Contents:

1 Features 1
  1.1 How is this tool different than others? 1

2 Installing nrkdownload 3
  2.1 Installing or upgrading to the latest release of nrkdownload 3
  2.2 Installing in development mode: 3
  2.3 Uninstalling nrkdownload 3
  2.4 OS considerations 4

3 Installing FFmpeg 5
  3.1 For Windows 5
  3.2 For Linux 5
  3.3 For MacOS 5

4 Usage 7
  4.1 Options 7
  4.2 Configurable download directory 8

5 Examples 9
  5.1 Dowloading series 9
  5.2 Dowloading seasons 10
  5.3 Dowloading programs 10
  5.4 Dowloading podcasts 10

6 Contributing 13
  6.1 How to make changes 13
This is a commandline tool to download content from NRK (Norwegian public broadcaster). It supports downloading TV, radio and podcasts. The tool is written in Python, and is compatible with Python 2.7 and 3.x. It has been tested under Linux, Mac OS X and Windows.

As of autumn 2018, NRK has started to require a secret key in the header of the API requests. This download-tool works even after those restrictions. Users outside of Norway should note that some of the content is geo-restricted.

1.1 How is this tools different than others?

When you download a program with this tool, it doesn’t just download one file. If the program is part of a series, directories for the series and season is created. And the file is automatically named according to its episode and season number. Subtitles and additional images are also automatically downloaded. The subtitles are automatically embedded in the .m4v-file, so you could decide to delete the .srt-file. (I have found that in some tools (like VLC), the support for included subtitles is not perfect. That’s why the separate .srt-file is also there.)

The idea behind all of this is that the downloaded programs should integrate seamlessly into you favorite media server, e.g. Plex. If you for example download all the episodes of the popular series SKAM, you would get a directory-structure like this:

```
SKAM
  └── poster.jpg
    └── Season 01
        ├── SKAM - S01E01 - 1of11.jpg
        ├── SKAM - S01E01 - 1of11.m4v
        └── SKAM - S01E01 - 1of11.no.srt
    └── Season 02
        ├── SKAM - S02E01 - 1of12.jpg
        ├── SKAM - S02E01 - 1of12.m4v
        └── SKAM - S02E01 - 1of12.no.srt
    └── Season 03
```

(continues on next page)
Chapter 1. Features

<table>
<thead>
<tr>
<th>SKAM - S03E01 - 1of10.jpg</th>
</tr>
</thead>
<tbody>
<tr>
<td>SKAM - S03E01 - 1of10.m4v</td>
</tr>
<tr>
<td>SKAM - S03E01 - 1of10.no.srt</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SKAM - S03E10 - 10of10.jpg</th>
</tr>
</thead>
<tbody>
<tr>
<td>SKAM - S03E10 - 10of10.m4v</td>
</tr>
<tr>
<td>SKAM - S03E10 - 10of10.no.srt</td>
</tr>
</tbody>
</table>
Installing nrkdownload

2.1 Installing or upgrading to the latest release of nrkdownload

To install or upgrade to the latest release from PyPI, use the following command:

```
$ pip install -U --user nrkdownload
```

For considerations on Python 2 vs. 3 and the `--user` option, see the section on OS considerations below.

2.2 Installing in development mode:

If you want to change (and possibly contribute to) the code, first clone the GitHub repository. This will create a directory containing a local copy. Then install in develop mode from this directory:

```
$ git clone https://github.com/marhoy/nrk-download.git
$ pip install --user -e nrk-download
```

You will then be able to use the tool as usual, but the installation will be a pointer to your local repository. Whatever changes you make in your local repository will have immediate effect on the “installation”.

You are welcome to help developing nrkdownload, please see the section on Contributing.

2.3 Uninstalling nrkdownload

To uninstall nrkdownload, just type:

```
$ pip uninstall nrkdownload
```

NOTE: This will not uninstall the required packages that might have been installed together with nrkdownload. Type `pip list --user` to list all user-installed packages, and uninstall them if you know that you don’t need them anymore.
2.4 OS considerations

This tool is compatible with both Python 2 and 3. But unless you have other reasons to use Python 2, the latest Python 3.x is recommended.

If your system-wide Python installation is under control of a package manager like rpm or deb, you should avoid pip-installing python packages as root. This can be solved in several ways:

1. Install Python packages under your own home-directory by passing the --user option to the pip installer.
2. Install your own user-specific Python distribution, where you can later install packages. Anaconda (or Mini-conda) is a good choice. It also has good support for environment (see next).
3. Create a virtual environment using standard Python or conda (used by Anaconda/Miniconda) and install packages in that environment.

2.4.1 Special considerations for MacOS (OS X)

MacOS comes by default with an installation of Python 2.7. You can decide to go with this (i.e. not installing Anaconda as mentioned above). In order to install packages you need the package installer pip, and under macOS pip is not installed by default. You can install it by typing sudo easy_install pip. In order to utilize the --user scheme described above, you must also add ~/Library/Python/2.7/bin to your $PATH (edit your ~/.bash_profile). This enables installed Python scripts (like nrkdownload) to be available in the Terminal.

Also, if you get an UnicodeEncodeError, add the following line to your ~/.bash_profile: export LC_CTYPE=en_US.UTF-8

2.4.2 Special considerations for Linux

Your system might have both Python 2 and 3 installed as a part of the Linux-distribution. If Python 2 is the default, pip will be pointing to the Python 2 installation, whereas pip3 will point to the Python 3 installation. If that is the case for you, and you explicitly want to run nrkdownload under Python 3, you must replace pip with pip3 in the examples above.

2.4.3 Special considerations for Windows

Windows does not come with an installation of Python. You can choose to install version 2.7.x or the latest 3.x from python.org. If you want to learn and develop Python I would suggest Anaconda, which installs in your home-directory and comes with a nice selection of packages.
The videos and subtitles are downloaded using FFmpeg. It is available for all major operating systems. You need to install ffmpeg and make it available in your $PATH before you can use nrkdownload.

3.1 For Windows

Should be rather straight forward. Download the latest static build and run the installer.

3.2 For Linux

Depending on your Linux-distribution, you might have to add a package-repository in order to install ffmpeg. If you get stuck, try too Google installing ffmpeg for YOUR_LINUX_DISTRO.

3.3 For MacOS

Download the static build of the latest release. Open the .dmg-file and copy the binary file ffmpeg to e.g. a directory bin inside your home directory. Then, add ~/bin to your PATH.
Before NRK restricted the API, it was possible to search for content. With the current restrictions, you will need to specify an URL for the content you want to download: Start by browsing https://tv.nrk.no (possibly https://tv.nrksuper.no/) or https://radio.nrk.no until you find what you want. Copy the URLs and give it as arguments for this tool. You can list several URLs on the command line (separated by space), or line by line in a file. Content that are already downloaded will be skipped.

4.1 Options


Download series or programs from NRK, complete with images and subtitles.

positional arguments:

URL Specify download source(s). Browse https://tv.nrk.no/ or https://radio.nrk.no/ and copy the URL. The URL can point to a whole series, or just one episode.

optional arguments:

  -h, --help show this help message and exit
  --version show program’s version number and exit
  -d DIRECTORY The directory where the downloaded files will be placed
  -v, --verbose Increase verbosity. Can be repeated up to two times.
  -c, --cache Enable persistent caching of the API requests.
  -a, --all If URL matches several episodes: Download all episodes without asking.
  -l, --last If URL matches several episodes: Download the latest without asking.
  -f FILE, --file FILE Specify a file containing URLs, one URL per line. Specifying urls from a file will automatically enable --all and download all episodes from series.
The files are by default downloaded to ~/Downloads/nrkdownload. This can be changed by using the option -d as described above, or you can define the environment variable NRKDOWNLOAD_DIR

### 4.2 Configurable download directory

If you don’t specify anything, the files and directories will be created inside ~/Downloads/nrkdownload, where ~ means your home directory. If you want the downloads somewhere else, there are two ways to specify a different download directory:

- Define an environment variable named NRKDOWNLOAD_DIR
- Specify the download directory on the command line with the option -d download_dir

If you do both at the same time, the option from the command line will take precedence.

If you don’t know how to define an environment variable under your operating system, try to Google create environment variable and the name of your operating system. (Under Linux and macOS, you would want to edit your ~/.bash_profile)
5.1 Dowloading series

Let’s say you are interested in downloading all the available episodes about the rescue boat “Elias”. You would then search for “Elias” on https://tv.nrk.no and end up at a page with the URL https://tv.nrk.no/serie/elias. This is the URL you want to give as argument to nrkdownload:

```bash
$ nrkdownload https://tv.nrk.no/serie/elias
Not available for download: Elias - Elias: 1:26
Not available for download: Elias - Elias: 2:26

Not available for download: Elias - Sesong 2 - Elias: 26:26
```

Matching programs

```
0: Elias - Elias: 10:26
1: Elias - Elias: 11:26
2: Elias - Elias: 12:26
3: Elias - Elias: 13:26
4: Elias - Elias: 14:26
5: Elias - Elias: 15:26
6: Elias - Elias: 16:26
7: Elias - Elias: 17:26
8: Elias - Elias: 18:26
```

Enter a number or Python-style interval (e.g. 8 or -2: or : ). (q to quit):

All of the registered episodes will be listed, numbered from 0. Due to copyright-issues, some of the episodes might not be available for download. You are then asked to specify what episodes you want to download. You respond to this by typing an integer or a Python-style range. Some examples:

- **4** means the fourth episode.
- **:10** means the 10 first episodes from 0 to 9.
- **5:10** means the five episodes from 5 to 9.
nrkdownload Documentation

10: means all episodes from 10 to the last, including both 10 and the last.
: means all available episodes.

When you have made a selection, the download starts:

```
Ready to download 1 programs, with total duration 0:11:02
Downloading: 26% | 173/662 [00:10<00:11, 43.9s/s]
```

The progress bar shows the number of seconds of video to be downloaded, in this case 0:11:02 = 11*60 + 2 = 662 seconds. The next two times are the estimated remaining and total download time. The last number shows how many seconds of video are downloaded per second. So in this snapshot, the video was downloaded at 44x the playback speed.

If you have already downloaded some of the episodes, those will be automatically skipped (not overwritten).

5.1.1 Downloading without interaction

If you want to download all available episodes with no questions asked, use the `--all` option on the command line. Or if you want to download only the latest episode, use the `--last` option on the command line. If you use any of these options, you will not be asked for what program to download. This allows for running `nrkdownload` as a scheduled job, e.g. via cron.

5.2 Downloading seasons

If you are only interested in a specific season of a series, click around on the webpages until you are only looking at that season. As an example, if you wanted to download the third season of SKAM, you would use an URL like https://tv.nrk.no/serie/skam/sesong/3

5.3 Downloading programs

For downloading a specific program that is not part of a series, the URL could look like e.g. https://tv.nrk.no/program/KOID28004110/pushwagner

5.4 Downloading podcasts

Similarly to TV-series and episodes, you can specify the URL for the whole podcast series or for a specific episode. An example for a whole podcast series would be https://radio.nrk.no/podkast/sjakksnakk/ whereas https://radio.nrk.no/podkast/sjakksnakk/nrkno-poddkast-26616-142542-23102018091100 refers to a specific episode:

```
$ nrkdownload https://radio.nrk.no/podkast/sjakksnakk/
```

Matching programs
0: Sjakksnakk - Episode 1 - #1 Magnus Carlsen frykter å tape (2018-10-23)
1: Sjakksnakk - Episode 2 - #2 Simen Agdestein om ensomheten på toppen...
2: Sjakksnakk - Episode 3 - #3 Jon Ludvig Hammer om livet i Carlsens s...
3: Sjakksnakk - Episode 4 - #4 Ellen Carlsen og Heidi Røneid frykter a...
4: Sjakksnakk - Episode 5 - #5 Hans Olav Lahlum mener Carlsen søker sj...
5: Sjakksnakk - Episode 6 - #6 Torstein Bae og Atle Grønn tror VM blir...

(continues on next page)
Enter a number or Python-style interval (e.g. 8 or -2: or :). (q to quit): :

<table>
<thead>
<tr>
<th>Podcast</th>
<th>Progress</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 of 6</td>
<td>91.2Mb [00:00, 175Mb/s]</td>
</tr>
<tr>
<td>2 of 6</td>
<td>60.9Mb [00:00, 181Mb/s]</td>
</tr>
<tr>
<td>3 of 6</td>
<td>58.8Mb [00:00, 181Mb/s]</td>
</tr>
<tr>
<td>4 of 6</td>
<td>74.6Mb [00:00, 177Mb/s]</td>
</tr>
<tr>
<td>5 of 6</td>
<td>67.4Mb [00:00, 170Mb/s]</td>
</tr>
<tr>
<td>6 of 6</td>
<td>83.5Mb [00:00, 133Mb/s]</td>
</tr>
</tbody>
</table>
The workflow follows the Gitflow workflow where the master branch contains the stable code and is used for making releases. The develop branch is used for development, with possible feature branches.

6.1 How to make changes

You are welcome to help develop nrkdownload, following these steps:

1. Clone the GitHub repository
2. Make a new branch based on the develop branch
3. Add your code or bugfixes
4. Make a pull request towards the develop branch