Hands-on Lab Agenda

1 Table of contents

2 Agenda

Pre-requisites: Databricks environment / Local environment with Python installed, Active Azure Subscription, downloaded the required datasets.

Recommended environment configuration

Databricks

- Cluster Mode: Standard
- Databricks Runtime Version: 7.5 (includes Apache Spark 3.0.1, Scala 2.12)
- Worker type: Standard_DS3_v2, 14 Gb Memory, 4 cores, 0.75 DBU

Local environment

- Python version: 3.6.5 or higher

Libraries & SDKs:

```
pip install azure-ai-textanalytics –pre
pip install pandas
```

all the other libraries are native.

Github link: GitHub - ajakupov/Lab-NLP-On-Azure

2.1 NLP

Step	Data	Comments	Tools
Part One		Azure AutoML	
Construct text classifier	Amazon pet product reviews https://www.kaggle.com/c/amazon-pet-product-reviews-classification/data?select=train.csv	Launch the experiment at the beginning of the session and see the results two hours later. **Estimated time**: 5-10 min	Azure ML Services Interface
Part Two		Using the pre-built models	
Short presentation of customer cases		Expertime clients **Estimated time**: 5-10 min	

Uncover insights such as sentiment, entities, relations, and key phrases in unstructured text	Imbalanced Twitter sentiment dataset https://www.kaggle.com/arkhoshghalb/twitter-sentiment-analysis-hatred-speech?select=train.csv	Cognitive Services: Text Analytics API. Opinion mining for topic detection in long phrases. **Estimated time**: 15-20 min	Azure Databricks to call the API
Translate text	Amazon pet product reviews https://www.kaggle.com/c/amazon-pet-product-reviews-classification/data?select=train.csv	Cognitive Services: Translation API. Rapidly discuss custom translation service by Azure. **Estimated time**: 15-20 min	Azure Databricks to call the API
Part three		Using semi-supervised models	
Train bots using LUIS	Danone's FAQ https://www.danone.com/investor-relations/shareholders/faq.html	**Estimated time**: 15-20 min	• LUIS Interface for creating the bots. • Databricks notebooks for calling the created bot.
Part four		Building fully custom models	
Unsupervised learning	Imbalanced Twitter sentiment dataset https://www.kaggle.com/c/amazon-pet-product-reviews-classification/data?select=train.csv	Topic modeling using LDA. **Estimated time**: 15-20 min	Azure ML Designer

Construct text classifier with normalized and unnormalized data	Dataset 1: https://www.kaggle.com/c/amazon-pet-product-reviews-classification/data?select=train.csv Dataset 2: https://www.kaggle.com/arkhoshghalb/twitter-sentiment-analysis-hatred-speech?select=train.csv	Text classifier. Normalize imbalanced dataset using smote. We will discuss how to deploy the trained model as an API using Azure Container Instances **Estimated time**: 20-30 min	Azure ML Designer Databricks notebooks + AzureML SDK (demonstration)

2.2 Computer Vision

Pre-requisites: Databricks environment / Local environment with Python installed, Active Azure Subscription, downloaded the required datasets.

Recommended environment configuration

Databricks

- Cluster Mode: Standard
- Databricks Runtime Version: 7.5 (includes Apache Spark 3.0.1, Scala 2.12)
- Worker type: Standard_DS3_v2, 14 Gb Memory, 4 cores, 0.75 DBU

Local environment

- Python version: 3.6.5 (recommended for windows users, as tensorflow doesn't work with the higher python versions)

Libraries & SDKs:

```
pip install azure-cognitiveservices-vision-computervision
pip install opencv-python
pip install tensorflow
pip install Pillow
pip install numpy==1.16.2
```

Azure Free Trial:

To activate your free temporary subscription please follow the instructions in the bellow tutorial

Create your Azure free account today | Microsoft Azure

Github link: ajakupov/Lab-CV-On-Azure (github.com)

Step	Data	Comments	Tools
Part One		Using the pre-built models	
Short presentation of customer cases		Expertime clients **Estimated time**: 5-10 min	

Analyze visual content in different ways based on inputs and user choices	Serge's github: Azure-CognitiveServices-Labs/images at main · retkowsky/Azure-CognitiveServices-Labs (github.com)	Cognitive Services: Azure's Computer Vision service **Estimated time**: 15-20 min	Call from local environment with Python
OCR	Serge's github: Azure-CognitiveServices-Labs/images at main · retkowsky/Azure-CognitiveServices-Labs (github.com)	Azure OCR and Form Recognizer **Estimated time**: 15-20 min	Azure OCR: Local environment/Data bricks/Azure ML notebooks Form Recognizer: user interface
Part two		Using semi-supervised models	
Custom Vision	Data will be generated from webcam during the workshop	**Estimated time**: 30-40 min	• Train the model • Call via API • Export model to TensorFlow • Create a python app based on the model • Visualize the model using tensor board
Part three		Building fully custom models	
Data augmentation	Serge's github: Azure-CognitiveServices-Labs/images at main · retkowsky/Azure-CognitiveServices-Labs (github.com)	Data augmentation using python libraries **Estimated time**: 15-20 min	Local environment or Azure ML environment
Image classifier	Serge's github: Azure-CognitiveServices-Labs/images at main · retkowsky/Azure-	Train image classifier	Azure ML Designer

	CognitiveServices-Labs (github.com) Animal images dataset available on Azure ML	**Estimated time**: 20-30 min	