

---

# **Python GTK Spellcheck Documentation**

*Release 3.0*

**Maximilian Köhl & Carlos Jenkins**

June 27, 2016



<b>1</b>	<b>Features</b>	<b>3</b>
<b>2</b>	<b>Download</b>	<b>5</b>
<b>3</b>	<b>API Reference</b>	<b>7</b>
<b>4</b>	<b>Examples</b>	<b>11</b>
<b>5</b>	<b>License</b>	<b>13</b>



PyGtkSpellCheck is a spellchecking library written in pure Python for Gtk based on [Enchant](#). It supports both Gtk's Python bindings, [PyGObject](#) and [PyGtk](#), and for both Python 2 and 3 with automatic switching and binding autodetection. For automatic translation of the user interface it can use GEdit's translation files.



---

### Features

---

- Localized names of the available languages.
- Supports word, line and multiline ignore regexes.
- Supports ignore custom tags on Gtk's TextBuffer.
- Enable and disable of spellchecking with preferences memory.
- Supports hotswap of Gtk's TextBuffers.
- PyGObject and PyGtk compatible with automatic detection.
- Python 2 and 3 support.
- As Enchant, support for Hunspell (LibreOffice) and Aspell (GNU) dictionaries.





## 2.1 Source distribution

PyPI package available at: <http://pypi.python.org/pypi/pygtkspellcheck/>

```
pip install pygtkspellcheck
```

## 2.2 Ubuntu/Debian

Install packages:

- **Python 3:**

- `sudo apt-get install python3-gtkspellcheck`
- [https://github.com/downloads/koehlma/pygtkspellcheck/python3-gtkspellcheck\\_3.0-1\\_all.deb](https://github.com/downloads/koehlma/pygtkspellcheck/python3-gtkspellcheck_3.0-1_all.deb)

- **Python 2:**

- `sudo apt-get install python-gtkspellcheck`
- [https://github.com/downloads/koehlma/pygtkspellcheck/python-gtkspellcheck\\_3.0-1\\_all.deb](https://github.com/downloads/koehlma/pygtkspellcheck/python-gtkspellcheck_3.0-1_all.deb)

- **Documentation:**

- `sudo apt-get install python-gtkspellcheck-doc`
- [https://github.com/downloads/koehlma/pygtkspellcheck/python-gtkspellcheck-doc\\_3.0-1\\_all.deb](https://github.com/downloads/koehlma/pygtkspellcheck/python-gtkspellcheck-doc_3.0-1_all.deb)

## 2.3 Archlinux

Available in the Archlinux User Repository:

- **Python 3:**

- `yaourt -S python-gtkspellcheck`
- <https://aur.archlinux.org/packages.php?ID=61200>
- <https://github.com/downloads/koehlma/pygtkspellcheck/python-gtkspellcheck-3.0-1-any.pkg.tar.xz>

- **Python 2:**

- `yaourt -S python2-gtkspellcheck`

- <https://aur.archlinux.org/packages.php?ID=61199>
- <https://github.com/downloads/koehlma/pygtkspellcheck/python2-gtkspellcheck-3.0-1-any.pkg.tar.xz>

## 2.4 Hacking

Development repository is available at: <https://github.com/koehlma/pygtkspellcheck>

```
git clone git://github.com/koehlma/pygtkspellcheck.git
```

Or download last sources in a [ZIP](#) or [Tarball](#) file.

---

## API Reference

---

The main object is called Spellchecker and can be associated with any GtkTextView:

### 3.1 SpellChecker class reference

**class** `gtkspellcheck.spellcheck.SpellChecker` (*view*, *language='en'*, *prefix='gtkspellchecker'*,  
*collapse=True*, *params={}*)

Main spellchecking class, everything important happens here.

#### Parameters

- **view** – GtkTextView the SpellChecker should be attached to.
- **language** – the language which should be used for spellchecking. Use a combination of two letter lower-case ISO 639 language code with a two letter upper-case ISO 3166 country code, for example en\_US or de\_DE.
- **prefix** – a prefix for some internal GtkTextMarks.
- **collapse** – enclose suggestions in its own menu.
- **params** – dictionary with Enchant broker parameters that should be set e.g. *enchant.myspell.dictionary.path*.

#### languages

A list of supported languages.

#### **exists** (*language*)

checks if a language exists

**Parameters** **language** – language to check

#### **add\_to\_dictionary** (*word*)

Adds a word to user's dictionary.

**Parameters** **word** – the word to add

#### **append\_filter** (*regex*, *filter\_type*)

Append a new filter to the filter list. Filters are useful to ignore some misspelled words based on regular expressions.

#### Parameters

- **regex** – the regex used for filtering
- **filter\_type** – the type of the filter

Filter Types:

`SpellChecker.FILTER_WORD`: The regex must match the whole word you want to filter. The word separation is done by Pango's word separation algorithm so, for example, urls won't work here because they are split in many words.

`SpellChecker.FILTER_LINE`: If the expression you want to match is a single line expression use this type. It should not be an open end expression because then the rest of the line with the text you want to filter will become correct.

`SpellChecker.FILTER_TEXT`: Use this if you want to filter multiline expressions. The regex will be compiled with the *MULTILINE* flag. Same with open end expressions apply here.

**append\_ignore\_tag** (*tag*)

Appends a tag to the list of ignored tags. A string will be automatic resolved into a tag object.

**Parameters** *tag* – tag object or tag name

**buffer\_initialize** ()

Initialize the `GtkTextBuffer` associated with the `GtkTextView`. If you associate a new `GtkTextBuffer` with the `GtkTextView` call this method.

**check\_range** (*start, end, force\_all=False*)

Checks a specified range between two `GtkTextIters`.

**Parameters**

- **start** – start iter - checking starts here
- **end** – end iter - checking ends here

**disable** ()

Disable spellchecking.

**enable** ()

Enable spellchecking.

**enabled**

Enable or disable spellchecking

**ignore\_all** (*word*)

Ignores a word for the current session.

**Parameters** *word* – the word to ignore

**language**

The language used for spellchecking

**recheck** ()

Rechecks the spelling of the whole text.

**remove\_filter** (*regex, filter\_type*)

Remove a filter from the filter list.

**Parameters**

- **regex** – the regex which used for filtering
- **filter\_type** – the type of the filter

**remove\_ignore\_tag** (*tag*)

Removes a tag from the list of ignored tags. A string will be automatic resolved into a tag object.

**Parameters** *tag* – tag object or tag name

This library also includes a utility module to unpack LibreOffice `.oxt` extension dictionaries (Hunspell). This is especially useful for MS Windows users to include dictionaries for this library. Use this to extract the Hunspell dictionaries out of the extension and then pass to the Spellchecker the path to the location of the extraction in the `params` argument with the key `enchant.myspell.dictionary.path`.

## 3.2 `oxt_import` module reference

`gtkspellcheck.oxt_import.deflate_oxt` (*oxt\_path*, *extract\_path*, *override=False*,  
*move\_path=None*)  
Uncompress, read and install LibreOffice `.oxt` dictionaries extensions.

### Parameters

- **`oxt_path`** – path to a directory containing the `.oxt` extensions.
- **`extract_path`** – path to extract Hunspell dictionaries files.
- **`override`** – override files.
- **`move_path`** – Optional path to move the `.oxt` files after processing.

### Return type `None`

This function extracts the Hunspell dictionaries (`.dic` and `.aff` files) from all the `.oxt` extensions found on `oxt_path` directory to the `extract_path` directory.

Extensions like the ones found here:

<http://extensions.services.openoffice.org/dictionary>

In detail, this functions does the following:

1. Find all the `.oxt` extension files within `oxt_path`
2. Open (unzip) each extension.
3. Find the dictionary definition file within (*dictionaries.xcu*)
4. Parse the dictionary definition file and locate the dictionaries files.
5. Uncompress those files to `extract_path`.

By default file overriding is disabled, set `override` parameter to `True` if you want to enable it. As an additional option, each processed extension can be moved to `move_path`.



---

**Examples**

---

- [PyGObject Simple Example](#)
- [PyGtk Simple Example](#)





---

**License**

---

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.



**A**

add\_to\_dictionary() (gtkspellcheck.spellcheck.SpellChecker method), 7  
 append\_filter() (gtkspellcheck.spellcheck.SpellChecker method), 7  
 append\_ignore\_tag() (gtkspellcheck.spellcheck.SpellChecker method), 8  
 remove\_filter() (gtkspellcheck.spellcheck.SpellChecker method), 8  
 remove\_ignore\_tag() (gtkspellcheck.spellcheck.SpellChecker method), 8

**S**

SpellChecker (class in gtkspellcheck.spellcheck), 7  
 SpellChecker.exists() (built-in function), 7

**B**

buffer\_initialize() (gtkspellcheck.spellcheck.SpellChecker method), 8

**C**

check\_range() (gtkspellcheck.spellcheck.SpellChecker method), 8

**D**

deflate\_oxt() (in module gtkspellcheck.oxt\_import), 9  
 disable() (gtkspellcheck.spellcheck.SpellChecker method), 8

**E**

enable() (gtkspellcheck.spellcheck.SpellChecker method), 8  
 enabled (gtkspellcheck.spellcheck.SpellChecker attribute), 8

**I**

ignore\_all() (gtkspellcheck.spellcheck.SpellChecker method), 8

**L**

language (gtkspellcheck.spellcheck.SpellChecker attribute), 8  
 languages (SpellChecker attribute), 7

**R**

recheck() (gtkspellcheck.spellcheck.SpellChecker method), 8