The Journal of Open Source Software

Biomarker-NLP: A Python Package for Mining Biomarker Information for FDA-Approved Targeted Cancer Therapies

Junxia Lin[1], **Yuezheng He**[1], **Subha Madhavan**[1, 2], **Chul Kim**[3], **and Simina M. Boca**[1, 4]

1 Innovation Center for Biomedical Informatics, Georgetown University Medical Center, Washington, DC, USA **2** Competitive Intelligence, Oncology R&D, AstraZeneca, Gaithersburg, MD, USA **3** Division of Hematology and Oncology, Georgetown Lombardi Comprehensive Cancer Center, Georgetown University, Washington, District of Columbia, USA **4** Early Biometrics & Statistical Innovation, Data Science & Artificial Intelligence, R&D, AstraZeneca, Gaithersburg, MD, USA

DOI: 10.21105/joss.0XXXX

Software

- Review
- Repository
- Archive

Editor: Editor Name

Submitted: 01 January XXXX
Published: 01 January XXXX

Summary

This Biomarker-NLP (`biomarker_nlp`) package aims to provide natural language processing (NLP) functionalities for mining and processing biomarker information for targeted cancer therapies approved by the US Food and Drug Administration (FDA). Treatment biomarkers are specific molecular changes, including mutations and gene or protein expression measurements, that are used to decide whether a certain therapy should be prescribed to an individual. Thus, they are often included in the FDA-approved therapy labels, especially for targeted cancer therapies. Our tool pulls information from two webpages in HTML format: 1) The National Cancer Institute (NCI)'s list of FDA-approved targeted cancer therapies `[@Therapy]` and 2) The National Library of Medicine (NLM)'s DailyMed database of drug labels `[@DailyMed]`. Biomarker-NLP parses the NCI and DailyMed HTML pages using tools in the `lxml` library developed by `@Behnel:2005`. It allows users to quickly and easily scrape certain pieces of information from NCI and DailyMed without requiring them to consider the HTML tree structure. The free text biomarker information is mined and structured into fixed entities, including therapy name, disease name (cancer type), gene or protein in biomarker, name of therapies prescribed in combination, etc. For recognizing the biomarker entities, such as genes and proteins, we utilize the pre-trained named-entity recognition (NER) models from `ScispaCy [@Neumann:2019]`. In addition, as negated biomarkers can be important but challenging to extract, our package provides tools to detect negations in sentences through two pre-trained negation models from `@Khandelwal:2020` `NegBERT` program, which applies a transfer learning approach. One model is the negation cue detection model that detects the negation cues in a sentence, while the second is the negation scope detection model that recognizes the scope of negation in a sentence. As the `NegBERT` program does not provide the output models, we performed the training step and published these two models for free use, integrating them into our package so that users can easily mine the negated biomarker information by using the relevant functions. As an example,

```
>>> from biomarker_nlp import negation_cue_scope
>>> from biomarker_nlp.negation_negbert import *
>>> modelCue = torch.load('/path/to/negation/cue/detection/model') # path
to the model file
>>> modelScope = torch.load('/path/to/negation/scope/detection/model') #
path to the model file
>>> txt = "KEYTRUDA is not recommended for treatment of patients with PMB
```

```
CL who require urgent cytoreductive therapy."
# detect negation cue
>>> negation_cue_scope.negation_detect(text = txt, modelCue = modelCue)
True
# extract the negation scope
>>> negation_cue_scope.negation_scope(text = txt, modelCue = modelCue, mo
delScope = modelScope)
['KEYTRUDA is', 'recommended for treatment of patients with PMBCL who']
```

As we can see from the output above, the `negation_detect()` function detects if a negation cue is presented in a sentence. Afterwards, the `negation_scope()` function extracts the scope of the associated negation cue from the sentence. Then, we can use other functions from the package to detect the biomarkers presented in the resulting scope phrases to get the negated biomarkers.

Statement of need

The NCI website represents a convenient starting location for exploring targeted cancer therapies for patients and physicians, as well as bioinformaticians and biomedical researchers. DailyMed is a database that provides official label information for about 140,000 FDA-approved and FDA-regulated products submitted to the FDA [@DailyMed]. For a drug or a biological product, its label contains prescribing information in a structured textual format. Each label includes various sections, such as the indication and usage and dosage and administration [@DailyMed]. Within each section, information is mostly in free-text format. The information contains a variety of biomedical data, including biomarker information. This biomarker information is valuable for and currently being used by a wide range of stakeholders, such as doctors, bioinformaticians, healthcare providers, and biomedical researchers. However, as this labeling information is in free-text format and may be updated with new indications, searching through multiple labels and reading every word in order to perform curation activities is often time-consuming, low efficiency labor for bioinformaticians. Here, we present Biomarker-NLP, a Python package to process biomedical text and extract biomarker information from NCI and DailyMed efficiently. Curators will still be required to check the NLP output, but are expected to spend substantially less time on these activities. Thus, Biomarker-NLP will be integrated into an "AI-augmented curation" workflow, building on similar work by @Mahmood:2017, which developed a system to extract associations between genomic anomalies and drug responses from the biomedical literature, but focused on cancer therapy labels.

Acknowledgements

This work was completed as part of a project funded by a pilot award (P30CA051008, PI of pilot: Kim).

Declarations

Subha Madhavan and Simina M. Boca are currently employees and minor shareholders of AstraZeneca.

References