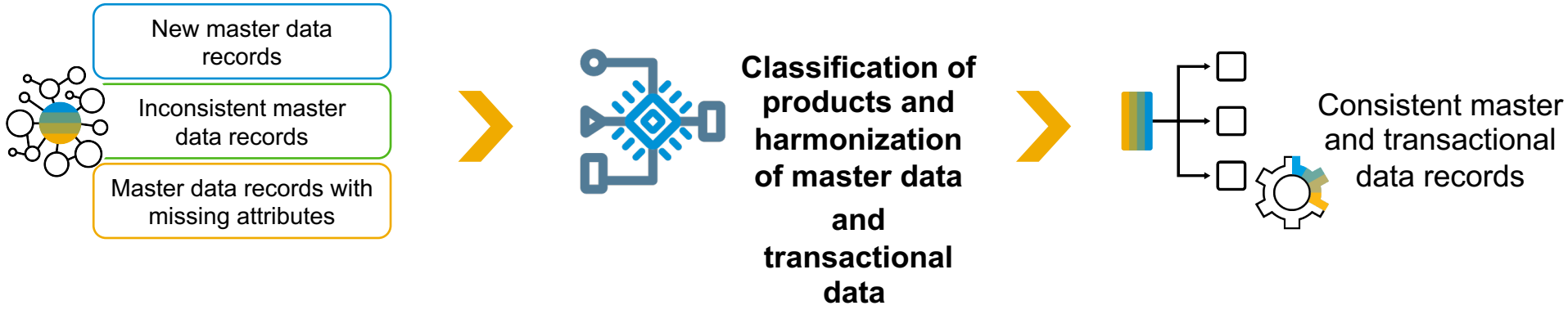
Data Classification with Python SDK and SAP AI Business Services

Speaker's Name, SAP
Month 00, 2020

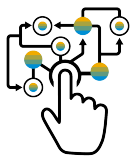
DATA CLASSIFICATION WITH
Data Attribute Recommendation

Data Attribute Recommendation

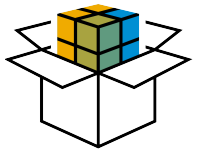
Automate master data / transactional data management tasks



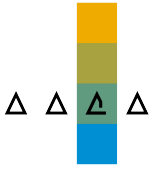
Data Attribute Recommendation helps to classify entities such as products, stores and users into multiple classes, using free text, numbers and categories as input.



Automate and speed up master data creation and maintenance



Gain easier and faster master data insights



Reduce errors and manual efforts when maintaining master data

WORKSHOP HANDS-ON

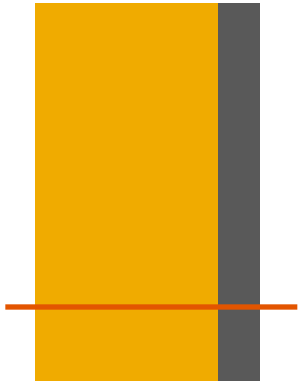
Data Attribute Recommendation Hands-On

- Follow the exercises in the github.com/SAP-samples/teched2020-INT260 repository

Training-Process

Data Requirements:

- The data set should be **>3,000 documents**
  At best: use **all data that is available** on the topic
- The **higher the quality** of the data, the better the results
  Contradictory data worsens the result
- At least **one free text** should be used for training (the more, the better)



Training data consists of **text input** and **labels**

Model is initialized based on the categories found in the **labels.**

Data is split into training testing and validation sets in 8:1:1 ratio.

Use Cases for
Data Attribute Recommendation

Scenario 1 – Match point of sales information to your own product hierarchy

BUSINESS PROBLEM

- Struggle with the **organization and management of large volumes of documents**
- Large amounts of unorganized, yet business critical documents (like contracts, products) are **preventing the workforce to work efficiently**

SOLUTION

Data Attribute Recommendation:

- Can be used for a **prediction performance**: bulk and individual classifications of materials and its characteristics are done online
- **Enables intelligent master data governance** at companies struggling with the creation of new material requests

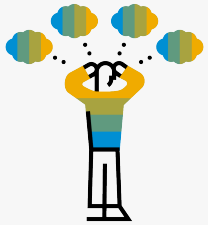
BENEFITS

- **The only way** to master the chaos is by using automation
- **Reducing** the manual work
- **Speeding up** and **improving the quality**
- **Easily Customizable**
- **Improve quality of data**

SAP SOLUTIONS IN THIS USE CASE:

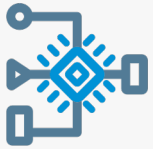
- *Data Attribute Recommendation via SAP Cloud Platform Enterprise Agreement*
- *SAP Demand Signal Management*
- *SAP Cloud Platform Enterprise Agreement*

Scenario 2 – Get suggestions of material class and its characteristics

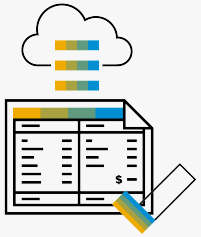


New material = big effort

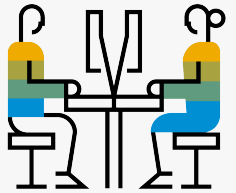
Incorrectly chosen material classes



Data Attribute Recommendation



Automation of repetitive work by prediction performance



Value-creative use of labor-hours

BUSINESS PROBLEM

Creating new material requests:

- Big, time consuming effort
- **Many requests coming back** due to incorrectly chosen material class

SOLUTION

Data Attribute Recommendation:

- Identifies relevant matching patterns
- Following these patterns it predicts to which representation in the product hierarchy the incoming product record belongs which helps to gain direct insights in the incoming point of sales information

BENEFITS

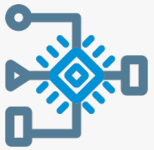
- Enables organizations to analyze valuable information about their competitors in comparison to themselves
- **Acceleration** of the process
- **Reduction of more than 20% of the effort** for new material creation
- **Automation of the repetitive work** and use labor hours on **high value tasks**

SAP SOLUTIONS IN THIS USE CASE:

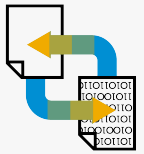
- *Data Attribute Recommendation via SAP Cloud Platform Enterprise Agreement*

Scenario 3 – Commodity Code Prediction

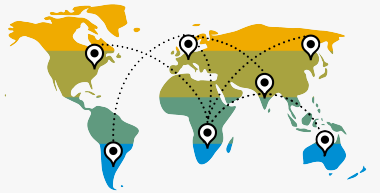
New Commodity,
no Commodity Code



Data Attribute
Recommendation



Commodity
Code



New Commodity is
Imported / Exported

BUSINESS PROBLEM

- **International trade demands commodity codes**
- **No correct commodity code**
  → **Rejection** of assignment goods by customs
  → **Waiting on borders** = **high costs** for the organization
- **Maintaining** such master data
  → is supported by rules
  → leads to **inconsistencies & left-overs for manual processing**

SOLUTION

Data Attribute Recommendation can predict:

- **Correct values by learning the logic behind** complex fields *(such as the commodity code)*
- **Dependencies** to multiple data attributes

BENEFITS

- **Reduces manual efforts** in commodity code assignment
- **Increases the validity** of assigned commodity codes
- **Decreased costs:** caused by labor & mal-assignment of commodity codes

SAP SOLUTIONS IN THIS USE CASE:

- *Data Attribute Recommendation via SAP Cloud Platform Enterprise Agreement*

DATA CLASSIFICATION WITH SERVICE TICKET INTELLIGENCE

Service Ticket Intelligence with SAP Artificial Intelligence

Reimagine Customer Service with Automated Processes

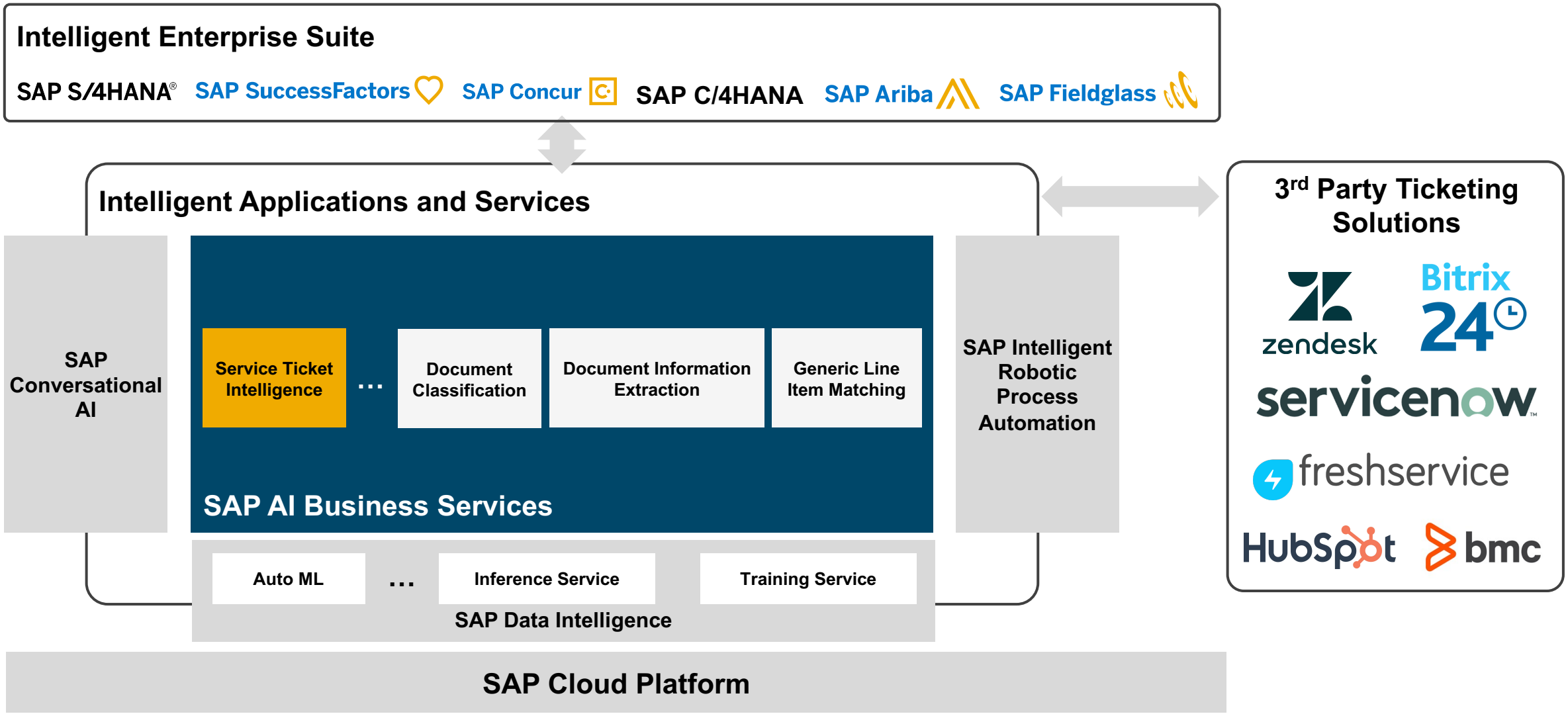
Process customer issues faster and deliver great customer experience

Speed up service response times

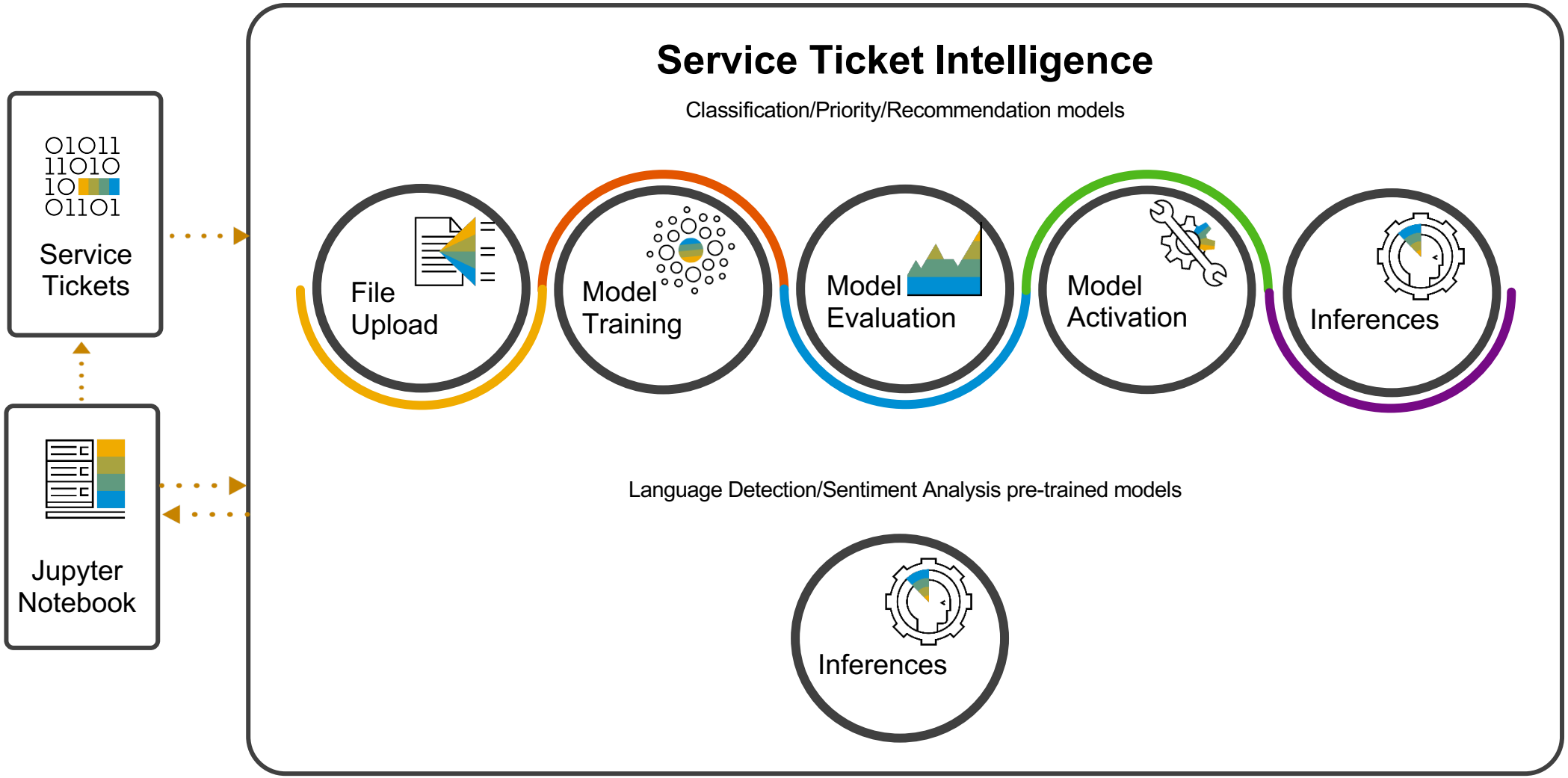
Have higher rates of problem resolution and ticket closure

Increase customer satisfaction

Service Ticket Intelligence can be integrated with various ticket handling applications



Service Ticket Intelligence Overview
Ticket Classification and Recommendation Scenarios



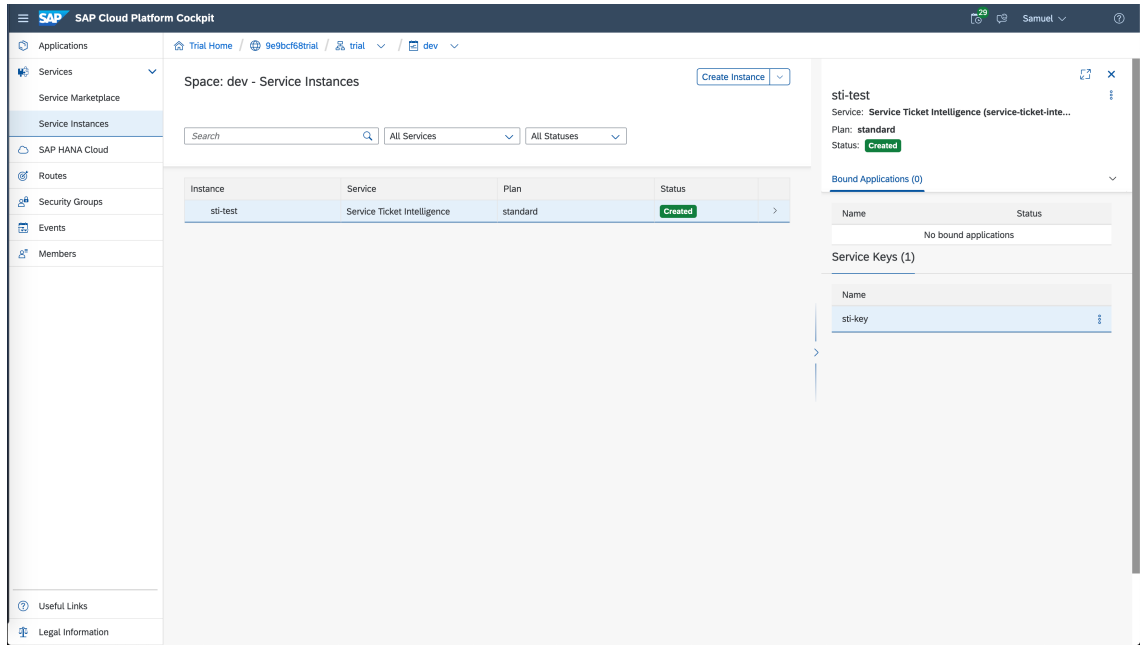
WORKSHOP HANDS-ON

Exercise 1 – Provision and retrieve service keys for Service Ticket Intelligence

1. Register for a SAP CP trial account and login to https://account.hanatrial.ondemand.com/cockpit/
2. Follow Exercise 1 to locate your service keys
3. Find your service credentials in the service keys

Service Key Parameter	Value
uaa-url	
uaa-clientid	
uaa-clientsecret	
sti_service_url	

4. Your service keys are required for Exercise 2



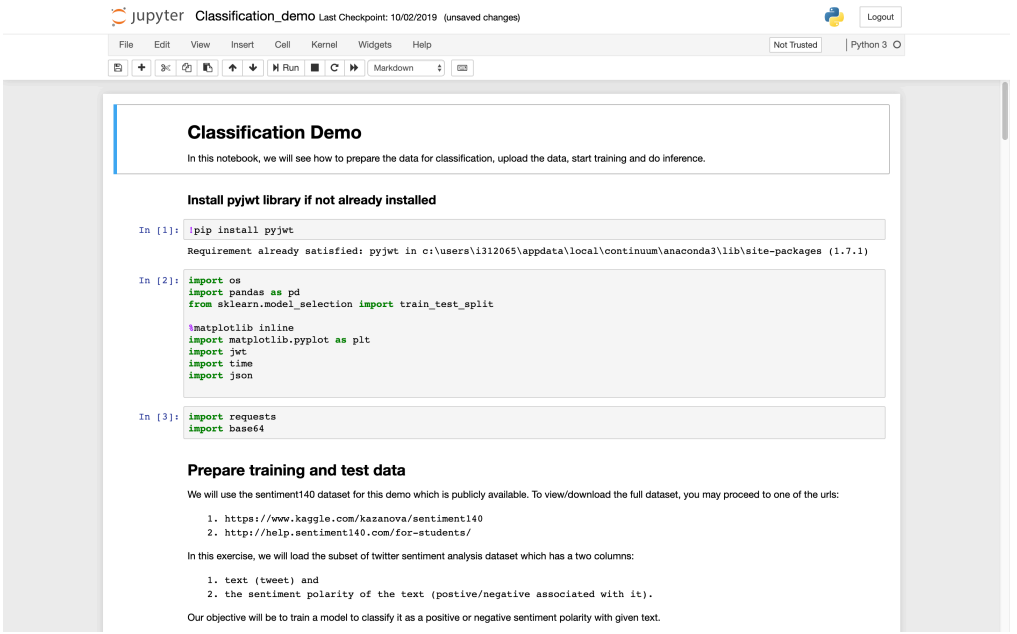
Learning point: The client ID provided has full privileges to the service. The client application should grant different access privileges to different user roles (e.g. admin versus end user). Access and secret keys should not be given to non-admin users. Read more in STI Security Guide.

Exercise 2 – Ticket Classification Scenario
Data preparation, model training & evaluation, inferences

1. Jupyter Notebooks are interactive playgrounds to code and are often used in data science to explore datasets

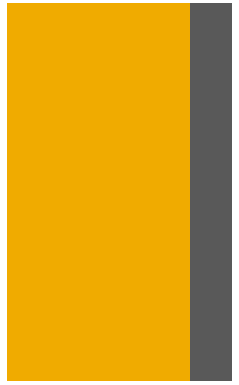
2. Open the Classification Jupyter demo notebook in Exercise 2 and follow the steps

3. Dataset used in this demo will be in the context of service request complaints from the finance industry

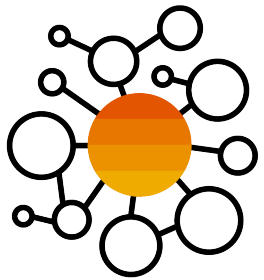


Learning point: STI Classification models are retrainable with multiple input labels for prediction. The input file should be UTF-8 formatted csv. Read more at STI API Reference > Model Training API

What happens during model training?



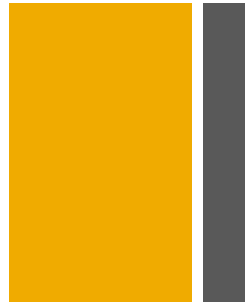
Training data upload via file upload or oData connection with **text input** and **labels**



Model is initialized based on the categories found in the **labels.**

Data is split into training and validation sets.

Model training process automatically runs iterations of model training and validation to optimize the classification results across the categories found



Training Set

1 Train model

Model

2

Validation Set

Validate model by comparing true and predicted labels

3

4

True labels

Predicted labels

STI Ticket Classification (Categorization) Model

New Ticket

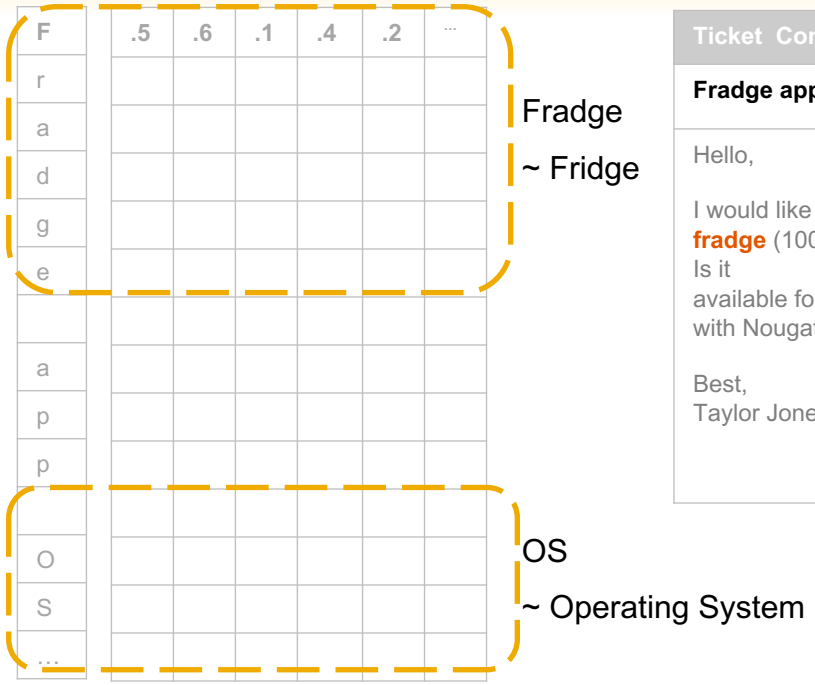
Ticket 99094
Fradge app OS version
Hello, I would like to connect to my smart fridge (10003380) using the app. Is it available for Lollipop (5.0) or only with Nougat (7.0)? Best, Taylor Jones

INPUT → MODEL → OUTPUT

Proposals

Ticket	Category	Confidence
99094	Product Support	92%
.....	.....	.....

Character-level Tokenization | **Token-level CNN*** | **Sentence-level Attention**



Ticket Content
Fradge app OS version
Hello, I would like to **connect** to my smart **fradge** (10003380) using the **app**. Is it available for Lollipop (5.0) or only with Nougat (7.0)? Best, Taylor Jones

- Language independent approach (technically supports all languages)
- One model for all languages (good to deal with insufficient training datasets of minor languages)
- Support tickets of mixed languages

*Convolutional Neural Network

Model Training Best Practices

Recommended number of records

- At least 1,000 samples for every unique label value

Minimum number of records

- At least 20 samples for every unique label (labels, e.g. categories, with less than 20 training examples are treated as outliers and are removed automatically by STI)

Data distribution

- Uniform data distribution across all unique labels (good to have)

Business activity level

- At least 10,000 service tickets per month

Generally speaking, the more data the better..

Evaluating model accuracy

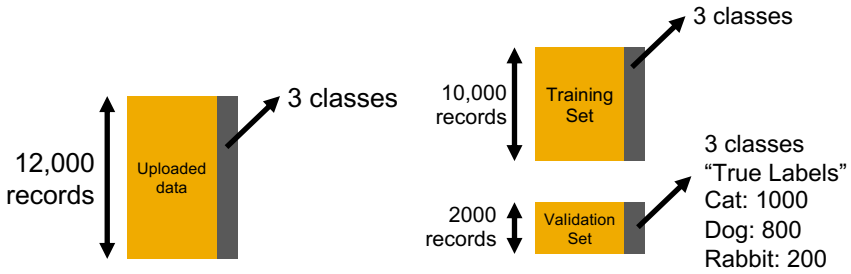
- A model training report in the form of a confusion matrix on the validation set, and corresponding precision, recall and f1 values for each category label in the dataset is generated at the end of the model training process. This report (in JSON format) can be retrieved from the Get Model Accuracy API

- Overall model accuracy represents the test results from using the model to make predictions on the validation set. This value can be retrieved from the Get Model Status API as "combined_accuracy" in the Model Status Object.

Best practices to improve model accuracy

- Prior to model training, evaluate how much training data is available, the absolute number of categories, and the distribution of data across these categories.

- If a dataset has a large number of categories and semantically similar categories (or data which is very similar between two categories e.g. request for refund vs request for replacement), this typically makes the data noisy and thus results in poorer accuracies.



There are 1100 actual cats (true labels), of which 850 were accurately classified as cats, 200 were classified as dogs and 50 were classified as rabbits.

Precision rate for the 'cat' label is 0.77 (850 / 1100).

Of the 1080 cats predicted by the model (predicted labels), 850 were truly cats.

Recall rate for the 'cat' label is 0.79 (850 / 1080).

The f1 score is a harmonic mean of the precision and recall values, which therefore gives a single score that represents both precision and recall rates for a particular label.

Overall model accuracy

Sum of total correct predictions over all predictions made on the validation set.

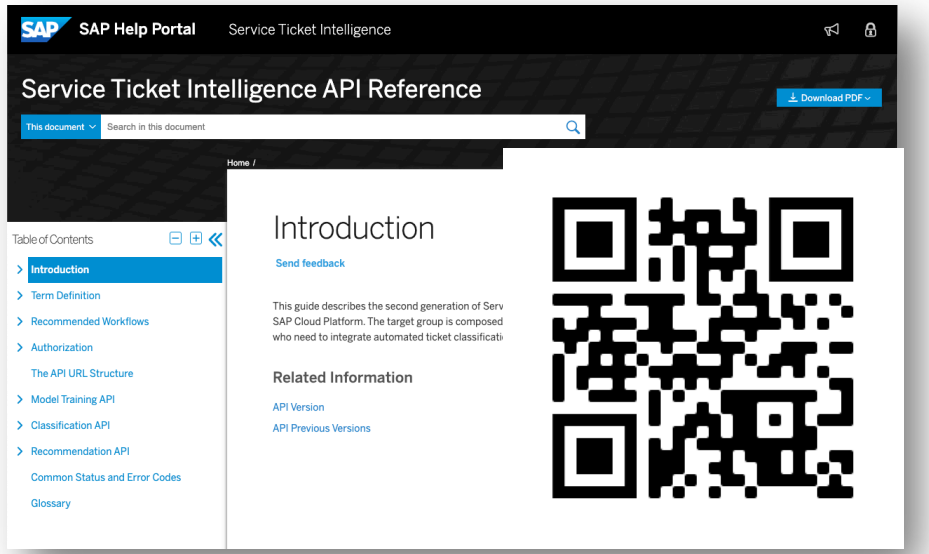
(850 + 200 + 100) / 2000 = 0.575

True label				
	Cat	850	200	50
	Dog	150	200	50
	Rabbit	80	20	100
		Cat	Dog	Rabbit
		Predicted label		

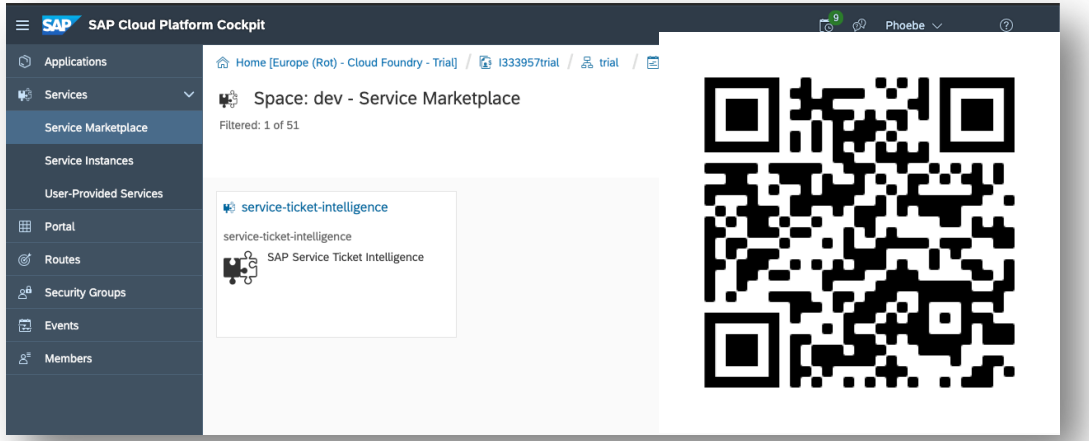
```
confusion_matrix : {
     labels: [
              'cat',
              'dog',
              'rabbit'
     ],
     values: [
              [850, 200, 50],
              [150, 200, 50],
              [80, 20, 100]
     ]
}
```

See term definition for the Confusion Matrix Object

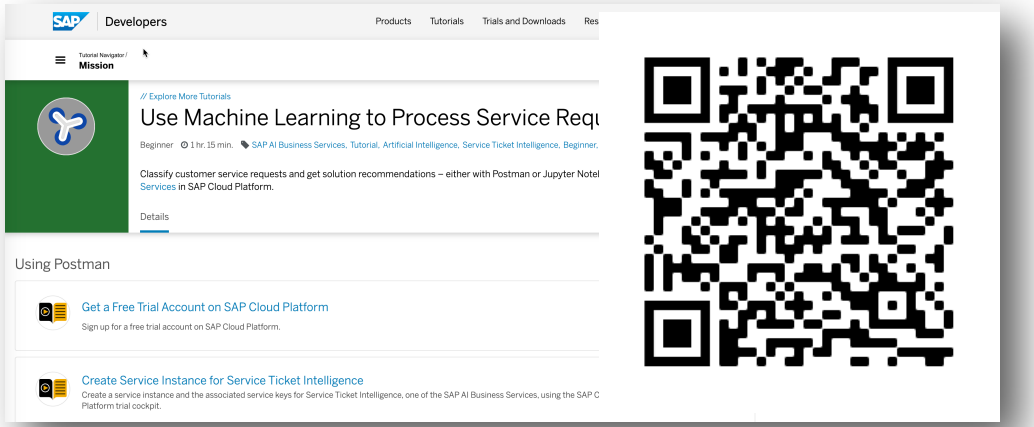
Developer resources



API documentation on help.sap.com/stint > API Reference



Personal Trial on SAP Cloud Platform



Developer Tutorials

YOU ASK, I ANSWER

Thank you.

Contact information:

F name L name
Title
Address
Phone number

Partner logo

THE BEST RUN SAP