Day5 機械学習
サービス導入ハンズオン

ペパボ研究所 財津大夏 / GMO PEPABO inc.

2021.07.21　機械学習研修 Day5

GMOペパボ

カリキュラム概要と目標

- **Day3: 機械学習を自分の言葉で説明できる**
  - データから知識を学習させるための方法について、初歩的な知識を導入する
  - レッスン＆ハンズオンを通して、手を動かしながら知識の解釈を高める
- **Day4: 機械学習を使って課題を解決する流れを思い浮かべることができる**
  - 機械学習で課題を解決するために考慮すべき点について、基礎的な知識を導入する
  - レッスン＆ハンズオンを通して、手を動かしながら知識の解釈を高める
- **Day5: 機械学習をサービスに導入するまでの流れを思い浮かべることができる**
  - 配属先のサービスで機械学習を使った施策を実現するために、知っておくべき知識を導入する
  - Day1~4までの学びを用いて、擬似的なサービス導入準備をハンズオンで体験する

ハンズオン

GMOペパボ

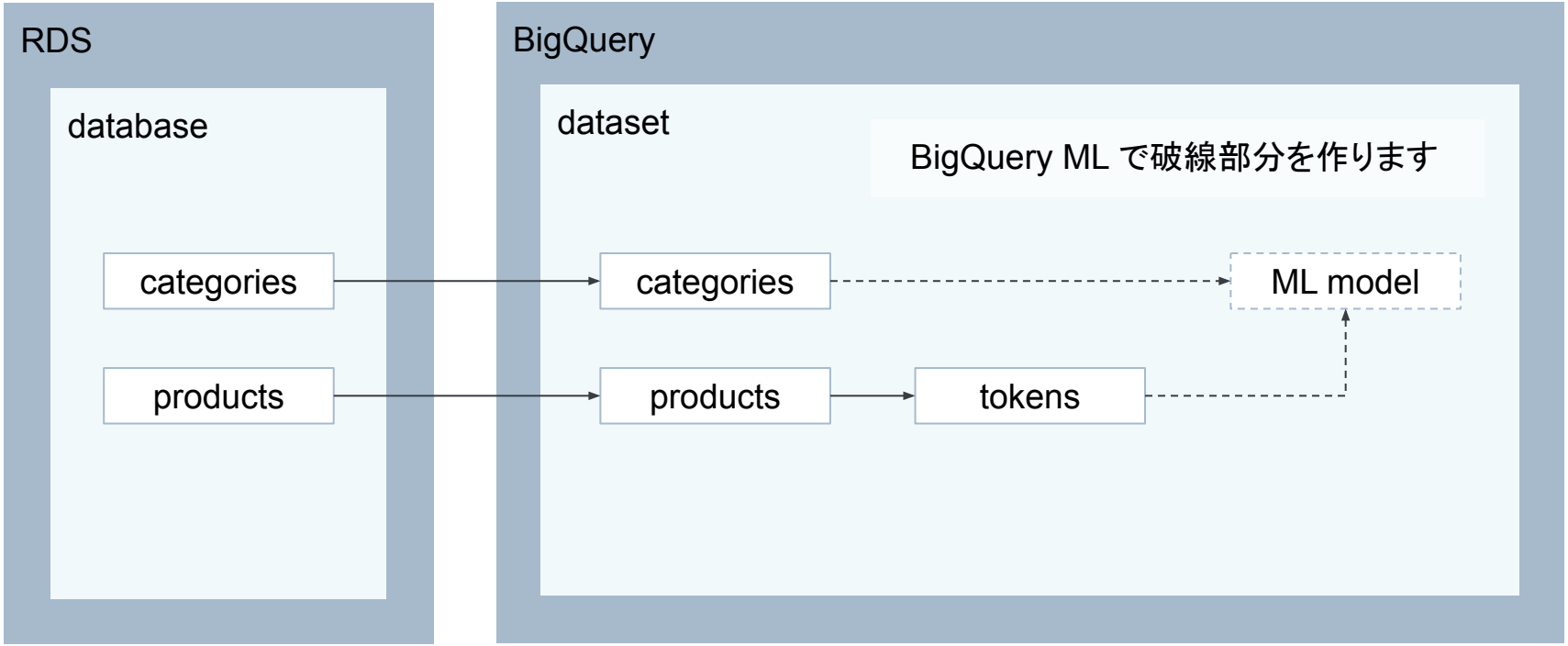
1. 題材の説明（10 分）
    1. お題
    2. パイプラインの構成
    3. BigQuery ML って何？
2. 入力データを作ってみよう（20 分）
3. モデルを作ってみよう（20 分）
4. できあがったモデルを見てみよう（20 分）
5. 予測をやってみよう（20 分）
6. 性能を改善してみよう（20 分）

お題

minne に登録された作品名から作品のカテゴリグループを推定する

- minne では作品の販売時にカテゴリを指定する
  - カテゴリグループ（大カテゴリ）数は 18
  - カテゴリ（小カテゴリ）数は約 240
  - カテゴリを選ぶのが大変
- カテゴリグループの候補を示すと便利そう

想定しているパイプライン



BigQuery ML って何？

" BigQuery ML を使用すると、BigQuery で標準 SQL クエリを使用して、機械学習モデルを作成し実行できます。BigQuery ML では、既存の SQL ツールやスキルを活用できるので、誰でも簡単に機械学習を利用できます。BigQuery ML では、データを移動する必要がないため、開発スピードを向上させることができます。"

BigQuery ML の概要 | Google Cloud より引用

BigQuery ML って何？

- 😍 BigQuery ML のみでモデリングとバッチ予測が可能
  - Cloud Composer を使ったパイプライン運用に載せられる
  - Python 等でのコーディングが不要
- 😍 利用できるモデルが多い
- 😍 特徴量エンジニアリングもできる
  - 欠損値の補完, 標準化, エンコーディングなど
- 😍 AI Platform にモデルをエクスポートすればオンライン予測も可能
- 🥺 モデルの入力の形式によってはエクスポート不可

Step1: 入力データを作ってみよう（BigQuery 演習のおさらい）

- 特徴量は「作品名の形態素」
- ラベルは「カテゴリグループ」
- 特徴量とラベルをカラムとして返すクエリを作成してみましょう
  - minne_tokenized_product_names_training テーブルと minne_categories テーブルと minne_products テーブル
  - 特徴量が Bag of Words の場合は ARRAY of STRING を入力するとモデル作成時にマルチホットエンコーディングされます
  - ARRAY の作り方は 標準 SQL の配列関数 | BigQuery | Google Cloud や 標準 SQL の集計関数 | BigQuery | Google Cloud を見てみましょう

Step1: 入力データを作ってみよう（BigQuery 演習のおさらい）

Row	token	category_group
201	世界	Clothing
	に	
	一つ	
	だけ	
	の	
	ジャケット	
202	オレンジ	Accessories
	×	
	ピンク	
	♡	
	ベ	

Step1: 入力データを作ってみよう（BigQuery 演習のおさらい）

```
SELECT
  ARRAY_AGG(token.surface_form) OVER (PARTITION BY minne_tokenized_product_names_training.id) AS token,
  category_groups.name_en AS category_group,
FROM
  training.minne_tokenized_product_names_training, UNNEST(token) AS token
  LEFT JOIN training.minne_products AS products USING (id)
  LEFT JOIN training.minne_categories AS categories ON products.category_id = categories.id
  LEFT JOIN training.minne_categories AS category_groups ON categories.category_group = category_groups.id
WHERE
  category_groups.id IS NOT NULL
;
```

Step2: モデルを作ってみよう

- 今回はロジスティック回帰モデルを使います
- モデルは CREATE MODEL 文で作成できます
- The CREATE MODEL statement ｜ BigQuery ML ｜ Google Cloud を読んでやってみよう！！
- モデル名は「minne_category_classifier_{your_name}」にしてください
- オプションを変えて複数作ってもOK です
- 学習に 15 分くらい掛かるのでこのあと休憩時間を入れます

Step2: モデルを作ってみよう

```
CREATE OR REPLACE MODEL training.minne_category_classifier_{your_name}
OPTIONS (
   MODEL_TYPE='LOGISTIC_REG',
   INPUT_LABEL_COLS = ['category_group']
) AS
SELECT
  ARRAY_AGG(token.surface_form) OVER (PARTITION BY minne_tokenized_product_names_training.id) AS token,
  category_groups.name_en AS category_group,
FROM
  training.minne_tokenized_product_names_training, UNNEST(token) AS token
  LEFT JOIN training.minne_products AS products USING (id)
  LEFT JOIN training.minne_categories AS categories ON products.category_id = categories.id
  LEFT JOIN training.minne_categories AS category_groups ON categories.category_group = category_groups.id
WHERE
  category_groups.id IS NOT NULL
;
```

GMOペパボ

Step3: できあがったモデルを見てみよう（Details）

- 作成したモデルを BigQuery の Web UI から選択すると、
  モデルの学習時のパラメータや過程、性能を確認できます

Model Details

EDIT

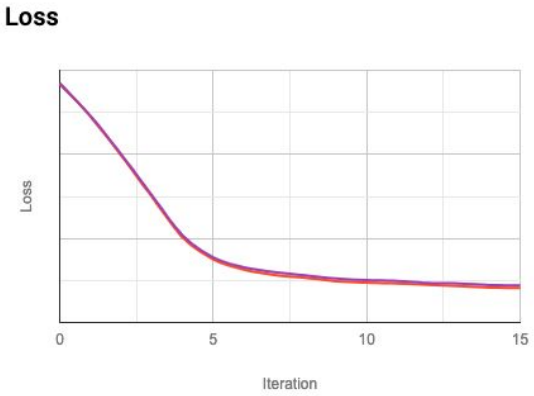
Model ID	[redacted]
Description	
Labels	
Date created	Friday, July 9, 2021 at 4:33:56 PM GMT+09:00
Model expiration	Never
Date modified	Friday, July 9, 2021 at 4:33:56 PM GMT+09:00
Data location	US
Model type	LOGISTIC_REGRESSION
Loss type	Mean log loss

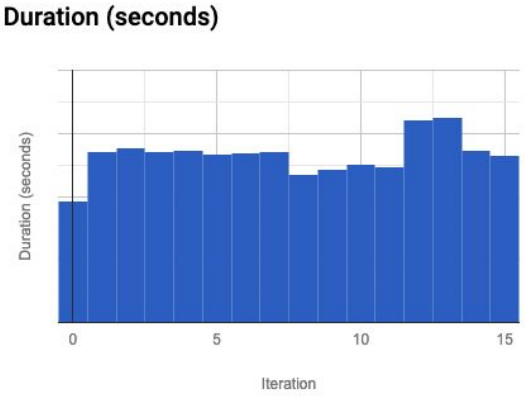
Training Options

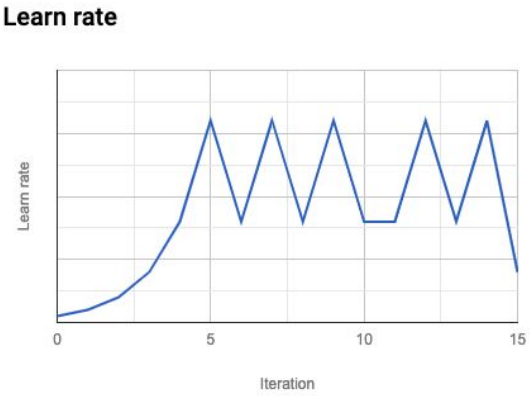
Training options are the optional parameters that were added in the script to create this model.

Max allowed iterations	20
Actual iterations	16
L1 regularization	0.00
L2 regularization	0.00
Early stop	true
Min relative progress	0.01
Learn rate strategy	Line search
Line search initial learn rate	0.10

Step3: できあがったモデルを見てみよう（Training）







GMOペパボ

Step3: できあがったモデルを見てみよう（Evaluation）

- 予め分離した評価データとML.EVALUATE() を使って評価することもできます

Aggregate Metrics

Threshold	0.0000
Precision	0.7384
Recall	0.9166
Accuracy	0.8917
F1 score	0.7962
Log loss	0.2938
ROC AUC	0.9944

Confusion matrix

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in gray).

True label \ Predicted label	Accessories	Aroma and candles	Art	Bags and purses	Children	Clothing	Dolls	Face Masks	Food	Housewares
Accessories	89%	0%	0%	1%	1%	1%	1%	0%	0%	0%
Aroma and candles	–	100%	–	–	–	–	–	–	–	–
Art	3%	–	91%	–	–	–	1%	–	–	–
Bags and purses	2%	–	0%	88%	3%	0%	0%	0%	0%	0%
Children	2%	–	0%	3%	85%	2%	1%	1%	0%	1%
Clothing	1%	–	–	0%	3%	91%	0%	0%	–	0%
Dolls	0%	–	2%	2%	2%	1%	89%	–	–	0%
Face Masks	–	–	–	–	–	–	–	100%	–	–
Food	–	–	–	–	–	–	–	–	100%	–
Housewares	2%	–	–	–	–	–	–	–	–	94%

Step4: 予測をやってみよう

- ML.PREDICT() で予測を行えます
- The ML.PREDICT function | BigQuery ML | Google Cloud を読んで
  やってみましょう
- 入力は minne_tokenized_product_names_prediction テーブル
  または任意の文字列の形態素
  - kuromoji | Atilika のデモでブラウザから任意の文字列を形態素解析できます

Row	predicted_category_group	predicted_category_group_probs.label	predicted_category_group_probs.prob	token
1	Food	Food	0.24478127629161406	コーヒー
		Stationery	0.10206730908741644	セット
		Toys	0.097950756496826982	
		Material	0.077603533837188432	

GMOペパボ

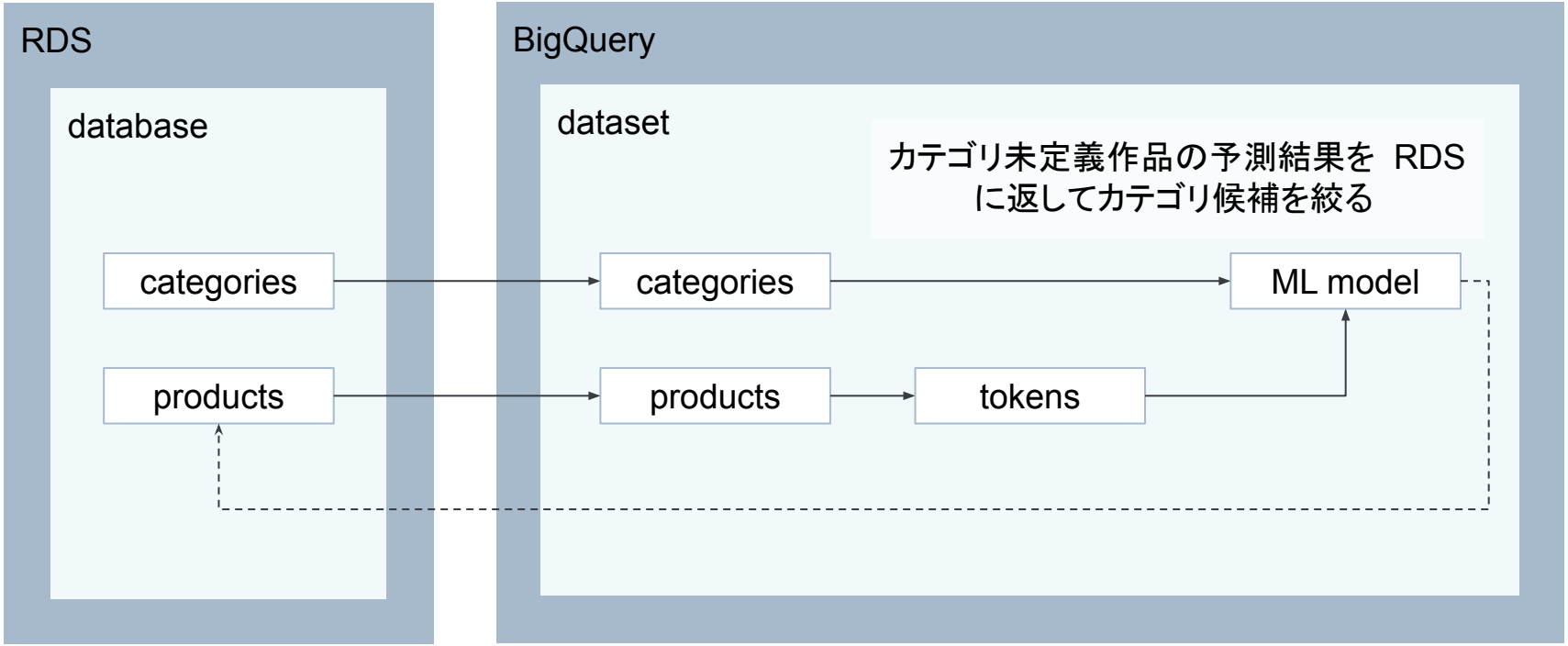
Step4: 予測をやってみよう

```
SELECT
  *
FROM
  ML.PREDICT(
    MODEL rand.minne_category_classifier_logistic_reg,
    (SELECT
      ARRAY(
       #「深煎コーヒー3種セット」
        SELECT 'コーヒー' AS token
        UNION ALL
        SELECT 'セット' AS token
    ) AS token
  ))
;
```

Surface form	Part-of-Speech	Base form
深	形容詞,自立,*,*	深い
煎	動詞,自立,*,*	煎る
コーヒー	名詞,一般,*,*	コーヒー
3	名詞,数,*,*	*
種	名詞,接尾,助数詞,*	種
セット	名詞,一般,*,*	セット

Row	predicted_category_group	predicted_category_group_probs.label	predicted_category_group_probs.prob	token
1	Food	Food	0.24478127629161406	コーヒー
		Stationery	0.10206730908741644	セット
		Toys	0.097950756496826982	
		Material	0.07760353383718843 2	

Step4: 予測をやってみよう



Additional: 性能を改善してみよう

- 例えば
  - 品詞を絞るとどうなるか？
  - 重み付けを変えるとどうなるか？
  - 「○○様専用」作品 や「お知らせ」作品

Additional: 性能を改善してみよう

```
SELECT
  ARRAY_AGG(token.surface_form) OVER (PARTITION BY tokenized_product_names.id) AS token,
  category_groups.name_en AS category_group,
FROM
    minne.tokenized_product_names, UNNEST(token) AS token
    LEFT JOIN minne.product_attributes USING (id)
    LEFT JOIN minne.category_attributes AS categories ON product_attributes.category_id = categories.id
    LEFT JOIN minne.category_attributes AS category_groups ON categories.category_group = category_groups.id
WHERE
    tokenized_product_names.product_name NOT LIKE '%様専用%'
    AND (token.part_of_speech LIKE '名詞,一般%' OR token.part_of_speech LIKE '名詞,固有名詞%')
    AND category_groups.id IS NOT NULL
;
```

お疲れさまでした！

GMOペパボ