Using machine learning in Microsoft Dynamics AX 2012 R3 – Product recommendations in the Call center module

This document showcases Microsoft Cognitive Services Recommendations API integration that is included in the Cortana intelligence services management module of Microsoft Dynamics AX 2012 R3 (Cumulative Update 12). We utilize the power of machine learning to incorporate product recommendations in the Call center module's Customer service process.

Table of Contents

Overview

Microsoft Cognitive Services Recommendations API – exposes a general-purpose recommender system capability wrapped in a simple to use REST API. **KB 3201875** introduces this machine learning capability in Microsoft Dynamics AX 2012 R3 by way of deep integration into the platform, with the newly introduced **Cortana intelligence services management** module in Cumulative Update 12.

In this document we describe how you can use that capability to customize the Call center module for generating product recommendations in several key scenarios with relative ease.

Pre-requisites

The following pre-requisites and configuration are required for consuming this sample.

Product baseline

- Microsoft AX 2012 R3 with Cumulative Update 12 OR
- Microsoft AX 2012 R3 with Cumulative Update 11 with Cortana Intelligence services management hotfix applied (KB 3201875) OR
- Microsoft AX 2012 R3 with Cumulative Update 8 or higher with Entity Store hotfix (KB 3147499) and Cortana Intelligence services management hotfix applied (KB 3201875)

Cortana intelligence services management

A detailed description of the capabilities of the Cortana Intelligence services management hotfix and its capabilities can be found here - https://go.microsoft.com/fwlink/?linkid=835925. At a high level, the following must be configured

- Setup *Cognitive service account* detail in the **Global parameters** form

- Create one or more Recommendations API model using **Recommendation models** form with "Model entity" field set to any relevant label for e.g., '*CallCenterEntity*' as is used in the rest of the document.

- Update valid catalog and usage datasets to the model either manually or in an automated fashion

- Create one or more builds of type – *Recommendation*, with one build set as the active build for *scoring*.

NOTE: It is important you follow the whitepaper before proceeding with the rest of the document

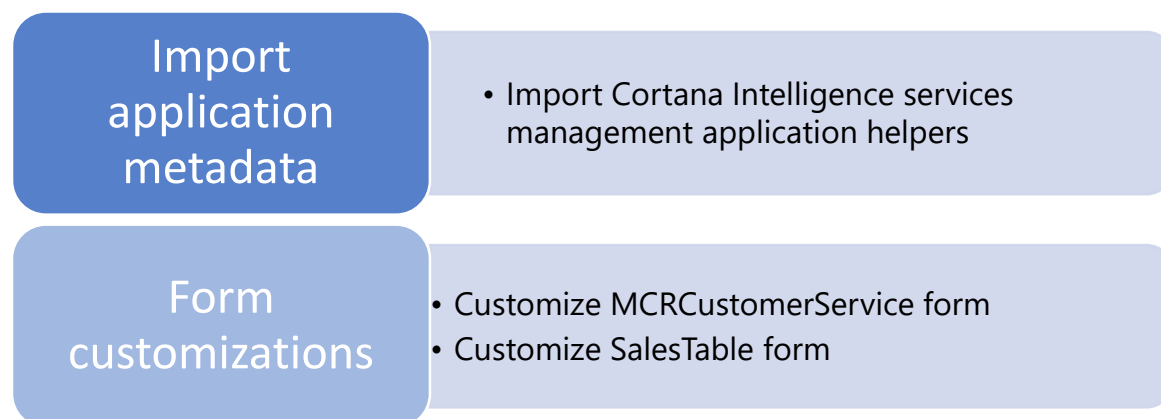# Call center module in AX 2012 R3

The Call center functionality in AX 2012 R3 includes features that are designed to make it easier for a call center worker to take order via phone and handle customer service throughout the order fulfillment process. In addition, the module supports several functions such as

- Enter payment information directly into the sales order
- View a detailed summary of charges and payments before order creation.
- Options for controlling pricing,
- Access data about customers, products, and prices from the sales order form
- Access functionality for tracking customer history and order status.

See **Appendix D** for additional information.

Customizing Call center for product recommendations

The Customer service form in the Call center module allows several common tasks to be completed by the call center worker. One of the core functionality is to create new sales orders. A call center worker can also search for items and/or edit existing sales orders on the customer service form. It is also possible to setup items for cross-sell/up-sell and to make offers to customers as those items are added to the order. We now present to you a sample customization in this functionality for generating real-time product recommendations. This functionality will leverage the Cortana intelligence management service management feature and use your product catalog and sales transaction data to generate product recommendations that reflect real trends and customer preferences. This sample does not showcase any integration with the existing cross-sell/up-sell functionality in AX, however, one can extend this sample to integrate into that business process also.

Import application metadata	• Import Cortana Intelligence services management application helpers
Form customizations	• Customize MCRCustomerService form • Customize SalesTable form

Import Cortana Intelligence services management application helpers

To begin the customization, you must first import a few metadata artifacts that can be found in the repository.

AxCISen-us.ald – This label file contains labels for the metadata included in the helper project. You can choose to merge this into an existing label file in your customization layer or wrap this into its own model along with the rest of the application metadata below.

**CortanaIntelligenceApps_CallCenter.xpo** – helper project that contains following metadata

- Class that wraps the Recommendations API to generate recommendations for a sales order
- Tables to store the product recommendations and for gathering associated metric
- Form to display recommended products and a display menu for the recommendation form
- Form part that can also be used in your form to display recommendations
- Security privilege that can be added to any duty for accessing to metadata above

You can import the label file using the "Create from File" option on the Label Files node in AOT. Import the project *xpo* once the label file is imported. Compile and run the software update checklist to ensure there are no unreconciled changes.

Form customizations

Customize MCRCustomerService form

- To customize **MCRCustomerService** form, add the form to the model created and add a new button group control under the *Customer* action pane tab. Create a new command button under the new button group. Set a valid text and button display properties on the command button as shown below.
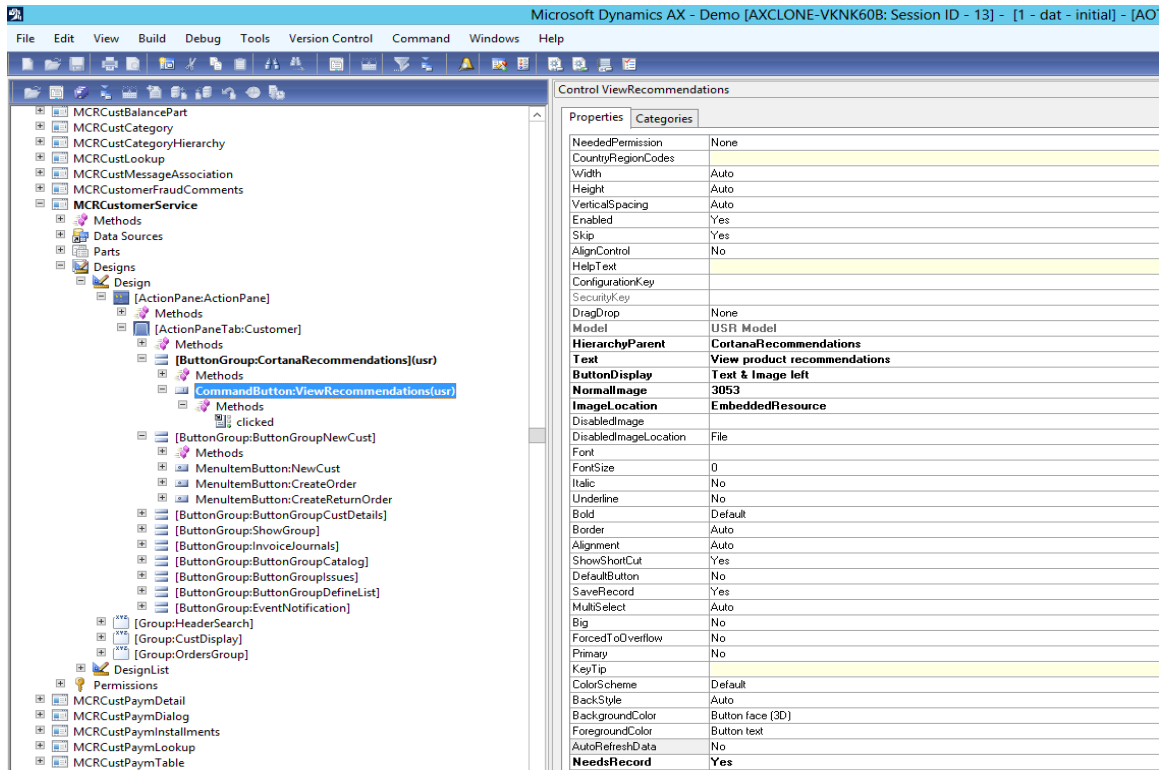
- Override *clicked* method of the command button and add the following code.

```
void clicked()
{
    // A literal constant that must be also specified on the Recommendations model's
    // "Model Entity" field. The default model of this type will be used in generating
    // product recommendations
    #define.CallCenterEntity('CallCenterEntity')

    Args _args;
    MenuFunction menuFunction;
    super();

    // Generate recommendations
    CortanaAppUtils::getRecommendationsForSalesOrderAndEntity(SalesTable.SalesId,
                                                    #CallCenterEntity);

    // Display recommendations in CortanaRecommendationsForm form
    menuFunction = new MenuFunction(menuitemdisplaystr(CortanaRecommendationsForm),
                              MenuItemType::Display);
    _args = new Args();
    _args.caller(element);
    _args.record(SalesTable);
    menuFunction.run(_args);

}
```

- Compile **MCRCustomerService** form and ensure there are no errors.

Customize SalesTable form

The Call center worker uses the Sales order form to create new orders. Our customization adds a new form part to this form to enable displaying product recommendations. We also update the *SalesLine* datasource's *write* method to invoke the Recommendations API for this purpose. The product recommendations are written to a table and displayed in the form part added.

- Open **SalesTable** form in your developer workspace and add a new part reference under the Parts node

- Reference the **CortanaRecommendationFormPart** menu in the properties and setup the form part properties as shown below
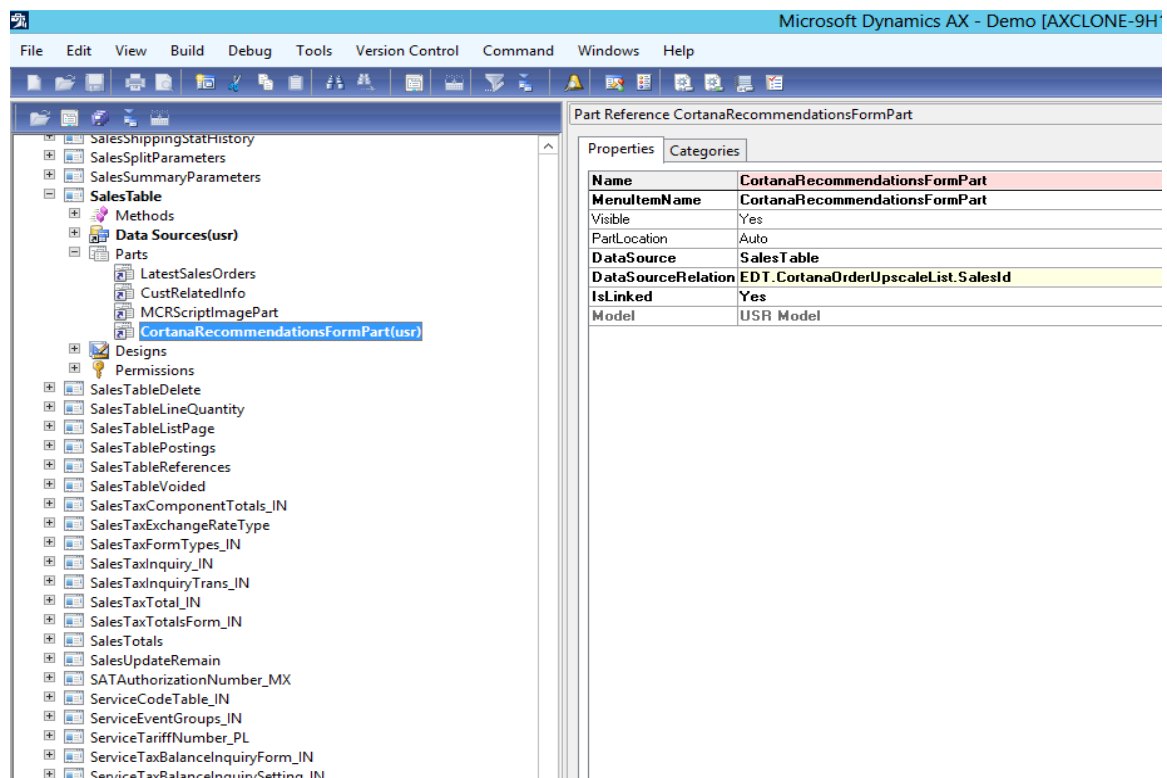
- Right click on the **SalesTable** form and click View code**.**

- Add the following line of code at the start of the *SalesLine::write* method

```
// A literal constant that must be also specified on the Recommendations model's
// "Model Entity" field. The default model of this type will be used in generating
// product recommendations
#define.CallCenterEntity('CallCenterEntity')
```

- Add the following lines of code at the end of the *SalesLine::write* method.

```
// Track new sales line additions. This will be used at a later point to gather metric
// about the efficacy of the recommendations made
CortanaAppUtils::trackProductRecommendations(SalesTable.SalesId, salesLine.ItemId);

// If a new sales line is added, trigger generating recommendations
CortanaAppUtils::getRecommendationsForSalesOrder(SalesTable,
                                                 #CallCenterEntity);
```

- In SalesTable form add following code in *close* function before the call to *super()*.

```
// Sends telemetry data to Microsoft's Application Insight service
CortanaAppUtils::sendRecommendationMetric(SalesTable.SalesId);

super();
```

- Compile **SalesTable** form and ensure there are no errors.

NOTE: See Appendix C for additional detail on the telemetry data gathered

Using Product recommendations in Call center module

Once the customization is complete, launch Customer service form from *Call center > Journals > Customer service* menu, select any existing sales order and then click the "View product recommendations" action pane menu to view the recommendations for the order. In this case, the Recommendations API takes into consideration the basket of items in the order to generate product recommendations.

NOTE: The illustration above shows the recommendations generated for sales order – 012412 that can be found in the DEMO dataset.

To see the product recommendations in the Sales order form, click *New > Sales order* action pane menu in the Customer service form to create a new sales order. You should now see the form part that was added earlier. When you add a new sales line and save the changes, the *SalesLine::write* logic is executed that generates and stores the recommendations made. Refreshing the form part at this point will display relevant product recommendations for the item(s) added.

NOTE: The illustration above shows the recommendations in the sales order form.

There are a few things to note as you consume this sample.

- The Recommendation API service calls typically take a few seconds to return except the first time when the latency is between 5 – 10 seconds.

- The default number of product recommendations is always less than or equal to 10. The Global parameters form in the Cortana intelligence services management module contains a parameter that dictates the number of product recommendations that must be generated for the *FBT* and *Recommendation* build types. You can update these values to make the API return the desired number of product recommendations.

- The recommendations fetched from the API are stored in a table indexed by the sales order id. Invoking the – *CortanaAppUtils::getRecommendations** methods results in a service call to the Recommendations API. Given that the API charges are based on the number of calls made, it is

possible to modify our logic to cache these results for a period of time. It is however important to note that the quality of recommendations is directly dependent upon the quality of the usage data (sales transactions) that are used in creating a model as well as the context of the product recommendations request i.e., the state of the sales order basket. Any caching thus made will impact the quality of recommendations made.

Conclusion

In this document, we show how you can leverage the Recommendations API integration in Cortana intelligence services management module to apply machine learning to the Call center sales order creation process. The Cortana intelligence services management whitepaper (Appendix D) outlines several other interesting scenarios that can be implemented with the Recommendations API integration. It is also important to note that the Microsoft Cognitive Services provides an easy way to integrate other machine learning capabilities in Microsoft Dynamics AX. There exist several business processes and use cases that can be improved for better efficiency and a great customer experience.

Appendix A: Cortana Intelligence Suite

Microsoft Cortana intelligence suite is a collection of technologies for information management, big data storage, machine learning and analytics and data visualizations. Cortana assistant on Windows 10, Chabot made with Bot framework, Cognitive services and many other applications utilize one or many technologies marketed under the brand of Cortana intelligence.
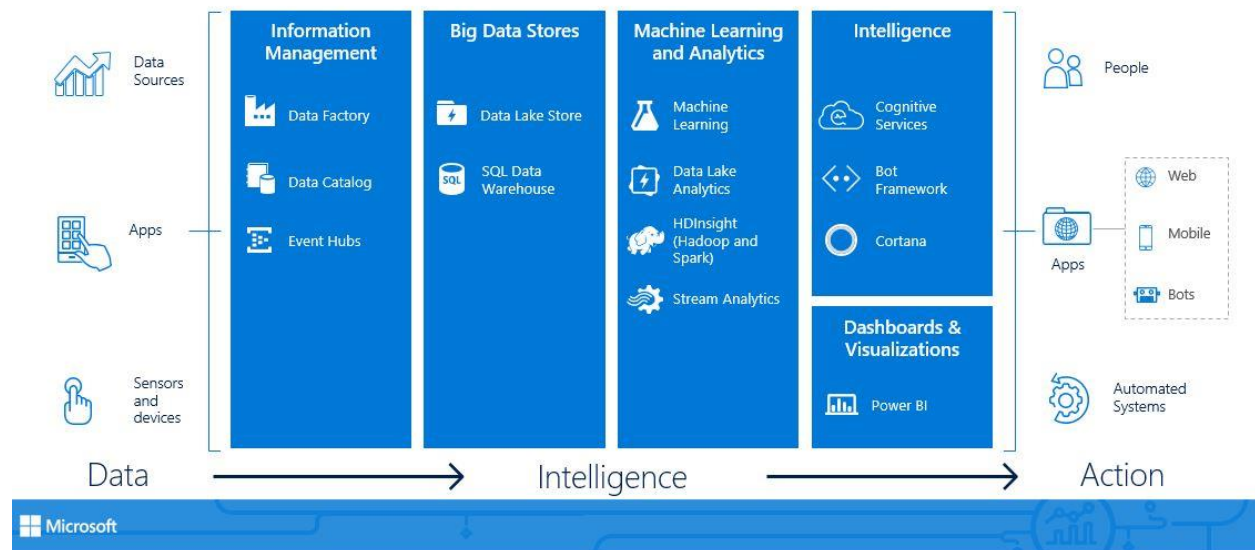
1

See **Appendix C** in the Cortana intelligence services management whitepaper for details.

Appendix B: Microsoft Cognitive Services

The Microsoft Cognitive Services APIs are a suite of several general-purpose Machine Learning APIs that are made available in Microsoft Azure and can be used for any number of applications. These APIs simplify the whole process by abstracting away the complex machine learning models and the operationalization aspects so that users can focus on real business problems. There are several categories of Machine Learning APIs and the **Knowledge** category includes the **Recommendations API.** See **Appendix C** in Cortana intelligence services management whitepaper for details.

Appendix C: Recommendations API usage telemetry

In the sample provided we use a couple of functions to gather telemetry about how efficient the product recommendations are in terms of upscale percentage. This is measured as a function of the number of items added or removed from amongst the recommended items.

CortanaAppUtils::trackProductRecommendations – records the recommended items that were added to a sales order.

CortanaAppUtils::sendRecommendationMetric – sends usage related metric to Microsoft's telemetry service. Note that these tracked metrics do not contain any PII (Personally Identifiable Information) from your organization and we strongly recommend that you include the related customization. The following information is gathered from the usage

- *UpscalePercent* - Improvement in sales using Microsoft Cognitive Services Recommendations API
- *NumRecoItemsAdded* – Number of items which were added by looking at recommendations
- *NumRecoItemsRemoved* – Number of recommended items that were removed from sales order (these items were added earlier by looking at recommendations)
- *NumUniqueRecoItemsSold* – Number of unique recommended items in this sales order
- *CountOfRecoSKUsSold* – Number of recommended item SKUs added in this sales order

Appendix D: References

- AX 2012 R3 Cumulative Update What's New document - https://go.microsoft.com/fwlink/?linkid=834752
- Whitepaper: Cortana intelligence services management service - https://go.microsoft.com/fwlink/?linkid=835925
- Call center module in Microsoft Dynamics AX 2012 R3 - https://technet.microsoft.com/en-us/library/dn497816.aspx
- Common tasks using Customer service form in Call center module - https://technet.microsoft.com/en-us/library/dn497711.aspx
- Sales order creation in call center - https://technet.microsoft.com/en-us/library/dn631664.aspx