SentenCy: An Open-source Tool for Improving Biomedical Named Entity Recognition

Grant Delong, BA[1], Henry K. Philofsky, MD[2], James Elmore, MD[1], Stacey Shriner[1], Sara Hunt, LPN[1], Casey Cauthorn, MS[1], Elliot G. Mitchell, PhD[1], David K. Vawdrey, PhD[1,2], Abdul A. Tariq, PhD[1]

[1]Geisinger Health System, Danville, PA; [2]Columbia University, New York, NY

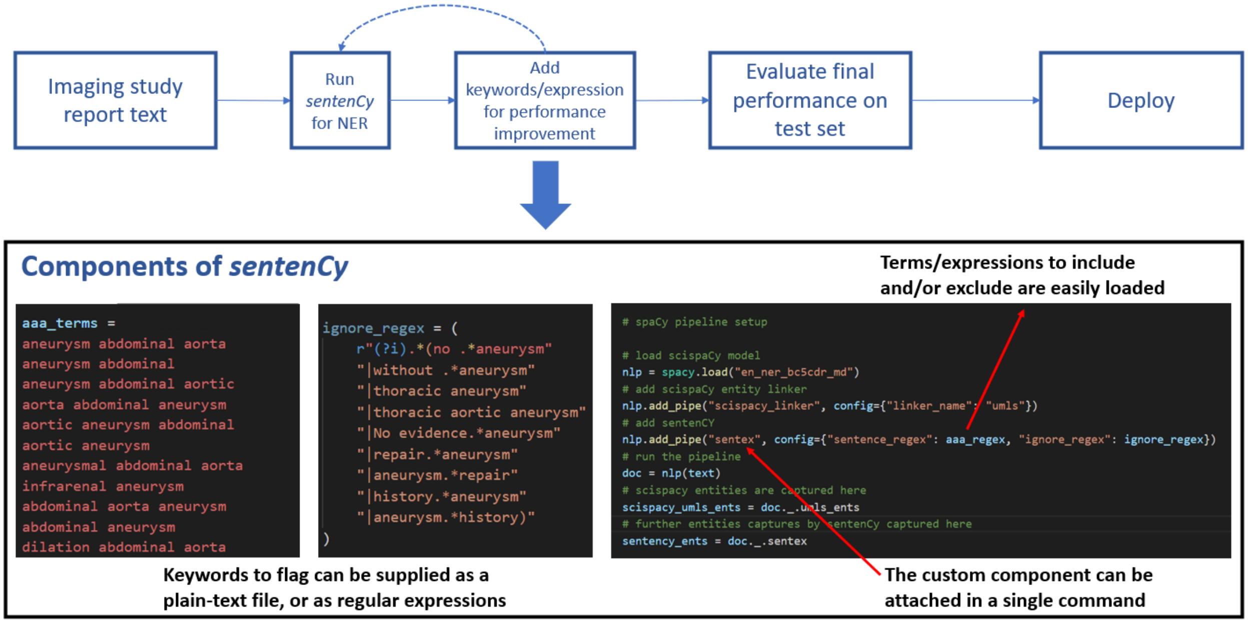
Introduction

Named entity recognition (NER) is an application of natural language processing (NLP) that refers to the automatic identification of terms related to specified concepts[1]. In clinical applications, NER can be used to rapidly tag and extract disease-specific information from biomedical data such as physician notes[2], removing the need for manual review.

A popular library for biomedical NER is *scispaCy,* which extends the functionality of the NLP library *spaCy* for biomedical text[3]. *scispaCy* is growing in popularity and usage within healthcare and biomedicine; many clinical NLP tasks are being ported over to the modular, extensible tool for its ease of development and deployment[4].

In this work, we describe *sentenCy*, an open-source *spaCy*-based library we developed that can be used in conjunction with *scispaCy* to improve NER capabilities in clinical applications. Out of the box, *spaCy* enables use of regular expressions in token-level matching (i.e., matching individual words or punctuation), or document-level matching (which treat the whole text of a clinical note as a single unified entity)[5]. However, this limits functionality for NER because token-level matching has too narrow a scope, as the context surrounding a token is ignored; while document-level matching has too wide a scope, as concepts mentioned far apart in a note are forced to become linked. *sentenCy* adds the novel capability of distinctly capturing information added by each sentence separately in a clinical note. Moreover, since it follows *spaCy's* application programming interface (API), *sentenCy* can be combined with *scispaCy* and other *spaCy* components within a single NLP pipeline.

To showcase its capabilities, we used *sentenCy* to replace and to surpass the performance of a proprietary commercial solution for identifying abdominal aortic aneurysm (AAA) from imaging study reports. *SentenCy* is available in a github repository (https://github.com/g-delong/sentency), enabling other health systems to use it, and we hope to have it included among the list of projects within the *spa Cy* universe[6].


Imaging study report text
Run *sentenCy* for NER
Add keywords/expression for performance improvement
Evaluate final performance on test set
Deploy
Components of *sentenCy*

```
aaa_terms =
aneurysm abdominal aorta
aneurysm abdominal
aneurysm abdominal aortic
aorta abdominal aneurysm
aortic aneurysm abdominal
aortic aneurysm
aneurysmal abdominal aorta
infrarenal aneurysm
abdominal aorta aneurysm
abdominal aneurysm
dilation abdominal aorta
```

```
ignore_regex = (
    r"(?i).*(no .*aneurysm"
    "|without .*aneurysm"
    "|thoracic aneurysm"
    "|thoracic aortic aneurysm"
    "|No evidence.*aneurysm"
    "|repair.*aneurysm"
    "|aneurysm.*repair"
    "|history.*aneurysm"
    "|aneurysm.*history)"
)
```

Keywords to flag can be supplied as a plain-text file, or as regular expressions

Terms/expressions to include and/or exclude are easily loaded

```
# spaCy pipeline setup

# load scispaCy model
nlp = spacy.load("en_ner_bc5cdr_md")
# add scispaCy entity linker
nlp.add_pipe("scispacy_linker", config={"linker_name": "umls"})
# add sentenCY
nlp.add_pipe("sentex", config={"sentence_regex": aaa_regex, "ignore_regex": ignore_regex})
# run the pipeline
doc = nlp(text)
# scispacy entities are captured here
scispacy_umls_ents = doc._.umls_ents
# further entities captures by sentenCy captured here
sentency_ents = doc._.sentex
```

The custom component can be attached in a single command

Figure 1. Users can supply keywords or regular expressions, and *sentenCy* can add these rules to *scispaCy's* medical NLP algorithms. Supplied keywords can also be automatically converted into regular expressions.

Methods

Creation of the custom component

SentenCy is built as a pipeline component that can be added to a *spaCy* NLP pipeline. In Figure 1, we demonstrate a typical workflow that leverages *sentenCy* for sentence-level NER. Addition of keywords, regular expressions, and the instantiation of the pipeline component can all be accomplished through the overarching objected-oriented framework supplied by *spaCy*.

Validation against commercial solution for identifying abdominal aortic aneurysms (AAA)

We used *sentenCy* in combination with *scispaCy' s* pre-trained NER model and UMLS entity linker to tag 23,063 imaging reports from abdominal and chest CT, MRI, and ultrasound as being positive or negative for AAA (this set corresponds to all such imaging studies performed by Geisinger in December 2021). We investigated the output and generated a list of keywords and regular expressions indicating the presence or absence of AAAs. This annotation phase happened over three iterations. Keywords were meant to enhance the sensitivity and specificity of the detection algorithm. Once this list was finalized, we processed a held-out set of 24,521 imaging reports (all studies conducted in August 2021). We also processed these reports through the commercial vendor solution in a head-to-head performance comparison. We then selected 48 studies for manual review by three expert reviewers: one physician and two nurses (16 distinct studies each). Final performance statistics were generated on the basis of these 48 studies.

Results and Discussion – Table 1 compares the performance of our *sentenCy*-based solution to the commercial solution which had been in use at Geisinger for several years.

Table 1: Performance statistics of the *sentenCy*-based solution vs. Commercial solution to detect abdominal aortic aneurysm.

Metric	*sentenCy*-based Solution	Commercial Solution
Precision	83.3%	70.8%
Recall	69.0%	58.6%
F-1 score	0.755	0.642
Total cases flagged	632	536

Using the 48 human-reviewed cases as the gold standard for the presence/absence of AAA, our homegrown solution outperforms the commercial solution across three different measures of accuracy. Although neither algorithm achieves performance comparable to state-of-the-art for AAA identification[7], it should be noted this level of performance was achieved after only three rounds of iterative additions to our ruleset before the gold-standard dataset was available, indicating the scalability of the approach. With additional iterations, we expect to achieve performance comparable to state-of-the-art.

Conclusion – We developed an open-source library, *sentenCy*, to enable sentence-level named-entity-recognition and extend the NER capabilities of the biomedical NLP library *scispaCy. sentenCy* has an easy-to-use interface for adding customized keywords and expressions for disease identification. We found that our solution outperformed a commercial algorithm for AAA identification that was used at Geisinger for several years. In the spirit of open and reproducible science, we have made our extension available for public use so other researchers and health systems may deploy and extend it.

References

1. Mohit B. (2014) Named Entity Recognition. In: Zitouni I. (eds) Natural Language Processing of Semitic Languages. Theory and Applications of Natural Language Processing. Springer, Berlin, Heidelberg.
2. Kocaman V., Talby D. (2021) Biomedical Named Entity Recognition at Scale. In: Del Bimbo A. et al. (eds) Pattern Recognition. ICPR International Workshops and Challenges. ICPR 2021. Lecture Notes in Computer Science, vol 12661. Springer, Cham.
3. Neumann M, King D, Beltagy I, Ammar W. ScispaCy: fast and robust models for biomedical natural language processing. arXiv preprint arXiv:1902.07669. 2019 Feb 20.
4. Eyre H, Chapman AB, Peterson KS, Shi J, Alba PR, Jones MM, Box TL, DuVall SL, Patterson OV. Launching into clinical space with medspaCy: a new clinical text processing toolkit in Python. arXiv preprint arXiv:2106.07799. 2021 Jun 14.
5. Rule-based matching · Spacy Usage Documentation [Internet]. Rule-based matching. [cited 2022Mar9]. Available from: https://spacy.io/usage/rule-based-matching#matcher
6. Overview · spacy universe [Internet]. Overview. [cited 2022Mar9]. Available from: https://spacy.io/universe/
7. Sohn S, Ye Z, Liu H, Chute CG, Kullo IJ. Identifying abdominal aortic aneurysm cases and controls using natural language processing of radiology reports. AMIA summits on translational science proceedings. 2013;2013:249.