

---

# **News Buddy Documentation**

***Release 0.1***

**Jernej Virag**

January 25, 2016



<b>1</b>	<b>Installation</b>	<b>3</b>
1.1	Standalone Solr installation . . . . .	3
1.2	Newsbuddy setup . . . . .	4
1.3	Production setup with WSGi server . . . . .	5
<b>2</b>	<b>Application start</b>	<b>7</b>
2.1	Development server . . . . .	7
2.2	Production server . . . . .	7
<b>3</b>	<b>Start worker</b>	<b>9</b>
<b>4</b>	<b>Known installaton issues</b>	<b>11</b>
4.1	I have an error on pip install!! . . . . .	11
4.2	I have an error <i>ImportError: No module named _sqlite3!!</i> . . . . .	11
<b>5</b>	<b>Indices and tables</b>	<b>13</b>



A daemon that scrapes news sources and can return news relevant to a query. Uses Solr as a backend.

Contents:



---

## Installation

---

In order to install/run Newsbuddy you need:

- Python 2.6 or newer
- PostgreSQL database 9.x or newer
- Java 1.6 or newer
- (optional) memcached cache server

### 1.1 Standalone Solr installation

While it is **STRONGLY** recommended for Solr to be deployed in a proper servlet container (e.g. Tomcat, Glassfish, etc.), using standalone Jetty is perfectly fine for development purposes.

Newsbuddy requires a recent Solr installation with Lemmagen slovenian language lemmatizer for stemming.

1. Get and unpack Solr from [Solr Homepage](#)
2. Create Solr Home directory, which will hold Solr database, configuration and Newsbuddy index:

```
mkdir ~/solr_home
```

3. Copy pre-made solr.xml file from unpacked distribution:

```
cp ./solr-4.6.0/solr/solr.xml ~/solr_home
```

4. Download [Lemmagen lemmatizer JAR file](#) and copy it into lib subfolder of home directory:

```
mkdir ~/solr_home/lib
wget https://bitbucket.org/mavrik/slovene_lemmatizer/downloads/lemmatizer_solr_1.1.jar
cp lemmatizer_solr_1.1.jar ~/solr_home/lib
```

5. Create Newsbuddy Solr core and copy core configuration files from Newsbuddy source:

```
mkdir ~/solr_home/news
touch ~/solr_home/news/core.properties
mkdir ~/solr_home/news/conf
cp -rv news-buddy/solr/config/* ~/solr_home/news/conf/
```

6. Start Solr with start.jar in unpacked directory with solr\_home directory as solr.solr.home parameter:

```
java -Xmx2G -Dsolr.solr.home=~/solr_home/ -jar ./solr-4.6.0/start.jar &
```

Solr startup must execute without any exceptions. Output log must include:

```
INFO si.virag.solr.LemmagenLemmatizer - Initialized lemmatizer with language mlteast-sl
```

which shows proper lemmatizer configuration.

## 1.2 Newsbuddy setup

Install required memcached lib first (needed even if you won't use memcached):

```
apt-get install libmemcached-dev
```

In the project directory run, to create a virtualenv:

```
virtualenv --no-site-packages .  
source bin/activate
```

Then we need to install the dependencies:

```
pip install -r requirements.txt
```

Copy the default project setting so we can modify them later:

```
cp settings.default.py settings.py
```

### 1.2.1 Setup database

Create a database for newsbuddy in your Postgresql and give proper permissions to user.

Updated settings.py with connection string to your database:

```
DB_CONNECTION_STRING = "postgresql://DATABASE_HOST/DATABASE_NAME"
```

e.g.:

```
DB_CONNECTION_STRING = "postgresql://localhost/news"
```

### 1.2.2 Setup Solr endpoint

Set Solr HTTP endpoint. If you followed instructions at the start of this document your Solr endpoint configuration will be:

```
SOLR_ENDPOINT_URLS = { "si" : "http://localhost:8983/solr/news/" }  
SOLR_DEFAULT_ENDPOINT = "si"
```

### 1.2.3 Setup local path

Setup LOCAL\_URL variable to show to newsbuddy HTTP path. Make sure the URL works from local machine. If you're running a development server the value will be:

```
LOCAL_URL = "http://localhost:8005"
```



### 1.2.4 (Optional) Setup memcached

Newsbuddy performs better with memcached caching server. Install it on your distribution, then updated MEM-CACHED\_URL variable to point to your server instance:

```
MEMCACHED_URL = "127.0.0.1"
```

### 1.2.5 (Optional) Compile minified JavaScript

Minifying JavaScript increases Newsbuddy performance on the clientside. For JS compilation you will need:

- node.js
- Grunt

Before running grunt on newsbuddy for the first time:

```
npm install
```

To compile JS run:

```
grunt
```

in toplevel Newsbuddy directory. This will compile and minify JS files into ui/dist/nb.min.js file.

## 1.3 Production setup with WSGi server

If you wish to run newsbuddy in production with WSGi server, follow steps in previous section, then copy production WSGi configuration file:

```
cp production.default.ini production.ini
```

and update paths to newsbuddy in that file:

```
chdir = /home/newsbuddy/news-buddy virtualenv = /home/newsbuddy/news-buddy
```

INI file can then be used as a parameter for WSGi server.



---

## **Application start**

---

To start this application solr must be running.

The application is the main pyramid web application which provides web interface and search functionality.

### **2.1 Development server**

To start the development server run run:

```
python news_buddy/newsbuddy.py runserver
```

This will start server on port 8005.

### **2.2 Production server**

Production deployment should be run within a WSGi server. See “Production setup with WSGi server” in install docs for instructions on how to create an INI file for that kind of server.



---

### Start worker

---

Make sure you installed the buddy (see installation docs).

The worker is a python script that parses the RSS websites and articles and populates SOLR and resets statistics in the web app. It should be run periodically for regular scraping.

To run the worker your Solr and WSGI application must run.

After activating the virtualenv run:

```
python news_buddy/newsbuddy.py parse-news
```



---

## Known installaton issues

---

### 4.1 I have an error on pip install!!

Make sure you can access the libmemcache while installing.

### 4.2 I have an error *ImportError: No module named \_sqlite3!!*

Make sure you have python with sqlite3 compiled and the .so linked.





---

## Indices and tables

---

- `genindex`
- `search`