

## API Reference

# Audio prediction API

API Version: 0.1.0

API used for testing and predicting audio data on trained models.

### Quick instructions:

1. Make sure you have a .ckpt model inside of the --model-dir directory.
2. Click on GET /models
3. Send a request to get all available models.
4. Copy the model path you want to use for prediction.
5. Click on POST /model/predict-files
6. Paste the model path into the model\_ckpt\_path field.
7. Choose the audio files you want to predict.
8. Predict labels for the audio files.

# INDEX

<b>1. MODEL INFERENCE</b>	<b>3</b>
1.1 POST /model/test-directory-stream	3
1.2 POST /model/test-directory	5
1.3 POST /model/predict-directory-stream	7
1.4 POST /model/predict-directory	8
1.5 POST /model/predict-files	8
<b>2. RESOURCES</b>	<b>10</b>
2.1 GET /models	10
2.2 GET /dataset-types	10

# API

## 1. MODEL INFERENCE

Endpoints used for predicting audio data on trained models. Curl example for multiple audios:

```
curl -X 'POST' \http://localhost:8090/model/predict-files?model_ckpt_path=models%2Fmy_model.ckpt' \-H 'accept: application/json' \-H 'Content-Type: multipart/form-data' \-F 'audio_files=@track_23.wav;type=audio/wav'
```

### 1.1 POST /model/test-directory-stream

#### Test Directory Stream

Streams classes of audios from a directory and returns metrics for multiclass classification. Streaming is useful if your directory is large and you don't want to wait for the whole directory to be processed. If you want to process the whole directory at once, use the endpoint test-directory

#### REQUEST

##### QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
model_ckpt_path	path	

REQUEST BODY - application/json

#### RESPONSE

STATUS CODE - 200: Successful Response

##### RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
accuracy*	number	
hamming_distance*	number	
f1*	number	
precision*	number	
recall*	number	
instruments/cello_accuracy*	number	
instruments/clarinet_accuracy*	number	
instruments/flute_accuracy*	number	
instruments/acoustic_guitar_accuracy*	number	
instruments/electric_guitar_accuracy*	number	
instruments/organ_accuracy*	number	
instruments/piano_accuracy*	number	
instruments/saxophone_accuracy*	number	
instruments/trumpet_accuracy*	number	
instruments/violin_accuracy*	number	

NAME	TYPE	DESCRIPTION
instruments/human_voice_accuracy*	number	
instruments/cello_hamming_distance*	number	
instruments/clarinet_hamming_distance*	number	
instruments/flute_hamming_distance*	number	
instruments/acoustic_guitar_hamming_distance*	number	
instruments/electric_guitar_hamming_distance*	number	
instruments/organ_hamming_distance*	number	
instruments/piano_hamming_distance*	number	
instruments/saxophone_hamming_distance*	number	
instruments/trumpet_hamming_distance*	number	
instruments/violin_hamming_distance*	number	
instruments/human_voice_hamming_distance*	number	
instruments/cello_f1*	number	
instruments/clarinet_f1*	number	
instruments/flute_f1*	number	
instruments/acoustic_guitar_f1*	number	
instruments/electric_guitar_f1*	number	
instruments/organ_f1*	number	
instruments/piano_f1*	number	
instruments/saxophone_f1*	number	
instruments/trumpet_f1*	number	
instruments/violin_f1*	number	
instruments/human_voice_f1*	number	
instruments/cello_precision*	number	
instruments/clarinet_precision*	number	
instruments/flute_precision*	number	
instruments/acoustic_guitar_precision*	number	
instruments/electric_guitar_precision*	number	
instruments/organ_precision*	number	
instruments/piano_precision*	number	
instruments/saxophone_precision*	number	
instruments/trumpet_precision*	number	
instruments/violin_precision*	number	
instruments/human_voice_precision*	number	
instruments/cello_recall*	number	
instruments/clarinet_recall*	number	
instruments/flute_recall*	number	
instruments/acoustic_guitar_recall*	number	
instruments/electric_guitar_recall*	number	
instruments/organ_recall*	number	
instruments/piano_recall*	number	
instruments/saxophone_recall*	number	

NAME	TYPE	DESCRIPTION
instruments/trumpet_recall*	number	
instruments/violin_recall*	number	
instruments/human_voice_recall*	number	
predictions*	object	

STATUS CODE - 422: Validation Error

RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
detail	array	
loc*	array	
ANY:OF	object	
prop0	string	
prop1	integer	
msg*	string	
type*	string	

## 1.2 POST /model/test-directory

### Test Directory

Infers classes of audios from a directory and returns metrics for multiclass classification.

### REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
model_ckpt_path	path	

REQUEST BODY - application/json

### RESPONSE

STATUS CODE - 200: Successful Response

RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
accuracy*	number	
hamming_distance*	number	
f1*	number	
precision*	number	
recall*	number	
instruments/cello_accuracy*	number	
instruments/clarinet_accuracy*	number	
instruments/flute_accuracy*	number	

NAME	TYPE	DESCRIPTION
instruments/acoustic_guitar_accuracy*	number	
instruments/electric_guitar_accuracy*	number	
instruments/organ_accuracy*	number	
instruments/piano_accuracy*	number	
instruments/saxophone_accuracy*	number	
instruments/trumpet_accuracy*	number	
instruments/violin_accuracy*	number	
instruments/human_voice_accuracy*	number	
instruments/cello_hamming_distance*	number	
instruments/clarinet_hamming_distance*	number	
instruments/flute_hamming_distance*	number	
instruments/acoustic_guitar_hamming_distance*	number	
instruments/electric_guitar_hamming_distance*	number	
instruments/organ_hamming_distance*	number	
instruments/piano_hamming_distance*	number	
instruments/saxophone_hamming_distance*	number	
instruments/trumpet_hamming_distance*	number	
instruments/violin_hamming_distance*	number	
instruments/human_voice_hamming_distance*	number	
instruments/cello_f1*	number	
instruments/clarinet_f1*	number	
instruments/flute_f1*	number	
instruments/acoustic_guitar_f1*	number	
instruments/electric_guitar_f1*	number	
instruments/organ_f1*	number	
instruments/piano_f1*	number	
instruments/saxophone_f1*	number	
instruments/trumpet_f1*	number	
instruments/violin_f1*	number	
instruments/human_voice_f1*	number	
instruments/cello_precision*	number	
instruments/clarinet_precision*	number	
instruments/flute_precision*	number	
instruments/acoustic_guitar_precision*	number	
instruments/electric_guitar_precision*	number	
instruments/organ_precision*	number	
instruments/piano_precision*	number	
instruments/saxophone_precision*	number	
instruments/trumpet_precision*	number	
instruments/violin_precision*	number	
instruments/human_voice_precision*	number	
instruments/cello_recall*	number	

NAME	TYPE	DESCRIPTION
instruments/clarinet_recall*	number	
instruments/flute_recall*	number	
instruments/acoustic_guitar_recall*	number	
instruments/electric_guitar_recall*	number	
instruments/organ_recall*	number	
instruments/piano_recall*	number	
instruments/saxophone_recall*	number	
instruments/trumpet_recall*	number	
instruments/violin_recall*	number	
instruments/human_voice_recall*	number	
predictions*	object	

STATUS CODE - 422: Validation Error

RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
detail	array	
loc*	array	
ANY:OF	object	
prop0	string	
prop1	integer	
msg*	string	
type*	string	

## 1.3 POST /model/predict-directory-stream

### Predict Directory Stream

Streams classes of audios from a directory. Streaming is useful if your directory is large and you don't want to wait for the whole directory to be processed. If you want to process the whole directory at once, use the endpoint predict-directory. **You don't have to specify the dataset structure type is always inference**

### REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
model_ckpt_path	path	

REQUEST BODY - application/json

### RESPONSE

STATUS CODE - 200: Successful Response

RESPONSE MODEL - application/json

STATUS CODE - 422: Validation Error

RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
detail	array	
loc*	array	
ANY:OF	object	
prop0	string	
prop1	integer	
msg*	string	
type*	string	

1.4 POST /model/predict-directory

Predict Directory

Infers classes of audios from a directory.You don't have to specify the dataset structure type is always inference

REQUEST

QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
model_ckpt_path	path	

REQUEST BODY - application/json

RESPONSE

STATUS CODE - 200: Successful Response

RESPONSE MODEL - application/json

STATUS CODE - 422: Validation Error

RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
detail	array	
loc*	array	
ANY:OF	object	
prop0	string	
prop1	integer	
msg*	string	
type*	string	

1.5 POST /model/predict-files



## Predict Files

Infers classes of audios from a list of files. You don't have to specify the dataset structure type is always inference

## REQUEST

### QUERY PARAMETERS

NAME	TYPE	DESCRIPTION
model_ckpt_path	path	

### FORM DATA PARAMETERS

NAME	TYPE	DESCRIPTION
audio_files	array of string	

## RESPONSE

### STATUS CODE - 200: Successful Response

#### RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
ARRAY OF OBJECT WITH BELOW STRUCTURE		
cel*	number	
cla*	number	
flu*	number	
gac*	number	
gel*	number	
org*	number	
pia*	number	
sax*	number	
tru*	number	
vio*	number	
voi*	number	

### STATUS CODE - 422: Validation Error

#### RESPONSE MODEL - application/json

NAME	TYPE	DESCRIPTION
OBJECT WITH BELOW STRUCTURE		
detail	array	
loc*	array	
ANY:OF	object	
prop0	string	
prop1	integer	
msg*	string	
type*	string	

## 2. RESOURCES

Models and datasets available for inference.

### 2.1 GET /models

#### Get Models

Returns paths to all available models on the server. Use these model paths for all POST request predictions. Models are fetched from the directory --model-dir. Only models with the extension \*.ckpt are fetched.

#### REQUEST

No request parameters

#### RESPONSE

STATUS CODE - 200: Successful Response

RESPONSE MODEL - application/json

### 2.2 GET /dataset-types

#### Get Supported Datasets

Returns dataset types, each dataset type has a strategy for loading labels from a given directory. If you don't have labels or don't care about them, you can use the empty string "" which only loads the audio. If you want to load labels use the following dataset types: ['irmastrain', 'irmastest', 'csv', 'inference', 'openmic']

#### REQUEST

No request parameters

#### RESPONSE

STATUS CODE - 200: Successful Response

RESPONSE MODEL - application/json

---