CSE6250 Projects: Big Data Analytics for Healthcare

Jimeng Sun

***Abstract*—CSE6250 Big Data Analytics for Healthcare is a graduate level course focusing on practical big data technology for health analytic applications. One big part of this course is to conduct an individual/group project that addresses a real-world data science problem in healthcare. The project should provide an end-to-end coverage of data science activities in addressing a real healthcare problem. The project should utilize big data systems such as Hadoop and Spark, machine learning algorithms that are covered in this class and real-world health related data. I hope that the best projects (with some additional effort) can lead to publications at the best medical informatics venues such as Journal of the American Medical Informatics Association (JAMIA), Journal of Biomedical Informatics (JBI), Journal of Medical Internet Research (JMIR), Artificial Intelligence in Medicine, IEEE Journal of Biomedical and Health Informatics (JBHI).**

This document provides the project guideline such as expectation, timeline, deliverables. We also introduce recommended project topics for selection but you are welcome to propose your own project as long as they are related to big data technology covered in this course and addressing healthcare problems.

***Index Terms*—Big data, Health analytics, Data mining, Machine learning**

I. Introduction

BIG data and healthcare applications interact closely nowadays thanks to the advancement in electronic data capturing technology such as electronic health records, on-body sensors and genome sequencing. This course is about learning practical skills on big data systems, scalable machine learning algorithms and their applications to healthcare. Through (painful) homework exercises, all the students should have by now learned big data systems and acquired sufficient knowledge about healthcare data. We believe you are ready to take on the next level of challenges as a data scientist in healthcare. That is, you are going to propose, execute and report an awesome data science project. The final results of this project includes **1) a publishable report and 2) a convincing presentation, and 3) reusable software and sufficient documentation from your project.**

Next we will cover the project life cycle, timeline, deliverables, grading scheme and project topics.

II. Project Life Cycle

As a data scientist working on a real-world project, you have to be able to conduct all aspects of the big data project independently in a timely manner. In particular, here are some tasks that a data scientist will have to conduct in a big data project: project initiation, project execution and project report.

J. Sun is with School of Computational Science and Engineering Georgia Institute of Technology, Atlanta, GA, 30332 USA e-mail: (see http://www.sunlab.org).

A. Project initiation

As a data scientist, projects are not always there for you to work on. You have to create them and convince your boss (e.g., your CEO) to fund that. Before your project is officially launched, you have to conduct many steps to make that happen. Here are the checklist of things that you should do during the project initiation.

1) Identify and motivate the problems that you want to address in your project.
2) Conduct literature search to understand the state of arts and the gap for solving the problem.
3) Formulate the data science problem in details (e.g., classification vs. predictive modeling vs. clustering problem).
4) Identify clearly the success metric that you would like to use (e.g., AUC, accuracy, recall, speedup in running time).
5) Setup the analytic infrastructure for your project (including both hardware and software environment, e.g., Azure or local clusters with Spark, python and all necessary packages).
6) Discover the key data that will be used in your project and make sure an efficient path for obtaining the dataset. This is a crucial step and can be quite time-consuming, so do it on the first day and never stops until the project completion.
7) Generate initial statistics over the raw data to make sure the data quality is good enough and the key assumption about the data are met.
8) Identify the high-level technical approaches for the project (e.g., what algorithms to use or pipelines to use).
9) Prepare a timeline and milestones of deliverables for the entire project.
10) It's **required to incorporate big data tools** say Hadoop, Spark, Pyspark in your project. For example, you can utilize Spark to do data preprocessing or create ETL pipeline then switch to python for ML/DL model training.
11) **3-4 students per team** is encouraged because the rubrics is consistent regardless of team size. If you want to do it individually, please make sure your workload level is qualified because there is no special considerations working as an individual as opposed to collaborating with others.
12) **No late submission** allowed or grace period used towards project

All the above steps in project initiation should be demonstrated in your proposal.

B. Project execution

Once your project is approved, you should quickly work on getting results and iterate with your sponsors on the progress. Iteration is the key. The first iteration should be fast and positive otherwise you are at risk losing momentum from the sponsors/project owners (e.g., your boss, clinical experts, your partners from another organization). This successful execution will lead to long-term sustainability of your team and will greatly improve your reputation in the organization, so please focus on that. **You are supposed to distribute the tasks evenly among your team at the initial project phase. Please cite any useful references say papers, github, websites, etc.. Once you are found cheating, we will report it to College of Computing.**

1) Gather data that will be used in your project if you haven't already.
2) Design the study (e.g., define cohort, target and features; carefully split data into training, validation to avoid overfitting)
3) Clean and process the data.
4) Develop and implement the modeling pipeline.
5) Evaluate the model candidates on the performance metrics.
6) Interpret the results from your model (e.g., show predictive features, compare to literature in terms of finding, present as cool visualization).

All the steps in project execution should be done by the paper draft due date and iterate at least another time by the final due day.

C. Project report

Finally, you are close to the end of the project. You need to summarize what you have done and learned throughout the project. This will be a comprehensive, concise and well-written report as the foundation for future projects. This can lead to publications and other external communication. You will probably need to give a presentation to your sponsors. So do the best you can in written report and presentation. Bad delivery at this step can overshadow all the great work your team have put in throughout the project, so do spend sufficient time to prepare a slick presentation and write a comprehensive report.

- Your final report should consists of the following sections.
  1) Title and abstract
  2) Introduction and background
  3) Problem formulation
  4) Approach and implementation
  5) Experimental evaluation
  6) Conclusion
- Prepare a presentation deck and deliver a convincing and informative presentation.
- Clean up and package your code, and document the necessary steps for future usages by others.

Please use the above process to guide your own project for this semester and possibly your future data science career.

III. Logistics

Next we summarize the timeline and deliverables for your project in this semester.

A. Timeline (*we don't allow late submission for proposal / draft / final report)*

Due Date	Task Description
Sep 27	Project group formation
Oct 11	Project proposal submission
Nov 8	Project draft
Dec 6	Final Submission (final paper + code + presentation)

B. Deliverables

Everyone needs to submit each milestone report onto Gradescope at team level, remember to include your teammates in submission. Please check out instructions how to submit group project on Gradescope. You can find more detailed submission requirements as below.

0) Paper Templates:

- Please use either MS Word or LaTex template from the link provided below for your proposal, draft, and final paper, but you should submit it in PDF format at the end.
- AMIA Templates [Word][LaTex]

1) Project Proposal:

- Up to 3 pages write-up + 1 page of references (with minimum 6 reference papers)
- Guide:
  - Motivation: Why is this problem important? Why do we care this problem?
  - Literature Survey: Conduct literature search to understand the state of arts and the gap for solving the problem. Formulate the data science problem in details (e.g., classification vs. predictive modeling vs. clustering problem).
  - Data: Describe the dataset you use, and elaborate on how you would play with the data in your project. Preliminary results are encouraged but not required. It is recommended to try to cover as many aspects as described in project initiation to give you a better navigation in later period of project phase. This is a crucial step, please do it on the first day and never stops until the project.
  - Approach: Identify the high-level technical approaches for the project (e.g., what algorithms to use or pipelines to use). Identify clearly the success metric that you would like to use (e.g., AUC, accuracy, recall, speedup in running time).
  - Experimental Setup: Setup the analytic infrastructure for your project (including both hardware and software environment, e.g., AWS or local clusters with Spark, python and all necessary packages).

  - Timeline: Prepare a timeline and milestones of deliverables with reasonably proposed task distributions for the entire project.

2) Project draft:

- Up to 5-page write-up + 1 page of references (minimum 8 reference papers)
- All sections which are common in research publication such as Abstract, Introduction, Approach/Metrics, Experimental Results, Discussion, and Conclusion/Optimization must be there.
- Guide
  - Describe your method/approach clearly, concisely, but specifically. It should be at the level of that any reader can follow and reproduce your work after she read your paper.
  - Even if your current results are not good as expected, there must be analyses about what possible reasons and solutions/plans are. It is same for when your results are good also.
  - Here the 'results', especially for the draft, can be any valuable results. For example, results from simple model which can be used as one of baseline in your final paper, results from intermediate steps prior to your ultimate target task, results with a tiny subset of your dataset generated for a verification of your pipeline, etc., all those followed by your own analysis can be used.
  - Grading of both your draft and final paper is not based on some 'numbers' from metrics. Instead, it will be on how comprehensive your project/paper is; how well the problem and the task are defined; whether your approach including feature extraction/processing and modeling are reasonable and convincing; how well you described these things, etc.
  - Show us the continued optimization plan you will try and why do you think it's useful to improve the model performance.

Please check the course web page for the review format and follow it.

3) Final report/Submission requirements:

- Check with the requirements from above sections C of 'project report'. Please watch out the format and make it publishable.
- You are also required to share with us what kind of challenges you met in this project and how you learnt from it.
- 6-8 pages write-up + 1 page reference, see sample papers for reference.
- Put the Youtube link of 5-8mins presentation(demo by one representative or multiple students, set an access key if you want) under the report title.
- PPT slides to summarize your paper.
- Software implementation and documentation. Codes with clear comments and Readme (.md) file are required; dataset is not a must to share. If you would like to submit the repository link, please set it as **private** using github.gatech.edu, bitbucket, google drive, google cloud, etc, and put the link under the report title as well.
- **Team contributions** will be included at the end of final report (after reference section). Please contact Ming (mliu302) via email when necessary and penalty will be reflected sometimes for those due to inactive contributions.
- Deliverables: Besides submitting the final report to Gradescope at team level. Please also zip your final report, codes, slides and name it as team #-topic e.g. team5-chestXray then submit on Canvas at individual level.

C. Grading scheme

Your draft and final paper should be in a form of regular research publication. It means all sections common in typical research publications such as Abstract, Introduction, Method, Experimental Results, Discussion, and Conclusion must be there even with different section names or structures. You should organize well and write clearly each section so that it is easily readable for other readers.

Please keep on right track and actively contribute to your team during all the periods, team contribution will be evaluated during final report phase.

Here are the grading guideline for your project.

- Project 25%
  - 3% proposal
  - 7% paper draft
  - 5% final presentation
  - 10% final paper

IV. Project Topics

We introduce several project topics for your consideration but you can also propose your own project outside this scope as long as your project uses big data tools (e.g., Hadoop and Spark) and is about healthcare applications.

A. Chest X-ray Disease Diagnosis

Mentor: Xuesong Pan ⟨xpan79@gatech.edu⟩, Jingyi Li ⟨jli647@gatech.edu⟩

X-rays are the oldest and most frequently used form of medical imaging, but they require significant training for clinicians to read correctly. This makes the analysis of x-rays costly, time consuming, and prone to error. Luckily, the latest big data techniques, especially deep learning, are making automated analysis of x-ray images increasingly more realistic, and groups are publishing large x-ray imaging dataset to help researchers train, test, and improve their approaches. Creating an automated diagnosis system would speed up processing, reduce effort from clinicians, reduce errors, and make x-rays more practical for diagnoses that currently rely on more expensive but easier to analyze technologies like computerized tomography.

The goal of this project is to reproduce and improve previous study or propose a new study using at least one of the given dataset (see below). If you only use CheXpert data[17], you might also want to paricipate in the CheXpert competition.

If you can get a good ranking, it's encouraged to share with us then.

- **Dataset**:
  - NIH Chest X-ray Dataset
  - CheXpert Dataset (Register to get the access)
  - JSRT Database (Register to get the access)
- **Related Work**:
  - Liu et al. [26] present a novel method termed as segmentation-based deep fusion network (SDFN), which leverages the higher-resolution information of local lung regions. Specifically, the local lung regions were identified and cropped by the Lung Region Generator (LRG). Evaluated by the NIH benchmark split on the Chest X-ray 14 Dataset, the experimental result demonstrated that the developed method achieved more accurate disease classification compared with the available approaches via the receiver operating characteristic (ROC) analyses.
  - Irvin et al. [17] present a large labeled dataset and investigate different approaches to using the uncertainty labels for training convolutional neural networks that output the probability of these observations given the available frontal and lateral radiographs.
  - Wang et al. [34] introduce the Chest X-ray dataset (containing over 110K images) with an overview of the data sources, methodology for deriving the labels, and provide some initial benchmark results using different pre-trained convolutional neural networks to classify each disease type. They also use a weakly-supervised localization techniques to understand where in the image the network believes the disease occurs and use a subset of 1,600 images to evaluate the accuracy of this localization approach.
  - Rajpurkar et al. [31] significantly improve on the modeling techniques to achieve state-of-the-art results using a much deeper and more sophisticated 121-layer DenseNet architecture. They evaluate the effectiveness of this network versus radiologists and find their network provides even better results. They also use an alternative localization approach to understand where in the image the network identifies a disease but don't provide specific validation results.
  - Guan et al. [11] proposes a category-wise residual attention learning framework(CRAL) for multi-label chestX-ray image classification. CRAL improves the performance of CNN and benefits from both relevant and irrelevant pathologies.
  - Zongyuan Ge et al. [8] propose a deep network architecture based on fine-grained classification concept incorporating a novel error function called multi-label softmax loss (MSML) which handles both the presence of multiple data labels and imabalance across classes which is common in most medical problems.
  - Li et al. [24] present a method to identify diseases and their locations with a small set of training data by using a convolutional neural network (CNN) to learn the entire image as well as patch sliced images to learn local information. They compare their model with pre-trained ResNet-50, and find their model generates better results.

B. Deep learning in Drug Discovery

Mentor: Tianfan Fu ⟨tfu42@gatech.edu⟩

Designing molecules or chemical compounds with desired properties is a fundamental task in drug discovery. For example, if we are given a molecule with low solubility as an input molecule, we want to generate its paraphrase with high solubility. Since the number of drug-like molecule is large estimated between 10^{23} and 10^{60} [30], traditional methods such high throughput screening (HTS) is not scalable. One particular task in drug discovery is called *lead generation*, where after a drug candidate (a *hit*) is identified via HTS, enhanced similar candidates are created and tested in order to find a lead compound with better properties than the original hit. To model lead optimization as a machine learning problem, the training data involves a set of paired molecules that map input molecule X to target molecule Y with more desirable properties. The goal is to learn a generating model that can produce target molecules with better properties from an input molecule.

- **Dataset**: ZINC [32]. Processed training and test data for three molecule optimization tasks is available at https://github.com/wengong-jin/iclr19-graph2graph.
- **Related Work**:
  - One research line is to formulate molecular generation as a sequence generation problem. Most of these methods are based on the simplified molecular-input line-entry system (SMILES), a line notation describing the molecular structure using short ASCII strings [36]. Character Variational Auto-Encoder (C-VAE) generates SMILES string character-by-character [10]. Grammar VAE (G-VAE) generates SMILES following syntactic constraints given by a context-free grammar [23]. Syntax-directed VAE (SD-VAE) that incorporates both syntactic and semantic constraints of SMILES via attribute grammar [6]. However, many generated SMILES strings using these methods are not even syntactically valid which lead to inferior results.
  - Another kind of method is to directly generate molecule graphs. Comparing sequence based method, graph-based methods bypass the need of generating valid sequences (e.g., SMILES strings) all together. As a result, all the generated molecules are valid and often with improved properties. The original idea of [19][1] is to eliminate cycles in a molecular graph by representing the graph as a scaffolding tree where nodes are substructures such as rings and atoms. Then a two-level generating function is used to create such a tree then decode the tree into a new molecular graph. Recently another enhancement is produced called graph-to-graph translation model [20][2], which extends the junction

[1] https://github.com/wengong-jin/icml18-jtnn
[2] https://github.com/wengong-jin/iclr19-graph2graph

variational autoencoder via attention mechanism and generative adversarial networks. It is able to translate the current molecule to a similar molecule with pre-specified desired property (e.g., drug-likeness). Also a Reinforcement Learning (RL) based method with graph convolutional policy network has been proposed to generate molecular graphs with desirable properties [38], [40][3].

Task In this assignment, we focus on reproduce the experiment of any of these molecule generation methods.

C. NLP for Healthcare

Mentor: Jaewoo Park ⟨jpark965@gatech.edu⟩, Ming Liu ⟨mliu302@gatech.edu⟩

Unstructured healthcare data like clinical notes usually contain much richer information than structured data such as structured parts of electronic health records (EHR) and insurance claims records. However, it is difficult to manually extract useful information from unstructured data in terms of time and labor cost. Therefore, it is getting important more and more to handle the operations of ETL (Extract, Transform, and Load) using Natural Language Processing (NLP) in healthcare domain.

- **Resources**:
  - MIMIC III, which is useful in many cases. Please submit your data access request to MIT using your GT-Email and MIMIC completion report which lists modules taken with dates and scores. Request is made per person, not per team
  - Criteria2Query: Automatically Transforming Clinical Research Eligibility Criteria Text to OMOP Common Data Model (CDM)-based Cohort Queries
  - github for EliIE[22]
  - github for dataset and algorithm used in [28]
- **Related Work**:
  - James, Sarah, Jimeng, etc. [18] present CAML, a convolutional neural network for multi-label document classification, which aggregates information across the document using a convolutional neural network, and uses an attention mechanism to select the most relevant segments for each of the thousands of possible codes. According to their evaluation, the method is accurate, achieving precision better than the prior state of the art.
  - Kang et al. [22] presented an open-source information extraction system called Eligibility Criteria Information Extraction (EliIE) for parsing and formalizing free-text clinical research eligibility criteria (EC) following Observational Medical Outcomes Partnership Common Data Model (OMOP CDM) version 5.0. EliIE parses EC in 4 steps: (1) clinical entity and attribute recognition, (2) negation detection, (3) relation extraction, and (4) concept normalization and output structuring. A sequence labeling–based method was developed for automatic entity and attribute recognition. Negation detection was supported by NegEx and a set of predefined rules. Relation extraction was achieved by a support vector machine classifier. They further performed terminology-based concept normalization and output structuring. According to their evaluation, machine learning-based EliIE outperforms existing systems and shows promise to improve.
  - Ma and Weng [28] investigated the correlation between drug safety label changes and study population focus shift patterns for existing interventional drug trials. They defined the Convergent Focus Shift (CFS) pattern for each prescription drug as the converged focus in post-marketing trials compared to that in premarketing trials. They hypothesized that drugs with potential safety warnings have different CFS patterns compared to those without warnings. They demonstrated the added value of linked public data and the feasibility of integrating ClinicalTrials.gov summaries and drug safety labels for post-marketing surveillance.
  - Ma and Weng [27] presented a method for identifying questionable exclusion criteria for 38 mental disorders. They extracted common eligibility features (CEFs) from all trials on these disorders from ClinicalTrials.gov. Network Analysis showed scale-free property of the CEF network, indicating uneven usage frequencies among CEFs. By comparing these CEFs' term frequencies in clinical trials' exclusion criteria and in the PubMed Medical Encyclopedia for matching conditions, they identified unjustified potential overuse of exclusion CEFs in mental disorder trials. Then they discussed the limitations in current exclusion criteria designs and made recommendations for achieving more patient-centered exclusion criteria definitions.
  - He et al. [14] developed a method for profiling the collective populations targeted for recruitment by multiple clinical studies addressing the same medical condition using one eligibility feature each time. Using a previously published database COMPACT as the backend, they designed a scalable method for visual aggregate analysis of clinical trial eligibility features. This method consists of four modules for eligibility feature frequency analysis, query builder, distribution analysis, and visualization, respectively. This method is capable of analyzing (1) frequently used qualitative and quantitative features for recruiting subjects for a selected medical condition, (2) distribution of study enrollment on consecutive value points or value intervals of each quantitative feature, and (3) distribution of studies on the boundary values, permissible value ranges, and value range widths of each feature.

D. Longitudinal predictions on ICU data

Mentor: Quan Guo ⟨qguo48@gatech.edu⟩

Accurate knowledge of a patient's disease state and trajectory is critical in a clinical setting. Modern electronic healthcare records contain an increasingly large amount of

[3]https://github.com/bowenliu16/rl_graph_generation and https://github.com/google-research/google-research/tree/master/mol_dqn

data, and the ability to automatically identify the factors that influence patient outcomes stand to greatly improve the efficiency and quality of care. The goal of this project might be to repeat and improve previous study or to propose a new study using the publicly available Medical Information Mart for Intensive Care (MIMIC-III) database.

Dataset. MIMIC III (Submit your data access request to MIT via above link using your GT-Email and MIMIC completion report which lists modules taken with dates and scores. Request is made per person, not per team) has timestamps which tell when variables were measured or an event happened. These variables include but not limit to vital signs (heart rate, blood pressure, respiratory rate, SpO2, etc), lab results and medications; events include prescriptions, procedures, start/end of interventions.

MIMIC III Waveform Subset Matched provides additional physiologic signals (waveforms) and continous vital signs (numerics). Waveforms include one or more ECG signals, arterial blood pressure (ABP) waveforms, fingertip photoplethysmogram (PPG) signals, etc. This matched subset can be aligned with the above MIMIC III clinical database using the patient subject_id.

By using this temporal data with RNN (or its variants - GRU, LSTM), you can make time series predictions for early detection of disease onset, in-hospital mortality or forecast length of stay etc. You must present detailed steps such as the prediction target, feature selection, feature construction, predictive model and performance evaluation. You may initially start with a small subset of data as you develop your model locally. However, after fine-tuning it, your final paper must be based on results from the entire data.

1) MIMIC-III Benchmark Tasks: Hrayr et al. [13] recently introduced four clinical prediction benchmarks using the MIMIC III data: in-hospital mortality, decompensation, length of stay, and phenotype. You can focus on any of these benchmark tasks, and extend the current solution by either proposing any new methods or enriching the current benchmark dataset. For the latter extension, You can add more clinical variables (features) that have not been included in the dataset yet such as medications, infusions, treatments and waveforms. After you modify the dataset, you will need to show how the performance of baselines for each task is changed also. Therefore, the goal of this project will be fully explored variable study supported by some literature survey and utilizing big data analytics with supporting results from each predictive modeling or classification task.

2) Other ideas: You can try to come up with not only the above predictions, but also any other potentially beneficial benchmark tasks. Then, the goal of the project is to construct cohort and build some baseline models for the task which can enlarge the current benchmark set.

- **Main Reference**: *Harutyunyan, Hrayr, et al. Multitask learning and benchmarking with clinical time series data. Scientific data 6.1 (2019):96* [13] (Code)
- **Related Work**:
  - Ghassemi et al. [9] examined the use of latent variable models (viz. Latent Dirichlet Allocation) to decompose free-text hospital notes into meaningful features, and the predictive power of these features for patient mortality. This work considered three prediction regimes: (1) baseline prediction, (2) dynamic (time-varying) outcome prediction, and (3) retrospective outcome prediction. In each, their prediction task differs from the familiar time-varying situation whereby data accumulates; since fewer patients have long ICU stays, as they move forward in time fewer patients are available and the prediction task becomes increasingly difficult. Note that MIMIC-II (not III) was used in this work.
  - Xu et al. [37] explored a richer dataset, in which physiological signals including Electrocardiogram (ECG), Photoplethysmography (PPG), vital signs and so on were continuously recorded along with the discrete clinical data. They proposed an attention-based RNN model that can efficiently encode the long-term multi-channel dense signals and predict mortality and length of stay in real time. The dataset used in the work is the recently released MIMIC-III Waveform Database Matched Subset.
  - You can have a look at the implementation of Doctor AI paper [5]. Github details are in the paper.

E. Sepsis Prediction

Mentor: Yanbo ⟨yxu465@gatech.edu⟩

Sepsis is a leading cause of death in the United States, with mortality highest among patients who develop septic shock. Early aggressive treatment decreases morbidity and mortality. While general-purpose illness severity scoring systems are useful for predicting general deterioration or mortality, they typically cannot distinguish with high sensitivity and specificity which patients are at highest risk of developing specific acute condition.

While the general pipeline might be similar to those required in other topics such as Section IV-D, you may need to identify the onset of sepsis/septic shock (or any other task you are interested in) to label the data. A useful toolkit can be found here (code from [21])

- **Dataset**: MIMIC III See the instruction in the ICU prediction tasks topic above (Section IV-D).
- **Related Work**:
  - Johnson et al. [21] demonstrated that administrative and physiology based approaches result in cohorts of severely ill patients with variable outcome frequencies. In addition, it was shown that the Sepsis-3 criteria, among the methods assessed in the paper, identified the largest and the healthiest cohort.
  - Henry et al. [15] analyzed routinely available physiological and laboratory data from intensive care unit patients and developed "TREWScore," a targeted real-time early warning score that predicts which patients will develop septic shock.
  - Desautels et al. [7] applied a newly proposed definition for sepsis, Sepsis-3, as a gold standard for the implementation of their predictive algorithm, InSight, a machine learning classification system that uses multivariable combinations of easily obtained patient data (vitals, peripheral capillary oxygen saturation, Glasgow Coma Score, and

age), to predict sepsis using MIMIC-III dataset, restricted to intensive care unit (ICU) patients aged 15 years or more. Following the Sepsis-3 definitions of the sepsis syndrome, they compared the classification performance of InSight versus quick sequential organ failure assessment (qSOFA), modified early warning score (MEWS), systemic inflammatory response syndrome (SIRS), simplified acute physiology score (SAPS) II, and sequential organ failure assessment (SOFA) to determine whether or not patients will become septic at a fixed period of time before onset.
- Some references from other topics might be useful also. For example, Xu et al. [37] proposed an attention-based RNN model for modeling long-term EHR data and predict mortality and length of stay in real time. Similarly, one may utilize a RNN type of model to predict sepsis/septic shock based on patient's history, and then add attention mechanism to interpret the results.

F. Learning from Sleep Data

Mentor: Sowmya ⟨svenkatachari6@gatech.edu⟩, Siddharth Sridhar ⟨ssridhar78@gatech.edu⟩

Physiological signals acquired during sleep, like electroencephalography (EEG), can be used to computationally determine sleep apnea, cardiovascular disorders, sleep stage annotations, etc. There are many benefits which can be attained from these tasks. For instance, using a single channel sleep stage detector patients can be monitored at home using wearable EEG acquiring devices.

- **Dataset**: You are free to use any public dataset. These are just 3 examples:
  - Sleep Heart Health Study (you have to request data access),
  - PhysioNet: The Sleep-EDF database [Expanded] (smaller, but immediately downloadable)
  - ISRUC-SLEEP Dataset (smaller, but immediately downloadable)
- **Related Work**:
  - Tsinalis et al. [33] used convolutional neural networks (CNNs) for automatic sleep stage scoring based on single-channel electroencephalography (EEG) to learn task-specific filters for classification without using prior domain knowledge. Automatic Sleep Stage Scoring with Single-Channel EEG Using Convolutional Neural Networks.
  - Biswal et al. [3] proposes a method to automatically annotate sleep stages from EEG data which is collected from overnight sleep studies. There are 5 different sleep stages N1,N2,N3,REM, Wake and usually trained clinicians annotate sleep EEG to identify these sleep stages. However, this is a very time consuming and labor intensive task. Therefore, the authors describe a system which uses expert defined features with recurrent neural network to annotate an entire sleep study. The results are presented to clinician using web based visualization which can be used to annotate mistakes made the model to further improve the results. They compared the result with other methods such as convolutional neural network, etc.
  - Zhao et al. [39] introduce a predictive model that combines convolutional and recurrent neural networks to extract sleep-specific subject invariant features from RF signals and capture the temporal progression of sleep by using radio measurements without any attached sensors on subjects. They applied a modified adversarial training regime that discards extraneous information specific to individuals or measurement conditions, while retaining all information relevant to the predictive task.
  - Zhao et al. [25] introduce a predictive model that combines convolutional and LSTM neural networks to achieve a high F1 score for Automated Polysomnography (PSG) scoring.
  - Al-Hussaini et al. [2] used prototypes for interpretable automatic sleep stage scoring using multi-channel physiological signals. The interpretation is derived from rules used by human experts in the field.

G. Unsupervised Phenotyping via Tensor Factorization

Mentor: Xinze Wang (xwang992@gatech.edu)

The goal of this project is to design a new or existing tensor factorization method to extract meaningful phenotypes from raw and noisy electornic health records and apply the factorization model outputs to do prediction task.

*1) **Data Set: Centers for Medicare and Medicaid (CMS)**:* [4] The data set is CMS 2008-2010 Data Entrepreneurs' Synthetic Public Use File (DE-SynPUF). The goal of CMS data set is to provide a set of realistic data by protecting the privacy of Medicare beneficiaries by using 5% of real data to synthetically construct the whole dataset.

*2) **Related Work**:* Marble [16] is a sparse non-negative tensor factorization approach which deals with count data. Marble decomposes the tensor into two parts: a bias tensor and an interaction tensor. The bias tensor represents the common characteristics among all patients, while the interaction tensor decomposes the tensor into R different phenotypes. Marble fits a Poisson distribution to the count data and minimizes the KL divergence. Moreover, to improve the interpretability of extracted phenotypes, Marble imposes sparsity constraints by zeroing out the values smaller than a threshold.

Rubik [35] uses the Alternating Direction Method of Multipliers (ADMM) [4] algorithm to deal with binary tensors and minimize the Euclidean distance of the real and predicted tensors. Rubik, like Marble, decomposes the tensor into bias and interaction parts. Phenotypes extracted by Rubik are often more meaningful than Marble for two reasons: 1) **Orthogonality Constraint:** Rubik incorporates orthogonality constraints into the problem. Orthogonality can help produce more distinct phenotypes by reducing the overlapping components. 2) **Guidance Knowledge:** Rubik incorporates some predefined knowledge to the objective function to improve the interpretability of phenotypes.

[4] https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE_Syn_PUF.html

In [12], the authors proposed a PARAFAC2 model which produces unique solutions for dense data sets. Unique solutions ensure that the pursued solution is not an arbitrarily rotated version of the actual latent factors [12]. Recently, SPARTan was proposed for PARAFAC2 modeling on large and sparse data [29]. A specialized Matricized-Tensor-Times-Khatri-Rao-Product (MTTKRP) was designed to efficiently decompose the tensor both in terms of speed and memory. Finally, in [1], the authors proposed COPA, a constrained PARAFAC2 model which applies non-negativity, smoothness, and sparsity constraints to the resulting factor matrices.

H. COVID-19 X-ray Classification

Mentor: Artur Bessa Cabral (acabral3@gatech.edu), Ardavan (Ari) Afshar (aafshar8@gatech.edu)

X-ray image classification is a very important task for clinical tasks. In the recent years, there have been X-ray image datasets released for training machine learning models. It has been shown that COVID-19 can be detected from X-ray images. Chest X-rays are the first choice in terms of the initial imaging test when caring for patients with suspected COVID-19. Chest X-rays are the preferred initial imaging modality when pneumonia is suspected and the radiation dose of CXR (0.02 mSv for a PA film) is lower than the radiation dose of chest CT scans (7 mSv), putting the patients less at risk of radiation-related diseases such as cancer. Thus COVID-19 diagnostic plays an important role in clinical workflows for helping clinicians perform more important tasks.

In this task, the goal is to develop models for automatic COVID-19 detection using chest X-ray images. The trained model developed should be able to provide accurate diagnostics for binary classification (COVID vs. No-Findings) and multi-class classification (COVID vs. No-Findings vs. Pneumonia) settings.

- **Dataset**: You are free to use below public datasets:
  - COVID-Chest-xray-dataset: This dataset contains 542 frontal chest X-ray images from 262 people from 26 countries. This also contains clinical attributes about survival, ICU stay, intubation events, blood tests, location, as well as freeform clinical notes for each image/case.
  - COVID-19 dataset: This dataset contains COVID-19, no-findings and pneumonia images. This dataset is a combination of the data from multiple sources. This contains 127 COVID-19 images collected from COVID-Chest-xray-dataset and Normal and pneumonia images are collected from ChestX-ray8 database.

  There are some other sources which could be checked if more images/data were added to the list such as public databases on websites such as
  - Radiopaedia.org: the Italian Society of Medical and Interventional Radiology.
  - Figure1.com.
  - BIMCV-COVID19+: a large dataset with chest X-ray images CXR (CR, DX) and computed tomography (CT) imaging of COVID-19 patients. This also contains pathologies, polymerase chain reaction (PCR), immunoglobulin G (IgG) and immunoglobulin M (IgM) diagnostic antibody tests and radiographic reports from Medical Imaging Databank in Valencian Region Medical Image Bank (BIMCV).
- **Tasks**:
  - Binary classification: In this binary classification settings, the x-ray images can be classified as COVID vs. No-Findings images. This binary classification settings allows models to use x-ray images collected from other sources which contains normal, pneumonia and other indications.
  - Multi-class classification: Similarly in this classficiation settings, models can be developed for COVID vs. No-Findings vs. Pneumonia. As Pneumonia x-ray images look similar to COVID x-ray images, it is important to be able distinguish between these two classes. This multi-class classification settings can be further extended to other known x-ray such as Atelectasis, Cardiomegaly, Consolidation, Edema, Enlarged Cardiomediastinum, Fracture, Lung Lesion, Lung Opacity, Pleural Effusion, Pneumonia, Pneumothorax, Pleural Other, Support Devices, etc.. These labels are available in MIMIC-CXR dataset.

  If you have any other ideas, please make sure to contact Ming(mliu302@gatech.edu) for approval in order to verify the validity of idea/dataset/approach before diving into too much work.

V. Conclusion

Best of luck on your project and data science rocks!

Acknowledgment

Thanks all the TAs for their time and efforts in creating the course material together. Thank all the students for their dedication and feedback.

References

[1] A. Afshar, I. Perros, E. E. Papalexakis, E. Searles, J. Ho, and J. Sun. Copa: Constrained parafac2 for sparse & large datasets. *arXiv preprint arXiv:1803.04572*, 2018.

[2] I. Al-Hussaini, C. Xiao, M. B. Westover, and J. Sun. Sleeper: interpretable sleep staging via prototypes from expert rules. In *Machine Learning for Healthcare Conference*, pages 721–739, 2019.

[3] S. Biswal, J. Kulas, H. Sun, B. Goparaju, M. Brandon Westover, M. T. Bianchi, and J. Sun. SLEEPNET: Automated sleep staging system via deep learning. 26 July 2017.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[5] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun. Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine Learning for Healthcare Conference*, pages 301–318, 2016.

[6] H. Dai, Y. Tian, B. Dai, S. Skiena, and L. Song. Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*, 2018.

[7] T. Desautels, J. Calvert, J. Hoffman, M. Jay, Y. Kerem, L. Shieh, D. Shimabukuro, U. Chettipally, M. D. Feldman, C. Barton, D. J. Wales, and R. Das. Prediction of sepsis in the intensive care unit with minimal electronic health record data: A machine learning approach. *JMIR Med Inform*, 4(3):e28, 30 Sept. 2016.

[8] Z. Ge, D. Mahapatra, S. Sedai, R. Garnavi, and R. Chakravorty. Chest x-rays classification: A multi-label and fine-grained problem. *CoRR*, abs/1807.07247, 2018.

[9] M. Ghassemi, T. Naumann, F. Doshi-Velez, N. Brimmer, R. Joshi, A. Rumshisky, and P. Szolovits. Unfolding Physiological State: Mortality Modelling in Intensive Care Units. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 75–84, New York, NY, USA, 2014. ACM.

[10] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

[11] Q. Guan and Y. Huang. Multi-label chest x-ray image classification via category-wise residual attention learning. *Pattern Recognition Letters*, 2018.

[12] R. A. Harshman. PARAFAC2: Mathematical and technical notes. *UCLA Working Papers in Phonetics*, 22:30–44, 1972b.

[13] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. Ver Steeg, and A. Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):96, 2019.

[14] Z. He, S. Carini, I. Sim, and C. Weng. Visual aggregate analysis of eligibility features of clinical trials. *Journal of biomedical informatics*, 54:241–255, 2015.

[15] K. E. Henry, D. N. Hager, P. J. Pronovost, and S. Saria. A targeted real-time early warning score (trewscore) for septic shock. *Science translational medicine*, 7(299):299ra122–299ra122, 2015.

[16] J. C. Ho, J. Ghosh, and J. Sun. Marble: high-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 115–124. ACM, 2014.

[17] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *arXiv preprint arXiv:1901.07031*, 2019.

[18] J. D. J. S. J. E. James Mullenbach, Sarah Wiegreffe. Explainable prediction of medical codes from clinical text. *ACL Anthology*, 2018.

[19] W. Jin, R. Barzilay, and T. Jaakkola. Junction tree variational autoencoder for molecular graph generation. *ICML*, 2018.

[20] W. Jin, K. Yang, R. Barzilay, and T. Jaakkola. Learning multimodal graph-to-graph translation for molecular optimization. *ICLR*, 2019.

[21] A. E. Johnson, J. Aboab, J. D. Raffa, T. J. Pollard, R. O. Deliberato, L. A. Celi, and D. J. Stone. A comparative analysis of sepsis identification methods in an electronic database. *Critical care medicine*, 46(4):494–499, 2018.

[22] T. Kang, S. Zhang, Y. Tang, G. W. Hruby, A. Rusanov, N. Elhadad, and C. Weng. Eliie: An open-source information extraction system for clinical trial eligibility criteria. *Journal of the American Medical Informatics Association*, 24(6):1062–1071, 2017.

[23] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato. Grammar variational autoencoder. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1945–1954. JMLR. org, 2017.

[24] Z. Li, C. Wang, M. Han, Y. Xue, W. Wei, L.-J. Li, and L. Fei-Fei. Thoracic disease identification and localization with limited supervision. *CVPR*, 2018.

[25] R. U. D. K. Linda Zhang, Daniel Fabbri. Automated sleep stage scoring of the sleep heart health study using deep neural networks. 2019.

[26] H. Liu, L. Wang, Y. Nan, F. Jin, and J. Pu. Sdfn: Segmentation-based deep fusion network for thoracic disease classification in chest x-ray images. *arXiv preprint arXiv:1810.12959*, 2018.

[27] H. Ma and C. Weng. Identification of questionable exclusion criteria in mental disorder clinical trials using a medical encyclopedia. In *Biocomputing 2016: Proceedings of the Pacific Symposium*, pages 219–230. World Scientific, 2016.

[28] H. Ma and C. Weng. Prediction of black box warning by mining patterns of convergent focus shift in clinical trial study populations using linked public data. *Journal of biomedical informatics*, 60:132–144, 2016.

[29] I. Perros, E. E. Papalexakis, F. Wang, R. Vuduc, E. Searles, M. Thompson, and J. Sun. Spartan: Scalable parafac2 for large & sparse data. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 375–384. ACM, 2017.

[30] P. G. Polishchuk, T. I. Madzhidov, and A. Varnek. Estimation of the size of drug-like chemical space based on gdb-17 data. *Journal of computer-aided molecular design*, 27(8):675–679, 2013.

[31] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, M. P. Lungren, and A. Y. Ng. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

[32] T. Sterling and J. J. Irwin. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11):2324–2337, 2015.

[33] O. Tsinalis, P. M. Matthews, Y. Guo, and S. Zafeiriou. Automatic sleep stage scoring with single-channel eeg using convolutional neural networks. *arXiv preprint arXiv:1610.01683*, 2016.

[34] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3462–3471. IEEE, 2017.

[35] Y. Wang, R. Chen, J. Ghosh, J. C. Denny, A. Kho, Y. Chen, B. A. Malin, and J. Sun. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1265–1274. ACM, 2015.

[36] D. Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

[37] Y. Xu, S. Biswal, S. R. Deshpande, K. O. Maher, and J. Sun. Raim: Recurrent attentive and intensive model of multimodal patient monitoring data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2565–2573. ACM, 2018.

[38] J. You, B. Liu, R. Ying, V. Pande, and J. Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. In *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 6412–6422, USA, 2018. Curran Associates Inc.

[39] M. Zhao, S. Yue, D. Katabi, T. S. Jaakkola, and M. T. Bianchi. Learning sleep stages from radio signals: A conditional adversarial architecture. In *International Conference on Machine Learning*, pages 4100–4109, 2017.

[40] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, and P. Riley. Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):10752, 2019.