# Dynamics 365 CE trial setup

Go to <https://trials.dynamics.com/> to setup a Dynamics 365 Customer Engagement trail environment.

Select the “Sales” option and follow the instructions to complete the trial setup.

The Dynamics 365 trial will include PowerApps (<https://web.powerapps.com>) and Flow (<https://flow.microsoft.com>).

# Azure Trial setup for Text Analytics and Machine Learning

Go to <https://azure.microsoft.com/en-us/free/> to setup a trial Azure account with access to all Azure free services and a $200 credit to try premium services for 30 days.

# Dynamics 365 CE Customization

Create a new solution to contain the Opportunity form and the new fields we’ll be creating:

1. Navigate to <https://web.powerapps.com> and login with the proper Azure AD credentials
2. From PowerApps – make sure the proper environment is selected. Go to the Solutions folder from the left pane.
3. Select New Solution from the menu bar
4. Display Name: Customizations for Opportunity Prediction
5. Publisher: select “CDS Default Publisher”
6. Version: provide a version number, i.e. 0.1.0
7. Click “Create” at the bottom of the screen

Add the Opportunity entity and form to the new solution:

1. Click the new solution from the Solutions list to open it
2. On the toolbar, select “Add Existing” and select “Entity”
3. In the Add Existing Components dialog, use the Search bar to find the Opportunity entity
4. Select the Opportunity entity and click “Next”
5. On the Selected Entities screen, click the “Select Components” link
6. Click on “Forms”
7. Select the Opportunity main form and click “Add”
8. Click “Add”
9. Now the solution has one entity and one form

Add new fields to the Opportunity entity:

1. In the solution, click on the Opportunity entity. You’ll see that there are no fields listed.
2. Click “Add Field” on the toolbar
3. Display Name: Current Situation Sentiment Score
   1. Customer Need Sentiment Score
   2. Proposed Solution Sentiment Score
   3. Notes Average Sentiment Score
   4. ML Outcome Prediction
4. Data Type: Text
5. After creating the 5 new fields, click “Save Entity” at the bottom of the screen

Add the new fields to the Opportunity form for display:

1. Select Forms and open the main Opportunity form in the form designer
   1. This is the new form designer – recently updated with the April Release Wave
2. Drag the new fields from the left panel to the Summary tab on the form
3. Click Save in the upper right corner of the screen
4. Close the Form Designer and click “Done”
5. Click the solution name to move back out to the solution
6. Click “Publish all customizations” on the toolbar

Open the Dynamics 365 app and navigate to Opportunities.

Open an existing record or create a new record just to check the fields on the form.

# Import test data – Opportunities and Notes

Create a new account record called “Microsoft.” We’ll use this as the parent customer for a set of opportunities we’ll be importing.

1. From the Sales area, select Accounts
2. Click “New” on the toolbar to open a new account record.
3. Account Name: Microsoft
4. Complete any other required fields
5. Click Save & Close from the toolbar

Open the Power Platform Admin Center to access the Data Import wizard

1. <https://admin.powerplatform.microsoft.com/>
2. Select “Environments” folder from the left pane
3. Select the environment you’re working in
4. Click “Settings” from the toolbar
5. Under “Data Management”, click the “Data Import Wizard”
6. On the toolbar, click Import Data
7. Select the local file to import Opportunities: “Opportunity Advanced Find View All Required Columns.csv”
   1. “Note – New.csv”
8. Click “Next” and click “Next again
9. In the Microsoft Dynamics 365 Record Types list, select “Opportunity”
10. Click Next
11. The system will automatically map the fields except one field – “(Do Not Modify) Opportunity”
    1. You need to select “Opportunity (Primary Key)” from the drop down
12. Click “Next”
13. Click “Next” on the Mapping Summary screen and click “Submit” to start the import
14. You can check the status in the “My Imports” view
15. Follow the same steps to import the Notes data as well

# Text Analytics service setup in Azure

We need to provision an instance of Text Analytics in Azure Cognitive Services. You must have an Azure Subscription to provision services. (This can be a trial account in Azure, if necessary.)

Navigate to <https://portal.azure.com> and login with credentials that allow service creation.

First, create a new Resource Group to contain the Azure services for this lab:

1. Click on “Create a resource” in the left pane at the top.
2. Type “Resource group” in the search box and select “Resource group” from the results list.
3. Click “Create” after reviewing the details of what a resource group is.
4. Subscription: Select the proper subscription that will contain the new resource group.
5. Resource Group: give the resource group a meaningful name, i.e. OpportunityProbabilityLab
6. Region: probably want East US or East US 2
7. Click “Review and Create” at the bottom of the screen
8. Click “Create” if the information is correct

The new resource group will be created and you’ll see a notification message attached to the bell icon in the toolbar.

* Click on “Go To Resource Group” to easily jump to the resource group blade

Within the new resource group, we’re going to provision the Text Analytics service:

1. In the resource group blade, Select “Add” on the toolbar
2. In the search box, type “Text Analytics” to show a list of related services
3. Select the “Text Analytics” option from the list
4. Click “Create” after reviewing the details of the Text Analytics service
5. Name: give your service a meaningful, i.e. TextAnalyticsLab
6. Subscription: select the subscription containing the resource group you just created
7. Location: probably East US or East US 2
8. Pricing Tier: select F0 for this lab
9. Resource Group: select the resource group you created in the previous steps
10. Click “Create”

If you open the Notifications (bell icon on the tool bar), you’ll see the service is provisioned and you can click “Go to resource.” There are 2 pieces of information we’ll need in the next step in order to connect to our new Text Analytics service:

1. In the Text Analytics service, Quick View might be selected by default. You can click the “Keys” link from the first step – “Grab your keys”, or click the “Keys” blade in the left pane. Use the button to copy the key to the clipboard and paste it in Notepad or someplace to refer to later.
   1. Key: i.e. c1a274f956af4bbfbfe8aefa3a915a90
2. Click the “Overview” blade in the left pane and use the button to copy the Endpoint to the clipboard. Past the endpoint in Notepad or someplace to refer to later.
   1. Endpoint URL: i.e. <https://eastus>2.api.cognitive.microsoft.com/text/analytics/v2.0

# Create Flow to capture sentiment analysis from Opportunity fields and related Notes

We’re going to create a Flow that will loop through our Opportunity records and calculate the sentiment score for those fields. It will also calculate an average sentiment score for the Note records related to each Opportunity. We’ll update each Opportunity record with these values, which will become the source data for our Machine Learning model.

1. Navigate to <https://Flow.Microsoft.com>
2. Use the same login credentials used for the Dynamics customization earlier
3. Make sure you select the environment that matches your CDS environment from the upper righthand corner of the toolbar
4. Select the “My Flows” folder from the left pane
5. Click “New” on the toolbar and select “Create from blank”. Click “Create from Blank”
6. In the connector search box, type “Schedule”
7. Select the “Recurrence” trigger from the results list.
   1. Interval: 1
   2. Frequency: select “Day”
8. Add a new step – search for “Variable” and select the “Initialize variable” action
   1. Create 5 variables for use later in the Flow (create a new step for each)

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Initial Value** |
| Opportunities | Array |  |
| NoteText | String |  |
| SumOfSentimentScore | Float | 0 |
| Count | Integer | 0 |
| AverageSentimentScore | Float | 0 |

1. Add a new step – search for “Common Data Service” and select the “List records” action
   1. Environment: “(Current)”
   2. Entity Name: “Opportunities”
2. Add a new step – search for “Control” and select the “Apply to Each” action
   1. Select an output from previous steps: Use the Dynamic content list to select “value” from the Opportunities listed in the previous step
3. Add a new step inside the control loop – search for “Text Analytics” and select the “Detect Sentiment” action
   1. Text: from the Dynamic content list, select the “Current Situation” field from the Opportunity record
   2. NOTE: On the first use of the Text Analytics connector, you’ll need to configure the connection to the Text Analytics service you created in Azure by providing the Key and Endpoint URL from earlier.
4. Add another “Detect Sentiment” step inside the control loop
   1. Text: from the Dynamic content list, select the “Customer Need” field from the Opportunity record
5. Add a third “Detect Sentiment” step inside the control loop
   1. Text: from the Dynamic content list, select the “Proposed Solution” field from the Opportunity record
6. Add a new step – search for “variable” and select the “Append to array” action
   1. Name: select the “Opportunities” variable created earlier
   2. Value: paste the following JSON:
      1. {“CurrentSituation”: “”, “CustomerNeed”: “”, “ProposedSolution”: “”, “OpportunityId”: “”}
   3. For the value of “CurrentSituation”, use the Dynamic content list to select the “Score” from the CurrentSituation Detect Sentiment step created earlier
   4. For the value of “CustomerNeed”, use the Dynamic content list to select the “Score” from the CustomerNeed Detect Sentiment step created earlier
   5. For the value of “ProposedSolution”, use the Dynamic content list to select the “Score” from the ProposedSolution Detect Sentiment step created earlier
   6. For the value of “OpportunityId”, use the Dynamic content list to select the “Opportunity” field from the Opportunity record
7. Add a new step – search for “Common Data Service” and select the “List records” action
   1. Environment: “(Current)”
   2. Entity Name: “Notes”
   3. Show Advanced Options
   4. Filter Query (we’re going to filter on the ID of the current opportunity in our loop:
      1. “\_objectid\_value eq “
      2. Use the Dynamics content list to select the “Opportunity” field from the Opportunity record
8. Add a new step – search for “Control” and select the “Apply to Each” action
   1. Select an output from previous steps: Use the Dynamic content list to select “value” from the Notes listed in the previous step
9. On the new loop step, click the three dots (ellipsis) and select “Settings” from the menu
   1. Turn on concurrency control
   2. Set Degree of Parallelism to 1
   3. Click “Done”
10. Add a new step inside the Notes loop – search for “variable” and select the “Set variable” action
    1. Name: select the “NoteText” variable created earlier
    2. Value: use the Dynamic content list to select the “Description” field from the Note record
11. Add a new step inside the Notes loop – search for “Text Analytics” and select the “Detect Sentiment” action
    1. Text: from the Dynamic content list, select the “NoteText” variable from the variables list
12. Add a new step inside the Notes loop – search for “variable” and select the “Increment variable” action
    1. Name: select the “SumOfSentimentScore” variable created earlier
    2. Value: use the Dynamic content list to select the “Score” field from the previous “NoteText” Detect Sentiment step
13. Add a new step inside the Notes loop – search for “variable” and select the “Increment variable” action
    1. Name: select the “Count” variable created earlier
    2. Value: 1
14. Add a new step outside the Notes loop, but still inside the Opportunities loop – search for “Control” and select the “Condition” action
    1. Value: use the Dynamic content list to select the “Count” variable from the list of variables
    2. Operator: select “is greater than”
    3. Value: 0
15. Add a new step inside the “yes” branch – search for “variable” and select the “Set variable”
    1. Name: select the “AverageSentimentScore” variable created earlier
    2. Value: set this expression: “div(variables(‘SumOfSentimentScore’), variables(‘Count’))”
16. Add a new step inside the “yes” branch – search for “variable” and select the “Set variable”
    1. Name: select the “Count” variable created earlier
    2. Value: 0
17. Add a new step inside the “yes” branch – search for “Common Data Service” and select the “Update record” action
    1. Environment: “(Current)”
    2. Entity Name: “Opportunities”
    3. Record identifier: use the Dynamic content list to select the “Opportunity” field from the Opportunity record
    4. Current Situation Sentiment Score: use the Dynamic content list to select the “Score” from the CurrentSituation Detect Sentiment step created earlier
    5. Customer Need Sentiment Score: use the Dynamic content list to select the “Score” from the CustomerNeed Detect Sentiment step created earlier
    6. Notes Average Sentiment Analysis: use the Dynamic content list to select the “AverageSentimentScore” variable from the list of variables
    7. Proposed Solution Sentiment Score: use the Dynamic content list to select the “Score” from the ProposedSolution Detect Sentiment step created earlier
18. Add a new step inside the “yes” branch – search for “variable” and select the “Set variable”
    1. Name: select the “SumOfSentimentScore” variable created earlier
    2. Value: 0
19. Add a new step inside the “**no**” branch – search for “Common Data Service” and select the “Update record” action
    1. Environment: “(Current)”
    2. Entity Name: “Opportunities”
    3. Record identifier: use the Dynamic content list to select the “Opportunity” field from the Opportunity record
    4. Current Situation Sentiment Score: use the Dynamic content list to select the “Score” from the CurrentSituation Detect Sentiment step created earlier
    5. Customer Need Sentiment Score: use the Dynamic content list to select the “Score” from the CustomerNeed Detect Sentiment step created earlier
    6. Proposed Solution Sentiment Score: use the Dynamic content list to select the “Score” from the ProposedSolution Detect Sentiment step created earlier
20. Update the title of the Flow in the upper left corner of the toolbar
    1. For example: SentimentFromOpportunityAndNotes
21. Save the Flow

# Machine Learning Studio Workspace setup in Azure

We’re going to create and Azure Machine Learning Studio Workspace where we can configure a custom machine learning model to predict the outcome of an Opportunity based on the sentiment scores that we calculated and updated in our previous Flow.

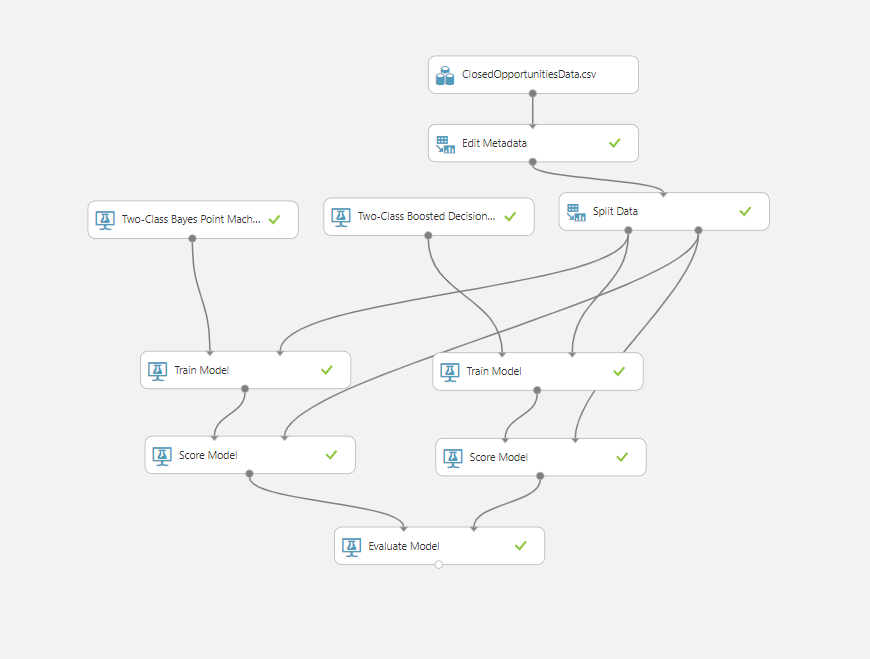
1. Navigate to <https://portal.azure.com> and drill into the resource group created earlier for the Text Analytics service.
2. Click “Add” on the toolbar
3. Type “machine learning” in the search bar and select “Machine Learning Studio Workspace” from the results list
4. Click “Create” after reviewing the details for the Machine Learning Studio
5. Name: give your service a meaningful name, i.e. MLWorkspaceLab
6. Subscription: select the subscription containing the resource group created earlier
7. Resource Group: select the resource group you created in the previous steps
8. Location: probably South Central US
9. Storage Account: select “Create new” and accept the default name
10. Workspace pricing tier: select “Standard”
11. Web service plane: select “Create new” and accept the default name
12. Click “Create”

# Machine Learning experiment creation in ML Studio

We’ll create a Machine Learning experiment in Azure and train it using sample data. We are going to predict if we “win” or “lose” the opportunity based on the available feature set, including the sentiment scores.

Open your AML Studio Workspace (studio.azureml.net).

1. Upload ClosedOpportunitiesData.csv as a “New Dataset”. You can download the file here: <https://davewdemoblobs.blob.core.windows.net/amls-flow-workshop/ClosedOpportunitiesData.csv>
2. Create a new Experiment (blank).
3. Add the Training dataset
4. Add an “Edit Metadata” to make the following features “categorical”
   1. Potential Customer
   2. Account
   3. Decide Go/No-Go
   4. Rating
   5. Status
5. Run and “Visualize” the experiment to ensure the output is correct.
6. “Split Data” 70/30 into training and validation datasets
7. We want to compare 2 different binary classifiers to predict the “status”. Add the necessary components so your workspace looks like the screenshot. Helpful hints:
   1. The “model” is always the left side input
   2. The “data is always the right side input
      1. For Train Model the data is from the 70% training data
      2. For Score Model the data is from the 30% validation data



Which algorithm did better? Which do you want to use going forward?

Now we want to use this model to perform “inferencing”. Inferencing is where we use a model in conjunction with incoming data to predict the “status”.

1. Click Run to run the model end-to-end
2. Select the Train Model option that you want to use going forward.
3. Then click Setup Web Service | Predictive Web Service
4. Watch the screen animation carefully. Note that all this does is removes the static data set and allows data to be input via a web service and removes the extra model information we don’t need.
5. Click save
6. Click run to test it
7. Click Deploy Web Service (classic)
8. Note the following for later:
   1. API Key:
      1. Example: sQ7kGZkbusogb+XAtzdzDsQC97pGYOvMKJtsrMGyy5p4TtUacNxXuPW4udGfom6VjFWKyHoJLKTN+z175+2zfA==
   2. Request/ResponseURI:
      1. <https://ussouthcentral.services.azureml.net/workspaces/b2b3a698dfd547e58677cdff82fea33f/services/7484bfd938da4f259630dde9c1aa8c3c/execute?api-version=2.0&details=true>
   3. Batch URI
      1. <https://ussouthcentral.services.azureml.net/workspaces/b2b3a698dfd547e58677cdff82fea33f/services/7484bfd938da4f259630dde9c1aa8c3c/jobs?api-version=2.0>

# Create Flow to update open Opportunities with their prediction from ML model

We’ll create a Flow to run weekly for all open Opportunities and update the prediction using the ML model.

1. Navigate to <https://Flow.Microsoft.com>
2. Use the same login credentials used for the Dynamics customization earlier
3. Make sure you select the environment that matches your CDS environment from the upper righthand corner of the toolbar
4. Select the “My Flows” folder from the left pane
5. Click “New” on the toolbar and select “Create from blank”. Click “Create from Blank”
6. In the connector search box, type “Schedule”
7. Select the “Recurrence” trigger from the results list.
   1. Interval: 1
   2. Frequency: select “Day”
8. Add a new step – search for “Variable” and select the “Initialize variable” action
   1. Name: MLOutcome
   2. Type: String
9. Add a new step – search for “Common Data Service” and select the “List records” action
   1. Environment: “(Current)”
   2. Entity Name: “Opportunities”
   3. Expand “Advanced Options”
      1. Set “Filter Query” to following Odata query:

statecode eq 0 and name ne null and actualvalue ne null and budgetamount ne null and new\_currentsituationsentimentscore ne null and new\_proposedsolutionsentimentscore ne null and new\_notesaveragesentimentanalysisscore ne null and new\_customerneedsentimentscore ne null and opportunityratingcode ne null

1. Add a new step – search for “Control” and select the “Apply to Each” action
   1. Select an output from previous steps: Use the Dynamic content list to select “value” from the Opportunities listed in the previous step
2. On the new Apply to Each loop step, click the three dots (ellipsis) and select “Settings” from the menu
   1. Turn on concurrency control
   2. Set Degree of Parallelism to 1
   3. Click “Done”
3. Add a new step inside the loop – search for “HTTP Request” and select “HTTP”
   1. Method: select POST
   2. URI: enter the URL for the ML web service created in the previous section
      1. <https://ussouthcentral.services.azureml.net/workspaces/eab59cc90bbf4457a7f378ace969c6ea/services/296686da1fd3445fb41b526617aa752f/execute?api-version=2.0&details=true>
   3. Headers: add the following 3 headers as key/value pairs:
      1. Accept : application/json
      2. Authorization : Bearer <*your ML web service API key from previous section*>
         1. Bearer eTrmT4e7qlfcYsSvixQmLqearwdh6/tYg8ija3hPNA+Z62QUu3MRo7ZUcztm6ZpXsq3lKh3tqTouSn0tLaru0A==
      3. Content-Type : application/json
   4. Body:
      1. {"GlobalParameters": {}, "Inputs": { "input1": { "ColumnNames": [ "Topic", "Potential Customer", "Account", "Est. Revenue", "Actual Revenue", "Budget Amount", "Current Situation Sentiment Score", "Customer Need Sentiment Score", "Proposed Solution Sentiment Score", "Notes Sentiment Score", "Decide Go/No-Go", "Rating", "Status" ], "Values": [ [ "", "", "", "", "", "", "", "", "", "", "", "", "" ] ] } } }
      2. In the “Values” array, use the Dynamic content list to fill in the fields in the same order as the “ColumnNames” list.
4. Add a new step inside the loop – search for “Data Operations” and select “Parse JSON”
   1. Content: Use the Dynamic content list to select the “Body” output from the previous step
   2. Schema: use the following JSON:

{"Results": {"output1": {

"type": "table",

"value": {

"ColumnNames": [

"Topic",

"Potential Customer",

"Account",

"Est. Revenue",

"Actual Revenue",

"Budget Amount",

"Current Situation Sentiment Score",

"Customer Need Sentiment Score",

"Proposed Solution Sentiment Score",

"Notes Sentiment Score",

"Decide Go/No-Go",

"Rating",

"Status",

"Scored Labels",

"Scored Probabilities"

],

"ColumnTypes": [

"String",

"String",

"String",

"Int32",

"Int32",

"Int32",

"Double",

"Double",

"Double",

"Double",

"String",

"String",

"String",

"String",

"Double"

],

"Values": [

[

"value",

"value",

"value",

"0",

"0",

"0",

"0",

"0",

"0",

"0",

"value",

"value",

"value",

"Lost",

"0.102679125964642"

]

]

}

}

}

}

1. Add a new step inside the loop – search for “variable” and select the “Set variable”
   1. Name: select the “MLOutcome” variable created earlier
   2. Value: set this expression: “body('Parse\_JSON')?['Results']?['output1']?['value']?['Values'][0][13]”
2. Add a new step inside the loop – search for “Common Data Service” and select the “Update record” action
   1. Environment: “(Current)”
   2. Entity Name: “Opportunities”
   3. Record identifier: use the Dynamic content list to select the “Opportunity” field from the Opportunity record
   4. ML Outcome Prediction: use the Dynamic content list to select the “MLOutcome” variable