Voice Features API

Release

Bogdan Ivashchenko

Dec 28, 2020

Chapter 1

API Documentation

1.1 Summary

Resource	Operation	Description
Compare features	POST /compare	Return comparison distance between two sets of features.
Get features	GET /features	Get features for record.

1.2 API Details

GET /features

Get features for voice record.

Example request:

```
GET /features?url=https://example.com/record.wav HTTP/1.1
Host: example.com
Accept: application/json
```

Example response:

```
HTTP/1.1 200 OK
Vary: Accept
Content-Type: application/json

{
    'features': {
        'd_vector': [0.01,0.01,0.0,0.0,0.01,0.14],
        'mfcc': [-0.22,-0.12,-0.39,-0.51,-0.37],
        'pncc': [0.04,0.04,0.15,0.18,0.14],
        'lfcc': [-0.85,-0.76,-0.69,-0.55,-0.35]
    }
}
```

Query Parameters

- **`url`** – url path to the wav file
- **`dtln`** – True/False; use speech improvement algorithm or not

Request JSON Object

- **features** (*dict*) – feature set for second
- **d_vector** (*array*) – d-vector for voice record
- **mfcc** (*array*) – mfcc feature vector for voice record
- **pfcc** (*array*) – pfcc feature vector for voice record
- **lfcc** (*array*) – lfcc feature vector for voice record

Response Headers

- Content-Type – application/json

Status Codes

- 200 OK – features extracted

POST /compare

Compare two sets of features. Return comparison distance between two sets of records.

Example request:

```
POST /compare HTTP/1.1
Host: example.com
Accept: application/json

{
    'features1': {
        'd_vector': [0.01,0.01,0.0,0.0,0.01,0.14],
        'mfcc': [-0.22,-0.12,-0.39,-0.51,-0.37],
        'pncc': [0.04,0.04,0.15,0.18,0.14],
        'lfcc': [-0.85,-0.76,-0.69,-0.55,-0.35]
    },
    'features2': {
        'd_vector': [0.02,0.03,0.1,0.1,0.015,0.14],
        'mfcc': [-0.42,-0.22,-0.4,-0.67,-0.3],
        'pncc': [0.05,0.01,0.25,0.1,0.35],
        'lfcc': [-0.79,-0.9,-0.7,-0.67,-0.45]
    }
}
```

Example response:

```
HTTP/1.1 200 OK
Vary: Accept
Content-Type: application/json

{
    'similarity': 0.7
}
```

Request JSON Object

- **features1** (*dict*) – feature set for first record
- **features2** (*dict*) – feature set for second record
- **d_vector** (*array*) – d-vector for voice record
- **mfcc** (*array*) – mfcc feature vector for voice record
- **pfcc** (*array*) – pfcc feature vector for voice record

- **`lfcc`** (*`array`*) – lfcc feature vector for voice record

Response JSON Object

- **`similarity`** (*`float`*) – comparison distance between two feature sets

Response Headers

- Content-Type – application/json

Status Codes

- 200 OK – features compared

Chapter 2

Featurizer Documentation

2.1 Summary

Featurizer provides ability to read audio records as file or by url, get features set from audio record, compare different feature sets and plot spectrogram and features plots.

2.2 Featurizer Details

class `featurizer.`**`AudioFeaturizer`**

AudioFeaturizer allows to get features from audio records.

This class allows to read audio record as file or by URL, get features from audio record and compare different sets of features to get similarity.

Parameters

- **`names_to_norm`** (*`list`*) – Names of features that should be normilized.
- **`path_to_temp_file`** (*`str`*) – File name of temporary saved audio record.
- **`feature_len`** (*`int`*) – Default lenght of feature vector.
- **`d_weight`** (*`float`*) – Weight for d-vector feature, which is used in process of comparison.
- **`lfcc_weight`** (*`float`*) – Weight for lfcc feature, which is used in process of comparison.
- **`pncc_weight`** (*`float`*) – Weight for pncc feature, which is used in process of comparison.
- **`mfcc_weight`** (*`float`*) – Weight for mfcc feature, which is used in process of comparison.

`compare_two_features_sets` (*features_1*, *features_2*)

Compare two feature sets.

Parameters

- **`features_1`** (*`dict`*) – First feature set.
- **`features_2`** (*`dict`*) – Second feature set.

Returns Float number which shows similarity between two feature sets.

cosine_similarity (*x, y*)

Compare two feature lists by cosine distance.

Parameters
- **x** (*list*) – First feature vector.
- **y** (*list*) – Second feature vector.

Returns Float number from 0 to 1 which shows similarity between two vectors.

encoder

VoiceEncoder: VoiceEncoder that used to get d-vector from audio record.

features_to_json_serializable (*all_features*)

Format feature set to json-serializable.

Parameters **all_features** (*dict*) – Feature set.

Returns Json-serializable feature set dictionary.

get_all_features (*record, sample_rate=16000, normalize_dim=False*)

Get all list of features from audio record.

Parameters
- **record** (*object*) – Record object to get feature from.
- **sample_rate** (*int*) – Sample rate for audio record.
- **normalize_dim** (*bool*) – Normilize vectors or not.

Returns Dictionary of all features

get_all_features_limited (*record, sample_rate=16000*)

Get all list of features from audio record and normalize by default dimension and lenght.

Parameters
- **record** (*object*) – Record object to get feature from.
- **sample_rate** (*int*) – Sample rate for audio record.

Returns Dictionary of all features

get_d_vector (*record, sample_rate*)

Get d-vector feature from audio record.

Args: :param record: Record object to get feature from. :type record: object :param sample_rate: Sample rate for audio record. :type sample_rate: int

Returns D-vector feature vector

get_lfcc (*record, sample_rate*)

Get LFCC feature from audio record.

LFCC paper

Parameters
- **record** (*object*) – Record object to get feature from.
- **sample_rate** (*int*) – Sample rate for audio record.

Returns LFCC feature vector

get_mfcc (*record, sample_rate*)

Get MFCC feature from audio record.

MFCC paper

Parameters
- **record** (*object*) – Record object to get feature from.
- **sample_rate** (*int*) – Sample rate for audio record.

Returns MFCC feature vector

get_pncc (*record, sample_rate*)

Get PNCC feature from audio record.

PNCC paper

Parameters
- **record** (*object*) – Record object to get feature from.
- **sample_rate** (*int*) – Sample rate for audio record.

Returns PNCC feature vector

is_in_norm_list (*feature_name*)

Check feature name in default normaliation list.

Parameters **feature_name** (*str*) – Name of the feature.

Returns True or False for feature name in default normalize list.

norm_dim (*features*)

Normalize feature vector to 1-dimensional with default lenght.

Parameters **features** (*list*) – Feature vector.

Returns Normalized features

read_file (*file_name, sample_rate=16000*)

Read audio record as file.

Parameters
- **file_name** (*str*) – Path to the audio file.
- **sample_rate** (*int*) – Sample rate for audio record.

Returns Record and sample rate

read_file_by_url (*url, sample_rate=16000*)

Read audio record by URL.

Parameters
- **url** (*str*) – URL to the audio file.
- **sample_rate** (*int*) – Sample rate for audio record.

Returns Record and sample rate

visualize_features (*features, feature_index='feature', frame_index='frame index', normalize_dim=False*)

Build plot of features for audio record.

Parameters
- **features** (*list*) – Feature vector.
- **feature_index** (*str*) – Feature name index.
- **frame_index** (*str*) – Frame name index.
- **normalize_dim** (*bool*) – Normilize vectors or not.

visualize_spectrogram (*record, sample_rate*)

Build plot of spectrogram for audio record.

Parameters

- **record** (*object*) – Record object.
- **sample_rate** (*int*) – Sample rate for audio record.

Chapter 3

DTLN Documentation

3.1 Summary

DTLN model helps to clean audio record from noise and improves quality of speech in record.

DTLN paper

3.2 DTLN Details

class `dtln.`**`DTLNproc`** (*model_path='./dtln/pretrained_model/'*)

DTLNproc allows to clean noise and improve speech in audio record.

`load_model` (*model_path*)

Load model into the memory for further usage.

Parameters **`model_path`** (*`str`*) – path to the weights and config of DTLN model.

Returns Model object.

`model`

model: model loaded to the memory

`model_path`

model_path: path to the weights and config of DTLN model.

`process_audio` (*audio_path*, *output_path*)

Load record from file and clean record from noise and improve speech quality.

Parameters

- **`audio_path`** (*`str`*) – Path to the audio file.
- **`output_path`** (*`str`*) – Path ot the processed and saved audio file.

`process_record` (*record*, *fs*)

Clean record from noise and improve speech quality.

Parameters

- **`record`** (*`object`*) – Record object for improvement.
- **`fs`** (*`int`*) – Sample rate of record.

Returns Clean and improved record object.

- Documentation in PDF format.

- *Index*
- *Module Index*
- *Search Page*

/compare

/features

Index