
Singapore Tourism Board

– Workshop: Survey Analysis



MARCH 3

IBM

Authored by: ANINDYO SARKAR

Use Case Background

Tourist Feedback Analysis

Singapore Tourism Board receives survey / feedback results from tourists for various attractions in Singapore. The intent of this exercise is to analyze the survey results and get meaningful insights from the feedback data that can be used to positively influence the tourist footfall. The feedback is a free form text entered by tourists at different attractions. For our scope we are using sample feedback from Gardens by the Bay. STB has identified the following entities on which feedback needs to be analyzed:

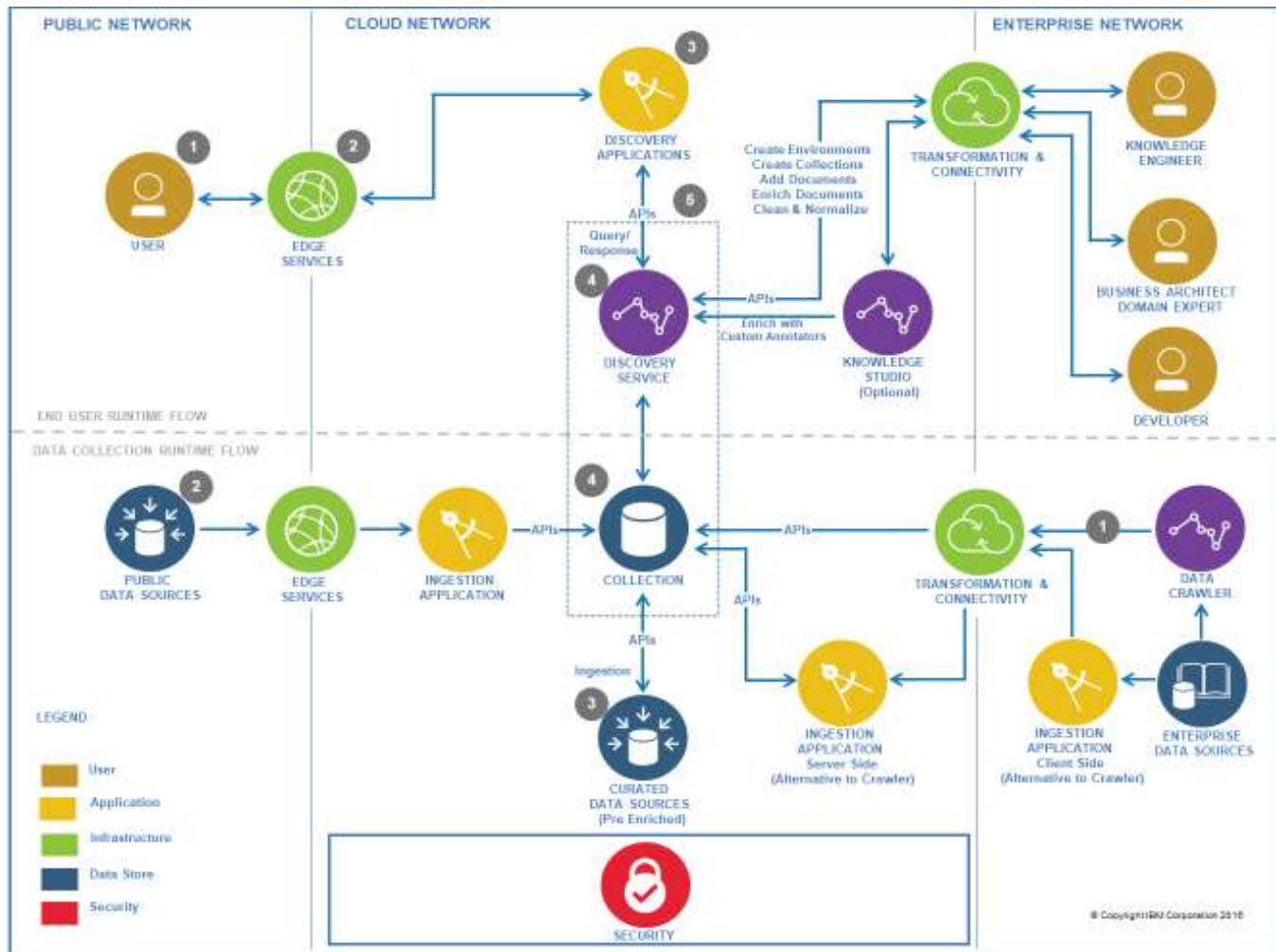
- **Sightseeing:** View of the place, aesthetics, scenery, ambience etc.
- **Transportation:** Ease of getting there, transportation details, transport options etc.
- **Service Quality:** Feedback regarding human services/quality/timeliness etc.
- **F&B:** Cuisine, culinary or dietary requirements, range, hygiene etc.
- **Shopping / Retail:** Adequacy, range, variations, relevance etc.
- **Information accessibility:** Info availability, opening hours, information accessibility
- **Facilities:** Washroom, aircon, wheelchair accessibility, lifts, escalators, drinking water
- **Cultural shows:** Events, performances and tours
- **Price:** Affordability, value of money, ticket prices

The idea of the exercise is to analyze feedback on the above entities and collect useful insights which can be used for future planning.

To perform this exercise, we shall leverage IBM Watson capabilities. IBM Watson has powerful text analytics capabilities such as:

- **Natural language** understanding for building AI models to annotate plain text
- **Tone Analyzer** to understand tone of a sentence
- **Natural Language Classifier** to classify text into pre-determined groups
- **Watson Knowledge Studio** to build custom annotation models
- **Watson Discovery** to search for useful insights from plain text document corpus
- **Watson Studio Local** to build custom models using Python/R/Scala etc.

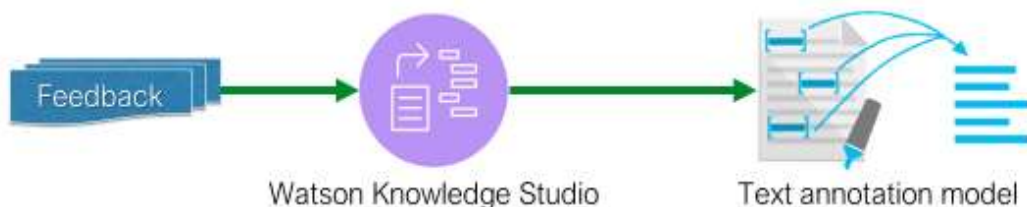
Solution Architecture



Workshop Hands-on scope

Building the application to orchestrate the overall workflow and provide the user interfaces is outside the scope of this workshop. This workshop will focus on the underlying AI capabilities that the application will use to perform its job. Specifically, we shall focus on two areas as shown below:

Build Annotation Model



Identify Entities of interest using annotation model & extract sentiment using tone analyzer



Provisioning services in IBM Cloud

Follow the link below to provision services in IBM Cloud. Look at “**Before you begin**” section under each service. Just provision the services at this point. No need to complete the entire tutorial in the link:

Knowledge Studio:

https://cloud.ibm.com/docs/services/watson-knowledge-studio?topic=watson-knowledge-studio-wks_tutintro#instance

Natural Language Understanding:

<https://cloud.ibm.com/docs/services/natural-language-understanding?topic=natural-language-understanding-getting-started#before-you-begin>

Tone Analyzer: <https://cloud.ibm.com/docs/services/tone-analyzer?topic=tone-analyzer-gettingStarted#prerequisites>

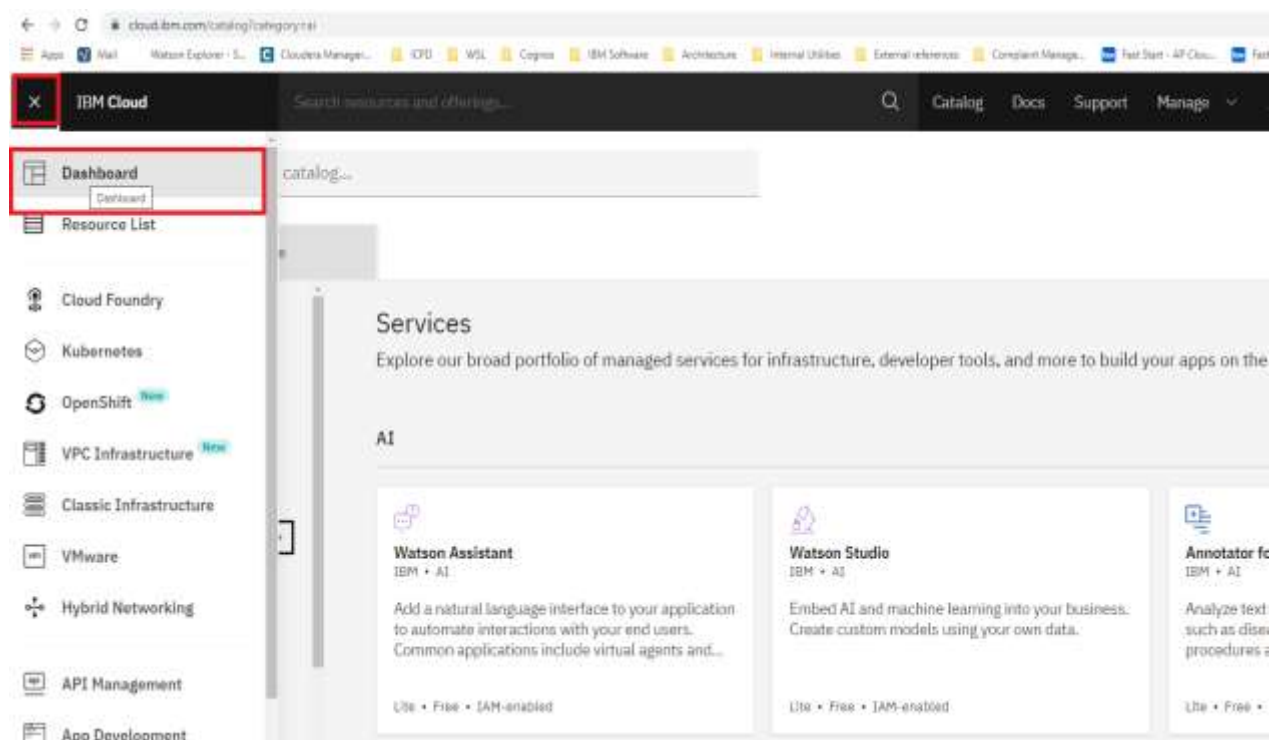
Natural Language Classifier: <https://cloud.ibm.com/docs/services/natural-language-classifier?topic=natural-language-classifier-natural-language-classifier#prerequisites>

If you need a Python runtime then you can also subscribe to Watson Studio and use Notebook IDE with a Python runtime. Once you have provisioned the services, you can go to your Dashboard and see the services you have subscribed to.

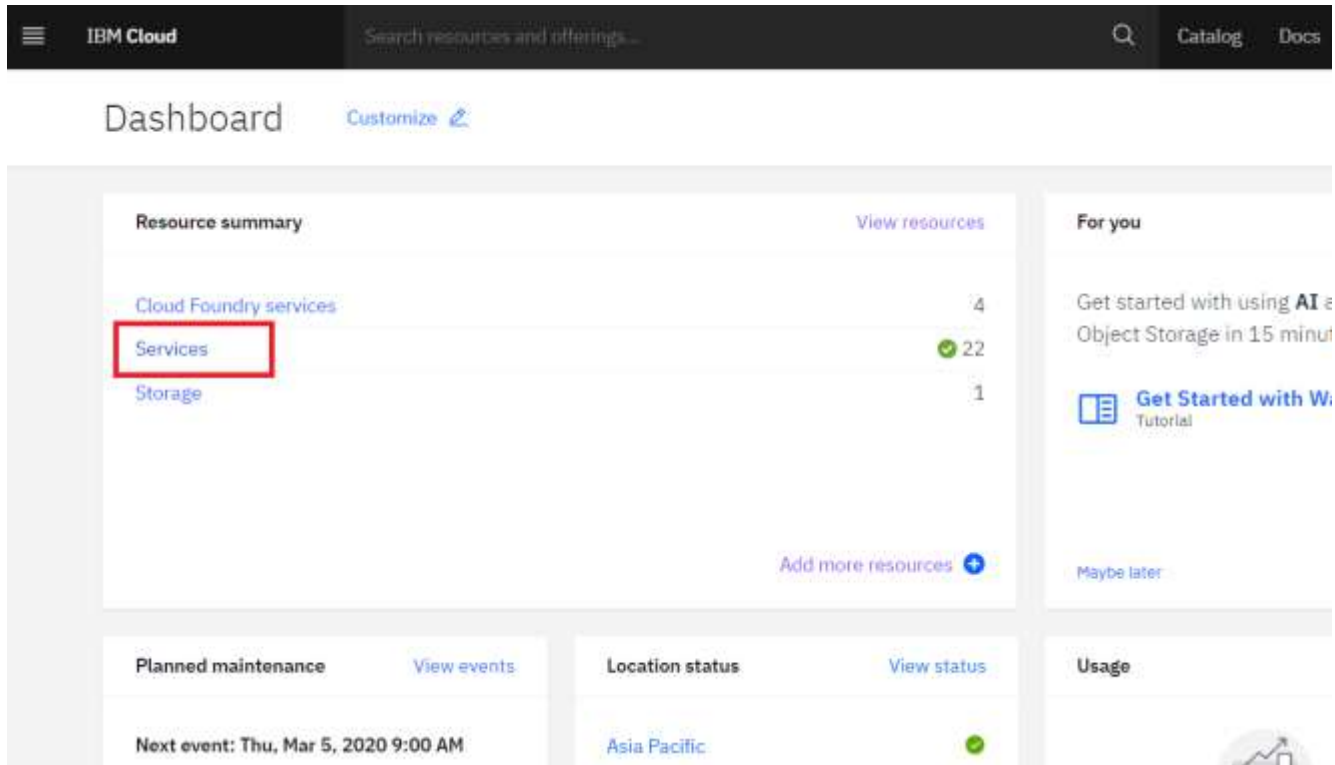
Build annotator model using WKC

1. Launch Watson Knowledge Studio

Go to cloud.ibm.com and click on **Dashboard** from the top left ribbon



The Dashboard shows the list of **Services**.



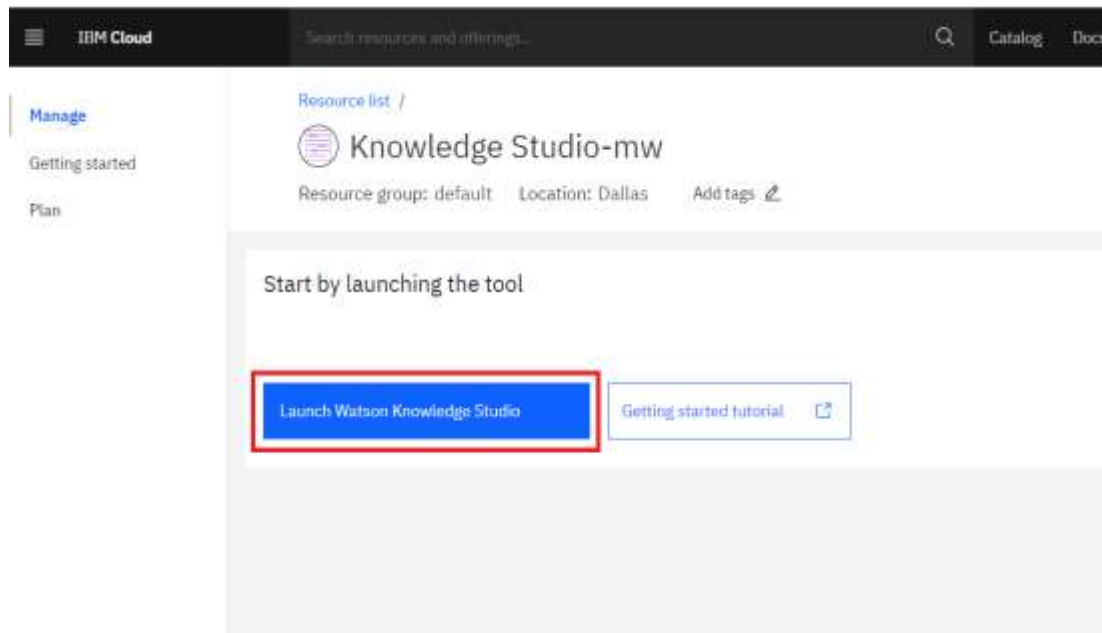
Click on **Services** to see your subscriptions and then click on your **Knowledge-Studio-xx** instance

Resource list

Name	Group	Location	Offering	Status
Q: Filter by name or IP address...	Filter by group or org...	Filter...	Q: Filter...	Q: Filter...
Clusters (0)				
Cloud Foundry apps (0)				
Cloud Foundry services (4)				
Services (22)				
Analytics Engine-xj	default	Dallas	Analytics Engine	Active
Annotator for Clinical Data-51	default	Dallas	Annotator for Clinical Data	Active
Annotator for Clinical Data-ty	default	Dallas	Annotator for Clinical Data	Active
Compare and Comply-2d	default	Dallas	Compare and Comply	Active
Compare and Comply-00	default	Dallas	Compare and Comply	Active
Discovery-kb	default	Dallas	Discovery	Active
IBM Cognos Dashboard Embedded-16	default	Dallas	IBM Cognos Dashboard Emb...	Active
Knowledge Studio-3f	default	Dallas	Knowledge Studio	Active
Knowledge Studio-mw	default	Dallas	Knowledge Studio	Active

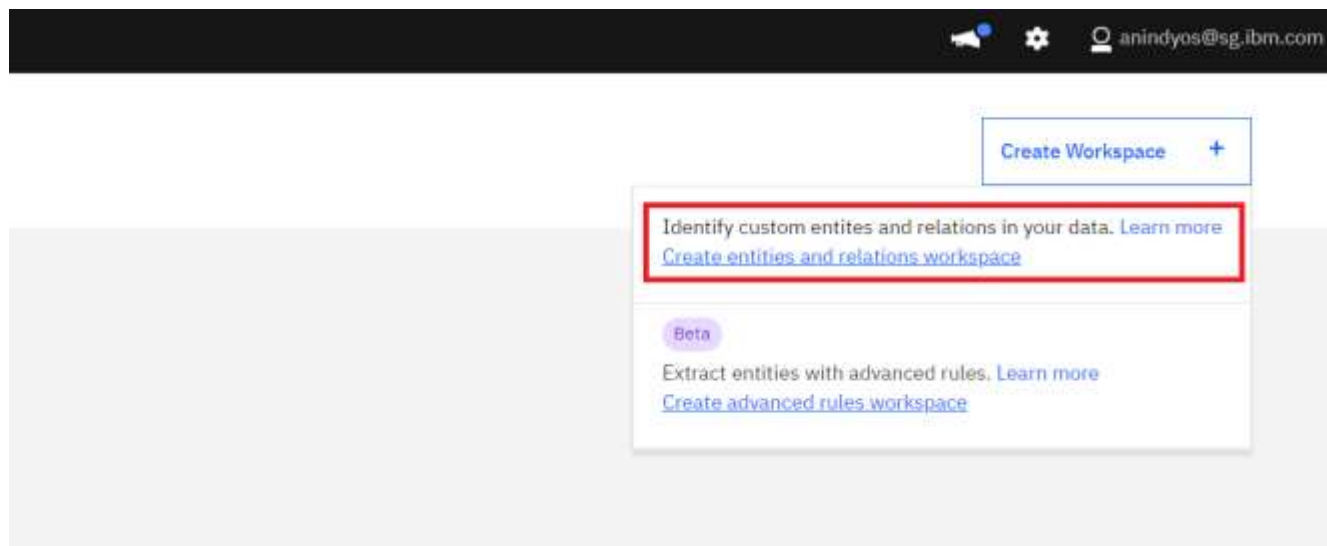
Knowledge Studio-mw

Knowledge Studio launch page opens. Click on **Launch Watson Knowledge Studio** to open Knowledge-Studio.

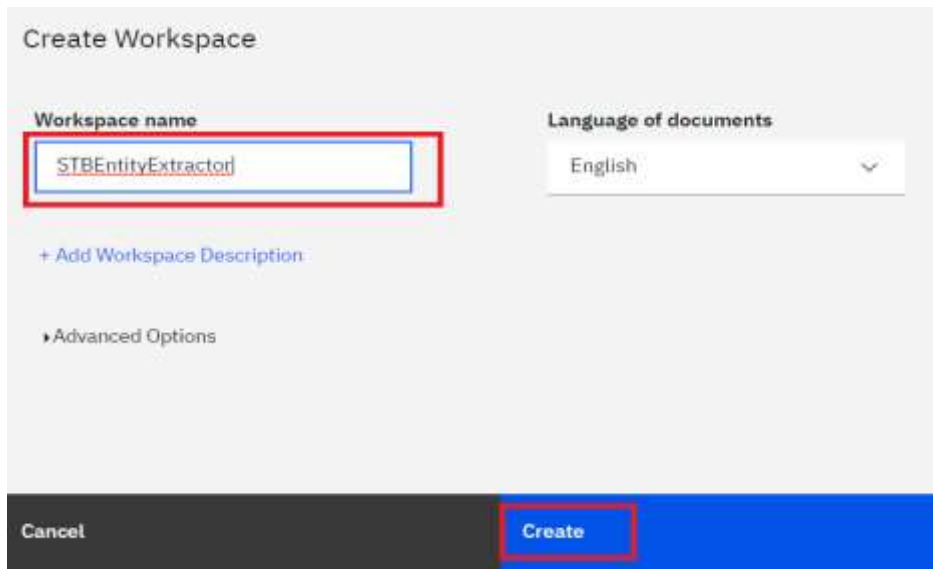


2. Create Workspace

Create a workspace by clicking on **Create Workspace** on top right and choosing the top option of Create entities and relations workspace.



Give it a proper name like STBEntityExtractor.



Create Workspace

Workspace name
STBEntityExtractor

Language of documents
English

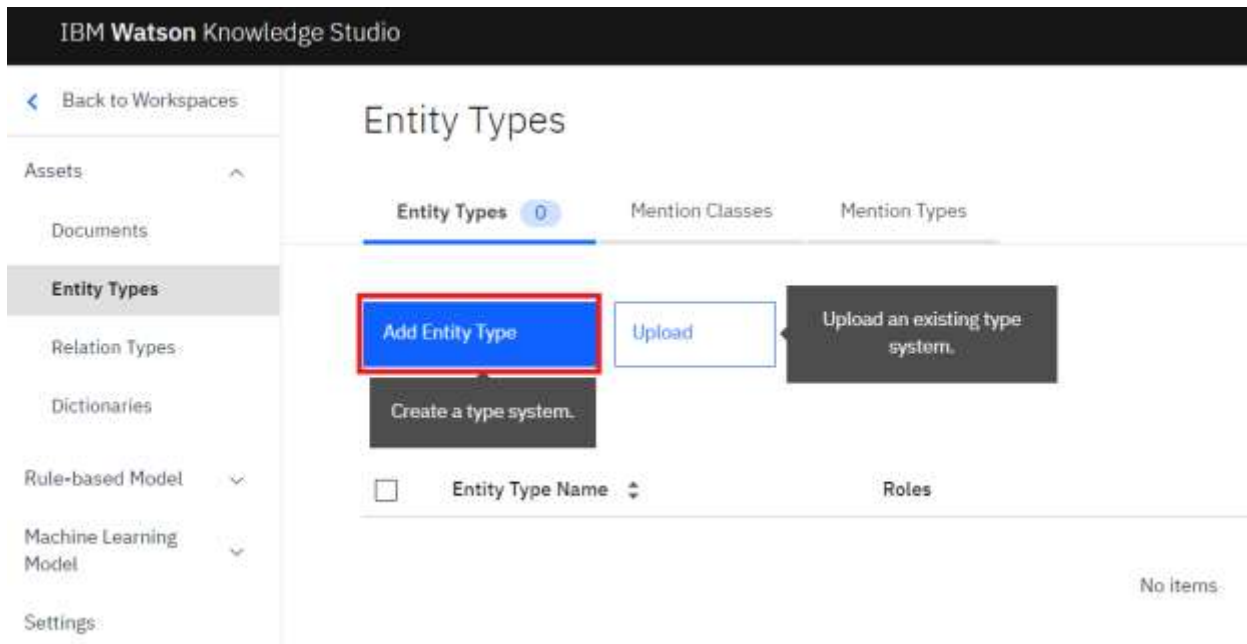
+ Add Workspace Description

Advanced Options

Cancel Create

3. Create Entity Types

To create Entities, click on **Add Entity Type**



IBM Watson Knowledge Studio

Back to Workspaces

Assets

- Documents
- Entity Types**
- Relation Types
- Dictionaries

Rule-based Model

Machine Learning Model

Settings

Entity Types

Entity Types 0 Mention Classes Mention Types

Add Entity Type Upload

Create a type system. Upload an existing type system.

☐ Entity Type Name Roles

No items

Enter the entity type value and click on **Save**

Entity Types

Entity Types 0 Mention Classes Mention Types

Add Entity Type Delete Entity Types Upload Download Types

Enter text to filter

Entity Type Name	Roles	Subtypes	Action
<input checked="" type="checkbox"/> Transportation	<input type="text" value="Select a role"/> Add	<input type="text" value="Enter a subtype"/> Add	<input type="button" value="Cancel"/> <input checked="" type="button" value="Save"/>

Transportation (This type)

Repeat this operation for all entity types

IBM Watson Knowledge Studio

Back to Workspaces

Assets

- Documents
- Entity Types
- Relation Types
- Dictionaries

Rule-based Model

Machine Learning Model

Settings

Help

Entity Types

Entity Types 0 Mention Classes Mention Types

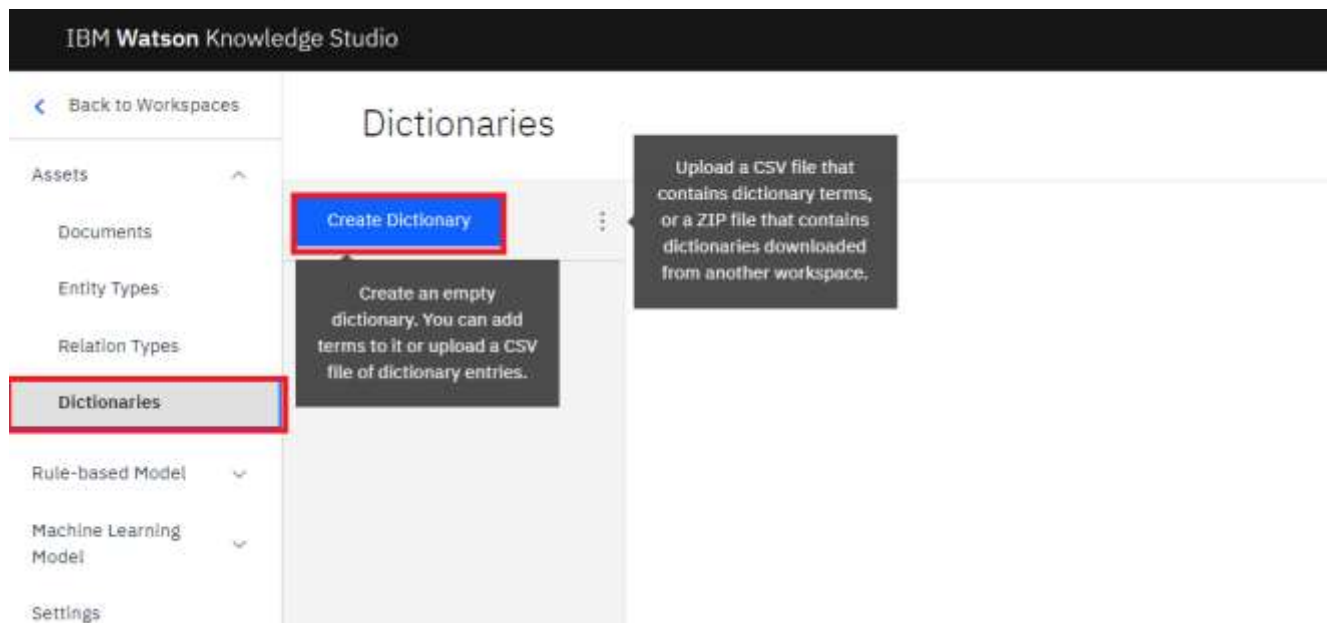
Add Entity Type Delete Entity Types Upload

Entity Type Name	Roles	Subtypes
<input checked="" type="checkbox"/> Price	Price	
<input type="checkbox"/> Shows	Shows	
<input type="checkbox"/> Facilities	Facilities	
<input type="checkbox"/> Accessibility	Accessibility	
<input type="checkbox"/> Retail	Retail	
<input type="checkbox"/> FnB	FnB	
<input type="checkbox"/> ServiceQuality	ServiceQuality	
<input type="checkbox"/> Sightseeing	Sightseeing	
<input type="checkbox"/> Transportation	Transportation	

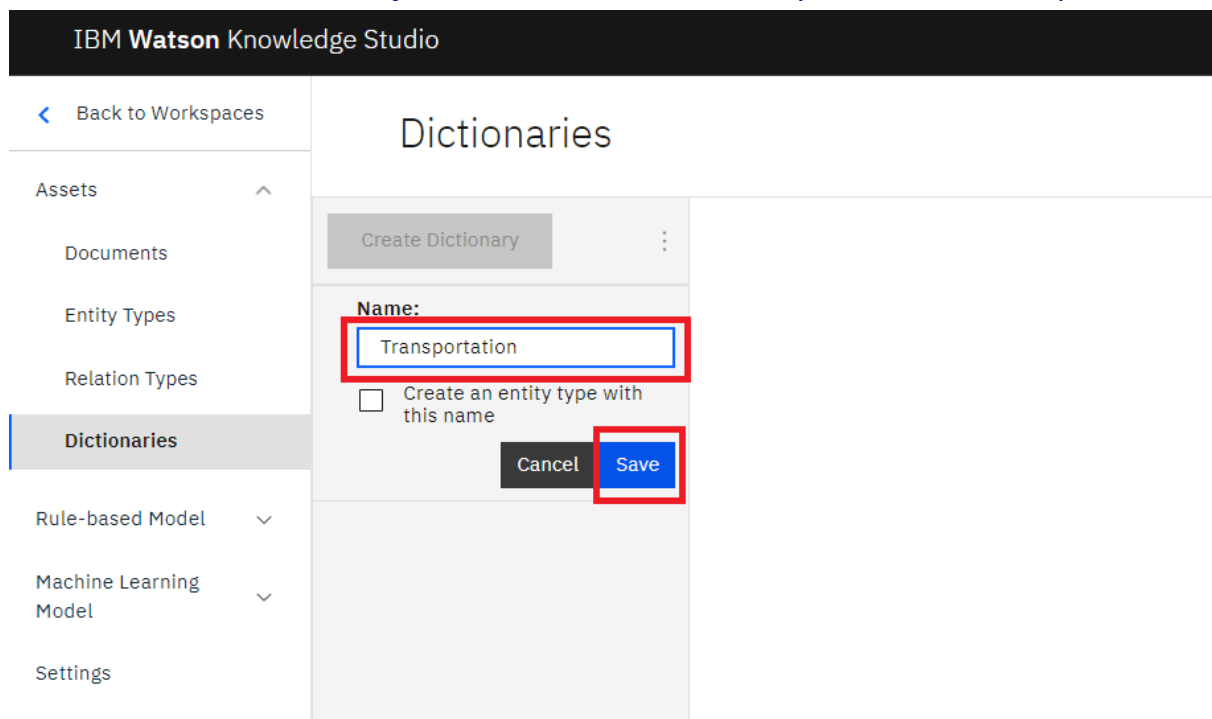
4. Create Dictionaries

Though not mandatory, to help your model to accurately identify your entities you may define dictionaries and associate them with your entities. This helps when we are focusing on a narrow or finite set of values and using those values to identify the entity, which is mostly the case in this scenario. Hence we shall define dictionary for most of our entities.

Click on **Dictionary** and **Create Dictionary**



Click on **Create Dictionary** to create a new dictionary. Name it “Transportation” and **Save** it.



Associate Entity Type “Transportation” to this Dictionary. Create Lemma forms example MRT and fill up Surface Forms as shown below. Click Save to save the entry.

Create Dictionary

Transportation

1

Transportation

Language: English | 1 entry

Entity type:

Transportation

Rule class:

None

Add Entry

Delete Entries

Upload

Download

Enter text to filter

Lemma

MRT

Surface Forms

MRT

subway

metro

tube

tunnel

train

Enter text

Part of Speech

Noun

Action

Cancel

Save

Repeat the above step to create other Dictionary items under “Transportation” Dictionary as shown below:

Transportation

Language: English | 6 entries

Entity type:

Transportation

Rule class:

None

Add Entry

Upload

Download

Entry suggestions are unavailable

Open Suggestions

Enter text to filter

Lemma	Surface Forms	Part of Speech	Action
Taxi	Taxi, cab, Grab, Uber, Gojek, Transcab, Grab share, Comfort cab, Comfort taxi	Noun	Edit Delete
bus	bus, shuttle	Noun	Edit Delete
Flight	Flight, by air, plane, aeroplane, airbus	Noun	Edit Delete
MRT	MRT, subway, metro, tube, tunnel, train	Noun	Edit Delete
Ferry	Ferry	Noun	Edit Delete
walk	walk, stroll, jog, climb	Verb	Edit Delete

Similarly create Dictionary for the remaining entities

13

Sample dictionaries created for some of the above entities are attached herewith for reference.



FnB_158321174885
9.csv



Shows_1583211328
452.csv



Facilities_15832112
45299.csv



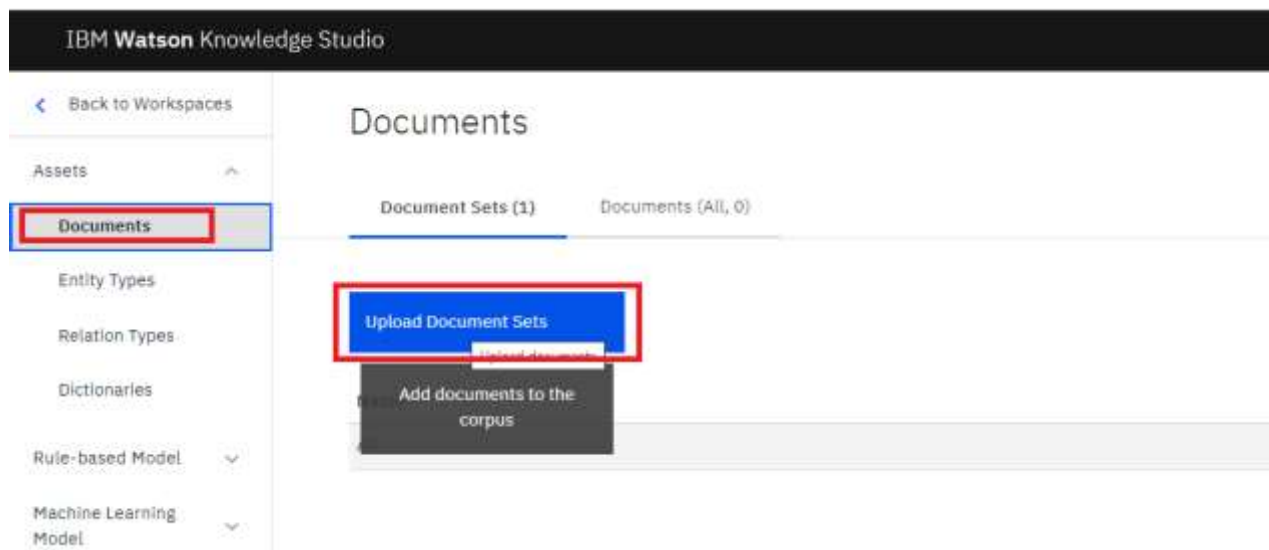
Retail_15832108785
63.csv

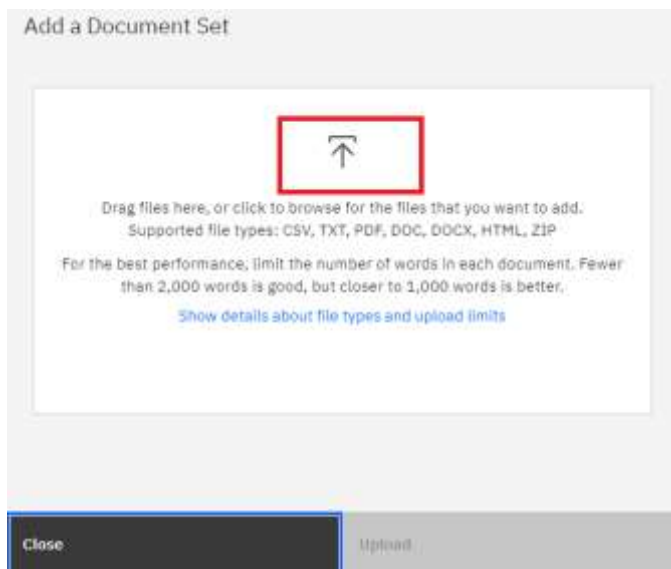


Price_158321275788
8.csv

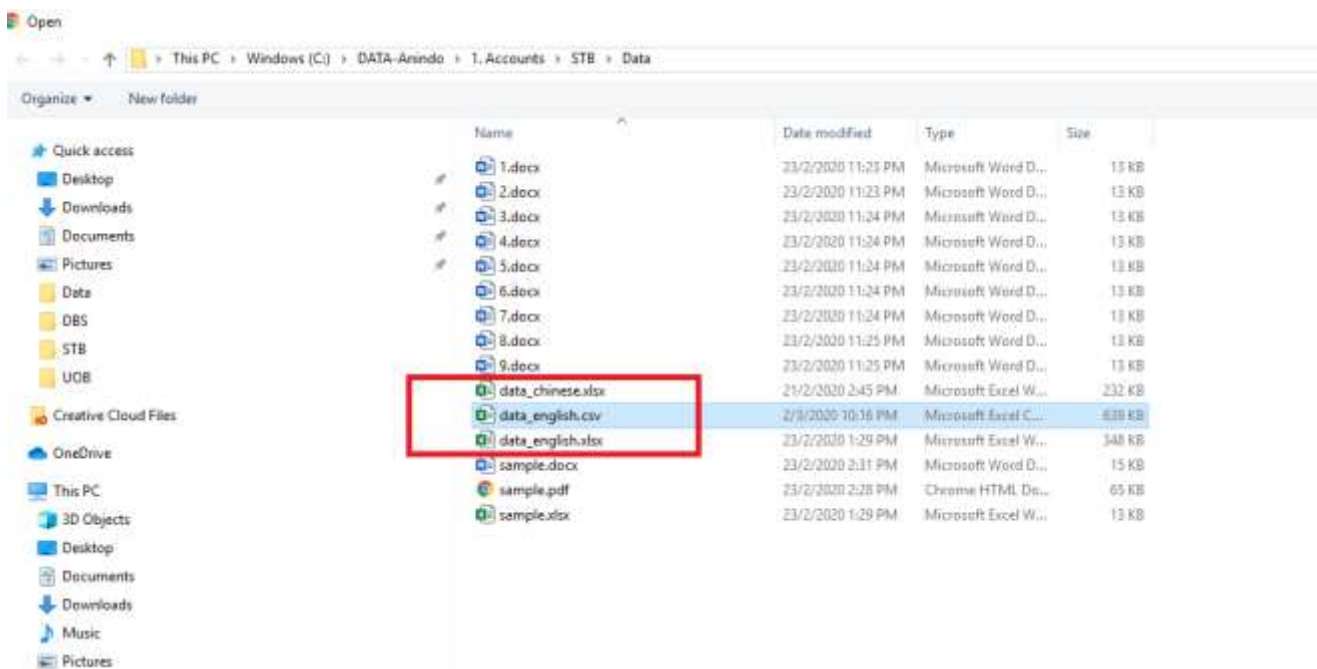
5. Upload Training data set

After all dictionaries have been created, upload the training data set as a csv file. Click on **Documents** and **Upload Document Sets** to upload

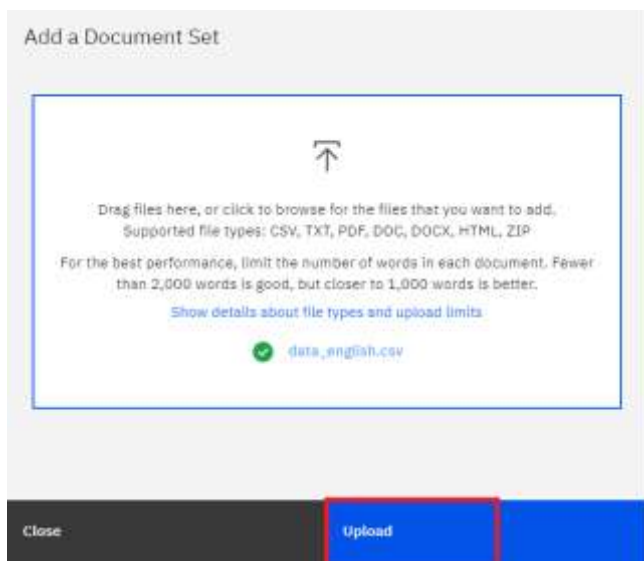




Select the csv and open it.



Click on **Upload**. Wait till upload completes. There are resource limitations in the free account. Hence use a small sample as this workshop is for functionality illustration only. You may not be able to upload large number of files/documents/rows. Select first 30 rows of the document and create a subset csv file that we shall use for this exercise.



Verify that document set is uploaded properly.

Documents

Document Sets (2) Documents (All, 1963)

Upload Document Sets To begin annotating documents, go to [Annotations](#) page. Download Document Sets

Name	Documents	Last Modified	Action
All	1963	-	
data_english.csv	1963	03/03/2020	Rename Delete

6. Annotate

First we would pre-annotate the documents based on the dictionaries that we have defined. To do that go to **Machine Learning Model -> Pre-annotate**

Back to Workspaces

Assets

Documents

Entity Types

Relation Types

Dictionaries

Rule-based Model

Machine Learning Model

Pre-annotation

Annotations

Documents

Document Sets (2) Documents (All, 1963)

[Upload Document Sets](#)

To begin annotating documents, go to the [Documents](#) page.

Name
All
data_english.csv

And apply the pre annotator

IBM Watson Knowledge Studio

Back to Workspaces

Assets

Documents

Entity Types

Relation Types

Dictionaries

Rule-based Model

Machine Learning Model

Pre-annotation

Pre-annotation

Dictionaries Natural Language Understanding

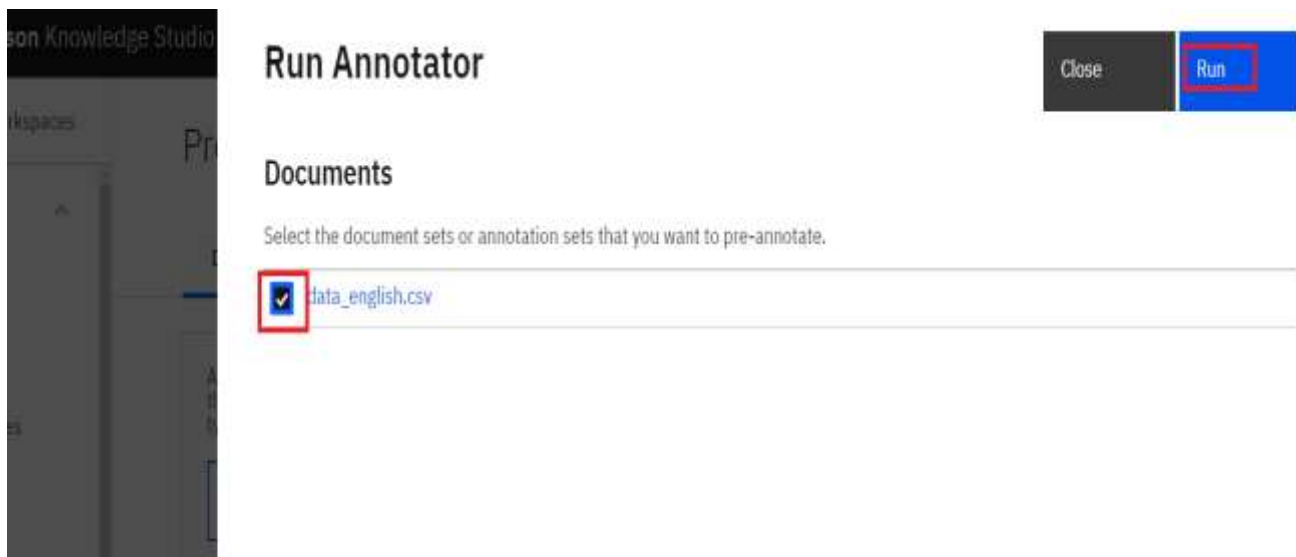
A dictionary annotator creates an annotation for each mention in the text that matches a term in a dictionary. Dictionaries must be mapped to a type or class in order to apply an annotation.

[Apply This Pre-annotator](#)

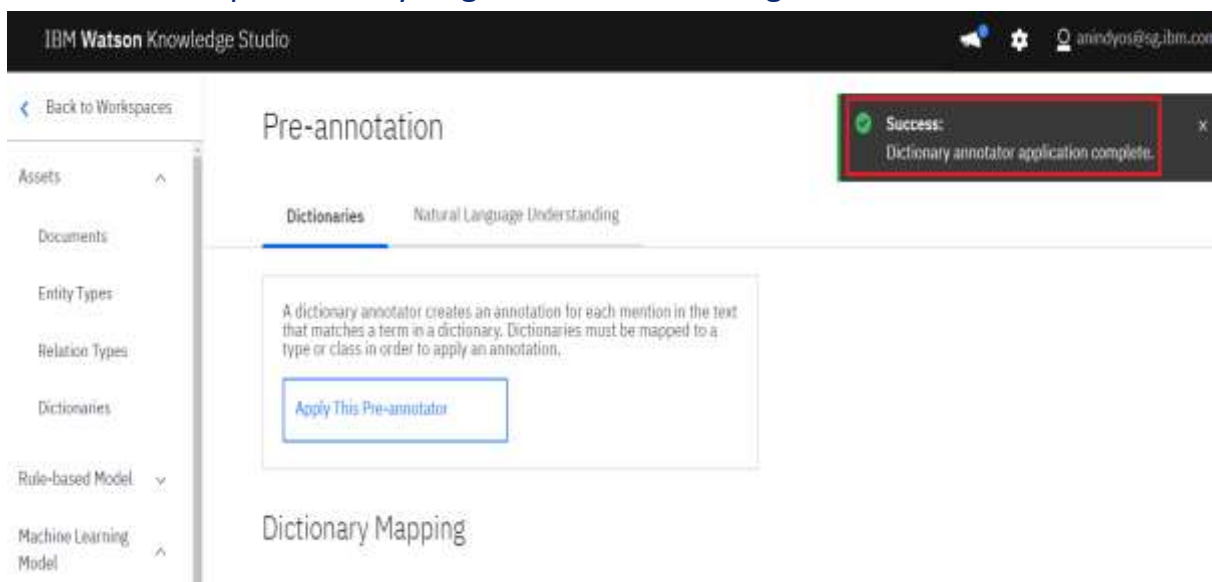
Dictionary Mapping

Manage dictionaries from the [Dictionaries](#) page.

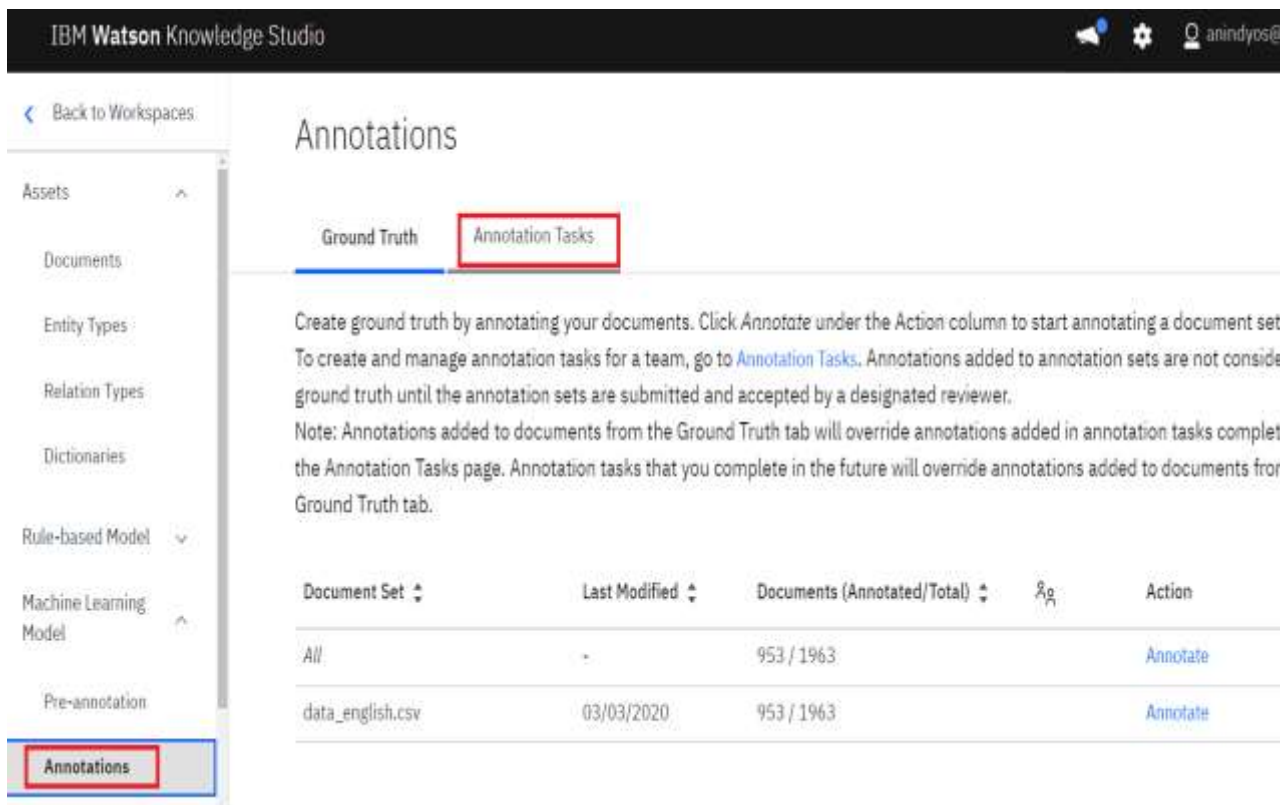
Choose the document set and click on **Run**



Wait till it completes and you get a **Success** message.



Once this automated pre-annotation is done we shall perform some manual annotation. Click on **Annotations**. You can see the Ground truth. The ground truth shows the training set with dictionary-based annotator applied, without any manual annotation. To perform manual annotation, we need to setup an **Annotation Task**.



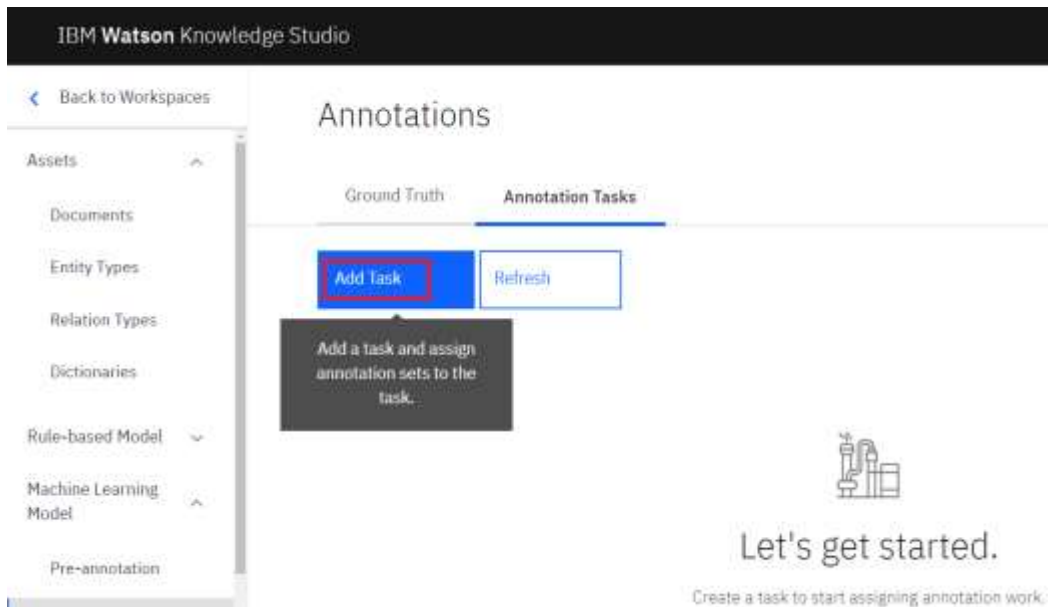
The screenshot shows the IBM Watson Knowledge Studio interface. The top header is dark with the text 'IBM Watson Knowledge Studio' and a user profile 'anindya@'. On the left is a sidebar with a 'Back to Workspaces' link and a list of assets: Documents, Entity Types, Relation Types, Dictionaries, Rule-based Model, Machine Learning Model, and Pre-annotation. The 'Annotations' asset is highlighted with a red box. The main content area is titled 'Annotations' and has two tabs: 'Ground Truth' and 'Annotation Tasks', with the latter highlighted by a red box. Below the tabs is instructional text about creating ground truth and managing annotation tasks. At the bottom is a table with columns for Document Set, Last Modified, Documents (Annotated/Total), and Action.

Document Set	Last Modified	Documents (Annotated/Total)	Action
All	-	953 / 1963	Annotate
data_english.csv	03/03/2020	953 / 1963	Annotate

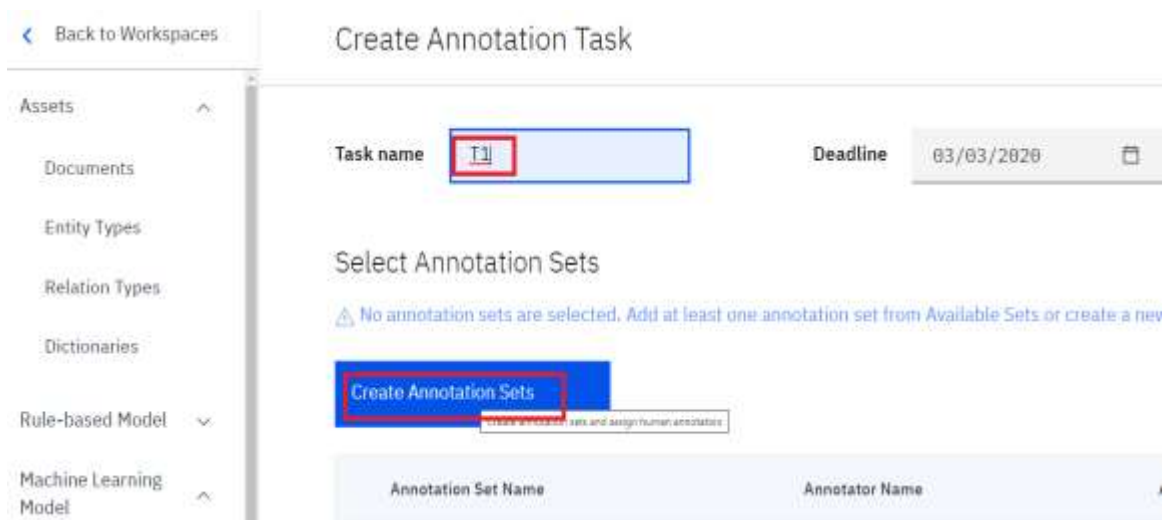
Annotation tasks can be performed in a group, distributed to different team members with some overlapping content. The purpose of the overlap is to ensure that human error is identified, reviewed and corrected. Any document that is annotated differently by different annotators can be flagged by system as a discrepancy and can be sent for review. Once review is completed, it can be **Accepted** or **Rejected** and sent back to the team. All the annotation tasks need to be **Accepted** before it can be submitted for training.

For our scenario, we shall use a single user annotator instead of team annotation. We shall create one Task and assign it to one user. The user shall complete all the annotation by himself and submit for review.

To create Annotation Task click on **Annotation Task** under **Annotations** and then on **Add Task**.



Give the Task a name and **Create Annotation Set**



Fill in the details, specify an annotator (you can use your name) and give the Set a name and click on **Generate**.

Create Annotation Sets

Base set All ▼

Overlap 100 % of 1963 documents

New Sets [+ Add another set and human annotator](#)

To view inter-annotator agreement scores, assign at least two human annotators and ensure that a percentage of documents overlap between the sets.

Annotator Anindyo ▼

Set name S1

Cancel **Generate**

The annotation set gets created. You can **Save** the annotation task now.

Create Annotation Task

Cancel **Save**

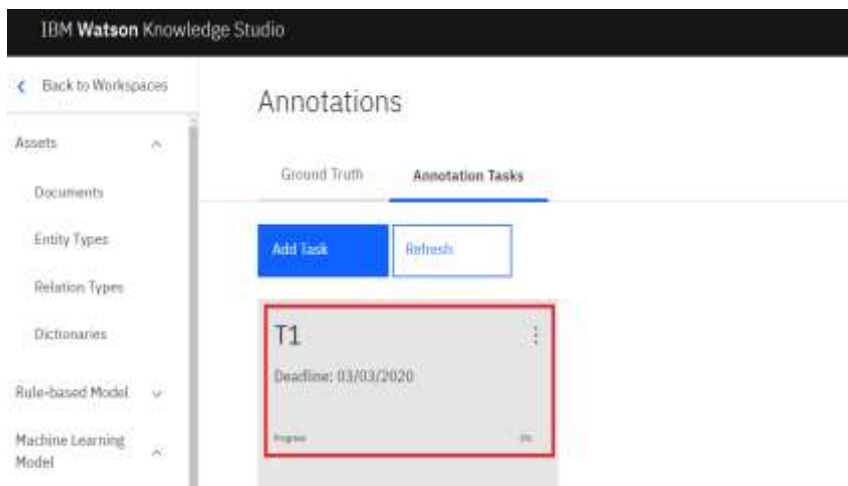
Task name T1 **Deadline** 03/03/2020 📅

Select Annotation Sets

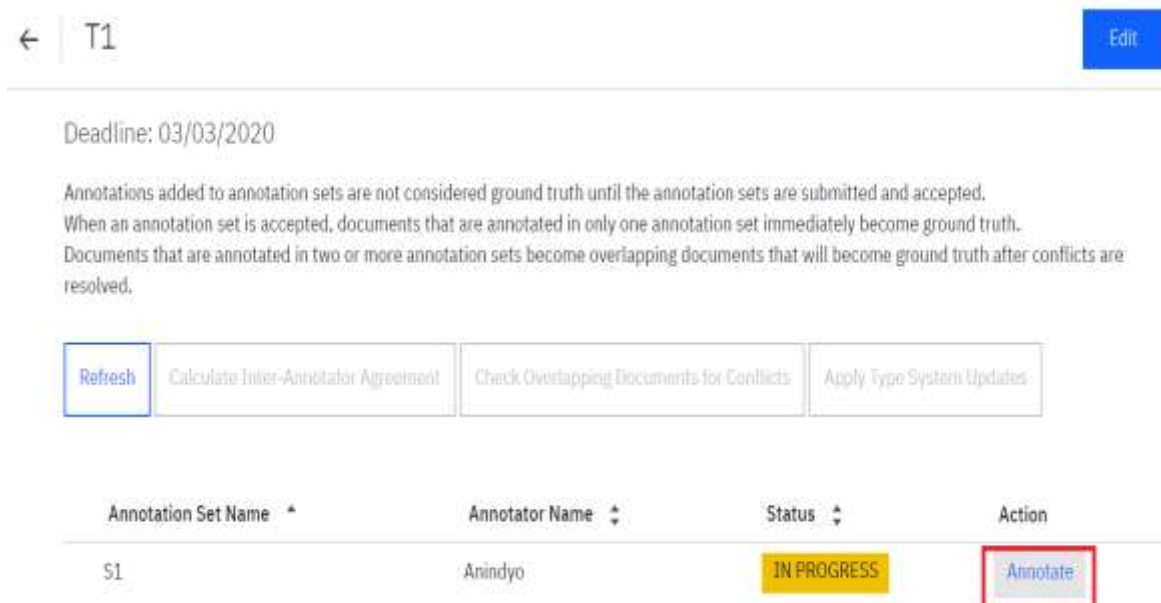
[Create Annotation Sets](#)

Annotation Set Name	Annotator Name	Add to Task / Remove from Task	Action
▼ Selected Sets (1)			
S1	Anindyo	Remove from Task	...

Click on the **Task** to start annotation



and click on **Annotate**



Open a document to annotate. On the middle section of the canvas you can see the document and on the right panel are the entities. We can mark the section of the document that we think matches an entity representation

← Back to Task | Open document list

Ready

Alphab... 14pt 1

Entity Mention

Type	Subtype	Role
Accessibility		
Facilities		
FnB		
Price		
Retail		
ServiceQuality		
Shows		
Sightseeing		
Transportation		

101

- 1 Just a wonderful green space and evening light show.
- 2 A must see.
- 3 The flower dome was great as was the waterfall dome.
- 4 You have to visit this free garden.

And click on the corresponding entity to annotate that text with the corresponding entity. The section of the text will be highlighted with the color code of the entity

← Back to Task | Open document list

Ready

Alphab... 14pt

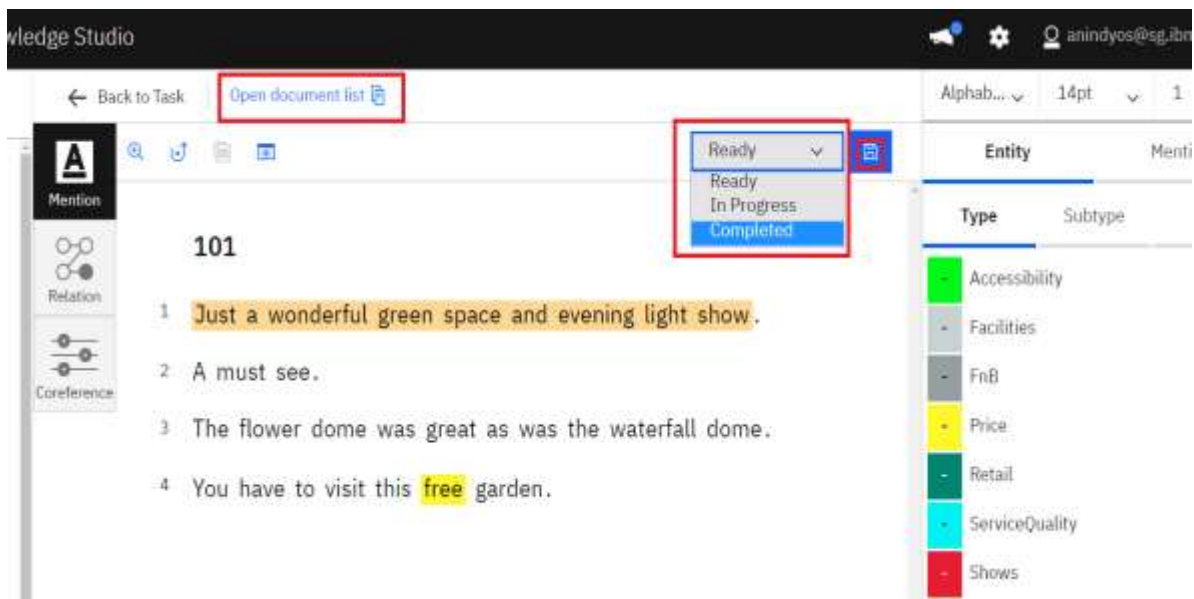
Entity

Type	Subtype
Accessibility	
Facilities	
FnB	
Price	
Retail	
ServiceQuality	
Shows	
Sightseeing	

101

- 1 Just a wonderful green space and evening light show.
- 2 A must see.
- 3 The flower dome was great as was the waterfall dome.
- 4 You have to visit this free garden.

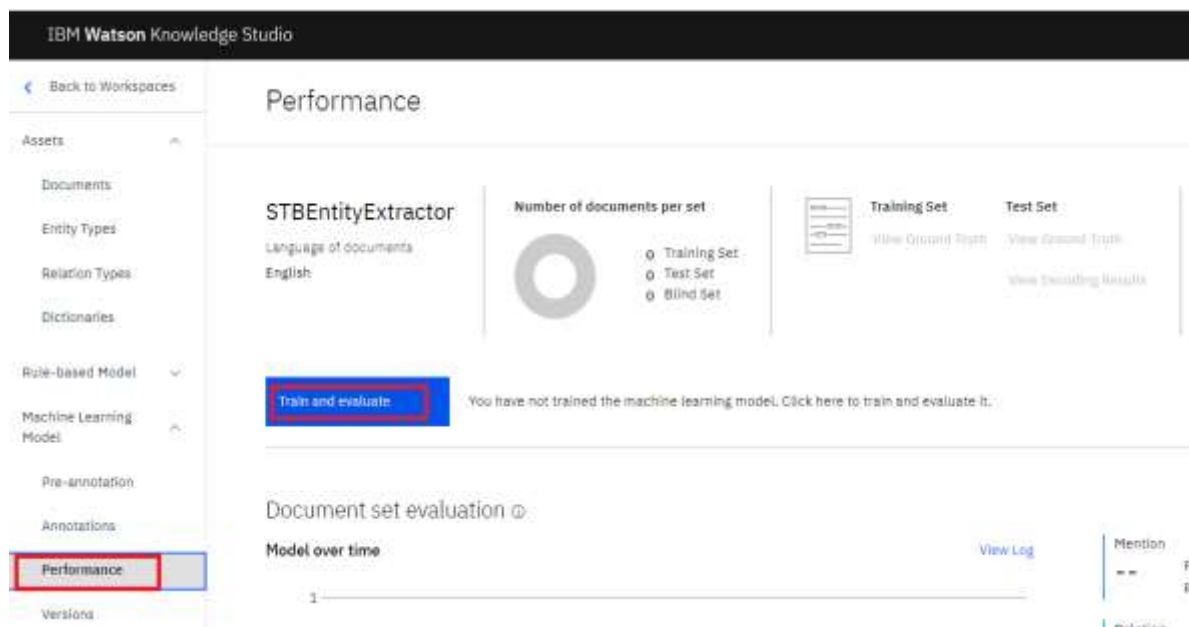
After you have completed annotating the document., change status to **Completed** and save the document



Perform this exercise for all the documents. You would notice that dictionary-based annotations are already done, and no action is required for those.

7. Train & Deploy model

After accepting the task, go to **Performance** under **Machine Learning Model** to train the model.



Follow instructions as shown in the link below:

https://cloud.ibm.com/docs/services/watson-knowledge-studio?topic=watson-knowledge-studio-wks_tutml_intro#wks_tutless_ml8

8. Deploy Model

9. Test Model using API

To extract entities, your application shall call the NLU (Natural language Understanding) API instance where we have deployed the annotator model. In this workshop since we don't have the application, we shall test the same using curl commands. You can use Python or java etc. to test the same as well if you have the corresponding runtimes.

*** To Get Entities, run the below curl command from your laptop command prompt

```
curl --user "apikey":"<Your API Key>" "<Your NLU instance url>/v1/analyze?version=2019-07-12" --request POST --header "Content-Type: application/json" --data @<Your json file name with fully qualified path example C:\Users\AnindyoSarkar\Documents\transportfeedback.json>
```

Sample json file attached below for reference:



The output may not be always 100% accurate as our training set is limited. We would need to train the model with a larger training set which would be representative of the production data to get more accurate results. Normal data science life cycle should be established and deployed

which would review the accuracy at regular intervals and train model continuously to keep up the accuracy of the deployed model

Get Sentiments of entities

Once the transactions with entities of interest are identified, the custom application is expected to store this information in a json database like Mongo DB for further processing.

The next step of the process would be to identify sentiments associated with specific entities. For the same we shall use Tone Analyzer API in this workshop. The sentence or entire document which has the identified transaction shall be submitted to the Tone Analyzer API and sentiments retrieved. The custom application would then join this sentiment with the identified entity and save the result in a relational DB.

1. Use demo application

For our workshop lets use the demo application to test the tone of the feedback. Click the link below to access the application:

https://tone-analyzer-demo.ng.bluemix.net/?_ga=2.167287678.1795414099.1583208908-1313120485.1583208908&cm_mc_uid=62290322875615819138240&cm_mc_sid_50200000=99633231583227288022&cm_mc_sid_52640000=99799621583227288067

or google “**Watson Tone Analyzer demo**” and access the application from the result returned.

Choose “**Your Own Text**” in the same use cases



Tone Analyzer

This service uses linguistic analysis to detect joy, fear, sadness, anger, analytical, confident and tentative tones found in text.

*This system is for demonstration purposes only and is not intended to process Personal Data. No Personal Data is to be entered into this system as it may not have the necessary controls in place to meet the requirements of the General Data Protection Regulation (EU) 2016/679.

By using this application, you agree to the [Terms of Use](#)

Resources:

[Documentation](#)

[API Reference](#)

[Fork on Github](#)

Start for free in IBM Cloud

Sample use cases

Choose an example to learn how you can adjust the tone of your content to change people's perceptions, or improve its effectiveness. [Learn more](#)

☐ Tweets ☐ Online Review ☐ Email message ☐ Product Review in French ☒ Your own text

Analyzing Customer Engagement Data? Try out the [Tone Analyzer Customer Engagement Endpoint](#)

Paste any document content from our sample and click on **Analyze**

Output

The Tone Analyzer Service analyzes text at the document level and the sentence level. Use the document level analysis to get a sense of the overall tone of the document, and use the sentence level analysis to identify specific areas of your content where tones are the strongest.

To understand how to interpret your tone score, see [Understand your Tone Score](#)

Document-level

Tones

Anger ☐ Fear ☐ Joy ☒ Sadness ☐ Analytical ☐ Confident ☐ Tentative ☐

< > [View JSON](#)

Sentence-level

Identify sentences with stronger tones in context or sorted by score. Highlighted sentences indicate the likelihood of a tone present. If more than one tone is present, the stronger one is shown. Click on a sentence to see a breakdown of all tones.

Tones

Analytical

Confident

In context

Confident: A person's degree of certainty

Review the overall output. You can view the json and see the individual scores.

You can also review the sentence level feedback and identify sections of the feedback where that sentiment was detected.

2. Call API

You can achieve the same result by using the Tone Analyzer API instead of using the demo application. For detailed instructions use the link below:

<https://cloud.ibm.com/apidocs/tone-analyzer>

This brings us to the end of our workshop.

The End