaque fichier image compressé peut être transféré en moins de 30 minutes. Par exemple, si la vitesse de transfert de votre réseau vers l'application est de 100 ko/s, la taille maximale d'un fichier image compressé est de 200 Mo.

Procédure

1. Sélectionnez **Données > Nouveau groupe d'images > Nouveau groupe d'images sans libellé**.
2. Dans la boîte de dialogue **Nouveau groupe d'images sans libellé**, spécifiez un nom pour le groupe d'images.
3. Spécifiez le type de groupe d'images :
 - L'option **Caractéristiques uniques** signifie que les images du groupe ne possèdent qu'un type de défaut.
 - L'option **Caractéristiques multiples** signifie que les images du groupe contiennent un ou plusieurs types de défaut.
4. Cliquez sur le lien de transfert pour transférer un fichier compressé qui contient des images sans libellé.
5. Cliquez sur **Ajouter un groupe d'images** pour créer le nouveau groupe d'images sans libellé.
6. Dans la boîte de dialogue **Règles d'échantillonnage**, définissez le ratio des images dont le libellé doit être créé manuellement, puis cliquez sur **Confirmer**.

Résultats

Une fois que vous avez effectué cette tâche, vous pouvez commencer à affecter manuellement un libellé aux échantillons d'image.

Affectation manuelle des noms d'échantillon d'image

Vous pouvez affecter manuellement les noms des images contenues dans un groupe d'images sans libellé.

Procédure

1. Dans la fenêtre **Tous les groupes d'images sans libellé**, cliquez sur **Afficher** en regard du groupe qui contient les images auxquelles vous devez affecter un libellé.
2. Cliquez sur l'icône des règles d'échantillonnage pour définir la fréquence d'échantillonnage, qui indique le nombre d'images auxquelles vous devez affecter manuellement un libellé. Cliquez sur **Confirmer**.
3. Pour chaque échantillon d'image, affectez manuellement un nom à l'image en sélectionnant le libellé correspondant, puis sauvegardez l'image. Vous pouvez choisir des libellés existants ou en créer des nouveaux. Pour les groupes à caractéristiques uniques, un libellé correspond à un groupe de données. Pour les groupes à caractéristiques multiples, un groupe de données de même nom est créé. Tous les libellés utilisés dans le groupe sans libellé sont sauvegardés comme balises dans le groupe de données correspondant.
4. Une fois qu'un libellé a été affecté à chacun des échantillons d'image, vous pouvez cliquer sur **Exécuter l'affectation automatique de libellés** pour lancer le processus d'affectation automatique de libellés.

Affectation automatique des noms d'échantillon d'image

Maximo PQI SaaS Visual Insights utilise des techniques d'apprentissage en profondeur reposant sur un réseau de neurones convolutif pour affecter automatiquement des libellés aux images.

Procédure

1. Dans la fenêtre **Tous les groupes d'images sans libellé**, cliquez sur **Afficher** en regard du groupe qui contient les images auxquelles vous devez affecter un libellé.
2. Cliquez sur **Exécuter l'affectation automatique de libellés** pour lancer le processus d'affectation automatique de libellés. Vous pouvez cliquer sur **Exécution en arrière-plan** pour exécuter le processus d'affectation automatique de libellés en arrière-plan.
3. Une fois ce processus terminé, cliquez sur **Afficher** pour examiner les résultats.
4. Pour chaque image dont le libellé a été affecté automatiquement, confirmez ou modifiez le libellé.
5. Si nécessaire exécutez à nouveau le processus d'affectation automatique de libellés.
6. Cliquez sur **Afficher le récapitulatif** pour afficher le libellé et les informations sur la précision du modèle.
7. Une fois que vous êtes satisfait de la précision du modèle, cliquez sur **Accepter**. Toutes les images auxquelles un libellé a été affecté sont sauvegardées sous forme de fichier compressé avec libellé. Vous pouvez retrouver ce fichier dans la sous-fenêtre **Données**.

Création de modèle

Pour obtenir une précision de modèle convenable, vous devez définir des hyperparamètres soigneusement en fonction de vos ensembles d'images et du type de modèle sélectionné.

Lorsque vous créez un modèle, vous pouvez choisir les paramètres de modèle recommandés ou décider de les personnaliser. Les paramètres de modèle recommandés ne sont pas adaptés à tous les scénarios. Ils constituent simplement une référence. Il peut être nécessaire de les personnaliser.

Lorsque vous entraînez un modèle, vous devez sélectionner les ensembles d'images requis. Certaines images sont utilisées en tant qu'ensembles de données d'apprentissage et d'autres en tant qu'ensembles de données de validation. Le rapport dépend de la règle d'échantillonnage de l'entraînement et de la validation. Par défaut, 80 % des images sont utilisées pour l'entraînement et 20 % pour la validation.

Il est utile de comprendre la terminologie suivante liée au réseau de neurones :

Epoque

Une époque est égale à un cycle d'entraînement complet sur toutes les images d'entraînement.

Taille de lot

Nombre d'exemples d'entraînement d'une passe. Plus la taille de lot est élevée, plus l'espace mémoire dont vous avez besoin est élevé.

Itération

Le nombre d'itérations est égal au nombre de passes, chaque passe utilisant *taille_lot* nombre d'images.

Par exemple, si vous disposez de 1000 images d'entraînement et que la taille de lot est égale à 50, 20 itérations sont nécessaires pour terminer 1 époque. Si vous définissez l'époque sur 10, 200 itérations sont nécessaires pour mener à bien l'entraînement.

Si vous sélectionnez le type de modèle classification, la taille de lot est prédéfinie dans la définition réseau. Pour GoogLeNet, la taille de lot d'entraînement est 32 et la taille de lot de test est 16. Pour AlexNet, la taille de lot d'entraînement est 128 et la taille de lot de test est 32. Pour LeNet, la taille de lot d'entraînement est 64 et la taille de lot de test est 32. Vous pouvez ajuster les hyperparamètres en fonction de la taille de lot, du nombre total d'images et de la règle d'échantillonnage.

Création de modèles

Une fois les types de défaut ajoutés, le gestionnaire de modèles crée un modèle. Celui-ci peut être sauvegardé en tant que brouillon ou entraîné immédiatement.

Pourquoi et quand exécuter cette tâche

Vous pouvez consulter les détails d'un modèle dans l'onglet **Tous les modèles**. Les détails d'un modèle incluent les versions du modèle. Différentes versions de modèle sont générées par différents ensembles d'images.

Les noms de modèle doivent être uniques.

Procédure

1. Sélectionnez **Créer un modèle** > **Créer un modèle**.
2. Dans l'onglet **Dispositions générales**, mettez à jour les informations.

Le type de produit est utilisé pour mapper le modèle. Vous choisissez le type de produit hors de Maximo PQI SaaS Visual Insights en fonction des normes de votre société. Le type de produit est le lien entre le modèle et la ligne de production. Le contrôleur de périphérie sélectionne le modèle en fonction des informations de type de produit qui figurent dans l'image.

3. Dans l'onglet **Type de modèle**, mettez à jour les informations de type de modèle. Dans la zone **Type de modèle**, vous pouvez spécifier la classification ou la détection de plusieurs objets, ou choisir de transférer un modèle entraîné existant. Si vous sélectionnez la classification ou la détection de plusieurs objets, vous pouvez sélectionner les paramètres de modèle recommandés ou décider de les personnaliser.

Si vous choisissez de personnaliser les paramètres de modèle, spécifiez la méthode de détection, les hyperparamètres et les règles d'échantillonnage. Chaque hyperparamètre possède une portée de valeur valide. Si vous entrez une valeur qui n'est pas comprise dans cette portée, un message d'erreur s'affiche. En raison de la limitation des ressources, une valeur maximale est définie pour l'époque, l'itération et la taille de lot. La valeur maximale pour l'itération est liée au type de paiement du titulaire. Pour les utilisateurs disposant d'un abonnement payant, la valeur maximale de l'itération est 40000

pour FRCNN, 20000 pour YOLO ou 5000 pour SSD. Pour les utilisateurs se servant d'une version d'essai, la valeur maximale de l'itération est 3000 pour FRCNN, 2000 pour YOLO ou 500 pour SSD. La taille de lot de YOLO et de SSD est 32. Pour CNN, la valeur d'époque maximale est 30000 pour les utilisateurs disposant d'un abonnement payant et 5000 pour les utilisateurs se servant d'une version d'essai.

Si vous transférez un fichier compressé de modèle entraîné existant, les valeurs des paramètres de modèle sont lues depuis le fichier et affichées dans la boîte de dialogue. Si la valeur de l'itération ou de l'époque et la taille de lot dépassent les valeurs maximales, elles sont remplacées par les valeurs maximales.

4. Dans l'onglet **Ensembles d'images**, sélectionnez les ensembles d'images à utiliser pour entraîner le modèle.
5. Dans l'onglet **Règles globales**, définissez les paramètres de règle de nouvel entraînement, de règle d'échantillonnage et de règle d'inspection manuelle.
6. Dans l'onglet **Récapitulatif**, sélectionnez **Sauvegarder en tant que brouillon** pour sauvegarder le modèle ou **Entraîner** pour l'entraîner.
7. Dans la sous-fenêtre **Tous les modèles**, affichez le modèle. Si la version du modèle est Brouillon, vous pouvez cliquer sur **Afficher** pour afficher les détails de l'instance de modèle. Vous pouvez éditer les groupes d'images avant d'entraîner le modèle.

Utilisation du catalogue de modèles

Le catalogue de modèles est une bibliothèque de modèles partagés qui peuvent être réutilisés pour des solutions métier et partagés par des titulaires.

Le catalogue de modèles contient les modèles ci-dessous.

Domaine	Nom du modèle	Description
Electronique - Fabrication de matériel	Server Hardware Defect Inspection	Ce modèle de détection d'objet détecte les composants manquants pour les serveurs IBM. Il prend en charge sept défauts majeurs, à savoir missingChinLabel, missingPowerLabel, missingIBMLogo, missingDASD, incorrectDASD, missingCover et missingPowerButton.
Automobile - Usure de la voiture	Car Wearout Defect Inspection	Ce modèle de détection d'objet détecte les cas d'usure. Les constructeurs automobiles testent des dizaines de milliers d'activités d'ouverture et de fermeture et détectent les usures dont le niveau n'est pas acceptable. Ce modèle a pour but de détecter les éraflures.
Automobile - Siège de voiture	Car Seat Defect Inspection	Ce modèle de détection d'objet détecte les irrégularités sur les sièges de voiture. Le résultat de l'inspection identifie si un siège de voiture présente une irrégularité et situe le défaut sur l'image.

Test des modèles dans le catalogue de modèles

Vous pouvez tester des modèles dans le catalogue de modèles en transférant votre propre image et en prenant connaissance du résultat d'évaluation. Vous pouvez transférer plusieurs images et consulter les résultats d'évaluation individuellement.

Procédure

1. Sélectionnez **Créer un modèle** > **Parcourir le catalogue de modèles**.
2. Dans la page **Catalogue de modèles**, sélectionnez le modèle partagé à utiliser.
3. Cliquez sur **Transférer plus** pour transférer une image. Le modèle évalue l'image et affiche le résultat.

Création de modèles à l'aide du catalogue de modèles

Vous pouvez créer des modèles en utilisant des modèles partagés du catalogue de modèles. Pour utiliser un modèle partagé, importez la définition de modèle et utilisez vos propres ensembles de données pour entraîner votre modèle.

Procédure

1. Sélectionnez **Créer un modèle** > **Parcourir le catalogue de modèles**.
2. Dans la page **Modèles partagés**, sélectionnez le modèle partagé à utiliser.
3. Cliquez sur **Utiliser un modèle**.
4. Dans l'onglet **Dispositions générales**, mettez à jour les informations. Le type de produit est utilisé pour mapper le modèle.
5. Dans l'onglet **Type de modèle**, mettez à jour les informations de type de modèle et de paramètre de modèle.
6. Dans l'onglet **Ensembles d'images**, sélectionnez les ensembles d'images à utiliser pour entraîner le modèle.
7. Dans l'onglet **Règles globales**, mettez à jour la règle de nouvel entraînement et la règle d'inspection manuelle si nécessaire.
8. Dans l'onglet **Récapitulatif**, cliquez sur **Sauvegarder en tant que brouillon**. Le statut du modèle est Brouillon.

Entraînement des modèles

Maximo PQI SaaS Visual Insights utilise des techniques d'apprentissage en profondeur reposant sur un réseau de neurones convolutif pour entraîner les modèles. Lorsque vous entraînez un modèle, le modèle apprend les caractéristiques des images libellées et génère un modèle exécutable intégrant ces connaissances. Un modèle entraîné peut appliquer la classification ou la détection d'objet dans des images de la ligne de produits.

Pourquoi et quand exécuter cette tâche

Vous pouvez entraîner un nouveau modèle ou un modèle qui a été créé à partir d'un modèle existant dans le catalogue de modèles. Vous pouvez entraîner des modèles dont le statut est Brouillon, Echec ou Rejeté. Au cours de l'entraînement, vous pouvez consulter le statut de l'entraînement et les fichiers journaux. Une fois l'entraînement terminé, vous pouvez consulter la courbe d'entraînement qui présente les valeurs de perte de la phase d'entraînement, de perte de la phase de test et de précision de la phase de test.

Procédure

1. Dans la sous-fenêtre **Tous les modèles**, sélectionnez un modèle.
2. Cliquez sur **Afficher** pour afficher les détails de l'instance de modèle.
3. Facultatif : changez les groupes d'images qui sont utilisés par le modèle.

4. Cliquez sur **Entraîner** pour démarrer l'entraînement.

Le statut de l'entraînement devient En attente étant donné que les données sont préparées et que le travail d'entraînement est mis en file d'attente sur le serveur. Lorsque l'entraînement commence, le statut devient Entraînement. Une fois l'entraînement terminé, le statut devient Entraîné.

5. Une fois l'entraînement terminé, prenez connaissance des résultats de l'entraînement dans le **tableau de bord d'entraînement**. Si le modèle prend en charge les instantanés, ceux-ci sont affichés dans le tableau de bord. Vous pouvez consulter des informations pour chaque instantané. Vous pouvez utiliser un instantané en le sélectionnant et en cliquant sur **Utiliser**. Vous pouvez sélectionner un instantané intermédiaire à utiliser comme fichier de modèle final. Vous pouvez aussi afficher et télécharger les fichiers journaux de l'entraînement.

Modèles entraînés

Une fois qu'un modèle a été créé ou édité, le gestionnaire de modèles peut l'entraîner. Une fois que le modèle a été entraîné, son statut devient Entraîné. Le gestionnaire de modèles valide le modèle entraîné et l'accepte ou le rejette. Il valide le modèle à l'aide d'ensembles d'images de validation.

Si le gestionnaire de modèles choisit de transférer le modèle entraîné lors de la création d'un modèle, le statut du modèle devient Accepté à la fin du processus de transfert.

Structure des fichiers de modèle

Maximo PQI SaaS Visual Insights prend en charge les types de modèle de classification du réseau neuronal convolutif (CNN) et de détection d'objet.

Modèles de classification CNN (réseau neuronal convolutif) pour Caffe

Le modèle de classification CNN (réseau neuronal convolutif) doit être un fichier compressé unique qui contient la structure de répertoire et les fichiers requis.

Fichier de modèle compressé

Le fichier de modèle compressé doit contenir les répertoires et les fichiers suivants :

- `model.config` (fichier, requis)
- `sink.config` (fichier, requis)
- `parameter.config` (fichier, facultatif)
- `cnet1` (répertoire). Le répertoire `cnet1` doit contenir les fichiers suivants :
 - `labels.txt` (fichier, requis)
 - `deploy.prototxt` (fichier, requis)
 - `mean.binaryproto` (fichier, requis)
 - `info.json` (fichier, facultatif)
 - `snapshot.caffemodel` (fichier, requis)
 - `solver.prototxt` (fichier, requis)
 - `train_val.prototxt` (fichier, requis)

Chaque fichier doit avoir la structure appropriée et comporter les mots clés corrects. Les fichiers sont décrits dans les sections ci-dessous.

model.config

Le texte ci-dessous est un exemple de contenu du fichier `model.config`. Les mots clés apparaissent en gras.

```
submodel{  
  module {  
    type: "ChipROIExtractor"  
    ref_file: "parameter.config"  }  
}
```

```

}
module {
  type:"ClassificationNet"
  net_name:"cnet1"
}
}
sink{
  type:"SinkFilter"
  ref_file:"sink.config"
}

```

Ce fichier doit contenir au moins un module du type **ClassificationNet** dans le sous-modèle. Le mot clé `ref_file` désigne les autres fichiers de configuration dans le fichier compressé du modèle de classification CNN. Le mot clé `net_name` référence le nom du dossier contenant le modèle CNN. Il n'est pas nécessaire de changer les informations `sink` sauf si votre fichier `sink.config` a un nom différent.

sink.config

Le contenu du fichier `sink.config` est le suivant. Il n'est pas nécessaire de l'éditer.

```

keyword:"position"
keyword:"probableTypes"

```

labels.txt

Le fichier `labels.txt` contient tous les noms de classe en fonction desquels ce modèle de classification est classifié. Chaque nom de classe doit se trouver sur une ligne distincte, conformément à l'exemple ci-dessous.

```

defect1
defect2
defect3

```

Autres fichiers

Tous les fichiers `nom_fichier.prototxt` sont des fichiers de définition de modèle qui sont requis lorsque vous entraînez un modèle CNN.

Les fichiers `snapshot.caffemodel` et `mean.binaryproto` sont des fichiers de sortie qui sont créés une fois l'entraînement du modèle terminé.

Modèles de classification CNN (réseau neuronal convolutif) pour TensorFlow

Le modèle de classification CNN (réseau neuronal convolutif) qui est entraîné avec TensorFlow doit être un fichier compressé unique qui contient la structure de répertoire et les fichiers requis.

Fichier de modèle compressé

Le fichier de modèle compressé doit contenir les répertoires et les fichiers suivants :

- `model.config` (fichier, requis)
- `source.config` (fichier, requis)
- `cnet1.config` (fichier, requis)
- `cnet1` (répertoire). Le répertoire `cnet1` doit contenir les fichiers suivants :
 - `labels.txt` (fichier, requis)
 - `checkpoint` (fichier, requis)
 - `model.ckpt-10000.data-00000-of-00001` (fichier, requis)
 - `model.ckpt-10000.index` (fichier, facultatif)
 - `model.ckpt-10000.meta` (fichier, requis)

Chaque fichier doit avoir la structure appropriée et comporter les mots clés corrects. Les fichiers sont décrits dans les sections ci-dessous.

model.config

Le texte ci-dessous est un exemple de contenu du fichier `model.config`. Les mots clés apparaissent en gras.

```
ipc_buffer_size:80000000
use_ipc_buffer:1
gpu_runnable:1

input_field_mapper {
  entry {
    app_field: "imageFile"
    model_field: "image"
  }
}
source {
  type:"utility.ImageFileSource"
  name:"source"
}
sink{
  type:"utility.Sink"
  name:"sink"
  port_num_config{
    input_num:1
  }
}
module {
  type:"tf_classification.TF_ClassificationNet"
  name:"cnet1"
}
connection{
  from_module:"source"
  from_port:"0"
  to_module:"cnet1"
  to_port:"0"
}
connection{
  from_module:"cnet1"
  from_port:"0"
  to_module:"sink"
  to_port:"0"
}
process{
  module_group {
    module:"cnet1"    }
  gpu_runnable:1
}
```

Ce fichier doit contenir au moins un module du type **tf_classification.TF_ClassificationNet** dans **input_field_mapper**. Le nom de module, par exemple `cnet1`, référence le nom du dossier contenant le modèle CNN. Vous pouvez ajouter des modules en fonction de vos besoins. Il n'est pas nécessaire de changer les informations source sauf si vous avez attribué un autre nom à votre fichier `source.config`. Si vous changez le nom de la source, vous devez changer le nom du module dans le paramètre `connection` en conséquence.

source.config

Le contenu du fichier `source.config` est le suivant. Il n'est pas nécessaire de l'éditer.

```
param {
  name: "image"
  value: "color"
}
param {
  name: "gray_image"
  value: "gray"
}
```

cnet1.config

Le contenu du fichier `cnet1.config` est le suivant. Le nom indiqué par **param** dans ce fichier de configuration ne peut pas être changé. La valeur est le nom de variable et la taille d'image que vous avez définis lorsque vous avez entraîné le modèle.

```
param {
  name: "input_name"
  value: 'input'
}
param {
  name: "output_name"
  value: 'InceptionV3/Predictions/Softmax'
}
param {
  name: "resize_width"
  value: '299'
}
param {
  name: "resize_height"
  value: '299'
}
```

labels.txt

Le fichier `labels.txt` contient tous les noms de classe en fonction desquels ce modèle de classification est classifié. Chaque nom de classe doit se trouver sur une ligne distincte, conformément à l'exemple ci-dessous.

```
defect1
defect2
defect3
```

checkpoint

Le contenu du fichier `checkpoint` est le suivant. Ce fichier est généré automatiquement lorsque vous entraînez le modèle.

```
model_checkpoint_path: "model.ckpt-10000"
all_model_checkpoint_paths: "model.ckpt-10000"
```

Modèle de détection d'objet

Le modèle de détection d'objet doit être un fichier compressé unique qui contient la structure de répertoire et les fichiers requis. Les algorithmes de détection d'objet suivants sont pris en charge : Faster Region-based Convolutional Neural Network (Faster R-CNN), You Only Look Once (YOLO) V2 et Single Shot Multibox Detector (SSD).

Faster R-CNN

Le modèle de détection d'objet Faster R-CNN doit être un fichier compressé unique qui contient la structure de répertoire et les fichiers requis.

Fichier de modèle compressé

Le fichier de modèle compressé doit contenir tous les fichiers suivants :

- `labels.txt`
- `faster_rcnn_final.caffemodel`
- `model.config`
- `stage1_fast_rcnn_solver30k40k.pt`
- `stage1_fast_rcnn_train.pt`
- `stage1_rpn_solver60k80k.pt`
- `stage1_rpn_train.pt`

- stage2_fast_rcnn_solver30k40k.pt
- stage2_fast_rcnn_train.pt
- stage2_rpn_solver60k80k.pt
- stage2_rpn_train.pt
- faster_rcnn_test.pt
- rpn_test.pt

Chaque fichier doit avoir la structure appropriée et comporter les mots clés corrects. Les fichiers sont décrits dans les sections ci-dessous.

labels.txt

Le fichier labels.txt contient les noms de tous les objets que ce modèle de détection d'objet détecte. Chaque objet doit se trouver sur une ligne distincte, conformément à l'exemple ci-dessous.

```
defect1
defect2
defect3
```

faster_rcnn_final.caffemodel

Ce fichier contient la sortie une fois le modèle Faster R-CNN entraîné. Le nom de fichier doit correspondre à la définition de **model** dans model.config. Actuellement, deux réseaux sont pris en charge pour Faster R-CNN : ZF et VGG16. Pour ZF-net, le nom doit commencer par ZF, par exemple ZF_mobile_final.caffemodel. Pour VGG16-net, le nom doit commencer par VGG16, par exemple VGG16_mobile_final.caffemodel.

model.config

Les mots clés dans l'exemple suivant apparaissent en gras :

```
{
  "modelType": "FRCNN",
  "model": "faster_rcnn_final.caffemodel",
  "solvers":
"stage1_rpn_solver60k80k.pt,stage1_fast_rcnn_solver30k40k.pt,stage2_rpn_solver60k80k.pt,stage2_f
ast_rcnn_solver30k40k.pt",
  "net_file":
"stage1_rpn_train.pt,stage1_fast_rcnn_train.pt,stage2_rpn_train.pt,stage2_fast_rcnn_train.pt",
  "deploy_net": "faster_rcnn_test.pt",
  "parameters": {
    "iteration": "40000,80000,40000,80000",
    "learningRate": 0.001,
    "stepsize": "10000",
    "gamma": "0.1"
  }
}
```

La valeur de modelType est toujours FRCNN. La valeur de model est le nom du fichier de modèle. La valeur de solvers est la liste des fichiers solver qui sont utilisés au cours de l'entraînement du modèle. La valeur de net_file est la liste des fichiers de définition réseau. La valeur de deploy_net est le nom de la définition réseau d'évaluation. Les valeurs dans la section parameters correspondent à tous les hyperparamètres du modèle Faster R-CNN. La valeur d'iteration est une valeur de chaîne composée de quatre nombres séparés par une virgule, qui présente le nombre d'itérations des quatre stades du processus d'entraînement.

Fichiers *.pt

Les fichiers dont l'extension est .pt sont des fichiers de définition de modèle. Le modèle pascol_voc fourni par Faster R-CNN est pris en charge. Un exemple de fichier se trouve dans le répertoire d'installation Faster R-CNN models/pascal_voc/nom_reseau/faster_rcnn_alt_opt/, où nom_reseau est ZF ou VGG16.

YOLO V2

Le modèle de détection d'objet YOLO V2 doit être un fichier compressé unique qui contient la structure de répertoire et les fichiers requis.

Fichier de modèle compressé

Le fichier de modèle compressé doit contenir les fichiers suivants :

- labels.txt
- model.config
- yolo_final.weights
- Yolo.cfg

Chaque fichier doit avoir la structure appropriée et comporter les mots clés corrects. Les fichiers sont décrits dans les sections ci-dessous.

labels.txt

Le fichier labels.txt contient les noms de tous les objets que ce modèle de détection d'objet détecte. Chaque objet doit se trouver sur une ligne distincte, conformément à l'exemple ci-dessous.

```
defect1
defect2
defect3
```

model.config

Les mots clés dans l'exemple suivant apparaissent en gras :

```
{
  "modelType": "YOLO",
  "modelCfg": "Yolo.cfg",
  "model": "yolo_final.weights",
  "parameters": {
    "iteration": "40000",
    "batchSize": 16,
    "learningRate": 0.001,
    "subBatchSize": 2,
    "steps": "100,15000,25000,35000",
    "scales": "1,10,0.1,0.1"
  }
}
```

La valeur de modelType est toujours YOLO. La valeur de modelCfg est le nom du fichier de définition réseau d'apprentissage en profondeur. La valeur de model est le nom du fichier de modèle réel. Les valeurs dans la section parameters sont des hyperparamètres du modèle YOLO V2. La valeur de batchSize dans la section parameters ne peut pas être supérieure à 32.

yolo_final.weights

Ce fichier contient la sortie une fois le modèle YOLO entraîné. Le nom de fichier doit correspondre à la définition de **model** dans model.config.

Yolo.cfg

Ce fichier de définition de modèle inclut des définitions réseau, des hyperparamètres et des paramètres d'ancre. Un exemple de ce fichier figure dans le répertoire d'installation YOLO darknet/cfg/. Ce fichier doit correspondre au fichier weights.

SSD

Le modèle de détection SSD (Single Shot MultiBox Detector) doit être un fichier compressé unique qui contient la structure de répertoire et les fichiers requis.

Fichier de modèle compressé

Le fichier de modèle compressé doit contenir tous les fichiers suivants :

- labels.txt
- solver.prototxt
- deploy.prototxt
- model.config
- SSD.caffemodel

Chaque fichier doit avoir la structure appropriée et comporter les mots clés corrects. Les fichiers sont décrits dans les sections ci-dessous.

labels.txt

Le fichier labels.txt contient les noms de tous les objets que ce modèle de détection d'objet détecte. Chaque objet doit se trouver sur une ligne distincte, conformément à l'exemple ci-dessous.

```
defect1
defect2
defect3
```

solver.prototxt

Ce fichier contient tous les hyperparamètres pour le modèle SSD.

deploy.prototxt

Ce fichier contient la définition réseau du modèle entraîné.

model.config

Les mots clés dans l'exemple suivant apparaissent en gras :

```
{ "modelType": "SSD",
  "parameters": {
    "learningRate": 0.01,
    "iteration": 10000,
    "steps": "6000,8000",
    "batchSize": 16,
    "learningRatePolicy": "multistep",
    "display": 10,
    "snapshot": 1000
  }
}
```

La valeur de modelType est toujours SSD. Les valeurs dans la section parameters sont des hyperparamètres du modèle SSD.

SSD.caffemodel

Ce fichier contient la sortie une fois le modèle SSD entraîné.

Modèles validés

Une fois qu'un modèle a été entraîné, son statut devient Entraîné. Le gestionnaire de modèles peut valider la version de modèle et accepter ou rejeter le modèle entraîné. Il valide le modèle à l'aide d'ensembles d'images de validation. Une fois qu'un modèle a été validé, il peut être utilisé et déployé.

Vous pouvez valider une version de modèle dont le statut est Entraîné. Une version de modèle entraîné ou ayant fait l'objet d'un nouvel entraînement peut déclencher le processus de validation. Lorsque vous validez un rapport, vous devez gérer les ensembles d'images. Chaque groupe d'images doit comporter au moins un ensemble d'images de validation pour que le modèle puisse être validé. Vous devez utiliser des ensembles d'images et des ensembles d'images d'entraînement différents pour valider la version de modèle. Pour lancer le processus de validation, sélectionnez **Valider**.

Une fois le processus de validation terminé, un rapport indiquant la précision du modèle est généré. Vous pouvez créer les types de rapport suivants :

Rapport de modèle de classification

Dans le rapport de modèle de classification, une image contient un défaut maximum. La matrice de confusion est utilisée pour générer le rapport, dans lequel chaque colonne représente un type de groupe d'images réel dans les ensembles de données de validation. Chaque ligne représente le type de groupe d'images prévu. La dernière ligne du graphique représente les résultats agrégés.

Rapport de modèle de détection d'objet

Dans le rapport de modèle de détection d'objet, une image contient plusieurs défauts. Dans ce rapport, la précision moyenne et le rappel sont calculés afin d'indiquer la précision du modèle.

Pour afficher le rapport, sélectionnez **Afficher le rapport**. Dans la fenêtre **Rapport de validation**, vous pouvez revalider, rejeter ou accepter et déployer le modèle.

Si la première version de modèle est rejetée, vous pouvez changer les paramètres d'entraînement, puis entraîner à nouveau le modèle. Si une version de modèle qui n'est pas la première version est rejetée, vous pouvez entraîner à nouveau une nouvelle version de modèle en utilisant de nouveaux ensembles d'images.

Distribution de modèles entraînés sur des serveurs de périphérie

Une fois que le gestionnaire de modèles a accepté la version de modèle entraînée, le modèle est distribué sur des serveurs de périphérie pour pouvoir être inspecté.

Procédure

1. Sélectionnez une version de modèle dont le statut est Validé. Lorsque vous déployez un modèle pour la première fois, cliquez sur **Afficher le rapport** pour afficher le rapport de validation.
2. Cliquez sur **Accepter et publier** dans la page Rapport de validation.
3. La boîte de dialogue Publier et déployer s'ouvre. Sélectionnez les serveurs de périphérie sur lesquels déployer le modèle, puis cliquez sur **Publier et déployer** pour déployer le modèle.
4. Si le modèle a été déployé sur des serveurs de périphérie, cliquez sur **Gérer le déploiement** pour ouvrir la boîte de dialogue de déploiement. Sélectionnez les serveurs de périphérie sur lesquels procéder au déploiement ou annuler le déploiement, puis cliquez sur **Déployer** ou **Annuler le déploiement**.

Nouvel entraînement des modèles

Le nouvel entraînement d'un modèle crée une nouvelle version de modèle. Lorsque vous créez une demande de modèle, vous définissez la règle de nouvel entraînement. Celle-ci constitue la condition qui déclenche le nouvel entraînement automatique. En cas de doute sur la précision actuelle du modèle, le gestionnaire de modèles peut sélectionner manuellement les fichiers image afin de déclencher le processus de nouvel entraînement.

Procédure

1. Sélectionnez une version de modèle dont le statut est Déployé, puis sélectionnez **Nouvel entraînement**.
2. Gérez les ensembles d'images. Chaque groupe d'images doit comporter au moins un ensemble d'images de nouvel entraînement pour que le modèle puisse faire l'objet d'un nouvel entraînement.

Remarque : Pour un nouvel entraînement de la version de modèle, il est recommandé d'utiliser des ensembles d'images différents contenant davantage d'images.

3. Sélectionnez **Nouvel entraînement** pour lancer le processus de nouvel entraînement.

Que faire ensuite

Le gestionnaire de modèles vérifie que le modèle nouvellement entraîné est accepté. Il peut déployer la nouvelle version ; dans ce cas, l'ancienne version de modèle n'est plus déployée.

Utilisation de modèles avec un serveur de périphérie autonome

Vous utilisez des modèles avec un serveur de périphérie autonome pour évaluer des images dans la ligne de production, synchroniser les données avec le centre, et nettoyer les données transférées. Pour pouvoir utiliser des modèles avec un serveur de périphérie autonome, vous devez les publier et les déployer sur le serveur de périphérie autonome.

Publication des modèles

Avant de déployer manuellement un modèle sur un serveur de périphérie autonome, vous devez le publier dans l'application centrale.

Pourquoi et quand exécuter cette tâche

Le modèle doit déjà avoir été validé pour que vous puissiez le publier.

Procédure

1. Connectez-vous à l'interface utilisateur de l'application centrale et sélectionnez une instance de modèle.
2. Sélectionnez **Afficher le rapport** pour afficher le rapport de validation.
3. Sélectionnez **Accepter et publier** pour publier l'instance de modèle.

Résultats

Une fois le modèle publié, le statut du modèle devient Prêt pour le déploiement.

Déploiement d'une instance de modèle

Utilisez l'API pour déployer une instance de modèle sur le serveur de périphérie autonome.

Procédure

1. Obtenez tous les modèles publiés dans l'application centrale avec la commande suivante :
GET `https://<hôte_machine_périphérie>:8449/api/getAvailableModels`
2. Déployez une instance de modèle sur le serveur de périphérie autonome avec la commande suivante :
POST `https://<hôte_machine_périphérie>:8449/api/deployModel`
Utilisez le corps qui est renvoyé par l'API `getAvailableModels`.

Que faire ensuite

Pour plus d'informations sur les détails de l'API, voir les rubriques relatives aux services de serveur de périphérie autonomes.

Annulation du déploiement des modèles

Utilisez l'API pour annuler le déploiement de modèles sur le serveur de périphérie autonome.

Procédure

Annulez le déploiement de modèle avec la commande suivante :

POST https://<hôte_machine_périphérie>:8449/api/undeployModel

Utilisez le corps suivant :

```
{"model_id": "id_modèle", "model_instance_id": "id_instance_modèle"}
```

Que faire ensuite

Pour plus d'informations sur les détails de l'API, voir les rubriques relatives aux services de serveurs de périphérie autonomes.

Chapitre 4. Consultation des résultats d'inspection

Une fois que les résultats d'inspection ont été envoyés à l'application centrale, l'inspecteur et le superviseur d'inspecteur peuvent accéder à l'onglet Vérification de défaut pour afficher et filtrer les résultats d'inspection et apporter les modifications nécessaires.

Images

L'inspecteur et le superviseur d'inspecteur peuvent afficher des images pour déterminer si elles sont classées comme défauts existants ou non, et pour savoir si quelqu'un d'autre les a consultées. En affichant les images, l'inspecteur ou le superviseur peut déterminer quelles sont les opérations à effectuer lorsqu'il vérifie les défauts.

L'inspecteur affiche des images non vérifiées et vérifiées. Les images non vérifiées sont des images qui ont uniquement été évaluées par le modèle et qui n'ont pas été vérifiées par un inspecteur. Les images vérifiées sont des images qui ont été évaluées par le modèle et qui ont déjà été vérifiées par un inspecteur.

L'inspecteur affiche des images non confirmées et des images confirmées. Les images non confirmées sont des images qui ont uniquement été évaluées par le modèle et qui n'ont pas été confirmées par un inspecteur. Les images confirmées sont des images qui ont été évaluées par le modèle et confirmées par un inspecteur.

Le superviseur d'inspecteur peut afficher des objets, même s'ils sont inconnus. Les objets inconnus sont des images qui ont été signalées par un inspecteur comme défaut inconnu car il n'a pas pu les classer comme défaut existant. Ces images sont mises en évidence dans la liste du superviseur d'inspecteur.

Filtrage des défauts

L'inspecteur et le superviseur d'inspecteur peuvent appliquer des filtres à la présentation des cellules et à la liste de défauts.

Procédure

1. Dans la fenêtre **Tous les postes de travail**, sélectionnez un poste de travail afin d'afficher la liste des objets non confirmés, confirmés et inconnus.
2. Cliquez sur l'icône de filtre.
3. Entrez une valeur pour une condition afin de définir le filtre, puis sélectionnez l'icône d'ajout. Le filtre est appliqué à la liste immédiatement.

Vérification des défauts

L'inspecteur et le superviseur d'inspecteur consultent les résultats de l'inspection et apportent les modifications nécessaires.

Pourquoi et quand exécuter cette tâche

Lorsque vous sélectionnez une image, le défaut candidat et la fiabilité correspondante s'affichent. Le premier défaut est sélectionné par défaut. L'inspecteur peut sélectionner "inconnu" comme type de défaut si le type de défaut est inconnu.

Procédure

1. Sélectionnez une image pour afficher ses détails et les résultats de l'inspection.

2. Sélectionnez **Editer le zoom** pour effectuer un zoom avant ou arrière sur l'image ou faites glisser l'image afin de localiser une position.
3. Sélectionnez **Définir le zoom** pour repasser en mode édition. Vous pouvez ajouter, redimensionner, déplacer et afficher les détails d'un cadre de défaut.
4. Sélectionnez le cadre de défaut à confirmer et affichez les détails du type de défaut et du niveau de fiabilité. Vous pouvez changer le type de défaut ou supprimer la position.
5. Cliquez sur **Confirmer**.
6. Si l'image n'appartient à aucun défaut existant, le superviseur d'inspecteur peut créer un groupe d'images. Le nouveau défaut est ajouté à la liste des candidats pour les images qui se trouvent sous le même modèle.

Transfert d'images à l'aide du simulateur

Vous pouvez utiliser le simulateur pour envoyer manuellement des images à un serveur de périphérie. Ce processus simule l'envoi d'images depuis la ligne de produits à un serveur de périphérie. Vous pouvez envoyer vos propres images ou utiliser des images prédéfinies sur le serveur.

Pourquoi et quand exécuter cette tâche

Procédure

1. Sélectionnez **Simulateur** dans le menu principal.
2. Spécifiez un type de produit et une cellule.
3. Pour transférer votre propre image, sélectionnez **Utiliser ma propre image** et accédez à votre image. Ou bien, pour utiliser des images qui se trouvent sur le serveur, désélectionnez **Utiliser ma propre image** et spécifiez le nombre total d'images et la fréquence d'envoi.
4. Cliquez sur **Démarrer**.
5. Cliquez sur **Afficher le résultat de l'analyse**.

Chapitre 5. Tableau de bord des indicateurs clés de performance

Le superviseur d'inspecteur utilise le tableau de bord des indicateurs clés de performance pour vérifier le taux de défaut au niveau des images et le taux de défaut au niveau des emplacements. Ces indicateurs peuvent fournir les informations dont vous avez besoin pour demander à l'équipe des technologies de l'information de procéder à un nouvel entraînement du modèle ou d'ajuster la procédure de fabrication.

Le tableau de bord des indicateurs clés de performance se trouve dans l'onglet **Indicateur clé de performance**. Vous pouvez sélectionner tous les postes de travail ou un poste de travail spécifique. Cette sélection a un impact sur la portée que vous utilisez. Vous pouvez également basculer entre les vues en temps réel et les vues historiques. Dans la vue en temps réel, les données des indicateurs clés de performance sont actualisées toutes les cinq secondes. Les valeurs des indicateurs clés de performance incluent le nombre de défauts par unité et le taux de défaut. Le nombre de défauts par unité est un nombre de défauts spécifique divisé par le nombre total d'images. Il représente la fréquence d'occurrence d'un type de défaut. Le taux de défaut est le nombre d'images contenant un ou plusieurs défauts divisé par le nombre total d'images. Le taux de défaut représente le taux de défaut du produit. Chaque ligne du graphique représente la valeur d'indicateur clé de performance dans l'intervalle de cinq secondes en cours. La vue historique présente les données d'indicateur clé de performance historiques. Vous pouvez éditer la date de début et la date de fin pour déterminer l'intervalle de temps. Les indicateurs clés de performance dans la vue historique incluent le nombre de défauts par unité et le taux de défaut.

La vue historique possède trois granularités : toutes les heures, tous les jours et tous les mois. Dans le graphique dont la granularité de temps est Toutes les heures, chaque point représente une heure. Par exemple, une valeur d'indicateur clé de performance de 24 points représente 24 heures. Dans le graphique dont la granularité de temps est Tous les jours, chaque point représente un jour. Par exemple, une valeur d'indicateur clé de performance de 30 points représente 30 jours. Dans le graphique dont la granularité de temps est Tous les mois, chaque point représente un mois. Par exemple, une valeur de l'indicateur clé de performance de 12 points représente 12 mois. Il existe des relations entre l'intervalle de temps sélectionné et la granularité. Si le nombre de points est trop important dans une granularité pour un intervalle sélectionné, cette granularité est désactivée jusqu'à ce que vous réduisiez l'intervalle. Si vous souhaitez actualiser le graphique, vous pouvez modifier l'intervalle ou cliquer sur **Actualiser**. Les données des indicateurs clés de performance historiques sont calculées périodiquement par le serveur, il y a donc un certain délai en fonction de la configuration. La période par défaut de calcul des données des indicateurs clé de performance historiques est de 1 heure.

Par défaut, seuls les cinq premiers types de défaut sont affichés dans le graphique représentant le nombre de défauts par unité. Si vous souhaitez vérifier d'autres types de défaut, vous pouvez sélectionner un ou plusieurs types de défaut sous le graphique, puis cliquer sur **Afficher la tendance**. Un nouveau graphique d'indicateurs clés de performance affiche les types de défaut sélectionnés. Si la valeur d'indicateur clé de performance d'un type de défaut est 0, elle ne peut pas être sélectionnée pour afficher une tendance.

Chapitre 6. Intégration à Prescriptive Quality

Vous pouvez exporter des données de taux de défaut historiques depuis Maximo PQI SaaS Visual Insights au format CSV et importer les données dans Prescriptive Quality.

Vous pouvez exporter les données depuis le tableau de bord des indicateurs clés de performance lorsque vous affichez les taux de défaut historiques. Pour afficher ces données, sélectionnez **Historique** dans le tableau de bord des indicateurs clés de performance, puis sélectionnez **Taux de défaut**. Lorsque vous cliquez sur **Exporter**, les données sont exportées en fonction de la cellule, de l'intervalle et de la granularité temporelle sélectionnés.

Les colonnes ci-dessous figurent dans le fichier CSV.

Tableau 3. Colonnes du fichier CSV

Colonne	Description
ATTRIBUTE_NAME	Cette valeur est toujours VI_DEFECT_IMAGE_RATE.
DATE	Date au format yyyy-MM-jj lorsque l'intervalle est daily (tous les jours) ou monthly (tous les mois) ou au format yyyy-MM-jj HH:mm:ss lorsque l'intervalle est hourly (toutes les heures).
CELL_NAME	Nom de la cellule dans laquelle survient le défaut. Il s'agit de l'une des dimensions dans Prescriptive Quality.
PRODUCED_QTY	Nombre total d'images dans la cellule et l'intervalle sélectionnés.
TESTED_QTY	Nombre total d'images qui ont été testées.
FAILED_QTY	Nombre d'images qui présentent des défauts.

Chapitre 7. Interface de programmation

L'interface de programmation permet d'effectuer des actions telles que gérer des groupes de données, des fichiers de données, des modèles, des résultats d'inspection et des serveurs de périphérie.

Flux de travaux d'API

La plupart des appels API de Maximo PQI SaaS Visual Insights possèdent des prérequis ou des corrélations avec d'autres appels API. De plus, seuls certains rôles utilisateur ont accès à tous les appels API. Par conséquent, il est important de comprendre le flux de travaux d'API pour chaque rôle.

Gestionnaire de modèles

En général, le gestionnaire de modèles utilise les appels API dans l'ordre suivant :

- Création d'un groupe de données. Voir [Création d'un groupe de données](#) dans la rubrique [Services de groupe de données](#).
- Transfert d'un fichier de données dans un groupe de données. Voir [Transfert de fichiers de données dans un groupe de données](#) dans la rubrique [Services de fichier de données](#).
- Création d'un modèle. Voir [Création d'un modèle](#) dans la rubrique [Services de modèle](#).
- Création d'une instance de modèle. Voir [Création d'une instance de modèle](#) dans la rubrique [Services d'instance de modèle](#).
- Entraînement du modèle. Voir [Entraînement d'une instance de modèle](#). Le processus d'entraînement peut durer de quelques minutes à plusieurs heures selon le nombre d'images et les paramètres d'entraînement. Pour consulter le statut de l'instance de modèle, voir [Obtention d'une instance de modèle particulière](#).
- Une fois l'entraînement en ligne terminé, le gestionnaire de modèles peut valider le modèle. Voir [Validation de l'instance de modèle](#).
- Création d'un serveur de périphérie. Voir [Création d'un serveur de périphérie](#) dans la rubrique [Services de serveur de périphérie](#).
- Déploiement du modèle sur le serveur de périphérie ou rejet du modèle. Voir [Déploiement d'une instance de modèle](#) ou [Rejet d'une instance de modèle](#) dans [Services d'action d'instance de modèle](#).
- Nouvel entraînement du modèle déployé ou annulation du déploiement du modèle. Voir [Nouvel entraînement de l'instance de modèle](#) ou [Annulation du déploiement de l'instance de modèle](#) dans la rubrique [Services d'action d'instance de modèle](#).

Inspecteur

En général, l'inspecteur utilise les appels API dans l'ordre suivant :

- Evaluation d'une image. Voir [Evaluation d'une image](#) dans la rubrique [Service d'évaluation](#).
- Obtention de la vue générale d'une cellule de résultat d'inspection. Voir [Obtention de la vue générale d'une cellule de résultat d'inspection](#) dans la rubrique [Services de résultat d'inspection](#).
- Obtention de la liste de résultats d'inspection. Voir [Obtention de la liste de résultats d'inspection](#) dans la rubrique [Services de résultat d'inspection](#).
- Confirmation des résultats d'inspection. Voir [Confirmation des résultats d'inspection](#) dans la rubrique [Services de résultat d'inspection](#).

Superviseur

En général, le superviseur utilise les appels API dans l'ordre suivant :

- Evaluation d'une image. Voir [Evaluation d'une image](#) dans la rubrique [Service d'évaluation](#).

- Obtention de la vue générale d'une cellule de résultat d'inspection. Voir Obtention de la vue générale d'une cellule de résultat d'inspection dans la rubrique Services de résultat d'inspection.
- Obtention de la liste de résultats d'inspection. Voir Obtention de la liste de résultats d'inspection dans la rubrique Services de résultat d'inspection.
- Confirmation des résultats d'inspection. Voir Confirmation des résultats d'inspection dans la rubrique Services de résultat d'inspection.
- Obtention d'un fichier de taux d'images de défaut. Voir Obtention d'un fichier de taux d'images de défaut dans la rubrique Service d'intégration QEWS.

Administrateur du serveur de périphérie autonome

L'administrateur du serveur de périphérie autonome doit disposer de données d'identification pour la machine de périphérie autonome. Généralement, il utilise les appels API dans l'ordre suivant :

- Obtenir les modèles disponibles. Voir Obtenir les modèles disponibles dans Services de serveur de périphérie autonome.
- Déployer un modèle. Voir Déployer un modèle dans Services de serveur de périphérie autonome.
- Déployer tous les modèles. Voir Déployer tous les modèles dans Services de serveur de périphérie autonome.
- Transférer et évaluer une image sur le serveur de périphérie. Voir Transférer et évaluer une image sur le serveur de périphérie dans Services de serveur de périphérie autonome. La ligne de production ou des services externes utilisent généralement cette API.
- Synchroniser le résultat de l'inspection du serveur de périphérie dans l'application centrale. Voir Synchroniser le résultat de l'inspection du serveur de périphérie dans l'application centrale dans Services de serveur de périphérie autonome.
- Nettoyer un résultat d'inspection que vous avez synchronisé avec l'application centrale. Voir Nettoyer un résultat d'inspection que vous avez synchronisé avec l'application centrale dans Services de serveur de périphérie autonome.
- Annuler le déploiement d'un modèle. Voir Annuler le déploiement d'un modèle dans Services de serveur de périphérie autonome.

Préparation pour l'utilisation d'appels API

Avant d'utiliser un appel API, assurez-vous d'avoir préparé l'utilisateur et la solution et d'avoir obtenu une clé d'API.

Procédez comme suit pour préparer l'utilisateur et la solution :

- Assurez-vous que l'utilisateur qui est connecté a accès aux appels API souhaités.
- Pour les appels API requérant un ID de solution, spécifiez `vi` comme solution.
- Assurez-vous que l'utilisateur dispose de la clé d'API qu'IBM a créé pour lui.

Réponses de service

Une fois qu'un appel API est terminé, il fournit une réponse indiquant la réussite ou une erreur.

Les services d'API fournissent les réponses suivantes :

- Réponses indiquant la réussite :
 - Code de statut 201 pour les services de création
 - Code de statut 200 pour les autres services
- Réponses indiquant une erreur de base de données :
 - Code de statut 500

- Messages d'erreur, par exemple "Les paramètres manquent ou leur format n'est pas valide pour la création d'un groupe de données."
- Erreurs d'autorisation
 - Code de statut 401
 - Messages d'erreur, par exemple "APIKEY ou Authorization doit être défini dans l'en-tête."

Services de groupe de données

Les services de groupe de données permettent d'effectuer des tâches telles qu'obtenir tous les groupes de données, obtenir un groupe de données particulier et créer un groupe de données.

Obtenir tous les groupes de données

Permet d'obtenir tous les groupes de données existants. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataGroup

Méthode

Type de demande GET

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "id": "892d05cd-84e5-4bae-b57f-edbbc3f8c598",
    "groupName": "Mist",
    "createdBy": "auto_mm",
    "description": "test_description",
    "dataFormat": "jpg",
    "parameters": {
      "isHybrid": "false",
      "isDefect": "true"
    }
  },
  {
    "id": "9bebc890-0237-4a3b-8908-9f5f53fdb3a2",
    "groupName": "NoDefect",
    "createdBy": "auto_mm",
    "description": "test_description",
    "dataFormat": "jpg",
    "parameters": {
      "isHybrid": "false",
      "isDefect": "true"
    }
  }
]
```

```

{
  "id": "b989aae2-f1ae-4a77-93bb-4e9e4f18264d",
  "groupName": "Defect",
  "createdBy": "auto_mm",
  "description": "test_description",
  "dataFormat": "jpg",
  "parameters": {
    "isHybrid": "false",
    "isDefect": "true"
  },
  "updatedAt": "2017-11-29T16:47:53.536+08:00"
}
]

```

Éléments de réponse

id: *chaîne*. ID du groupe de données.

groupName: *chaîne*. Nom du groupe de données.

createdBy: *chaîne*. Propriétaire du groupe de données.

description: *chaîne*. Description du groupe de données.

dataFormat: *chaîne*. Format du groupe de données.

isHybrid: *valeur booléenne*. True indique la détection d'objet. False indique la classification.

isDefect: *valeur booléenne*. True signifie qu'il s'agit d'un défaut. False signifie qu'il ne s'agit pas d'un défaut.

updatedAt: *heure*. Heure de la dernière mise à jour du groupe de données.

Exemple d'appel

```

curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aa
b2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.91:9447/ibm/iotm/service/dataGroup?user=auto_mm&solution=vi"

```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtention de tous les groupes de données avec des fichiers de données

Permet d'obtenir tous les groupes de données existants comportant des fichiers de données. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataGroup

Méthode

Type de demande GET

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

dataFiles: *[]*. Utilisé pour identifier et obtenir des fichiers de données. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "id": "3e420dbc-88fb-463a-83ee-a317688b02e0",
    "groupName": "test",
    "createdBy": "auto_mm",
    "description": "test_description",
    "dataFormat": "jpg",
    "parameters": {
      "isHybrid": "false",
      "isDefect": "true"
    },
    "dataFiles": [
      {
        "id": "3e420dbc-88fb-463a-83ee-a317688b02e0_539bf24a-afc9-4f43-9ca9-b7ec0648e781",
        "groupId": "3e420dbc-88fb-463a-83ee-a317688b02e0",
        "createdBy": "auto_mm",
        "fileName": "MisT.zip",
        "fileCount": "3",
        "updatedAt": "1511949371332",
        "fileUrl": "/user/AUTO/VI/datagroup/3e420dbc-88fb-463a-83ee-a317688b02e0/1511949371332.zip"
      },
      {
        "id": "3e420dbc-88fb-463a-83ee-a317688b02e0_f76b31fb-f803-4d0e-995e-b597bf3c761d",
        "groupId": "3e420dbc-88fb-463a-83ee-a317688b02e0",
        "createdBy": "auto_mm",
        "fileName": "Defect.zip",
        "fileCount": "3",
        "updatedAt": "1511949368998",
        "fileUrl": "/user/AUTO/VI/datagroup/3e420dbc-88fb-463a-83ee-a317688b02e0/1511949368998.zip"
      }
    ],
    "updatedAt": "2017-11-29T16:59:43.873+08:00"
  }
]
```

Éléments de réponse

id: *chaîne*. ID du groupe de données.

groupName: *chaîne*. Nom du groupe de données.

createdBy: *chaîne*. Propriétaire du groupe de données.

description: *chaîne*. Description du groupe de données.

dataFormat: *chaîne*. Format du groupe de données.

isHybrid: *valeur booléenne*. True indique la détection d'objet. False indique la classification.

isDefect: *valeur booléenne*. True signifie qu'il s'agit d'un défaut. False signifie qu'il ne s'agit pas d'un défaut.

dataFiles: *objet JSON*. Contient des informations sur les fichiers de données.

id: *chaîne*. ID du fichier de données.

groupId: *chaîne*. ID du groupe de données.

createdBy: *chaîne*. Propriétaire du fichier de données.

fileName: *chaîne*. Nom du fichier.

fileCount: *chaîne*. Nombre de fichiers.

updatedAt: <heure>. Heure de la dernière mise à jour du fichier de données.

fileUrl: *chaîne*. Chemin d'accès au fichier qui est stocké sur le serveur.

updatedAt: *heure*. Heure de la dernière mise à jour du groupe de données.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d
4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112.229.91:
9447/ibm/iotm/service/dataGroup?user=auto_mm&solution=vi &dataFiles=[]"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir un groupe de données particulier

Permet d'obtenir un groupe de données particulier. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataGroup/*ID_groupe*

Méthode

Type de demande GET

Paramètres d'URL

groupId: *chaîne*. ID du groupe de données.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "tags": "",
    "id": "3e420dbc-88fb-463a-83ee-a317688b02e0",
    "groupName": "test",
    "createdBy": "auto_mm",
    "description": "test_description",
    "dataFormat": "jpg",
    "parameters": {
      "isHybrid": "false",
      "isDefect": "true"
    },
    "updatedAt": "2017-11-29T16:59:43.873+08:00"
  }
]
```

Éléments de réponse

tags: *chaîne*. Balise du groupe de données.

id: *chaîne*. ID du groupe de données.

groupName: *chaîne*. Nom du groupe de données.

createdBy: *chaîne*. Propriétaire du groupe de données.
description: *chaîne*. Description du groupe de données.
dataFormat: *chaîne*. Format du groupe de données.
isHybrid: *valeur booléenne*. True indique la détection d'objet. False indique la classification.
isDefect: *valeur booléenne*. True signifie qu'il s'agit d'un défaut. False signifie qu'il ne s'agit pas d'un défaut.
updatedAt: *heure*. Heure de la dernière mise à jour du groupe de données.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77
d4f0579e533dd34gg959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.91:9447/ibm/iotm/service/dataGroup/3e420dbc-88fb-463a-8
3ee-a317688b02e0?user=auto_mm&solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Créer un groupe de données

Permet de créer un groupe de données. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataGroup

Méthode

Type de demande POST

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.
solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.
tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.
APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

groupName: *chaîne*. Nom du groupe de données.
description: *chaîne*. Description du groupe de données.
dataFormat: *chaîne*. Format du groupe de données.
isHybrid: *valeur booléenne*. True indique la détection d'objet. False indique la classification.
isDefect: *valeur booléenne*. True signifie qu'il s'agit d'un défaut. False signifie qu'il ne s'agit pas d'un défaut.

Exemple de corps

```
[
  {
    "groupName": "test",
    "description": "test_description",
    "parameters": {
      "isHybrid": "false",
      "isDefect": "true"
    }
  }
]
```

```
}  
]
```

Réponse indiquant la réussite

```
201  
[  
  {  
    "id": "7dac9493-4d8c-4472-a4ca-ae450ccaea5d",  
    "groupName": "test",  
    "tags": null,  
    "dataFormat": "jpg",  
    "createdBy": "auto_mm",  
    "updatedAt": "2017-11-30T09:59:09.597+08:00",  
    "parameters": {  
      "isHybrid": "false",  
      "isDefect": "true"  
    },  
    "description": "test_description"  
  }  
]
```

Éléments de réponse

id: *chaîne*. ID du groupe de données.

groupName: *chaîne*. Nom du groupe de données.

tags: *chaîne*. Balise du groupe de données.

createdBy: *chaîne*. Propriétaire du groupe de données.

description: *chaîne*. Description du groupe de données.

dataFormat: *chaîne*. Format du groupe de données.

isHybrid: *valeur booléenne*. True indique la détection d'objet. False indique la classification.

isDefect: *valeur booléenne*. True signifie qu'il s'agit d'un défaut. False signifie qu'il ne s'agit pas d'un défaut.

updatedAt: *heure*. Heure de la dernière mise à jour du groupe de données.

Exemple d'appel

```
curl -k -H  
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f05  
79e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H "Content-Type:  
application/json;charset=UTF-8" "https://9.112.229.91:9447/ibm/iotm/service/  
dataGroup?user=auto_mm&solution=vi" --data '[{"groupName":"test","dataFormat":  
"jpg", "description": "test_description", "parameters":{"isHybrid":"false", "isDefect"  
:"true"}}]'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Mettre à jour un groupe de données

Permet de mettre à jour un groupe de données. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataGroup/*ID_groupe*

Méthode

Type de demande PUT

Paramètres d'URL

groupId: *chaîne*. ID du groupe de données. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

description: *chaîne*. Description du groupe de données.

isHybrid: *valeur booléenne*. True indique la détection d'objet. False indique la classification.

isDefect: *valeur booléenne*. True signifie qu'il s'agit d'un défaut. False signifie qu'il ne s'agit pas d'un défaut.

Exemple de corps

```
200
[
  {
    "description": "test",
    "parameters": {
      "isDefect": "true",
      "isHybrid": "true" }
  }
]
```

Réponse indiquant la réussite

```
[{}]
```

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533d
d34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H "Content-Type:
application/json;charset=UTF-8" -X PUT "https://9.112.229.91:9447/ibm/iotm/service
/dataGroup/7dac9493-4d8c-4472-a4ca-ae450cca5d?user=auto_mm&solution=vi" --data
' [{"description": "test", "parameters": {"isHybrid": "true", "isDefect": "true"}} ]'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Supprimer des groupes de données

Permet de supprimer des groupes de données existants. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataGroup/*ID_groupe*

Méthode

Type de demande DELETE

Paramètres d'URL

groupId: *chaîne*. ID du groupe de données. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

```
[{}]
```

Réponse indiquant la réussite

```
200  
[{}]
```

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -X DELETE -H "APIKEY:8b796582  
25de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f05  
79e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"  
"https://9.112.229.91:9447/ibm/iotm/service/dataGroup/  
ed568556-1462-45e9-9319-4bfa8186d2cd?user=auto_mm&solution=  
vi" -d '[{}]
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services de fichier de données

Les services de fichier de données permettent d'effectuer des tâches telles qu'obtenir tous les fichiers de données appartenant à un groupe de données, obtenir un fichier de données particulier et télécharger un fichier binaire.

Obtenir tous les fichiers de données appartenant à un groupe de données

Permet d'obtenir tous les fichiers de données qui appartiennent à un groupe de données. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataFile

Méthode

Type de demande GET

Paramètres d'URL

groupId: *chaîne*. ID du groupe de données. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "id": "06c870e5-1701-419a-8a84-0f40ee82fe33_a7ff1772f-07a0-4d33-9efe-f2e45a2dc904",
    "groupId": "06c870e5-1701-419a-8a84-0f40ee82fe33",
    "createdBy": "modelmanager1",
    "fileName": "NG.zip",
    "updatedAt": "1502185276752",
    "fileCount": "52",
    "fileUrl": "/user/T1/VIQ/datagroup/06c870e5-1701-419a-8a84-0f40ee82fe33/1502185276752.zip"
  },
  {
    "id": "06c870e5-1701-419a-8a84-0f40ee82fe33_b3af214c-b603-43f4-944d-40277b9a6fd9",
    "groupId": "06c870e5-1701-419a-8a84-0f40ee82fe33",
    "createdBy": "modelmanager1",
    "fileName": "IoT4M_March2017_VI_Source.zip",
    "updatedAt": "1502184389077",
    "fileCount": "52",
    "fileUrl": "/user/T1/VIQ/datagroup/06c870e5-1701-419a-8a84-0f40ee82fe33/1502184389077.zip"
  },
  {
    "id": "06c870e5-1701-419a-8a84-0f40ee82fe33_c901e409-d9db-4f02-8d9c-fa6002929701",
    "groupId": "06c870e5-1701-419a-8a84-0f40ee82fe33",
    "createdBy": "modelmanager1",
    "fileName": "NG.zip",
    "updatedAt": "1502185248433",
    "fileCount": "52",
    "fileUrl": "/user/T1/VIQ/datagroup/06c870e5-1701-419a-8a84-0f40ee82fe33/1502185248433.zip"
  }
]
```

Éléments de réponse

id: *chaîne*. ID du fichier de données.

groupId: *chaîne*. ID du groupe de données.

createdBy: *chaîne*. Propriétaire du fichier de données.

fileName: *chaîne*. Nom du fichier.

fileCount: *chaîne*. Nombre de fichiers.

updatedAt: *heure*. Heure de la dernière mise à jour du fichier de données.

fileUrl: *chaîne*. Chemin d'accès au fichier qui est stocké sur le serveur.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34gggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.91:9447/ibm/iotm/service/dataFile?user=auto_mm&solution=vi&groupId=ed568556-1462-45e9-9319-4bfa8186d2cd"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir un fichier de données particulier

Permet d'obtenir un fichier de données particulier. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataFile

Méthode

Type de demande GET

Paramètres d'URL

fileId: *chaîne*. ID du fichier de données. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "id": "06c870e5-1701-419a-8a84-0f40ee82fe33_c901e409-d9db-4f02-8d9c-fa6002929701",
    "groupId": "c901e409-d9db-4f02-8d9c-fa6002929701",
    "createdBy": "modelmanager1",
    "fileName": "NG.zip",
    "updatedAt": "1502185248433",
    "fileCount": "52",
    "fileUrl": "/user/T1/VIQ/datagroup/06c870e5-1701-419a-8a84-0f40ee82fe33/1502185248433.zip"
  }
]
```

Éléments de réponse

id: *chaîne*. ID du fichier de données.

groupId: *chaîne*. ID du groupe de données.

createdBy: *chaîne*. Propriétaire du fichier de données.

fileName: *chaîne*. Nom du fichier.

updatedAt: *heure*. Heure de la dernière mise à jour du fichier de données.

fileUrl: Chemin d'accès au fichier qui est stocké sur le serveur.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fd485cc6f275f5" "https://9.112
```

```
.229.91:9447/ibm/iotm/service/dataFile/06c870e5-1701-419a-8a84-0f40ee82fe33_c901e409-d9db-4f02-8d9c-fa6002929701?user=auto_mm&solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Télécharger le contenu binaire du fichier de données

Permet de télécharger le contenu binaire d'un fichier de données. Le gestionnaire de modèles et le spécialiste des données disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataFileBinary

Méthode

Type de demande GET

Paramètres d'URL

fileId: *chaîne*. ID du fichier de données. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

Contenu binaire du fichier de données.

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f05
79e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.
112.229.91:9447/ibm/iotm/service/ dataFileBinary ?fileId=06c870e5-1701-419a
-8a84-0f40ee82fe33_c901e409-d9db-4f02-8d9c-fa6002929701&user=auto_mm&solution
=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Transférer des fichiers de données dans un groupe de données

Permet de transférer un ou plusieurs fichiers de données dans un groupe de données. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataFileBinary

Méthode

Type de demande POST

Paramètres d'URL

groupId: *chaîne*. ID du groupe de données. Obligatoire.
user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.
solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.
tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: multipart/form-data.
Content-Disposition: form-data; name="files[]"; filename=*votre nom de fichier*.
APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Contenu binaire du fichier de données.

Réponse indiquant la réussite

```
200
{"result":{"Defect.zip":{"id":"9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b","count":"3","name":"Defect.zip","errorMsg":"","updatedAtTime":"1512021663498","url":"\\/user\\AUTO\\VI\\datagroup\\9f7aa9d6-24d6-4611-8d4d-79a52d2bab02\\1512021663498.zip"},"error_message":{"}}
```

Éléments de réponse

id: *chaîne*. ID du fichier de données.
name: *chaîne*. Nom du fichier.
count: *chaîne*. Nombre d'images.
updatedAtTime: *heure*. Heure de la dernière mise à jour du fichier de données.
fileUrl: . Chemin d'accès au fichier qui est stocké sur le serveur.
error_message: . Message d'erreur d'une réponse de transfert.

Exemple d'appel

```
curl -k -X POST -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
-H "Content-Type:multipart/form-data" -H "Content-Disposition:
form-data; name=\"files[]\"; filename=\"Defect.zip\" \"https://9.112.229.91:9447
/ibm/iotm/service/dataFileBinary?user=auto_mm&solution=vi&groupId=9f7aa9d6-24d6-4611-8d4d-79a52d2bab02\" --connect-timeout 600 -F file=@C:\\code\\Automation_98\\
API\\vi\\VI_API\\file\\Defect.zip
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Supprimer un fichier de données particulier

Permet de supprimer un fichier de données particulier. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/dataFile/*ID_fichier*

Méthode

Type de demande DELETE

Paramètres d'URL

fileId: *chaîne*. ID du fichier de données. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

```
[{}]
```

Réponse indiquant la réussite

```
200  
[{}]
```

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -X DELETE -H "APIKEY:8b79658  
225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f057  
9e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"  
"https://9.112.229.91:9447/ibm/iotm/service/dataFile/  
9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_e2cc3d4a-bae9-  
4600-9dff-d80fa3ee8832?user=auto_mm&solution=vi" -d '[{}]
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services de groupe de données sans libellé

Les services de groupe de données sans libellé permettent d'effectuer des tâches telles que le transfert d'un fichier image compressé sans libellé, la création d'un groupe de données sans libellé, et l'obtention d'un groupe de données sans libellé.

Transférer un fichier image compressé sans libellé

Permet de transférer un fichier compressé sans libellé sur le serveur. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/unLabeledImageZipServlet

Méthode

Type de demande POST

Paramètres d'URL

groupType: *chaîne*. Type de groupe de données. La valeur doit être *classification* ou *objectdetection*. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: multipart/form-data.

Content-Disposition: form-data; name="files[]"; filename=*votre nom de fichier*.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucune

Exemple de corps

Contenu binaire du fichier de données.

Réponse indiquant la réussite

```
200 {"message": {}, "result": {"imageZip": {"imageZipId": "b76b33b0-26d6-4f85-83ea-e520edbe5ff8", "labeledImageCount": 0, "labelCount": 0, "imageZipName": "images.zip", "count": 85, "updatedAt": "2019-03-19 00:18:53.304", "url": "\user\TENANT\VI\unlabeledgroup\b76b33b0-26d6-4f85-83ea-e520edbe5ff8.zip"}}
```

Éléments de réponse

imageZip: *objet JSON*. Informations relatives au fichier image compressé transféré. Elles incluent les paramètres imageZipId, LabeledImageCount, labelCount, imageZipName, count, updatedAt et URL.

imageZipId: *chaîne*. ID affecté au fichier image compressé téléchargé.

LabeledImageCount: *entier*. Nombre d'images qui ont été préalablement libellées dans le fichier image compressé.

labelCount: *entier*. Nombre de libellés affectés aux images dans le fichier image compressé.

imageZipName: *chaîne*. Nom du fichier image compressé.

count: *entier*. Nombre total d'images dans le fichier image compressé.

updatedAt: *heure*. Heure du transfert du fichier image compressé.

Url: *chaîne*. Chemin d'accès au fichier qui est stocké sur le serveur.

error_message: *objet JSON*. Message d'erreur pour une réponse de transfert.

Exemple d'appel

```
curl -k -X POST -H"APIKEY:apikey"  
-H "Content-Type:multipart/form-data" -H "Content-Disposition: form-data;  
name=\"files[]\"; filename=\"image.zip\"" "https://iotm.predictivesolutionsapps  
.ibmcloud.com/ibm/iotm/service/unLabeledImageZipServlet?user=auto_mm&solution=vi&  
groupType=classification" --connect-timeout 600  
-F file=@C:\\code\\Automation_98\\API\\vi\\VI_API\\file\\image.zip
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Créer un groupe de données sans libellé

Crée un groupe de données sans libellé avec le fichier image compressé transféré. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/unLabeledGroup

Méthode

Type de demande POST

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: multipart/form-data.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

groupName: *chaîne*. Nom du groupe de données.

groupType: *chaîne*. Type de groupe de données. La valeur doit être *classification* ou *objectdetection*.

imageZip: *tableau JSON*. Informations relatives au fichier image compressé transféré. Vous pouvez obtenir les informations depuis la réponse lorsque vous transférez un fichier image compressé sans libellé. Elles incluent les paramètres imageZipId, LabeledImageCount, labelCount, imageZipName, count, updatedAt et URL.

imageZipId: *chaîne*. ID affecté au fichier image compressé téléchargé.

LabeledImageCount: *entier*. Nombre d'images qui ont été préalablement libellées dans le fichier image compressé.

labelCount: *entier*. Nombre de libellés affectés aux images dans le fichier image compressé.

imageZipName: *chaîne*. Nom du fichier image compressé.

count: *entier*. Nombre total d'images dans le fichier image compressé.

updatedAt: *heure*. Heure du transfert du fichier image compressé.

Url: *chaîne*. Chemin d'accès au fichier qui est stocké sur le serveur.

Exemple de corps

```
[{"groupName":"DI","groupType":"classification","imageZip":[{"imageZipId":"b76b33b0-26d6-4f85-83ea-e520edbe5ff8","labeledImageCount":0,"labelCount":0,"imageZipName":"image.zip","count":85,"updatedAt":"2019-03-19 00:18:53.304","url":"/user/TENANT/VI/unlabeledgroup/b76b33b0-26d6-4f85-83ea-e520edbe5ff8.zip"}]}
```

Réponse indiquant la réussite

```
200  
[{"groupName":"DI","id":"04f4eaf3-dc93-4ad8-b2e2-d3fbd202c758","groupType":"classification","imageZip":[{"imageZipId":"b76b33b0-26d6-4f85-83ea-e520edbe5ff8","labeledImageCount":0,"labelCount":0,"imageZipName":"image.zip","count":85,"updatedAt":"2019-03-19 00:18:53.304","url":"/user/TENANT/VI/unlabeledgroup/b76b33b0-26d6-4f85-83ea-e520edbe5ff8.zip"}]}
```

Éléments de réponse

id: *chaîne*. ID du groupe de données.

groupName: *chaîne*. Nom du groupe de données.

groupType: *chaîne*. Type de groupe de données.

createdBy: *chaîne*. Propriétaire du groupe de données.

groupStatus: *chaîne*. Statut du groupe de données sans libellé.

updatedAt: *heure*. Heure de la dernière mise à jour du groupe de données.

imageZip: *tableau JSON*. Informations relatives aux fichiers image compressés associés. Chaque élément inclut les paramètres imageZipId, LabeledImageCount, labelCount, imageZipName, count, updatedAt et URL.

imageZipId: *chaîne*. ID affecté au fichier image compressé téléchargé.

LabeledImageCount: *entier*. Nombre d'images qui ont été préalablement libellées dans le fichier image compressé.

labelCount: *entier*. Nombre de libellés affectés aux images dans le fichier image compressé.

imageZipName: *chaîne*. Nom du fichier image compressé.

count: *entier*. Nombre total d'images dans le fichier image compressé.

updatedAt: *heure*. Heure du transfert du fichier image compressé.

url: *chaîne*. Chemin d'accès au fichier qui est stocké sur le serveur.

Exemple d'appel

```
curl -k -X POST -H "APIKEY:apikey" -H  
"Content-Type:application/json;charset=UTF-8"  
"https://iotm.predictivesolutionsapps.ibmcloud.com/  
ibm/iotm/service/unLabeledGroup?user=auto_mm&solution=vi"  
--data [{"groupName":"DI","groupType":"classification",  
"imageZip":[{"imageZipId":"b76b33b0-26d6-4f85-83ea-e520edbe5ff8",  
"labeledImageCount":0,"labelCount":0,"imageZipName":"image.zip",  
"count":85,"updatedAt":"2019-03-19 00:18:53.304","url":  
"/user/TENANT/VI/unlabeledgroup/b76b33b0-26d6-4f85-83ea-  
e520edbe5ff8.zip"}]}
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir un groupe de données sans libellé

Permet d'obtenir un groupe de données sans libellé. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/unLabeledGroup/*ID_groupe*

Méthode

Type de demande GET

Paramètres d'URL

groupId: *chaîne*. ID du groupe de données.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucune

Exemple de corps

Aucune

Réponse indiquant la réussite

```
200
[{"groupName": "DI", "id": "04f4eaf3-dc93-4ad8-b2e2-d3fbd202c758",
"groupType": "classification", "imageZip": [{"imageZipId": "b76b33b0-
26d6-4f85-83ea-e520edbe5ff8", "labeledImageCount": 0, "labelCount": 0,
"imageZipName": "image.zip", "count": 85, "updatedAt": "2019-03-19
00:18:53.304", "url": "/user/TENANT/VI/unlabeledgroup/b76b33b0-26d6-
4f85-83ea-e520edbe5ff8.zip"}]}
```

Éléments de réponse

id: *chaîne*. ID du groupe de données.

groupName: *chaîne*. Nom du groupe de données.

groupType: *chaîne*. Type de groupe de données.

createdBy: *chaîne*. Propriétaire du groupe de données.

groupStatus: *chaîne*. Statut du groupe de données sans libellé.

updatedAt: *heure*. Heure de la dernière mise à jour du groupe de données.

imageZip: *tableau JSON*. Informations relatives aux fichiers image compressés associés. Chaque élément inclut les paramètres imageZipId, LabeledImageCount, labelCount, imageZipName, count, updatedAt et URL.

imageZipId: *chaîne*. ID affecté au fichier image compressé téléchargé.

LabeledImageCount: *entier*. Nombre d'images qui ont été préalablement libellées dans le fichier image compressé.

labelCount: *entier*. Nombre de libellés affectés aux images dans le fichier image compressé.

imageZipName: *chaîne*. Nom du fichier image compressé.

count: *entier*. Nombre total d'images dans le fichier image compressé.

updatedAt: *heure*. Heure du transfert du fichier image compressé.

Url: *chaîne*. Chemin d'accès au fichier qui est stocké sur le serveur.

Exemple d'appel

```
curl -k -H "APIKEY:apikey"  
"https://iotm.predictivesolutionsapps.ibmcloud.com/ibm/iotm/service  
/unLabeledGroup/04f4eaf3-dc93-4ad8-b2e2-d3fbd202c758?user=auto_mm&  
solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services de modèle

Les services de modèle permettent d'effectuer des tâches telles qu'obtenir tous les modèles, obtenir un modèle particulier et créer un modèle.

Obtenir tous les modèles

Permet d'obtenir tous les modèles existants. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/model

Méthode

Type de demande GET

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200  
[  
  {  
    "modelId": "a9150695-5dc5-4bef-937d-c71fc186bc14",  
    "groupIds": "e83f9faf-2b08-44d1-81b0-7c69e6407329",  
    "modelName": "appscan test01",  
    "createdBy": "demoadm@cn.ibm.com",  
    "modelType": "classification",  
    "statusStatistics": {},  
    "dataFormat": "png",  
    "retrainPolicy": {  
      "scheduler": "Weekly, Sunday",  
      "ratio": "",  
      "reuseDayNumber": "",  
      "maxPieceNumber": "",  
      "archiveDataType": "0",  
      "imageNumber": 1000,  
      "accuracy": "70"  
    },  
  },  
]
```

```

    "parameters": {
      "imageSize": "100*100",
      "confidence": 100,
      "trainParam": {
        "epoch": 50,
        "ratioTrain": 80,
        "stepsize": 30,
        "test_iter": 20,
        "snapshot": 100,
        "momentum": 0.9,
        "ratioVal": 20,
        "learningRate": 0.0001,
        "display": 1,
        "learningRatePolicy": "step",
        "weight_decay": 0.00001,
        "test_epoc": 10,
        "network": "GoogLeNet",
        "algorithm": "CNN",
        "test_interval": 5,
        "gamma": 0.1,
        "solver_type": "SGD",
        "maxIter": 600
      }
    },
    "updatedAt": "2018-11-16T00:16:20.032-06:00",
    "productType": "appscan test01"
  }
}
]

```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

modelName: *chaîne*. Nom du modèle.

createdBy: *chaîne*. Propriétaire du modèle.

updatedAt: *heure*. Heure de la dernière mise à jour du modèle.

productType: *chaîne*. Type de produit du modèle.

description: *chaîne*. Description du modèle.

StatusStatistics: *chaîne*. Statut du modèle.

groupIds: *chaîne*. ID du groupe de données du modèle.

dataFormat: *chaîne*. Format des données de modèle.

retrainPolicy: *objet JSON*. Règle de nouvel entraînement du modèle.

scheduler: *date*. Date de conservation du modèle, de lundi à dimanche (Monday à Sunday).

imageNumber: *entier*. Un nombre d'images supérieur à ce nombre peut être entraîné à nouveau.

accuracy: *double*. Une valeur de précision du modèle inférieure à ce nombre peut être entraînée à nouveau.

parameters: *objet JSON*. Contient des informations sur la taille de l'image et la fiabilité du modèle ainsi que la zone trainParam.

imageSize: *double * double*. Taille de l'image.

confidence: *double*. Valeur de fiabilité définie pour le modèle.

trainParam: *objet JSON*. Contient les informations de paramètre d'entraînement.

Exemple d'appel

```

curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e
533dd34gggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112
.229.91:9447/ibm/iotm/service/model?user=auto_mm&solution=vi"

```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtention d'un modèle particulier

Permet d'obtenir un modèle existant particulier. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/model/<ID_modèle>

Méthode

Type de demande GET

Paramètres d'URL

modelId: *chaîne*. ID du modèle. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "modelId": "f39df888-7559-4283-ae9d-b3504118dddd",
    "groupIds": "9e744bc4-0a22-4bf3-9e63-fa39135842de",
    "modelName": "arthuryolo22000v2",
    "createdBy": "demoadm@cn.ibm.com",
    "modelType": "objectdetection",
    "description": "arthuryolo22000v2",
    "statusStatistics": {
      "trained": "1"
    },
    "dataFormat": "png",
    "retrainPolicy": {
      "scheduler": "Weekly,Sunday",
      "ratio": "",
      "reuseDayNumber": "",
      "maxPieceNumber": "",
      "archiveDataType": "0",
      "imageNumber": 1000,
      "accuracy": "70"
    },
    "trainedBy": "demoadm@cn.ibm.com",
    "parameters": {
      "JobInstanceMap": {
        "4a9605a2-03e3-43ff-9a47-d7638a0145d9": "f39df888-7559-4283-ae9d-b3504118dddd_1541744213466"
      },
      "odRetrainUrl": "/user/Q3T1/VI/modellist/f39df888-7559-4283-ae9d-b3504118dddd_1541761094044.zip",
      "imageSize": "100*100",
      "confidence": 100,
      "trainParam": {
        "ratioTrain": 80,
        "subBatchSize": 2,
        "scales": "1,1",
        "ratioVal": 20,

```

```

        "learningRate": 0.0001,
        "batchSize": 16,
        "steps": "6000,10000",
        "recommnd": 0,
        "network": "YoloV2",
        "iteration": 10000,
        "algorithm": "YOLO"
    },
    "updatedTime": "2018-11-09T00:24:19.949-06:00",
    "productType": "arthuryolo22000v2"
}
]

```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

modelName: *chaîne*. Nom du modèle.

createdBy: *chaîne*. Propriétaire du modèle.

updatedTime: *heure*. Heure de la dernière mise à jour du modèle.

productType: *chaîne*. Type de produit du modèle.

description: *chaîne*. Description du modèle.

StatusStatistics: *chaîne*. Statut du modèle.

groupIds: *chaîne*. ID du groupe de données du modèle.

dataFormat: *chaîne*. Format des données de modèle.

retrainPolicy: *objet JSON*. Règle de nouvel entraînement du modèle.

scheduler: *date*. Date de conservation du modèle, de lundi à dimanche (Monday à Sunday).

imageNumber: *entier*. Un nombre d'images supérieur à ce nombre peut être entraîné à nouveau.

accuracy: *double*. Une valeur de précision du modèle inférieure à ce nombre peut être entraînée à nouveau.

parameters: *objet JSON*. Contient des informations sur la taille de l'image et la fiabilité du modèle ainsi que les zones odRetrainUrl, JobInstanceMap et trainParam.

imageSize: *double * double*. Taille de l'image.

confidence: *double*. Valeur de fiabilité définie pour le modèle.

trainParam: *objet JSON*. Inclut les informations d'algorithmes, de réseau et d'hyperparamètre.

Exemple d'appel

```

curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533
dd34gggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112.229.91:
9447/ibm/iotm/service/model/ab219b13-16c0-4c7b-ae74-721f4719e314?user=auto_mm&so
lution=vi"

```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Créer un modèle

Permet de créer un modèle. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/model

Méthode

Type de demande POST

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: *application/json*.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

modelName: *chaîne*. Nom du modèle.

modelType: *chaîne*. Type de modèle. Les options sont "classification" et "objectdetection".

createdBy: *chaîne*. Propriétaire du modèle.

description: *chaîne*. Description du modèle.

groupIds: *chaîne*. ID du groupe de données du modèle.

dataFormat: *chaîne*. Format des données de modèle.

retrainPolicy: *objet JSON*. Règle de nouvel entraînement du modèle.

scheduler: *date*. Date de conservation du modèle, de lundi à dimanche (Monday à Sunday).

imageNumber: *entier*. Un nombre d'images supérieur à ce nombre peut être entraîné à nouveau.

accuracy: *double*. Une valeur de précision du modèle inférieure à ce nombre peut être entraînée à nouveau.

parameters: *objet JSON*. Contient la taille d'image, les informations relatives à la fiabilité du modèle, ainsi que les propriétés des paramètres d'entraînement, notamment des informations sur l'algorithme, le réseau et les hyperparamètres.

Les paramètres d'entraînement, les hyperparamètres et les valeurs de type de modèle correspondent à des relations. Si la valeur de type de modèle est "classification", la valeur d'"algorithm" est "CNN" et la valeur de "network" correspondante peut être "GoogleNet", "LeNet" ou "AlexNet" (3 options). Les hyperparamètres correspondants incluent "learningRate", qui est un nombre réel compris entre 0 et 1, "maxIter" ou "epoch", qui est un entier positif, "stepsize", qui est un entier positif, "gamma", qui est un nombre réel compris entre 0 et 1, "learningRatePolicy", dont la valeur est "step", "test_iter" ou "test_epoch", qui est un entier positif, "test_interval", qui est un entier positif, "snapshot", qui est un entier supérieur ou égal à maxIter/20, "ratioTrain", qui correspond au rapport des images utilisées pour l'entraînement (entiers entre 0 et 100) et "ratioVal", qui correspond au rapport des images utilisées pour la validation (entier entre 0 et 100).

Si la valeur de type de modèle est "objectdetection", la valeur d'"algorithm" est "FRCNN", "YOLO" ou "SSD". Si la valeur d'"algorithm" est "FRCNN", la valeur de "network" correspondante peut être "ZfNet" ou "VGG16" (2 options). Les hyperparamètres correspondants incluent "learningRate", qui est un nombre réel compris entre 0 et 1, "iteration", qui est une liste d'entiers positifs séparés par une virgule (par exemple 10000, 10000, 10000, 10000), "stepsize", qui est un entier positif, "gamma", qui est un nombre réel compris entre 0 et 1, "ratioTrain", qui est un entier compris entre 0 et 100 et "ratioVal", qui est un entier compris entre 0 et 100.

Si la valeur d'"algorithm" est "YOLO", la valeur de "network" correspondante peut être "YoloV2", "TinyYolo" ou "YoloV1" (3 options). Les hyperparamètres correspondants incluent "learningRate", qui est un nombre réel compris entre 0 et 1, "iteration", qui est un entier positif, "steps", qui est une liste d'entiers positifs séparés par une virgule (par exemple 100, 1000, 5000, 8000), "batchSize", qui est un entier positif, "scales", qui est une liste de nombres réels séparés par une virgule (par exemple 0.1,

0.1, 0.1, 0.1), "subBatchSize", qui est un entier positif, "ratioTrain", qui est un entier compris entre 0 et 100 et "ratioVal", qui est un entier compris entre 0 et 100.

Si la valeur d'"algorithm" est "SSD", la valeur de "network" correspondante est "SSD300". Les hyperparamètres correspondants incluent "learningRate", qui est un nombre réel compris entre 0 et 1, "iteration", qui est un entier positif, "steps", qui est une liste d'entiers positifs séparés par une virgule (par exemple 100, 1000, 5000, 8000), "batchSize", qui est un entier positif, "learningRatePolicy", dont la valeur est "multistep", "snapshot" qui est un entier égal ou supérieur à iteration/20, "ratioTrain", qui est un entier compris entre 0 et 100 et "ratioVal", qui est un entier compris entre 0 et 100.

confidence: *double*. Valeur de fiabilité définie pour le modèle.

productType: *chaîne*. Type de produit du modèle.

Exemple de corps

```
[{"modelName": "walkermodel1", "productType": "walkermodel1", "dataFormat": "png", "modelType": "classification", "retrainPolicy": {"imageNumber": "1000", "accuracy": "70", "scheduler": "Weekly, Sunday", "reuseDayNumber": "", "maxPieceNumber": "", "ratio": ""}, "description": "this is a test", "groupIds": "b5164942-370c-418d-9fc2-f6ffd860b79d", "parameters": {"imageSize": "100*100", "confidence": 100, "trainParam": {"ratioTrain": 80, "ratioVal": 20, "gamma": 0.1, "stepsize": 33, "maxIter": 100, "learningRate": 0.01, "learningRatePolicy": "step", "test_iter": 1, "test_interval": 2, "snapshot": 10, "network": "GoogLeNet", "algorithm": "CNN"}}}]
```

Réponse indiquant la réussite

201

```
[{"modelId": "4d0dfe01-7a26-4071-92fb-ec597e7863ce", "modelName": "walkermodel1", "description": "this is a test", "productType": "walkermodel1", "groupIds": "b5164942-370c-418d-9fc2-f6ffd860b79d", "dataFormat": "png", "createdBy": "automm@cn.ibm.com", "updatedAt": "2018-02-13T00:44:31.288-06:00", "parameters": {"imageSize": "100*100", "confidence": 100, "trainParam": {"ratioTrain": 80, "stepsize": 33, "test_iter": 1, "snapshot": 10, "momentum": 0.9, "ratioVal": 20, "learningRate": 0.01, "display": 1, "learningRatePolicy": "step", "weight_decay": 1.0E-5, "network": "GoogLeNet", "test_interval": 2, "algorithm": "CNN", "gamma": 0.1, "solver_type": "SGD", "maxIter": 100}}, "retrainPolicy": {"ratio": "", "scheduler": "Weekly, Sunday", "maxPieceNumber": "", "reuseDayNumber": "", "imageNumber": "1000", "accuracy": "70"}}}]
```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

modelName: *chaîne*. Nom du modèle.

productType: *chaîne*. Type de produit.

createdBy: *chaîne*. Propriétaire du modèle.

updatedAt: *heure*. Heure de la dernière mise à jour du modèle.

productType: *chaîne*. Type de produit du modèle.

description: *chaîne*. Description du modèle.

groupIds: *chaîne*. ID du groupe de données du modèle.

dataFormat: *chaîne*. Format des données de modèle.

retrainPolicy: *objet JSON*. Règle de nouvel entraînement du modèle.

scheduler: *date*. Date de conservation du modèle, de lundi à dimanche (Monday à Sunday).

imageNumber: *entier*. Un nombre d'images supérieur à ce nombre peut être entraîné à nouveau.

accuracy: *double*. Une valeur de précision du modèle inférieure à ce nombre peut être entraînée à nouveau.

parameters: *objet* JSON. Contient des informations sur la taille de l'image et la fiabilité du modèle.

confidence: *double*. Valeur de fiabilité définie pour le modèle.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c
77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
-H "Content-Type:application/json;charset=UTF-8" "https://9.112.229.91/
ibm/iotm/service/model?&tenant=VITest&solution=vi&user=auto_mm"
--data '{"modelName":"testcreatemodel11","imagegroupList":[{"id":
\b5164942-370c-418d-9fc2-f6ffd860b79d\',"dataFiles\":[{"id":\
\b5164942-370c-418d-9fc2-f6ffd860b79d--b5164942-370c-418d-9fc2-
f6ffd860b79d_74674410-06df-4a9e-a9e9-12f5038efc22\',"name":\
"Good_val.zip\',"count":5,\'updatedTime\':"12/27/2017,
10:38:13 AM"}],\'name\':"Good\',"updatedTime\':"12/27/2017,
10:36:10 AM"}]','productType":"testcreatemodel11","dataFormat":
"png","modelType":"classification","retrainPolicy":{"imageNumber":
"1000","accuracy":"70","scheduler":"Weekly, Sunday","reuseDayNumber":
"","maxPieceNumber":"","ratio":""},"description":"this is a test",
"groupIds":"b5164942-370c-418d-9fc2-f6ffd860b79d","parameters":
{"imageSize":"100*100","confidence":100,"trainParam":{"ratioTrain":
80,"ratioVal":20,"gamma":0.1,"stepsize":33,"maxIter":100,"learningRate":
0.01,"learningRatePolicy":"step","test_iter":1,"test_interval":2,
"snapshot":10,"network":"GoogLeNet","algorithm":"CNN"}}}'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Mettre à jour un modèle

Permet de mettre à jour un modèle existant particulier. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/model/*ID_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelId: *chaîne*. ID du modèle. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

groupIds: *chaîne*. ID du groupe de données.

Exemple de corps

```
[{"groupIds":"45805b3c-f234-4f18-a55e-61ce203a7db4,678a3f42-f9d4-48a2-bb
29-c226d61a08ea"}]
```

Réponse indiquant la réussite

```
200  
[{}]
```

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -H  
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd  
34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H "Content-Type:application  
/json;charset=UTF-8" -X PUT "https://9.112.229.91:9447/ibm/iotm/service/model/106ef  
cb7-1338-4ed2-b8ca-27b45e11e865?user=auto_mm&solution=vi" --data '{"groupIds":"9f  
7aa9d6-24d6-4611-8d4d-79a52d2bab02"}'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Supprimer un modèle particulier

Permet de supprimer un modèle existant particulier et toutes les instances de modèle liées. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/model/*ID_modèle*

Méthode

Type de demande DELETE

Paramètres d'URL

modelId: *chaîne*. ID du modèle. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

```
[{}]
```

Réponse indiquant la réussite

```
200  
[{}]
```

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -X DELETE -H "APIKEY:8b79658225de5  
3488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e53  
3dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"  
"https://9.112.229.91:9447/ibm/iotm/service/model/
```

```
106efcb7-1338-4ed2-b8ca-27b45e11e865?user=auto_mm&solution=
vi" -d '[{}]
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir tous les modèles partagés

Permet d'obtenir tous les modèles partagés. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/model/

Méthode

Type de demande GET

Paramètres d'URL

category: 'shared'. Utilisé pour identifier le catalogue de modèles. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
[{"modelId": "0f9c8d47-5ccc-4e52-8e0d-0596e792a2cd",
"groupIds": "e5e83a2d-5814-45b3-af40-74096198cd39", "modelName":
"Car Seat Defect Inspection", "createdBy": "vimodelmanager@163.com",
"modelType": "objectdetection", "description": "This is the object
detection model we build to detect the wrinkle defect for car seat.
The inspection result will identify whether there is wrinkle defect
on the car and localize the defect on the image.", "statusStatistics":
{"deployed": "1"}, "dataFormat": "jpg", "retrainPolicy": {"scheduler":
"Weekly, Sunday", "ratio": "", "reuseDayNumber": "", "maxPieceNumber":
"", "imageNumber": "1000", "accuracy": "70"}, "trainedBy":
"vimodelmanager@163.com", "parameters": {"odRetrainUrl":
"/user\\VI\\VI\\modelist\\carseatfrcnn_1516957839865.zip",
"lastRetrainData":
"[{"groupId": "e5e83a2d-5814-45b3-af40-74096198cd39"}]",
"imageSize": "100*100", "confidence": "100"}, "updatedAt":
"2018-01-26T03:11:05.804-06:00", "productType": "carseat"}]
```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

groupIds: *chaîne*. ID du groupe de données du modèle.

modelName: *chaîne*. Nom du modèle.

createdBy: *chaîne*. Propriétaire du modèle.

modelType: *chaîne*. Type de modèle.

description: *chaîne*. Description du modèle.

updatedAt: *heure*. Heure de la dernière mise à jour du modèle.

productType: *chaîne*. Type de produit du modèle.
StatusStatistics: *chaîne*. Statut du modèle.
dataFormat: *chaîne*. Format des données de modèle.
retrainPolicy: *objet JSON*. Règle de nouvel entraînement du modèle.
scheduler: *date*. Date de conservation du modèle, de lundi à dimanche (Monday à Sunday).
imageNumber: *entier*. Un nombre d'images supérieur à ce nombre peut être entraîné à nouveau.
accuracy: *double*. Une valeur de précision du modèle inférieure à ce nombre peut être entraînée à nouveau.
parameters: *objet JSON*. Contient des informations sur la taille de l'image et la fiabilité du modèle.
imageSize: *double * double*. Taille de l'image.
confidence: *double*. Valeur de fiabilité définie pour le modèle.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb
8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713
bf305a158fdf485cc6f275f5" " https://9.112.229.91:9447/ibm/
iotm/service/model?category=shared&tenant=Q3T1&solution=
vi&user=testuser"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services d'instance de modèle

Les services d'instance de modèle permettent d'effectuer des tâches telles que créer une instance de modèle, renvoyer une instance de modèle appartenant à un modèle et renvoyer une instance de modèle particulière.

Créer une instance de modèle

Permet de créer une instance de modèle. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance

Méthode

Type de demande POST

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

modelID: *chaîne*. ID du modèle.

trainData: *objet JSON*. Données d'apprentissage du modèle.

groupId: *chaîne*. ID du groupe de données.

fileIds: *chaîne*. ID du fichier de données.

Exemple de corps

```
[
  {
    "modelId": "91a21e63-17e6-4e0e-bbe8-5c6b9d599a35",
    "trainData": [
      {
        "groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02",
        "fileIds": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b"
      },
      {
        "groupId": "d84c028e-235a-4c13-8a77-894500b8a736",
        "fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894"
      },
      {
        "groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb",
        "fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-e74c1578eebe"
      }
    ]
  }
]
```

Réponse indiquant la réussite

```
200
[
  {
    "status": "deployed",
    "validateData": [
      {
        "groupId": "1a7d4c81-36ff-4a3e-a614-9e996fb380ba"
      }
    ],
    "instanceId": "4cc40ab4-d3b3-4911-b7a0-90281248c075_1542340917709",
    "validateResult": {
      "matrix": [
        [ "", "scratch", "Chris" ],
        [ "mAP", "-", "-" ],
        [ "recall", "-", "-" ]
      ],
      "type": "objectdetection",
      "accuracy": "-",
      "fileCount": "0"
    },
    "trainedBy": "demoadm@cn.ibm.com",
    "validateTime": "2018-11-15T22:01:57.726-06:00",
    "fileCount": "0",
    "modelId": "4cc40ab4-d3b3-4911-b7a0-90281248c075",
    "createdBy": "demoadm@cn.ibm.com",
    "instanceName": "1",
    "parameters": {
      "edges": [
        { "id": "1542102975873" },
        { "id": "1508476898000" }
      ],
      "modelType": "objectdetection",
      "attached": true,
      "edgesDeploying": []
    },
    "isDeletable": false,
    "modelUrl": "/user/Q3T1/VI/modellist/frcnnmodel_1542340914671.zip",
    "trainData": [
      { "groupId": "1a7d4c81-36ff-4a3e-a614-9e996fb380ba" }
    ],
    "accuracy": "-"
  }
]
```


Paramètres d'URL

modelId: *chaîne*. ID du modèle. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "modelId": "91a21e63-17e6-4e0e-bbe8-5c6b9d599a35",
    "createdBy": "auto_mm",
    "status": "draft",
    "instanceName": "1",
    "instanceId": "91a21e63-17e6-4e0e-bbe8-5c6b9d599a35_1512029557869",
    "trainData": [
      {
        "fileIds": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b",
        "groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02"
      },
      {
        "fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894",
        "groupId": "d84c028e-235a-4c13-8a77-894500b8a736"
      },
      {
        "fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-e74c1578eebe",
        "groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb"
      }
    ],
    "fileCount": "9",
    "updatedAt": "2017-11-30T16:12:37.869+08:00"
  }
]
```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

createdBy: *chaîne*. Propriétaire de l'instance de modèle.

status: *chaîne*. Statut de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

instanceName: *entier*. Nom de l'instance de modèle.

trainData: *objet JSON*. Données d'apprentissage du modèle.

groupId: *chaîne*. ID du groupe de données.

fileIds: *chaîne*. ID du fichier de données.

fileCount: *entier*. Nombre total d'images de modèle.

updatedAt: *heure*. Heure de la dernière mise à jour de l'instance de modèle.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533d
d34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112.229.91:9447
/ibm/iotm/service/modelInstance?modelId=91a21e63-17e6-4e0e-bbe8-5c6b9d599a35&user=
auto_mm&solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir une instance de modèle particulière

Permet d'obtenir une instance de modèle particulière. Le gestionnaire de modèles et le spécialiste des données disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande GET

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "modelId": "91a21e63-17e6-4e0e-bbe8-5c6b9d599a35",
    "createdBy": "auto_mm",
    "status": "draft",
    "instanceName": "1",
    "instanceId": "91a21e63-17e6-4e0e-bbe8-5c6b9d599a35_1512029557869",
    "trainData": [
      {
        "fileIds": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b",
        "groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02"
      },
      {
        "fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894",
        "groupId": "d84c028e-235a-4c13-8a77-894500b8a736"
      },
      {
        "fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-e74c1578eebe",
        "groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb"
      }
    ],
    "updatedAt": "2017-11-30T16:12:37.869+08:00"
  }
]
```

```
}  
]
```

Éléments de réponse

`instanceId`: *chaîne*. ID de l'instance de modèle.
`modelId`: *chaîne*. ID du modèle.
`instanceName`: *entier*. Nom de l'instance de modèle.
`createdBy`: *chaîne*. Propriétaire de l'instance de modèle.
`updatedAtTime`: *heure*. Heure de la dernière mise à jour de l'instance de modèle.
`status`: *chaîne*. Statut de l'instance de modèle.
`trainData`: *objet JSON*. Données d'apprentissage du modèle.
`groupId`: *chaîne*. ID du groupe de données.
`fileIds`: *chaîne*. ID du fichier de données.

Exemple d'appel

```
curl -k -H  
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c  
77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"  
"https://9.112.229.91:9447/ibm/iotm/service/modelInstance/91a21e63  
-17e6-4e0e-bbe8-5c6b9d599a35_1512029557869?user=auto_mm&solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir le résultat de la validation de l'instance de modèle

Permet d'obtenir le résultat de la validation de l'instance de modèle pour le rapport. Le gestionnaire de modèles et le spécialiste des données disposent des droits permettant d'utiliser cette API.

URL

`/ibm/iotm/service/modelInstance/ID_instance_modèle/validateResult`

Méthode

Type de demande GET

Paramètres d'URL

`modelInstanceId`: *chaîne*. ID de l'instance de modèle. Obligatoire.
`user`: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.
`solution`: *chaîne*. Utilisé pour identifier la solution. Obligatoire.
`tenant`: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

`APIKEY`: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200  
[
```

```

{
  "instanceName": "FirstBlood_2017-08-08 19:21:53.888",
  "validateResult": {
    "matrix": [
      [
        "",
        "type1",
        "type2",
        "type3"
      ],
      [
        "type1",
        "77,35,45",
        "0,0,40",
        "6,3,50"
      ],
      [
        "type2",
        "17,8,45",
        "100,40,40",
        "4,2,50"
      ],
      [
        "type3",
        "4,2,45",
        "0,0,40",
        "90,45,50"
      ],
      [
        "Total",
        "45",
        "40",
        "50"
      ]
    ],
    "accuracy": "84.56"
  },
  "validateTime": "2017-08-09T22:58:46.416+08:00",
  "fileCount": "200"
}
]

```

Éléments de réponse

`instanceName`: *entier*. Nom de l'instance de modèle.

`validateResults`: *objet JSON*. Résultat de la validation de l'instance de modèle.

`validateTime`: *heure*. Heure de la validation de l'instance de modèle.

`fileCount`: *entier*. Nombre d'images de l'instance de modèle.

`accuracy`: *double*. Valeur de précision de l'instance de modèle.

Exemple d'appel

```

curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e53
3dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112.229.91:9447/
ibm/iotm/service/modelInstance/91a21e63-17e6-4e0e-bbe8-5c6b9d599a35_1512029557869/
validateResult?user=auto_mm&solution=vi"

```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services de résultat d'inspection

Les services de résultat d'inspection permettent d'effectuer des tâches telles qu'obtenir une liste de résultats d'inspection, obtenir les détails d'un résultat d'inspection et obtenir une vue générale d'une cellule de résultat d'inspection.

Obtenir la liste de résultats d'inspection

Permet d'obtenir la liste de résultats d'inspection. L'inspecteur et le superviseur disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/inspectResult

Méthode

Type de demande GET

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

isConfirmed: *entier*. 0 signifie que le résultat d'inspection n'est pas confirmé. 1 signifie que le résultat d'inspection est confirmé. 2 signifie que le résultat d'inspection est inconnu. Facultatif.

cell: *chaîne*. Utilisé pour identifier la cellule. Facultatif.

inspectTime: *heure*. Utilisé pour identifier l'heure de l'inspection.

reverse: *valeur booléenne*. Utilisé pour identifier l'ordre de la liste de résultats d'inspection.

pageCount: *entier*. Nombre de pages.

sampleRate: *entier*. Fréquence de vérification d'échantillon.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[{"id":"plant1,line1,cell1_1511768220935_AUTO|auto_1511768206483.jpg",
"isConfirmed":"0","inspectFileUrl":"imageserver/911222954/\\/AUTO|
auto_plant1_line1_cell1_1511768206483.jpg","isUpdated":"0",
"defectTypeCount":"1","cell":"plant1,line1,cell1","instanceId":"ebf9
4e98-958c-48b0-8377-826241ffcd8_1511765289053","defectCount":"1",
"confidence":"99.87","inspectId":"auto_1511768206483.jpg",
"inspectTime":"1511768220935"},{"id":"plant1,line1,
cell1_1511761780903_AUTO|auto_1511761766470.jpg","isConfirmed":"0",
inspectFileUrl":"imageserver/911222954/\\/AUTO|
auto_plant1_line1_cell1_1511761766470.jpg","isUpdated":"0",
"defectTypeCount":"1","cell":"plant1,line1,cell1","instanceId":
"9ab990d6-465a-48b0-838c-416c8e4a9879_1511761328538","defectCount":
"1","confidence":"99.79","inspectId":"auto_1511761766470.jpg",
"inspectTime":"1511761780903"},
{"id":"plant1,line1,cell1_1511761559536_AUTO|auto_1511761541254.jpg","isConfirmed":"0","i
nspectFileUrl":"imageserver/911222954/\\/AUTO|
auto_plant1_line1_cell1_1511761541254.jpg","isUpdated":"0",
"defectTypeCount":"1","cell":"plant1,line1,cell1","instanceId":
"9ab990d6-465a-48b0-838c-416c8e4a9879_1511761328538","defectCount":
```

```
"1", "confidence": "99.87", "inspectId": "auto_1511761541254.jpg",
"inspectTime": "1511761559536"}, {"id": "plant1,line1,
cell1_1511421414687_AUTO|auto_1511421410914.jpg", "isConfirmed": "0",
"inspectFileUrl": "imageserver/911222954/\\AUTO|
auto_plant1_line1_cell1_1511421410914.jpg", "isUpdated": "0",
"defectTypeCount": "1", "cell": "plant1,line1,cell1", "instanceId":
"ae509ec0-e1d6-4827-afd1-a8af598eb28c_1511419442550",
"defectCount": "1", "confidence": "99.99", "inspectId":
"auto_1511421410914.jpg", "inspectTime": "1511421414687"}]
```

Eléments de réponse

id: *chaîne*. ID de l'inspection.

isConfirmed: *entier*. 0 signifie que le résultat d'inspection n'est pas confirmé. 1 signifie que le résultat d'inspection est confirmé. 2 signifie que le résultat d'inspection est inconnu.

cell: *chaîne*. Informations sur la cellule.

inspectFileUrl: *chaîne*. URL du fichier d'inspection.

defectTypeCount: *entier*. Nombre de types de défaut.

defectCount: *entier*. Nombre de défauts.

confidence: *double*. Niveau de fiabilité du défaut.

inspectId: *chaîne*. ID de l'inspection.

inspectTime: *heure*. Heure de l'inspection.

Exemple d'appel

```
curl -k -H
"APIKEY: 8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579
e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112.22
9.91:9447/ibm/iotm/service/inspectResult?user=auto_inspector&solution=vi&isCon
firmed=0&cell=plant1,line1,cell1&sampleRate=10&pageCount=1&reverse=true"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir les détails du résultat d'inspection

Permet d'obtenir les détails du résultat d'inspection. L'inspecteur et le superviseur disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/inspectResult/ID_résultat_inspection

Méthode

Type de demande GET

Paramètres d'URL

inspectResultId: *chaîne*. Utilisé pour identifier l'ID d'inspection.

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "isConfirmed": "0",
    "inspectFileUrl": "imageserver/edge1/image1.jpg",
    "isUpdated": "0",
    "cell": "plant1,line1,cell1",
    "instanceId": "6e4bfc6b-0ddb-4c85-81e7-0ea789681ae8_775e39b9-fedd-4418-a5b3-70de6b86fbec",
    "inspectResult": [
      {
        "position": {
          "height": 331,
          "width": 467,
          "y": 1653,
          "x": 897
        },
        "probableTypes": [
          {
            "confidence": 100,
            "type": "NoDefect"
          },
          {
            "confidence": 0,
            "type": "Defect"
          },
          {
            "confidence": 0,
            "type": "MisT"
          }
        ]
      }
    ]
  }
]
```

Éléments de réponse

isConfirmed: *entier*. 0 signifie que le résultat d'inspection n'est pas confirmé. 1 signifie que le résultat d'inspection est confirmé. 2 signifie que le résultat d'inspection est inconnu.

inspectFileUrl: *chaîne*. URL du fichier d'inspection.

inspectId: *chaîne*. ID de l'inspection.

inspectResult: *objet JSON*. Résultat de l'inspection.

position: *tableau JSON*. Hauteur, largeur et coordonnées x et y de l'inspection.

probableTypes: *objet JSON*. Chaque type de défaut et son niveau de fiabilité.

Exemple d'appel

```
curl -k -H
"APIKEY: 8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2
c77d4f0579e533dd34gggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.91:9447/ibm/iotm/service/inspectResult/plant1,line1,
cell1_1511166988402_AUTO|auto_1511166972283.jpg?user=auto_inspector&
solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir la vue générale d'une cellule de résultat d'inspection

Permet d'obtenir la vue générale d'une cellule de résultat d'inspection. L'inspecteur et le superviseur disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/inspectResultCell

Méthode

Type de demande GET

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Aucun

Réponse indiquant la réussite

```
200
[
  {
    "confirmed": 1,
    "cell": "plant1,line1,cell1",
    "unknown": 0,
    "unconfirmed": 10
  }
]
```

Éléments de réponse

confirmed: *entier*. Numéro d'inspecteur confirmé.

unconfirmed: *entier*. Numéro d'inspecteur non confirmé.

unknown: *entier*. Inspecteur inconnu.

cell: *chaîne*. Cellule d'image.

Exemple d'appel

```
curl -k -H
"APIKEY: 8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c7
7d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "
https://9.112.229.91:9447/ibm/iotm/service/inspectResultCell?user=auto_
inspector&solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Confirmer les résultats d'inspection

Permet de confirmer les résultats d'inspection. L'inspecteur et le superviseur disposent des droits permettant d'utiliser cette API.

URL

`/ibm/iotm/service/inspectResult/ID_résultat_inspection`

Méthode

Type de demande PUT

Paramètres d'URL

`inspectResultId`: *chaîne*. Utilisé pour identifier l'ID d'inspecteur.

`user`: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

`solution`: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

`tenant`: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

`position`: *tableau JSON*. Hauteur, largeur et coordonnées x et y de l'inspection.

`probableTypes`: *objet JSON*. Chaque type de défaut et son niveau de fiabilité.

`logs`: *objet JSON*. Contient l'action de confirmation, l'opérateur et l'heure de mise à jour.

`description`: *chaîne*. Description de la confirmation de l'inspecteur.

Exemple de corps

```
[
  {
    "confirmedResult": [
      {
        "position": {
          "height": 3231,
          "width": 467,
          "y": 1653,
          "x": 897
        },
        "probableTypes": [
          {
            "confidence": 100,
            "type": "NoDefect"
          },
          {
            "confidence": 0,
            "type": "Defect"
          },
          {
            "confidence": 0,
            "type": "MisT"
          }
        ],
        "logs": [{"action": "confirm",
          "operator": "inspector1",
          "updateTime": "1503657054321"}]
      }
    ],
    "description": "VI result is OK"
  }
]
```

Réponse indiquant la réussite

```
200
[{}]
```

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -H"APIKEY: 8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8
c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H
"Content-Type:application/json;charset=UTF-8" -X PUT "https://9.112.229.91:9447/ibm
/iotm/service/inspectResult/plant1,line1,cell1_1511507025284_AUTO|auto_1511507012702
.jpg?user=auto_inspector&solution=vi" --data '[ { "confirmedResult": [ {
"position": { "height": 430, "width": 552, "x": 1254, "y": 2123 },
"probableTypes": [ { "confidence": 99.87, "type": "Defect" }, {
"confidence": 0.1, "type": "MisT" }, { "confidence": 0.03, "type": "NoDefect"
} ], "logs": [{"action": "confirm", "operator": "auto_super",
"updateTime": "1503657054321"}] } ], "description": "VI result is OK" } ]'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Supprimer les résultats d'inspection

Supprime les résultats d'inspection. L'inspecteur et le superviseur disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/inspectResult

Méthode

Type de demande DELETE

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

cell: *chaîne*. Utilisé pour identifier la cellule. Obligatoire.

inspectTime: *heure*. Utilisé pour identifier l'heure de l'inspection. Les résultats d'inspection générés avant cette heure sont supprimés. Facultatif.

instanceId: *chaîne*. Utilisé pour identifier l'ID d'instance. Les résultats d'inspection générés par cet ID d'instance sont supprimés. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

Aucun.

Réponse indiquant la réussite

```
200
[{}]
```

Eléments de réponse

Aucun.

Exemple d'appel

```
curl -k -H "APIKEY: 8b79658225de53488321fb
7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34
gg959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
-H "Content-Type:application/json;charset=UTF-8" -X DELETE
"https:// 9.112.229.91:9447/ibm/iotm/service/inspectResult?instanceId=
04b61494-99ec-49f4-9a0f-0125650c40dd_1520316716077&
tenant=Q3T1&solution=vi&user=demoadm@cn.ibm.com&cell=plant2,line2,cell2&inspectTime=
1520324778324"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services d'action d'instance de modèle

Les services d'action d'instance de modèle permettent d'effectuer des tâches telles que soumettre une instance de modèle, joindre un fichier de modèle et valider une instance de modèle.

Entraîner une instance de modèle

Entraînez une instance de modèle en utilisant un modèle d'apprentissage en profondeur. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

action: *chaîne*. Action. Définissez train. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-Type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

```
[{}]
```

Réponse indiquant la réussite

```
200
[{"status": "waiting", "instanceId":
"1e760614-ab0e-4c2a-be2e-435505087fe4_1518060853002",
"parameters": {"modelType": "classification", "trainType":
"online_train"}}]
```

Éléments de réponse

status: *chaîne*. Statut du nouvel entraînement de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

updatedAtTime: *heure*. Heure de la dernière mise à jour de l'instance de modèle.

parameters: *objet JSON*. Informations sur le type de modèle et le type d'entraînement.

Exemple d'appel

```
curl -k -X PUT -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5af
b8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc37
13bf305a158fdf485cc6f275f5" -H "Content-Type:
application/json;charset=UTF-8"
"https://int_iotm.predictivesolutionsapps.ibmcloud.com/
ibm/iotm/service/modelInstance/1e760614-ab0e-4c2a-be2e-
435505087fe4_1518060853002?action=train&tenant=Q3T1&
solution=vi&user=admin@example.com
" --data ' [{}]
```

Remarques

Ce service prépare des images et procède à un entraînement en ligne. Vous pouvez consulter le statut de l'instance de modèle en utilisant l'appel API "Obtention d'une instance de modèle particulière".

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Télécharger un fichier journal d'entraînement

Téléchargez un fichier journal pour l'entraînement d'un modèle. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/vi/service/logFile

Méthode

Type de demande GET

Paramètres d'URL

instanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

Aucun.

Réponse indiquant la réussite

Contenu binaire du fichier journal.

Éléments de réponse

Aucun.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d
```

```
4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.91:9447/ibm/iotm/vi/service/logFile?instanceId=06c87
0e5-1701-419a-8a84-0f40ee82fe33_11260029297&user=auto_mm&solution=vi"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Valider l'instance de modèle

Permet de valider une instance de modèle entraînée. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

action: *chaîne*. Action. Définissez validate. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.

Content-Disposition: form-data; name="files[]"; filename="model.zip".

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

validateData: *objet JSON*. Données de validation du modèle.

groupId: *chaîne*. ID du groupe de données.

fileIds: *chaîne*. ID du fichier de données.

Exemple de corps

```
[{"validateData": [{"fileIds": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b", "groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02"}, {"fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894", "groupId": "d84c028e-235a-4c13-8a77-894500b8a736"}, {"fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-e74c1578eebe", "groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb"} ] }
```

Réponse indiquant la réussite

```
200
[{"validateData": [{"fileIds": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b", "groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02"}, {"fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894", "groupId": "d84c028e-235a-4c13-8a77-894500b8a736"}, {"fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-e74c1578eebe", "groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb"} ], "status": "validating", "instanceId": "ef12294e-2d09-4acd-9946-00a09bdeeba2_1512092239371"}]
```

Éléments de réponse

validateData: *objet JSON*. Données de validation du modèle.

groupId: *chaîne*. ID du groupe de données.

fileIds: *chaîne*. ID du fichier de données.

Exemple d'appel

```
curl -k -H "APIKEY:8b79658225de53488321fb7bb657f9f161ac
d2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf30
5a158fd485cc6f275f5" -H "Content-Type:application/json;charset=UTF-8" -X PUT
"https://9.112.229.91:9447/ibm/iotm/service/modelInstance/ef12294e-2d09-
4acd-9946-00a09bdeeba2_1512092239371?action=validate&user=auto_mm&solution=vi"
--data '[{"validateData": [{"fileIds": "9f7aa9d6-24d6-4611-8d4d-
79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b", "groupId":
"9f7aa9d6-24d6-4611-8d4d-79a52d2bab02"}, {"fileIds":
"d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894",
"groupId": "d84c028e-235a-4c13-8a77-894500b8a736"},
{"fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_
bd0c8246-7208-44d8-a2fa-e74c1578eebe",
"groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb"} ] ]'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Rejeter une instance de modèle

Permet de rejeter une instance de modèle validée. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

action: *chaîne*. Action. Définissez reject. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

```
[{}]
```

Réponse indiquant la réussite

```
200
[
  {
    "status": "rejected",
    "instanceId": "e23b6796-4f4a-4c5d-8fb7-fa3f729f591c_1510812725969"
  }
]
```

Éléments de réponse

status: *chaîne*. Statut de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

Exemple d'appel

```
curl -k -H "APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H "Content-Type:application/json; charset=UTF-8" -X PUT "https://9.112.229.91:9447/ibm/iotm/service/modelInstance/ef12294e-2d09-4acd-9946-00a09bdeeba2_1512092239371?action=reject&user=auto_mm&solution=vi" --data '{}'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Déployer une instance de modèle

Permet de déployer une instance de modèle acceptée. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

action: *chaîne*. Action. Définissez publish. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

Parameters: *objet JSON*. ID des serveurs de périphérie sur lesquels le modèle va être déployé.

En-têtes

Content-type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

```
[{"parameters": {"edges": [{"id": "1508476898000"}, {"id": "1542102975873"}, ...]}}
```

Réponse indiquant la réussite

```
[{"status": "deployed", "instanceId": "96b651a9-2ba0-4392-aca4-1463320a19c2_1542354262146"}]
```

Éléments de réponse

status: *chaîne*. Statut de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

parameters: *objet JSON*. Nom et adresse IP du serveur de périphérie.

modelType: *chaîne*. Type de modèle.

Exemple d'appel

```
curl -k -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab
2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
-H "Content-Type:application/json;charset=UTF-8" -X PUT
"https://9.112.229.91:9447/ibm/iotm/service/modelInstance/ef12294e-
2d09-4acd-9946-00a09bdeeba2_1512092239371?action=publish&user=
auto_mm&solution=vi" --data '{"parameters":{"edges":[{"id":
"1508476898000"}, {"id": "1542102975873"}, ...]}}'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Entraîner à nouveau l'instance de modèle

Entraînez à nouveau une instance de modèle déployée. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

action: *chaîne*. Action. Définissez retrain. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-Type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

retrainData: *objet JSON*. Données de nouvel entraînement pour le modèle.

groupId: *chaîne*. ID du groupe de données.

fileIds: *chaîne*. ID du fichier de données.

Exemple de corps

```
[{"trainData": [ {"fileIds": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b",
"groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02" },
{ "fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-ae1b9f7db894",
"groupId": "d84c028e-235a-4c13-8a77-894500b8a736" },
{ "fileIds": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-e74c1578eebe",
"groupId": "c5f11ed9-db49-403a-972c-cd6cb1d0f8eb"
} ] }
```

Réponse indiquant la réussite

```
200
[{"modelId": "ef12294e-2d09-4acd-9946-00a09bdeeba2", "status":
"waiting", "instanceId": "ef12294e-2d09-4acd-9946-00a09bdeeba2_
1512095854903", "trainData": [{"fileIds": "9f7aa9d6-24d6-4611-
8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-bbca93abd80b",
```

```
"groupId": "9f7aa9d6-24d6-4611-8d4d-79a52d2bab02"},
{"fileIds": "d84c028e-235a-4c13-8a77-894500b8a736_
fe201c83-b138-4dca-b918-ae1b9f7db894", "groupId":
"d84c028e-235a-4c13-8a77-894500b8a736"}, {"fileIds":
"c5f11ed9-db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-
44d8-a2fa-e74c1578eebe", "groupId": "c5f11ed9-db49-403a-
972c-cd6cb1d0f8eb"}]}}
```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

status: *chaîne*. Statut du nouvel entraînement de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

trainData: *objet JSON*. Données d'apprentissage du modèle.

groupId: *chaîne*. ID du groupe de données.

fileIds: *chaîne*. ID du fichier de données.

Exemple d'appel

```
curl -k -H "APIKEY:8b79658225de53488321fb
7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd
34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H
"Content-Type:application/json;charset=UTF-8" -X PUT
"https://9.112.229.91:9447/ibm/iotm/service/modelInstance/
ef12294e-2d09-4acd-9946-00a09bdeeba2_1512092239371?
action=retrain&user=auto_mm&solution=vi" --data
' [{"trainData": [ {"fileIds": "9f7aa9d6-24d6-
4611-8d4d-79a52d2bab02_76e8b2cb-057a-4904-8fad-
bbca93abd80b", "groupId": "9f7aa9d6-24d6-4611-
8d4d-79a52d2bab02" }, {"fileIds": "d84c028e-
235a-4c13-8a77-894500b8a736_fe201c83-b138-4dca-b918-
ae1b9f7db894", "groupId": "d84c028e-235a-4c13-
8a77-894500b8a736" }, {"fileIds": "c5f11ed9-
db49-403a-972c-cd6cb1d0f8eb_bd0c8246-7208-44d8-a2fa-
e74c1578eebe", "groupId": "c5f11ed9-db49-403a-972c-
cd6cb1d0f8eb" } ] } ]'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Annuler le déploiement d'une instance de modèle

Permet d'annuler le déploiement d'une instance de modèle déployée. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/modelInstance/*ID_instance_modèle*

Méthode

Type de demande PUT

Paramètres d'URL

modelInstanceId: *chaîne*. ID de l'instance de modèle. Obligatoire.

action: *chaîne*. Action. Définissez un `deploy`. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

Parameters: *objet JSON*. ID du serveur de périphérie sur lequel le déploiement de l'instance de modèle va être annulé.

En-têtes

Content-Type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

```
[{"parameters": {"edges": [{"id": "1508476898000"}]}}
```

Réponse indiquant la réussite

```
200
[{"status": "undeployed", "instanceId": "ef12294e-2d09-4acd-9946-00a09bdeeba2_1512092239371"}]
```

Éléments de réponse

status: *chaîne*. Statut du nouvel entraînement de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

Exemple d'appel

```
curl -k -H "APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H "Content-Type:application/json;charset=UTF-8" -X PUT "https://9.112.229.91:9447/ibm/iotm/service/modelInstance/ef12294e-2d09-4acd-9946-00a09bdeeba2_1512092239371?action=undeploy&user=auto_mm&solution=vi" --data '[{"parameters": {"edges": [{"id": "1508476898000"}]}}'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services de serveur de périphérie

Les services de serveur de périphérie permettent d'effectuer des tâches telles que créer un serveur de périphérie, obtenir un serveur de périphérie et mettre à jour un serveur de périphérie.

Créer un serveur de périphérie

Permet de créer un serveur de périphérie. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/vi/service/edge

Méthode

Type de demande POST

Paramètres d'URL

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-Type: application/json.

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

`name`: *chaîne*. Nom du serveur de périphérie. Obligatoire.

`ip`: *IP*. Adresse IP du serveur de périphérie. Obligatoire.

`port`: *entier*. Port du serveur de périphérie. Obligatoire.

`userName`: *chaîne*. Nom d'utilisateur pour la machine virtuelle du serveur de périphérie. Obligatoire.

`passWord`: *chaîne*. Mot de passe pour la machine virtuelle du serveur de périphérie. Obligatoire.

`clusterType`: *chaîne*. Type de cluster du serveur de périphérie. Spécifiez `slave` ou `master`. Obligatoire.

`online`: *chaîne*. Indique si le serveur de périphérie est connecté. Spécifiez `true` si le serveur de périphérie est connecté ou `false` s'il est déconnecté. Obligatoire.

Exemple de corps

```
{ "name": "edge_auto", "ip": "9.112.229.54",  
  "port": "22", "userName": "root", "passWord": "U2FtcGx1QDY2NiE=",  
  "clusterType": "mater", "online": "true" }
```

Réponse indiquant la réussite

```
{ "port": "22", "clusterType": "master", "deployPath": "\\home\\pmqopsadmin", "hierarchyList":  
  [ "plant  
  1,line1,cell1", "plant2,line2,cell2"], "servicePass": "passw0rd@", "deletePolicy":  
  { "schedule": false, "rate  
  ": 0.2, "onDay": 7, "afterDay": "1", "frequency": "weekly" }, "version": "2018Q4.01", "ip": "viedge1", "o  
  nline  
  ": "true", "id": "1508476898000", "upgradeFlag": "false", "createdBy": "demoadm@cn.ibm.com", "servic  
  eUser":  
  "admin", "name": "master", "sshUser": "pmqopsadmin", "updatedAt": "2018-11-16T03:54:13.382-06:00  
  " } }
```

Éléments de réponse

`id`: *chaîne*. ID du serveur de périphérie.

`deletePolicy`: *objet JSON*. Règle de suppression du résultat de l'inspection sur le serveur de périphérie.

`version`: *chaîne*. Informations de version du serveur de périphérie.

`upgradeFlag`: *chaîne*. Indique si le serveur de périphérie doit être mis à niveau.

Exemple d'appel

```
curl -k -H "APIKEY:8b79658225d  
e53488321fb7bb657f9f161acd2ea830a5afb8c149  
f6aab2c77d4f0579e533dd34ggb959317ac69ff  
73f886fc3713bf305a158fdf485cc6f275f5" -H  
"Content-Type:application/json;charset=UTF-8"  
"https://9.112.229.91:9447/ibm/iotm/vi/service/edge?  
user=auto_mm&solution=vi " --data '{ "name":  
"edge_auto", "ip": "9.112.229.54", "port": "22",  
"userName": "root", "passWord": "U2FtcGx1QDY2NiE=",  
"clusterType": "master", "online": "true" }'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Obtenir un serveur de périphérie

Permet d'obtenir un serveur de périphérie. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/vi/service/edge

Méthode

Type de demande GET

Paramètres d'URL

`user`: *chaîne*. Identifie l'utilisateur. Obligatoire.

`solution`: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

`tenant`: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

`hierarchyList`: Informations sur la hiérarchie des cellules, par exemple `plant1`, `line1`, `cell1`. Facultatif.

En-têtes

`APIKEY`: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

Aucun.

Réponse indiquant la réussite

```
200
[{"port": "22", "id": "1512098793912", "clusterType": "slave", "hierarchyList":
["plant1", "line1", "cell1"], "createdBy": "auto_mm", "description": "test edge",
"name": "edge_auto", "updatedAt": "2017-12-01T10:58:26.559+08:00", "online":
"false", "ip": "9.112.229.54"}]
```

Éléments de réponse

`name`: *chaîne*. Nom du serveur de périphérie.

`ip`: *IP*. Adresse IP du serveur de périphérie.

`port`: *entier*. Port du serveur de périphérie.

`clusterType`: *chaîne*. Type de cluster du serveur de périphérie. Spécifiez `slave` ou `master`.

`online`: *chaîne*. Etat de connexion du serveur de périphérie : `true` signifie que le serveur de périphérie est connecté et `false` qu'il est déconnecté.

`hierarchyList`: *chaîne*. Hiérarchie du serveur de périphérie.

`description`: *chaîne*. Description du serveur de périphérie.

`createdBy`: *chaîne*. Propriétaire du serveur de périphérie.

`updatedAt`: *heure*. Heure de la dernière mise à jour du serveur de périphérie.

Exemple d'appel

```
curl -k -H "APIKEY:8b79658225de53488321fb7bb657
f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff
73f886fc3713bf305a158fdf485cc6f275f5" "https://9.112.229.91:9447/ibm/iotm/vi/
service/edge?user=auto_mm&solution=vi"
```

Remarques

Le système utilise les données `APIKEY` pour l'authentification. Si vous ne fournissez pas de données `APIKEY`, le système refusera votre demande.

Supprimer un serveur de périphérie

Permet de supprimer un serveur de périphérie. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

`/ibm/iotm/vi/service/edge/ID_serveur_périphérie`

Méthode

Type de demande DELETE

Paramètres d'URL

`edgeId`: ID du serveur de périphérie. Obligatoire.

`user`: *chaîne*. Identifie l'utilisateur. Obligatoire.

`solution`: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

`tenant`: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

`APIKEY`: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

[{}]

Réponse indiquant la réussite

```
200
[ {} ]
```

Éléments de réponse

Aucun.

Exemple d'appel

```
curl -k -X DELETE -H "APIKEY:8b7965
8225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab
2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a
158fdf485cc6f275f5" "https://9.112.229.91:9447/ibm/iotm/
vi/service/edge/1512098793912?user=auto_mm&solution=
vi" -d '[ {} ]'
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Mettre à niveau un serveur de périphérie

Permet de mettre à niveau un serveur de périphérie. Les utilisateurs qui disposent des droits permettant de créer un serveur de périphérie disposent des droits permettant d'utiliser cette API.

Pour mettre à niveau un serveur de périphérie, vous devez utiliser un appel API afin d'obtenir les fichiers de serveur de périphérie requis, notamment `edgeDeployed.sh` et `edge-vi.zip`.

Pour mettre à niveau un serveur de périphérie sur le système de périphérie, l'utilisateur SSH spécifié lors de la création du serveur de périphérie doit exécuter le script shell sur le système de périphérie.

URL

`/ibm/iotm/vi/service/edgeFile`

Méthode

Type de demande GET

Paramètres d'URL

`edgeID`: ID du serveur de périphérie. Obligatoire.

`user`: *chaîne*. Identifie l'utilisateur. Obligatoire.

`solution`: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

`tenant`: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

Version: *chaîne*. La valeur peut être shell pour le téléchargement d'un script shell, ou ubuntu, redhat ou power pour le type de système Linux sur le système de périphérie.

En-têtes

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Réponse indiquant la réussite

```
200
```

Fichier edgeDeployed.sh ou vi_edge-bin_vi.zip.

Éléments de réponse

Aucun

Exemple d'appel

```
curl -k -X GET -H "APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" "https://int_iotm.predictivesolutionsapps.ibmcloud.com/ibm/iotm/vi/service/edgeFile?tenant=Q3T1&solution=vi&user=modelmanager1&version=shell&edgeId=1508476898000"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Service d'évaluation

Le service d'évaluation permet d'évaluer une image.

Évaluer une image

Permet d'évaluer une image. L'inspecteur et le superviseur disposent des droits permettant d'utiliser cette API.

URL

/ibm/iotm/vi/service/uploadScoreImage

Méthode

Type de demande POST

Paramètres d'URL

productType: *chaîne*. Type de produit du modèle. Obligatoire.

cell: *chaîne*. Cellule de l'image. Obligatoire.

user: *chaîne*. Identifie l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

tenant: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

En-têtes

Content-Type: application/binary

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

Contenu binaire du fichier image.

Réponse indiquant la réussite

```
200
//pour l'exemple de détection d'objet :
{
  "detections": [
    {
      "position": {
        "height": 60,
        "width": 253,
        "x": 1594,
        "y": 773
      },
      "probableTypes": [
        {
          "confidence": 99.798703193664551,
          "type": "scratch"
        }
      ],
      "properties": []
    }
  ],
  "timestamp": "2017-09-28 10:32:13.528415"
}

//pour l'exemple de classification :
{
  "Timestamp": "2017-09-12 18:43:52",
  "detections": [
    {
      "position": {
        "height": 436,
        "width": 537,
        "x": 1574,
        "y": 1588
      },
      "probableTypes": [
        {
          "confidence": 100.0,
          "type": "Defect"
        },
        {
          "confidence": 0.0,
          "type": "MisT"
        },
        {
          "confidence": 0.0,
          "type": "NoDefect"
        }
      ]
    }
  ]
}
```

Éléments de réponse

position: *tableau* JSON. Hauteur, largeur et coordonnées x et x de l'inspection.

probableTypes: *objet* JSON. Type et niveau de fiabilité de chaque défaut.

Timestamp: *heure*. Heure de l'évaluation de l'image.

Exemple d'appel

```
curl -k -X POST -H "APIKEY:8b79658225de53488321
fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb9593
17ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H
"Content-Type: application/binary" "https://9.112.229.91:9447/
ibm/iotm/vi/service/uploadScoreImage?user=auto_super&productType=
auto&cell=plant1_line1_cell1&solution=vi" --connect-timeout 600
--data-binary @C:\code\Automation_98\API\vi\VI_API\file\test.JPG
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Service d'intégration QEWS

Le service d'intégration QEWS permet d'obtenir un fichier de taux d'images de défaut.

Obtenir un fichier de taux d'images de défaut

Permet d'obtenir un fichier contenant des données sur le taux d'images de défaut. Le superviseur dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/service/defectImageRateFile

Méthode

Type de demande GET

Paramètres d'URL

`user`: *chaîne*. Identifie l'utilisateur. Obligatoire.

`solution`: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

`tenant`: *chaîne*. Utilisé pour identifier le titulaire. Facultatif.

`date from`. Date de début au format aaaa-MM-jj ou aaaa-MM-jj HH:mm:ss. Facultatif.

`date to`. Date de fin au format aaaa-MM-jj ou aaaa-MM-jj HH:mm:ss. Facultatif.

`cell name`. Nom de la cellule. Facultatif.

`interval`. Intervalle du taux d'images de défaut. Spécifiez `daily`, `hourly` ou `monthly`. Facultatif.

En-têtes

`APIKEY`: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

Aucun.

Réponse indiquant la réussite

200. Contenu binaire du fichier de données. Le fichier de données est un fichier au format CSV.

Éléments de réponse

`ATTRIBUTE_NAME:VI_DEFECT_IMAGE_RATE`, valeur fixe.

`DATE`: date au format aaaa-MM-jj lorsque l'intervalle est `daily` (tous les jours) ou `monthly` (tous les mois) et au format aaaa-MM-jj HH:mm:ss lorsque l'intervalle est `hourly` (toutes les heures).

`CELL_NAME`: nom de la cellule. Dans cet élément de réponse, la virgule (,) est remplacée par un trait de soulignement (_) car les virgules ne sont pas prises en charge.

Exemples d'appel

```
curl -k -H "APIKEY:8b79658225de53488
321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533
dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.89:9447/ibm/iotm/service/
defectImageRateFile?user=supervisor1&solution=
vi&tenant=T1" --output "C:\Users\IBM_ADMIN\Downloads\a.csv"
```

```
curl -k -H "APIKEY:8b79658225de53488
321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533
dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5"
"https://9.112.229.89:9447/ibm/iotm/service/
defectImageRateFile?user=supervisor1&solution=
vi&tenant=T1&dateFrom=2017-10-23%2011:00:00&dateTo=
2017-11-14%2018:40:00&cellName=plant1,line1,cell1&
```

```
interval=hourly" --output "C:\Users\IBM_ADMIN\
Downloads\cell11_hourly.csv"
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Service composite

Le service composite permet d'obtenir un fichier compressé contenant un modèle entraîné et d'enregistrer le modèle dans Maximo PQI SaaS Visual Insights.

Enregistrer un modèle

Permet d'obtenir un fichier compressé contenant un modèle entraîné et d'enregistrer le modèle dans Maximo PQI SaaS Visual Insights. Le gestionnaire de modèles dispose des droits permettant d'utiliser cette API.

URL

/ibm/iotm/vi/service/registerModel

Méthode

Type de demande POST

Paramètres d'URL

user: *chaîne*. Utilisé pour identifier l'utilisateur. Obligatoire.

solution: *chaîne*. Utilisé pour identifier la solution. Obligatoire.

name: *chaîne*. Nom du modèle.

productType: *chaîne*. Type de produit associé au modèle.

En-têtes

Content-Type: multipart/form-data.

Content-Disposition: form-data; name="files[]"; filename="model.zip".

APIKEY: *clé chiffrée*. Clé d'API utilisée pour l'authentification. Obligatoire.

Paramètres de données

Aucun

Exemple de corps

Contenu binaire du fichier de modèle.

Réponse indiquant la réussite

```
200
{"errorMessage":"","modelId":"3914ad2c-3b7b-47ad-974c-e5b6bc2075ef",
"createdBy":"modelmanager1","status":"validated","instanceId":
"3914ad2c-3b7b-47ad-974c-e5b6bc2075ef_1517796906688",
"updatedAt":"2018-02-05 10:15:06.608"}
```

Éléments de réponse

modelId: *chaîne*. ID du modèle.

createdBy: *chaîne*. Propriétaire du modèle.

status: *chaîne*. Statut de l'instance de modèle.

instanceId: *chaîne*. ID de l'instance de modèle.

updatedAt: *heure*. Heure de la dernière mise à jour de l'instance de modèle.

Exemple d'appel

```
curl -k -X POST -H
"APIKEY:8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77
d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5" -H
"Content-Type:multipart/form-data" -H "Content-Disposition: form-data;
name="files[]"; filename="walkermodell.zip"
"https://int_iotm.predictivesolutionsapps.ibmcloud.com/ibm/iotm/vi/
service/registerModel?user=demoadm@cn.ibm.com&solution=vi&name=
samplename&productType=sampletype" --connect-timeout 6000 -F
file=@walkermodell.zip
```

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Services de serveurs de périphérie autonomes

Les services de serveurs de périphérie autonomes vous permettent d'effectuer des tâches telles que l'obtention des modèles disponibles, le déploiement de modèles, le transfert d'évaluations d'image, la synchronisation des résultats d'inspection, et le nettoyage des résultats d'inspection.

Les services de serveurs de périphérie autonomes s'exécutent sur la machine de périphérie autonome et non dans le centre Maximo PQI SaaS Visual Insights. Vous devez utiliser des hôtes, des ports et des données d'identification différents.

Obtenir les modèles disponibles

Permet d'obtenir tous les modèles disponibles. L'administrateur du serveur de périphérie autonome peut utiliser cette API.

URL

https://<hôte_machine_périphérie>:8449/api/getAvailableModels

Méthode

Type de demande GET

Paramètres d'URL

Aucun.

En-têtes

Authorization: *nom_utilisateur/mot_de_passe*

Paramètres de données

Aucun.

Exemple de corps

Aucun.

Réponse indiquant la réussite

```
{ "model_list": [
  {
    "coviacenter": {
      "apikey": "8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d
4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5",
      "credential": "ZGVtb2FkbUBjbi5pYm0uY29tO1EzVDE=",
      "url": "https://int_iotm.predictivesolutionsapps.ibmcloud.com/ibm/iotm/
service/apiWrapper?apiName=addInspectResult",
      "username": "demoadm@cn.ibm.com"
    },
    "groupIds": "0c4f11d7-b255-4c05-9c87-694e9f912fa9",
    "model_id": "049dabe5-d865-4f09-862f-fb7430afef6d",
    "model_instance_id": "049dabe5-d865-4f09-862f-fb7430afef6d_1518401501376",
    "model_type": "classification",
    "model_url": "049dabe5-d865-4f09-862f-fb7430afef6d_1518401599389.zip",
```

```
    "product_type": "Q3T1|jianyuan45",  
    "status": "deployed"  
  },  
  ...  
]}
```

Éléments de réponse

model_list: *chaîne*. Liste des modèles publiés.

model_id: *chaîne*. ID du modèle publié.

model_instance_id: *chaîne*. ID d'instance du modèle publié.

model_type: *chaîne*. Type du modèle publié.

model_url: *chaîne*. URL du modèle publié.

product_type: *chaîne*. Type de produit du modèle publié.

coviacenter: *objet JSON*. Informations de centre requises pour le serveur de périphérie.

Exemple d'appel

```
curl -k -H 'Authorization: Basic YWRtaW46cGFzc3cwcmRA' -X GET https://{localhost}  
:8449/api/getAvailableModel
```

Déployer un modèle

Déploye un modèle. L'administrateur du serveur de périphérie autonome peut utiliser cette API.

URL

https://<hôte_machine_périphérie>:8449/api/deployModel

Méthode

Type de demande POST

Paramètres d'URL

Aucun.

En-têtes

Authorization: *nom_utilisateur/mot_de_passe*

Paramètres de données

model_list: *chaîne*. Liste des modèles publiés.

model_id: *chaîne*. ID du modèle publié.

model_instance_id: *chaîne*. ID d'instance du modèle publié.

model_type: *chaîne*. Type du modèle publié.

model_url: *chaîne*. URL du modèle publié.

product_type: *chaîne*. Type de produit du modèle publié.

coviacenter: *objet JSON*. Informations de centre requises pour le serveur de périphérie.

edgeId: *chaîne*. ID du serveur de périphérie en cours.

Exemple de corps

```
{  
  "model_id": "56608a3d-df64-4ef0-85a6-f88778cf583b",  
  "model_url": "56608a3d-df64-4ef0-85a6-f88778cf583b_1540279487176.zip",  
  "edgeId": "1540275842203",  
  "coviacenter": {  
    "url": "https://int_iotm.predictivesolutionsapps.ibmcloud.com/  
ibm/iotm/service/apiWrapper?apiName=addInspectResult",  
    "credential": "c3VwZXJ2aXNvcjE6UTNUMQ==",  
    "username": "supervisor1",  
    "apikey": "8b79658225de53488321fb7bb657f9f161acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34ggb959317ac69ff73f886fc3713bf305a158fdf485cc6f275f5",  
    "groupIds": "b18a8d56-7a4c-4266-9280-49258c3fd798",  
    "model_type": "classification",  
  }  
}
```

```
"product_type": "Q3T1|qwtest12", "model_instance_id": "56608a3d-df64-4ef0-85a6-f88778cf583b_1540279416752", "model_name": "qwtest12"}
```

Réponse indiquant la réussite

```
{ "status": "success" }
```

Éléments de réponse

Status: *chaîne*. Réussite ou échec.

Exemple d'appel

```
curl -k -H 'Authorization: Basic YWRtaW46cGFzc3cwcmRA'  
-H 'Content-Type:application/json;charset=UTF-8' -X POST --data  
'{"model_id": "56608a3d-df64-4ef0-85a6-f88778cf583b", "model_url":  
"56608a3d-df64-4ef0-85a6-f88778cf583b_1540279487176.zip", "edgeId":  
"1540275842203", "coviacenter": { "url": "https://int_iotm.  
predictivesolutionsapps.ibmcloud.com/ibm/iotm/service/apiWrapper?  
apiName=addInspectResult", "credential": "c3VwZXJ2aXNvcjE6UTNUMQ==",  
"username": "supervisor1", "apikey": "8b79658225de53488321fb7bb657f9f161  
acd2ea830a5afb8c149f6aab2c77d4f0579e533dd34gg959317ac  
69ff73f886fc3713bf305a158fdf485cc6f275f5" }, "groupIds": "b18a8d56-7a4c-  
4266-9280-49258c3fd798", "model_type": "classification", "product type":  
"Q3T1|qwtest12", "model_instance_id": "56608a3d-df64-4ef0-85a6-f88778cf5  
83b_1540279416752", "model_name": "qwtest12"}' https://{localhost}:  
8449/api/deployModel
```

Annuler le déploiement du modèle

Permet d'annuler le déploiement d'un modèle. L'administrateur du serveur de périphérie autonome peut utiliser cette API.

URL

https://<hôte_machine_périphérie>:8449/api/undeployModel

Méthode

Type de demande POST

Paramètres d'URL

Aucun.

En-têtes

Authorization: *nom_utilisateur/mot_de_passe*

Paramètres de données

model_id: *chaîne*. ID du modèle dont le déploiement est annulé.

model_instance_id: *chaîne*. ID d'instance du modèle dont le déploiement est annulé.

Exemple de corps

```
{ "model_id": "5868dcd7-7032-4101-9ba8-797dea9005aa", "model_instance_id": "5868dcd7-  
7032-4101-9ba8-797dea9005aa_1523253620387" }
```

Réponse indiquant la réussite

```
{ "status": "success" }
```

Éléments de réponse

Status: *chaîne*. Réussite ou échec.

Exemple d'appel

```
curl -k -H 'Authorization: Basic YWRtaW46cGFzc3cwcmRA'  
-H 'Content-Type:application/json;charset=UTF-8' -X POST --data  
'{"model_id": "8e474fb5-97dd-4f5d-8415-14c72855c6a5", "model_instance_id":  
"8e474fb5-97dd-4f5d-8415-14c72855c6a5_1532404443311"}' https://{local  
host}:8449/api/undeployModel
```

Transférer et évaluer une image sur le serveur de périphérie

Transfère et évalue une image sur le serveur de périphérie. La ligne de production ou des services externes utilisent généralement cette API.

URL

/api/uploadScoreImage

Méthode

Type de demande POST

Paramètres d'URL

productType: *chaîne*. Type de produit du modèle. Obligatoire.

cell: *chaîne*. Cellule de l'image. Obligatoire.

En-têtes

Content-Type: application/binary.

Authorization: . Code d'autorisation de base pour le service de serveur de périphérie. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

Contenu binaire du fichier image.

Réponse indiquant la réussite

Exemple de détection d'objet :

```
200
{
  "detections": [
    {
      "position": {
        "height": 60,
        "width": 253,
        "x": 1594,
        "y": 773
      },
      "probableTypes": [
        {
          "confidence": 99.798703193664551,
          "type": "scratch"
        }
      ],
      "properties": []
    }
  ],
  "timestamp": "2017-09-28 10:32:13.528415"
}
```

Exemple de classification :

```
200
{
  "Timestamp": "2017-09-12 18:43:52",
  "detections": [
    {
      "position": {
        "height": 436,
        "width": 537,
        "x": 1574,
        "y": 1588
      },
      "probableTypes": [
        {
          "confidence": 100.0,
          "type": "Defect"
        }
      ],
    }
  ]
}
```

```

    "confidence": 0.0,
    "type": "MisT"
  },
  "confidence": 0.0,
  "type": "NoDefect"
}
]
}
]
}

```

Exemple d'appel

```

curl -k --cert /home/pmqopsadmin/vi_edge-bin_vi/vi_edge/https/cert.pem --key
/home/pmqopsadmin/vi_edge-bin_vi/vi_edge/https/key.pem -H 'Authorization:
Basic autorisation' -T 'chemin_fichier' -X POST
https://localhost
:8449/api/uploadScoreImage?productType=type_produit&cell=cellule

```

La valeur *autorisation* est l'autorisation de base permettant d'accéder au service de serveur de périphérie. La valeur *chemin_fichier* est le chemin d'accès au fichier à transférer. La valeur *localhost* est l'adresse IP du système sur lequel le serveur de périphérie est déployé. La valeur *type_produit* est le type de produit du modèle que vous avez utilisé pour évaluer l'image que vous avez transférée. La valeur *cellule* est la cellule dans laquelle envoyer le résultat de l'évaluation.

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Synchroniser le résultat de l'inspection du serveur de périphérie dans l'application centrale

Synchronise le résultat de l'inspection du serveur de périphérie dans l'application centrale. L'administrateur du serveur de périphérie autonome peut utiliser cette API.

URL

`/api/syncInspectResult`

Méthode

Type de demande POST

Paramètres d'URL

Aucun

En-têtes

Content-Type: application/json.

Authorization: . Code d'autorisation de base pour le service de serveur de périphérie. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

```

{
  "percentage":100
}

```

Réponse indiquant la réussite

```

200
{
  "message": "success"
}

```

Exemple d'appel

```
curl -k --cert /home/pmqopsadmin/vi_edge-bin_vi/vi_edge/https/cert.pem --key /home/pmqopsadmin/vi_edge-bin_vi/vi_edge/https/key.pem -H "Content-Type: application/json" -H 'Authorization: Basic YWRtaW46cGFzc3cwcmRA' -X POST --data '{ "percentage": pourcentage }' https://localhost:8449/api/syncInspectResult
```

La valeur *pourcentage* correspond au pourcentage des résultats d'inspection à envoyer. La valeur *localhost* est l'adresse IP à laquelle le serveur de périphérie est déployé.

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Nettoyer un résultat d'inspection que vous avez synchronisé avec l'application centrale

Nettoie un résultat d'inspection que vous avez synchronisé avec l'application centrale. L'administrateur du serveur de périphérie autonome peut utiliser cette API.

URL

/api/cleanInspectResult

Méthode

Type de demande POST

Paramètres d'URL

Aucun

En-têtes

Content-Type: application/json.

Authorization: . Code d'autorisation de base pour le service de serveur de périphérie. Obligatoire.

Paramètres de données

Aucun.

Exemple de corps

```
{
  "fromDayBefore": 1
}
```

Réponse indiquant la réussite

```
{
  "message": "success"
}
```

Exemple d'appel

```
curl -k --cert /home/pmqopsadmin/vi_edge-bin_vi/vi_edge/https/cert.pem --key /home/pmqopsadmin/vi_edge-bin_vi/vi_edge/https/key.pem -H "Content-Type: application/json" -H 'Authorization: Basic YWRtaW46cGFzc3cwcmRA' -X POST --data '{ "fromDayBefore": nombre_jours }' https://localhost:8449/api/cleanInspectResult
```

La valeur *nombre_jours* correspond au nombre de jours pour lesquels conserver les données. Par exemple, si vous définissez la valeur 0 pour *nombre_jours*, toutes les données antérieures à aujourd'hui sont supprimées. Si vous définissez la valeur 1 pour *nombre_jours*, toutes les données antérieures à hier sont supprimées. La valeur *localhost* est l'adresse IP à laquelle le serveur de périphérie est déployé.

Remarques

Le système utilise les données APIKEY pour l'authentification. Si vous ne fournissez pas de données APIKEY, le système refusera votre demande.

Chapitre 8. Enregistrement, déploiement et test d'un modèle à l'aide de l'API

La fonction Conseils pour les API est une interface Web qui vous guide tout au long d'un scénario d'utilisation d'API commun : le processus d'enregistrement, de déploiement et de test d'un modèle entraîné. Elle automatise de nombreuses étapes du processus pour simplifier l'exécution des tâches.

Pourquoi et quand exécuter cette tâche

La première étape consiste à enregistrer un modèle. Vous pouvez importer un fichier compressé contenant un modèle entraîné et l'enregistrer dans le produit.

La deuxième étape consiste à déployer le modèle. Vous pouvez déployer un modèle sur des systèmes de périphérie pour qu'il puisse être utilisé afin d'inspecter des images.

La troisième étape consiste à tester le modèle. Vous pouvez utiliser vos propres images pour tester le modèle déployé et prendre connaissance du résultat d'évaluation.

Procédure

1. Sélectionnez **Conseils pour les API** dans le menu déroulant.
2. Cliquez sur **Essayer d'enregistrer le modèle** pour ouvrir la page **Conseils complets pour les API**.
3. Entrez le nom de modèle et le type de produit.
4. Cliquez sur **Sélectionner** pour sélectionner un fichier compressé contenant un modèle entraîné.
5. Cliquez sur **Enregistrer** pour enregistrer le modèle dans le produit.
Le résultat de l'enregistrement est affiché dans la zone **Résultat**.
6. Cliquez sur **Suivant**.
7. Si nécessaire, entrez l'ID d'instance.
8. Cliquez sur **Déployer** pour déployer le modèle.
Le résultat du déploiement est affiché dans la zone **Résultat**.
9. Cliquez sur **Suivant**.
10. Entrez le type de produit et les informations de cellule, sélectionnez une image, puis cliquez sur **Evaluer** pour démarrer le processus d'évaluation.
Le résultat de l'évaluation est affiché dans la zone **Résultat**.

Chapitre 9. Traitement des incidents

Consultez les fichiers journaux et les messages d'erreur car ils peuvent vous aider à traiter les incidents liés au produit.

Fichiers journaux

Consultez les fichiers journaux du produit pour plus d'informations sur les cas d'erreur.

Fichier journal	Description
<code>/chemin_Liberty/usr/servers/ VICenterServer/logs/messages.log</code>	Messages WebSphere Application Server Liberty
<code>/home/nom_utilisateur/ edgeDeploy_nom_titulaire.log</code>	Messages du serveur de périphérie
<code>/home/nom_utilisateur/vi_edge-bin_vi/ vi_task_manager/master.log</code>	Messages du gestionnaire de tâches
<code>/home/nom_utilisateur/vi_edge-bin_vi/ vi_edge/Service.log</code>	Messages du service

Messages

Messages du serveur de périphérie

Les messages ci-après apparaissent lors de cas d'erreur pour les systèmes de périphérie.

Message	Action
Un serveur de périphérie maître existe déjà ; un seul serveur de périphérie maître est pris en charge.	Spécifiez un nom de serveur de périphérie qui n'est pas utilisé.
Les zones requises ne doivent pas être vides.	Indiquez une valeur dans toutes les zones requises.
Connexion à la machine impossible. Utilisez l'adresse IP, le nom d'utilisateur SSH et le mot de passe corrects.	Vérifiez que l'adresse IP, le nom d'utilisateur SSH et le mot de passe sont des valeurs valides.
Pour pouvoir gérer le serveur de périphérie, vous devez disposer des droits appropriés.	Assurez-vous d'être connecté en tant qu'utilisateur disposant des droits de gestionnaire de modèles. Si vous ne parvenez pas à résoudre le problème, contactez un administrateur système.
L'adresse IP n'est pas valide. Entrez l'adresse IP correcte, par exemple 10.172.0.23.	Vérifiez la valeur de l'adresse IP.
Le serveur de périphérie n'a pas été créé en raison d'une erreur de base de données.	Vérifiez le service Apache HBase dans la console Ambari.
Un serveur de périphérie associé à la même adresse IP existe.	Vous ne pouvez pas créer deux systèmes de périphérie sur le même ordinateur. Utilisez l'adresse IP d'un autre ordinateur pour créer un autre serveur de périphérie.

Tableau 4. Messages d'erreur pour les systèmes de périphérie (suite)

Message	Action
Echec du transfert des scripts sur la machine de périphérie. Vérifiez l'existence du chemin de déploiement et les droits associés.	Vérifiez que le chemin de déploiement existe et qu'il est associé aux droits d'accès appropriés.
Echec du démarrage des services sur la machine de périphérie.	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez le fichier journal sur le système de périphérie.
Echec de l'exécution des scripts sur la machine de périphérie.	Consultez le fichier journal du système de périphérie.
Le nom du serveur de périphérie est un doublon. Utilisez un autre nom de serveur de périphérie.	Utilisez un autre nom pour créer un nouveau serveur de périphérie.
Le serveur de périphérie esclave ne peut pas être créé s'il n'existe pas de serveur de périphérie maître.	Vous devez créer un serveur de périphérie maître avant de créer un serveur de périphérie esclave.
Echec de la suppression du serveur de périphérie car il est utilisé par le modèle <i>nom_modèle</i> .	Le serveur de périphérie ne peut pas être supprimé s'il est utilisé par une version de modèle. Arrêtez toutes les versions de modèle en cours d'exécution sur le serveur de périphérie avant de supprimer le serveur de périphérie.
Le serveur de périphérie n'a pas été supprimé en raison d'une erreur de base de données.	Vérifiez le service Apache HBase dans la console Ambari.
Vous ne pouvez pas supprimer le serveur de périphérie maître car il existe des serveurs de périphérie esclaves.	Le serveur de périphérie ne peut pas être supprimé s'il possède des serveurs de périphérie esclave. Supprimez tous les serveurs de périphérie esclave avant de supprimer le serveur de périphérie maître.
Echec de la suppression du serveur de périphérie depuis le référentiel principal : <i>nom_référentiel</i> .	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez les fichiers journaux du service.

Messages pour les fichiers image compressés

Les messages ci-après apparaissent lors de cas d'erreur pour les fichiers image compressés.

Tableau 5. Messages d'erreur pour les fichiers image compressés

Message	Action
Le fichier compressé que vous avez transféré ne contient pas d'image à l'emplacement prévu.	Le fichier image compressé transféré doit contenir des images. Toutes les images dans le fichier image compressé doivent se trouver dans une structure à plat ne comportant pas de sous-répertoire.
Le fichier n'est pas valide. Vous devez transférer des fichiers zip uniquement.	Le seul format de fichier pris en charge est .zip.
Les fichiers XML d'annotation manquent dans le fichier compressé transféré.	Ajoutez les fichiers XML d'annotations au fichier image compressé, puis transférez à nouveau le fichier image compressé.

Tableau 5. Messages d'erreur pour les fichiers image compressés (suite)

Message	Action
Le fichier labels.txt manque dans le fichier compressé transféré.	Ajoutez le fichier labels.txt au fichier image compressé, puis transférez à nouveau le fichier image compressé.
L'entrée n'est pas valide. Vous devez spécifier l'ID.	Aucun ID de fichier n'a été fourni dans la demande d'API. Indiquez l'ID de fichier.
L'entrée n'est pas valide. Vous ne pouvez pas supprimer plus de 100 entités.	Vous ne pouvez pas supprimer plus de 100 fichiers compressés à la fois.
Echec de la suppression du fichier de données.	Le fichier de données n'a pas pu être supprimé.
Le fichier de données spécifié a été cité dans le modèle <i>nom_modèle</i> . Il ne peut pas être supprimé.	Le fichier image compressé est référencé par un modèle. Il ne peut pas être supprimé.

Messages pour les groupes d'images

Les messages ci-après apparaissent lors de cas d'erreur pour les groupes d'images.

Tableau 6. Messages d'erreur pour les groupes d'images

Message	Action
Le nom de groupe ne peut pas être vide lorsque vous créez un groupe de données.	Spécifiez un nom pour le nouveau groupe.
Des fichiers de données existent pour le groupe de données sélectionné. La mise à jour a échoué. Assurez-vous que le groupe de données sélectionné ne contient pas de fichier de données.	Si le groupe d'images est créé avec des fichiers de données, le type de groupe d'images ne peut pas être mis à jour. Si le groupe d'images n'est pas créé avec des fichiers de données, une erreur d'opération Apache HBase peut être à l'origine de cette erreur.
Echec de la création d'un groupe de données car des paramètres manquent ou leur format n'est pas valide.	Assurez-vous que des paramètres sont spécifiés pour le nouveau groupe de données et que leur format est valide.
Le nom de groupe est un doublon. Utilisez un autre nom de groupe.	Spécifiez un nom unique pour le nouveau groupe.
Erreur lors de la création du groupe de données.	Une erreur Apache HBase est survenue lors de la création du groupe de données. Vérifiez le service Apache HBase dans la console Ambari.
Le groupe de données n'existe pas.	L'ID de groupe dans la demande manque ou n'est pas valide.
Erreur lors de la suppression de fichiers de données dans le groupe de données spécifié.	Une erreur Hadoop Distributed File System (HDFS) est survenue lors de la suppression de fichiers de données.
Le groupe de données spécifié a été cité dans un modèle. Il ne peut pas être supprimé.	Si un groupe est utilisé dans un modèle, il ne peut pas être supprimé.
Erreur lors de la suppression du groupe de données.	Une erreur Apache HBase est survenue lors de la suppression du groupe de données. Vérifiez le service Apache HBase dans la console Ambari.

Messages pour les modèles

Les messages ci-après apparaissent lors de cas d'erreur pour les modèles.

<i>Tableau 7. Messages d'erreur pour les modèles</i>	
Message	Action
Le nom de modèle ne peut pas être vide lorsque vous créez un modèle.	Spécifiez un nom pour le modèle.
Le nom de modèle ne doit pas comporter plus de 128 caractères et ne peut contenir que des lettres, des chiffres, des espaces et des traits de soulignement.	Spécifiez un nom de modèle valide.
Le type de produit ne peut pas être vide lorsque vous créez un modèle.	Spécifiez le type de produit.
Le type de produit ne doit pas comporter plus de 128 caractères et ne peut contenir que des lettres, des chiffres et des espaces.	Spécifiez un type de produit valide.
Les ID de groupe de données ne peuvent pas être vides lorsque vous créez un modèle.	Spécifiez une valeur pour <code>groupIds</code> dans l'appel API.
Le format de données ne peut pas être vide lorsque vous créez un modèle.	Spécifiez une valeur pour <code>dataFormat</code> dans l'appel API.
Le format de la règle de nouvel entraînement doit être valide lorsque vous créez un modèle.	Spécifiez une valeur valide pour <code>retrainPolicy</code> dans l'appel API.
Le type de modèle ne peut pas être vide lorsque vous créez un modèle.	Spécifiez une valeur pour <code>modelType</code> dans l'appel API.
Le type de modèle doit être <code>classification</code> ou <code>objectdetection</code> .	Spécifiez la valeur <code>classification</code> ou <code>objectdetection</code> pour <code>modelType</code> dans l'appel API.
Le format des paramètres doit être valide lorsque vous créez un modèle.	Spécifiez une valeur valide pour <code>parameters</code> dans l'appel API.
Echec de la création du modèle car la valeur du paramètre <code>trainParam</code> dans les paramètres n'est pas valide.	Spécifiez une valeur valide pour <code>trainParam</code> dans l'appel API.
Les ID des groupes de données ne sont pas valides.	Spécifiez une valeur valide pour <code>groupIds</code> dans l'appel API.
Une erreur est survenue car les groupes de données doivent tous être de type "Caractéristiques multiples" ou tous être de type "Caractéristique unique".	Dans l'appel API, les groupes dans <code>groupIds</code> doivent posséder le même type de groupe.
Echec de la création du modèle.	Une erreur Apache HBase est survenue lors de la création du modèle. Vérifiez le service Apache HBase dans la console Ambari.
L'entrée n'est pas valide. Vous ne pouvez pas supprimer plus de 100 entités.	Spécifiez moins de 100 modèles à supprimer.
L'entrée n'est pas valide. Vous devez spécifier l'ID.	Indiquez l'ID de modèle dans la demande.
Le modèle ne peut pas être mis à jour car l'ID de modèle est vide. Entrez un ID de modèle.	Indiquez l'ID de modèle dans la demande.

Tableau 7. Messages d'erreur pour les modèles (suite)

Message	Action
Echec de la mise à jour des groupes de modèles car la version de modèle n'a pas le statut de brouillon.	Assurez-vous que les modèles à mettre à jour sont associés au statut de brouillon.
Echec de la mise à jour des groupes de modèle car la valeur isHybrid des groupes de données est différente.	Vous ne pouvez pas changer le type de modèle une fois que le modèle a été créé.

Messages pour les instances de modèle

Les messages ci-après apparaissent lors de cas d'erreur pour les instances de modèle.

Tableau 8. Messages d'erreur pour les instances de modèle

Message	Action
L'ID de modèle ne peut pas être vide lorsque vous créez une instance de modèle.	Créez une instance de modèle une fois le modèle correspondant créé.
Les données d'entraînement ne peuvent pas être vides lorsque vous créez une instance de modèle.	Spécifiez une valeur valide pour <code>trainData</code> dans l'appel API.
Echec de l'enregistrement du modèle sur le serveur d'entraînement.	Vérifiez le service qui est démarré sur le serveur d'entraînement.
Echec de la création du modèle car la valeur du paramètre <code>trainParam</code> dans les paramètres n'est pas valide.	Spécifiez une valeur valide pour <code>trainParam</code> dans l'appel API.
Les données d'apprentissage ne sont pas valides. Un groupe de données au moins doit exister.	Assurez-vous que chaque groupe de données appartenant à l'instance possède des fichiers de données valides.
Les données d'apprentissage ne sont pas valides. Vérifiez le rapport d'entraînement et de validation pour vous assurer qu'au moins une image existe dans l'ensemble d'entraînement et de tests pour chaque groupe de données.	Assurez-vous qu'au moins une image existe dans les ensembles d'entraînement et de tests pour chaque groupe de données.
n modèle(s) se trouve(nt) dans la file d'attente pour l'entraînement. Vous ne pouvez pas soumettre de nouvelle demande d'entraînement tant que le ou les modèles n'ont pas été traités ou annulés.	Annulez les travaux ou attendez que les travaux en cours soient terminés.
Echec de l'annulation de l'entraînement.	Vérifiez le service Apache HBase dans la console Ambari.
Le corps d'entrée n'est pas valide.	Vérifiez votre entrée de demande et relancez l'opération.
Le statut précédent du modèle <code>nom_modèle</code> a été restauré suite à un dépassement du délai de validation.	Consultez le journal sur le serveur de validation. Pour le modèle de classification, il s'agit de <code>/home/pmqopsadmin/vi_edge-bin_vi/vi_score_engine_restful/back.log</code> . Pour le modèle de détection d'objet, il s'agit de <code>/home/pmqopsadmin/vi_obj_detection_retrain/RESTAPI/model/frcnn_log.txt</code> ou <code>ssd_log.txt</code> . Résolvez toute erreur, puis relancez l'opération.

Tableau 8. Messages d'erreur pour les instances de modèle (suite)

Message	Action
Echec de la validation de la version de modèle.	Consultez le journal sur le serveur de validation. Pour le modèle de classification, il s'agit du fichier <code>journal /home/pmqopsadmin/vi_edge-bin_vi/vi_score_engine_restful/back.log</code> . Pour le modèle de détection d'objet, il s'agit du fichier <code>journal /home/pmqopsadmin/vi_obj_detection_retrain/RESTAPI/model/frcnn_log.txt</code> ou <code>ssd_log.txt</code> . Résolvez toute erreur, puis relancez l'opération.
Le modèle n'existe pas.	Aucun modèle associé à cet ID de modèle n'a été trouvé. Vérifiez l'ID de modèle.
Echec du rejet de la version de modèle.	Vérifiez le service Apache HBase dans la console Ambari.
Echec du déploiement de la version de modèle car le fichier de modèle est vide.	Une erreur est survenue lors de la création du modèle. Créez un autre modèle.
Aucun serveur de périphérie maître sur lequel déployer le modèle.	Créez un serveur de périphérie maître avant de déployer le modèle.
IP de serveur de périphérie non valide <i>adresse_IP</i> ; le serveur de périphérie maître seulement peut être utilisé pour déployer le modèle.	Utilisez l'adresse IP correcte pour le serveur de périphérie maître afin de déployer un modèle.
Impossible de déployer le modèle sur le serveur de périphérie car le modèle <i>nom_modèle</i> est déployé avec le même type de produit. Vous ne pouvez pas déployer un modèle sur un serveur de périphérie si le même type de produit s'exécute déjà sur le serveur de périphérie. Vous devez d'abord annuler le déploiement de l'autre modèle.	Vous ne pouvez pas déployer un modèle sur un serveur de périphérie si le même type de produit s'exécute déjà sur le serveur de périphérie.
Erreur de base de données lors de la création du fichier de données.	Vérifiez le service Apache HBase dans la console Ambari.
Le fichier <i>nom_fichier</i> n'a pas été transféré dans l'espace de stockage.	Vérifiez le système de fichiers Hadoop Distributed File System.
Echec de la mise à jour du groupe de données en raison d'une erreur de base de données.	Vérifiez le service Apache HBase dans la console Ambari.
Le statut de l'instance de modèle ne prend pas en charge cette action.	Assurez-vous que le statut de l'instance de modèle est approprié, puis réexécutez l'action.
Les données d'apprentissage ne sont pas valides. Un fichier de données au moins doit exister dans chaque groupe de données.	Assurez-vous que chaque groupe de données appartenant à l'instance de modèle possède des fichiers de données valides.
Les données d'apprentissage ne sont pas valides. Vérifiez le rapport d'entraînement et de validation pour vous assurer qu'au moins une image existe dans l'ensemble d'entraînement et de tests pour chaque groupe de données.	Assurez-vous qu'au moins une image existe dans les ensembles d'entraînement et de tests pour chaque groupe de données.
Echec du nouvel entraînement du modèle en raison d'une erreur de base de données.	Vérifiez le service Apache HBase dans la console Ambari.

Tableau 8. Messages d'erreur pour les instances de modèle (suite)

Message	Action
Echec du déploiement de la version de modèle sur le serveur de périphérie <i>nom_serveur_périphérie</i> .	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez le fichier <code>Service.log</code> sur le serveur de périphérie.
Echec de l'annulation de la version de modèle sur le serveur de périphérie <i>nom_serveur_périphérie</i> .	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez le fichier <code>Service.log</code> sur le serveur de périphérie.
Echec de l'annulation du déploiement de la version de modèle en raison d'une erreur de base de données.	Vérifiez le service Apache HBase dans la console Ambari.
Echec de l'annulation de l'entraînement.	Vérifiez le service Apache HBase dans la console Ambari.
La valeur d'instantané n'est pas valide. Impossible d'utiliser l'instantané.	La valeur d'instantané n'existe pas ou n'est pas valide.
Le fichier d'instantané n'existe pas sur le serveur d'entraînement.	Ignorez l'instantané ou entraînez à nouveau le modèle.
L'instance de modèle ne peut pas être supprimée car l'ID d'instance est vide. Entrez un ID d'instance.	Entrez un ID d'instance valide.
Echec de la suppression de l'instance de modèle en raison d'une erreur de base de données.	Vérifiez le service Apache HBase dans la console Ambari.
L'instance de modèle dont le statut est <i>status</i> ne peut pas être supprimée.	Une instance de modèle associée au statut spécifié ne peut pas être supprimée.

Messages pour le guide d'API

Les messages ci-après apparaissent lors de cas d'erreur pour le guide d'API.

Tableau 9. Messages d'erreur pour le guide d'API

Message	Action
Aucun modèle trouvé pour le type de produit spécifié.	Si cette erreur survient suite à un appel API, vérifiez que vous avez envoyé le type de produit approprié. Si elle se produit dans l'interface utilisateur, essayez un autre type de produit.
Aucune information relative à l'emplacement de déploiement du modèle.	Vérifiez que le modèle correspondant est déployé.
Echec du transfert de l'image sur le serveur de périphérie <i>nom_serveur_périphérie</i> .	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez le fichier <code>Service.log</code> sur le serveur de périphérie.
Echec de l'obtention du résultat de l'évaluation de l'image depuis le serveur de périphérie <i>nom_serveur_périphérie</i> .	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez le fichier <code>Service.log</code> sur le serveur de périphérie.
Le délai d'attente a été dépassé. Impossible d'obtenir le résultat de l'évaluation de l'image depuis le serveur de périphérie <i>nom_serveur_périphérie</i> .	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Consultez le fichier <code>Service.log</code> sur le serveur de périphérie.

Tableau 9. Messages d'erreur pour le guide d'API (suite)

Message	Action
Le nom de modèle ne peut pas être vide lorsque vous créez un modèle.	Spécifiez un nom pour le modèle.
Le nom de modèle ne doit pas comporter plus de 128 caractères et ne peut contenir que des lettres, des chiffres, des espaces et des traits de soulignement.	Spécifiez un nom de modèle valide.
Le type de produit ne peut pas être vide lorsque vous créez un modèle.	Spécifiez le type de produit.
Le type de produit ne doit pas comporter plus de 128 caractères et ne peut contenir que des lettres, des chiffres et des espaces.	Spécifiez un type de produit valide.
Le nom de modèle est un doublon. Utilisez un autre nom de modèle.	Spécifiez un nom unique pour le modèle.
Le fichier <i>nom_fichier</i> n'a pas été transféré dans l'espace de stockage.	Vérifiez le système de fichiers Hadoop Distributed File System.
Echec du transfert du fichier de modèle.	Vérifiez votre connexion réseau et relancez le transfert.

Messages pour le simulateur

Les messages ci-après apparaissent lors de cas d'erreur pour le simulateur.

Tableau 10. Messages d'erreur pour le simulateur

Message	Action
Echec de l'envoi des images simulées.	Assurez-vous que le service est démarré sur le serveur de périphérie maître. Vérifiez le fichier <i>Service.log</i> .

Remarques

Le présent document a été développé pour des produits et des services proposés aux Etats-Unis et peut être mis à disposition par IBM dans d'autres langues. Toutefois, il peut être nécessaire de posséder une copie du produit ou de la version du produit dans cette langue pour pouvoir y accéder.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails sur les produits et services actuellement disponibles dans votre pays, adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, un programme ou un service IBM n'est pas conçue pour stipuler ou impliquer que seul le produit, le programme ou le service IBM peut être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
U.S.A.*

Pour le Canada, veuillez adresser votre courrier à :

*IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada*

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse IBM suivante :

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT". IBM DECLINE TOUTE RESPONSABILITE, EXPLICITE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE NON-CONTREFAÇON ET D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des erreurs typographiques. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
U.S.A.*

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA (IBM Customer Agreement), des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performances et les exemples de clients ne sont présentés qu'à des fins d'illustration. Les performances réelles peuvent varier en fonction des configurations et des conditions d'exploitation spécifiques.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Ces informations sont fournies uniquement à titre de planification. Elles sont susceptibles d'être modifiées avant la mise à disposition des produits décrits.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

LICENCE DE COPYRIGHT :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. IBM ne peut donc pas garantir ni impliquer la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les exemples de programme sont fournis en l'état, sans garantie d'aucune sorte. IBM ne saurait être tenu pour responsable des éventuels préjudices résultant de votre utilisation des exemples de programme.

Marques

IBM, le logo IBM et ibm.com sont des marques d'International Business Machines dans de nombreux pays. Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark information" à l'adresse www.ibm.com/legal/copytrade.shtml.

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses filiales.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

Dispositions relatives à la documentation du produit

Les droits d'utilisation relatifs à ces publications sont soumis aux dispositions suivantes.

Applicabilité

Ces dispositions s'ajoutent à celles applicables au site Web d'IBM.

Usage personnel

Vous pouvez reproduire ces publications pour votre usage personnel, non commercial, sous réserve que toutes les mentions de propriété soient conservées. Vous ne pouvez distribuer ou publier tout ou partie de ces publications ou en faire des oeuvres dérivées sans le consentement exprès d'IBM.

Usage commercial

Vous pouvez reproduire, distribuer et afficher ces publications uniquement au sein de votre entreprise, sous réserve que toutes les mentions de propriété soient conservées. Vous ne pouvez reproduire, distribuer, afficher ou publier tout ou partie de ces publications en dehors de votre entreprise, ou en faire des oeuvres dérivées, sans le consentement exprès d'IBM.

Droits

Excepté les droits d'utilisation expressément accordés dans ce document, aucun autre droit, licence ou autorisation, implicite ou explicite, n'est accordé pour ces publications ou autres informations, données, logiciels ou droits de propriété intellectuelle contenus dans ces publications.

IBM se réserve le droit de retirer les autorisations accordées ici si, à sa discrétion, l'utilisation des publications s'avère préjudiciable à ses intérêts ou si, selon son appréciation, les instructions susmentionnées n'ont pas été respectées.

Vous ne pouvez télécharger, exporter ou réexporter ces informations qu'en total accord avec toutes les lois et règlements applicables dans votre pays, y compris les lois et règlements américains relatifs à l'exportation.

IBM N'OCTROIE AUCUNE GARANTIE SUR LE CONTENU DE CES PUBLICATIONS. LES PUBLICATIONS SONT LIVREES EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES PUBLICATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Déclaration IBM de confidentialité sur Internet

Les Logiciels IBM, y compris les Logiciels sous forme de services ("Offres Logiciels") peuvent utiliser des cookies ou d'autres technologies pour collecter des informations sur l'utilisation des produits, améliorer l'acquis utilisateur, personnaliser les interactions avec celui-ci, ou dans d'autres buts. Bien souvent, aucune information personnelle identifiable n'est collectée par les Offres Logiciels. Certaines Offres Logiciels vous permettent cependant de le faire. Si la présente Offre Logiciels utilise des cookies pour collecter des informations personnelles identifiables, des informations spécifiques sur cette utilisation sont fournies ci-dessous.

Selon la configuration déployée, la présente Offre Logiciels peut utiliser des cookies de session et des cookies persistants destinés à collecter le nom, le nom d'utilisateur, le mot de passe des utilisateurs ou toute autre information personnelle identifiable pour les fonctions de gestion des sessions, d'authentification, de configuration de connexion unique, ou à des fins de suivi d'utilisation ou de

fonctionnement. Ces cookies peuvent être désactivés mais, dans ce cas, les fonctionnalités qui leur sont associées seront très probablement désactivées.

Si les configurations déployées de cette Offre Logiciels vous permettent, en tant que client, de collecter des informations permettant d'identifier les utilisateurs par l'intermédiaire de cookies ou par d'autres techniques, vous devez solliciter un avis juridique sur la réglementation applicable à ce type de collecte, notamment en termes d'information et de consentement.

Pour plus d'informations sur l'utilisation à ces fins des différentes technologies, y compris celle des cookies, consultez les Points principaux de la Déclaration IBM de confidentialité sur Internet à l'adresse <http://www.ibm.com/privacy/fr/fr>, la section "Cookies, pixels espions et autres technologies de la Déclaration IBM de confidentialité sur Internet" à l'adresse <http://www.ibm.com/privacy/details/fr/fr>, ainsi que la page IBM Software Products and Software-as-a-Service Privacy Statement" à l'adresse <http://www.ibm.com/software/info/product-privacy>.

