PROJECT INFORMATION

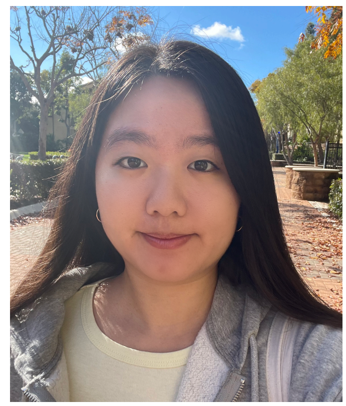
Project Title

(Try to choose a catchy title. Max 20 words).

CHOC: ATOMAC Guideline-based Algorithm

Team Information

Team Name: CHOC **Team # on Canvas: 13**	**TEAM CHOC**
Team member 1 (Team Lead) *(Last Name, name; student ID; UCI email; picture):* Yu, Kenny; 20618794; kennetgy@uci.edu	
Team member 2 Linture, Marcus; 54884520; mlinture@uci.edu	

Team member 3 *(Last Name, name: student ID: UCI email, picture):* Nguyen, Calvin; 7073295797; calvinn7@uci.edu	
Team member 4 *(Last Name, name: student ID: UCI email, picture):* Hu, Kelly; 77102740; kellyh12@uci.edu	
Team member 5 *(Last Name, name: student ID: UCI email, picture):* Leong, Evan; 81124108; leonger@uci.edu	

INTRODUCTION

Motivation/Background

(Describe the problem you want to solve and why it is important. Max 300 words).

There are many children who undergo accidents which lead to organ injuries that may involve their liver, kidneys, or pancreas for example. These organ injuries may often lead to internal or unstable bleeding among the suffering children. Children that suffer from these injuries tend to undergo surgery to halt the bleeding immediately based on guidelines that are referred to as the ATOMAC Pediatric SOI Guidelines to Assess Hemodynamic Stability. Clinical providers can gain a lot of insight and assistance into whether or not a child's bleeding requires immediate surgery and attention. If a child does not mark enough boxes from the guidelines and does not require surgery, clinical providers and doctors will deem it safe for the internal bleeding to safely go away on its own over time. Therefore, we want to make it possible for these medical providers to simulate these conditions so that they can determine whether or not a child should undergo surgery based on the ATOMAC guidelines.

State of the Art / Innovation

(Describe how the problem is solved today (if it is), and the innovation/advance of your project. Max 200 words).

Currently, there is research to recognize hemodynamic instability in patients using machine learning. A Hemodynamic Stability Index (HSI) was developed to be used for determination of risk for potential hemodynamic interventions. While this research is closely related to our problem, this was aimed for providing better bedside care in the intensive care unit for all hemodynamic instability issues. While that research was for more broad care, our focus in this project is dedicated to a specific injury to childrens' solid organs. The traditional way of determining hemodynamic stability for these injuries is dependent on the doctors' expertises and quick judgment, but our project will help give a faster prognosis to make life-saving decisions.

Project Goals

(Describe the project general goals. Max 200 words).

Our goal is to implement a functional machine learning simulation model to successfully determine when patients should be discharged, how critical of a condition they are undergoing is, and if the child requires surgery or not. Our simulation model will be built based on a Lagrangian Neural Network model and using the ATOMAC algorithm. We may look to build a front end interface for easy accessibility of our model if time permits. After implementation of this model, we will look to create a poster board displaying our findings and implementation in preparation for the ICS Project Expo held at the end of Spring quarter.

Assumptions

(Describe the assumptions (if any) you are making to solve the problem. Max 180 words).

We are assuming that the users of this model are doctors, primarily surgeons that work at CHOC, who are knowledgeable of medical terminologies and processes regarding child healthcare such as the ATOMAC guidelines. Additionally, it is assumed that the users have access to the necessary data that the model requires in order to function and output a result. This data includes but is not limited to assessments such as suspicion of bleeding, clinical signs of shock, grade of injury, and parameters such as Hemoglobin (HB), blood pressure (BP), and Heart Rate (HR).

SYSTEM ARCHITECTURE OVERVIEW

High Level Diagram

(Provide an overview of the entire proposed system, identifying the main components that would be developed/used for the product and their interfaces. Max 500 words or 1 page).

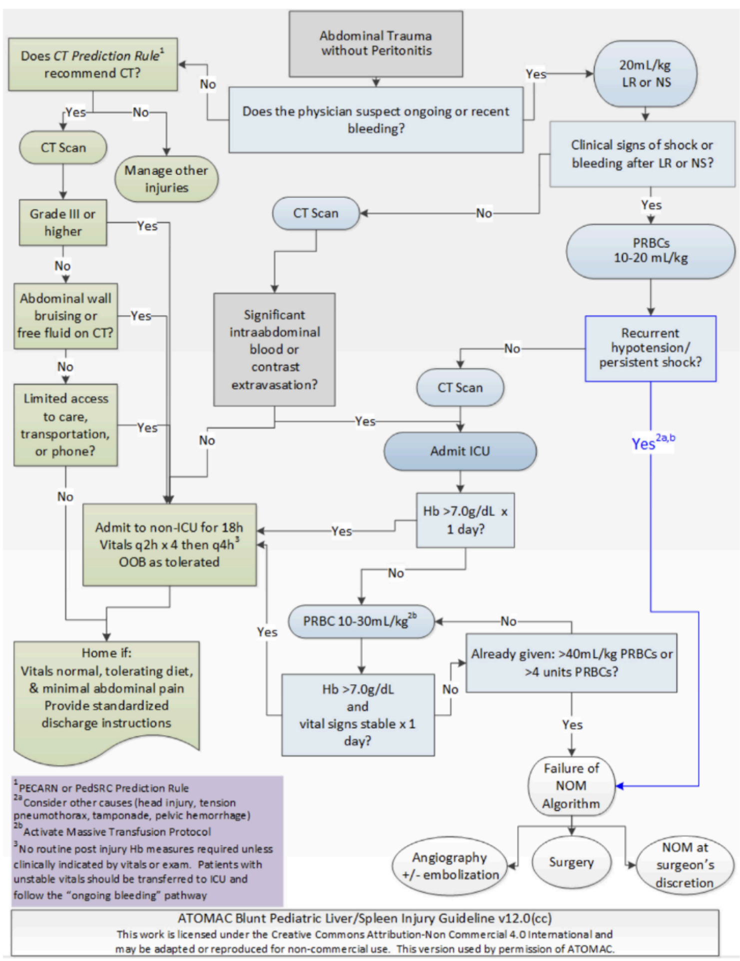


Figure 3. ATOMAC guideline v12.0 for management of blunt liver or spleen injury. LR=lactated ringers; NS=normal saline; PRBCs=packed red blood cells; CT = computed tomography; ICU= Intensive Care Unit; Hb=hemoglobin; NPO=nothing by mouth; q6h=every 6 hours; NOM=nonoperative management; ICU=intensive care unit; q2h=every 2 hours. PECARN=Pediatric Emergency Care Applied Research Network; PedSRC = Pediatric Surgery Research Consortium; For Abdominal CT prediction rules, see (1) Streck CJ, Vogel AM, Zhang J, et al 2017 and (2) Wisner DH, Kuppermann N, Cooper A, et al.2015. Algorithm used by permission of ATOMAC.

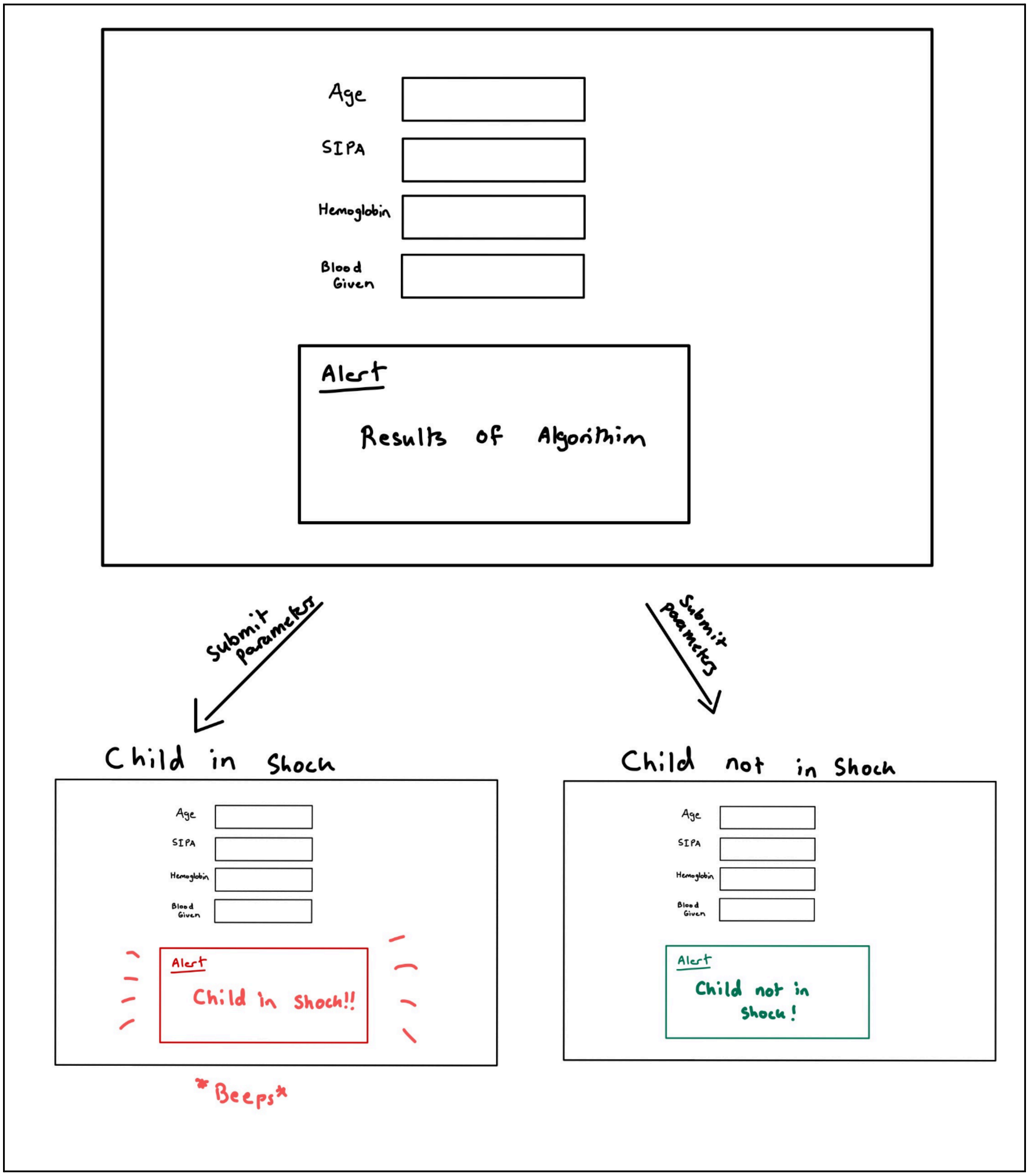
The figure above (boxes in blue) depicts the decision to make using a neural network. The neural network is connected to a database to be trained with. The input and output will be linked to the client's side.

User Interaction and Design

(Provide an overview of the main interfaces and their design for users' interaction with the proposed product. Max 500 words or 1 page).

Web Interface using Flask, Python, HTML and CSS

This will be a working diagram since we don't know exact specifications of the UI.



REQUIREMENTS

User Stories and Use Cases

(Describe functional and non-functional requirements of your project by describing user stories and use cases. Please add at least a total of 10 entries (stories or/and cases). Max 1,000 words or 2 pages).

User Stories

1. As a user, I can enter data, so that I can use the algorithm.
    a. Enter data: Given some parameters (age, blood pressure, etc.), when inputted into the system, then the algorithm will display whether or not the patient is in recurring shock. (test1=user manually inputs data into system, and the algorithm is able to read and return info) (test2=user enters data, but is missing some, but algorithm should run as normal) (test3=user feeds in a stream of data, and algorithm runs as normal)

2. As a pseudodevice, I can alert the user when the patient is suffering from persistent shock, then the user can handle it.
    a. Alert user: Given a feed of parameters, when parameters are not within thresholds, then the pseudodevice will beep and tell the user to take action. (test1=algorithm uses data correctly returns shock/no shock to the user) (test2= algorithm uses data and returns any message to user)

3. As a developer, I can set the threshold for certain parameters, so that the algorithm can follow it for its use.
    a. Set threshold: Given a database, when fed into a neural network, then it can determine the cut points for each parameter. (test1=missing data still returns reasonable thresholds)

4. As a neural network, I can find cut points, so that I am able to differentiate between the severity of injuries.
    a. Find cut points: Given a dataset, when developed, then I can find cut points and differentiate between severity of injuries.

5. As a developer, I can understand the SIPA formula, so that I can use it as a parameter to the algorithm.
    a. Create parameter: Given a formula for SIPA, when understood, then the developer can create the parameter which is essential for the algorithm.
    b. SIPA: Shock Index Pediatric Adjusted

6. As a pediatrician conducting routine check-ups, I can use the device to notify me if a patient's vital signs indicate potential shock, so that I can investigate further and adjust their treatment plan accordingly.
    a. Given a pediatric patient with chronic illness, when their vital signs suggest shock during a routine check-up, then alert the attending physician.

Use Cases

1. Manually input data
    a. Actors: Doctor, ATOMAC Algorithm, Neural Network w/ Database, Interface System
    b. Precondition: Doctor chooses to input manually, Doctor has the data they need to input, and NN is trained
    c. Flow of Events (Basic Path):
        i. Doctor enters parameters into input boxes (age, blood pressure, heart rate, etc.)
        ii. Interface system retrieves manually inputted data.
        iii. Interface system delivers data to algorithm.
        iv. Algorithm uses pre-established trained neural network with new data.

      v. Algorithm returns that the patient is in shock to the Interface System.
      vi. Doctor receives the alert and takes action.
   d. Flow of Events (Alternative Paths):
      i. If some input values are empty, proceed as normal.
      ii. If the patient is not in shock, return that signal accordingly.
   e. Postcondition: Inputted data is erased and reset for the next Doctor's use.

2. Feeding in constant data
   a. Actors: Doctor, Pseudodevice, Data Provider, ATOMAC Algorithm, Neural Network w/ Database, Interface System
   b. Precondition: Doctor chooses to constantly feed data, Doctor has the data they need to input, NN is trained
   c. Flow of Events (Basic Path):
      i. Doctor starts feed of data.
      ii. Interface system delivers data to Algorithm.
      iii. Algorithm uses pre-established trained neural network with new data.
      iv. Algorithm periodically returns updates on the patient's health.
      v. Algorithm signals that the patient is in shock to the Interface System.
      vi. Doctor receives the alert and takes action.
   d. Flow of Events (Alternative Paths):
      i. Even if there is missing data, the algorithm persists.
      ii. If the patient never reaches the shock index, then the algorithm won't alert the doctor.
      iii. The doctor can stop the feed of data whenever.
   e. Postcondition: Data is erased from system and reset for the next Doctor's use.

3. Resource Optimization in Pediatric Trauma Bay
   a. Actors: Hospital, Doctor, Pseudodevice, Data Provider, ATOMAC Algorithm, Neural Network w/ Database, Interface System
   b. Precondition: Hospital is very busy and needs to allocate resources based on patient need.
   c. Flow of Events (Basic Path):
      i. Patient info is given to and monitored by the system.
      ii. Interface system delivers data to Algorithm.
      iii. Algorithm uses pre-established trained neural network with new data.
      iv. Algorithm constantly monitors patient health.
      v. Algorithm signals that the patient is in shock to the Interface System.
      vi. Hospital workers receive the alert and dedicate the proper resources to the most critical patients.
   d. Flow of Events (Alternative Paths):
      i. Even if there is missing data, the algorithm persists.
      ii. If the patient never reaches the shock index, then the algorithm won't alert the doctor.
      iii. If the hospital is able to provide thorough care for all patients the doctor can take over manually.
   e. Postcondition: Patient is adequately cared for based on their current need.

4. External Use (Outside of Trauma Bay)
   a. Actors: Emergency Responders (EMTs), Pseudodevice, Data Provider, ATOMAC Algorithm, Neural Network w/ Database, Interface System
   b. Precondition: EMTs are responding to a pediatric trauma emergency.
   c. Flow of Events (Basic Path):
      i. Patient info is given to and monitored by the system by EMT.
      ii. Interface system delivers data to Algorithm.
      iii. Algorithm uses pre-established trained neural network with new data.
      iv. Algorithm constantly monitors patient health.
      v. Algorithm signals that the patient is in shock to the Interface System.

vi. EMT is altered by the device and prioritizes patient transport to the hospital.

d. Flow of Events (Alternative Paths):
   i. Even if there is missing data, the algorithm persists.
   ii. If the patient never reaches the shock index, then the algorithm won't alert the EMT and judgment can be made accordingly.
   iii. If there are technical issues with the device or missing data inputs, the EMT relies on clinical judgment and traditional assessment methods.

e. Postcondition: Patient is given necessary response for their condition.

Coding and Testing

(Include links to your code (GitHub commits/issues) and describe testing and testing metrics for your project. Max 500 words or 1 page).

GitHub
https://github.com/EvLeong1/CS180A_CHOC

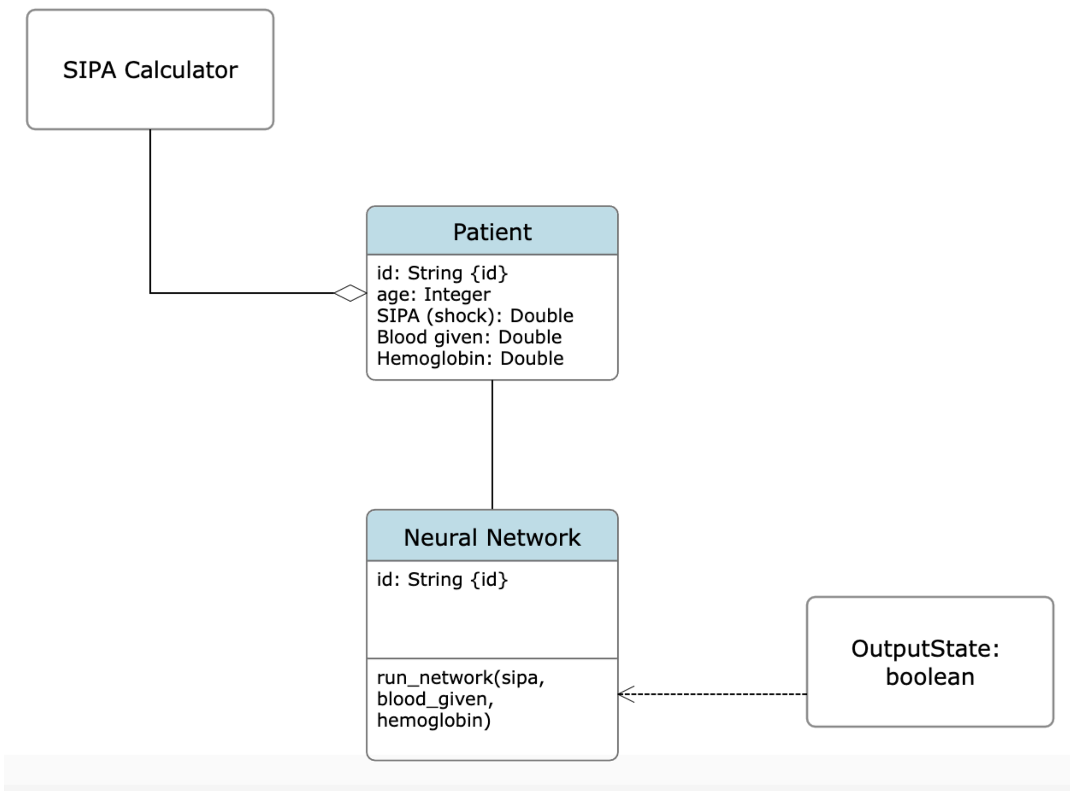
Jira (Sprint and Issue Tracking)
https://choc180a.atlassian.net/jira/software/projects/CCT/boards/1

Confluence
https://choc180a.atlassian.net/wiki/spaces/CCT/overview

A few testing methods that we will be utilizing involve the following but is not limited to whitebox testing, blackbox testing, regression testing, and branch testing. As part of the DevOps process workflow, we will also implement a CI/CD pipeline (continuous integration and continuous delivery/deployment) in order to ensure the most up to date code in github. With a GitHub or Jenkins CI/CD pipeline in place, our developers will be able to work on the same source code in an agile, timely, and effective manner without the worries of a build or run breaking. We will also write manual unit tests within our code to minimize our logical errors in code with assert statements and try/catch/throw exceptions.

SYSTEM MODELS

(Provide a visual design of the solution by showing contexts, sequences and behavioral sates using The Unified Modeling Language (UML). Max 1,000 words or 2 pages).



APPENDICES

(List/describe the platforms and technologies you plan to use (or currently using) to develop the solution). Max 200 words).

We will use Python to implement the Neural Network using PyTorch and Keras libraries. Additionally, we will implement a web interface using Flask, HTML, and Tailwind/CSS.