

Wavelet Prosody Toolkit

Installation Guide

Eirik Tengesdal 

eirik.tengesdal@oslomet.no

01.02.2026

This guide provides **cross-platform installation instructions** for the *Wavelet Prosody Toolkit* on Windows, macOS, and Linux. Two installation methods are provided: a global user-level installation for general use across multiple projects, and a per-project installation with isolated dependencies for reproducible research environments.

1 Introduction

The *Wavelet Prosody Toolkit* is a powerful tool for analysing prosodic features in speech. This guide helps you install it on Windows, macOS, or Linux, but no guarantees are made. It describes files that are included in the installation package, pertaining to two installation methods: a global user-level installation and a per-project installation.

You can download the installation scripts from here:

1.1 Support

For installation issues, please email Eirik Tengesdal at eirik.tengesdal@oslomet.no.

2 Which Installation Method?¹

2.1 Global Installation (recommended):

- Windows: [install_toolkit_standalone.ps1](#)
- macOS/Linux: [install_toolkit_standalone.sh](#)
- Use across multiple projects without reinstalling

2.2 Per-Project Installation:

- Windows: [install_toolkit_project.ps1](#)
- macOS/Linux: [install_toolkit_project.sh](#)
- Isolated dependencies and strict reproducibility

3 Option 1: Global User-Level Installation

This installs the toolkit once to your user Python environment, making it available from any directory.

Innholdsfortegnelse

1	Introduction	1
2	Which Installation Method?	1
3	Option 1: Global User-Level Installation	1
4	Option 2: Per-Project Installation	3
5	Prerequisites	4
6	Quick Start Example	5
7	Summary of Installation Files	6
8	Additional Resources	6
	Acknowledgements	6

¹For most users, the **global installation** is simpler and more convenient. Use per-project only when you need strict dependency isolation.

Default Locations:
Windows: %USERPROFILE%wavelet_prosody_toolkit
macOS: /wavelet_prosody_toolkit
Linux: /wavelet_prosody_toolkit

3.1 Installation

3.1.1 Windows (PowerShell)

```
1 # Navigate to where you downloaded the script
2 cd path\to\scripts
3
4 # Install to default location
5 .\install_toolkit_standalone.ps1
6
7 # OR install to custom location
8 .\install_toolkit_standalone.ps1 -InstallDir "C:\Tools\wavelet_prosody_toolkit"
```

 PowerShell

3.1.2 macOS/Linux (Bash)

```
1 # Make script executable
2 chmod +x install_toolkit_standalone.sh
3
4 # Install to default location
5 ./install_toolkit_standalone.sh
6
7 # OR install to custom location
8 ./install_toolkit_standalone.sh /path/to/your/preferred/location
```

 Shell

3.2 Usage

After installation, you can use the toolkit from **any directory**:

3.2.1 Windows

```
1 # Launch the GUI
2 python -m wavelet_prosody_toolkit.wavelet_gui
3
4 # Or use in your Python scripts
5 python your_analysis_script.py
```

 PowerShell

3.2.2 macOS/Linux

```
1 # Launch the GUI
2 python3 -m wavelet_prosody_toolkit.wavelet_gui
3
4 # Or use in your Python scripts
5 python3 your_analysis_script.py
```

 Shell

Python Usage

In your Python code (all platforms):

```
1 import wavelet_prosody_toolkit
2 # Use toolkit functions
```

 Python

3.3 Updating the Toolkit

3.3.1 Windows

```
1 cd $HOME\wavelet_prosody_toolkit
2 git pull
```

 PowerShell

3.3.2 macOS/Linux

```
1 cd ~/wavelet_prosody_toolkit
2 git pull
```

 Shell

4 Option 2: Per-Project Installation²

This creates a project-specific installation with its own virtual environment.

²Use this method when you need isolated dependencies for reproducible research environments.

4.1 Installation

4.1.1 Windows (PowerShell)

```
1 # In your project directory
2 .\install_toolkit_project.ps1
```

 PowerShell

4.1.2 macOS/Linux (Bash)

```
1 # In your project directory
2 chmod +x install_toolkit_project.sh
3 ./install_toolkit_project.sh
```

 Shell

4.2 Usage

4.2.1 Windows

```
1 # Activate the virtual environment
2 .\.venv\Scripts\Activate.ps1
3
4 # Launch the GUI
5 python -m wavelet_prosody_toolkit.wavelet_gui
6
7 # Run your analysis scripts
8 python your_analysis_script.py
```

Virtual Environment The toolkit is installed in an isolated environment (.venv) within your project directory.

4.2.2 macOS/Linux

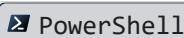
```
1 # Activate the virtual environment
2 source .venv/bin/activate
3
4 # Launch the GUI
5 python -m wavelet_prosody_toolkit.wavelet_gui
6
7 # Run your analysis scripts
8 python your_analysis_script.py
```



4.3 Updating the Toolkit

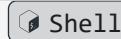
4.3.1 Windows

```
1 cd vendor\wavelet_prosody_toolkit
2 git pull
```



4.3.2 macOS/Linux

```
1 cd vendor/wavelet_prosody_toolkit
2 git pull
```



5 Prerequisites

Before running either script, ensure you have the following installed:

5.1 All Platforms

5.1.1 Python 3

Windows:

```
1 python --version
```



Download from: python.org/downloads

macOS/Linux:

```
1 python3 --version
```



5.1.2 Git

Windows:

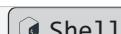
```
1 git --version
```



Download from: git-scm.com/download/win

macOS/Linux:

```
1 git --version
```



Python Version: Python 3.8 or later is recommended. The toolkit may work with earlier versions but has not been tested.

5.2 Platform-Specific

5.2.1 macOS Only

Xcode Command Line Tools:

```
1 xcode-select --install
```

Shell

5.2.2 Windows Only

Microsoft C++ Build Tools (if PyQt6 fails to install):

- Download [Visual Studio Build Tools](#)
- Select «Desktop development with C++»

Dependencies: The following packages will be automatically installed: PyWavelet, scipy, numpy

6 Quick Start Example

6.1 Global Installation

6.1.1 Windows

```
1 # Install once
2 .\install_toolkit_standalone.ps1
3
4 # Navigate to your project
5 cd C:\Users\YourName\my_prosody_analysis
6
7 # Create your analysis script
8 @"
9 import wavelet_prosody_toolkit as wpt
10 # Your analysis code here
11 @" | Out-File -Encoding utf8 analyze.py
12
13 # Run it
14 python analyze.py
```

PowerShell

6.1.2 macOS/Linux

```
1 # Install once
2 ./install_toolkit_standalone.sh
3
4 # Navigate to your project
5 cd ~/my_prosody_analysis
6
7 # Create your analysis script
8 cat > analyze.py << 'EOF'
9 import wavelet_prosody_toolkit as wpt
10 # Your analysis code here
11 EOF
12
13 # Run it
14 python3 analyze.py
```

Shell

6.2 Per-Project Installation

6.2.1 Windows

```
1 # In your project directory
2 .\install_toolkit_project.ps1
3
4 # Activate environment
5 .\venv\Scripts\Activate.ps1
6
7 # Create and run your analysis
8 python analyze.py
```

 PowerShell

6.2.2 macOS/Linux

```
1 # In your project directory
2 ./install_toolkit_project.sh
3
4 # Activate environment
5 source .venv/bin/activate
6
7 # Create and run your analysis
8 python analyze.py
```

 Shell

7 Summary of Installation Files

The following table summarises the installation scripts included in this package:

Purpose	Windows	macOS/Linux
Global installation	<code>install_toolkit_standalone.ps1</code>	<code>install_toolkit_standalone.sh</code>
Per-project installation	<code>install_toolkit_project.ps1</code>	<code>install_toolkit_project.sh</code>

File Distribution: All files are included in the distribution package and can be used independently.

8 Additional Resources

Test for Additional Resources:

- **Toolkit Repository:** github.com/asuni/wavelet_prosody_toolkit
- **Installation Method:** pip install in editable mode (allows toolkit updates via `git pull`)
- **Installation Type:** User-level (no admin/sudo required)

Support: For issues specific to the toolkit itself (not installation), please refer to the GitHub repository's issue tracker.

Acknowledgements

This installation package was created to facilitate the use of the Wavelet Prosody Toolkit across different operating systems and

research environments. Special thanks to the toolkit developers and the Python scientific computing community.