

AiToGo API

api/model/<int:model_id>/<string:sessionname>/<string:file_name>/<string:target_column>/<string:input_column>

Model Train API

GET/api/model/{model_id}/{sessionname}/{file_name}/{target_column}/{input_column}/train

Find Importance of input columns/model accuracy or Top five most frequent text or Forecast Results (90 data points)in HTML table format

:

Example:

Classification Model

model_id = 20 (20 all models, 21: CNN,22: RandomForestClassifier,23: SGDClassifier,24: XGBClassifier,

NLP Model

model_id = 30 (30 all models, 31: LSTM, 32: LinearSVC, 33: SVM, 34: XGBoost,

Forecast Model

model_id = 10 (10 all models, 11: LSTM, 12: ARIMA

Time Series Anomaly Detection

model_id = 40 (40 all models, 41: LSTM,

Proportion Anomaly Detection

model_id = 50 (50 all models, 51: One-Class, 52: IsolationForest, 53: Gaussian

sessionname =john.doe@gmail.com (log in name)

file_name = climate_changes.csv (csv file to be analyzed)

target column= label

input_column= weight:1.00 or 0.95 or 0.90 or 0.85(Classification) text (NLP) Date (Forecast)

Return:

classification: importance of input columns (larger weight value means more importance) / Model Accuracy / Predictor

nlp: Top five most frequent text / Model Accuracy / Predictor

Python Code:

```
import requests
```

```
url='https://AiToGo.us/api/model/'
```

```
resp = requests.get(url +
```

```
'21/john.doe@gmail.com/forecast1.csv/Product/id/train?api_key=bN5qyRgCFPVVdg16D1BOsw')
```

```
if resp.status_code == requests.codes.ok:
```

```
print(resp.text)
```

```
else:
```

```
print('status code: {}'.format(resp.status_code))
```

Parameters

Try it out

Name

api_key

string

(query)

Name

model_id *

integer
(path)

sessionname *

string
(path)

file_name *

string
(path)

target_column *

string
(path)

input_column *

string
(path)

Responses

Response content type



Code

Description

200

Success

api/model/<int:model_id>/<string:sessionname>/<string:file_name>

Data API

GET /api/model/{model_id}/{sessionname}/{file_name}/analysis

Find all columns from csv file for HTML file to use

For example:

model_id = 6 (6: data_analysis/return first 5 records, 7: find_column_name_target/return column names for target, 8: find_column_name_sequence/return column names for sequence)

sessionname = john.doe@gmail.com (log in name)

file_name = climate_changes.csv (csv file to be analyzed)

Return

column names of climate_changes.csv

Python Code:

```
import requests
```

```
url='https://AiToGo.us/api/model/'
```

```
resp = requests.get(url +
```

```
'6/john.doe@gmail.com/train.csv/analysis?api_key=bN5qyRgCFPVVdg16D1BOsw')
```

```
if resp.status_code == requests.codes.ok:
```

```
print(resp.text)
```

```
else:  
    print('status code: {}'.format(resp.status_code))
```

Parameters

Try it out

Name

api_key
string
(query)

model_id *
integer
(path)

sessionname *
string
(path)

file_name *
string
(path)

Responses

Response content type



Code

Description

200

Success

api/model/<string:modelfile_id>/<string:sessionname>/<string:predictor_name>

Predictor API

GET /api/model/{modelfile_id}/{sessionname}/{predictor_name}/save

Save trained model as predictor

For example:

Input:

word: classf_CNN.predictor

sessionname =john.doe@gmail.com (log in name)

predictor_name = my_new_name.predictor

Return: null

Python Code:

```
import requests
```

```
url='https://AiToGo.us/api/model/'
```

```
resp = requests.get(url +
```

```
'classf_CNN.predictor/john.doe@gmail.com/new_name/save?api_key=bN5qyRgCFPVVdg16D1BOs  
w')
```

```
if resp.status_code == requests.codes.ok:
```

```
print(resp.text)
else:
print('status code: {}'.format(resp.status_code))
```

Parameters

Try it out

Name

api_key
string
(query)

modelfile_id *
string
(path)

sessionname *
string
(path)

predictor_name *
string
(path)

Responses

Response content type



Code

Description

200

Success

api/model/<string:modelfile_id>/<string:sessionname>

Predictor Input API

POST /api/model/{modelfile_id}/{sessionname}/predict_post

Model Predict Post:

For example:

modelfile_id = classf_CNN.predictor or nlp_LSTM.predictor or p_anomaly_OneClassSVM.predictor

sessionname =john.doe@gmail.com (log in name)

input_column_text={"columns":["Date","Location","MinTemp","MaxTemp","Rainfall","Evaporation","Sunshine","WindGustDir","WindGustSpeed","WindDir9am","WindDir3pm","WindSpeed9am","WindSpeed3pm","Humidity9am","Humidity3pm","Pressure9am","Pressure3pm","Cloud9am","Cloud3pm","Temp9am","Temp3pm","RainToday"],

"index":[0],

"data":[[{"2008-12-

01","Albury",13.4,22.9,0.6,0.0,0.0,"W",44.0,"W","WNW",20.0,24.0,71.0,22.0,1007.7,1007.1,8.0,0.0,16.9,21.8,"No"}]]}

or plain text

Return: Predictor Name Predicted Value Probability Input Text

```

Python Code:
import requests
import json
url='https://AiToGo.us/api/model/'
headers = {'Content-type': 'application/json', 'Accept': 'text/plain'}
#classification post
data_classf =
{"columns":["ACCT_ID","ORDR_ID","UsageDate","RECENCY","ORDER_AMOUNT","PROD_ID","P
RODUCT_NAME","Margin"],"index":[0],"data":[[650199,828359,"2019-06-
19",481,19.99,305,"Unlimited Extension",25.312704]]}
json_classf = json.dumps(data_classf, separators=(',', ':'))
print('c_forecast_100.predictor post')
res
= requests.post(url+'c_forecast_100.predictor/jchiang1999@gmail.com/predict_post?api_key=b8nna
OrMPdovee0L9c_-sw', json=json_classf, headers=headers)
if res.ok:
print(res.json())
print(res)
#nlp post
data_nlp = "Fill in text to NLP predict"
json_classf = json.dumps(data_nlp, separators=(',', ':'))
print('a_nlp_LinearSVC.predictor post')
res
= requests.post(url+'nlp_LinearSVC.predictor/jchiang1999@gmail.com/predict_post?api_key=b8nna
OrMPdovee0L9c_-sw', json=json_classf, headers=headers)
if res.ok:
print(res.json())
print(res)
Parameters
Try it out

```

Name	Description
api_key string <i>(query)</i>	
input_column_text object <i>(body)</i>	Example Value <pre>{}</pre> Parameter content type 
modelfile_id * string <i>(path)</i>	
sessionname * string <i>(path)</i>	

Responses

Response content type



Code

Description

200

Success

GET/api/model/{modelfile_id}/{sessionname}/predictor_input

Model Predict:

For example:

Input:

modelfile_id = classf_CNN.predictor or nlp_LSTM.predictor or p_anomaly_OneClassSVM.predictor

sessionname = john.doe@gmail.com (log in name)

Return:

input format for predictor with html



Parameters

Try it out

Name

api_key

string

(query)

modelfile_id *

string

(path)

sessionname *

string

(path)

Responses

Response content type



Code

Description

200

Success

api/model/<string:modelfile_id>/<string:sessionname>/<string:input_column_text>

Model NLP Predict API

GET/api/model/{modelfile_id}/{sessionname}/{input_column_text}/classf_predict

Model Classification Predict:

For example:

Classification Model

modelfile_id = classf_CNN.predictor

sessionname =john.doe@gmail.com (log in name)

input_column_text={"columns":["Date","Location","MinTemp","MaxTemp","Rainfall","Evaporation","Sunshine","WindGustDir","WindGustSpeed","WindDir9am","WindDir3pm","WindSpeed9am","WindSpeed3pm","Humidity9am","Humidity3pm","Pressure9am","Pressure3pm","Cloud9am","Cloud3pm","Temp9am","Temp3pm","RainToday"],

"index":[0],

"data":[[{"2008-12-

01","Albury",13.4,22.9,0.6,0.0,0.0,"W",44.0,"W","WNW",20.0,24.0,71.0,22.0,1007.7,1007.1,8.0,0.0,16.9,21.8,"No"}]]}

Return:Predictor Name Predicted Value Probability Input Text

Python Code:

import requests

url='https://AiToGo.us/api/model/'

resp = requests.get(url +

'nlplinear.predictor/john.doe@gmail.com/{"columns":["ACCT_ID","ORDR_ID","UsageDate","RECENCY","ORDER_AMOUNT","PROD_ID","PRODUCT_NAME","Margin"],"index":[0],"data":[[650199,828359,"2019-06-19",481,19.99,305,"Unlimited

Extension",25.312704]]}/predict?api_key=bN5qyRgCFPVVdg16D1BOsw')

if resp.status_code == requests.codes.ok:

print(resp.text)

else:

print('status code: {}'.format(resp.status_code))

Parameters

Try it out

Name

api_key

string

(query)

modelfile_id *

string

(path)

sessionname *

string

(path)

input_column_text *

string

(path)

Responses

Response content type



Code

Description

200

Success

GET/api/model/{modelfile_id}/{sessionname}/{input_column_text}/nlp_predict

Model NLP Predict:

For example:

NLP Model

modelfile_id = nlp_LSTM.predictor

sessionname = john.doe@gmail.com (log in name)

input_column_text= 'NLPtext'

Return:Predictor Name Predicted Value Probability Input Text

Python Code:

```
import requests
```

```
url='https://AiToGo.us/api/model/'
```

```
resp = requests.get(url +
```

```
'classf_cnn.predictor/john.doe@gmail.com/NLPtext/nlp_predict?api_key=bN5qyRgCFPVVdg16D1BOsw')
```

```
if resp.status_code == requests.codes.ok:
```

```
print(resp.text)
```

```
else:
```

```
print('status code: {}'.format(resp.status_code))
```

Parameters

Try it out

Name

api_key

string

(query)

modelfile_id *

string

(path)

sessionname *

string

(path)

input_column_text *

string

(path)

Responses

Response content type



Code

Description

Code

Description

200

Success

GET/api/model/{modelfile_id}/{sessionname}/{input_column_text}/predict

Model Predict:

For example:

modelfile_id = classf_CNN.predictor or nlp_LSTM.predictor

sessionname = john.doe@gmail.com (log in name)

input_column_text={"columns":["Date","Location","MinTemp","MaxTemp","Rainfall","Evaporation","Sunshine","WindGustDir","WindGustSpeed","WindDir9am","WindDir3pm","WindSpeed9am","WindSpeed3pm","Humidity9am","Humidity3pm","Pressure9am","Pressure3pm","Cloud9am","Cloud3pm","Temp9am","Temp3pm","RainToday"],

"index":[0],

"data":[[{"2008-12-

01","Albury",13.4,22.9,0.6,0.0,0.0,"W",44.0,"W","WNW",20.0,24.0,71.0,22.0,1007.7,1007.1,8.0,0.0,16.9,21.8,"No"}]]}

or plain text

Return: Predictor Name Predicted Value Probability Input Text

Python Code:

import requests

url='https://AiToGo.us/api/model/'

resp = requests.get(url +

'nlplinear.predictor/john.doe@gmail.com/Fill%20in%20text%20to%20NLP/predict?api_key=bN5qyRgCFPVVdg16D1BOsw')

if resp.status_code == requests.codes.ok:

print(resp.text)

else:

print('status code: {}'.format(resp.status_code))

Parameters

Try it out

Name

api_key

string

(query)

modelfile_id *

string

(path)

sessionname *

string

(path)

input_column_text *

string

(path)

Responses

Response content type



Code

Description

200

Success

api/operation/<string:sessionname>/<string:file_name>

Operation File Delete

GET /api/operation/{sessionname}/{file_name}/file

Parameters

Try it out

Name

api_key
string
(query)

sessionname *
string
(path)

file_name *
string
(path)

Responses

Response content type



Code

Description

200

Success

api/operation/<string:modelfile_id>/<string:sessionname>

Operation Deployment

POST /api/operation/{modelfile_id}/{sessionname}/deploy

Parameters

Try it out

Name

Description

api_key

string

(query)

input_column_text

object

(body)

Example Value

{}

Parameter content type



modelfile_id *

string

(path)

sessionname *

string

(path)

Responses

Response content type



Code

Description

200

Success