
trunofficial Documentation

Release 0.2.4

Ritiek Malhotra

Sep 14, 2017

Contents

1	Trunofficial Objects	3
2	Trunofficial Sub-Objects	5
3	Creating Trunofficial Objects	7
3.1	Trunofficial Attributes	8
4	Phone Objects	11
4.1	Phone Attributes	11
5	Address Objects	13
5.1	Address Attributes	13
	Python Module Index	15

This is the documentation for trunofficial - unofficial API to the Truecaller phone number search

A quick start intro with usage examples is available in the [README](#)

Development / Source code / Bug reporting: github.com/ritiek/trunofficial/

CHAPTER 1

Trunofficial Objects

Trunofficial objects relate to information fetched from the Truecaller database. They hold metadata such as *owner name, id, provider* and *trace*

CHAPTER 2

Trunofficial Sub-Objects

Phone objects relate to information to the phone number. They hold phone-specific data such as *phone number*, *number type*, *country code* and *dial code*.

Creating Trunofficial Objects

Create a Trunofficial object or a list of Trunofficial objects using the `trunofficial.search()` function, giving a phone number as the argument.

```
trunofficial.search(numbers, cc=None)
```

Creates a new Trunofficial object.

Parameters

- **number** (*str*) – Phone number of the user
- **cc** (*str*) – Prioritize search in a specific country (default: determined by IP address)

Example:

```
import trunofficial

# To prioritize search in your country (as determined by your IP)
owner = trunofficial.search("2024561111")

# To prioritize search in a specific country with the country code
owner = trunofficial.search("2024561111", cc="US")
```

You can also search for multiple phone numbers at once and create a list of Trunofficial objects.

Example:

```
import trunofficial

# Look for matching phone numbers in US
owners = trunofficial.search("2024561111", "2067093100", cc="US")
first_owner = owners[0]
second_owner = owners[1]
```

Trunofficial Attributes

Once you have created a Trunofficial object using `trunofficial.search()`, several data attributes are available

Trunofficial.id

The id of the owner (*str*)

Trunofficial.name

The name of the owner (*str*)

Trunofficial.score

The score of the owner (*str*)

Trunofficial.access

The public accessibility of the database (*str*)

Trunofficial.enhanced

The availability of enhanced information (*str*)

Trunofficial.internet_address

The publicly accessible information like e-mail (*int*)

Trunofficial.badges

The badges earned by the owner (*str*)

Trunofficial.tags

The tags earned by the owner (*str*)

Trunofficial.sources

The sources available of the owner (*str*)

Trunofficial.provider

The phone number provider (*str*)

Trunofficial.trace

The available trace of the owner (*str*)

Trunofficial.sourcestats

The available source stats of the owner (*str*)

An example of accessing this owner metadata is shown below:

```
import trunofficial
owner = trunofficial.search("2024561111")
print(owner.id)
print(owner.name)
print(owner.score)
print(owner.access)
print(owner.enhanced)
print(owner.internet_address)
print(owner.badges)
print(owner.tags)
print(owner.sources)
```

Which will result in this output:

```
uLByRJydv5fh+1nHPzemqg==
Obama
0.8
PUBLIC
```

```
True  
[]  
[]  
[u'4', u'51']  
[]
```


class `trunofficial.Phone`

After you have created a *Trunofficial* object using `search()`, you can then access the phone information by using

`Trunofficial.phone`

Phone Attributes

A Phone object can be used to access the following attributes

`Phone.number`

The phone number of the owner formatted in e164 format

`Phone.numbertype`

The type of number of the owner

`Phone.national`

The phone number of the owner formatted in national format

`Phone.dialcode`

The dial code prefix of the phone number

`Phone.countrycode`

The country code s depicted by the phone number

`Phone.carrier`

The carrier of the phone number

`Phone.spamscore`

The spam score of the owner. Higher the score, the greater the spammer.

`Phone.spamtype`

The label of the spam type

Phone.**phonetype**

The label of the phone type

An example of accessing Phone attributes:

```
>>> import trunofficial
>>> owner = trunofficial.search("2024561111", cc="US")
>>> mobile = owner.phone
>>> mobile.number
u'+912024561111'
>>> mobile.numbertype
u'FIXED_LINE'
>>> mobile.carrier
u'BSNL'
```

class `trunofficial.Address`

After you have created a *Trunofficial* object using `search()`, you can then access the address information by using

`Trunofficial.addresss`

Address Attributes

An Address object can be used to access the following attributes

`Address.area`

The area as the phone number depicts

`Address.city`

The city as the phone number depicts

`Address.countrycode`

The country code as depicted by the location

`Address.timezone`

The time zone as depicted by the location

`Address.type`

The label of the address type

An example of accessing Address attributes:

```
>>> import trunofficial
>>> owner = trunofficial.search("2024561111")
>>> house = owner.address
>>> house.area
u'Pune, Maharashtra'
>>> house.city
u'Pune, Maharashtra'
```

```
>>> house.timezone  
u'GMT+05:30'
```

t

Trunofficial, 3

A

access (Trunofficial.Trunofficial attribute), 8
addresss (Trunofficial.Trunofficial attribute), 13
area (Trunofficial.Address attribute), 13

B

badges (Trunofficial.Trunofficial attribute), 8

C

carrier (Trunofficial.Phone attribute), 11
city (Trunofficial.Address attribute), 13
countrycode (Trunofficial.Address attribute), 13
countrycode (Trunofficial.Phone attribute), 11

D

dialcode (Trunofficial.Phone attribute), 11

E

enhanced (Trunofficial.Trunofficial attribute), 8

I

id (Trunofficial.Trunofficial attribute), 8
internet_address (Trunofficial.Trunofficial attribute), 8

N

name (Trunofficial.Trunofficial attribute), 8
national (Trunofficial.Phone attribute), 11
number (Trunofficial.Phone attribute), 11
numbertype (Trunofficial.Phone attribute), 11

P

phone (Trunofficial.Trunofficial attribute), 11
phonetype (Trunofficial.Phone attribute), 11
provider (Trunofficial.Trunofficial attribute), 8

S

score (Trunofficial.Trunofficial attribute), 8
sources (Trunofficial.Trunofficial attribute), 8

sourcestats (Trunofficial.Trunofficial attribute), 8
spamscore (Trunofficial.Phone attribute), 11
spamtype (Trunofficial.Phone attribute), 11

T

tags (Trunofficial.Trunofficial attribute), 8
timezone (Trunofficial.Address attribute), 13
trace (Trunofficial.Trunofficial attribute), 8
Trunofficial (module), 1
trunofficial.Address (class in Trunofficial), 13
trunofficial.Phone (class in Trunofficial), 11
trunofficial.search() (in module Trunofficial), 7
type (Trunofficial.Address attribute), 13