Project Report

|  |  |
| --- | --- |
| **Course Name (NICF)** | **NICF Diploma in Business Analytics** |
| Product Name (Marketing & Sales) | Professional Diploma in Data Science |
| **Module Name (NICF)** | **NICF Data Science Essentials** |
| Product Name (Marketing & Sales) | Data Science Essentials |

|  |  |  |  |
| --- | --- | --- | --- |
| Student name | | Assessor name | |
| Saminathan Renganayagi | | Rajendra Kissan | |
| Date issued | Completion date | | Submitted on |
| 28/4./21 | 5/5/21 | | 6/5/21 |
|  | |  | |
| Project title | News Categorization | | |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature: Date: |

Content

1. Requirement Analysis
2. Why Azure AML is a suitable solution for this scenario
3. How to: Import and Prepare Data into Azure ML Studio
4. How to: Preprocess Text
5. How to: Extract Features from the DataSet
   1. n-grams TF feature extraction
   2. unigrams TF-IDF feature extraction
6. How to: Train and evaluate models
7. How to compare the Algorithm and get the best Accuracy
   1. N- gram {from trans formed dataset )
   2. Unigram (from Extract ngram saved dataset)
8. How to: Deploy trained models as web services
   1. 5a. deploy web service with n-grams TF model
   2. 5b. deploy web service with unigrams TF-IDF model

**Project Overview**

In this project, News dataset consisting of news articles needed to be worked on.

The articles are across the following categories:

- Business

- Entertainment

- Politics

- Sports

- Technology

Proper features are to be extracted and effective models built to assign each article to the correct category. The following algorithms on Azure ML Studio are used and evaluation performed on each model:

- Multiclass Neural Network

- Multiclass Decision Jungle

- Multiclass Logistic Regression

- Multiclass Decision Forest

The end result is a web service that could predict the news category based on input text.

**Project Technical Environment**

This project uses the Microsoft Azure Machine Learning platform (studio.azureml.net), which is a cloud based environment to train, evaluate, deploy ML models, using drag and drop modules to build experiments. Provided News dataset is used.

Describe the Project with summary of analytical processes and project outcomes (Explain the Pr**1. Task1 Requirement Analysis**

1. Analyse the Requirements related to the Scenario and justify why Azure ML is a suitable data analytics solution.

Requirements and steps:

a. Data preprocessing - transformation, cleanup, remove duplicates, edit metadata

b. Text preprocessing - stop words, lower case, lemmatisation, remove special characters

c. Feature hashing - ngram (tf weights), unigram (tf-idf weights), filter based selection

d. Splitting into training and test sets, and further splitting of test set into validation and test

e. Hyperparameters tuning using k-folds

f. Compare different algorithms, choose the best model based on metrics and save the best model.

g. Deploy web service using best model.

Given the steps involved, Azure ML is a suitable data analytics solution as it could fulfil all the requirements, has an easy drag and drop interface to create the modules with necessary parameters, and facilitates easy modifications and changes that enables one to experiment with different parameters.

Task 1.:

a.import csv file

b. Select Columns in Dataset

c Clean Missing Data

d..Remove Duplicate Rows

Task 2:

e Preprocess Text

f. Select Columns in Dataset

g. Edit Metadata

h. feature hashing

i. Extract N-Gram Features from Text

Task3: N gram(TF-feature extraction)

1. Split Data

70:30=70% is splting for traind model

Split data

15:15=15% is splitting for input of cross validation model and another 15% go to input of scored model for testing

k. Multiclass Neural Network

l. Multiclass Decision Jungle

m. Multiclass Logistic Regression

n. Multiclass Decision Forest

o. Tune Model Hyperparameters

p. Score Model

score model=predicted value=scored label

q. Evaluate Model

f. find the best accuracy from overall accuracy for compare with multi class Algorithms

Task4:(Unigram TF-IDF feature Extraction ):

i. Split Data

k. Multiclass Neural Network

l. Multiclass Decision Jungle

m. Multiclass Logistic Regression

n. Multiclass Decision Forest

o. Tune Model Hyperparameters

p. Score Model

q. Evaluate Model

f. find the best accuracy for compare with multi class Algorithms

Task4:

Scored label Deploy webservices with N gram TF Feature

Task5:

Scored label Deploy Webservices with Unigram TF-IDF Feature

2**. Why Azure ML is a suitable solution for this scenario.**

I've divided the project into 5 parts:

Part 1 : Data and Text Preprocessing

Azure ML has modules to select the required columns in dataset, clean the missing data, remove duplicate rows, preprocess text and save results into csv file. It has the added flexibility of custom python script, which I've used to code to downsample data to deal with unbalanced dataset.

Part 2a : NGrams - Feature Hashing, Model training, hyperparameters tuning and evaluation

Extract N-grams from text is used for feature hashing, using n-grams size of 2, weighting function of TF weight, which is then further filtered to choose 2^12 records. Vocabulary is stored as ATD\_Vocab\_NGrams dataset.

Data is split into training (70%) , validation (15%) and testing (15%) using Split Data module.

Each different model (Multiclass Neural Network, Multiclass Decision Jungle, Multiclass Logistic Regression, Multiclass Decision Forest) is trained and best hyperparameter chosen using Tune Model Hyperparameters.

Each model is then evaluated with Confusion Matrix.

Part 2b : Unigrams - Feature Hashing, Model training, hyperparameters tuning and evaluation

Extract N-grams from text is used for feature hashing, using n-grams size of 1, weighting function of TF-IDF weight, which is then further filtered to choose 2^12 records. Vocabulary is stored as ATD\_Vocab\_Unigrams dataset. Data is split into training (70%) , validation (15%) and testing (15%) using Split Data module.

Each different model (Multiclass Neural Network, Multiclass Decision Jungle, Multiclass Logistic Regression, Multiclass Decision Forest) is trained and best hyperparameter chosen using Tune Model Hyperparameters.

Each model is then evaluated with Confusion Matrix.

Part 3a : Deployment of NGrams model

Web services input and output modules are created, where the input text is being preprocessed and feature hashed using previously created vocabulary. The saved best model Ngrams is then used to predict the scores and the predicted category output as result.

Part 3b : Deployment of Unigrams model

Web services input and output modules are created, where the input text is being preprocessed and feature hashed using previously created vocabulary. The saved best model Unigrams is then used to predict the scores and the predicted category output as result.

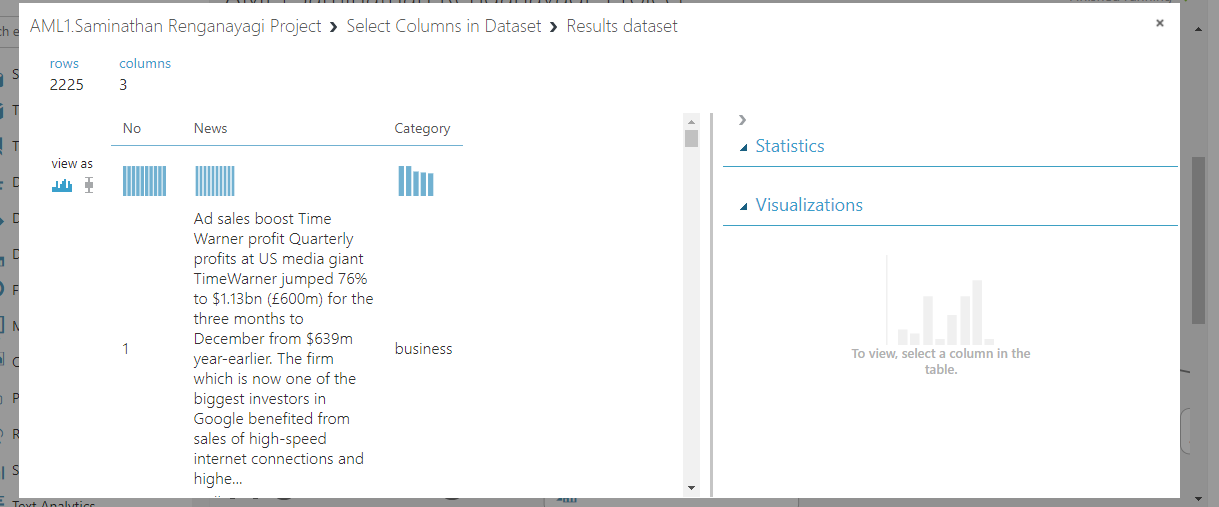
**Activity 1:**

**3. How to import and prepare data into Azure ML studio.**

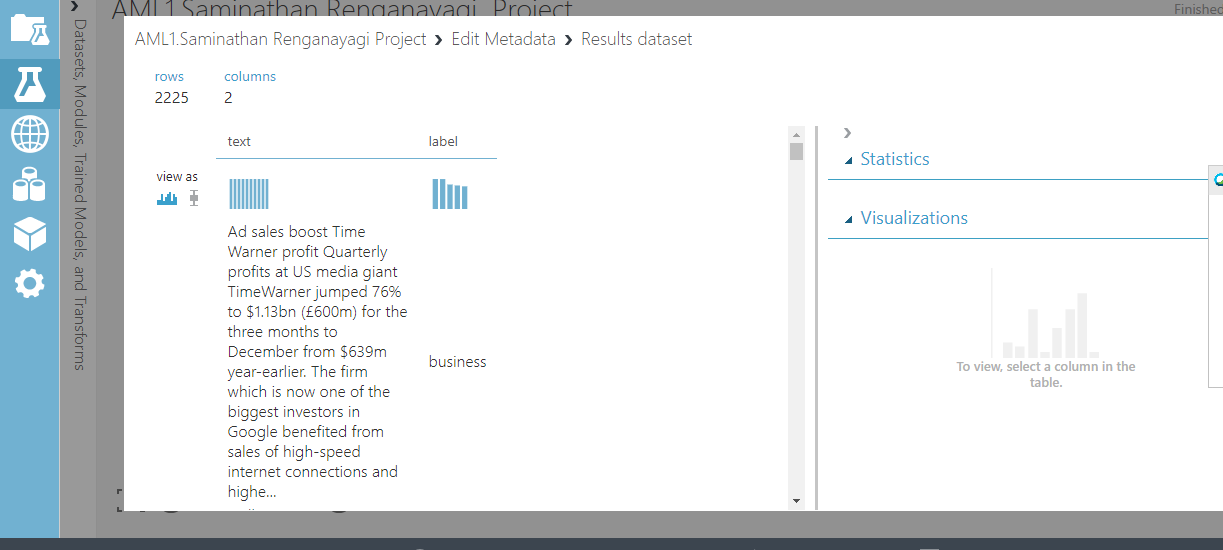
1. Import the Data into Azure AML. Name and describe the module you have used for importing the data.

ATDNewsData.csv has been uploaded into Azure ML Studio and saved as dataset.

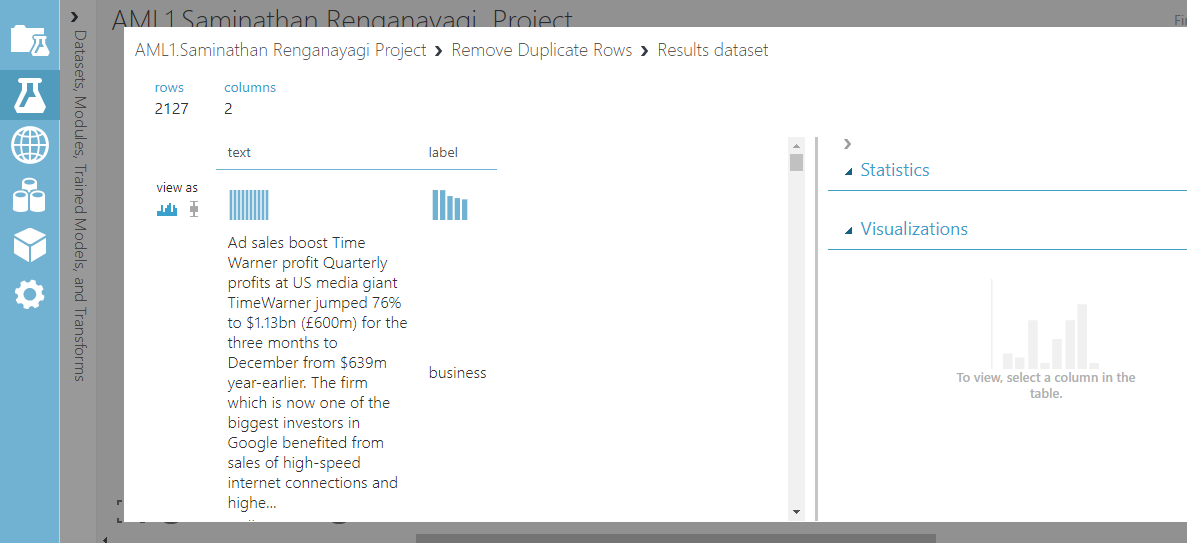
2. Identify the Classes and Number of Articles per class using visualisation.

i

Edit meta data



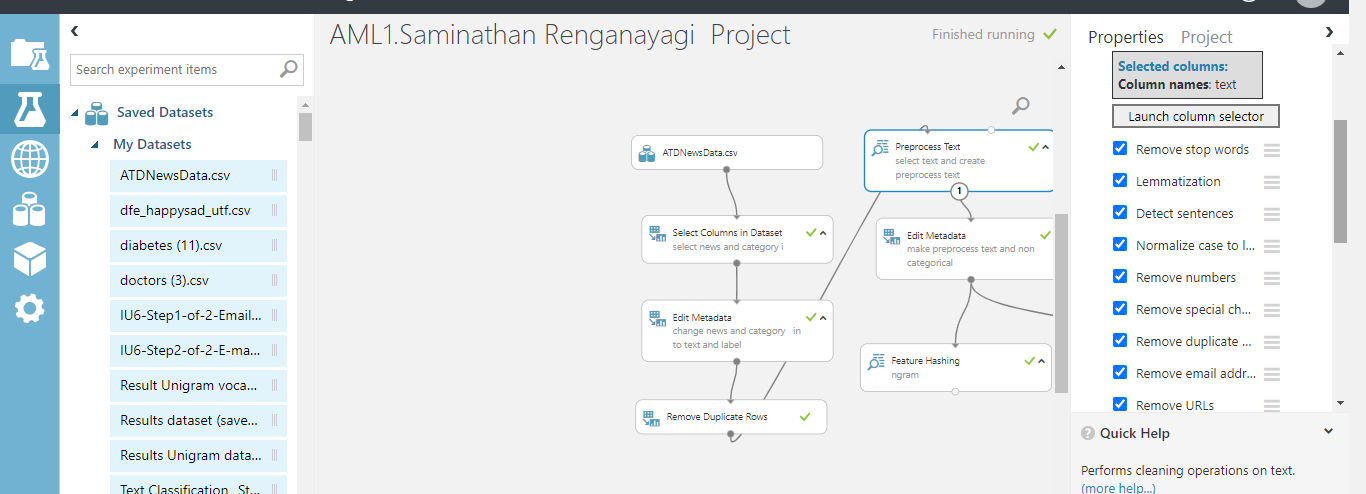
Remove the duplicate rows and column=2127 rows and 2 columns



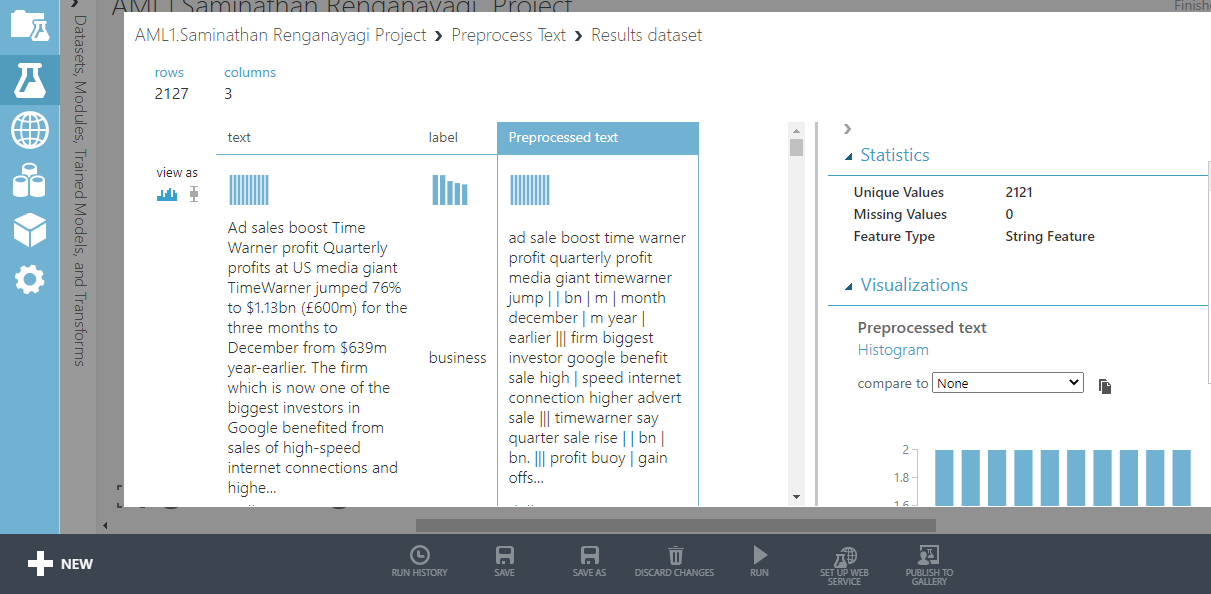
4. How to prepare the preprocess text=News/Text

a. Before the preprocess =2 columns-text,label

b. Screen Shot of preprocessing



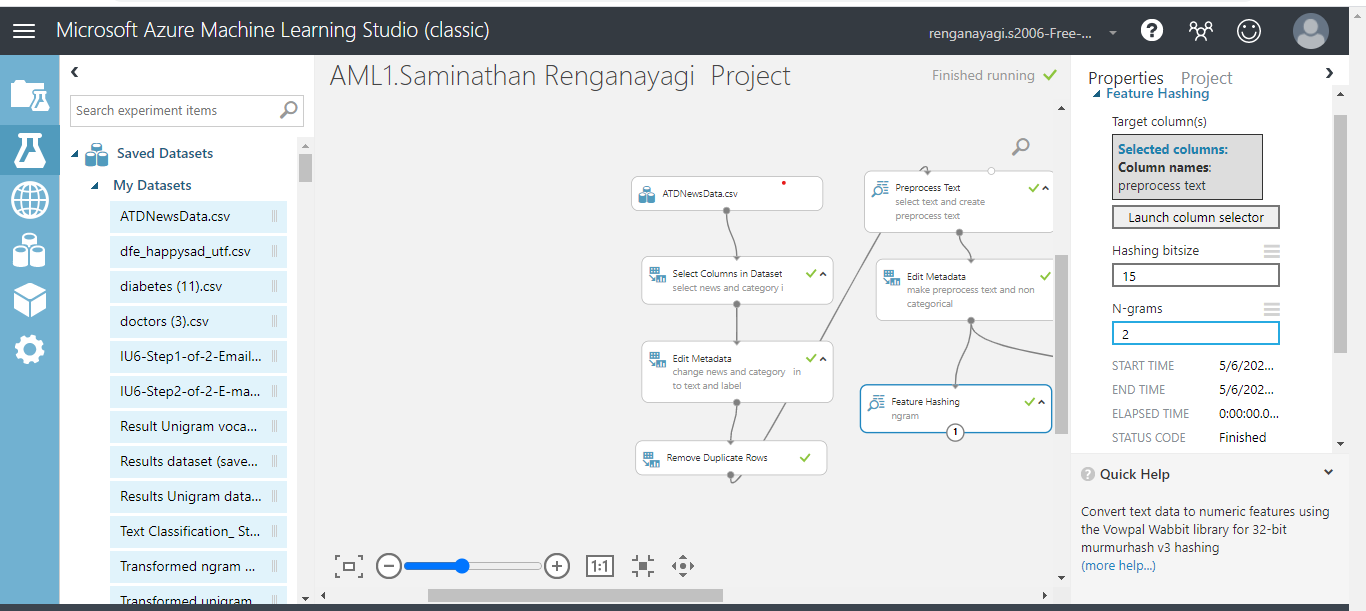
C: screen Shot of output=3 columns –text,label,preprocesstext



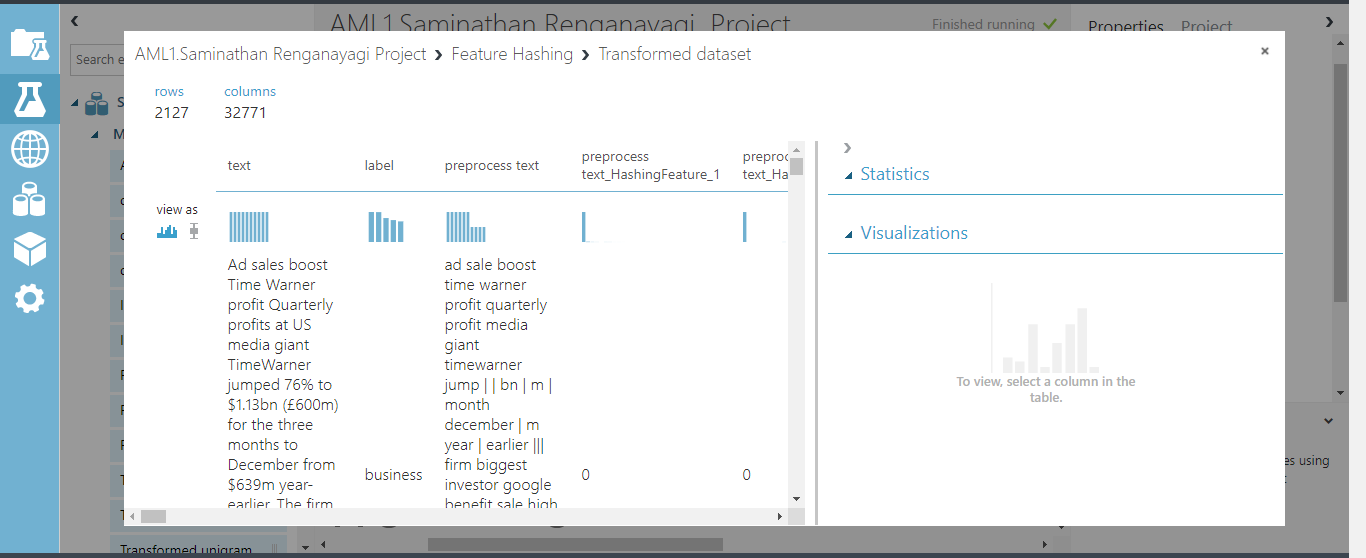
1. **How to extract the feature from the dataset**

A .Ngram TF features Extraction

* 1. screen shot of feature hashing extraction



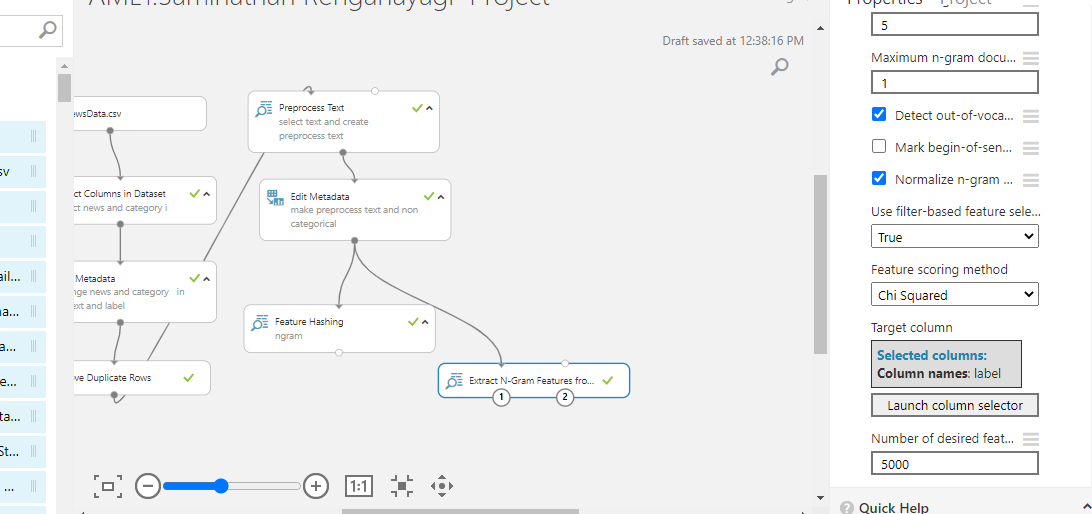
b.Screen shot after run 32771



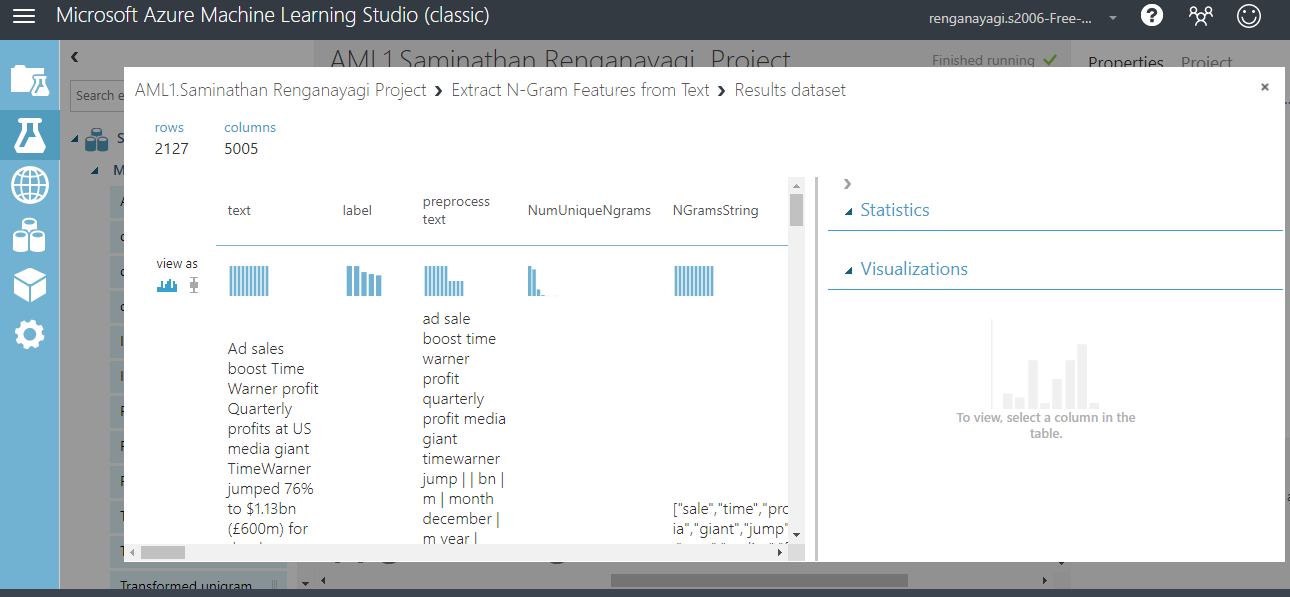
* 1. save dataset

**B. Unigram TF-IDF feature extraction from the text**

* 1. a.screen shot of unigram-extract Ngram feature from the text



Output of the unigram dataset



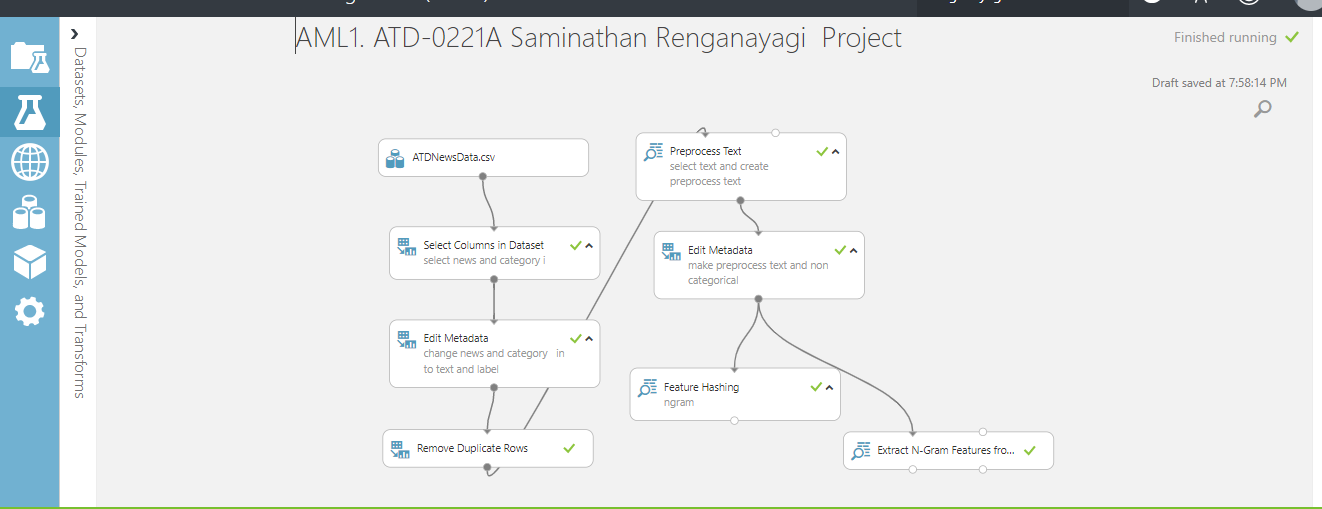
**Activity 1:**

: Data preprocessing - transformation, cleanup, remove duplicates, edit metadata

. Text preprocessing - stop words, lower case, lemmatisation, remove special characters

Feature hashing - ngram (tf weights), unigram (tf-idf weights), filter based selection

TASK(2---5)



**Activity 3:**

Data is split into training (70%) , validation (15%) and testing (15%) using Split Data module

Each different model (Multiclass Neural Network, Multiclass Decision Jungle, Multiclass Logistic Regression, Multiclass Decision Forest) is trained and best hyperparameter chosen using Tune Model Hyperparameters.

**How to extract Features from the DataSet**

This task has two sub-tasks:

Task 4a: n-grams TF feature extraction

Task 4b: unigrams TF-IDF feature extraction

You should extract features from text using Bag of Words Model and prepare the Term-Document Frequency Matrix and determine the TF-IDF for each term.

**How to train and evaluate models**

Train the model using Multiclass Neural Network, Multiclass Decision Jungle, Logistic Regression, and Multiclass Decision Forest.

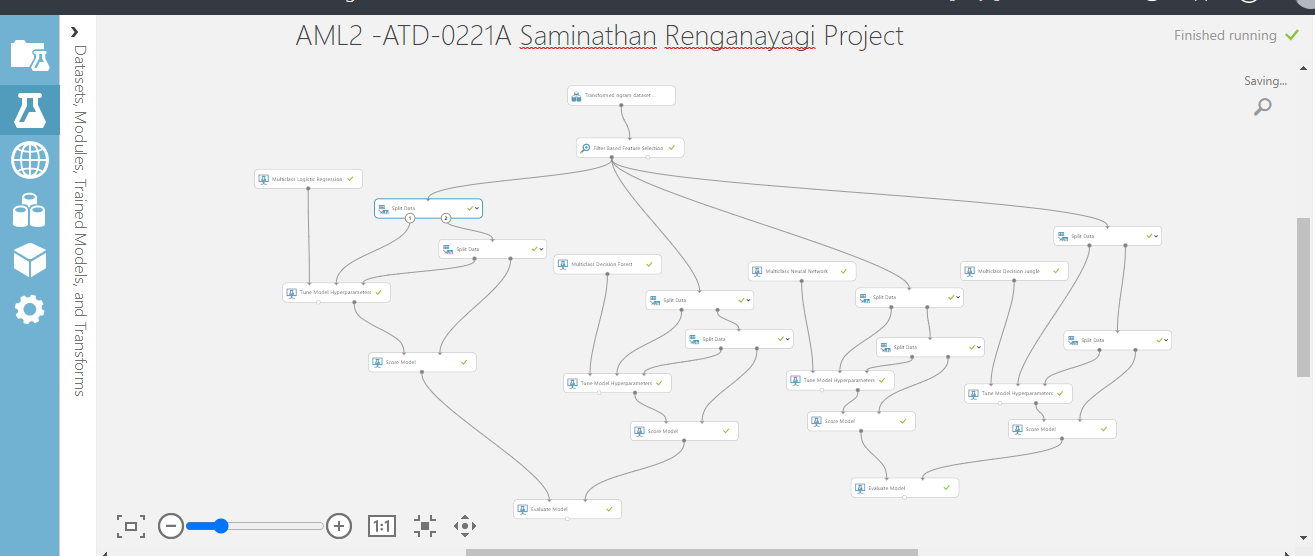
Review and Refine the Models.

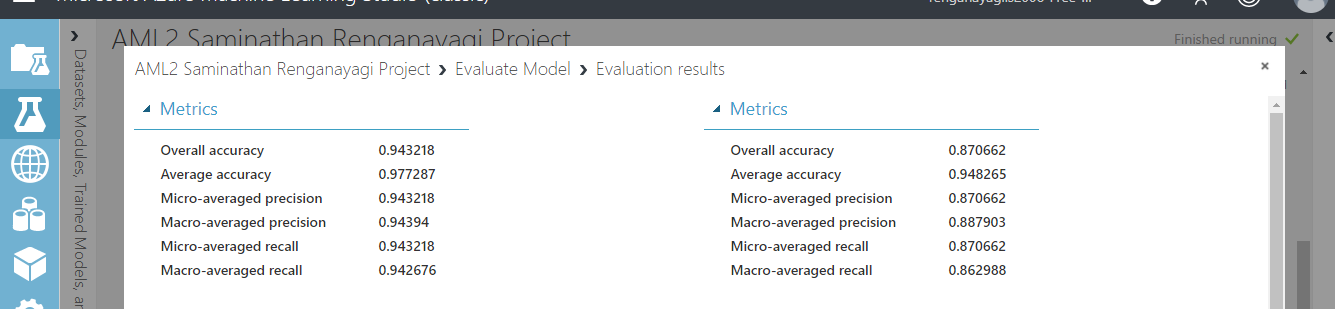
Prepare the Confusion Matrix for the above 4 models.

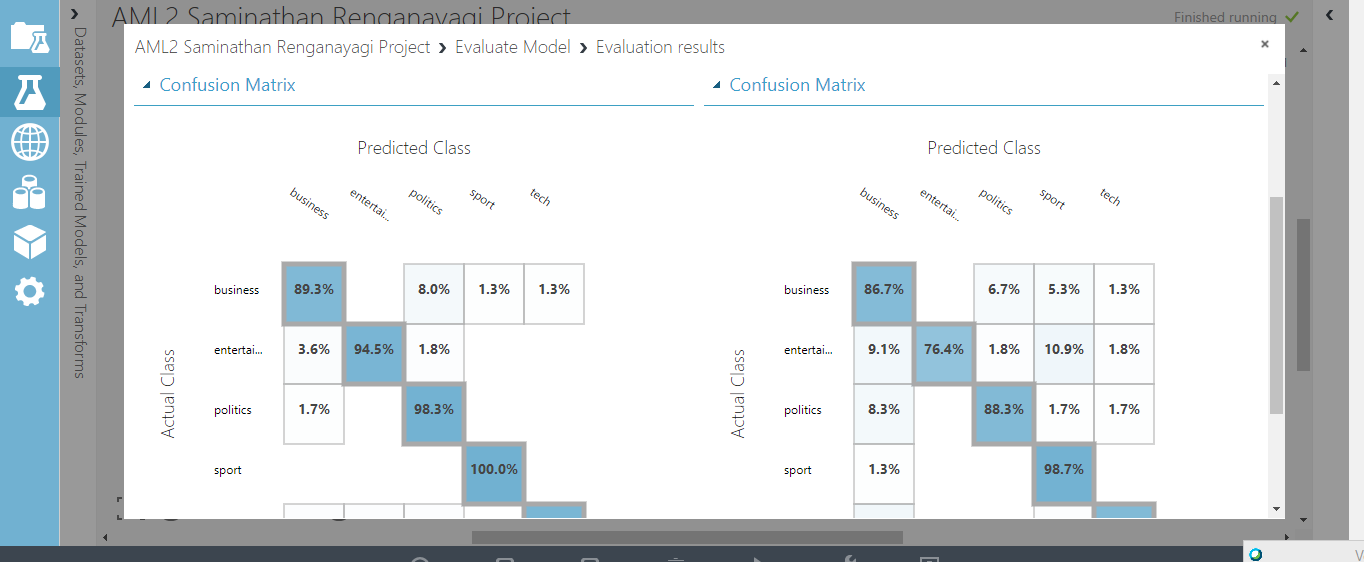
Evaulate which of the models is more effective and justify with reasons.

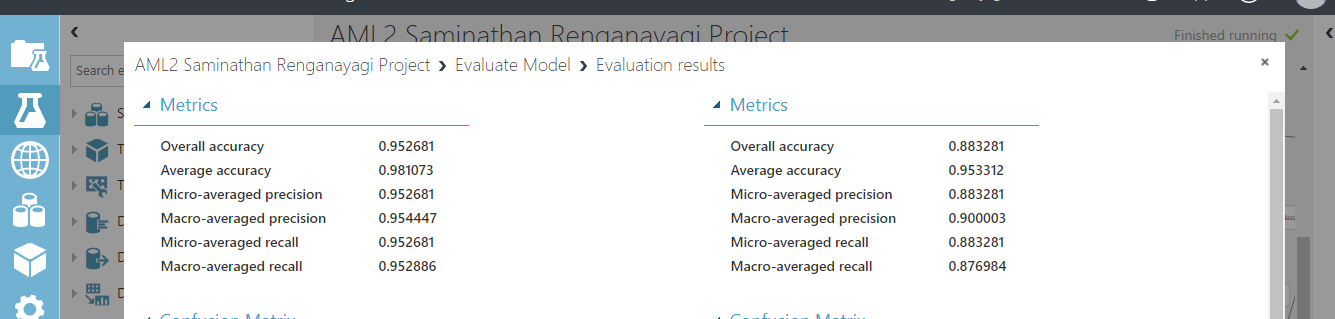
Each model is then evaluated with Confusion

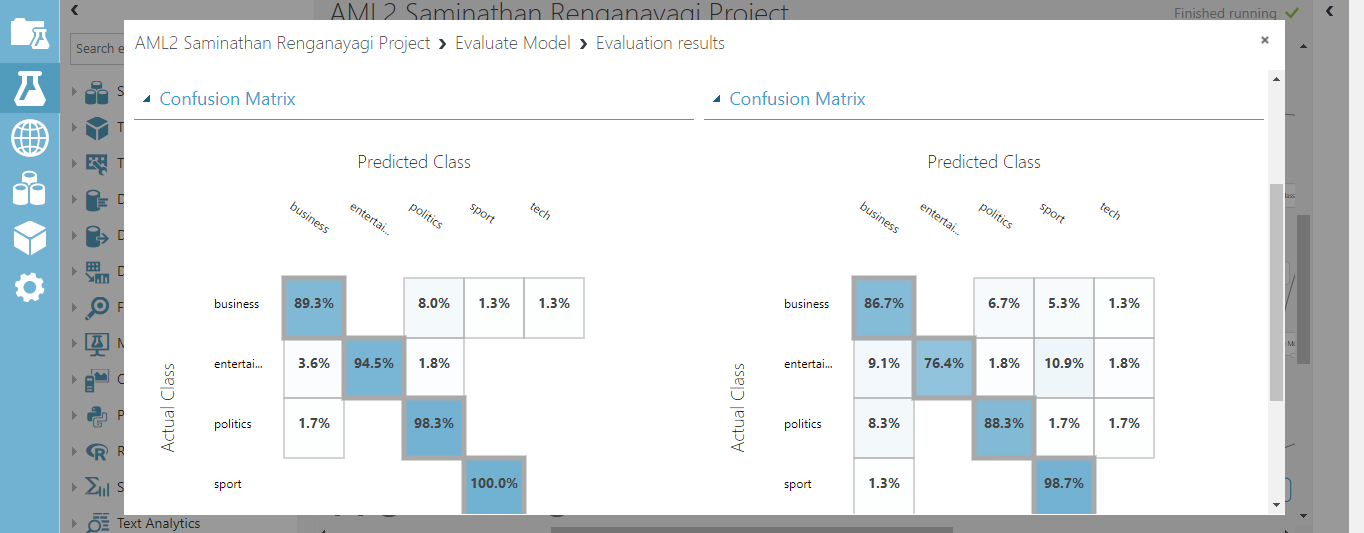
TASK(3)











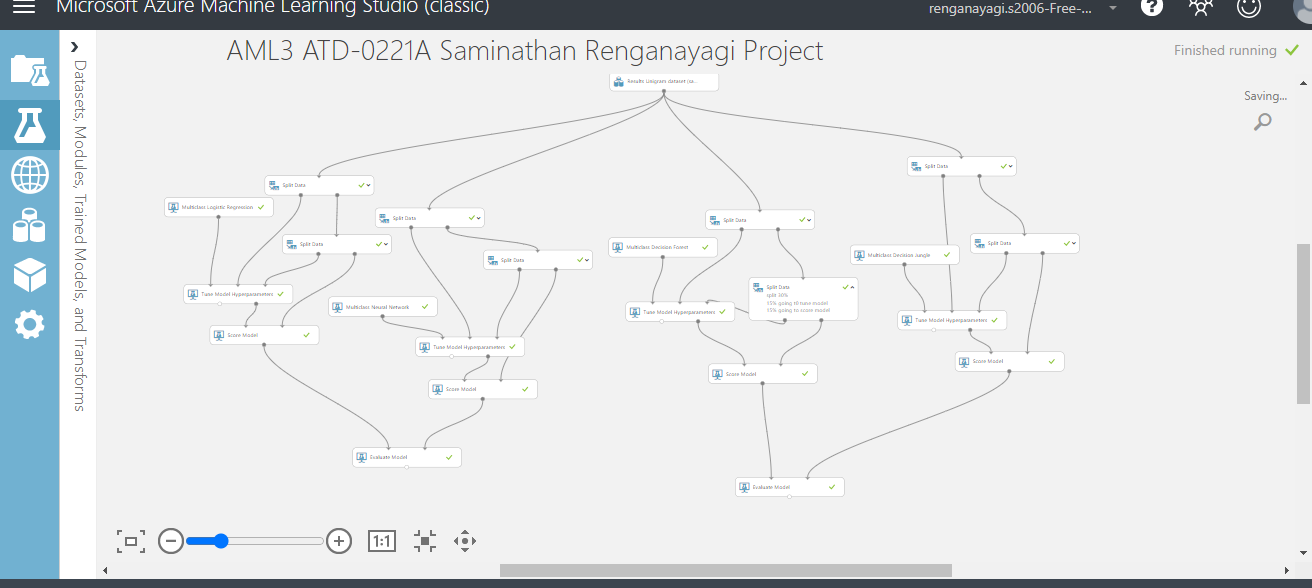
Activity 4:

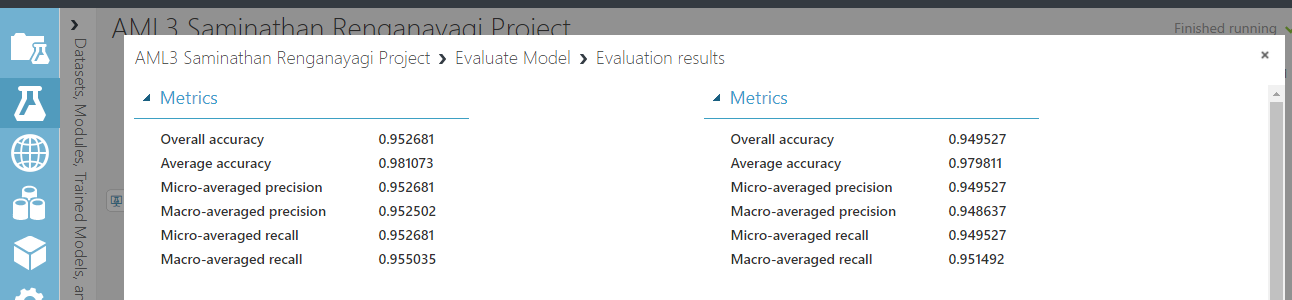
Extract N-grams from text is used for feature hashing, using n-grams size of 1, weighting function of TF-IDF weight, which is then further filtered to choose 2^12 records. Vocabulary is stored as ATD\_Vocab\_Unigrams dataset. Data is split into training (70%) , validation (15%) and testing (15%) using Split Data module.

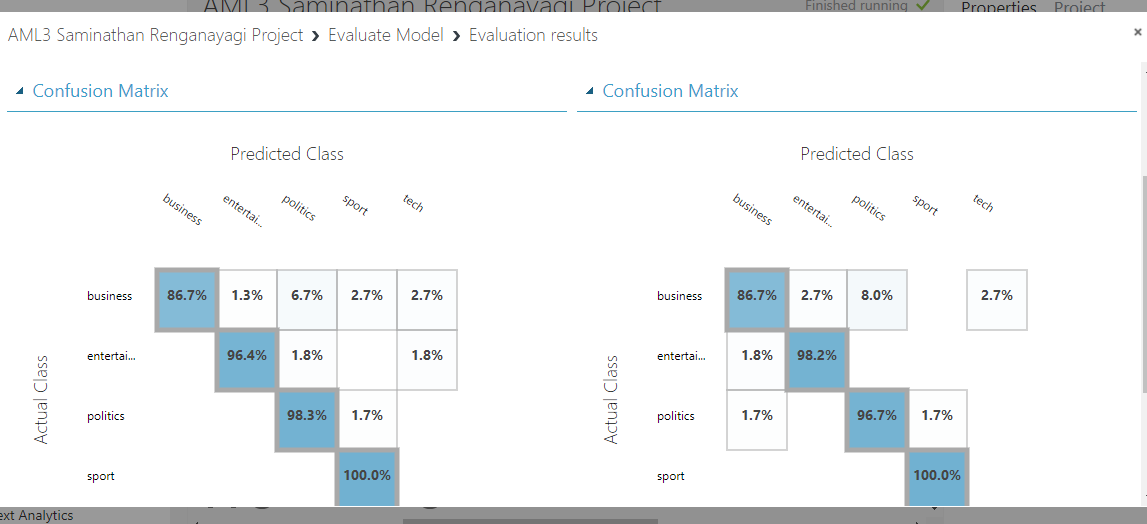
Each different model (Multiclass Neural Network, Multiclass Decision Jungle, Multiclass Logistic Regression, Multiclass Decision Forest) is trained and best hyperparameter chosen using Tune Model Hyperparameters.

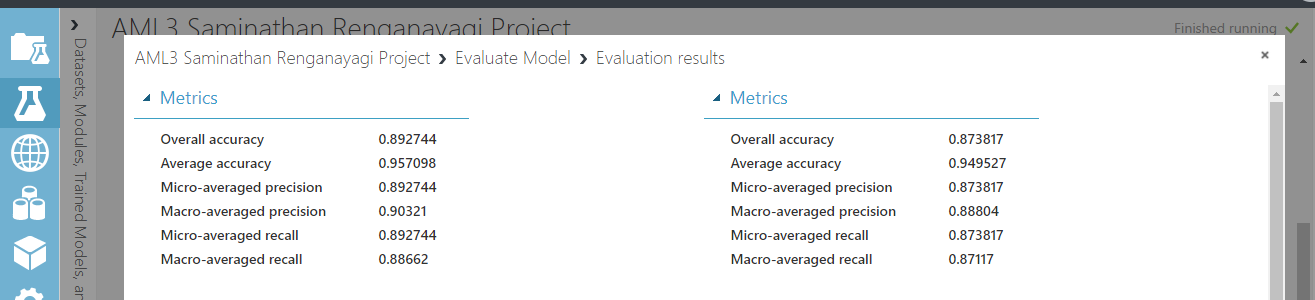
Each model is then evaluated with Confusion Matrix

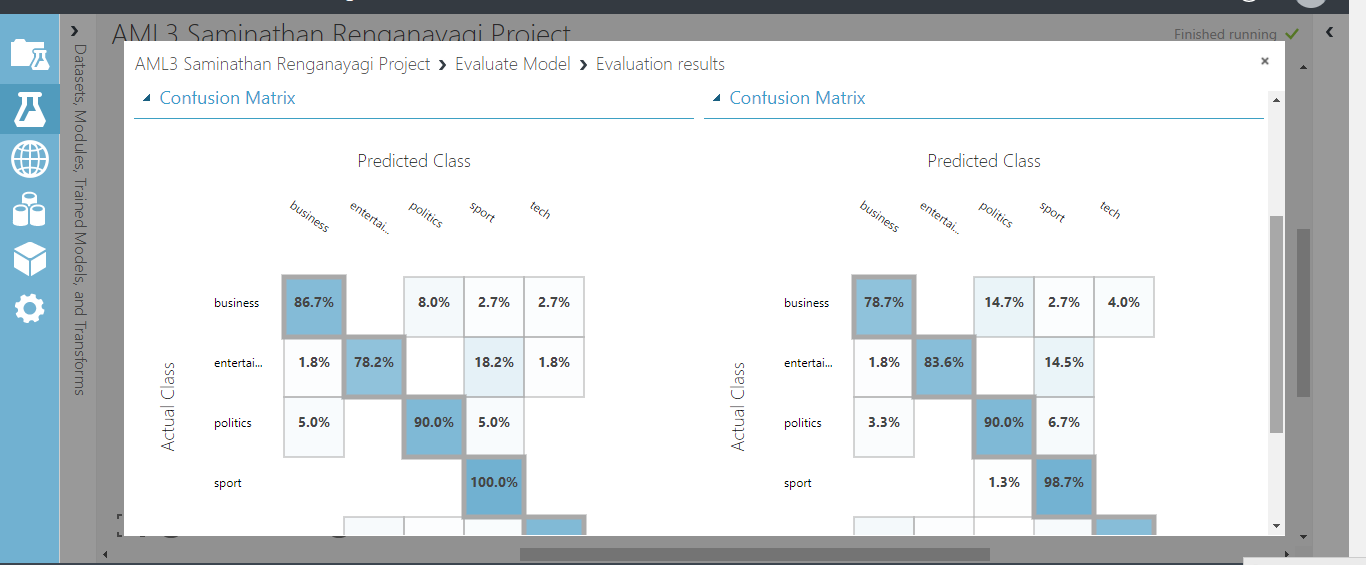
Task 4:



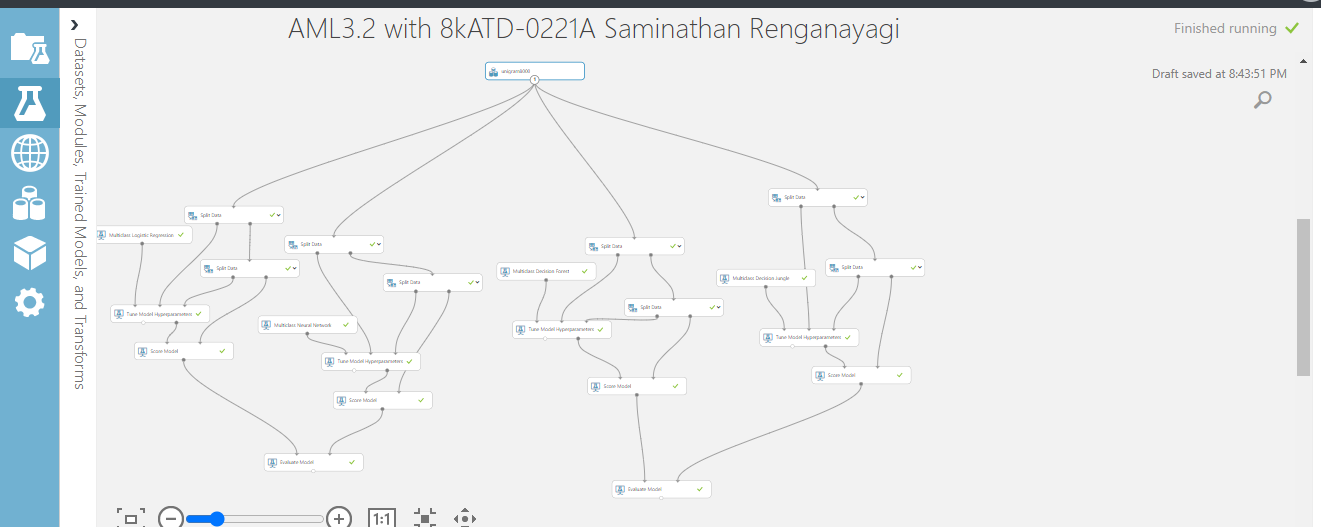


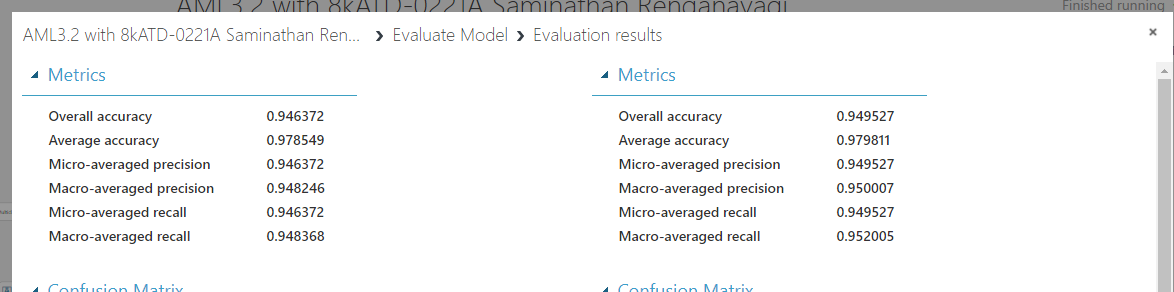


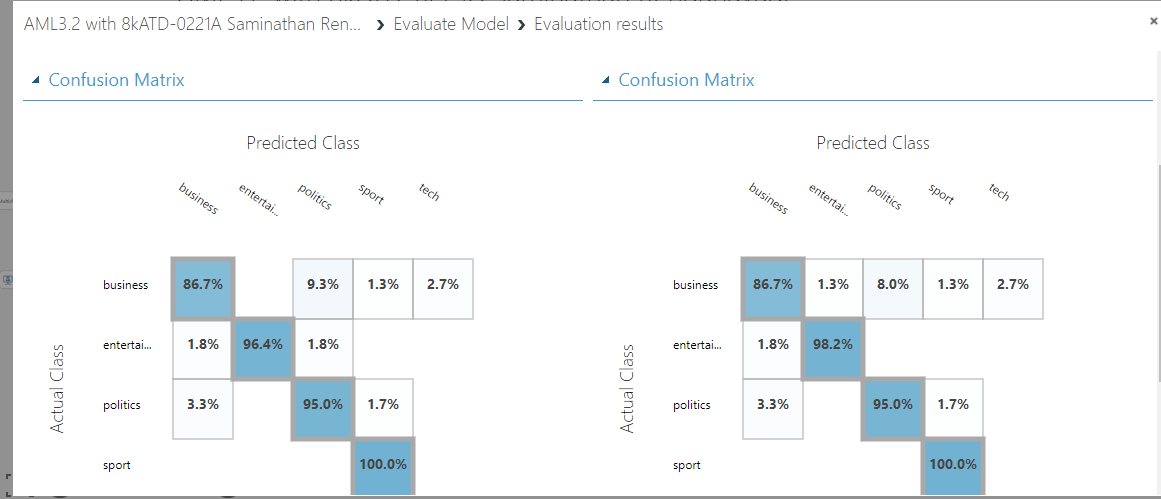


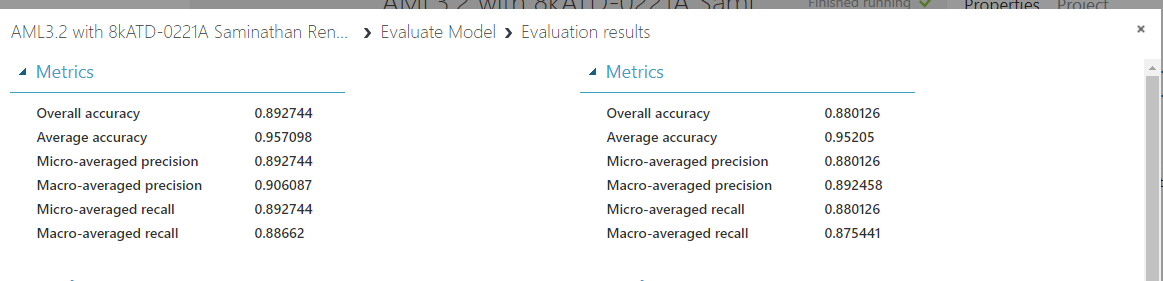


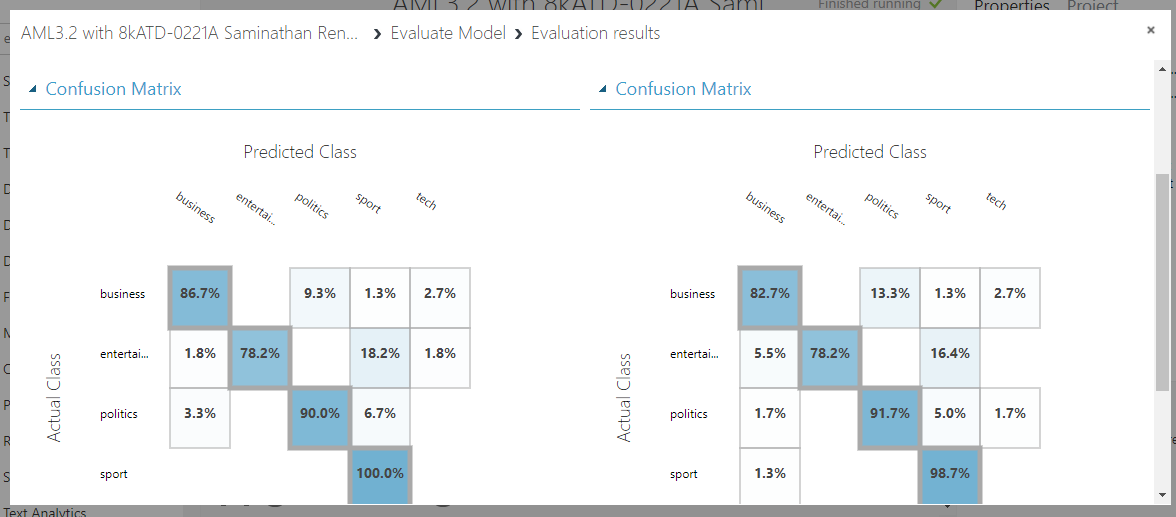
Features 8000 DF-IDF Ngram Feature extaction(Unigram)







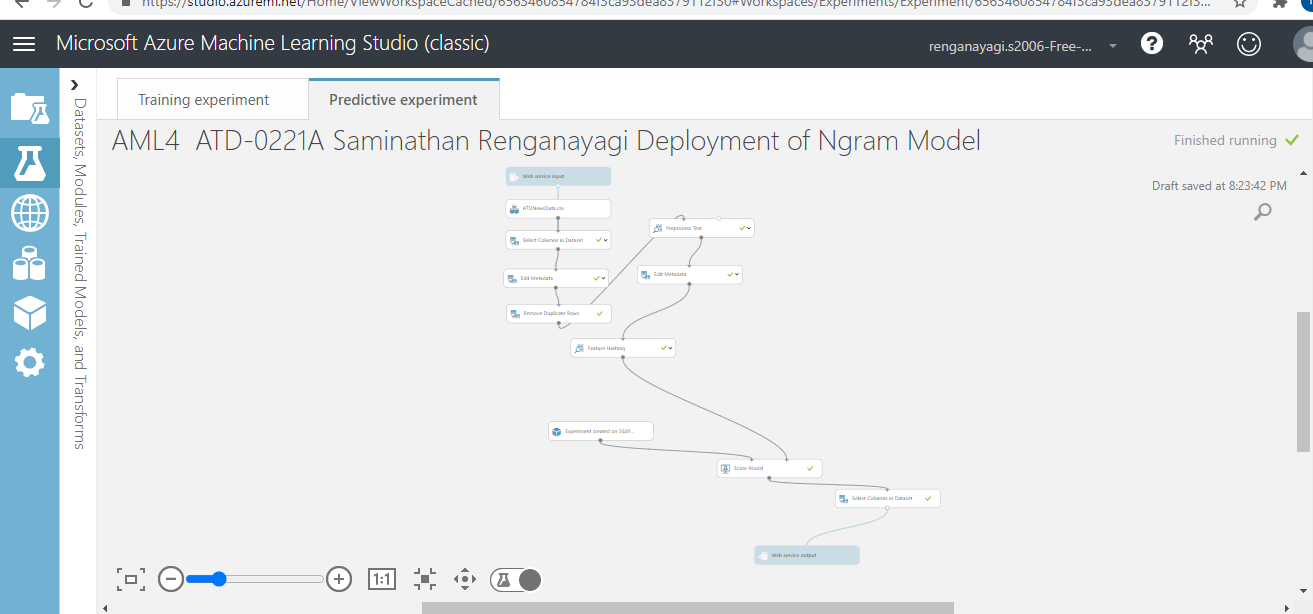


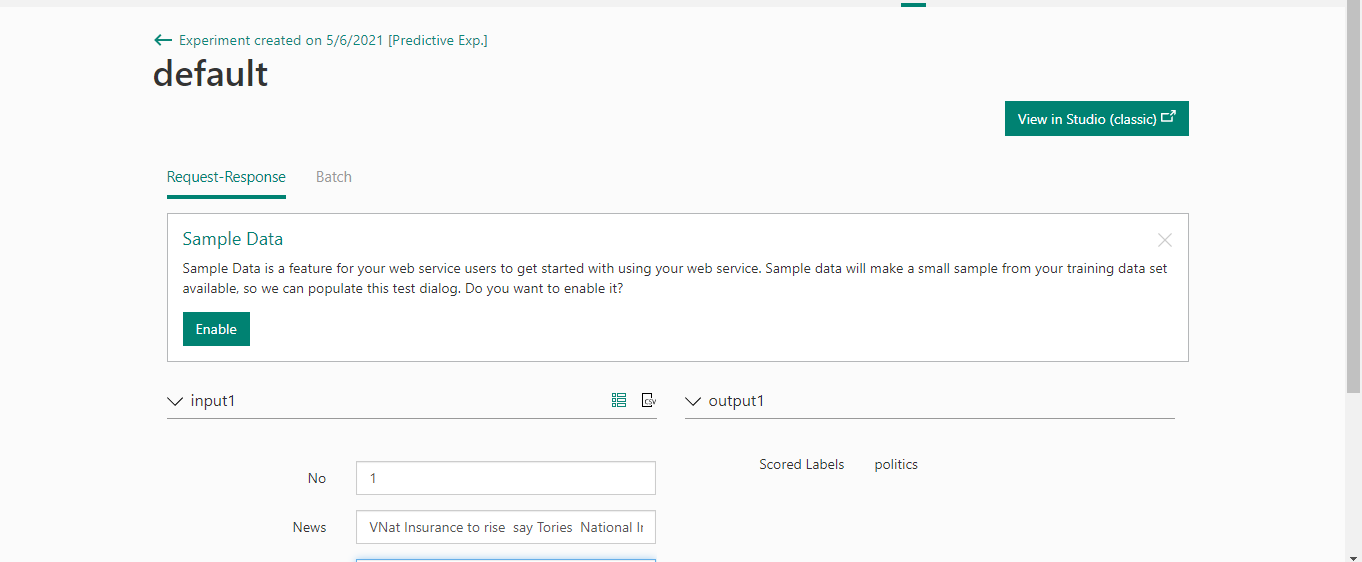


Activity 5a ::Deployment of NGrams model

Web services input and output modules are created, where the input text is being preprocessed and feature hashed using previously created vocabulary. The saved best model Ngrams is then used to predict the scores and the predicted category output as result.

TASK 5:





5b: Deployment of Unigrams model

Web services input and output modules are created, where the input text is being preprocessed and feature hashed using previously created vocabulary. The saved best

TASK 6:

