

---

# **Under The Sea Documentation**

***Release 1.1.9***

**Vu Anh**

**Feb 10, 2023**



<b>1</b>	<b>Underthesea - Vietnamese NLP Toolkit</b>	<b>3</b>
<b>2</b>	<b>AUTHORS</b>	<b>9</b>
<b>3</b>	<b>History</b>	<b>11</b>
<b>4</b>	<b>The API Documentation / Guide</b>	<b>17</b>
<b>5</b>	<b>Indices and tables</b>	<b>21</b>
	<b>Python Module Index</b>	<b>23</b>
	<b>Index</b>	<b>25</b>



Vietnamese NLP Toolkit



---

## Underthesea - Vietnamese NLP Toolkit

---



**underthesea** is a suite of open source Python modules, data sets and tutorials supporting research and development in Vietnamese Natural Language Processing.

**Version 1.3.0 out now!** [Underthesea meet deep learning!](#)

Free software	GNU General Public License v3
Live demo	<a href="http://undertheseanlp.com">undertheseanlp.com</a>
Colab notebooks	<a href="#">latest / stable</a>
Documentation	<a href="#">Underthesea Documentation</a>
Facebook	<a href="#">Underthesea Page</a>
Youtube	<a href="#">Underthesea NLP Channel</a>

## 1.1 Installation

To install underthesea, simply:

```
$ pip install underthesea
```

Satisfaction, guaranteed.

## 1.2 Usage

- 1. Sentence Segmentation
- 2. Word Segmentation
- 3. POS Tagging
- 4. Chunking
- 5. Dependency Parsing
- 6. Named Entity Recognition
- 7. Text Classification
- 8. Sentiment Analysis
- 9. Vietnamese NLP Resources

### 1.2.1 1. Sentence Segmentation

Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import sent_tokenize
>>> text = 'Taylor cho bit lúc đầu cô cảm thấy ngại vì cô bạn thân Amanda nhng ri mi th_
↳trôi qua nhanh chóng. Amanda cũng thôi mái vì mi quan h này.'

>>> sent_tokenize(text)
[
    "Taylor cho bit lúc đầu cô cảm thấy ngại vì cô bạn thân Amanda nhng ri mi th trôi qua_
↳nhanh chóng.",
    "Amanda cũng thôi mái vì mi quan h này."
]
```



## 1.2.2 2. Word Segmentation

Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import word_tokenize
>>> sentence = 'Chàng trai 9X Qung Tr khi nghip t nm sò'

>>> word_tokenize(sentence)
['Chàng trai', '9X', 'Qung Tr', 'khi nghip', 't', 'nm', 'sò']

>>> word_tokenize(sentence, format="text")
'Chàng_trai 9X Qung_Tr khi_nghip t nm sò'
```

## 1.2.3 3. POS Tagging

Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import pos_tag
>>> pos_tag('Ch tht chó ni ting Sài Gòn b truy quét')
[('Ch', 'N'),
 ('tht', 'N'),
 ('chó', 'N'),
 ('ni ting', 'A'),
 ('', 'E'),
 ('Sài Gòn', 'Np'),
 ('b', 'V'),
 ('truy quét', 'V')]
```

## 1.2.4 4. Chunking

Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import chunk
>>> text = 'Bác sĩ bây gi có th thn nhiên báo tin bnh nhân b ung th?'
>>> chunk(text)
[('Bác sĩ', 'N', 'B-NP'),
 ('bây gi', 'P', 'I-NP'),
 ('có th', 'R', 'B-VP'),
 ('thn nhiên', 'V', 'I-VP'),
 ('báo tin', 'N', 'B-NP'),
 ('bnh nhân', 'N', 'I-NP'),
 ('b', 'V', 'B-VP'),
 ('ung th', 'N', 'I-VP'),
 ('?', 'CH', 'O')]
```

## 1.2.5 5. Dependency Parsing

Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import dependency_parse
>>> text = 'Ti 29/11, Vit Nam thêm 2 ca mc Covid-19'
>>> dependency_parse(text)
[('Ti', 5, 'obl:tmod'),
 ('29/11', 1, 'flat:date'),
 (',', 1, 'punct'),
 ('Vit Nam', 5, 'nsubj'),
 ('thêm', 0, 'root'),
 ('2', 7, 'nummod'),
 ('ca', 5, 'obj'),
 ('mc', 7, 'nmod'),
 ('Covid-19', 8, 'nummod')]
```

### 1.2.6 6. Named Entity Recognition

#### Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import ner
>>> text = 'Cha tit l lch trình ti Vit Nam ca Tng thng M Donald Trump'
>>> ner(text)
[('Cha', 'R', 'O', 'O'),
 ('tit l', 'V', 'B-VP', 'O'),
 ('lch trình', 'V', 'B-VP', 'O'),
 ('ti', 'E', 'B-PP', 'O'),
 ('Vit Nam', 'Np', 'B-NP', 'B-LOC'),
 ('ca', 'E', 'B-PP', 'O'),
 ('Tng thng', 'N', 'B-NP', 'O'),
 ('M', 'Np', 'B-NP', 'B-LOC'),
 ('Donald', 'Np', 'B-NP', 'B-PER'),
 ('Trump', 'Np', 'B-NP', 'I-PER')]
```

### 1.2.7 7. Text Classification

#### Download models

```
$ underthesea download-model TC_GENERAL
$ underthesea download-model TC_BANK
```

#### Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import classify

>>> classify('HLV đầu tiên Premier League b sa thi sau 4 vòng đầu')
['The thao']
>>> classify('Hi đng t vn kinh doanh Asean vinh danh giı thng quc t')
['Kinh doanh']

>> classify('Lãi suất t BIDV rt u đãi', domain='bank')
['INTEREST_RATE']
```

## 1.2.8 8. Sentiment Analysis

### Download models

```
$ underthesea download-model SA_GENERAL
$ underthesea download-model SA_BANK
```

### Usage

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import sentiment

>>> sentiment('hàng kém cht lg,chăn đp lên dính lông lá khp ngi. tht vng')
negative
>>> sentiment('Sn phm hi nh so vi tng tng nhng cht lng tt, đóng gói cn thn.')
positive

>>> sentiment('Đky qua đng link bài vit này t th 6 mà gi cha thy ai lhe ht', domain=
↳ 'bank')
['CUSTOMER_SUPPORT#negative']
>>> sentiment('Xem li vn thy xúc đng và t hào v BIDV ca mình', domain='bank')
['TRADEMARK#positive']
```

## 1.2.9 9. Vietnamese NLP Resources

### List resources

```
$ underthesea list-data
| Name          | Type          | License  | Year  | Directory          |
|-----+-----+-----+-----+-----+
| UTS2017-BANK  | Categorized  | Open     | 2017  | datasets/UTS2017-BANK |
| VNESES        | Plaintext    | Open     | 2012  | datasets/LTA         |
| VNTQ_BIG      | Plaintext    | Open     | 2012  | datasets/LTA         |
| VNTQ_SMALL    | Plaintext    | Open     | 2012  | datasets/LTA         |
| VNTC          | Categorized  | Open     | 2007  | datasets/VNTC        |

$ underthesea list-data --all
```

### Download resources

```
$ underthesea download-data VNTC
100%|| 74846806/74846806 [00:09<00:00, 8243779.16B/s]
Resource VNTC is downloaded in ~/.underthesea/datasets/VNTC folder
```

## 1.3 Up Coming Features

- Machine Translation
- Text to Speech
- Automatic Speech Recognition

## 1.4 Contributing

Do you want to contribute with underthesea development? Great! Please read more details at [CONTRIBUTING.rst](#).

### 2.1 Original Authors

- Vu Anh <anhv.ict91@gmail.com>

### 2.2 Awesome Contributors

- Nguyen Dang Duc Tai <tainguyen7595@gmail.com>
- Bui Nhat Anh <buinhatanh1208@gmail.com>
- Vuong Quoc Binh <binh@hauai.vn>
- Doan Viet Dung <doanvietdung273@gmail.com>

### 2.3 Thanks

Thanks to all the wonderful folks who have contributed to schedule over the years

- Nhu Bao Vu <nhubaovu@gmail.com>
- Hoai-Thu Vuong <thuvh87@gmail.com>



#### 3.1 6.1.0 (2023-02-08)

- GH-641: fix issue filename of dataset is not correct by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/642>

#### 3.2 6.0.3 (2023-01-25)

- GH-622: Initialize Dictionary page feature by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/633>
- GH-622: Add dictionary page by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/634>

#### 3.3 6.0.2 (2023-01-17)

- GH-628: Create unittest for django API by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/629>
- GH-607: add test frontend with jest by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/630>

**Full Changelog:** <https://github.com/undertheseanlp/underthesea/compare/v6.0.1...v6.0.2>

#### 3.4 6.0.1 (2023-01-08)

- GH-607: add Articles UI by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/620>
- GH-621: Corpus CP\_Vietnamese\_VLC\_v2\_2022 by @rain1024 in <https://github.com/undertheseanlp/underthesea/pull/624>

## 3.5 6.0.0 (2023-01-01)

- Happy New Year 2023! Let's bump up the version! (GH-616)

## 3.6 1.4.1 (2022-12-17)

- Create underthesea app (GH-607)
- Add viet2ipa module (GH-437)
- Training NER model with VLSP2016 dataset using BERT (GH-437)
- Remove unicode as a dependency (GH-569)

## 3.7 1.3.5 (2022-10-31)

- Add Text Normalization module (GH-534)
- Release underthesea\_core version 0.0.5a2 (GH-550)
- Support GLIBC\_2.17 (GH-530)
- Update resources path (GH-540)
- Fix function word\_tokenize (GH-528)

## 3.8 1.3.4 (2022-01-08)

- Demo chatbot with rasa (GH-513)
- Lite version of underthesea (GH-505)
- Increase word\_tokenize speed 1.5 times (GH-185)
- Add build for windows (GH-185)

## 3.9 1.3.3 (2021-09-02)

- Update torch and transformer dependency (GH-403)

## 3.10 1.3.2 (2021-08-04)

- Publish two ABSA open datasets (GH-417)
- Migrate from travis-ci to github actions (GH-410)
- Update ParserTrainer (GH-392)
- Add pipeline folder (GH-351)



### 3.11 1.3.1 (2021-01-11)

- Compatible with newer version of scikit-learn (GH-313)
- Retrain classification and sentiment models with latest version of scikit-learn (GH-381)
- Add ClassifierTrainer (from languageflow) (GH-381)
- Add 3 new datasets (GH-351)
- [Funny Update] Change underthesea's avatar (GH-371)
- [CI] Add Stale App: Automatically close stale Issues and Pull Requests that tend to accumulate during a project (GH-351)

### 3.12 1.3.0 (2020-12-11)

- Remove languageflow dependency (GH-364)
- Remove tabulate dependency (GH-364)
- Remove scores in text classification and sentiment section (GH-351)
- Add information of dependency\_parse module in info function (GH-351)
- Try to use Github Actions (GH-353)
- Dependency Parsing (GH-157)

### 3.13 1.2.3 (2020-11-28)

- Refactor config for resources (GH-300)
- Thêm API x lý d liu (GH-299)

### 3.14 1.2.2 (2020-11-04)

- Remove nltk strict version (GH-308)
- Add word\_hyphen rule (GH-290)
- Sanity check python version (GH-320)
- Handle exception case in sentiment module (GH-321)
- Cp nh t qun lý resources t languageflow (GH-295)
- Loi b languageflow trong quá trình cài đ t (GH-295)
- Cp nh t phiên bn fasttext (GH-304)

### 3.15 1.1.16 (2019-06-15)

- Bumping up version of the languageflow dependency (GH-231)
- Update phiên bản scikit-learn 0.20.2 (GH-229)
- Cập nhật các dependencies (GH-241)
- Cập nhật mô hình trên bộ dữ liệu VNTC (GH-246)
- Cập nhật mô hình trên bộ dữ liệu UTS2017\_BANK\_TC (GH-243)
- Cập nhật mô hình trên bộ dữ liệu UTS2017\_BANK\_SA (GH-244)
- Liệt kê các câu sentiment demo (GH-236)
- Tổng hợp cách đặt tên và quản lý model (GH-225)

### 3.16 1.1.12 (2019-03-13)

- Add sentence segmentation feature

### 3.17 1.1.9 (2019-01-01)

- Improve speed of word\_tokenize function
- Only support python 3.6+
- Use flake8 for style guide enforcement

### 3.18 1.1.8 (2018-06-20)

- Fix word\_tokenize error when text contains tab (t) character
- Fix regex\_tokenize with url

### 3.19 1.1.7 (2018-04-12)

- Rename word\_sent function to word\_tokenize
- Refactor version control in setup.py file and \_\_init\_\_.py file
- Update documentation badge url

### 3.20 1.1.6 (2017-12-26)

- New feature: aspect sentiment analysis
- Integrate with languageflow 1.1.6
- Fix bug tokenize string with '=' (#159)

### 3.21 1.1.5 (2017-10-12)

- New feature: named entity recognition
- Refactor and update model for word\_sent, pos\_tag, chunking

### 3.22 1.1.4 (2017-09-12)

- New feature: text classification
- [bug] Fix Text error
- [doc] Add facebook link

### 3.23 1.1.3 (2017-08-30)

- Add live demo: <https://underthesea.herokuapp.com/>

### 3.24 1.1.2 (2017-08-22)

- Add dictionary

### 3.25 1.1.1 (2017-07-05)

- Support Python 3
- Refactor feature\_engineering code

### 3.26 1.1.0 (2017-05-30)

- Add chunking feature
- Add pos\_tag feature
- Add word\_sent feature, fix performance
- Add Corpus class
- Add Transformer classes
- Integrated with dictionary of Ho Ngoc Duc
- Add travis-CI, auto build with PyPI

### 3.27 1.0.0 (2017-03-01)

- First release on PyPI.
- First release on Readthedocs



If you are looking for information on a specific function, class, or method, this part of the documentation is for you.

## 4.1 Developer Interface

### 4.1.1 word\_tokenize

`underthesea.word_tokenize(sentence, format=None, use_token_normalize=True)`

Vietnamese word segmentation

#### Parameters

- **sentence** (*str*) – raw sentence
- **format** (*str*, *optional*) – format option. Defaults to None. use format='text' for text format
- **use\_token\_normalize** (*bool*) – True if use token\_normalize

**Returns** word tokens

**Return type** `list of str`

#### Examples

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import word_tokenize
>>> sentence = "Bác sĩ bây giờ có thể tự nhiên báo tin bệnh nhân bị ung thư"
```

```
>>> word_tokenize(sentence)
["Bác sĩ", "bây giờ", "có thể", "tự nhiên", "báo tin", "bệnh nhân", "bị", "ung thư"]
```

```
>>> word_tokenize(sentence, format="text")
"Bác_sĩ bây_gi có_th thn_nhiên báo_tin bnh_nhân b ung_th"
```

### 4.1.2 pos\_tag

`underthesea.pos_tag(sentence, format=None)`

Vietnamese POS tagging

**Parameters** `sentence` (`{unicode, str}`) – Raw sentence

**Returns** `tokens` – tagged sentence

**Return type** list of tuple with word, pos tag

#### Examples

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import pos_tag
>>> sentence = "Ch tht chó ni ting TPHCM b truy quét"
>>> pos_tag(sentence)
[('Ch', 'N'),
 ('tht', 'N'),
 ('chó', 'N'),
 ('ni ting', 'A'),
 ('', 'E'),
 ('TPHCM', 'Np'),
 ('b', 'V'),
 ('truy quét', 'V')]
```

### 4.1.3 chunking

`underthesea.chunk(sentence, format=None)`

Vietnamese chunking

**Parameters** `sentence` (`{unicode, str}`) – raw sentence

**Returns** `tokens` – tagged sentence

**Return type** list of tuple with word, pos tag, chunking tag

#### Examples

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import chunk
>>> sentence = "Nghị vn 4 thi th Triu Tiên trôi dt b bin Nht Bn"
>>> chunk(sentence)
[('Nghị vn', 'N', 'B-NP'),
 ('4', 'M', 'B-NP'),
 ('thi th', 'N', 'B-NP'),
 ('Triu Tiên', 'Np', 'B-NP'),
 ('trôi dt', 'V', 'B-VP'),
 ('b bin', 'N', 'B-NP'),
 ('Nht Bn', 'Np', 'B-NP')]
```

### 4.1.4 ner

`underthesea.ner(sentence, format=None, deep=False)`

Location and classify named entities in text

**Parameters** `sentence` (`{unicode, str}`) – raw sentence

**Returns** `tokens`

**Return type** list of tuple with word, pos tag, chunking tag, ner tag tagged sentence

#### Examples

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea import ner
>>> sentence = "Ông Putin ca ngi nhng thành tu vĩ đi ca Liên Xô"
>>> ner(sentence)
[('Ông', 'Nc', 'B-NP', 'O'),
 ('Putin', 'Np', 'B-NP', 'B-PER'),
 ('ca ngi', 'V', 'B-VP', 'O'),
 ('nhng', 'L', 'B-NP', 'O'),
 ('thành tu', 'N', 'B-NP', 'O'),
 ('vĩ đi', 'A', 'B-AP', 'O'),
 ('ca', 'E', 'B-PP', 'O'),
 ('Liên Xô', 'Np', 'B-NP', 'B-LOC')]
```

### 4.1.5 classify

Install dependencies and download default model

```
$ pip install Cython
$ pip install future scipy numpy scikit-learn
$ pip install -U fasttext --no-cache-dir --no-deps --force-reinstall
$ underthesea data
```

`underthesea.classify(X, domain=None)`

Text classification

**Parameters**

- `X` (`{unicode, str}`) – raw sentence
- `domain` (`{None, 'bank'}`) –  
**domain of text**
  - None: general domain
  - bank: bank domain

**Returns** `tokens` – categories of sentence

**Return type** `list`

### 4.1.6 sentiment

Install dependencies

```
$ pip install future scipy numpy scikit-learn==0.19.2 joblib
```

`underthesea.sentiment(X, domain='general')`

Sentiment Analysis

### Parameters

- **X** (*str*) – raw sentence
- **domain** (*str*) – domain of text (bank or general). Default: *general*

### Returns

- **Text** (*Text of input sentence*)
- **Labels** (*Sentiment of sentence*)

### Examples

```
>>> from underthesea import sentiment
>>> sentence = "Chuyen tin k nhn Dc tien"
>>> sentiment(sentence, domain='bank')
[MONEY_TRANSFER#negative (1.0)]
```

## 4.1.7 viet2ipa

`underthesea.pipeline.ipa.viet2ipa(text: str, *args, **kwargs)`

Generate ipa of the syllable

Vietnamese syllabic structure (Anh & Trang 2022)

syllable = onset + rhyme + tone

rhyme = medial + nuclear vowel + (coda)

### Parameters

- **text** (*str*) – represents syllable
- **dialect** (*str*) – Either the string “north” or “south”. Default: *north*
- **eight** (*boolean*) – If true, use eight tone format, else use six tone format. Default: *False*
- **tone** (*str*) – Either the string “ipa” or “number”. Default: *number*

**Returns** A *string*. Represents ipa of the syllable

### Examples

```
>>> # -*- coding: utf-8 -*-
>>> from underthesea.pipeline.ipa import viet2ipa
>>> viet2ipa("trng")
to32
```



## CHAPTER 5

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**u**

`underthesea`, [17](#)



## C

`chunk()` (*in module underthesea*), [18](#)  
`classify()` (*in module underthesea*), [19](#)

## N

`ner()` (*in module underthesea*), [19](#)

## P

`pos_tag()` (*in module underthesea*), [18](#)

## S

`sentiment()` (*in module underthesea*), [20](#)

## U

`underthesea` (*module*), [17](#)

## V

`viet2ipa()` (*in module underthesea.pipeline.ipa*), [20](#)

## W

`word_tokenize()` (*in module underthesea*), [17](#)