A Docker based workflow for building machine learning model datasets utilizing the OHDSI common data model

Janos G. Hajagos, Ph.D.
Department of Biomedical Informatics
Stony Brook University

Objective:

The goal of this work is to build a reproducible and deployable pipeline for making OHDSI CDM data available to data scientists. Building machine learning models on EHR data is a challenge due to the large number of features and temporal variation in underlying measurements. The end products of the pipeline are HDF5 (Hierarchical Data Format version 5) files which can be used for training neural networks.

Key Concepts:

OHDSI Common Data Model allows healthcare data from various sources to be stored in a single schema with a standardized vocabulary. It grew out of the work to rigorously evaluate methods and data sets for detecting adverse drug events.

HDF5 is a flexible file container for storing arrays in an organized structure. The concept of groups which is similar to file paths allows the data to be stored in a hierarchy. It supports a range of data types and compression methods. It has been used for storing and analyzing the data for the LIGO experiment to detect gravitational waves, see: (https://losc.ligo.org/s/events/LVT151012/LOSC_Event_tutorial_LVT151012.html).

Docker containerization system for automating the deployment and use of complex software with multiple dependencies.

Health Facts is a de-identified database of EHR (Electronic Health Record) and administrative data from multiple institutions. The database is maintained by Cerner.

HealtheIntent: is a Cerner population health platform that creates a single population health record from multiple EHRs. It maps data elements to a common set of standard vocabularies.

Keras an API for building and training deep neural networks. Tensorflow uses Keras as the API for specifying model structure.

LSTM (Long Short-Term model) is a form of an RNN (Recurrent Neural Network) that allows feedback to be utilized in the learning process.

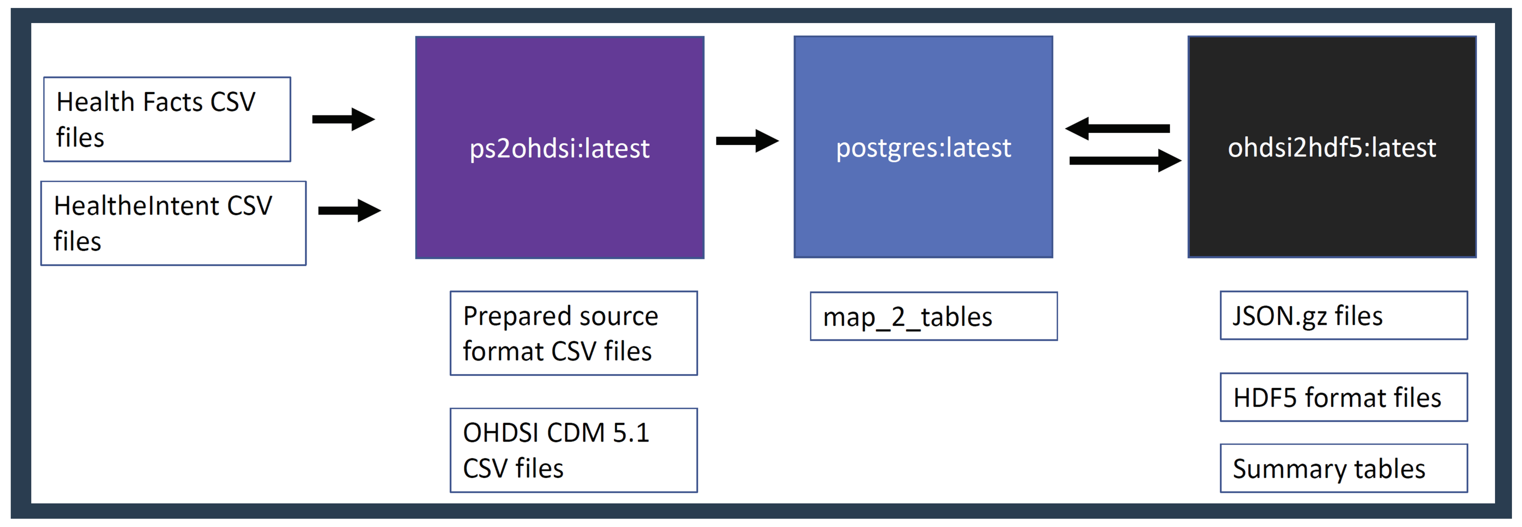
Methods:

Data was extracted from the Cerner's Health Facts de-identified database for adult (18+ years) inpatient visits with discharges occurring over a two-year period (2016-2017). Five separates facilities with the largest inpatient volume were included in the extraction. Data for a three-year period (2016-2018) were extracted from Stony Brook Medicine acute care facilities for adult patients from Cerner's HealtheIntent platform.

Both datasets were mapped to the OHDSI CDM (Version 5.3) using a workflow orchestrated with a Docker server running on a RedHat Enterprise Linux on-premise virtual machine. The code for all Docker machines is at: https://github.com/jhajagos/Dockers4Health-CareDB2HDF52ML. This involves first mapping each source to a standardized intermediary format. The intermediary format is then mapped to the OHDSI CDM. OHDSI mapped data was then loaded into a PostgreSQL database server. Data in a PostgreSQL server was then extracted and mapped to HDF5 matrix format. HDF5 files were used by a data science team to build predictive models for coded diagnoses in Python using scikit-learn and Keras libraries.

There are two potential types of HDF5 files that are built with the dockerized workflow. The first type automatically encodes using dummy variables all coded diagnoses, drug exposures, and coded measurement and observation results (e.g., high blood creatinine). For numeric results, such as, blood glucose the mean, median, max, min, first, and last result are calculated for each inpatient visit. Each inpatient visit is represented as a single row vector. The second type of HDF5 file is built using sequential data such as the change in recorded blood glucose during an inpatient visit. Each visit is represented as a matrix where each row in the matrix is a temporal change in measurement values or exposure to a specific drug.

Docker based workflow



https://github.com/jhajagos/Dockers4HealthCareDB2HDF52ML

Mapping Results:

CDM Table	HDF5 Group	Health Facts 5 institutions: Number of records in table	Health Facts 5 institutions: Number of columns in matrix	Institutional HealtheIntent: Number of records in table	Institutional HealtheIntent: Number of columns in matrix
Person	/ohdsi/person/	187,187	13	66,096	30
visit_occurrence	/ohdsi/visit_occurrence/	275,407	72	102,028	121
condition_occurrence	/ohdsi/condition_occurrence/	4,241,357	9,358	2,678,585	7,695
procedure_occurrence	/ohdsi/procedure_occurrence/	149,996	4,038	419,445	9,921
measurement	/ohdsi/measurement/count/	141,359,529	994	101,338,048	2,209
observation	/ohdsi/observation/count/	3,229,858	708	31,890,686	666
drug_exposure	/ohdsi/drug_exposure/count/	3,399,480	1,343	5,861,098	7,654

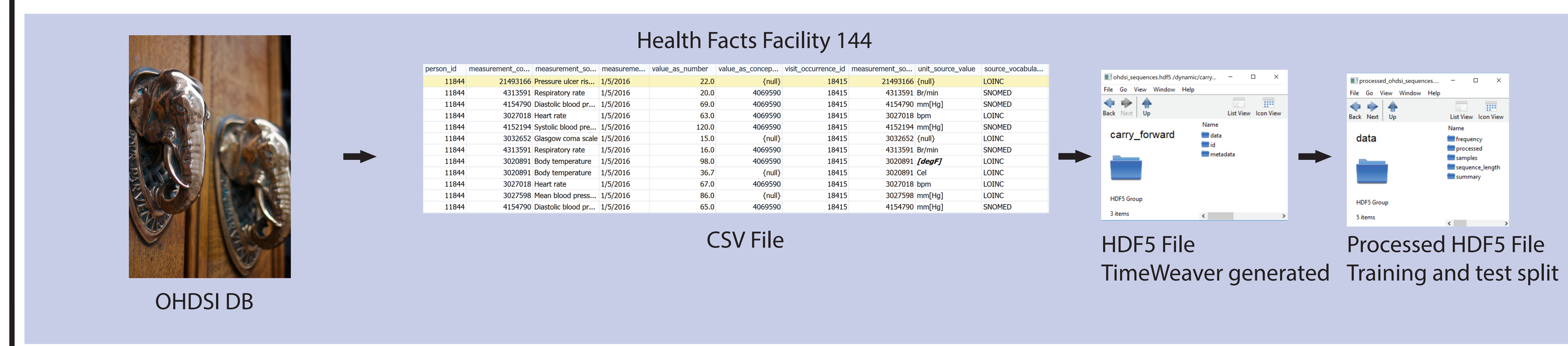
Institutional HealtheIntent

c1	c2	c3	c4	non-zero	to_include	fraction non-zero	unique	fraction_unique
4113006	Assisted	Ability to perform activities of everyday life	categorical	17,594	1	0.511	14,301	0.550
4113006	Dependent	Ability to perform activities of everyday life	categorical	5,428	1	0.158	4,390	0.169
4113006	Independent	Ability to perform activities of everyday life	categorical	14,447	1	0.419	12,274	0.472
4121059	Assisted	Eating, feeding and drinking abilities	categorical	8,713	1	0.253	6,912	0.266
4121059	Dependent	Eating, feeding and drinking abilities	categorical	3,279	1	0.095	2,716	0.105
4121059	Edentulous	Eating, feeding and drinking abilities	categorical	460	1	0.013	436	0.017
4121059	Independent	Eating, feeding and drinking abilities	categorical	26,832	1	0.779	21,040	0.810
4121059	Nil by mouth	Eating, feeding and drinking abilities	categorical	8,981	1	0.261	7,644	0.294
4121059	None	Eating, feeding and drinking abilities	categorical	26,246	1	0.762	20,639	0.794
4137801	Continuous	Coughing	categorical	602	1	0.017	573	0.022
4137801	None	Coughing	categorical	27,569	1	0.800	21,223	0.817
4137801	Occasional	Coughing	categorical	7,114	1	0.206	5,804	0.223
4137801	Weak	Coughing	categorical	1,044	1	0.030	994	0.038
4139528	Active Durable Power of Attorne	Active advance directive	categorical	785	1	0.023	746	0.029
4139528	Active living will	Active advance directive	categorical	1,377	1	0.040	1,285	0.049
4173786	Adequate	Antenatal care	categorical	2,785	1	0.081	2,682	0.103
4186106	None	Anticoagulation contraindicated	categorical	21,935	1	0.636	17,510	0.674
4222407	Condom	Urinary catheterization status	categorical	360	1	0.010	328	0.013
4222407	Discontinued	Urinary catheterization status	categorical	8,228	1	0.239	7,622	0.293
4222407	Evaluation procedure	Urinary catheterization status	categorical	9,420	1	0.273	8,387	0.323
4222407	Insert	Urinary catheterization status	categorical	7,514	1	0.218	6,786	0.261
4222862	Doppler device	Posterior tibial pulse	categorical	371	1	0.011	349	0.013
4222862	Normal	Posterior tibial pulse	categorical	1,470	1	0.043	1,434	0.055
4222862	Thready pulse	Posterior tibial pulse	categorical	528	1	0.015	505	0.019
4224770	None	Social support status	categorical	439	1	0.013	416	0.016
4224770	Parent	Social support status	categorical	2,192	1	0.064	1,744	0.067
4224770	Partner in relationship	Social support status	categorical	1,237	1	0.036	1,065	0.041
4224770	Sibling	Social support status	categorical	2,180	1	0.063	1,767	0.068
4224770	Spouse	Social support status	categorical	7,664	1	0.222	5,951	0.229

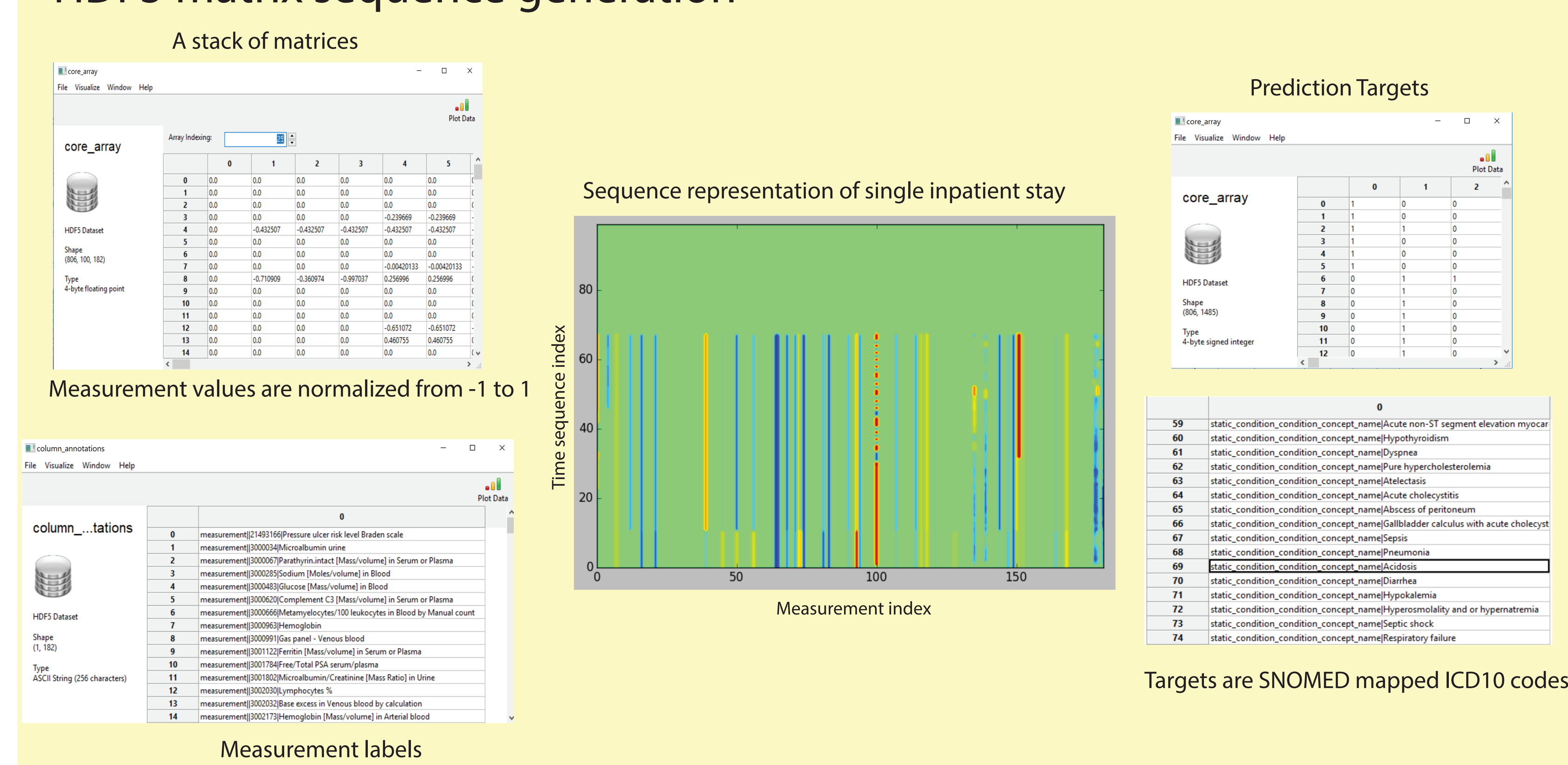
Health Facts 5 institutions

c1	c2	c3	fraction non-zero	unique	fraction_unique
3023599	Within reference range	MCV [Entitic volume] by Automated count	0.510	97,625	0.522
3024128	High	Total Bilirubin serum/plasma	0.037	7,704	0.041
3024128	Within reference range	Total Bilirubin serum/plasma	0.127	25,377	0.136
3024354	Other	Oxygen [Partial pressure] in Venous blood	0.033	7,859	0.042
3024561	Low	Albumin serum/plasma	0.078	15,327	0.082
3024561	Within reference range	Albumin serum/plasma	0.105	22,303	0.119
3024928	Other	Oxygen saturation in Venous blood	0.031	7,520	0.040
3024929	High	Platelets [#/volume] in Blood by Automated count	0.093	18,776	0.100
3024929	Low	Platelets [#/volume] in Blood by Automated count	0.170	33,772	0.180
3024929	Within reference range	Platelets [#/volume] in Blood by Automated count	0.518	101,002	0.540
3026023	Low	Comprehensive metabolic panel serum/plasma	0.021	4,582	0.024
3026023	Normal	Comprehensive metabolic panel serum/plasma	0.112	19,959	0.107
3026782	Within reference range	Osmolality of Urine	0.013	3,403	0.018
3027008	Other	Opiates [Presence] in Urine	0.014	3,330	0.018
3027018	High	Heart rate	0.184	35,901	0.192
3027018	Low	Heart rate	0.071	16,217	0.087
3027018	Normal	Heart rate	0.421	82,243	0.439
3027114	High	Cholesterol [Mass/volume] in Serum or Plasma	0.019	5,044	0.027
3027114	Within reference range	Cholesterol [Mass/volume] in Serum or Plasma	0.078	18,657	0.100
3027219	High	Urea nitrogen [Mass/volume] in Venous blood	0.013	2,920	0.016
3027219	Within reference range	Urea nitrogen [Mass/volume] in Venous blood	0.011	2,879	0.015
3027245	Within reference range	Hepatitis A virus IgM Ab [Units/volume] in Serum by Immunoassay	0.016	4,199	0.022
3027273	Other	Bicarbonate [Moles/volume] in Venous blood	0.033	7,858	0.042
3027388	High	Alanine aminotransferase [Enzymatic activity/volume] in Serum or	0.041	9,253	0.049
3027388	Within reference range	Alanine aminotransferase [Enzymatic activity/volume] in Serum or	0.121	24,016	0.128
3027484	Low	Hemoglobin [Mass/volume] in Blood by calculation	0.014	3,335	0.018

Training an LSTM model to predict SNOMED conditions



HDF5 matrix sequence generation



Tensorflow model built using Keras API

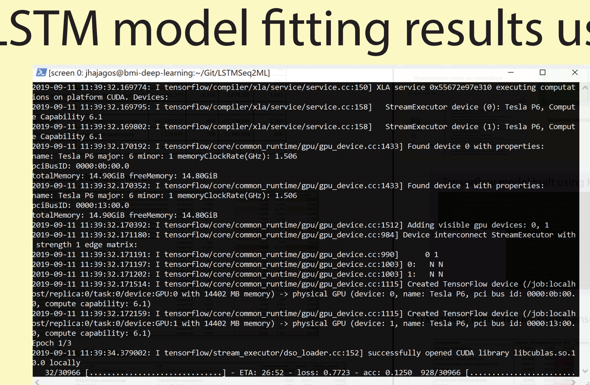
```python
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM, Dropout, Masking, SimpleRNN
import numpy as np
import h5py
import pprint
import argparse

model = Sequential()
model.add(Masking(mask_value=0.0, input_shape=f5_train_array.shape[1:]))
model.add(LSTM(128, activation="tanh", return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(128, activation="tanh"))
model.add(Dropout(0.2))
model.add(Dense(64, activation="relu"))
model.add(Dropout(0.2))

model.add(Dense(1, activation="sigmoid"))
opt = tf.keras.optimizers.Adam(lr=1e-3, decay=1e-5)

model.compile(loss="binary_crossentropy", optimizer=opt, metrics=["accuracy"])
# This can be refactored as explicit casts are not needed
model.fit(np.array(f5_train_array, dtype="float32"), np.array(f5_target[:, target_index], dtype="int32"), epochs=3,
        validation_data=(np.array(f5_test_array, dtype="float32"),
        np.array(f5_test_target[:,target_index],dtype="int32")))
```

LSTM model fitting results using Tensorflow GPU acceleration



GPU accelerated training

Concept name	concept_id	Model loss	AUC ROC	Positive cases	Frequency
Nicotine dependence	4209423	0.386	0.717	5,007	16.2%
Anemia	439777	0.299	0.852	4,124	13.3%
Acute renal failure syndrome	197320	0.230	0.932	3,621	11.7%
Hypo-osmolality and or hyponatremia	435515	0.143	0.929	1,931	6.2%
Dyspnea	312437	0.172	0.859	1,764	5.7%
Hyperglycemia due to type 2 diabetes mellitus	37016349	0.135	0.912	1,470	4.7%
Urinary tract infectious disease	81902	0.153	0.801	1,426	4.6%
Acidosis	435517	0.110	0.892	1,051	3.4%

Training set contains 30,906 instances and test set 6,277 instances

There are 190 different measurements with a maximum of 100 time steps

Conclusion:

This work demonstrates the feasibility of mapping real world inpatient EHR datasets with large number of clinical measurements into a usable format for machine learning. The OHDSI CDM provides a robust data model to represent clinical data. Once clinical data has been transformed into the OHDSI CDM the mapping process can be run using a Docker based workflow. While Docker does not solve all deployment issues it simplifies the use of complex scientific software and dependencies. The HDF5 files make it easy to build and train complex models on OHDSI data such as LSTMs.

Abstract:

A Docker based pipeline was developed to map data from EHRs (electronic health records) in OHDSI CDM (common data model) to multiple HDF5 formats. Two real world inpatient datasets were mapped to HDF5: a large de-identified EHR database for 5 institutions and an institutional EHR database for 3 years of adult inpatient stays. The HDF5 datasets were then used by a data science team to build predictive models for coded conditions.

Links:

Docker Pipelines: https://github.com/jhajagos/Dockers4HealthCareDB2HDF52ML
TransformDBtoHDF5ML (map DB to HDF5) : https://github.com/jhajagos/TransformDBtoHDF5ML
MappingOHDSI2HDF5 (OHDSI templates): https://github.com/SBU-BMI/MappingOHDSI2HDF5
TimeWeaver (sequential data mapping): https://github.com/jhajagos/TimeWeaver

Presented at the 2019 OHDSI Symposium (September 16th)

References:

1. Miotto, R, Li, L., Kidd, BA, & Dudley, J. T. (2016). Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records. Scientific Reports, 6, 26094.
2. Neill, D. B. (2013). Using Artificial Intelligence to Improve Hospital Inpatient Care. IEEE Intelligent Systems, 92–95.
3. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
4. Chollet, F. (2015) Keras, GitHub. https://github.com/fchollet/keras