Title: The SMART Text2FHIR Pipeline

Authors:

Timothy A. Miller, PhD, Andrew J. McMurry, PhD, James Jones, Daniel Gottlieb, MPA, Kenneth D. Mandl, MD, MPH

Corresponding author:

Kenneth D. Mandl, MD, MPH
Computational Health Informatics Program, Boston Children's Hospital
Department of Pediatrics, Department of Biomedical Informatics, Harvard Medical School
401 Park Drive, Landmark Center, 5th Floor East, Boston, MA 02215, U.S.A.
Kenneth_Mandl@harvard.edu
617-355-4145

Full name, department, institution, city, and country of other co-authors:

Timothy A. Miller, PhD
Computational Health Informatics Program, Boston Children's Hospital
Department of Pediatrics, Harvard Medical School
Boston, MA, USA

Andrew J. McMurry, PhD
Computational Health Informatics Program, Boston Children's Hospital
Department of Pediatrics, Harvard Medical School
Boston, MA, USA

James Jones
Computational Health Informatics Program, Boston Children's Hospital
Boston, MA, USA

Daniel Gottlieb, MPA
Computational Health Informatics Program, Boston Children's Hospital
Boston, MA, USA

Keywords: Natural language processing; Interoperability; electronic health records

Word count: 1832

Abstract:

Objective: To implement an open source, free, and easily deployable high throughput natural language processing module to extract concepts from clinician notes and map them to Fast Healthcare Interoperability Resources (FHIR).

Materials and Methods: Using a popular open-source NLP tool (Apache cTAKES), we create FHIR resources that use modifier extensions to represent negation and NLP sourcing, and another extension to represent provenance of extracted concepts.

Results: The SMART Text2FHIR Pipeline is an open-source tool, released through standard package managers, and publicly available container images that implement the mappings, enabling ready conversion of clinical text to FHIR.

Discussion: With the increased data liquidity because of new interoperability regulations, NLP processes that can output FHIR can enable a common language for transporting structured and unstructured data. This framework can be valuable for critical public health or clinical research use cases.

Conclusion: Future work should include mapping more categories of NLP-extracted information into FHIR resources and mappings from additional open-source NLP tools.

Introduction and background

In the United States, billions of notes written by clinicians become a part of the electronic health record (EHR) each year. Notes contain a large percentage of the useful information in EHRs, when subjected to natural language processing (NLP).(1–7) While open source tools specialized to clinical NLP are in widespread use,(8–15) the variability in hospital data systems meant that each new NLP implementation has traditionally required substantial customization.

However, advances in both technology and regulation now make access to and processing of EHR-based text both turnkey and scalable. FHIR (Fast Healthcare Interoperability Resources) has become a de facto standard for presentation of data from EHR systems to external applications.

Further, interoperability provisions in the 21st Century Cures Act Rule require EHR vendors to support the SMART on FHIR(16,17) and SMART/HL7 Bulk FHIR Access (18) application programming interfaces (APIs). These interfaces allow standardized access to clinical text, along with structured information specified in the US Core for Data Interoperability (USCDI),(19) including laboratory testing, diagnoses, and medications.

This new data liquidity creates an unprecedented opportunity to apply NLP methods, which can convert unstructured text into structured information, for example, extracting from doctor's notes, references to diagnoses, signs and symptoms, procedures, and the relations among these elements.

Objective

We sought to implement an open source, free, high throughput, and easily deployable NLP module to extract concepts from clinician notes and map them to FHIR format. The result

would be text-based information intermingled and used alongside structured information, allowing downstream users (e.g., clinical researchers, quality analysts, regulators, etc.) to manipulate this data source for a variety of use cases.

Materials and Methods

Figure 1 shows an overview of the "SMART Text2FHIR" system. The core engine for "SMART Text2FHIR" is Apache cTAKES (clinical Text Analysis and Knowledge Extraction System),(8) an open-source clinical NLP software library for extracting information from unstructured text in EHRs. cTAKES is a highly configurable package with dozens of pipeline components that can be arranged by library users.

Perhaps the most important cTAKES component is the dictionary lookup module, which maps strings in input texts to normalized codes in source terminologies, as well as Concept Unique Identifiers (CUIs) in the Unified Medical Language System (UMLS) for interoperability. The default cTAKES pipeline uses SNOMED-CT and RxNORM source dictionaries, and we make use of that pipeline for this work. We use a fast rule-based model, the cTAKES Context(20) implementation, for negation detection. This module labels each mapped concept with a *polarity* attribute, which is one by default and minus one if the concept is negated in the text (e.g., *no fever*).

This library uses standard FHIR resource types and annotates them with NLP-specific data and metadata. Extensibility is a core feature of FHIR; specifically, extensions provide a standard way for users to address specific use cases by adding elements to the core resource structures. Every FHIR resource and data element can be extended, and each extension contains

a URI-based identifier and additional data (which may be comprised of other, nested extensions). Some FHIR elements can also be extended with "modifier" extensions, which alter the interpretation of the resource or element, and may not be ignored by applications or users of the data. SMART Text2FHIR uses both FHIR regular and modifier *extensions*.

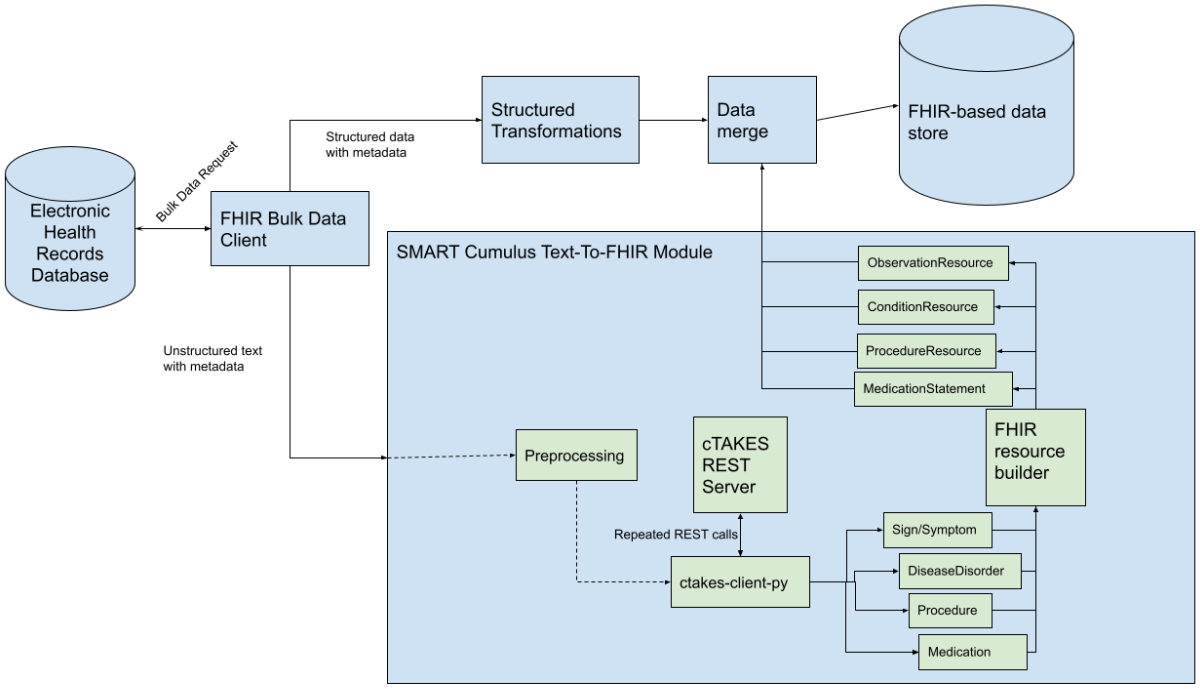


Figure 1: A diagram of the information flow that the SMART Text2FHIR module sits within.

Mappings were developed that convert outputs from broad semantic groups in the UMLS (and the cTAKES "type system" which closely mirrors it) to FHIR resources. Procedures are mapped to the Procedure Resource, Signs/Symptoms are mapped to Observation Resources,

Diseases/Disorders are mapped to Condition Resources, and Medications are mapped to MedicationStatement Resources.

Next, extensions to track provenance of NLP outputs were implemented – one important attribute of our implementation is that the automatic and less-certain nature of NLP-extracted resources is highlighted, and that provenance information is preserved so that downstream users can manually validate or even replicate NLP system behavior if required for their use cases.

We track the NLP system version used to produce the FHIR resource with an "nlp-source" modifier extension that includes "algorithm" and "version" values. The algorithm field is a human-readable description of the algorithm, and the version field is a URL pointing to the location of the NLP software. Representing this information as a modifier extension rather than a regular extension is important, as this helps indicate to downstream users that these concepts (e.g., codes representing diseases) do not have the same epistemological status as mappings from coded data, and need to be treated differently by any systems that ingest them.

To track the position of each extracted NLP concept in its original document, we added a *derivation-reference* extension which has been proposed for inclusion in the next release of FHIR; it has integer-typed *offset* and *length* fields to represent character offsets of the concept in the source document, which, in combination with the *reference* field that points to the original document, would allow a downstream user to find the original text span corresponding to the extracted concept.

Finally, a modifier extension to represent negation of concepts called "nlp-polarity" was also implemented. Negation is common in EHR text, as it is often important to, for example, rule out a competing diagnosis or explicitly assert the absence of a symptom. This was implemented

as a modifier extension since it fundamentally changes the meaning of the concept and contains a boolean value withthe value *False* for negated concepts and *True* otherwise.

Results

SMART Text2FHIR is an extension to Apache cTAKES that performs NLP-to-FHIR mappings with easily obtainable open-source artifacts. cTAKES has an existing REST server implementation, which returns a json-formatted output string with an ad hoc data model. Our new modifications include creating a GitHub repository for building a Docker image of the cTAKES REST server, a publicly available pre-built Docker image from that repository, and an easily installable python client that implements the mapping from the cTAKES REST output string to our proposed FHIR representations. We use these components as follows:

Publicly available Docker image

This publicly available Docker image[1] is built from the ctakes-covid-container repository on Github.[2] This sets up the cTAKES REST API with a customized NLP pipeline enriched to be able to find COVID-19-related signs and symptoms. The relevant components of this pipeline are a dictionary lookup that finds mentions of terms in SNOMED-CT, RxNORM, and a customized COVID-19 dictionary and maps them to UMLS CUIs (for SNOMED-CT and RxNORM) or custom codes. The cTAKES pre-built dictionaries require UMLS authentication at container start-up, so this image accepts a UMLS API key as an environment variable in its run command.

[1] https://hub.docker.com/r/smartonfhir/ctakes-covid
[2] https://github.com/Machine-Learning-for-Medical-Language/ctakes-covid-container

cTAKES REST client

Once the Docker container running the cTAKES REST server is started, it can be called by a python library that we have developed called ctakes-client-py.[3] This client library can be installed with the pip (package installer for python) tool. The client provides functions to simplify REST calls to the server and organizes the returned JSON data by semantic type. The text2fhir module of the client provides methods for converting each semantic type into the appropriate resource. This module also provides a single simplified entry point that takes the unstructured FHIR in and returns a collection of FHIR resources that are ready for downstream use.

Example outputs

Figure 2 shows a snippet of an example input text, and the resulting FHIR output. The full example can be seen in the online supplement.

[3] https://github.com/Machine-Learning-for-Medical-Language/ctakes-client-py

Reason for Visit: Patient complains of fever, cough.

HPI: Patient is an 8-year-old female presenting today for worsening fever and cough. Patient denies sore throat, headache, fatigue.

PMH: Asthma since age 7

PSH: Tonsillectomy and Adenoidectomy, PDA closure, tympanostomy tube placement.

FHI: reviewed and non-contributory

SH: Lives at home with parents and older sister

Immunizations: Up to date

ROS:
Pertinent positives and negatives noted above in HPI. All other systems of a 10 system review are negative.

Home Medications:
Flovent, albuterol prn

Medications Prescribed This Visit:
acetaminophen (acetaminophen 160 mg/5 mL oral liquid), 131.2 mg = 4.1 mL, PO, Q4hr, PRN


```
{
    "id": "05cb5888-2c9a-41ad-b877-eaa48c018657",
    "extension": [ {
        "extension": [ {
            "url": "reference",
            "valueReference": {
                "reference": "DocumentReference/44ce9f27-f6ca-44f2-a79b-c88ba8abfe8b"
            }
          }, {
            "url": "offset",
            "valueInteger": 39
          }, {
            "url": "length",
            "valueInteger": 5
          } ],
        "url": "http://hl7.org/fhir/StructureDefinition/derivation-reference"
      } ],
    "modifierExtension": [ {
        "extension": [ {
            "url": "algorithm",
            "valueString": "ctakesclient"
          }, {
            "url": "version",
            "valueString":
"https://github.com/Machine-Learning-for-Medical-Language/ctakes-client-py/releases/tag/v2.0.0"
          } ],
        "url": "http://fhir-registry.smarthealthit.org/StructureDefinition/nlp-source"
      }, {
        "url": "http://fhir-registry.smarthealthit.org/StructureDefinition/nlp-polarity",
        "valueBoolean": true
      } ],
    "code": {
      "coding": [ {
          "code": "386661006",
          "system": "http://snomed.info/sct"
        }, {
          "code": "C0015967",
          "system": "http://terminology.hl7.org/CodeSystem/umls"
        }, {
          "code": "50177009",
          "system": "http://snomed.info/sct"
        }, {
          "code": "C0015967",
          "system": "http://terminology.hl7.org/CodeSystem/umls"
        }
      ],
      "text": "fever"
    },
    "encounter": {
        "reference": "Encounter/8aa2fe54-bfe1-4e3d-a645-bd9f4b5fec78"
    },
    "status": "preliminary",
    "subject": {
        "reference": "Patient/d1129431-29c9-488e-97a4-00c8c5a0a4b1"
    },
    "resourceType": "Observation"
  },
```


Figure 2: An example snippet of a note (left), with multiple clinical concepts. The concept "fever" is converted to a FHIR resource represented by the JSON text on the right.

Discussion

We anticipate that SMART Text2FHIR can address many use cases that require aligning clinical text data alongside structured data. For example, in a multisite network, multiple collaborating research sites could run the same NLP to FHIR process, either pooling their data in a shared HIPAA-compliant cloud instance or as separate on-premises stores. The NLP output from all sites would then be in the same format from the start. This could, for example, serve as a foundation for sharing classifiers that operate over clinical text (21) at multiple centers. Another

example use case is outside researchers who want to run de-identified queries against EHR data (e.g., regulators or public health officials): however they formulate the query, the mapping to the internal FHIR representation can be re-used or shared across sites.

There are a few known open-source attempts to map from NLP outputs to FHIR. Apache cTAKES has a built-in ctakes-fhir module that can convert many elements of the cTAKES data model into FHIR resources. However, that implementation makes use only of the Basic Resource type, which offers great flexibility in adapting to new data types, but is more general than might be desired for many data types that already have more appropriate mappings to FHIR resources. This will make it difficult for users to leverage existing FHIR knowledge when reviewing NLP derived data, as well as add complexity when working with structured and unstructured data in a single study.

The NLP2FHIR system(22) contains several logical mappings from multiple NLP systems into FHIR resources, including mapping disease/disorder mappings to ConditionResource, mapping signs and symptoms to ObservationResource, and mapping procedures to ProcedureResource. However, one major missing element of this effort was a lack of provenance data – that is, the ability to work backwards from FHIR resources to the original source to validate NLP outputs if necessary. It also requires manual installation of several packages it depends on, making setup somewhat tricky. Our solution includes package-managed software installs that remove the difficulty of manually managing software versions in dependent libraries.

There are also existing commercial offerings from Microsoft (23) and Amazon Web Services (24), but our goal is to produce an open source alternative that can run economically at scale and be maintained by the research and FHIR communities.

Conclusion

The open-source software implementation is available on GitHub. We encourage readers to join the community developing these mappings to ensure future mappings are done in a logical and transparent manner.

Currently, negation (i.e., polarity) is the only status represented, but there are NLP methods for detecting, for example, uncertain/hedged hypothetical, and non-patient-related concepts that could easily be integrated using similar mechanisms.

While we focused on one widely used clinical NLP tool, cTAKES, extending to other tools requires minimal effort: First, a Docker wrapper that sets up a REST endpoint (we have previously implemented such a library for Medspacy[4]), and then an output mapping step. This could take the form of a conversion into the cTAKES ad hoc data model as a postprocess in the REST server, or the forking of cTAKES server-specific functions in the ctakes-client-py library.

Future work must include mapping more categories of NLP-extracted information into FHIR resources, for example, temporal information. Clinical notes often contain narrative information describing the disease course, with mentions of past and future (potential) events, and representing these important temporal details is crucial for many use cases.

Funding acknowledgement

Work reported in this publication was supported by three contracts 90AX0022, 90AX0019, 90AX0031, and 90C30007 from the Office of the National Coordinator of Health Information Technology; the Centers for Disease Control and Prevention of the U.S. Department of Health and Human Services (HHS) as part of a financial assistance award (The contents are those of the

[4] https://github.com/Machine-Learning-for-Medical-Language/medspacy-covid-container

author(s) and do not necessarily represent the official views of, nor an endorsement, by CDC/HHS, or the U.S. Government). A cooperative agreement from the National Center for Advancing Translational Sciences U01TR002623; Grants from the National Library of Medicine (R01LM012973, R01LM012918); and the Boston Children's Hospital PrecisionLink Initiative. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding sources.

Bibliography

1. Lin C, Karlson EW, Dligach D, Ramirez MP, Miller T a., Mo H, et al. Automatic identification of methotrexate-induced liver toxicity in patients with rheumatoid arthritis from the electronic medical record. J Am Med Inform Assoc. 2014;23–30.

2. MIMIC-III Benchmarks [Internet]. YerevaNN; 2022 [cited 2022 Mar 2]. Available from: https://github.com/YerevaNN/mimic3-benchmarks

3. Afshar M, Phillips A, Karnik N, Mueller J, To D, Gonzalez R, et al. Natural language processing and machine learning to identify alcohol misuse from the electronic health record in trauma patients: development and internal validation. J Am Med Inform Assoc. 2019;26(3):254–61.

4. Geva A, Gronsbell JL, Cai T, Cai T, Murphy SN, Lyons JC, et al. A Computable Phenotype Improves Cohort Ascertainment in a Pediatric Pulmonary Hypertension Registry. J Pediatr. 2017;188:224-231.e5.

5. Castro VM, Dligach D, Finan S, Yu S, Can A, Abd-El-Barr M, et al. Large-scale identification of patients with cerebral aneurysms using natural language processing. Neurology. 2017 Jan 10;88(2):164–8.

6. Liao KP, Cai T, Gainer V, Goryachev S, Zeng-Treitler Q, Raychaudhuri S, et al. Electronic medical records for discovery research in rheumatoid arthritis. Arthritis Care Res. 2010;62:1120–7.

7. Malden DE, Tartof SY, Ackerson BK, Hong V, Skarbinski J, Yau V, et al. Natural Language Processing for Improved Characterization of COVID-19 Symptoms: Observational Study of 350,000 Patients in a Large Integrated Health Care System. JMIR Public Health Surveill. 2022 Dec 30;8(12):e41529.

8. Savova GK, Masanz JJ, Ogren PV, Zheng J, Sohn S, Kipper-Schuler KC, et al. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. J Am Med Inform Assoc JAMIA. 2010 Oct;17(5):507–13.

9. Soysal E, Wang J, Jiang M, Wu Y, Pakhomov S, Liu H, et al. CLAMP – a toolkit for efficiently building customized clinical natural language processing pipelines. J Am Med Inform Assoc. 2018 Mar 1;25(3):331–6.

10. Eyre H, Chapman AB, Peterson KS, Shi J, Alba PR, Jones MM, et al. Launching into clinical space with medspaCy: a new clinical text processing toolkit in Python. In: AMIA Annual Symposium Proceedings. American Medical Informatics Association; 2021. p. 438.

11. Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Proc AMIA Symp. 2001;17–21.

12. Demner-Fushman D, Rogers WJ, Aronson AR. MetaMap Lite: an evaluation of a new Java implementation of MetaMap. J Am Med Inform Assoc. 2017;24(4):841–4.

13. MedXN: an open source medication extraction and normalization tool for clinical text | Journal of the American Medical Informatics Association | Oxford Academic [Internet]. [cited 2023 Feb 14]. Available from: https://academic.oup.com/jamia/article/21/5/858/760598

14. Miller TA, Avillach P, Mandl KD. Experiences implementing scalable, containerized, cloud-based NLP for extracting biobank participant phenotypes at scale. JAMIA Open. 2020 Jul;3(2):185–9.

15. Afshar M, Dligach D, Sharma B, Cai X, Boyda J, Birch S, et al. Development and application of a high throughput natural language processing architecture to convert all clinical documents in a clinical data warehouse into standardized medical vocabularies. J Am Med Inform Assoc. 2019 Nov 1;26(11):1364–9.

16. Mandel JC, Kreda DA, Mandl KD, Kohane IS, Ramoni RB. SMART on FHIR: a standards-based, interoperable apps platform for electronic health records. J Am Med Inform Assoc. 2016 Sep 1;23(5):899–908.

17. Mandl KD, Mandel JC, Murphy SN, Bernstam EV, Ramoni RL, Kreda DA, et al. The SMART Platform: early experience enabling substitutable applications for electronic health records. J Am Med Inform Assoc. 2012;19(4):597–603.

18. Mandl KD, Gottlieb D, Mandel JC, Ignatov V, Sayeed R, Grieve G, et al. Push button population health: the SMART/HL7 FHIR bulk data access application programming interface. NPJ Digit Med. 2020;3(1):151.

19. United States Core Data for Interoperability (USCDI) | Interoperability Standards Advisory (ISA) [Internet]. [cited 2023 Feb 14]. Available from: https://www.healthit.gov/isa/united-states-core-data-interoperability-uscdi

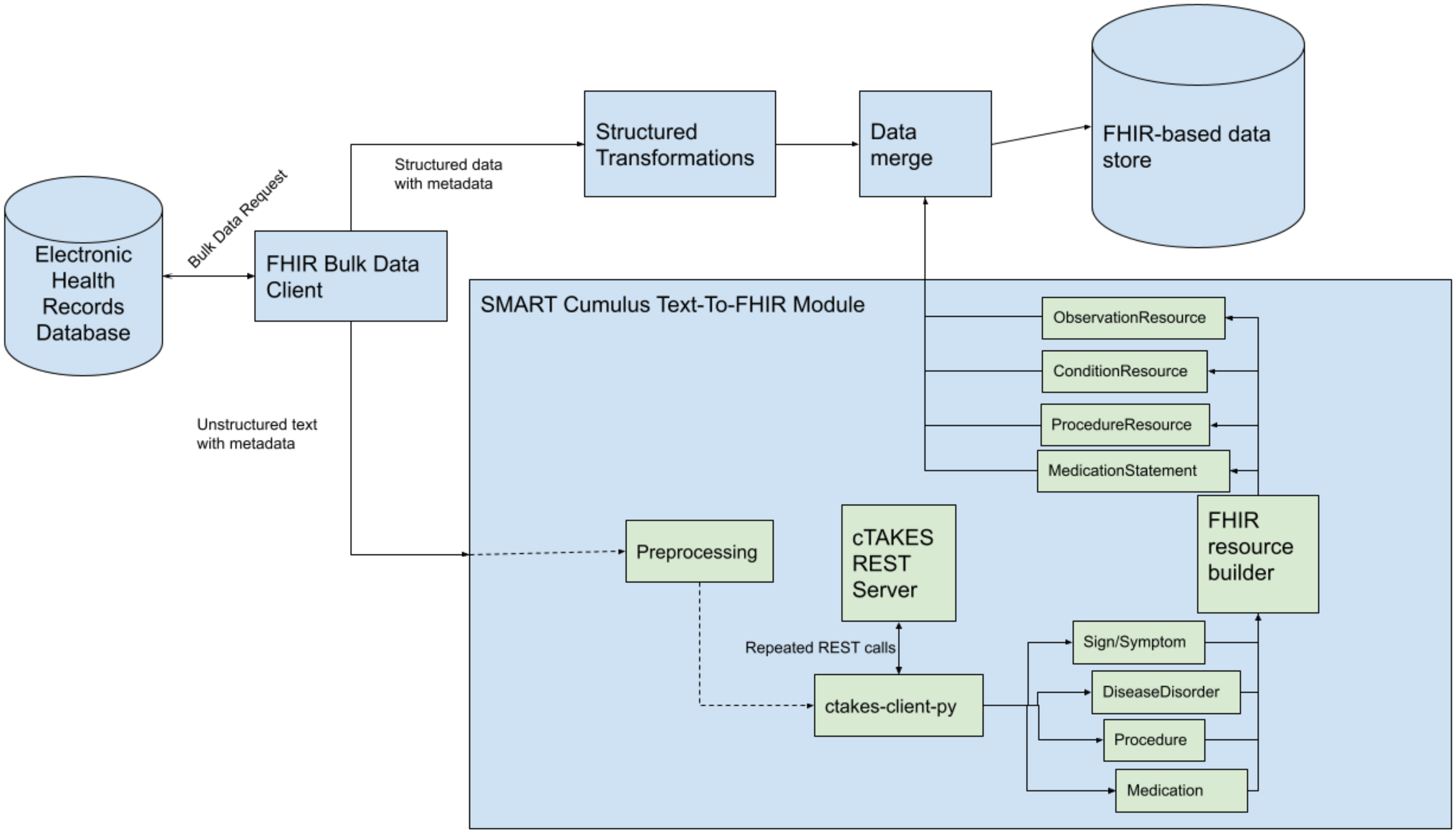
20. Harkema H, Dowling JN, Thornblade T, Chapman WW. ConText: an algorithm for determining negation, experiencer, and temporal status from clinical reports. J Biomed Inform. 2009;42(5):839–51.

21. Wang L, Zipursky A, Geva A, McMurry AJ, Mandl KD, Miller TA. A computable phenotype for patients with SARS-CoV2 testing that occurred outside the hospital [Internet]. medRxiv; 2023 [cited 2023 Feb 14]. p. 2023.01.19.23284738. Available from: https://www.medrxiv.org/content/10.1101/2023.01.19.23284738v1

22. Hong N, Wen A, Shen F, Sohn S, Wang C, Liu H, et al. Developing a scalable FHIR-based clinical data normalization pipeline for standardizing and integrating unstructured and structured electronic health record data. JAMIA Open. 2019 Dec 1;2(4):570–9.

23. From free text to FHIR: Text Analytics for health launches new feature to boost interoperability [Internet]. TECHCOMMUNITY.MICROSOFT.COM. 2022 [cited 2022 Dec 15]. Available from: https://techcommunity.microsoft.com/t5/ai-cognitive-services-blog/from-free-text-to-fhir-text-analytics-for-health-launches-new/ba-p/3257066

24. Achieve Healthcare Interoperability by integrating Amazon Comprehend Medical with FHIR | AWS for Industries [Internet]. 2019 [cited 2022 Dec 15]. Available from: https://aws.amazon.com/blogs/industries/achieve-healthcare-interoperability-by-integrating-amazon-comprehend-medical-with-fhir/


Electronic Health Records Database
Bulk Data Request
FHIR Bulk Data Client
Structured data with metadata
Structured Transformations
Data merge
FHIR-based data store
Unstructured text with metadata
SMART Cumulus Text-To-FHIR Module
ObservationResource
ConditionResource
ProcedureResource
MedicationStatement
FHIR resource builder
Preprocessing
cTAKES REST Server
Repeated REST calls
ctakes-client-py
Sign/Symptom
DiseaseDisorder
Procedure
Medication

Reason for Visit: Patient complains of fever, cough.

HPI: Patient is an 8-year-old female presenting today for worsening fever and cough. Patient denies sore throat, headache, fatigue.

PMH: Asthma since age 7

PSH: Tonsillectomy and Adenoidectomy, PDA closure, tympanostomy tube placement.

FHI: reviewed and non-contributory

SH: Lives at home with parents and older sister

Immunizations: Up to date

ROS:
Pertinent positives and negatives noted above in HPI. All other systems of a 10 system review are negative.

Home Medications:
Flovent, albuterol prn

Medications Prescribed This Visit:
acetaminophen (acetaminophen 160 mg/5 mL oral liquid), 131.2 mg = 4.1 mL, PO, Q4hr, PRN


```
{
    "id": "05cb5888-2c9a-41ad-b877-eaa48c018657",
    "extension": [ {
        "extension": [ {
            "url": "reference",
            "valueReference": {
              "reference": "DocumentReference/44ce9f27-f6ca-44f2-a79b-c88ba8abfe8b"
            }
          }, {
            "url": "offset",
            "valueInteger": 39
          }, {
            "url": "length",
            "valueInteger": 5
          } ],
        "url": "http://hl7.org/fhir/StructureDefinition/derivation-reference"
      } ],
    "modifierExtension": [ {
        "extension": [ {
            "url": "algorithm",
            "valueString": "ctakesclient"
          }, {
            "url": "version",
            "valueString":
"https://github.com/Machine-Learning-for-Medical-Language/ctakes-client-py/releases/tag/v2.0.0"
          } ],
        "url": "http://fhir-registry.smarthealthit.org/StructureDefinition/nlp-source"
      }, {
        "url": "http://fhir-registry.smarthealthit.org/StructureDefinition/nlp-polarity",
        "valueBoolean": true
      } ],
    "code": {
      "coding": [ {
          "code": "386661006",
          "system": "http://snomed.info/sct"
        }, {
          "code": "C0015967",
          "system": "http://terminology.hl7.org/CodeSystem/umls"
        }, {
          "code": "50177009",
          "system": "http://snomed.info/sct"
        }, {
          "code": "C0015967",
          "system": "http://terminology.hl7.org/CodeSystem/umls"
        }
      ],
      "text": "fever"
    },
    "encounter": {
      "reference": "Encounter/8aa2fe54-bfe1-4e3d-a645-bd9f4b5fec78"
    },
    "status": "preliminary",
    "subject": {
      "reference": "Patient/d1129431-29c9-488e-97a4-00c8c5a0a4b1"
    },
    "resourceType": "Observation"
},
```