



Sentiment Prediction

Real-time in SQL Server

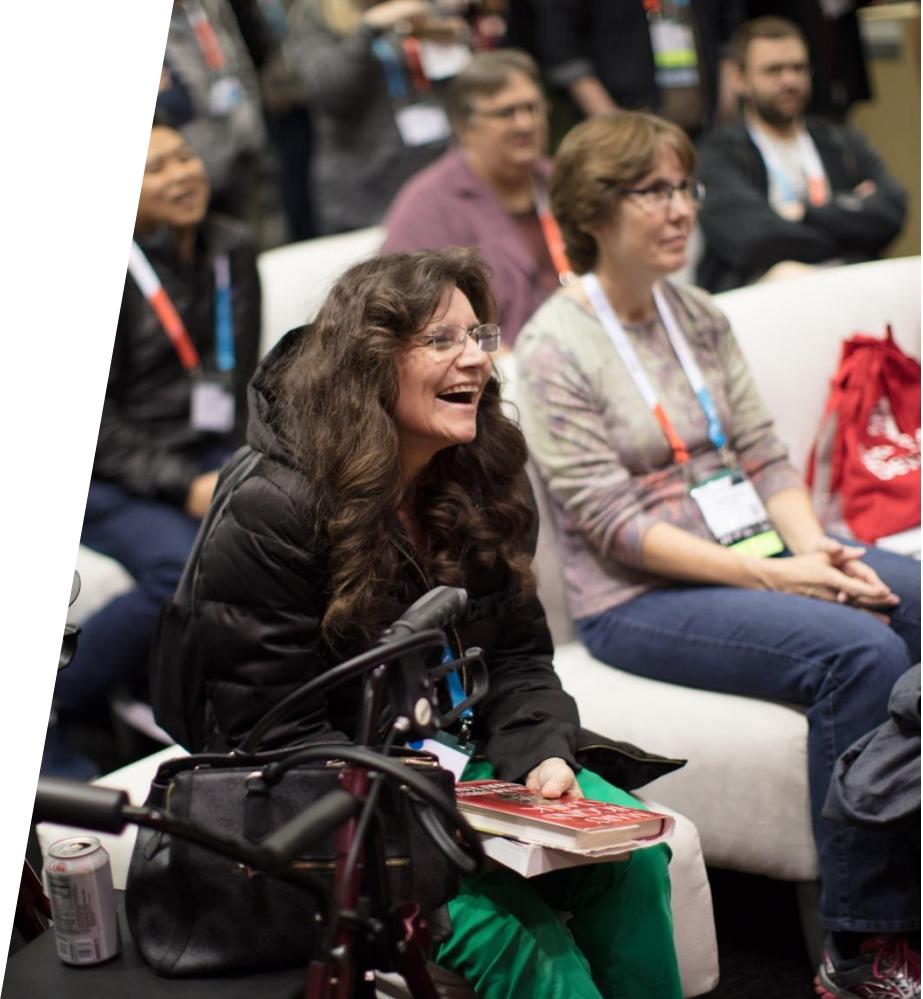
Hiram Fleitas

Download: github.com/hfleitas/seattle2019





Please silence cell phones



Explore everything PASS has to offer

Free Online Resources
Newsletters
PASS.org



24HOURS
OF PASS

Free online
webinar events



PASS
LOCAL
GROUPS



PASS
SQLSATURDAY

Local user groups
around the world

Free 1-day local
training events



PASS
MARATHON

Online special
interest user groups



PASS
VIRTUAL
GROUPS



PASS
VOLUNTEERS

Business analytics
training

Get involved



DBA2.o

Get data fast!

[Let's Work Together](#) [Contact](#) [About](#) [Training](#) [AI \(Demo\)](#)
[PBI \(Demo\)](#)

BIG News – PASS Summit 2019!

07/03/2019

Hiram

[Leave a comment](#)

[Edit](#)

I'm very excited to announce that I've been accepted as a speaker at PASS Summit 2019. I'm extremely grateful for the opportunity and look forward to PASS Summit 2019.
Category Session Status General Session The Power Platform Game Changer Accepted General Session Real-time Sentiment Prediction in SQL Server Accepted See you in Seattle, WA... [Continue reading →](#)

Microsoft Data & AI South FL (Nov 2019) Azure Data Factory

10/24/2019

Hiram

[Leave a comment](#)

Azure Data Factory From On-Prem to Data Warehouse Agenda: Azure Data Factory Data store Connectors 80x On-Premises DevOps Automation Notifications Resources by: Hiram Fleitas Twitter: @hiramfleitas Blog: <http://fleitasarts.com> Video Recording <https://youtu.be/TXxssUeqsOA> Resources fleitasarts.com github.com/hfleitas/rig netflix.com/trollhunters Azure Data Factory: GitHub Integration Azure Data Factory: Documentation Azure Data Factory: Copy Activity Performance Azure Data Factory: Connector Azure SQL Data Warehouse Connector... [Continue reading →](#)



Link to Poll:

bit.ly/mssql2019

Your participation is very important.

Session Evaluations

Submit by 5pm Friday,
November 15th to
win prizes.

3 WAYS TO ACCESS



Go to PASSsummit.com



Download the GuideBook App
and search: PASS Summit 2019



Follow the QR code link on session
signage



SCAN ME

Hiram Fleitas

Principal Database Architect,
Universal Property Casualty & Insurance Company



Bio

Father
Developer since 1995
SQL Server since 1999
Fleitas Arts, LLC since 2008
USCG Auxiliary Flotilla Staff Officer



 @hiramfleitas



 /hfeitas

 hiram@fleitasarts.com

 hiramfleitas

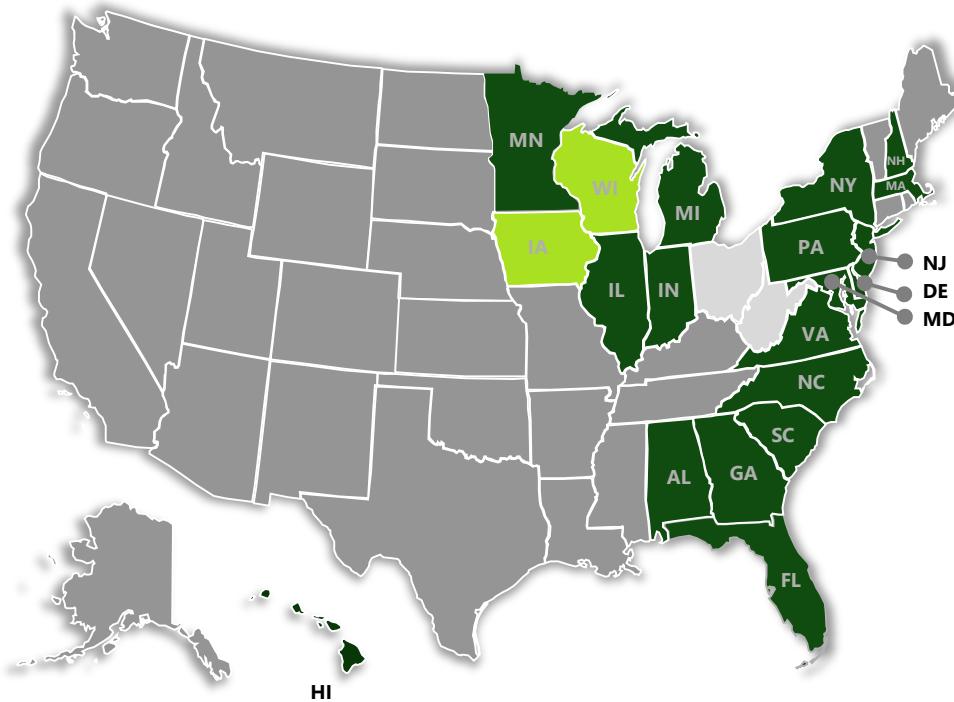
 fleitasarts.com



Universal Insurance Holdings (UVE): Growth Story

Doing business

Licensed



Leading holding company of personal residential
homeowners insurance and services

\$1B+

FY18 Direct Premium Written

#1

Best-in-class Return on Equity
(5-yr Average)

18

States actively doing business

>800K

Customer policies managed

800+

Full time equivalent employees

9,000+

Independent agents in the
distribution channel

Ft. Lauderdale, FL

Headquarters

A

Exceptional

Demotech financial stability rating



Agenda

1. Add ML Features
2. Grant Access & Config
3. Install Pre-Trained Models
4. Code in Python & T-SQL
5. Python Profiling
6. Real-time Scoring
7. Review Sentiment Results

Agenda

8. HammerDB 2016 vs 2017 vs 2019
9. Cognitive API
- 10.PDF
- 11.Paginated Report
- 12.Streaming
- 13.Spark

DEMO

one

- Add ML Features
- Grant Access & Configure
- Install Pre-Trained Models





Extract information from your text

Use the demo below to experiment with the Text Analytics API. Pick one of our examples or provide your own. Identify the language, sentiment, key phrases, and entities (Preview) of your text by clicking "Analyze".

See it in action

Destiny is a gift. Some go their entire lives, living existence as a quiet desperation. Never learning the truth that what feels as though a burden pushing down upon our shoulders, is actually, a sense of purpose that lifts us to greater heights. Never forget that fear is but the precursor to valor, that to strive and triumph in the face of fear, is what it means to be a hero. Don't think, Master Jim. Become!



Analyze

Analyzed text JSON

KEY PHRASES: face of fear, existence, triumph, valor, sense of purpose, entire lives, quiet desperation, shoulders, greater heights, precursor, Destiny, gift, Master Jim, burden, truth, hero

SENTIMENT:

78 %

NAMED ENTITIES: Destiny (1921 film) [Other]
valor [Location]
Don't (Billy Currington song) [Other]
Mastering (audio) [Other]
Jim [Person]

LINKED ENTITIES: Destiny is a gift. Some go their entire lives, living existence

Feature Selection

Select the Developer features to install.

- Global Rules
- Product Updates
- Install Setup Files
- Install Rules
- Installation Type
- Product Key
- License Terms
- Feature Selection**
 - Feature Rules
 - Instance Configuration
 - Server Configuration
 - Database Engine Configuration
 - Consent to install Microsoft R ...
 - Consent to install Python
 - Feature Configuration Rules
 - Ready to Install
 - Installation Progress
 - Complete

Looking for Reporting Services? [Download it from the web](#)

Features:

Instance Features

- Database Engine Services
- SQL Server Replication
- Machine Learning Services and Language Extensions
 - R
 - Python
 - Java
- Full-Text and Semantic Extractions for Search
- Data Quality Services
- PolyBase Query Service for External Data
 - Java connector for HDFS data sources

Feature description:

Includes extensions that enable integration with R, Python, Java and other programming languages using standard T-SQL statements.

Prerequisites for selected features:

Already installed:

- Windows PowerShell 3.0 or higher
- Microsoft Visual C++ 2017 Redistributable

Disk Space Requirements

Drive C: 2271 MB required, 591040 MB available

Select All

Unselect All

Instance root directory:

C:\Program Files\Microsoft SQL Server\



Shared feature directory:

C:\Program Files\Microsoft SQL Server\



Shared feature directory (x86):

C:\Program Files (x86)\Microsoft SQL Server\

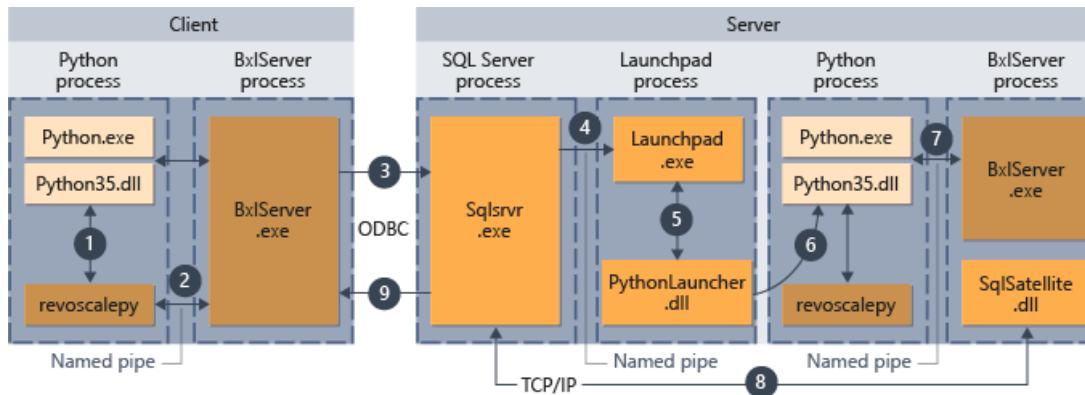
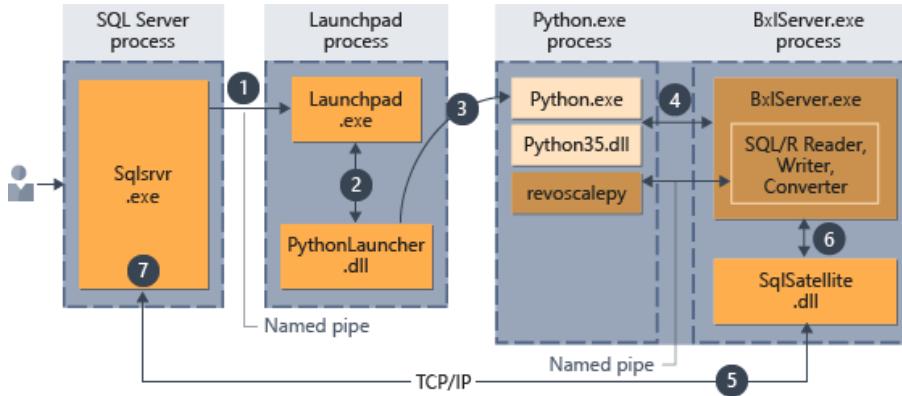


< Back

Next >

Cancel

Execution Architecture



Learn more:

<https://docs.microsoft.com/en-us/sql/advanced-analytics/concepts/extension-python>

<https://docs.microsoft.com/en-us/sql/advanced-analytics/install/sql-server-machine-learning-services-2019>

<https://docs.microsoft.com/en-us/sql/advanced-analytics/data-collection-ml-troubleshooting-process>



Computer Management (Local)	
System Tools	
> Task Scheduler	
> Event Viewer	
Shared Folders	
Shares	
Sessions	
Open Files	
Local Users and Groups	
Users	
Groups	
> Performance	
Device Manager	
Storage	
Disk Management	
> Services and Applications	

Name	Description	Actions
Access Control Assistance Operators	Members of this group can remotely query ...	Groups More Actions
Administrators	Administrators have complete and unrestricted access to all resources on the local computer and on remote computers.	Groups More Actions
Backup Operators	Backup Operators can override security restrictions to back up files and folders.	Groups More Actions
Cryptographic Operators	Members are authorized to perform cryptographic operations.	Groups More Actions
Distributed COM Users	Members are allowed to launch, activate and deactivate objects.	Groups More Actions
Event Log Readers	Members of this group can read event logs from the system.	Groups More Actions
Guests	Guests have the same access as members of the Guests group.	Groups More Actions
Hyper-V Administrators	Members of this group have complete and unrestricted access to all resources on the local computer and on remote computers.	Groups More Actions
IIS_IUSRS	Built-in group used by Internet Information Services.	Groups More Actions
Network Configuration Operators	Members in this group can have some administrative rights.	Groups More Actions
Performance Log Users	Members of this group may schedule log files.	Groups More Actions
Performance Monitor Users	Members of this group can access performance counter data.	Groups More Actions
Power Users	Power Users are included for backwards compatibility.	Groups More Actions
Remote Desktop Users	Members in this group are granted the right to connect to the computer via Remote Desktop.	Groups More Actions
Remote Management Users	Members of this group can access WMI resources.	Groups More Actions
Replicator	Supports file replication in a domain.	Groups More Actions
System Managed Accounts Group	Members of this group are managed by the Local Security Authority Subsystem Service (LSASS).	Groups More Actions
Users	Users are prevented from making accidental changes to system files.	Groups More Actions
docker-users	Users of Docker for Windows.	Groups More Actions
HelpLibraryUpdaters	SophosAdministrators may run Sophos Anti-Virus updates.	Groups More Actions
SophosAdministrator	Contains accounts used by Sophos Anti-Virus.	Groups More Actions
SophosOnAccess	SophosPowerUsers may run Sophos Anti-Virus.	Groups More Actions
SophosPowerUser	SophosUsers may run Sophos Anti-Virus without administrator privileges.	Groups More Actions
SophosUser	SophosAdministrators may run Sophos Anti-Virus with administrator privileges.	Groups More Actions
SQLRUserGroup	SQLRUserGroup	Groups More Actions
SQLServer2005SQLBrowserUser\$R90GTU6N	Members in the group have the required access.	Groups More Actions

SQLRUserGroup Properties

General

SQLRUserGroup

Description: SQLRUserGroup

Members:

- NT SERVICE\MSQLLaunchpad (S-1-5-80-3477044410-37626219...
- UPCICvhfiletas (S-1-5-21-366593668-1004689029-3092909805-167...

Changes to a user's group membership are not effective until the next time the user logs on.

Add... Remove OK Cancel Apply Help

```
18 exec sp_configure 'external scripts enabled', 1
19 reconfigure with override
20 go
21
22 declare @sql nvarchar(max);
23 -- only need to grant connect permissions!
24 select @sql = N'if not exists (select 1 from syslogins where name ='''+ @@servername +'\\SQLRUserGroup'')
25 begin
26 |   create login ['+ @@servername +'\\SQLRUserGroup] from windows
27 end'
28 print @sql; exec sp_executesql @sql;
29 go

30 -- Restart SQL Service & LAUNCHPAD.
31 -- Run PS as admin: .\Install-MLModels.ps1 MSSQLSERVER
32 -- Install Latest SQL Server CU, Reboot.
33 -- Run CMD as admin: FixPath.cmd
34 -- Verify WORKING_DIRECTORY in ...\\MSSQL\\Binn\\pythonlauncher.config
35 -- Run CMD as admin: AddToSQL-PreTrainedModels.cmd. It downloads & installs the pre-trained models.
```

Administrator: Windows PowerShell + ▾

```
PS C:\fleitasarts\SentimentPrediction\PS-CMD> .\Install-MLModels.ps1 MSSQLSERVER
WARNING: RSetup.exe not found for instance name= SHARED_SQL2017, sql version=140
WARNING: RSetup.exe not found for instance name= SHARED_SQL2019CTP3.0, sql version=150
INFO: processing instance MSSQL15.MSSQLSERVER
    Verifying R models [9.4.7]
    Installing R models [C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\MicrosoftML\mxLibs\x64]
SUCCESS installed R for MSSQL15.MSSQLSERVER
    Verifying Python models [9.4.7]
    Installing Python models [C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mxLibs]
SUCCESS installed Python for MSSQL15.MSSQLSERVER
PS C:\fleitasarts\SentimentPrediction\PS-CMD>
```

```
Administrator: Windows PowerShell X Administrator: cmd X + - ▾
c:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\SQL2019RC1\x64>
c:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\SQL2019RC1\x64>RSetup.exe /Install /component MLM /version 9.4.7.0 /language 1033 /destdir "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs"
Reading registry value HKCU\SOFTWARE\Microsoft\RSetup\MLM_DownloadUrl
Registry value:
RSetup.exe version: 9.4.7.3
Reading registry value HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\150\Bootstrap\Setup\R_SERV_CACHE
Registry value:
Reading registry value HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\150\Bootstrap\BootstrapDir
Registry value: C:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\
Searching for cache dir: C:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\\R_SERV_CACHE
Using default cache directory: C:\Users\hfleitas\AppData\Local\Temp\
Extracting C:\Users\hfleitas\AppData\Local\Temp\MLM_9.4.7.0_1033.cab to C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\
Extracting C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\AlexNet_Updated.model
Extracting C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\ImageNet1K_mean.xml
Extracting C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\pretrained.model
Extracting C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\ResNet_101_Updated.model
Extracting C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\ResNet_18_Updated.model
Extracting C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\Lib\site-packages\microsoftml\mlLibs\ResNet_50_Updated.model
c:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\SQL2019RC1\x64>
```

```
1 create or alter proc GetSentiment (
2     @text nvarchar(max)
3 )
4 as
5     declare @script nvarchar(max);
6     set @script = N'import pandas as p
7 from microsoftml import rx_featurize, get_sentiment
8 analyze_this = text
9 text_to_analyze = p.DataFrame(data=dict(Text=[analyze_this]))
10 sentiment_scores = rx_featurize(data=text_to_analyze, ml_transforms=[get_sentiment(cols=dict(scores="Text"))])
11 sentiment_scores["Sentiment"] = sentiment_scores.scores.apply(lambda score: "Positive" if score > 0.6 else "Negative")
12 exec sp_execute_external_script @language = N'Python',
13                             @script = @script,
14                             @output_data_1_name = N'sentiment_scores',
15                             @params = N'@text nvarchar(max)',
16                             @text = @text
17 with result sets (("Text" nvarchar(max), "Score" float, "Sentiment" nvarchar(30)));
18 go
```

1 exec dbo.GetSentiment N'These are not a normal stress reliever. First of all, they got sticky, hairy and dirty on
2 exec dbo.GetSentiment N'These are the cutest things ever!! Super fun to play with and the best part is that it la
3 exec dbo.GetSentiment N'I really did not like the taste of it'
4 exec dbo.GetSentiment N'It was surprisingly quite good!'
5 exec dbo.GetSentiment N'I will never ever ever go to that place again!!'
6 exec dbo.GetSentiment N'Destiny is a gift. Some go their entire lives, living existence as a quiet desperation. N

Total execution time: 00:00:30.283

	Text	Score	Sentiment
1	These are not a normal stres...	0.424483060836792	Negative

	Text	Score	Sentiment
1	These are the cutest things ...	0.869342148303986	Positive

	Text	Score	Sentiment
1	I really did not like the ta...	0.46178987622261	Negative

	Text	Score	Sentiment
1	It was surprisingly quite go...	0.960192441940308	Positive

	Text	Score	Sentiment
1	I will never ever ever go to...	0.310343533754349	Negative

	Text	Score	Sentiment
1	Destiny is a gift. Some go t...	0.5	Negative

DEMO

two

- Requirements.txt
- Common Vulnerabilities
- Python Profiling



The screenshot shows the Microsoft Visual Studio interface with the following details:

- Top Bar:** File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, Search (Ctrl+Q), SentimentML.
- Solution Explorer:** Shows the solution 'SentimentML' (1 of 1 project) containing Solution Items, Performance1.psess, Python, and Python Environments. A context menu is open over the Python Environments node, with the "Generate requirements.txt" option highlighted in green.
- Python Environments:** A pane titled "SQLServer2019" showing configuration details:
 - Configure (selected)
 - Packages (Conda)
 - Packages (PyPI)

Configuration fields include:

 - Description: SQLServer2019
 - Prefix path: C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES
 - Interpreter path: C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\python.exe
 - Windowed interpreter: C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\pythonw.exe
 - Language version: 3.7
 - Architecture: 64-bit
 - Path environment variable: PYTHONPATH

IT WON'T FAIL BECAUSE OF ME

hfiletas/SentimentPrediction: Pyt x +

github.com/hfiletas/SentimentPrediction Paused

Search or jump to... Pull requests Issues Marketplace Explore

hfiletas / SentimentPrediction Unwatch 1 Star 1 Fork 2

Code Issues 1 Pull requests 3 Projects 0 Wiki Security Insights Settings

Python and T-SQL solution for Sentiment analysis and Real-time predictions in SQL Server 2017+ <https://dba2o.wordpress.com>

sql-server python sentiment-analysis prediction tpcxbb-1gb-database microsoft Manage topics

162 commits 4 branches 0 packages 0 releases 1 contributor

⚠ We found potential security vulnerabilities in your dependencies.

Only the owner of this repository can see this message.

View security alerts

Branch: master New pull request Create new file Upload files Find file Clone or download

hfiletas fixed ps1 Latest commit c3a1397 3 days ago

PASSInsights201908-Dev fixed trollhunter notebook 2 months ago

PS-CMD fixed ps1 3 days ago

Python fix for upgrading py packages 5 months ago

Raffle States 5 months ago

SQL fixed sqlrusergroup permissions, added GetCognitiveAPIQuoteSentiment 2 months ago

.gitattributes Add .gitignore and .gitattributes. 2 years ago

.gitignore added ML.pdf 27 days ago

AIForBusiness.xcf deck updated, do cog stuff next 5 months ago

PASS

[hfleitas / SentimentPrediction](#)

Unwatch Star 1 Fork 2

[Code](#)[Issues 1](#)[Pull requests 3](#)[Projects 0](#)[Wiki](#)[Security](#)[Insights](#)[Settings](#)[Alerts](#)[Advisories](#)[Policy](#)

You are in the [Beta](#) of automated security fixes. [Opt out](#) or give us feedback.

[Learn more about automated security fixes.](#)

Security Alerts

[Automated security fixes](#) [Dismiss all](#)

3 Open ✓ 1 Closed

pillow

10 days ago by GitHub Python/requirements.txt #4

low severity

nltk

Aug 23, 2019 by GitHub Python/requirements.txt #3

high severity

werkzeug

Aug 21, 2019 by GitHub Python/requirements.txt #2

high severity

GitHub tracks known security vulnerabilities in some dependency manifest files. [Learn more about security alerts.](#)



Admin cmd or PowerShell:

```
python.exe -m pip install -U pkgname==x.y.z  
python.exe -m pip list
```

Python Environments ✎ X

Add Environment...

SQLServer2019

Overview werkzeug

Configure

Packages (Conda)

Packages (PyPI) **Werkzeug (0.14.1)**

Run command: pip install werkzeug

Install aiohttp-werkzeug
Install bottle-werkzeug
Install dataflake.wsgi.werkzeug
Install django-werkzeug
Install django-werkzeug-de
Install fspot-werkzeug
Install pylint-werkzeug
Install s3werkzeugcache
Install Werkzeug-ASGI
Install werkzeug-encrypted
Install werkzeug-graphql
Install Werkzeug-Raw
Install werkzeug-rfc7xx
Install Werkzeug (0.16.0)
Install yafowil.werkzeug

Visual Studio - Python support

Administrator privileges may be required to install, update or remove packages for this environment.

Elevate now
You may be prompted for credentials

→ Continue without Administrator privileges
This operation may fail if you do not have sufficient permissions

Always elevate when installing or removing packages
This dialog will not be shown again

Cancel

SQLServer2019

Custom environment

Details

CVE-2019-7164 [🔗](#)

Vulnerable versions: < 1.3.0
Patched version: 1.3.0

SQLAlchemy through 1.2.17 and 1.3.x through 1.3.0b2 allows SQL Injection via the order_by parameter.

CVE-2019-7548 [🔗](#)

Vulnerable versions: < 1.3.0
Patched version: 1.3.0

SQLAlchemy 1.2.17 has SQL Injection when the group_by parameter can be controlled.

CVE-2019-14751 [🔗](#)

high severity

Vulnerable versions: < 3.4.5

Patched version: 3.4.5

NLTK Downloader before 3.4.5 is vulnerable to a directory traversal, allowing attackers to write arbitrary files via a .. (dot dot slash) in an NLTK package (ZIP archive) that is mishandled during extraction.

CVE-2019-14806 [🔗](#)

high severity

Vulnerable versions: < 0.15.3
Patched version: 0.15.3

Pallets Werkzeug before 0.15.3, when used with Docker, has insufficient debugger PIN randomness because Docker containers share the same machine id.

CVE-2019-16865 [🔗](#)

low severity

Vulnerable versions: < 6.2.0
Patched version: 6.2.0

An issue was discovered in Pillow before 6.2.0. When reading specially crafted invalid image files, the library can either allocate very large amounts of memory or take an extremely long period of time to process the image.

GetSentimentExample - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test R Tools Analyze Window Help

Launch Python Profiling... Alt+F2

Python Performance GetSentimentExample_20180606.vsp X

Current View: Summary

Instrumentation Profiling Report
22.297 seconds of total execution time

Hot Path

Function Name	Elapsed Inclusive Time %	Elapsed Exclusive Time %
python.exe	100.00	0.00
GetSentimentExample (module)	100.00	0.00
<frozen importlib._find_and_load	72.15	0.00
microsoftml.modules.feature(rx_featurize	27.80	0.00

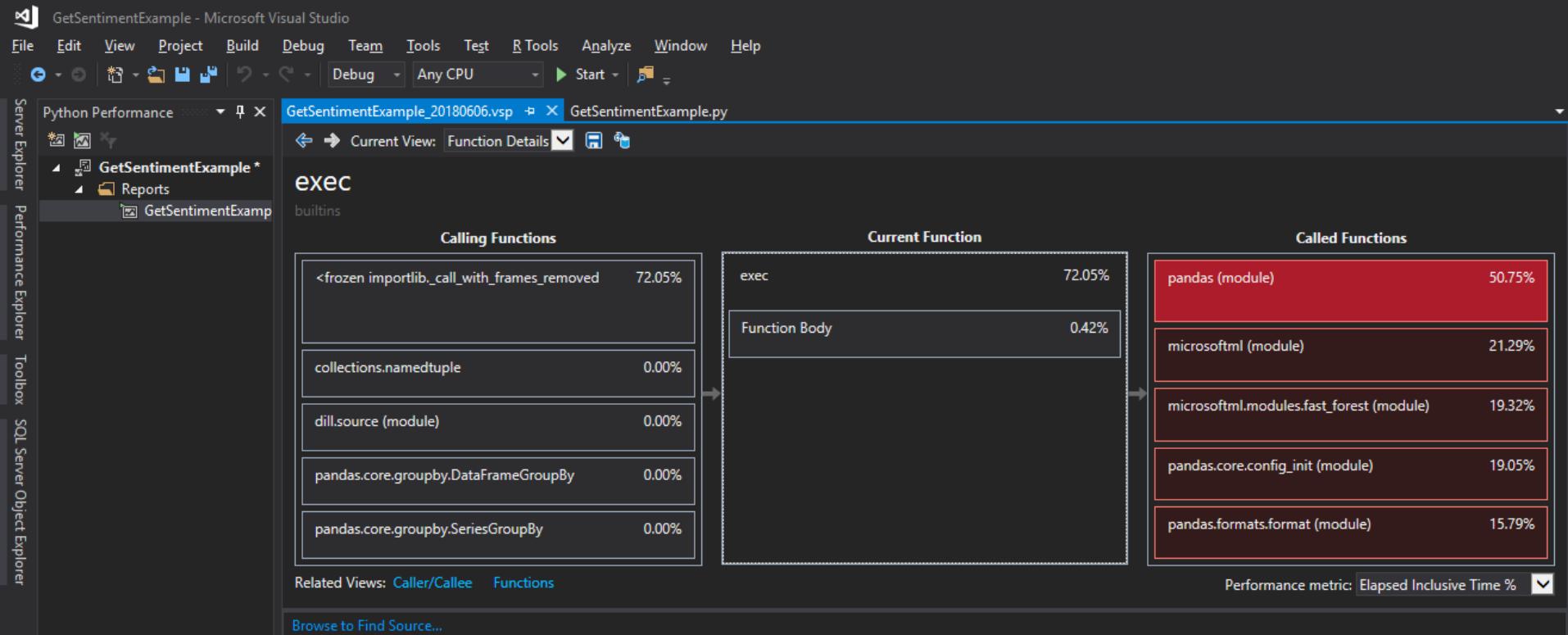
Related Views: Call Tree Functions

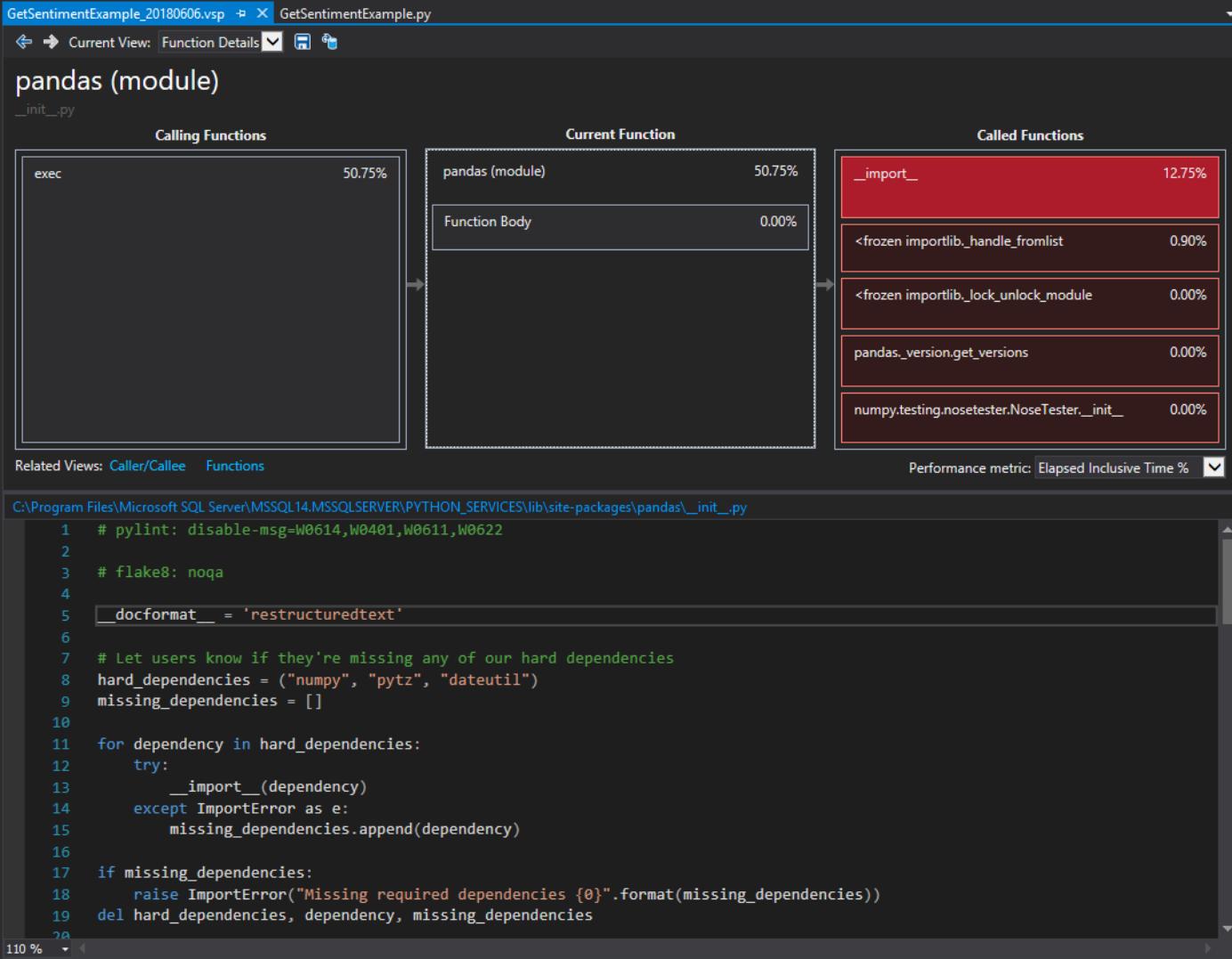
Functions Doing Most Individual Work

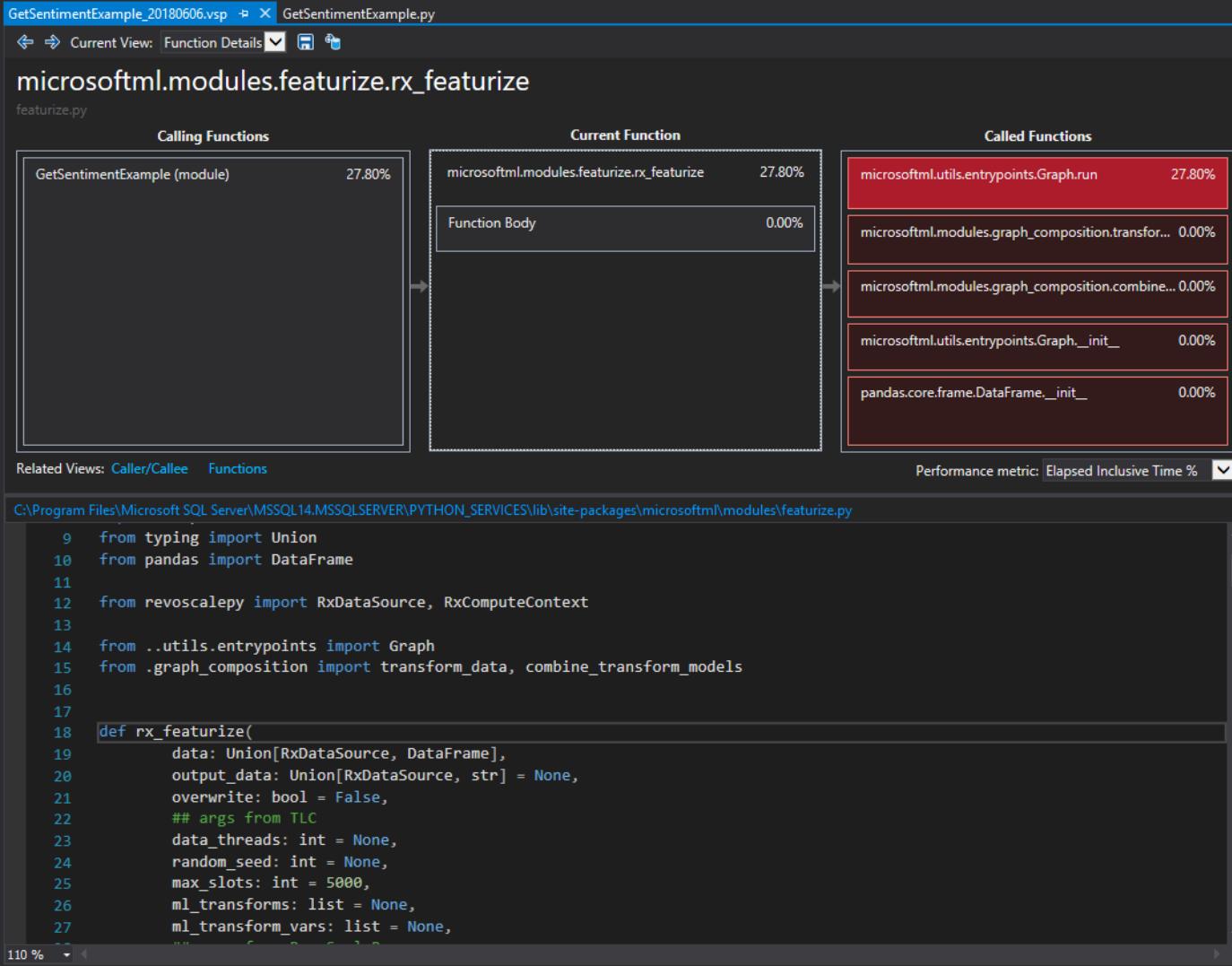
Name	Exclusive Time %
<frozen importlib.FileLoader.get_data	47.54
revoscalepy.RxSerializable.rx_native_call	27.88
compile	9.22
io.open	3.57
_imp.create_dynamic	3.32

Report

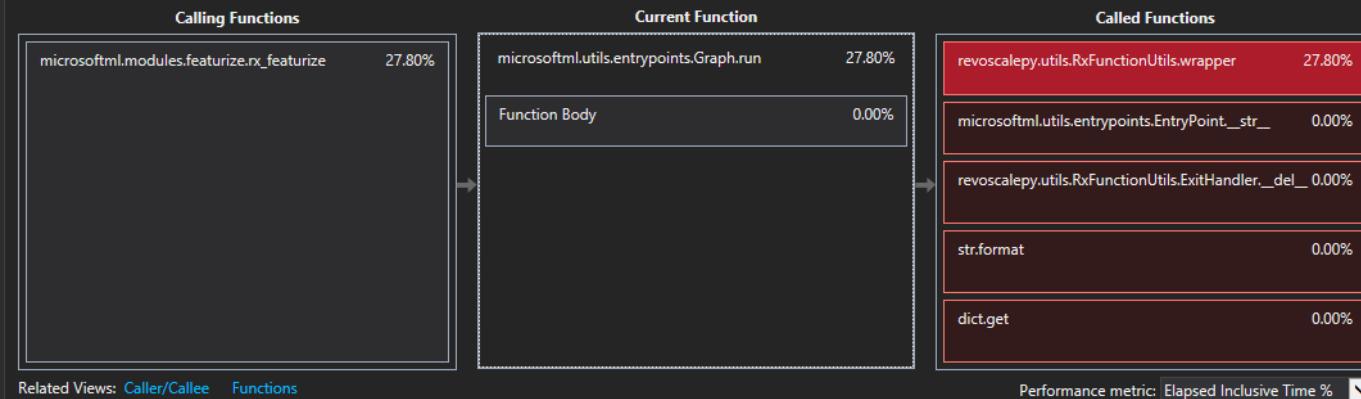
- Show Trimmed Call Tree
- Compare Reports...
- Export Report Data...
- Save Analyzed Report...
- Toggle Full Screen
- Set Symbol Path...



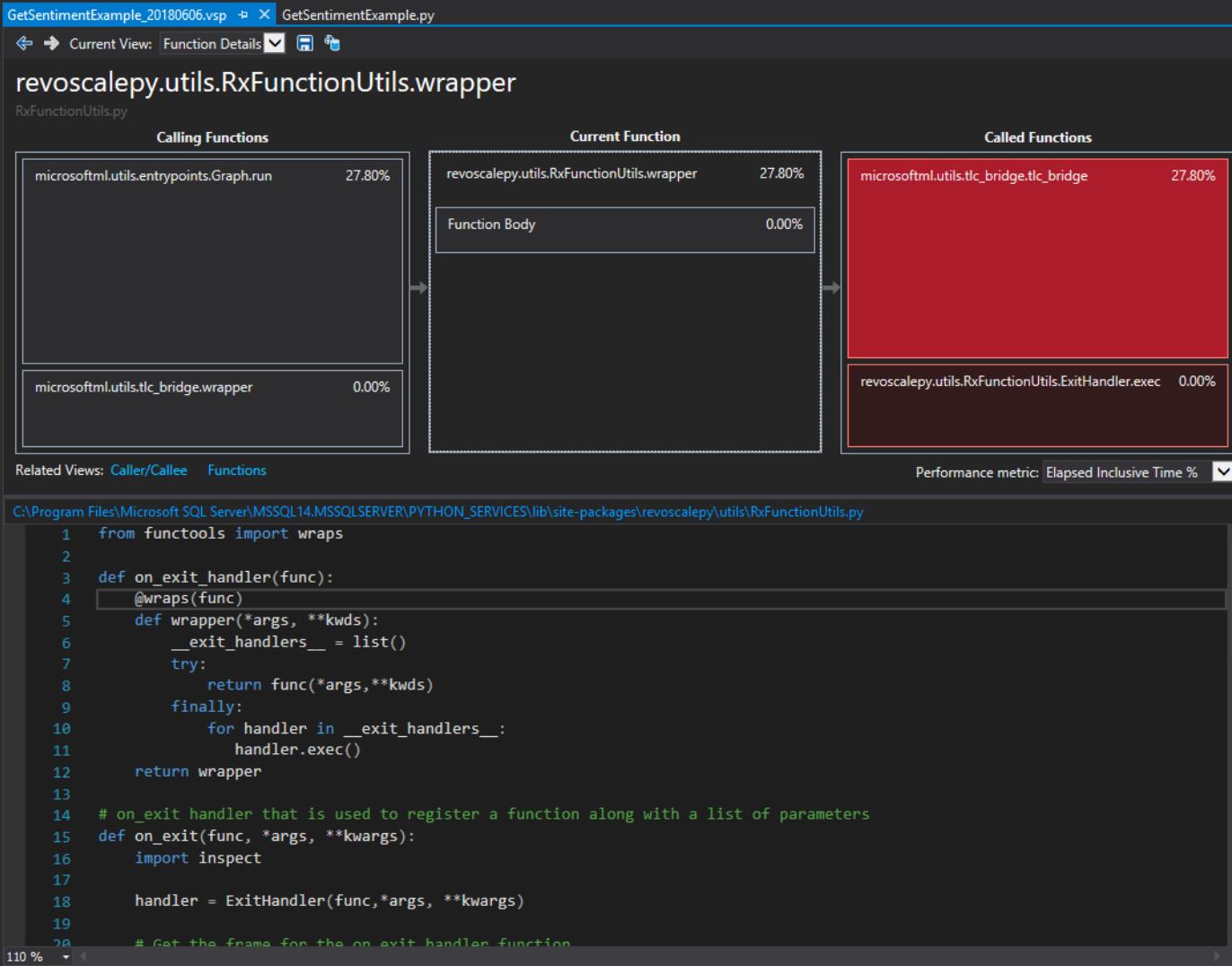


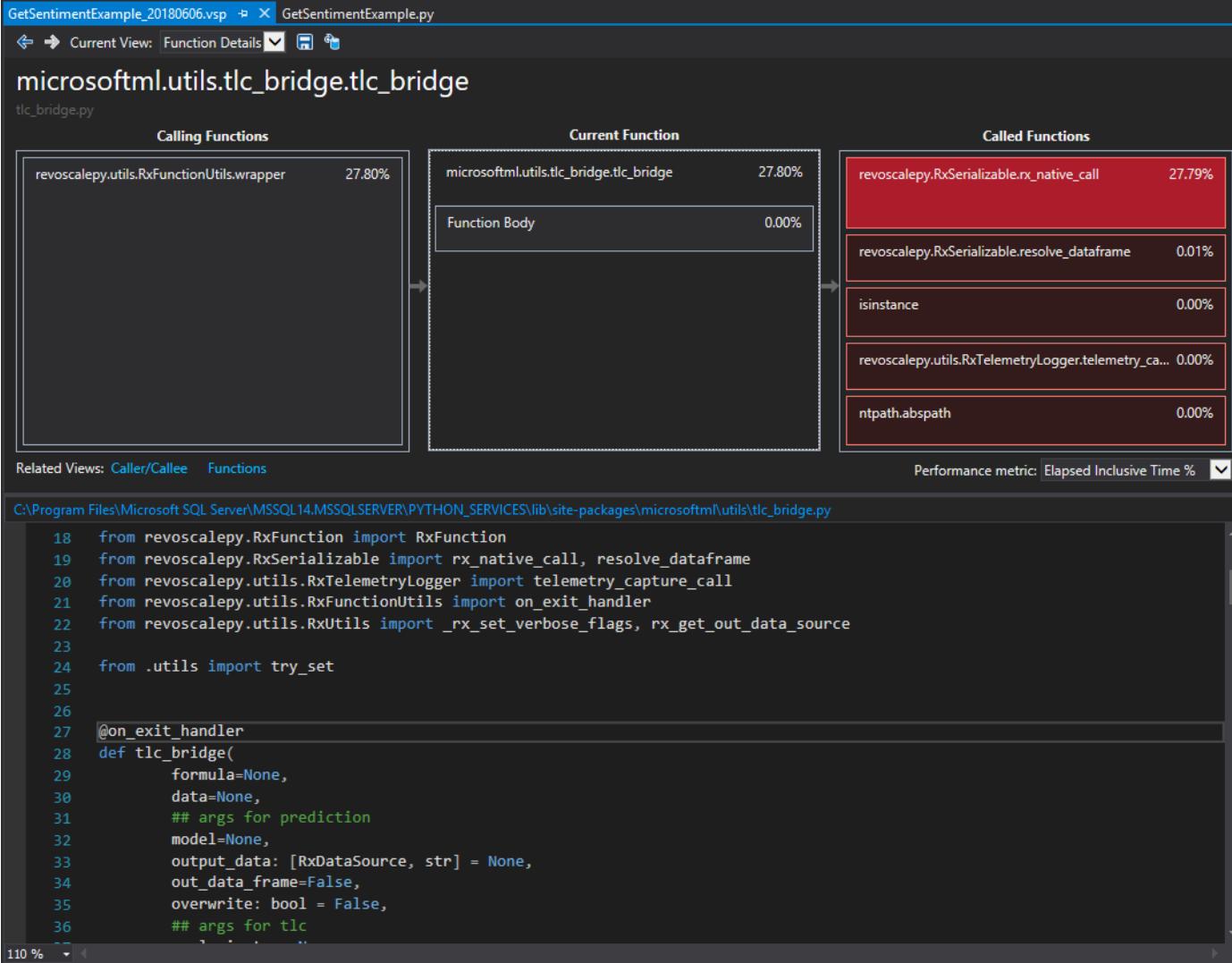


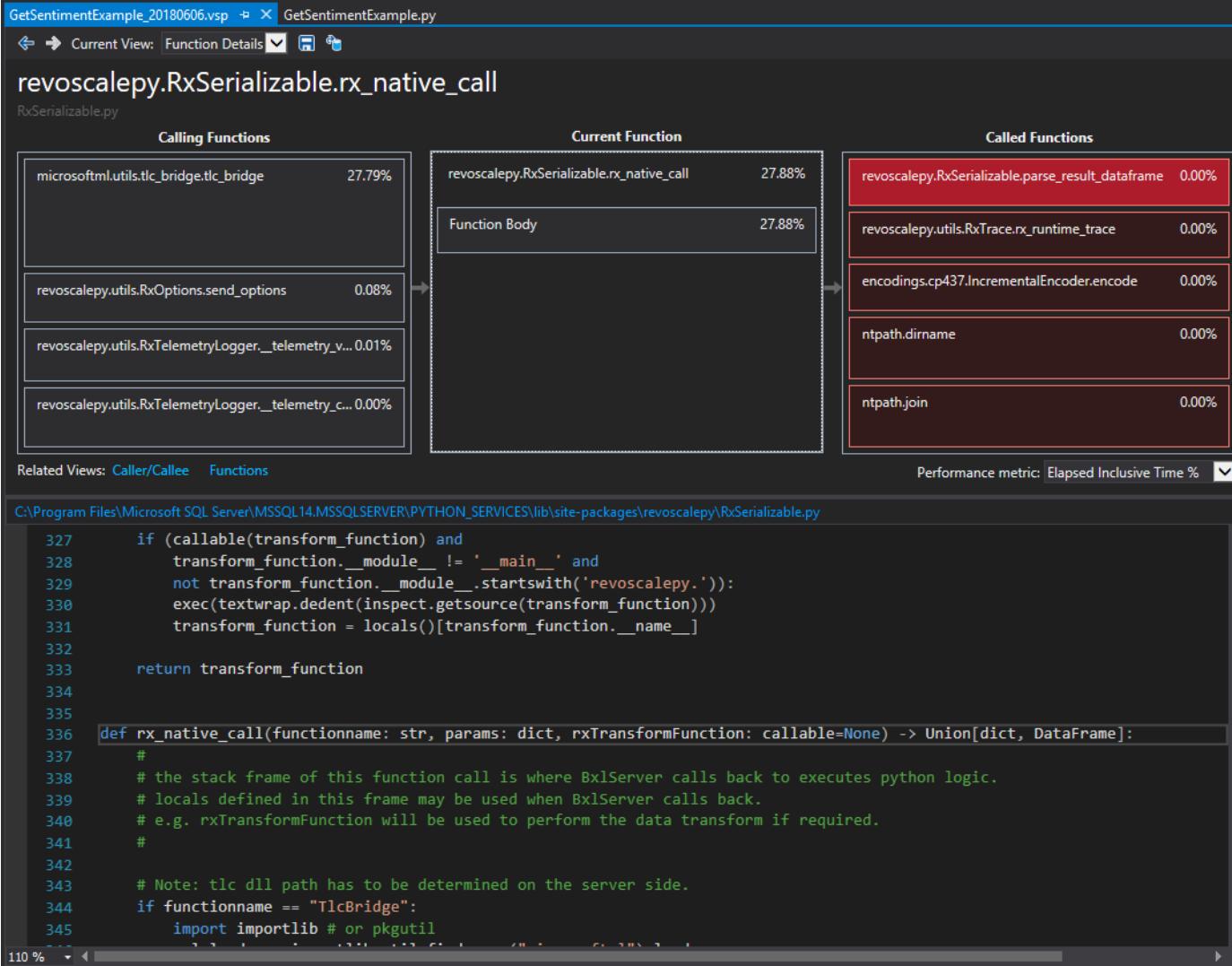
microsoftml.utils.entrypoints.Graph.run



C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\PYTHON_SERVICES\lib\site-packages\microsoftml\utils\entrypoints.py







DEMO

three

- In-Database ML
- realtime_scoring_only = True
- sp_rxPredict
- Performance Metrics



```
101  DROP TABLE IF EXISTS [dbo].[models]
102  GO
103  CREATE TABLE [dbo].[models](
104      [language] [varchar](30) NOT NULL,
105      [model_name] [varchar](30) NOT NULL,
106      [model] [varbinary](max) NOT NULL,
107      [create_time] [datetime2](7) NULL DEFAULT (sysdatetime()),
108      [created_by] [nvarchar](500) NULL DEFAULT (suser_sname()),
109      PRIMARY KEY CLUSTERED ( [language], [model_name] )
110  )
111  GO
```

```
118  CREATE OR ALTER VIEW product_reviews_training_data
119  AS
120  SELECT TOP(CAST( ( SELECT COUNT(*) FROM product_reviews)*.9 AS INT))
121      CAST(pr_review_content AS NVARCHAR(4000)) AS pr_review_content,
122      CASE
123          WHEN pr_review_rating <3 THEN 1
124          WHEN pr_review_rating =3 THEN 2
125      ELSE 3 END AS tag
126  FROM product_reviews;
127  GO
```

```
129  CREATE OR ALTER VIEW product_reviews_test_data
130  AS
131  SELECT TOP(CAST( ( SELECT COUNT(*) FROM product_reviews)*.1 AS INT))
132      CAST(pr_review_content AS NVARCHAR(4000)) AS pr_review_content,
133      CASE
134          WHEN pr_review_rating <3 THEN 1
135          WHEN pr_review_rating =3 THEN 2
136      ELSE 3 END AS tag
137  FROM product_reviews;
138  GO
```

Learn more:

<https://blogs.msdn.microsoft.com/sqlserverstorageengine/2017/11/01/sentiment-analysis-with-python-in-sql-server-machine-learning-services>

https://sqlchoice.blob.core.windows.net/sqlchoice/static/tpcxbb_1gb.bak



```

142 CREATE OR ALTER PROCEDURE [dbo].[create_text_classification_model]
143 AS
144 BEGIN
145     DECLARE @model varbinary(max), @train_script nvarchar(max);
146     --The Python script we want to execute
147     SET @train_script = N'
148         ##Import necessary packages
149         from microsoftml import rx_logistic_regression,featurize_text, n_gram
150         import pickle
151         ## Defining the tag column as a categorical type
152         training_data["tag"] = training_data["tag"].astype("category")
153
154         ## Create a machine learning model for multiclass text classification.
155         ## We are using a text featurizer function to split the text in features of 2-word chunks
156
157         #ngramLength=2: include not only "Word1", "Word2", but also "Word1 Word2"
158         #weighting="TfIdf": Term frequency & inverse document frequency
159         model = rx_logistic_regression(formula = "tag ~ features", data = training_data, method = "multiClass", ml_transforms=[
160             featurize_text(language="English",
161                         cols=dict(features="pr_review_content"),
162                         word_feature_extractor=n_gram(2, weighting="TfIdf")))
163
164         ## Serialize the model so that we can store it in a table
165         modelbin = pickle.dumps(model);
166
167     EXECUTE sp_execute_external_script
168         @language = N'Python'
169         , @script = @train_script
170         , @input_data_1 = N'SELECT * FROM product_reviews_training_data'
171         , @input_data_1_name = N'training_data'
172         , @params = N'@modelbin varbinary(max) OUTPUT'
173         , @modelbin = @model OUTPUT;
174         --Save model to DB Table
175         DELETE FROM dbo.models WHERE model_name = 'rx_logistic_regression' and language = 'Python';
176         INSERT INTO dbo.models (language, model_name, model) VALUES('Python', 'rx_logistic_regression', @model);
177 END;

```

Learn more:

<https://blogs.msdn.microsoft.com/sqlserverstorageengine/2017/11/01/sentiment-analysis-with-python-in-sql-server-machine-learning-services>

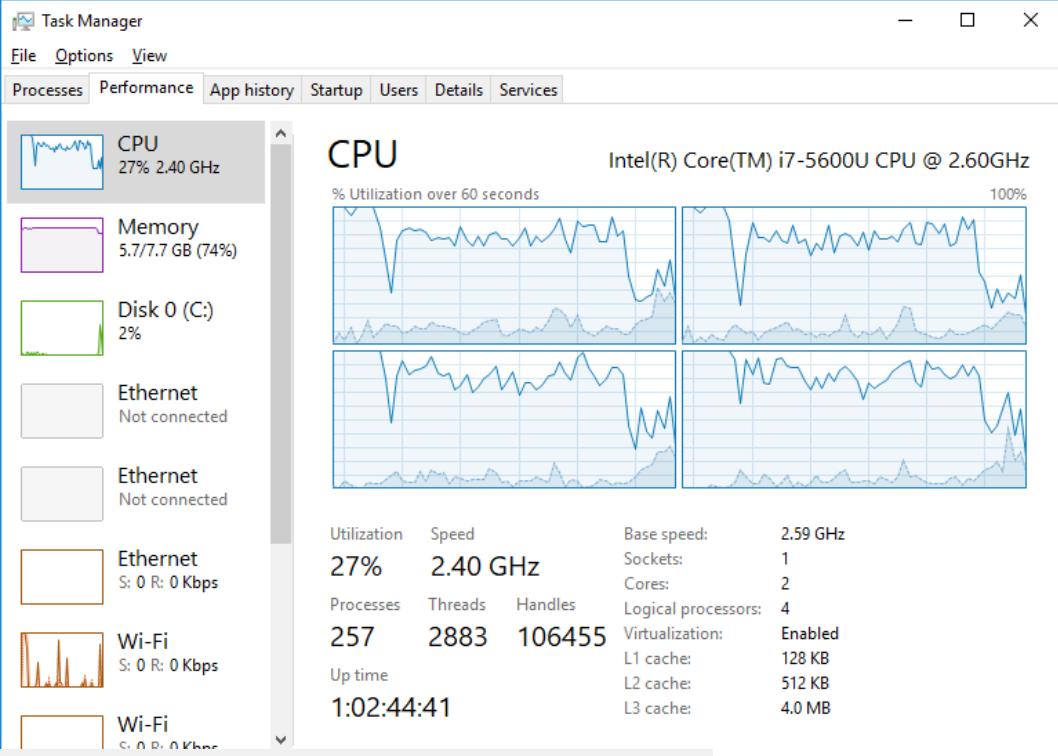
https://sqlchoice.blob.core.windows.net/sqlchoice/static/tpcxxb_1gb.bak



```

180 EXECUTE [dbo].[create_text_classification_model];
--Take a look at the model object saved in the model
182 SELECT * FROM dbo.models;
183 GO

```



Results Messages

language	model_name	model	create_time	created_by
Python	rx_logistic_regression	0x8003636D6963726F736F66746D6C2E6D6F64756C65732E...	2019-11-03 14:16:58.857	UPCIC\hfleitas

Total execution time
 SQL Server 2017: 00:01:35
 SQL Server 2019: 00:01:08

Query executed successfully.

(local) (15.0 RC1) | UPCIC\hfleitas (65) | tpcxbb_1gb | 00:01:08 | 1 rows

```
186 CREATE OR ALTER PROCEDURE [dbo].[predict_review_sentiment] AS
187 BEGIN
188     -- text classifier for online review sentiment classification (Positive, Negative, Neutral)
189     DECLARE @model_bin varbinary(max), @prediction_script nvarchar(max);
190     SELECT @model_bin = model from dbo.models WHERE model_name = 'rx_logistic_regression' and language = 'Python';
191
192     --The Python script we want to execute
193     SET @prediction_script = N'
194         from microsoftml import rx_predict
195         from revoscalepy import rx_data_step
196         import pickle
197
198         ## The input data from the query in @input_data_1 is populated in test_data
199         ## We are selecting 10% of the entire dataset for testing the model
200
201         ## Unserialize the model
202         model = pickle.loads(model_bin)
203
204         ## Use the rx_logistic_regression model
205         predictions = rx_predict(model = model, data = test_data, extra_vars_to_write = ["pr_review_content"], overwrite = True)
206
207         ## Converting to output data set
208         result = rx_data_step(predictions)
209
210         ## print(result)';
211
212     EXECUTE sp_execute_external_script
213         @language = N'Python'
214         , @script = @prediction_script
215         , @input_data_1 = N'SELECT * FROM product_reviews_test_data'
216         , @input_data_1_name = N'test_data'
217         , @output_data_1_name = N'result'
218         , @params = N'@model_bin varbinary(max)'
219         , @model_bin = @model_bin
220     WITH RESULT SETS (("Review" NVARCHAR(MAX), "PredictedLabel" FLOAT, "Predicted_Score_Negative" FLOAT, "Predicted_Score_Neutral" FLOAT, "Predicted_Score_Positive" FLOAT));
221 END
222 GO
```

229

EXECUTE [dbo].[predict_review_sentiment]

110 % ↑↓

T-SQL Results Message

	Review
1	Works fine. Easy to install. Some reviews talk about not fitting wall plates. Design great product to save money! Dont worry about leaving the light on anymore. It
2	Next time will go with the old metal handle- this is bonus.
3	Great Gift Great Value had to get used. And after 12 hours of use, they just thro
4	After trip to Paris and falling in love with Nutella crepes decided had to try it. am
5	Simply the best thing about them is that you can only use for one thing, so this o
6	This is the exact product that my mother used in the outlet/switch box. It does e
7	Not super magnet, but strong enough to set on the oven and the spatula is supp
8	Installed as bathroom fan timer. Easy to install. Some reviews talk about not fitting
9	Our home was built in 2003 and this fits just fine in the drawer until find one of th
10	Hi ,We are running pub here IN Marmaris Turkey Since long time we are looking
11	Terra cotta is the best!
12	One of my fingemail! It was very nicely made and the shaker has chance to han
13	We installed these on the fan to come on, and then the timer simply winds down
14	needed silicone coated whisk for cooking class and did not have time to get on
15	Great Gift Great Value really like the small quantity you get stranded, next to you
16	Laguiole knives are real hit with everyone, from kid...
17	Good sound timers that work as advertised.Intem...
18	AWESOME FEEDBACK FROM MY BEST FRIEND WHOM PURCHASED THI
19	love the retro glass look and says the styling makes it 100% easier to grate thing
20	love the product to save money! Dont worry about leaving the light was done??
21	AWESOME FEEDBACK FROM MY BEST FRIEND WHOM PURCHASED THI
22	Please can You send me of the plate supplied to fi...
23	Love this little grater. have used multiple Intematic...
24	Once the cork screw has started into the collection bin. have almost the entire s
25	excellent exactly what it's supposed to. We set it for...
26	The perfect timer to use with garlic lot and needed the salt grinder. This made p
27	Please can You send me of the Stars, thanks
28	glass globes would be perfect for bathroom fans as well

Query executed successfully at 3:07:58 PM

Results Messages

	Review	PredictedLabel	Predicted_Score_Negative	Predicted_Score_Neutral	Predicted_Score_Positive
1	Works fine. Easy to install. Some reviews talk abo...	3	0.0227801483124495	0.116715498268604	0.860504329204559
2	great product to save money! Dont worry about le...	3	0.00644614268094301	0.0120471781119704	0.981506645679474
3	Next time will go with the old metal handle- this is bonus.	3	0.025997199181374	0.0737180039286613	0.900284767150879
4	Great Gift Great Value had to get used. And after ...	3	0.0209195297211409	0.0431731045246124	0.935907244682312
5	After trip to Paris and falling in love with Nutella cre...	3	0.00612558424472809	0.00701949838548899	0.986854910850525
6	Simply the best thing about them is that you can o ...	3	0.0122916903346777	0.0282259378582239	0.955482371807098
7	This is the exact product that my mother used in th...	3	0.0128120584413409	0.0255865734070539	0.961601436138153
8	Not super magnet, but strong enough to set on the...	3	0.0264367256313562	0.0521628297865391	0.921400427818298
9	Installed as bathroom fan timer. Easy to install. So...	3	0.021312965080142	0.0903766602277756	0.888310253620148
10	Our home was built in 2003 and this fits just fine in ...	3	0.0157530382275581	0.044221280053101	0.940025627613068
11	Hi ,We are running pub here IN Marmaris Turkey S...	3	0.0192912872880697	0.0481909178197384	0.93251770734787
12	Terra cotta is the best!	3	0.00943857245147228	0.0192616432905197	0.971299707889557
13	One of my fingemail! It was very nicely made and t...	3	0.0126165235415101	0.0145046301186085	0.972878932952881
14	We installed these on the fan to come on, and the...	3	0.0215052589774132	0.0560405068099499	0.922454178333282
15	needed silicone coated whisk for cooking class an...	3	0.0275186914950609	0.058783922344462	0.913697242736816
16	Great Gift Great Value really like the small quantity...	3	0.00979716889560223	0.0222342982888222	0.96796840429306
17	Laguiole knives are real hit with everyone, from kid...	3	0.0337895713746548	0.0921972617506981	0.874013245105743
18	Good sound timers that work as advertised.Intem...	3	0.0178057551383972	0.0651285573840141	0.91705620422363
19	AWESOME FEEDBACK FROM MY BEST FRIEND WHOM PURCHASED THI	3	0.0269344560801983	0.0359061360359192	0.937159478664398
20	love the retro glass look and says the styling make...	3	0.01455203443748	0.0371924638748169	0.948254764080048
21	love the product to save money! Dont worry about le...	3	0.0599589124321938	0.0328954607248306	0.907145500183105
22	AWESOME FEEDBACK FROM MY BEST FRIEND WHOM PURCHASED THI	3	0.0269344560801983	0.0359061360359192	0.937159478664398
23	Please can You send me of the plate supplied to fi...	3	0.00973394140601158	0.0180833879858255	0.972182631492615
24	Love this little grater. have used multiple Intematic...	3	0.00598522322252393	0.00822614412754774	0.985788702964783
25	Once the cork screw has started into the collectio...	3	0.0135891530662775	0.0181965008378029	0.968214213848114
26	excellent exactly what it's supposed to. We set it for...	3	0.0345014269472073	0.0529183372855186	0.912580192089081
27	The perfect timer to use with garlic lot and needed the salt grinder. This made p	3	0.00933661591261625	0.0237888675183058	0.966874420642853
28	Please can You send me of the Stars, thanks	3	0.0234247632324696	0.047630175091743	0.928944826126099
29	glass globes would be perfect for bathroom fans a...	3	0.0136231174692512	0.0	

00:00:10 | 8,999 rows

Query executed successfully.

(local) (15.0 RC1) | UPCIC

	Predicted_Score_Neutral	Predicted_Score_Positive
0.116715498268604	0.860504329204559	
0.0120471781119704	0.981506645679474	
0.0737180039286613	0.900284767150879	
0.0431731045246124	0.935907244682312	
0.00701949838548899	0.986854910850525	
0.0282259378582239	0.959482371807098	
0.0255865734070539	0.961601436138153	
0.0521628297865391	0.921400427818298	
0.0903766602277756	0.888310253620148	
0.044221280053101	0.940025627613068	
0.0481909178197384	0.93251770734787	
0.0192616432905197	0.971299707889557	
0.015046301186085	0.972878932952881	
0.0560405068099499	0.922454178333282	
0.058783922344462	0.913697242736816	
0.0222342982888222	0.96796840429306	
0.0921972617506981	0.874013245105743	
0.0651285573840141	0.91705620422363	
0.0359061360359192	0.937159478664398	
0.0371924638748169	0.948254764080048	
0.0328954607248306	0.907145500183105	
0.0359061360359192	0.937159478664398	
0.0180833879858255	0.972182631492615	
0.00822614412754774	0.985788702964783	
0.0181965008378029	0.968214213848114	
0.0529183372855186	0.912580192089081	
0.0237888675183058	0.966874420642853	
0.047630175091743	0.928944826126099	
0.0290148910135031		

00:00:17 | 8999 rows

TU6N (14.0 RTM) | UPCIC

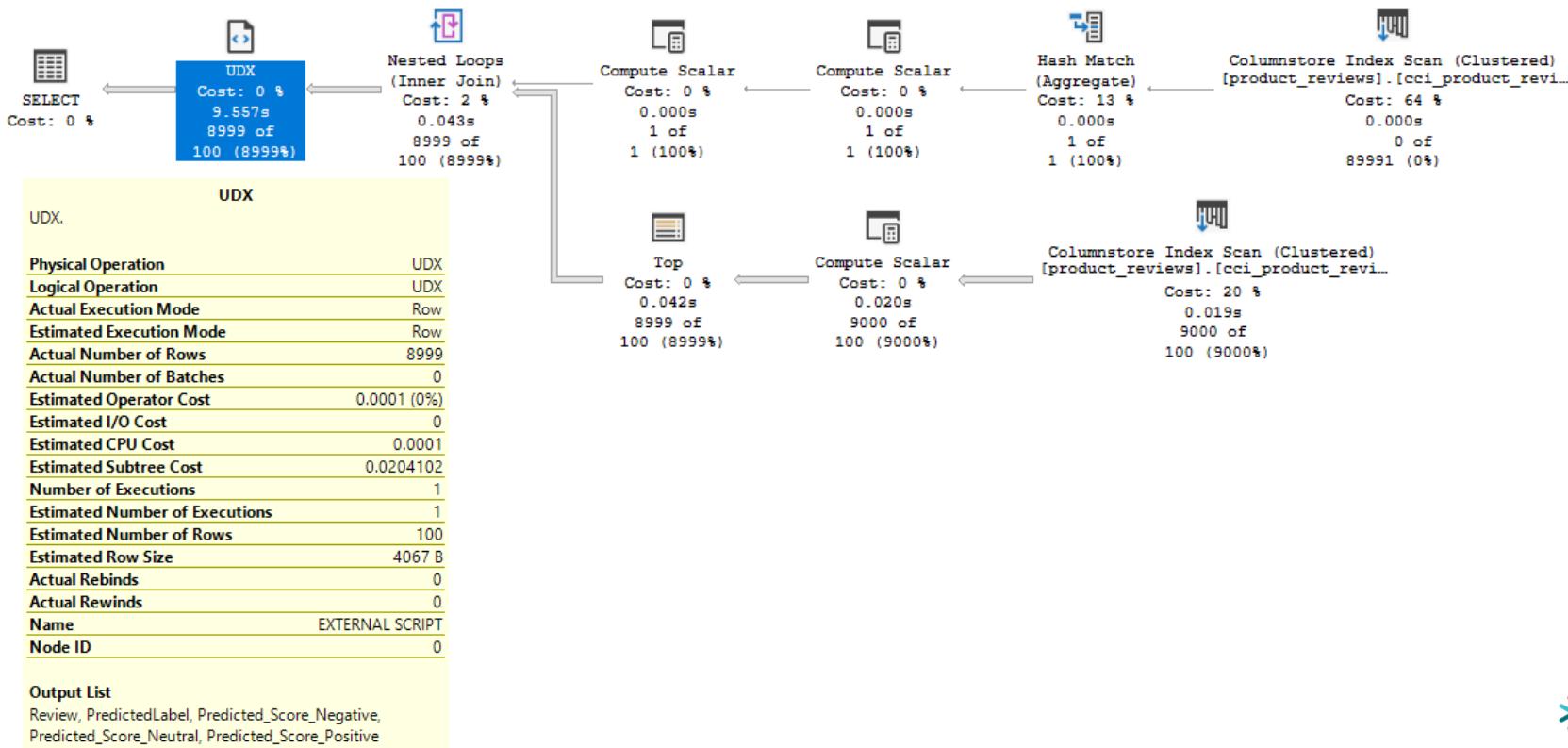


Total execution time

SQL Server 2017: 00:00:17

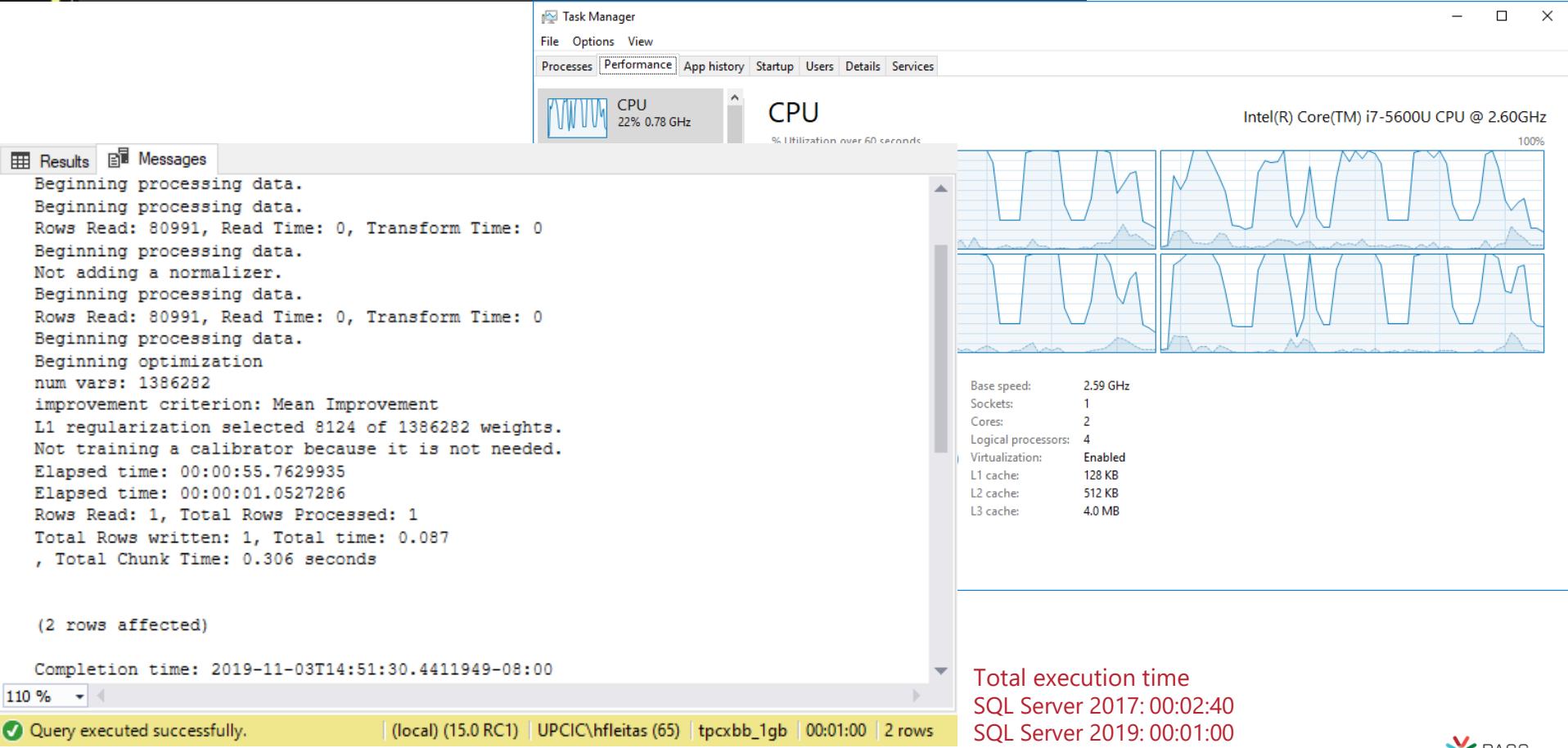
SQL Server 2019: 00:00:10

Execution Plan



```
254 CREATE OR ALTER PROCEDURE [dbo].CreatePyModelRealtimeScoringOnly AS
255 BEGIN
256     DECLARE @model varbinary(max), @train_script nvarchar(max);
257     --The Python script we want to execute
258     SET @train_script = N'
259         from microsoftml import rx_logistic_regression, featurize_text, n_gram
260         from revoscalepy import rx_serialize_model, RxOdbcData, rx_write_object, RxInSqlServer, rx_set_compute_context, RxLocalSeq
261         #import pickle
262
263         connection_string = "Driver=SQL Server;Server=localhost;Database=tpcxbb_1gb;Trusted_Connection=true;"
264         dest = RxOdbcData(connection_string, table = "models")
265
266         training_data["tag"] = training_data["tag"].astype("category")
267
268         #ngramLength=2: include not only "Word1", "Word2", but also "Word1 Word2"
269         #weighting="TfIdf": Term frequency & inverse document frequency
270
271         modelpy = rx_logistic_regression(formula = "tag ~ features",
272                                         data = training_data,
273                                         method = "multiClass",
274                                         ml_transforms=[featurize_text(language="English",
275                                             cols=dict(features="pr_review_content"),
276                                             word_feature_extractor=n_gram(2, weighting="TfIdf"))],
277                                         train_threads=1)
278
279         ## Serialize and write the model
280         modelbin = rx_serialize_model(modelpy, realtime_scoring_only = True)
281         #modelbin = pickle.dumps(model)
282         rx_write_object(dest, key_name="model_name", key="RevoMMLRealtimeScoring", value_name="model", value=modelbin, serialize=False, compress=None, overwrite=True);
283
284     EXECUTE sp_execute_external_script
285         @language = N'Python'
286         , @script = @train_script
287         , @input_data_1 = N'SELECT * FROM product_reviews_training_data'
288         , @input_data_1_name = N'training_data'
289
290     END;
291     GO
```

295 exec CreatePyModelRealtimeScoringOnly; --00:01:14.560 desktop, 00:02:40.351 laptop.



```
297 SELECT *, datalength(model) as Datalen FROM dbo.models; --(6MB w/rx_write_object vs 55MB w/pickle.dump)
```

Results Messages

	language	model_name	model	create_time	created_by	Datalen
1	Python	RevoMMLRealtimeScoring	0x626C6F62BA9FDBFE8DEB2F7451194F7092AE8C4147FCCD...	2019-11-03 14:51:29.700	UPCIC\hfleitas	6453907
2	Python	rx_logistic_regression	0x8003636D6963726F736F66746D6C2E6D6F64756C65732E6...	2019-11-03 14:16:58.857	UPCIC\hfleitas	55195076

Let's try sp_rxPredict

```
321 sp_configure 'show advanced options', 1;
322 [reconfigure;
323 go
324 sp_configure 'clr enabled', 1;
325 [reconfigure with override;
326 go
327 alter database tpcxbb_1gb set trustworthy on;
328 exec sp_changedbowner @loginame = sa, @map = false;
329 go
```



Enable Realtime Predictions (cmd as admin)



```
1 %%cmd
2 cd "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64\
3 RegisterRExt.exe /installRts
4 RegisterRExt.exe /installRts /database:tpcxbb_1gb
5 RegisterRExt.exe /installRts /database:FleitasArts
```

Learn more:

<https://docs.microsoft.com/en-us/sql/advanced-analytics/r/how-to-do-realtime-scoring>



Administrator: Windows PowerShell X Administrator: cmd + -

```
C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64>RegisterRExt.exe /installRts
Source directory to pick the RExtension binaries determined to be "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64\".
Installation flavor was not supplied
Connecting to SQL server...
Sql server binn directory is "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\Binn".
Enabling CLR
Adding trusted assemblies...
Adding Accessibility
Adding System.ServiceModel.Internals
Adding SMDiagnostics
Adding System.Runtime.Serialization
Adding System.Runtime.Serialization.Formatters.Soap
Adding System.IO.Compression
Adding System.IO.Compression.FileSystem
Adding System.Drawing
Adding System.Dynamic
Adding System.Windows.Forms
Adding System.Windows.Forms.DataVisualization
Adding Microsoft.CSharp
Creating Asymmetric Keys and Logins...
Asymmetric Key for Assembly 'Microsoft.RServer.ScoringLibrary' is [Microsoft_RServer_ScoringLibrary] and Login is [Microsoft_RServer_ScoringLibrary_login]
Asymmetric Key for Assembly 'Microsoft.RServer.ScoringLibrary.SqlServer' is [Microsoft_RServer_ScoringLibrary] and Login is [Microsoft_RServer_ScoringLibrary_login]
Asymmetric Key for Assembly 'Microsoft.RServer.NativeScorer' is [Microsoft_RServer_ScoringLibrary] and Login is [Microsoft_RServer_ScoringLibrary_login]
Asymmetric Key for Assembly 'Microsoft.MachineLearning.RServerScoring.Sql' is [Microsoft_MachineLearning_RServerScoring_Sql] and Login is [Microsoft_MachineLearning_RServerScoring_Sql_login]
Create Asymmetric Keys succeeded!
Copying configuration files...
C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64\net46\ScoringLibrary.SqlServer.config
C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\Binn\ScoringLibrary.config
Command RTSInstall on Instance succeeded!

C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64>
```

Administrator: Windows PowerShell

Administrator: cmd

```
C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64>RegisterRExt.exe /installRts /database:FleitasArts
Source directory to pick the RExtension binaries determined to be "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScale
R\rxLibs\x64\".
Installation flavor was not supplied
Connecting to SQL server...
Sql server binn directory is "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\Binn".
Creating Microsoft.RServer.ScoringLibrary...
Microsoft.RServer.ScoringLibrary created successfully.
Creating Microsoft.RServer.ScoringLibrary.SqlServer...
Microsoft.RServer.ScoringLibrary.SqlServer created successfully.
Creating Microsoft.RServer.NativeScorer...
Microsoft.RServer.NativeScorer created successfully.
Creating Accessibility...
Accessibility created successfully.
Creating System.ServiceModel.Internals...
System.ServiceModel.Internals created successfully.
Creating SMDiagnostics...
SMDiagnostics created successfully.
Creating System.Runtime.Serialization...
System.Runtime.Serialization created successfully.
Creating System.Runtime.Serialization.Formatters.soap...
System.Runtime.Serialization.Formatters.soap created successfully.
Creating System.IO.Compression...
System.IO.Compression created successfully.
Creating System.IO.Compression.FileSystem...
System.IO.Compression.FileSystem created successfully.
Creating system.drawing...
system.drawing created successfully.
Creating System.Dynamic...
System.Dynamic created successfully.
Creating System.Windows.Forms...
System.Windows.Forms created successfully.
Creating System.Windows.Forms.DataVisualization...
System.Windows.Forms.DataVisualization created successfully.
Creating Microsoft.CSharp...
Microsoft.CSharp created successfully.
Creating Microsoft.MachineLearning.RServerScoring.Sql...
Microsoft.MachineLearning.RServerScoring.Sql created successfully.
Creating stored procedure [dbo].[sp_rxPredict]
Creating and giving execute role for users...
Command RTSInstall succeeded!
```

C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\R_SERVICES\library\RevoScaleR\rxLibs\x64|



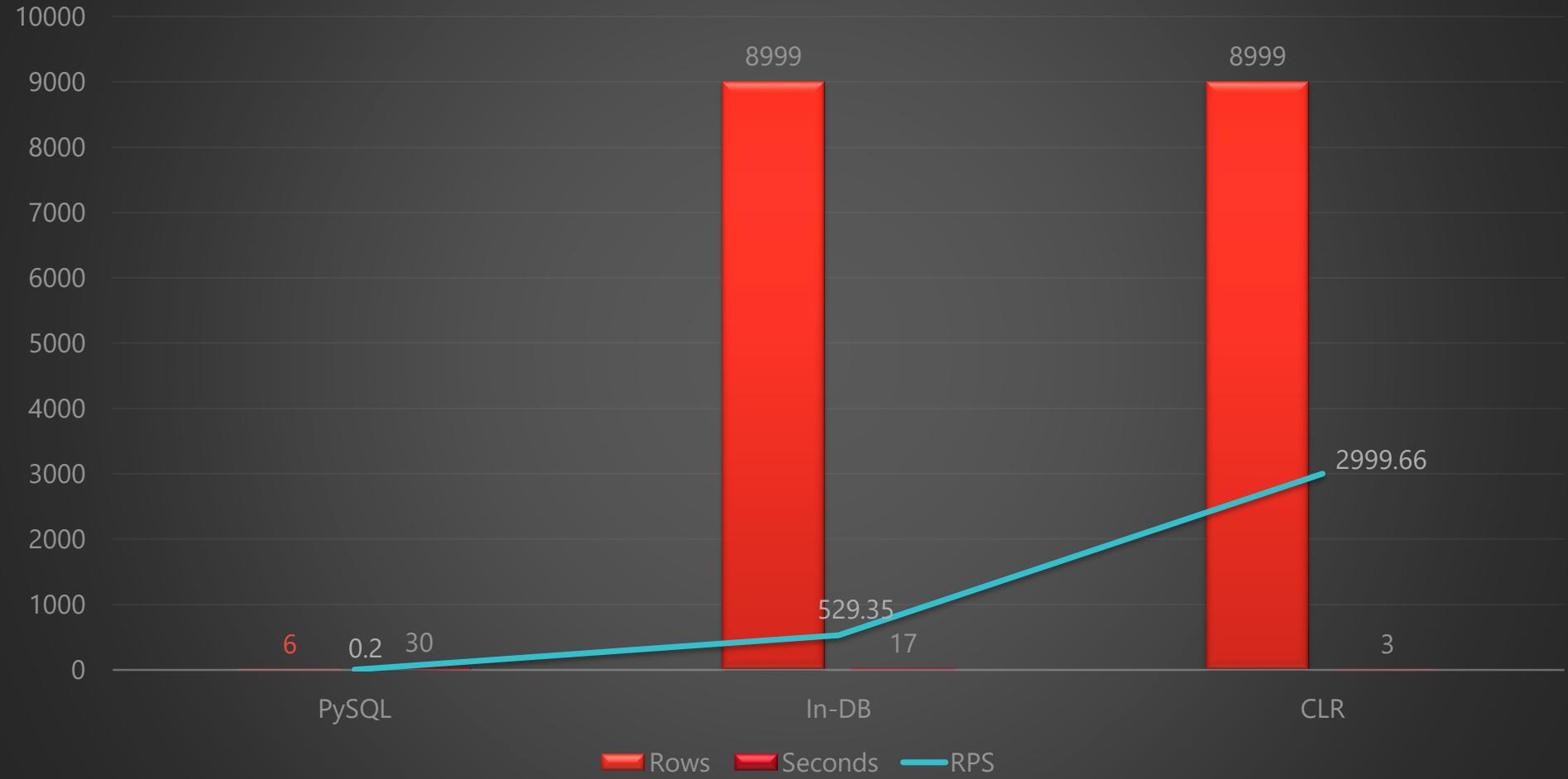
```
331 declare @model_bin varbinary(max)=null
332 select @model_bin = model from models where model_name = 'RevoMMLRealtimeScoring';
333 if @model_bin is not null begin
334 exec sp_rxPredict @model = @model_bin, @inputData = N'SELECT pr_review_content, cast(tag as varchar(1)) as tag FROM product_reviews_test_data' end
335 go --8,999 rows: sp_rxPredict 9sec vs python microsoftml rx_predict 13sec.
```

T-SQL ↻ Results ✓ Message

	PredictedLabel	Score.1	Score.2	Score.3
1	3	0.02261647	0.1164434	0.8609401
2	3	0.006391859	0.0122224	0.9813856
3	3	0.02605031	0.07372595	0.9002239
4	3	0.02085118	0.04318421	0.9359646
5	3	0.006275344	0.00719668	0.9865279
6	3	0.01219776	0.02834679	0.9594554
7	3	0.01289097	0.02562586	0.9614831
8	3	0.0262401	0.05194858	0.9218113
9	3	0.02124425	0.0905453	0.8882105
10	3	0.01573131	0.04357475	0.9406939
11	3	0.0192884	0.04815447	0.932557
12	3	0.009418746	0.0194117	0.9711696
13	3	0.01290902	0.01438312	0.9727077
14	3	0.02147118	0.05592076	0.9226081
15	3	0.02763326	0.05890254	0.9134642
16	3	0.009828706	0.02245608	0.9677153
17	3	0.03379261	0.09301157	0.8731958

✓ Query executed successfully at 1:16:10 PM | R90GTU6N (14.0 RTM) | UPCIC\hfleitas (52) | tp | 00:00:03 | 8999 rows

5x Faster (sp_rxPredict) – Laptop



DEMO

four

- (T-SQL) for json path
- (Python) json_normalize
- CognitiveAPI.ipynb
- Trollhunters.ipynb
- (PDF) TrumpUkraine.ipynb





Necessary steps

```
1 import requests
2 from pprint import pprint
3 from IPython.display import HTML
4
5 subscription_key = "6cec684a9d0f4650a5dce52fac5ba1f6"
6 text_analytics_base_url = "https://eastus2.api.cognitive.microsoft.com/text/analytics/v2.0/"
7 sentiment_url = text_analytics_base_url + "sentiment"
8
9 documents = {"documents": [
10     {
11         "id": 1,
12         "language": "en",
13         "text": "Works fine. Easy to install. Some reviews talk about not fitting wall plates. Designed for the best"
14     }
15 ]}
16
17 headers = {"Ocp-Apim-Subscription-Key": subscription_key}
18 response = requests.post(sentiment_url, headers=headers, json=documents)
19 sentiments = response.json()
20 pprint(sentiments)

{'documents': [{'id': '1', 'score': 0.9685916900634766}], 'errors': []}
```

Learn more:

<https://docs.microsoft.com/azure/cognitive-services/text-analytics/quickstarts/python>

<https://azure.microsoft.com/services/cognitive-services/text-analytics>





PASSummit2019.ipynb

CognitiveAPI.ipynb •



seattle2019 > 2SentimentPrediction > CognitiveAPI.ipynb

⊕ Code

⊕ Text

Kernel:

SQL

Attach To: .tpcxbb_1gb

Trusted

Run Cells

Clear Results

Manage Packages



JSON Path

- SQL
- PySpark3
- PySpark
- Spark | Scala
- Spark | R
- Python 3

```
1 select top 10
2      row_number() over (order by (select 1)) as id,
3      'en' as language,
4      pr_review_content as text
5 from    tpcxbb_1gb..product_reviews_training_data
6 for json path, root('documents')
```

...

(10 rows affected)

Total execution time: 00:00:00.1093385

	JSON_F52E2B61-18A1-11d1-B105-00805F49916B
1	{"documents": [{"id": 1, "language": "en", ...}}

Learn more:

<https://docs.microsoft.com/sql/relational-databases/json/format-nested-json-output-with-path-mode-sql-server>

```
55     },
56     {
57         "id": 10,
58         "language": "en",
59         "text": "Our home was built in 2003 and this fits just fine in the drawer until find one of those things that if
60     }
61 ]
62 }
63 print(type(documents))
64
65 headers = {"Ocp-Apim-Subscription-Key": subscription_key}
66 response = requests.post(sentiment_url, headers=headers, json=documents)
67 sentiments = response.json()
68 pprint(sentiments)
```

```
<class 'dict'>

{'documents': [{"id': '1', 'score': 0.9685916900634766},
    {'id': '2', 'score': 0.874512791633606},
    {'id': '3', 'score': 0.7775521278381348},
    {'id': '4', 'score': 0.1461590826511383},
    {'id': '5', 'score': 0.9813788533210754},
    {'id': '6', 'score': 0.8957217931747437},
    {'id': '7', 'score': 0.9916195869445801},
    {'id': '8', 'score': 0.08493909239768982},
    {'id': '9', 'score': 0.8297852277755737},
    {'id': '10', 'score': 0.7934412956237793}],
'errors': []}
```





```
1 drop view if exists JsonDocuments
2 go
3 create view JsonDocuments
4 as
5 select (
6     select top 10
7         row_number() over (order by (select 1)) as id,
8         'en' as language,
9         pr_review_content as text
10    from product_reviews_training_data
11    for json path, root('documents')
12 ) as documents
13 go
14 exec sp_execute_external_script @language = N'Python'
15     ,@script = N'DocOut = DocIn'
16     ,@input_data_1 = N'select * from JsonDocuments;'
17     ,@input_data_1_name = N'DocIn'
18     ,@output_data_1_name = N'DocOut'
19 with result sets ((DocOut varchar(max)));
20 go
```

Learn more:

<https://docs.microsoft.com/sql/relational-databases/json/format-nested-json-output-with-path-mode-sql-server>

<https://docs.microsoft.com/sql/relational-databases/system-stored-procedures/sp-execute-external-script-transact-sql>





PASSummit2019.ipynb

CognitiveAPI.ipynb •



seattle2019 > 2SentimentPrediction > CognitiveAPI.ipynb

⊕ Code ⊕ Text Kernel: Python 3 ▾ Attach To: localhost ▾ Trusted Run Cells Clear Results Manage Packages

```
7
8 conn = pyodbc.connect('Driver={SQL Server};Server=localhost;Database=tpcxbb_1gb;Trusted_Connection=yes;')
9 cursor = conn.cursor()
10 cursor.execute('select * from JsonDocuments')
11 row = cursor.fetchone() #pyodbc.Row
12 documents = json.loads(row[0]) #dict
13
14 headers = {"Ocp-Apim-Subscription-Key": subscription_key}
15 response = requests.post(sentiment_url, headers=headers, json=documents)
16
17 sentiments = response.json()
18 pprint(sentiments)

{'documents': [{"id": "1", "score": 0.9685916900634766},
    {"id": "2", "score": 0.874512791633606},
    {"id": "3", "score": 0.7775521278381348},
    {"id": "4", "score": 0.1461590826511383},
    {"id": "5", "score": 0.9813788533210754},
    {"id": "6", "score": 0.8957217931747437},
    {"id": "7", "score": 0.9916195869445801},
    {"id": "8", "score": 0.08493909239768982},
    {"id": "9", "score": 0.8297852277755737},
    {"id": "10", "score": 0.7934412956237793}],
'errors': []}
```



1



2



master* C X 36 ▲ 0



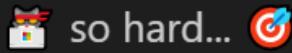
Upgrade via Notebook



upgrade with magics

<https://pypi.org/simple/urllib3/>

```
1 %%cmd
2 cd "C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES"
3 python -m pip install --upgrade C:\temp\urllib3-1.25.3-py2.py3-none-any.whl
```



```
1 declare @py nvarchar(max);
2
3 set @py = N'from pandas.io.json import json_normalize
4 rds = {"documents": [{"id": "1", "score": 0.97}, {"id": "2", "score": 0.87}, {"id": "3", "score": 0.78}], "errors": []} #dict
5 df = json_normalize(rds, "documents")
6 print(type(df), df, sep="\n") #DataFrame
7 ';
8
9 drop table if exists apiresults
10 create table apiresults (id int, score float)
11
12 insert into apiresults
13 exec sp_execute_external_script
14     @language = N'Python',
15     @script = @py,
16     @output_data_1_name =N'df'
17
18 select * from apiresults
```

Learn more:

[How to output values from sp_execute_external_script into table](#)

[Convert JSON API response to pandas Dataframe](#)



```
1 create or alter proc GetCognitiveAPI
2 as
3     declare @py nvarchar(max);
4
5     set @py = N'import requests
6 from pandas.io.json import json_normalize
7
8 subscription_key = "mykey"
9 text_analytics_base_url = "https://eastus2.api.cognitive.microsoft.com/text/analytics/v2.0/"
10 sentiment_url = text_analytics_base_url + "sentiment"
11
12 df = jsondocs
13
14 headers = {"Ocp-Apim-Subscription-Key": subscription_key, "content-type": "application/json"}
15 response = requests.post(sentiment_url, headers = headers, data = df.iloc[0][0].encode())
16
17 rds = response.json()
18 df2 = json_normalize(rds, "documents");
19
20 drop table if exists apiresults;
21 create table apiresults (id int, score float);
22
23 insert into apiresults
24 exec sp_execute_external_script @language = N'Python'
25     ,@script = @py
26     ,@input_data_1 = N'select * from JsonDocuments'
27     ,@input_data_1_name = N'jsondocs'
28     ,@output_data_1_name = N'df2'
29 select * from apiresults;
30 go
31 exec GetCognitiveAPI
32 go
```

Learn more:

<https://repl.it/languages/python3>





Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code

+ Text

Kernel: SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

Schedule Notebook

➡️ Trollhunter, Don't forget the 🔑

<https://vtropes.org/pmwiki/pmwiki.php/Quotes/Trollhunters>

```
[2] 1 drop table if exists quotes;
2
3 create table quotes (
4     quoteid      int          identity(1,1) primary key clustered,
5     character    varchar(128)  not null,
6     quote        nvarchar(max) not null,
7     sentiment    float
8 );
9
10 insert into quotes (character, quote)
11 values
12     ('The Magical Incantation that activates the Amulet', 'For the glory of Merlin, Daylight is mine to command!')
13 ,('Strickler', 'Now, I think I know what has you so distraught, Jim.')
14 ,('Jim', 'You do?')
15 ,('Strickler', 'It''s like I told you yesterday, you have a lot on your shoulders. Too much, in my opinion, for someone your age. And I think that this opportu
16 ,('Jim', 'Chess?')
17 ,('Strickler', '...I think it''s causing you anxiety. I know you want to be there for your mother, but it''s as a great poet once wrote, "'Do what''s good for
18 ,('Jim', 'Hey thanks for the advice. I like talking to you.')
19 ,('Strickler', 'Always.')
20
21 ,('Jim', 'So, the previous Trollhunter, what, retired?')
22 ,('AAARRRGHH!!!', 'Was felled.')
23 ,('Jim', 'Felled?')
24 ,('AAARRRGHH!!!', 'Means killed.')
25 ,('Blinky', 'Turned to stone and smashed. Kanjigar the Courageous was his name. Brutally slain by a ruthless troll named Bular.')
26 ,('Blinky', 'The B...')
```



master



X 0 ▲ 0





Trollhunters.ipynb ×

20190608-SQLSaturdaySF1864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

⊕ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

```
[2] 64
65 ,('Magical Incantation that appears on the Amulet', 'For the doom of Gunmar, Eclipse is mine to command!')
66
67 ,('Blinky', 'Not a rock. Dwärkstone.')
68 ,('Jim', 'Dork-stone?')
69 ,('Blinky', 'Dwärkstone.')
70 ,('Claire', 'Dork-stone?')
71 ,('Aaarrngghh!!!', 'Close enough.')
72 ,('Blinky', 'Dwärkstone is incredibly rare, highly volatile, and the only guaranteed way to expel a gruesome.')
73 ,('Toby', 'All I''m hearing is ""dork-stone."")'
74
75 ,('Otto', 'And once the vicious one realizes he has no further use for an old, blind troll... I- Well.. [Does the Throat-Slitting Gesture.]')
76 ,('Dictatious', '...well, what? You know I can''t see.')
77 ,('Otto', 'Oh. I moves my thumb across my throat. It is, as we say, a killing motion.')
78
79 ,('Claire', 'Let''s try "good cop, bad cop."')
80 ,('Blinky', 'I think I know where you''re going with this. I shall play this game.[Claire knocks on door] Open up, RotGut, or I''ll kick down this door! [RotGut')
81 ,('Gut', 'Ailment or curse?')
82 ,('Claire', 'Neither, you two-headed dirtbag! We need information. Gunmar''s got a mole down here, and you''re gonna tell us who it is!')
83 ,('Rot', 'Why is she being so mean to us, Gut?')
84 ,('Gut', 'We don''t share this sort of thing. Salesman-client confidentiality.')
85 ,('Claire', 'Oh, you wanna obstruct official Trollhunting business? Is that what you wanna do? Now you''ve done it! I feel my anger rising. I can''t control it')
86 ,('Blinky', 'Ahem. [Blinky shouts and pulls out a live dwärkstone.] If you don''t tell us what we need to know this instant, I''m turning us all into a smoking')
87 ,('Gut', 'Are you insane? You''re gonna kill us all!')
88 ,('Claire', 'Whoa whoa, Blinky, what are you doing?')
89 ,('Blinky', 'I''m buying baby. I''m buying the farm if they don''t start talking!')
90 ,('Claire', 'Wait, I thought I was the bad cop.')
91 ,('Blinky', 'I thought I was the bad cop. Your performance felt a little ho-hum.')
92 ,('Draal', 'It has been my honor ... fleshbag.')
93 ;
```

(72 rows affected)

Total execution time: 00:00:00.017





Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code

+ Text

Kernel:

SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

Schedule Notebook

```
1 drop view if exists JsonQuotes
2 go
3 create view JsonQuotes
4 as
5 select (
6     select top(select cast(count(*)*.8 as int) from Quotes)
7         quoteid as id,
8         'en' as language,
9         Quote as text
10    from Quotes
11    for json path, root('documents')
12 ) as documents
13 go
14 exec sp_execute_external_script @language = N'Python'
15     ,@script = N'DocOut = DocIn'
16     ,@input_data_1 = N'select * from JsonQuotes;'
17     ,@input_data_1_name = N'DocIn'
18     ,@output_data_1_name = N'DocOut'
19 with result sets ((DocOut varchar(max)));
20 go
```

Commands completed successfully.

Commands completed successfully.

(1 row affected)

Total execution time: 00:00:00.320

DocOut

```
1 [{"documents": [{"id": 1, "language": "en", "text": "For the glory of Merlin, Daylight is mine to command!"}, {"id": 2, "language": "en", "text": "Now, I think I ..."}]}
```



master





Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code + Text Kernel: SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

Schedule Notebook

Trollhunter, Don't forget the

<https://repl.it/languages/python3>

```
[17] 1 create or alter proc GetCognitiveAPIQuoteSentiment
2 as
3     set nocount on;
4     declare @py nvarchar(max);
5
6     set @py = N'import requests, pprint as pr
7 from pandas.io.json import json_normalize
8
9 subscription_key = "mykey"
10 text_analytics_base_url = "https://eastus2.api.cognitive.microsoft.com/text/analytics/v2.0/"
11 sentiment_url = text_analytics_base_url + "sentiment"
12
13 df = jsondocs
14
15 headers = {"Ocp-Apim-Subscription-Key": subscription_key, "content-type": "application/json"}
16 response = requests.post(sentiment_url, headers = headers, data = df.iloc[0][0].encode())
17
18 rds = response.json()
19 df2 = json_normalize(rds, "documents")
20
21 pr pprint(rds)
22 print(type(df2),df2,sep="\n")
23 ';
24
25 drop table if exists apiresults;
26 create table apiresults (id int, score float);
```



master 0 ▲ 0





Trollhunters.ipynb ×

20190608-SQLSaturdaySQL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

⊕ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

```
[17] 14
15     headers = {"Ocp-Apim-Subscription-Key": subscription_key, "content-type": "application/json"}
16     response = requests.post(sentiment_url, headers = headers, data = df.iloc[0][0].encode())
17
18     rds = response.json()
19     df2 = json_normalize(rds, "documents")
20
21     pr pprint(rds)
22     print(type(df2),df2,sep="\n")
23     ;
24
25     drop table if exists apiresults;
26     create table apiresults (id int, score float);
27
28     insert into apiresults
29     exec sp_execute_external_script @language = N'Python'
30         ,@script = @py
31         ,@input_data_1 = N'select * from JsonQuotes'
32         ,@input_data_1_name = N'jsondocs'
33         ,@output_data_1_name = N'df2'
34     select * from apiresults;
35
36     update q
37         set q.Sentiment = a.Score
38     from Quotes q
39     inner join apiresults a
40         on q.quoteid = a.id
41     where q.Sentiment is null;
42 go
43
44 exec GetCognitiveAPIQuoteSentiment;
45
46 select * from Quotes;
```

Commands completed successfully.



Trollhunters.ipynb ×

20190608-SQLSaturdaySF1864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

 Code Text Kernel: SQL

Attach To: Select Connection

 Not Trusted Run Cells Clear Results Schedule Notebook

Commands completed successfully.

(72 rows affected)

Total execution time: 00:00:01.279

	id	score
1	1	0.5
2	2	0.127195328474045
3	3	0.833639144897461
4	4	0.951666951179504
5	5	0.752673923969269
6	6	0.155267655849457
7	7	0.99489951133728
8	8	0.202319025993347
9	9	0.707312762737274
10	10	0.770378530025482
11	11	0.752673923969269
12	12	0.0333146154880524
13	13	0.126827985048294
14	14	0.751771688461304
15	15	0.959390044212341
16	16	0.734701573848724
17	17	0.833906531333923
18	18	0.862226605415344
19	19	0.026850736442566



Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

[Code](#) [Text](#) Kernel: [SQL](#) Attach To: [Select Connection](#) Not Trusted Run Cells Clear Results Schedule Notebook

2A | 2A | A 78A210A777626A4

	quoteid	character	quote	sentiment
1	1	The Magical Incantation that...	For the glory of Merlin, Day...	0.5
2	2	Strickler	Now, I think I know what has...	0.127195328474045
3	3	Jim	You do?	0.833639144897461
4	4	Strickler	It's like I told you yesterd...	0.951666951179504
5	5	Jim	Chess?	0.752673923969269
6	6	StricklerI think it's causing you ...	0.155267655849457
7	7	Jim	Hey thanks for the advice. I...	0.99489951133728
8	8	Strickler	Always.	0.202319025993347
9	9	Jim	So, the previous Trollhunter...	0.707312762737274
10	10	AAARRGGHH!!!	Was felled.	0.770378530025482
11	11	Jim	Felled?	0.752673923969269
12	12	AAARRGGHH!!!	Means killed.	0.0333146154880524
13	13	Blinky	Turned to stone and smashed...	0.126827985048294
14	14	Tobes	Don't worry dude. This Bular...	0.751771688461304
15	15	Blinky	The evidence does not sugges...	0.959390044212341
16	16	Jim	Then the other guy, he was j...	0.734701573848724
17	17	Blinky	Doubtful. Kanjigar was perha...	0.833906531333923
18	18	Jim	But not the best, I'mbettin...	0.862226605415344
19	19	Blinky	Oh, the very best. Many song...	0.926859736442566
20	20	Blinky	Destiny is a gift. Someon +	A 78A210A777626A4



Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code

+ Text

Kernel: SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

Schedule Notebook

```
1 select avg(sentiment) as AvgSentimtment, character
2 from quotes
3 where sentiment is not null
4 group by character
5 order by AvgSentimtment desc
```

(14 rows affected)

Total execution time: 00:00:00.026

	AvgSentimtment	character
1	0.865543842315674	Claire
2	0.751771688461304	Tobes
3	0.647038959539854	Jim
4	0.597320705652237	Blinky
5	0.563405146201452	Toby
6	0.529540811266218	AAARRRGHH!!
7	0.5	Barbara Lake
8	0.5	The Magical Incantation that...
9	0.359112240374088	Strickler
10	0.255251032114029	Vendel
11	0.189253866672516	Vendal
12	0.143676519393921	Magical Incantation that app...
13	0.127677083015442	Dictatious
14	0.07220658659935	Otto



master



0 ▲ 0





Trollhunters.ipynb ×

20190608-SQLSaturdaySF1864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

⊕ Code ⊕ Text Kernel: SQL ▾ Attach To: Select Connection ▾ Not Trusted Run Cells Clear Results Schedule Notebook

🚀 Schema

```
[8] 1 drop table if exists models;
2 create table models (
3     ModelID      int          identity(1,1) primary key clustered,
4     Language      varchar(30)   not null default('Python'),
5     Name          nvarchar(30)  not null,
6     Model         varbinary(max),
7     CreatedOn    datetime     default(getdate()),
8     CreatedBy    nvarchar(500)  default(suser_sname())
9 );
10 go
11
12 drop view if exists QuotesForTraining;
13 go
14 create or alter view QuotesForTraining
15 as
16     select top(select cast(count(*)*1 as int) from Quotes) --because where clause!
17         quote,
18         case
19             when sentiment <.1 then 0
20             when sentiment >=.1 and sentiment <.2 then 1
21             when sentiment >=.2 and sentiment <.3 then 2
22             when sentiment >=.3 and sentiment <.4 then 3
23             when sentiment >=.4 and sentiment <.5 then 4
24             when sentiment >=.5 and sentiment <.6 then 5
25             when sentiment >=.6 and sentiment <.7 then 6
26             when sentiment >=.7 and sentiment <.8 then 7
27             when sentiment >=.8 and sentiment <.9 then 8
28             when sentiment >=.9 and sentiment <1 then 9
29             when sentiment >=1 then 10
30             else null
```



Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

⊕ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

```
13 go
14 create or alter view QuotesForTraining
15 as
16     select top(select cast(count(*)*1 as int) from Quotes) --because where clause!
17         quote,
18         case
19             when sentiment <.1 then 0
20             when sentiment >=.1 and sentiment <.2 then 1
21             when sentiment >=.2 and sentiment <.3 then 2
22             when sentiment >=.3 and sentiment <.4 then 3
23             when sentiment >=.4 and sentiment <.5 then 4
24             when sentiment >=.5 and sentiment <.6 then 5
25             when sentiment >=.6 and sentiment <.7 then 6
26             when sentiment >=.7 and sentiment <.8 then 7
27             when sentiment >=.8 and sentiment <.9 then 8
28             when sentiment >=.9 and sentiment <1 then 9
29             when sentiment >=1 then 10
30             else null
31         end as tag
32     from quotes
33     where sentiment is not null
34     --order by tag;
35 go
36
37 drop view if exists QuotesForTesting;
38 go
39 create or alter view QuotesForTesting
40 as
41     select top(select cast(count(*)*.2 as int) from Quotes)
42         quote,
43         null as tag
44     from quotes
45     where sentiment is null;
```

46 Commands completed successfully.



Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

⊕ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

```
1 create or alter proc GetTrollhunterModel (
2     @Name nvarchar(30) = 'TrollhunterRealtime',
3     @Languange varchar(30) = 'Python'
4 )
5 as
6     declare @model varbinary(max), @train_script nvarchar(max), @Svr varchar(128) = @@servername, @Db nvarchar(128) = db_name();
7     delete top(1) from models where name = @Name and language = @Languange;
8
9     --The Python script we want to execute
10    set @train_script = N'
11 from microsoftml import rx_logistic_regression, featurize_text, n_gram
12 from revoscalepy import rx_serialize_model, RxOdbcData, rx_write_object, RxInSqlServer, rx_set_compute_context, RxLocalSeq
13
14 connection_string = "Driver=SQL Server;Server='+'@Svr+'';Database='+'@Db+'';Trusted_Connection=true;"
15 dest = RxOdbcData(connection_string, table = "models")
16
17 training_data["tag"] = training_data["tag"].astype("category")
18
19 modelpy = rx_logistic_regression(formula = "tag ~ features",
20                                     data = training_data,
21                                     method = "multiClass",
22                                     ml_transforms=[featurize_text(language="English",
23                                         cols=dict(features="quote"),
24                                         word_feature_extractor=n_gram(2, weighting="TfIdf"))],
25                                     train_threads=1)
26
27 modelbin = rx_serialize_model(modelpy, realtime_scoring_only = True)
28 rx_write_object(dest, key_name="Name", key=""+@Name+"", value_name="Model", value=modelbin, serialize=False, compress=None, overwrite=False); --overwrite=fals
29
30     exec sp_execute_external_script @language = N'Python'
31         ,@script = @train_script
32         ,@input_data_1 = N'select * from QuotesForTraining'
33         ,@input_data_1_name = N'training_data'
34     go
35
```





Trollhunters.ipynb ×

20190608-SQLSaturdaySF1864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

```
32     ,@input_data_1 = N'select * from QuotesForTraining'
33     ,@input_data_1_name = N'training_data'
34 go
35
36 exec GetTrollhunterModel; --00:00:02.919 home desktop.
37
38 select *, datalength(model) as Datalen from dbo.models;
```

Commands completed successfully.

(1 row affected)

STDERR message(s) from external script: C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\PYTHON_SERVICES\lib\site-packages\revoscalepy\RxSerializable.py:163: FutureWarning: Method .as_matrix will be removed in a future version. Use .values instead. ret[column_name] = np.array([x + 1 for x in datafram[i].cat.codes.as_matrix().tolist()]).astype(np.int32)

STDOUT message(s) from external script: Beginning processing data. Rows Read: 57, Read Time: 0, Transform Time: 0 Beginning processing data. Beginning processing data. Rows Read: 57, Read Time: 0, Transform Time: 0 Beginning processing data. Not adding a normalizer. Beginning processing data. Rows Read: 57, Read Time: 0, Transform Time: 0 Beginning processing data. Beginning optimization num vars: 7098 improvement criterion: Mean Improvement Warning: Premature convergence occurred. The OptimizationTolerance may be set too small. ro equals zero. Is your function linear? L1 regularization selected 7 of 7098 weights. Not training a calibrator because it is not needed. Elapsed time: 00:00:01.4911851 Elapsed time: 00:00:00.2200052 Rows Read: 1, Total Rows Processed: 1 Total Rows written: 1, Total time: 0.001 , Total Chunk Time: 0.045 seconds

(1 row affected)

Total execution time: 00:00:07.099

	ModelID	Language	Name	Model	CreatedOn	CreatedBy	Datalen
1	3	Python	TrollhunterRealtime	0x626C6F62BC23F8E3ABB42D41C2...	2019-08-19 00:31:56.843	UPCIC\hfleitas	19017

🏁 Finish



20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code

+ Text

Kernel:

SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

Schedule Notebook

🏁 Finish

```
[12] 1 select * from QuotesForTesting;
2 declare @model_bin varbinary(max)=null
3 select @model_bin = model from models where name = 'TrollhunterRealtime';
4
5 if @model_bin is not null
6 begin
7 exec sp_rxPredict
8     @model = @model_bin,
9     @inputData = N'select quote, cast(tag as varchar(1)) as tag from QuotesForTesting'
10
11    if object_id('tempdb.dbo.#upv') is not null drop table #upv;
12    create table #upv (
13        predictedlabel varchar(1),
14        score0 float,
15        score1 float,
16        score2 float,
17        -- score3 float,
18        -- score4 float,
19        score5 float,
20        -- score6 float,
21        score7 float,
22        score8 float,
23        score9 float
24    )
25    insert into #upv
26    exec sp_rxPredict
27        @model = @model_bin,
28        @inputData = N'select quote, cast(tag as varchar(1)) as tag from QuotesForTesting'
```





Trollhunters.ipynb ×

20190608-SQLSaturdaySF1864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

```
[12]      scorez float,
17      -- score3 float,
18      -- score4 float,
19      score5 float,
20      -- score6 float,
21      score7 float,
22      score8 float,
23      score9 float
24  )
25  insert into #upv
26  exec sp_rxPredict
27      @model = @model_bin,
28      @inputData = N'select quote, cast(tag as varchar(1)) as tag from QuotesForTesting'
29
30  select PredictedLabel, Score, Prediction
31  from  #upv
32  unpivot (
33      Prediction
34      for Score in (
35          score0,
36          score1,
37          score2,
38          -- score3,
39          -- score4,
40          score5,
41          -- score6,
42          score7,
43          score8,
44          score9
45      )
46  ) as scoreunpivot
47  group by PredictedLabel, Score, Prediction
48  order by Prediction desc
49 end;
```

(14 rows affected)



master





Trollhunters.ipynb ×

20190608-SQLSaturdaySFL864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

	PredictedLabel	Score.0	Score.1	Score.2	Score.5	Score.7	Score.8	Score.9
1	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
2	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
3	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
4	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
5	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
6	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
7	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
8	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
9	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
10	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
11	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
12	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
13	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
14	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629

	PredictedLabel	Score	Prediction
1	7	score7	0.263157486915588
2	7	score1	0.210526421666145
3	7	score8	0.157895624637604
4	7	score9	0.105262860655785
5	7	score0	0.105262331664562



Trollhunters.ipynb ×

20190608-SQLSaturdaySF1864 > SentimentPrediction > PASSInsights201908-Dev > Trollhunters.ipynb

+ Code + Text Kernel: SQL Attach To: Select Connection Not Trusted Run Cells Clear Results Schedule Notebook

1	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
2	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
3	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
4	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
5	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
6	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
7	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
8	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
9	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
10	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
11	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
12	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
13	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629
14	7	0.1052623	0.2105264	0.07017536	0.08772001	0.2631575	0.1578956	0.1052629

	PredictedLabel	Score	Prediction
1	7	score7	0.263157486915588
2	7	score1	0.210526421666145
3	7	score8	0.157895624637604
4	7	score9	0.105262860655785
5	7	score0	0.105262331664562
6	7	score5	0.087720014154911
7	7	score2	0.0701753571629524

~~SECRET//ORCON/NOFORN~~

[PkgNumberShort]

UNCLASSIFIED

PDF

Get Cognitive API

~~EYES ONLY~~
~~DO NOT COPY~~

Declassified by order of the President
September 24, 2019

MEMORANDUM OF TELEPHONE CONVERSATION

SUBJECT: (C) Telephone Conversation with President
Zelenskyy of Ukraine

PARTICIPANTS: President Zelenskyy of Ukraine

Notetakers: The White House Situation Room

DATE, TIME
AND PLACE: July 25, 2019, 9:03 - 9:33 a.m. EDT
Residence



20190608-SQLSaturdaySFL864 > Pixies > TrumpUkraine.ipynb

Code Text Kernel: SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

Mr. President, Don't forget the 🔑

<https://www.whitehouse.gov/wp-content/uploads/2019/09/Unclassified09.2019.pdf>

```
[5] 1 drop table if exists quotes;
2
3 create table quotes (
4     quoteid    int          identity(1,1) primary key clustered,
5     character   varchar(128)  not null,
6     quote      nvarchar(max) not null,
7     sentiment   float
8 );
9
10 insert into quotes (character, quote)
11 values
12 ('The President', 'Congratulations on a great victory. We all watched from the United States and you did a terrific job. The way you came from behind, -somebody who wasn''t given much of a chance, and I think you did a great job.')
13 ,('President Zelenskyy', 'You are absolutely right Mr. President. We did win big and we worked hard for this. We worked a lot but I would like to confess to you that I had an opportunity to learn from you and I think you did a great job.')
14 ,('The President', '[laughter] That's a very good idea. I think your country is very happy about that.')
15 ,('President Zelenskyy', 'Well yes, to tell you the truth, we are trying to work hard because we wanted to drain the swamp here in our country. We brought in many many new people. Not the old politician: ')
16 ,('The President', 'Well it''s very nice of you to say that. I will say that we do a lot for Ukraine. We spend a lot of effort and a lot of time. Much more than the European countries are doing and they')
17 ,('President Zelenskyy', 'Yes you are absolutely right. Not only 100 percent, but actually 1000 percent and I can tell you the following; I did talk to Angela Merkel and I did meet with her. I also met with')
18 ,('The President', 'I would like you to do us a favor though because our country has been through a lot and Ukraine knows a lot about it. I would like you to find out what happened with this whole situation.')
19 ,('President Zelenskyy', 'Yes it is very important for me and everything that you just mentioned earlier. For me as a President, it is very important and we are open for any future cooperation. We are ready to')
20 ,('The President', 'Good because I heard you had a prosecutor who was very good and he was shut down and that's really unfair. A lot of people are talking about that, the way they shut your very good prosecutor down.')
21 ,('President Zelenskyy', 'I wanted to tell you about the prosecutor. First of all I understand and I'm knowledgeable about the situation. Since we have won the absolute majority in our Parliament, the next')
22 ,('The President', 'Well, she's going to go through some things. I will have Mr. Giuliani give you a call and I am also going to have Attorney General Barr call and we will get to the bottom of it. I'm')
23 ,('President Zelenskyy', 'I would like to tell you that I also have quite a few Ukrainian friends that live in the United States. Actually last time I traveled to the United States, I stayed in New York near')
24 ,('The President', 'Good. Well, thank you very much and I appreciate that. I will tell Rudy and Attorney General Barr to call. Thank you. Whenever you would like to come to the White House, feel free to do so.')
25 ,('President Zelenskyy', 'Thank you very much. I would be very happy to come and would be happy to meet with you personally and get to know you better. I am looking forward to our meeting and I also would like to')
26 ,('The President', 'Okay, we can work that out. I look forward to seeing you in Washington and maybe in Poland because I think we are going to be there at that time.')
27 ,('President Zelenskyy', 'Thank you very much Mr. President.')
28 ,('The President', 'Congratulations on a fantastic job you've done. The whole world was watching. I'm not sure it was so much of an upset but congratulations.')
29 ,('President Zelenskyy', 'Thank you Mr. President bye-bye.')
30
31 ;
```

(18 rows affected)



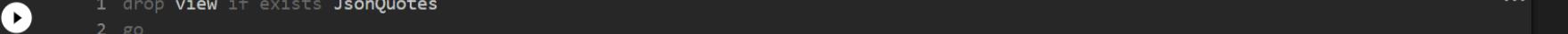


TrumpUkraine.ipynb ×

20190608-SQLSaturdaySFL864 > Pixies > TrumpUkraine.ipynb



⊕ Code ⊕ Text Kernel: SQL ▾ Attach To: Select Connection ▾ 🛡 Not Trusted ⏪ Run Cells ⌁ Clear Results



```
1 drop view if exists JsonQuotes
2 go
3 create view JsonQuotes
4 as
5 select (
6     select --top(select cast(count(*)*.8 as int) from Quotes)
7         quoteid as id,
8         'en' as language,
9         Quote as text
10    from Quotes
11    for json path, root('documents')
12 ) as documents
13 go
14 exec sp_execute_external_script @language = N'Python'
15     ,@script = N'DocOut = DocIn'
16     ,@input_data_1 = N'select * from JsonQuotes;'
17     ,@input_data_1_name = N'DocIn'
18     ,@output_data_1_name = N'DocOut'
19 with result sets ((DocOut varchar(max)));
20 go
```

...

Commands completed successfully.



master* 🔍 20 ⚠ 0

Plan Explorer on 😊 🔔

TrumpUkraine.ipynb ×

20190608-SQLSaturdaySFL864 > Pixies > TrumpUkraine.ipynb

 Code Text Kernel: SQL

Attach To: Select Connection

 Not Trusted Run Cells Clear Results Mr. President, Don't forget the <https://repl.it/languages/python3>

```
1 create or alter proc GetCognitiveAPIQuoteSentiment
2 as
3     set nocount on;
4     declare @py nvarchar(max);
5
6     set @py = N'import requests, pprint as pr
7 from pandas.io.json import json_normalize
8
9 subscription_key = "mykey" |
10 text_analytics_base_url = "https://eastus2.api.cognitive.microsoft.com/text/analytics/v2.1/"
11 sentiment_url = text_analytics_base_url + "sentiment"
12
13 df = jsondocs
14
15 headers = {"Ocp-Apim-Subscription-Key": subscription_key, "content-type": "application/json"}
16 response = requests.post(sentiment_url, headers = headers, data = df.iloc[0][0].encode())
17
18 rds = response.json()
19 df2 = json_normalize(rds, "documents")
20
21 pp pprint(rds)
```



TrumpUkraine.ipynb ×



20190608-SQLSaturdaySFL864 > Pixies > TrumpUkraine.ipynb

⊕ Code ⊕ Text Kernel: SQL ▾ Attach To: Select Connection ▾ Not Trusted Run Cells Clear Results

```
21 pr pprint(rds)
22 print(type(df2),df2,sep="\n")
23 ';
24
25 drop table if exists apiresults;
26 create table apiresults (id int, score float);
27
28 insert into apiresults
29 exec sp_execute_external_script @language = N'Python'
30     ,@script = @py
31     ,@input_data_1 = N'select * from JsonQuotes'
32     ,@input_data_1_name = N'jsondocs'
33     ,@output_data_1_name = N'df2'
34 select * from apiresults;
35
36 update q
37     set q.Sentiment = a.Score
38     from Quotes q
39     inner join apiresults a
40         on q.quoteid = a.id
41     where q.Sentiment is null;
42 go
43
44 exec GetCognitiveAPIQuoteSentiment;
45
46 select * from Quotes;
```

Commands completed successfully.



master* 20 ▲ 0

Plan Explorer on



TrumpUkraine.ipynb ×

20190608-SQLSaturdaySFL864 > Pixies > TrumpUkraine.ipynb



⊕ Code

⊕ Text

Kernel: SQL

Attach To: Select Connection

Not Trusted

Run Cells

Clear Results

❤️ Here's an idea.

```
1 select avg(sentiment) as AvgSentiment, character
2 from quotes
3 where sentiment is not null
4 group by character
5 order by AvgSentiment desc
```

(2 rows affected)

Total execution time: 00:00:00.015

	AvgSentiment	character
1	0.929387278027005	President Zelenskyy
2	0.831812553935581	The President



🚀 Schema

DEMO

five

- Paginated Report (RDL - PBIRS)
- Streaming (T-SQL)
- Streaming (Spark - BDC)



Summary

1. Add ML Features *exe components*
2. Grant Access *users, groups, add login & user*
3. Config *external scripts enabled, working directory*
4. Install Pre-Trained Open Source Models *PowerShell, RSetup.exe*

Summary

5. Code in Python & T-SQL *Visual Studio, SSMS, ADS*
6. Python Profiling *Visual Studio, Execution Plan, CPU*
7. Real-time scoring *rx_serialize_model, realtime_scoring_only = True*
8. Review Sentiment Results
sp_rxPredict wins!

Summary

7. Azure Data Studio Notebook *kernels*
8. Azure Cognitive Text Analytics API
requests, JSON
9. For JSON Path
10. GetCognitiviveAPI
11. Resources

Resources

1. fleitasarts.com
2. github.com/hfleitas/seattle19
3. netflix.com/trollhunters
4. SQL Server ML Services: [Tutorials](#)
5. Interactive deep learning: [Learn alert](#)
6. aka.ms/sqlworkshops
7. ailab.microsoft.com

Resources

8. SQL Server R Services: [Samples](#)
9. Components for Python: [Interaction](#)
10. Pre-Trained ML Models: [Install](#)
11. Threading ML: [Logistic Regression](#)
12. Azure SQL ML Preview: [Sign up](#)
13. aka.ms/ss19



Link to Poll:

bit.ly/mssql2019

Your participation is very important.

Session Evaluations

Submit by 5pm Friday,
November 15th to
win prizes.

3 WAYS TO ACCESS



Go to PASSsummit.com



Download the GuideBook App
and search: PASS Summit 2019



Follow the QR code link on session
signage



SCAN ME

Hiram Fleitas

Principal Database Architect,
Universal Property Casualty & Insurance Company



@hiramfleitas



hiram@fleitasarts.com



fleitasarts.com



/hfleitas



hiramfleitas

Thank
You

