Phenotyping Electronic Health Records

Christopher Meaney

Biostatistician
Department of Family and Community Medicine
University of Toronto

October 11, 2020

Phenotyping Electronic Health Records

- Phenotype specifies observable, possibly time-varying state of a patient.
- Phenotypes derived from EHR can be outcomes, covariates, incl/excl criterion, etc.
- Synonyms: case finding algorithms, code set engineering, etc.

Table 1

Applications of electronic phenotyping across study types

Study type	Use cases
Cross-sectional	Epidemiological research
	Hospital administration/resource allocation
	Adherence to diagnostic/treatment guidelines
	Quality measurement
Association (case-control/cohort)	Genome-wide association studies
	Pharmacovigilance
	Identifying clinical risk factors and protective factors
	Clinical decision support
	Clinical effectiveness research
	Predictive modeling
Experimental	Clinical trial recruitment
	Pragmatic trials
	Adaptive/randomized, embedded, multifactorial, adaptive platform trials

Banda et al. (2018). Phenotpying: Rule Based vs. ML. Ann Rev Biomedical Data Sci.

Phenotyping Design/Evaluation

- Rule/algo identify case positive/negative patients in EHR.
- How to know if phenotype algorithm any good?

Cross-sectional Sampling

- (Large) random sample EHR.
- Expert annotated gold-standard labels.
- Development/evaluation of phenotype algorithm.
- Limitations: cost/time, rare conditions, etc.

Case-Control (Two-stage) Sampling

- Identify most case positive patients in EHR.
- Random sample case negative patients EHR.
- Development/evaluation of phenotype algorithm.
- Limitations: cost/time, re-calibration, biases, etc.

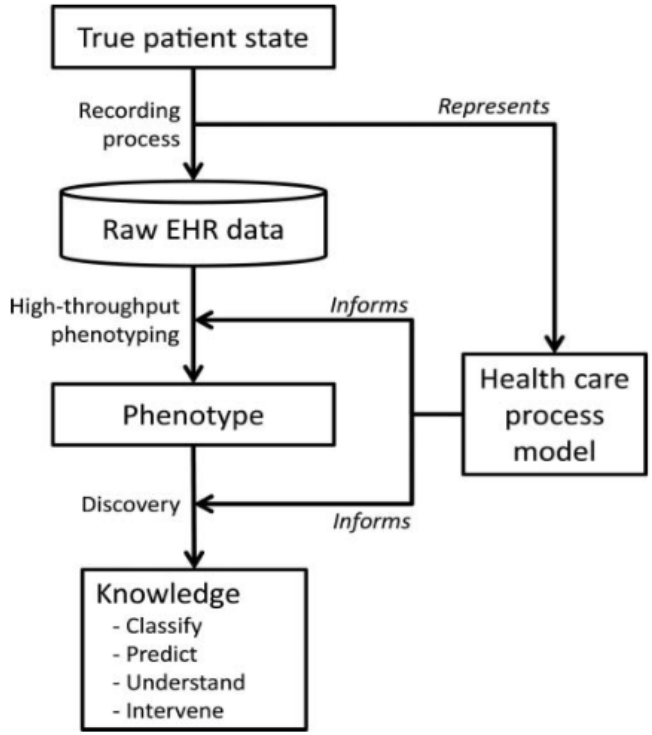


Figure 2 Phenotyping and discovery. The raw electronic health record (EHR) data are an indirect reflection of the true patient state due to the recording process. Attempts to create phenotypes and discover knowledge must account for the recording. The healthcare process model represents the salient features of the recording process and informs the phenotyping and discovery.

Hripcsak et al. (2012). Next Gen Phenotyping EHRs. JAMIA.

Annotation

- Two or more experts review and label charts for algorithm development.
- Major rate limiting step: timely and costly.
- Assess inter/intra-rater agreement (kappa, etc.).

Algorithm Development

- Rule based methods. Boolean logical queries. Expert derived, consensus timely.
- Supervised ML. Feature engineering timely. Overfit small train samples.
- Increasingly novel designs: attempt circumvent annotation/development/overfitting costs.
  - Active learning.
  - Learning with Noisy Labels (noise tolerant learning).
  - Semi-supervised learning.
  - Unsupervised learning.

Evaluating Phenotype Algorithms

- Held out test set evaluation? Test set random sample target population?
- Metrics: sensitivity, specificity, PPV, NPV, F1, etc.
- Generalizability? Geographic, temporal, etc.
- Portability, transparency, etc.

Figure 1. General model of automatic phenotyping algorithms.

Alzoubi et al. (2019). Auto Phenotyping EHRs: A Review. MDPI Electronics.

Trends in EHR Phenotyping Studies

- Rule based methods popular. Interpretable. Fast. Lack awareness language/context?
- Supervised ML gains popularity. Flexible, context aware. Black-box?
- Neural network language models (attention, transformers, BERT, etc.) state of art?
- Non-traditional designs considered to reduce time and cost of development/evaluation.
- Newer methods incorporate external knowledge bases and ontologies.
- Emphasis towards shareable phenotypes algorithms and/or modular/transportable pipelines.
  - PheKB, eMERGE, OHDSI, CALIBRE, etc.

Review Methods (Banda et al)

Primary method	Number of papers
Rule-based	19
Natural language processing	35
Standard machine learning	25
Learning with noisy data	11
Unsupervised phenotype discovery	11
Hybrid approaches	3
Collaborative frameworks	10
Total	89

Review Data Structures (Alzoubi et al)

Table 2. Categorise of feature extraction techniques.

Feature Extraction Method	Papers
Structured only	[18,35,38,90–96]
BoW only	[30,49]
Keyword search only	[29,54,97]
Concept extraction only	[31,59,69,72,73,75,98]
Structured + BoW	[42,50,52,99]
Structured + Keyword search	[6,16,17,32,55,58,60]
Structured + Concept extraction	[15,28,36,37,48,68,70,71,74,100]

Hripcsak et al. (2012). Next Gen Phenotyping EHRs. JAMIA.
Shivade et al. (2014). Review Identify Phenotype Cohorts EHRs. JAMIA.
Banda et al. (2018). Phenotpying: Rule Based vs. ML. Ann Rev Biomedical Data Sci.
Alzoubi et al. (2019). Auto Phenotyping EHRs: A Review. MDPI Electronics.

Trends in EHR Phenotyping Studies

- Rules/patterns common to phenotyping studies.

Table 2
List of phenotyping patterns derived from a review of 24 eMERGE phenotype algorithms.

Pattern name	Description	Benefits
Anchor date	Define a static date around which all queries and validations are anchored	Makes results reproducible, such that ongoing changes in a person's disease state do not invalidate existing validations
Composition of algorithms	Using an algorithm that was created and validated as a component of another algorithm	Promotes creation and reuse of validated phenotypes to be used in other phenotypes, without re-creating, or using sub-optimal, definitions
Consolidate multiple values	For quantities represented by repeated or multiple measures, provide a single computed value to represent the multiple values	Simplifies analysis and the amount of data that needs to be managed
Account for data outliers	Filter out noisy or incorrect values to help ensure calculations (including average) are not skewed	Improves accuracy of the phenotype definition
Ad hoc categories	Group codes, medications, etc. into ad hoc categories that are not part of a standard terminology or protocol	Simplifies the definition of a phenotype algorithm
Multi-mode sources	Account for information collected in multiple formats or sources across an EHR over time	Can improve accuracy of the phenotype definition by including all sources of data where information may solely be recorded
Established patient	Make sure the patient is seen within the healthcare system at a regular enough basis so that the information pertinent to the algorithm would be on record	Improves accuracy by making sure enough data is present to make an accurate determination about disease state
Confirm variable was checked	Make sure the patient has been seen by a healthcare professional, and that the encounter would be sufficient enough to measure the absence or presence of a disease or other observation	As patients who are not checked for a disease may in fact have that disease, which could confound analyses, ensures that patients have been checked
Qualifiers for evidence	Require additional qualifiers, such as severity, to exist before accepting a clinical observation	May improve accuracy, and also allows for stratification of disease based on its progression, state, and/or severity
Rule of N	Require at least N independent pieces of evidence substantiating a condition or event to reduce the chance that extraneous or incidental data is over interpreted as indicating the condition or event is present	Improves accuracy by correcting for data that could have been recorded without sufficient context
Use distinct time intervals	When requiring a count of items, make sure they happen on different dates and/or times, optionally with some time interval between them	Allows for a more specific phenotype definition by setting a time window in which a disease should have progressed, or ensures that observations are spaced apart to indicate an ongoing condition
Credentials of the actor	Require that a person with appropriate authority (e.g., a physician with specific credentials or practicing in a particular specialty department) recorded the clinical data	Increases confidence or precision in a diagnosis if a specialist has recorded it
Establish assertion status	Determine if assertion qualifiers affect the meaning of medical observations (e.g., the meaning of assertions about "cough" vary depending on whether they are qualified by negation, uncertainty, hypothetical, historical references)	Improves accuracy by establishing context around observations
Medications likely taken	Require more assurance that a patient was actually taking the medication, such as through claims data or having multiple prescriptions over time	Improves accuracy of phenotypes that rely on medication usage to confirm absence and/or presence of the phenotype
Medication details	When checking for medications, it may be necessary to look at dose, frequency and/or route	Allows more precise definition of phenotype where medication attributes are important
Evolving reference standards	Use ranges of dates in which vocabulary codes or lab ranges are valid, if the underlying standards are known to have changed	Increases accuracy of the phenotype by ensuring the right code is used
Transient condition caveats	For patients having transient conditions (e.g., pregnancy), take into account how those transient conditions may alter the interpretation of proximal clinical observations	Improves accuracy of analysis by removing variables that may be confounded by some condition
Medical setting of action	Explicitly require that data be collected (or not be collected) in a particular setting of interest (i.e. inpatient, outpatient)	Improves accuracy of the phenotype by using the encounter setting to add context to the interpretation of an observation
Context of evidence	Consider the context or setting in which a clinical observation is made. For example, when interpreting clinical text mentions of particular conditions, take into account how its interpretation may vary depending on the section of a report in which it appears (e.g., Past Medical History, Problem List, or Family History)	Improve accuracy by looking at the context of how something is recorded
Temporal dependencies	Consider the relationship over time between different events and/or ages at which events occurred	Provides more complete phenotype definitions where progression and temporal dependencies are important
Inception of condition	Explicitly define the date to use when determining the onset of a condition using multiple sources of information (medications, labs, diagnosis codes)	Provides consistency in the results of a phenotype algorithm

Rasmussen et al. (2014). Design Patterns EHR Phenotype Extraction Algorithms. JBI.

Novel Designs: Active Learning (Chen et al, 2014)

- Goal: learn high quality phenotype algorithm with smaller amount labelled data.
- How: (oracle) presents statistical algorithm with "most useful" features for learning.
- Caveat: what is the approach for identifying optimal learning instances from data set?

Active Learning Pipeline

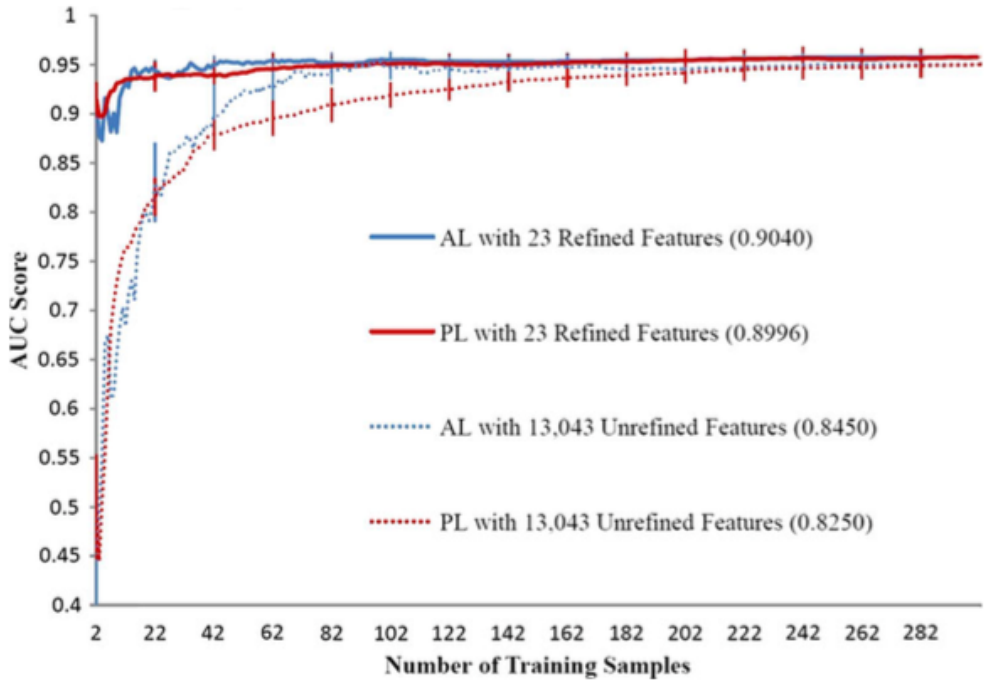
AL for ML-based phenotyping

When building ML classifiers for phenotyping, the conventional method is to select a set of samples randomly for annotation, which is also called passive learning (PL). On the contrary, AL uses a querying algorithm actively to select the most informative samples for annotation in an iterative and interactive fashion. In this study, we simulated a pool-based AL strategy[25] in ML-based phenotyping tasks using existing datasets. We assumed that all samples of a phenotype (eg, 376 samples for RA) were unlabeled and they formed the pool for AL. We then took the following steps:

1. Initialize—we randomly selected two samples from the pool (one control and one case) and used them to build the initial ML model.
2. Predict—we used the trained ML model to predict remaining samples in the pool.
3. Query—we used the uncertainty sampling algorithm[30] to find most informative samples, which are samples that have the most uncertain predicted labels by the ML model. For binary SVM classifiers, uncertainty sampling-based algorithm queries the samples that are closest to the hyperplane, which separates cases and controls. We queried one sample in each iteration.
4. Annotate and re-train the ML model—we annotated the next most informative sample from step 3 (by assigning labels from gold standard) and re-built the ML model by combining the new training sample with previous ones.
5. Repeat and stop—we repeated steps 2–4 until the stop criterion was met. In this study, the learning process stopped when the unlabeled pool was empty.

For PL, we ran the same procedure using random sampling as its querying algorithm.

Comparison Active/Traditional Learning



Chen et al. (2014). Active Learning High Throughput Phenotyping EHRs. JAMIA.

Novel Designs: Learning with Noisy Labels

- Goal: learn high quality phenotype algorithm with smaller amount labelled data.
- How: noisy positive/negative labels for training, based on anchor method (silver standard).
- Anchor is specific code/phrase/concept with high PPV (possibly low sensitivity).
- Amount noisy labels needed proportional to error/noise rate silver standard ($\frac{1}{(1-2\tau)^2}$).

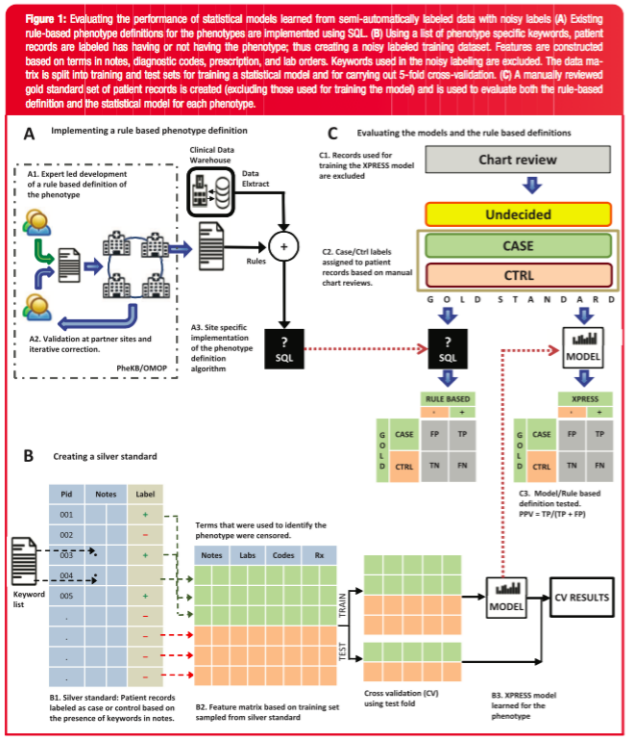


Figure 1: Evaluating the performance of statistical models learned from semi-automatically labeled data with noisy labels (A) Existing rule-based phenotype definitions for the phenotypes are implemented using SQL. (B) Using a list of phenotype specific keywords, patient records are labeled has having or not having the phenotype; thus creating a noisy labeled training dataset. Features are constructed based on terms in notes, diagnostic codes, prescription, and lab orders. Keywords used in the noisy labeling are excluded. The data matrix is split into training and test sets for training a statistical model and for carrying out 5-fold cross-validation. (C) A manually reviewed gold standard set of patient records is created (excluding those used for training the model) and is used to evaluate both the rule-based definition and the statistical model for each phenotype.

Agarwal et al. (2016). Learn Stat Models Phenotyping from Noisy Labelled Data. JAMIA.
Halpern et al. (2016). EHR Phenotyping using Anchor and Learn Method. JAMIA.
Banda et al. (2018). EHR Phenotyping with APHRODITE in OHDSI. JAMIA.

Novel Designs: Learning with Noisy Labels & Feature Engineering

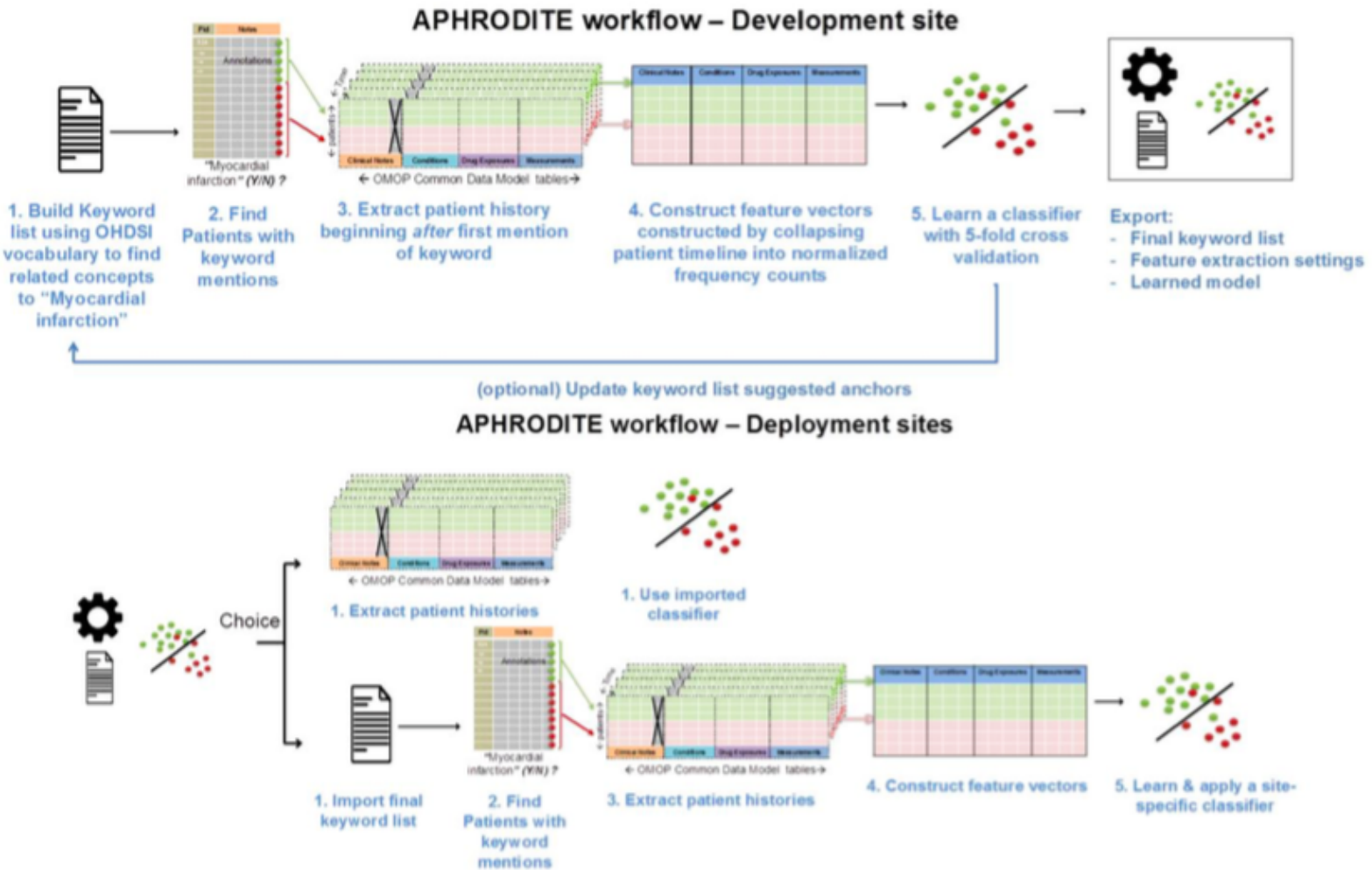


Figure 1. APHRODITE phenotype development/deployment framework schematics. Phenotype definitions are initially learned at development sites and exported for deployment. At deployment sites, users have a choice to use the final keyword list to learn their own site-specific models or use the pre-built classifier.

Banda et al. (2018). EHR Phenotyping with APHRODITE in OHDSI. JAMIA.

Novel Designs: Learning with Noisy Labels & Feature Engineering

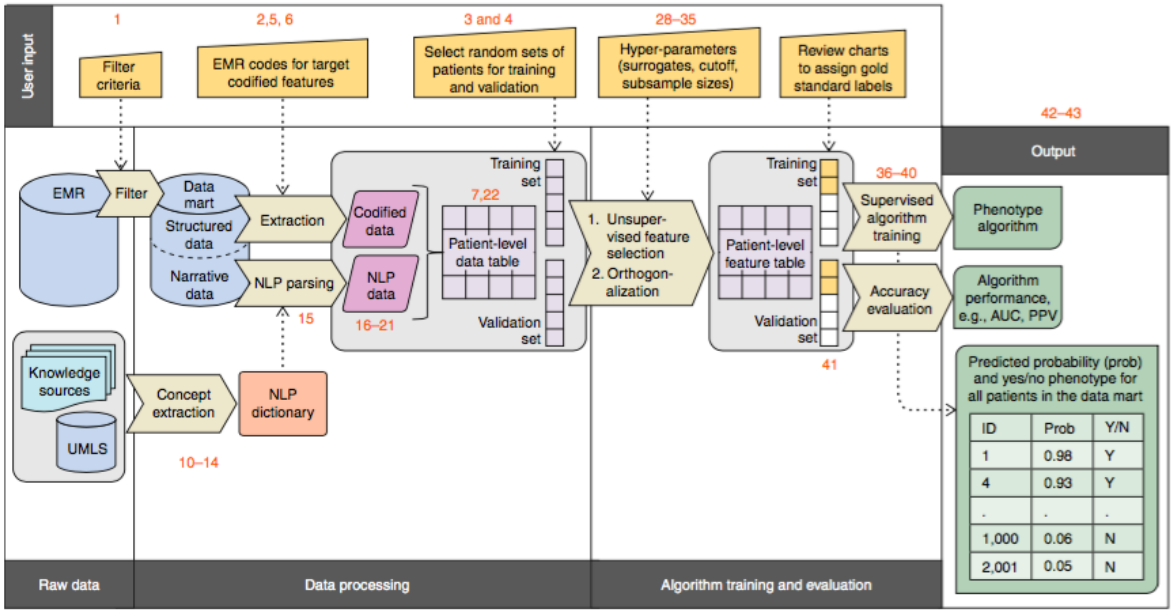


Fig. 4 | Detailed flow of PheCAP protocol. User inputs required at various steps in the PheCAP protocol are specified at the top of the figure as the protocol moves from data extraction, data processing, and algorithm training/validation to the final outputs: a phenotype algorithm, a probability of the phenotype for all subjects in the data mart, and a classification of the phenotype for each subject (yes or no). Orange numbers in the figure correspond to Procedure steps.

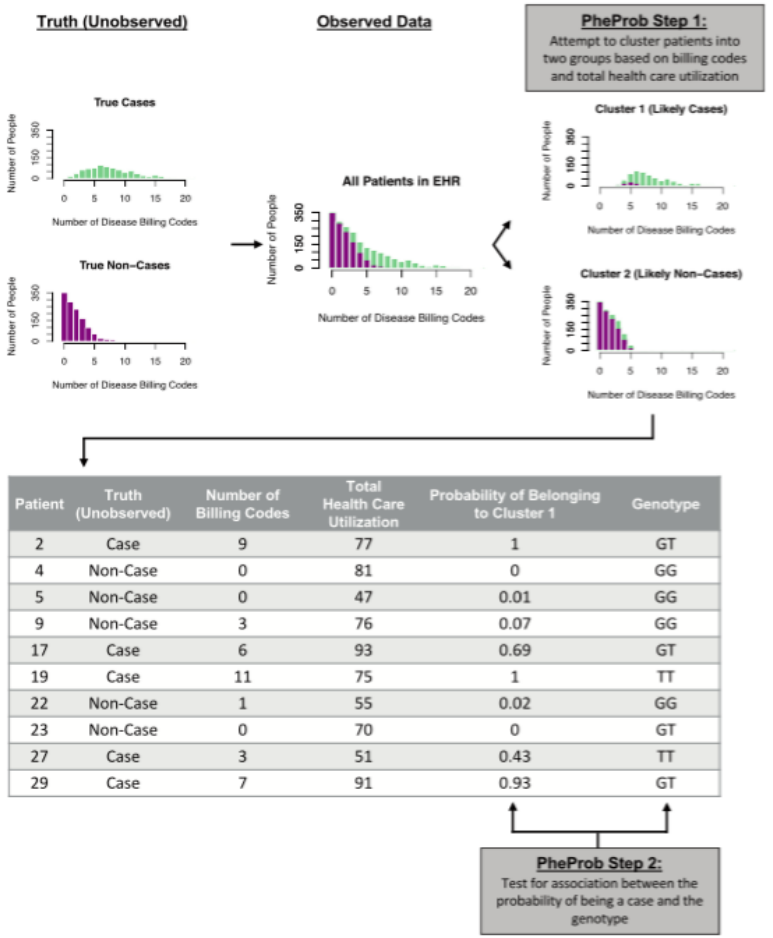
Timing

Steps 1 and 2, creation of the EMR data mart: ~24 h (variable depending on data size and infrastructure)
Steps 3 and 4, chart review and acquisition of gold standard labels: 1 week, depending on the availability of domain expert
Steps 5–7, identification and extraction of codified data features from the structured EMR data for the algorithm: ~12 h
Steps 8 and 9, preparation of analysis environment in R: <1 h
Steps 10–14, concept collection for candidate NLP features: ~12 h
Steps 15–22, note parsing to obtain NLP feature data: ~20+ h; can vary widely depending on size of dataset and size of NLP dictionary
Steps 23–27, loading of EMR data into R: <1 h
Steps 28–35, performing SAFE: <1 h
Steps 36–40, supervised algorithm training: <1 h
Step 41, algorithm validation: <1 h
Steps 42 and 43, application of the model: <1 h

Zhang et al. (2019). PheCAP: Phenotyping Common Semi-Sup Approach. Nature Protocols.

Novel Designs: Unsupervised Phenotyping

- Goal: learn high quality phenotype algorithm with no amount labelled data.
- How: surrogate/anchor features, and unsupervised learning algorithms.



Patient	Truth (Unobserved)	Number of Billing Codes	Total Health Care Utilization	Probability of Belonging to Cluster 1	Genotype
2	Case	9	77	1	GT
4	Non-Case	0	81	0	GG
5	Non-Case	0	47	0.01	GG
9	Non-Case	3	76	0.07	GG
17	Case	6	93	0.69	GT
19	Case	11	75	1	TT
22	Non-Case	1	55	0.02	GG
23	Non-Case	0	70	0	GT
27	Case	3	51	0.43	TT
29	Case	7	91	0.93	GT

Figure 1. Workflow of the PheProb method. True disease status of patients is unknown; instead, the number of billing codes for each disease is observed. The PheProb method clusters individuals based on billing codes, and tests for an association between a genetic marker such as a SNP and the clustering-based probability of being a case.

Sinnott et al. (2018). EHR Phenotyping using Anchor and Learn Method. JAMIA.
Yu et al. (2017). Enabling Phenotypic Big Data with PheNORM. JAMIA.

Novel Designs: Unsupervised Phenotyping

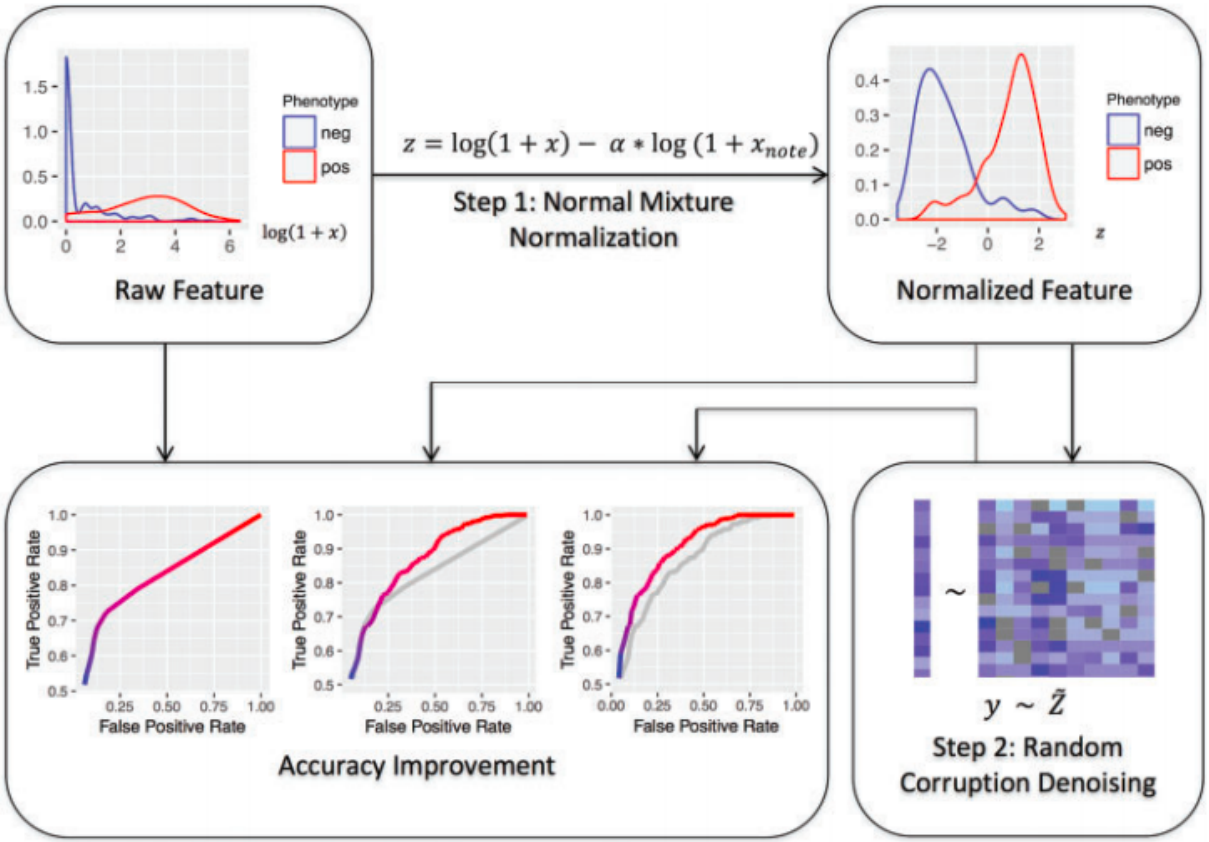


Figure 1. Workflow of PheNorm. Top left: density plot (after logarithm transformation) of a highly predictive feature (illustrated here using the ICD-9-CM count of ulcerative colitis from a Partners HealthCare EHR datamart), denoted by x, in patients who do (the right curve) and do not (the left curve) have the phenotype. Top right: Density plot of the ICD-9-CM count after the normal mixture transformation using the total number of notes in the patient's EHR, denoted by x_{note}. The densities of the phenotype positive and negative patients are approximately normally distributed, and the 2 populations are separated to a large degree. Bottom right: The transformed feature is denoised by self-regression of the transformed feature, denoted by y, onto the entire transformed and randomly corrupted feature set, denoted by $\tilde{Z}$ with dropout. The transformed features are then combined into a prediction formula for disease status classification based on the estimated regression coefficient. Bottom left: The receiver operating characteristic (ROC) curve of the feature or score in each step, with AUC growing steadily (gray curves are copies of the ROC curves from the previous steps).

Yu et al. (2017). Enabling Phenotypic Big Data with PheNORM. JAMIA.

Novel Designs: Learning with Noisy Labels & Zero Label Evaluation

- Goal: learn high quality phenotype algorithm with no amount labelled data.
- How: surrogate/anchor features for noisy model learning.
- How: Extend definition evaluation metrics, allowing zero labels.
- Caveat: Are evaluation metrics these designs valid/unbiased?

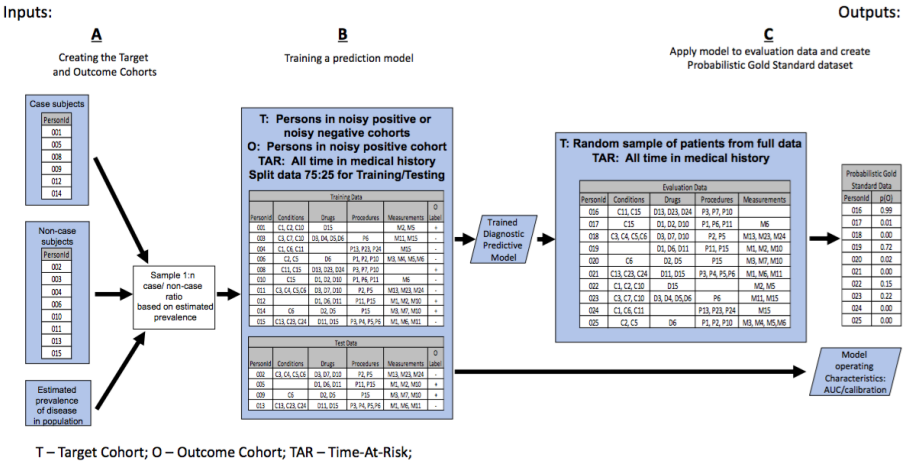


Fig. 1. The first portion of the PheValuator process flow creating the target and outcome cohorts to be used in training the diagnostic predictive model for developing the probabilistic gold standard phenotype data set to be used in phenotype algorithm evaluation.

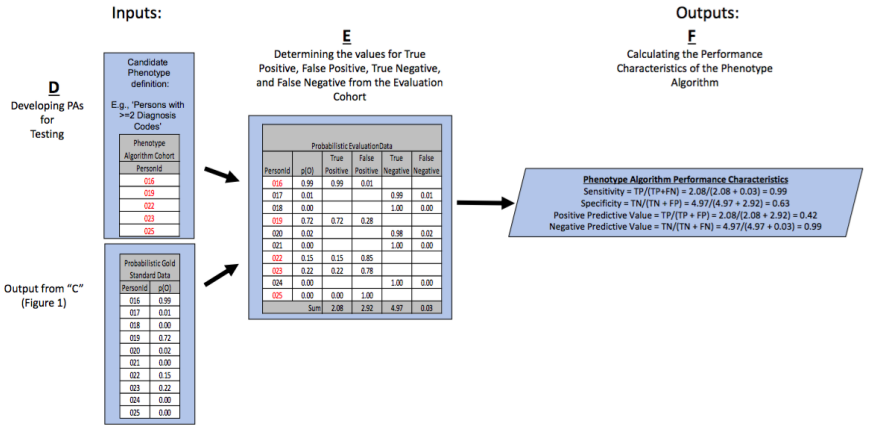


Fig. 2. The last portion of the PheValuator process flow using test phenotype algorithms along with the probabilistic gold standard phenotype data for developing the performance characteristics of the phenotype algorithm.

Swerdel et al. (2019). PheValuator: Devt and Evaluation Phenotyping Algorithms. JBI.

How do we Know if a Proposed Phenotyping Algorithm is Any Good?

- Good design. Strong evaluation. Transparent reporting.
- All phenotype algorithms should be evaluated to be trusted!! See RECORD statement.
- Caution: Many phenotyping strategies. Not all created equal. Susceptibility bias?

Phenotype Reporting (Chen et al)

Recommendations for Research	Recommendations for Reporting
Determine whether cohort definitions are precise enough to establish a reference standard, such as through assessment by multiple reviewers and their interrater reliability.	Describe the EHR data elements used to ascertain the presence or absence of a given variable (e.g., presence of an opioid drug prescription: any prescription record for a drug with the Anatomical Therapeutic Chemical category N02A, for opioid analgesics).
Evaluate the performance of algorithms for cohort identification (i.e., electronic phenotyping) to determine how many subjects an algorithm misses and how many are likely to be incorrectly included in a resulting data set.	Report the data elements, as well as the necessary and sufficient criteria to include or exclude a study subject.
This is consistent with RECORD guideline recommendations: 6.1: The methods of study population selection (such as codes or algorithms used to identify subjects) should be listed in detail. If this is not possible, an explanation should be provided.	If an executable algorithmic method was used to identify cohorts, define what reference standard was used and the performance metrics of human and algorithmic identification methods (e.g., precision, recall, and interrater kappa).
6.2: Any validation studies of the codes or algorithms used to select the population should be referenced. If validation was conducted for this study and not published elsewhere, detailed methods and results should be provided.	This is consistent with RECORD guideline recommendations: 12.1: Authors should describe the extent to which the investigators had access to the database population used to create the study population. 12.2: Authors should provide information on the data cleaning methods used in the study.

Biases in Evaluation? (Hong et al)

We assume the validation sample is a random sample of all patients. However, in practice, subjects may be selected into the validation sample due to convenience sampling or stratified sampling, which may result in biased estimates of accuracy if the estimation methods do not properly account for non-randomness in the ascertainment. This bias is known as work-up bias (Ransohoff and Feinstein, 1978) and verification bias (Begg and Greenes, 1983). Future work is needed to account for the verification bias in the validation set.

Chen et al. (2020). Reporting Obs Studies Using EHR Data. Ann Int Med.
Benchimol et al. (2020). RECORD Statement. PLoS One.
Hong et al, (2017). Semi-Sup Validation EHR Phenotyping. Biometrics.

Open Question

How Does/Should UTOPIAN Design and Evaluate Phenotype Studies?