**Evaluating Capabilities of current online news scraping software**

**Section 1: Motivation**

Two of the most effective open-source news scraping softwares available are [Newspaper3k by Lucas Ou-Yang](https://github.com/codelucas/newspaper/) and [news-please by Felix Hamborg](https://github.com/fhamborg/news-please/). I will attempt to compare these two available options in terms of features, accuracy and speed, and determine the suitability of these scraping libraries in handling:

* Websites of different formats
* Websites in different languages (English, French, Spanish, Portuguese, Russian, Chinese, Indonesian, Korean, Swahili, Arabic)
* Scraping of websites at a high speed (for real-time news coverage)
* Scraping of live news
* Scraping of subscription sites
* News articles with multiple pages (e.g. chinadaily)
* Interactive formats (e.g. BMJ)
* Articles which involve clicking to read more (e.g. CNN Arabic)

**Section 2: Comparison**

**Features:**

Shown below is a table comparing the current available features of Newspaper3k and News-please:

|  |  |  |
| --- | --- | --- |
| **Feature:** | **Newspaper3k** | **news-please** |
| Article Scraping | Yes | Yes |
| All 10 languages supported | Yes | Yes |
| Website Scraping | Yes | CLI Only |
| Robustness to website timeouts | Yes | No |
| SQL database integration | No | Yes |
| NLP (keyword extraction, summary) | Yes | No |

**Notes:**

* news-please partly uses newspaper to perform its scraping
* news-please is not robust in handling timeouts, as e.g. ITV news would never stop scraping and the process had to be manually killed.
* News-please has more informative errors when running on jupyter notebook

**Errors:**

Here are a compiled list of errors/failures for each scraper. Highlighted in light red are failures unique to that scraper (i.e. that the other scraper got correct):

**Newspaper3k:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Language** | **Country** | **Error** | |
| Al Waft | Arabic | EGY | Forbidden access |
| Al-Akhbar | Arabic | LEB | Forbidden access |
| Beijing News | Chinese | CHI | Incorrect parse |
| El Nacional | Spanish | ESP | Forbidden access |
| Health News Network | Chinese | CHI | Timeout |
| IPP Media | Swahili | TNZ | Timeout (possibly due to website) |
| Komsomolskaya Pravda | Russian | RUS | Timeout (possibly due to website) |

Total accuracy: 102/109 sites **(93.58%)**

**News-please:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Language** | **Country** | **Error** |
| Al Waft | Arabic | EGY | Forbidden access |
| Al-Akhbar | Arabic | LEB | Forbidden access |
| Beijing News | Chinese | CHI | Incorrect parse |
| El Nacional | Spanish | ESP | Forbidden access |
| ITV News | English | UK | Timeout |
| Komsomolskaya Pravda | Russian | RUS | Timeout (possibly due to website) |
| Kyunghyang Shinmun | Korean | KOR | Encoding error (still scrapes) |
| New York Times | English | USA | Forbidden access |
| The Telegraph | English | UK | (Slightly) incorrect parse |

Total accuracy: 100/109 sites **(91.74%)**

**Exceptional case handling:**

I tested the following exceptional cases:

-Live news article

-Subscription-restricted article

-Chinadaily article with multiple pages

-BMJ interactive article

-CNN Arabic Article which doesn’t originally include all content

**Newspaper3k:**

**News-please:**

**Speed:**

I tested both languages using news websites I had previously gathered in 10 different languages, and tested the average scraping time per language of 3 different articles on each website, 3 times per article (using timeit). I excluded previously identified crashes from the previous experiment.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Language** | **No. of sources** | **No. correct** | | **Average Time** | |
|  |  | **Newspaper3k** | **News-please** | **Newspaper3k** | **News-please** |
| **ENG** | 28 | 28 | 26 | 536.3332 | 704.3035 |
| **FRA** | 11 | 11 | 11 | 1548.167 | 806.2918 |
| **ESP** | 10 | 9 | 9 | 339.2322 | 435.0289 |
| **ZHO** | 9 | 8 | 9 | 3538.55 | 3137.15 |
| **RUS** | 9 | 8 | 8 | 632.5125 | 755.09 |
| **POR** | 10 | 10 | 10 | 1220.08 | 1289.466 |
| **IND** | 5 | 5 | 5 | 1152.32 | 1243.918 |
| **SWA** | 7 | 6 | 7 | 874.445 | 1253.669 |
| **KOR** | 6 | 6 | 6 | 1629.06 | 1766.223 |
| **ARA** | 14 | 12 | 11 | 513.9133 | 651.8164 |
| **Total** | **109** | **103** | **102** | **59.8734** | **70.29392** |

**Conclusion:**

Both scrapers do a reasonable job on the articles (both over 90% accuracy on 100+ sources in 10 languages, including in a practical speed test) and at similar speeds, which may suggest differences in speed may mostly be accounted for by difference in connection strength, website speed, etc.

Comparing both libraries, it seems as though newspaper is slightly more accurate and faster than news-please (although not all on languages) and as the capabilities are more useful in this project than those of news-please (NLP is more useful than a database, etc.) I believe newspaper3k is the best library in this use case.

There are some improvements to be made in the recognition of difficult sites such as Beijing news and the health news network, and in formats such as

**Limitations:**

-I did not have time to check the details of every article, so I chose a few difficult examples and a few control examples

-I have not tested the full website capability of these scrapers due to news-please only having this option on the command line

-I ignore when a news scraper misses an article a few times, as long as it can scrape. This is because I believe most failures come down to my internet connection rather than the library itself

-I believe Komsomolskaya Pravda was down during the experiment

**Appendix 1: Country codes**

A table to convert the 2/3 letter country codes used in the tables to a country/organisation:

|  |  |
| --- | --- |
| **Country Code** | **Country** |
| **ALG** | Algeria |
| **ARG** | Argentina |
| **AUS** | Australia |
| **BEL** | Belarus |
| **BRA** | Brazil |
| **CAN** | Canada |
| **CBV** | Cabo Verde (Cape Verde) |
| **CHI** | China |
| **COL** | Colombia |
| **DRC** | Democratic Republic of the Congo |
| **EGY** | Egypt |
| **ESP** | Spain |
| **FRA** | France |
| **GBS** | Guinea Bissau |
| **HK** | Hong Kong |
| **IDN** | Indonesia |
| **IRE** | Republic of Ireland |
| **IRQ** | Iraq |
| **ISR** | Israel |
| **JAM** | Jamaica |
| **KEN** | Kenya |
| **KOR** | South Korea |
| **KSA** | Saudi Arabia |
| **LEB** | Lebanon |
| **MEX** | Mexico |
| **MOR** | Morocco |
| **MOZ** | Mozambique |
| **NZ** | New Zealand |
| **POR** | Portugal |
| **QAT** | Qatar |
| **RUS** | Russia |
| **SNG** | Singapore |
| **SUD** | Sudan |
| **SUI** | Switzerland |
| **T&T** | Trinidad and Tobago |
| **TAI** | Taiwan |
| **TNZ** | Tanzania |
| **TUN** | Tunisia |
| **UAE** | United Arab Emirates |
| **UGA** | Uganda |
| **UK** | United Kingdom |
| **UKR** | Ukraine |
| **UN** | United Nations |
| **USA** | United States of America |
| **VEN** | Venezuela |

**Appendix 2: Average speed by source**

[Table]