MEDINFO 2023 — The Future Is Accessible
J. Bichel-Findlay et al. (Eds.)

doi:10.3233/SHTI231039

Relation Detection to Identify Stroke Assertions from Clinical Notes Using Natural Language Processing

Audrey YANG[a], Sam KAMIEN[a], Anahita DAVOUDI[b], Sy HWANG[a], Meet GANDHI[a], Ryan URBANOWICZ[c] and Danielle MOWERY[a,1]
[a] *University of Pennsylvania, Philadelphia, PA, USA*
[b] *VNS Health, New York, NY, USA*
[c] *Cedars-Sinai Medical Center, Los Angeles, CA, USA*
ORCiD ID: Audrey Yang https://orcid.org/0000-0003-3643-708X, Sam Kamien https://orcid.org/0000-0001-6434-4046, Anahita Davoudi https://orcid.org/0000-0003-4345-3889, Sy Hwang https://orcid.org/0000-0003-3851-9521, Meet Gandhi https://orcid.org/0000-0001-7064-2023, Ryan Urbanowicz https://orcid.org/0000-0002-0487-5555, Danielle Mowery https://orcid.org/0000-0003-3802-4457

Abstract. According to the World Stroke Organization, 12.2 million people worldwide will have their first stroke this year almost half of which will die as a result. Natural Language Processing (NLP) may improve stroke phenotyping; however, existing rule-based classifiers are rigid, resulting in inadequate performance. We report findings from a pilot study using NLP to improve relation detection for stroke assertion detection to support research studies and healthcare operations.

Keywords. Natural language processing, machine learning, electronic health records

1. Introduction

The COVID-19 virus has been linked to increased risk of ischemic and hemorrhagic strokes [1]. Current efforts to characterize and study thrombotic events are limited by inaccurate phenotyping using electronic health record data due to a lack of specificity in hospital billing codes [2]. Natural Language Processing (NLP) may improve stroke phenotyping; however, existing rule-based classifiers produce promising but inadequate recall and precision. A more accurate stroke classifier could facilitate large-scale studies of strokes and support precision therapeutics and prophylactic administration of anticoagulants for preventing and treating strokes. Our objectives are two-fold: 1) conduct an annotation study to identify target-modifier pairs for encoding stroke subtypes, and 2) train and test an NLP system that determines target-modifier relations to encode stroke subtypes, their affected anatomy, and their statuses from clinical notes.

[1]Corresponding Author: Danielle Mowery, email: dlmowery@pennmedicine.upenn.edu.

2. Methods

For this Institute Review Board-approved pilot study, we queried patients with hospital billing codes associated with ischemic and hemorrhagic strokes codes: 434, 434.01, 434.11, 434.91 and their clinical notes from the open-source, de-identified MIMIC-III dataset. We conducted this study in three stages: *stroke classification*, *relation annotation*, and *relation detection*.

We aimed to develop a rule-based algorithm for identifying and subclassifying stroke by subtypes of ischemic and hemorrhagic as well as encode the anatomical location affected. First, we applied a linguistic approach to encode clinical information from radiology and discharge summaries necessary for developing this system. We adapted the pyConText algorithm to encode targets (*thrombosis* for ischemic strokes and *bleeds* for hemorrhages) and modifiers (*uncertainty*, *experiencer*, *anatomy*, *historicity*, *hypothetical*, etc.) which are used to describe diagnostic statements about the presence, absence, or uncertainty about whether a patient experienced a stroke [3].[1] We seeded our semantic modifiers (*anatomy*: venous, arterial, cardiovascular, neurovascular, peripheral, pulmonary) from the Consortium for Clinical Characterization of COVID-19 by EHR (4CE) Acute Kidney Injury Working Group thrombotic knowledge base[2] and queried the Unified Medical Language System (UMLS) using scispaCy to generate synonyms for each anatomical location.[3] We had developed a corpus for developing and validating the pyConText algorithm to assert whether a patient has had an ischemic or hemorrhagic stroke and its assertion status (**negated, affirmed, uncertain, historical, hypothetical, familial**). We sampled 80 patients: 40 ischemic and 40 hemorrhagic stroke patients. We created 4 batches of 10 patients each (n=20 patients total). For each batch, co-authors AD and DM annotated the sentences that were flagged by the stroke subtype classifier. The classifier performance for asserting stroke status achieved an average Cohen's kappa of 63% for hemorrhages and 37% for ischemic stroke. We determined that performance for asserting stroke subtype status may improve by integrating a more sophisticated algorithm to accurately determine whether a target (*bleed*) is associated with a modifier (*anatomy*).

For target-modifier relation annotation, we sampled 100 ischemic and hemorrhagic stroke patients and created 5 batches of 10 patients each stroke type for each batch. For each clinical note, we identified sections of "observation and plan", "chief complaint", or "hospital course" and sentences using medSpaCy [4].[4] Each batch was written out to a .csv file with each row containing a distinct target-modifier pair with the associated sentence. Then, each target-modifier relation pair was reviewed and adjudicated by co-authors DM and SK. For each batch, we report the kappa (*k*), observed (*Ao*), positive (*pos*), and negative (*neg*) agreements.

We conducted a relation detection study in which the target-modifier pair relation detection was treated as a binary classification problem. Given target-modifier pairs, we encoded features described in Table 1 to train the machine learning prediction models.

Table 1. Relation detection features with description and examples.

Feature	Explanation	Example
Target type	Type of the recognized target, either bleeding or thrombosis	Bleeding: "aneurysm"; Thrombosis: "embolism"
Modifier type	One-hot-encoded set of 12 features that corresponded to our modifiers. We had status modifiers of negated, affirmed, uncertain, hypothetical, historical, familial; and location modifiers of venous, arterial, cardiovascular, neurovascular, peripheral arterial, pulmonary.	Affirmed existence: "Stable appearance"; Arterial anatomy: "aortic"; Historical: "prior"
Number of tokens between	The number of tokens in between (and right inclusive of) the target and modifier	Sentence: "Then using a **brain** retractor, we looked around the cavity of the clot and I didn't see any acute **hemorrhage**." Target: "hemorrhage", modifier: "brain". Tokens between: 16.
Number of recognized tokens between	The number of tokens in between (and not including) the target and modifier that are recognized as a modifier	Sentence: "Then using a **brain** retractor, we looked around the cavity of the clot and I didn't see any acute **hemorrhage**." Target: "hemorrhage", modifier: "brain". Recognized tokens between: 5.
Presence of "and" between	Whether "and" occurs in between the target and modifier. The presence of "and" indicates that the scope is being expanded.	Sentence: "Then using a brain retractor, we looked around the cavity of the clot **and** I didn't see any acute hemorrhage."
Presence of "but" between	Whether "but" occurs in between the target and modifier. The presence of "but" indicates a limiting scope.	Sentence: "Mesenteric ischemia cannot be ruled out on the basis of this film **but** there are no positive radiographic findings to support this diagnosis."
Syntactic dependency	Whether the target is a syntactic dependency of the modifier, the modifier is a syntactic dependency of the target, or if they have no relationship	Sentence: "Otherwise, **no signs of hemorrhage**." Target: "hemorrhage", modifier: "no signs of". Modifier dependent on target.
Cosine similarity	Leveraging a package by [6], the cosine similarity between the target and modifier	Sentence: "2) Extensive **subarachnoid hemorrhage**." Target: "hemorrhage", modifier: "subarachnoid". Cosine similarity: 0.4031.

We leveraged an auto machine learning pipeline called STREAMLINE [5].[5] We combined the first four batches as a development set; the fifth batch as a validation set. We used five-fold cross-validation on a stratified development set and selected the most informative features based on the union of features with significant mutual information and multiSURF scores. We trained models using supervised learning algorithms as well as rule learners e.g., Extended Supervised Tracking and Classifying System (ExSTraCS). Each model was trained with a hyperparameter sweep tuned with Optuna before we applied the optimized models on the validation set. About 75% of target-modifier relation pairs were associated in both development and validation sets.

3. Results

We annotated five batches of target-modifier relations from the pyConText algorithm in Table 2. At batch 4, we re-established the high agreement observed across all measures.

Table 2. Agreement over 5 batches of target-modifier relation pairs.

Target-Modifier Relations	*Ao*	*k*	*pos*	*neg*
batch 0 (n = 43)	0.86	0.67	0.90	0.77
batch 1 (n = 194)	0.91	0.76	0.94	0.83
batch 2 (n = 204)	0.94	0.85	0.96	0.88
batch 3 (n = 293)	0.86	0.62	0.91	0.71
batch 4 (n = 324)	0.94	0.83	0.96	0.87

Figure 1 (left) shows a graph of average AUC; Figure 1 (right) shows a graph of precision/recall curves (PRCs) for each algorithm.

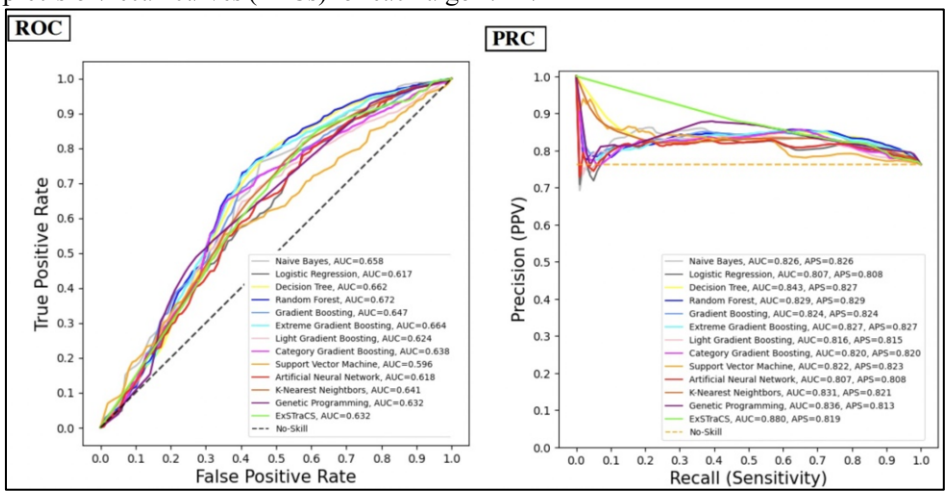

Figure 1. Performance curves for each classifier from the validation set.

In Figure 2, we visualized the aggregated composite feature importance bar plots from each algorithm weighted by the balanced accuracy of each prediction model such that the higher the accuracy achieved, the more it contributes to feature importance.

4. Discussion

We developed a rule-based stroke assertion classifier for discerning whether a patient experienced an acute ischemic or hemorrhagic stroke based on their radiology and discharge summaries. We observed that 25% of the relations detected by the prototype were incorrect. Supervised learners may inform whether a target-modifier pair should be associated. ExSTraCS's rules based on lexical, syntactic, semantic, and cosine similarity features could be integrated into the pyConText algorithm to improve stroke assertions.

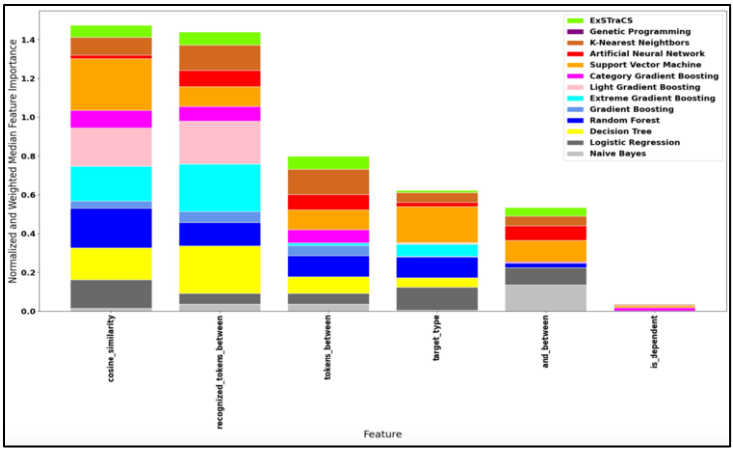

Figure 2. Graphs of feature importance by classifier from the development set.

5. Conclusions

Supervised learners may improve target-modifier relation detection with rich features.

References

[1] Le TT, Gutiérrez-Sacristán A, Son J, Hong C, South AM, Beaulieu-Jones BK, Loh NHW, Luo Y, Morris M, Ngiam KY, Patel LP, Samayamuthu MJ, Schriver E, Tan ALM, Moore J, Cai T, Omenn GS, Avillach P, Kohane IS; Consortium for Clinical Characterization of COVID-19 by EHR (4CE); Visweswaran S, Mowery DL, Xia Z. Multinational characterization of neurological phenotypes in patients hospitalized with COVID-19. Sci Rep. 2021 Oct;11(1):20238, doi: 10.1038/s41598-021-99481-9.

[2] Woodfield R, Grant I; UK Biobank Stroke Outcomes Group; UK Biobank Follow-Up and Outcomes Working Group; Sudlow CL. Accuracy of Electronic Health Record Data for Identifying Stroke Cases in Large-Scale Epidemiological Studies: A Systematic Review from the UK Biobank Stroke Outcomes Group. PLoS One. 2015 Oct;10(10):e0140533, doi: 10.1371/journal.pone.0140533.

[3] Chapman BE, Lee S, Kang HP, Chapman WW. Document-level classification of CT pulmonary angiography reports based on an extension of the ConText algorithm. J Biomed Inform. 2011 Oct;44(5):728-37, doi: 10.1016/j.jbi.2011.03.011.

[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. AMIA Annu Symp Proc. 2022 Feb 21;2021:438-47.

[5] Urbanowicz R, Zhang R, Cui Y, Suri P. STREAMLINE: A Simple, Transparent, End-To-End Automated Machine Learning Pipeline Facilitating Data Analysis and Algorithm Comparison. In Genetic Programming Theory and Practice XIX 2023 Mar 12 (pp. 201-231). Singapore: Springer Nature.

Endnotes

[1] https://github.com/dmowery/pyConText-carotid-stenosis-detection
[2] https://github.com/covidclinical/Phase2.1AKIRPackage/tree/master/FourCePhase2.1AKI/data
[3] https://github.com/allenai/scispacy
[4] https://github.com/medspacy/medspacy
[5] https://github.com/UrbsLab/STREAMLINE