Pubmed Parser: A Python Parser for PubMed Open-Access XML Subset and MEDLINE XML Dataset

Titipat Achakulvisut¹, Daniel E. Acuna², and Konrad Kording¹

¹ University of Pennsylvania ² Syracuse University

Summary

The number of biomedical publications is increasing exponentially every year. If we had the ability to access, manipulate, and link this information, we could extract knowledge that is perhaps hidden within the figures, text, and citations. In particular, the repositories made available by the PubMed and MEDLINE databases enable these kinds of applications at an unprecedented level. Examples of applications that can be built from this dataset range from predicting novel drug-drug interactions, classifying biomedical text data, searching specific oncological profiles, disambiguating author names, or automatically learning a biomedical ontology. Here, we describe Pubmed Parser (pubmed_parser), a software to mine Pubmed and MEDLINE efficiently. Pubmed Parser is built on top of Python and can therefore be integrated into a myriad of tools for machine learning such as scikit-learn and deep learning such as tensorflow and pytorch.

Pubmed Parser has the capability of parsing multiple pieces of information into structured datasets that other libraries such as medic or MEDLINEXMLToJSON do not have. medic, for example, does not output paragraphs and captions and has been discontinued since 2015. MEDLINEXMLToJSON, similarly, transforms an original XML file into a JSON file, keeping the same structure. It seems also that MEDLINEXMLToJSON development has been inactive since 2016. Our parser can be used within Python and provides results in Python dictionaries. It can parse multiple PubMed and MEDLINE data derivatives including article and journal metadata, authors and affiliations, references, figure captions, paragraphs, and more. For example, Pubmed Parser’s capabilities were used in Tang et al. (2019) to parse authorship lists, affiliations, and MeSH terms as part of a large-scale name disambiguation pipeline. It has also been used in deep learning pipelines such as one described in Nikolov, Pfeiffer, & Hahnloser (2018), a scientific articles summarization based on titles and abstracts. Parsing XML and HTML with Pubmed Parser allows very efficient production of dictionaries or JSON files that can easily be integrated into downstream pipelines. Moreover, the implemented functions can be scaled easily as part of other MapReduce-like infrastructures such as PySpark. This allows users to parse the most recently available corpus and customize the parsing to their needs. Below, we provide an example code to parse an XML file from MEDLINE corpus.

```python
import pubmed_parser as pp
parsed_articles = pp.parse_medline_xml('data/pubmed20n0014.xml.gz',
                                          year_info_only=True,
                                          nlm_category=False,
                                          author_list=False)
```

Pubmed Parser has already been used in published work for several different purposes, including author name disambiguation (Tang et al., 2019), information extraction and summarization.
(Abdeddaïm, Vimard, & Soualmia, 2018; Achakulvisut, Bhagavatula, Acuna, & Kording, 2019; Galea, Laponogov, & Veselkov, 2018; Mesbah, Bozzon, Loﬁ, & Houben, 2018; Nikolov et al., 2018), search engine optimization (Shahri & Kahanda, 2019; Ševa et al., 2019), and biomedical discovery (Miller, 2017; Rakhi, Tuwani, Mukherjee, & Bagler, 2018; Shahri & Kahanda, 2019). It has also been used in multiple biomedical and natural language class projects, and various data science blog posts relating to biomedical text analysis.

Acknowledgements

Titipat Achakulvisut was supported by the Royal Thai Government Scholarship grant #50AC002. Daniel E. Acuna is supported by National Science Foundation grant #1800956.

References


