Machine Learning for Understanding Aging

Design Document

sdmay20-41
Dr. Julie Dickerson
Ian Simon - Chief Engineer
Jacob Laing - Chief Engineer
Nathan Carter - Test Engineer
Samantha Williams - Meeting Scribe
Scott Rose - Meeting Facilitator
Thomas (Aria) Sheets - Report Manager
sdmay20-41@iastate.edu
http://sdmay20-41.sd.ece.iastate.edu/

6 October 2019 / Version 1

Executive Summary

Development Standards & Practices Used

1. Our product must ensure the privacy of the people whose data we are using to use in the creation of our machine-learning program.
    a. All Personal Health Information (PHI) must be anonymized.
    b. Any PHI that is transmitted must be encrypted.
2. Our product must be accessible, and the information output must be easily understandable.
3. Our product must be fast to learn, and up to today's standards of machine learning.
4. Our product must output results that are accurate so that others can use the data we obtain through our program with reliability.

Summary of Requirements

Functional Requirements:

- Program accurately assesses patterns in data related to aging.
- User can continue to input data to increase the accuracy of the program.
- Program outputs aspects of the input data that affects aging
- Projects completed by May 2020

Non-Functional Requirements:

- Written in Python
- Clear, well-documented code.
- Privacy of subjects included in test data is considered.
- Appropriate size of training data is used to properly train the program.
- The results of the running program are outputted in a user friendly format.

Applicable Courses from Iowa State University Curriculum

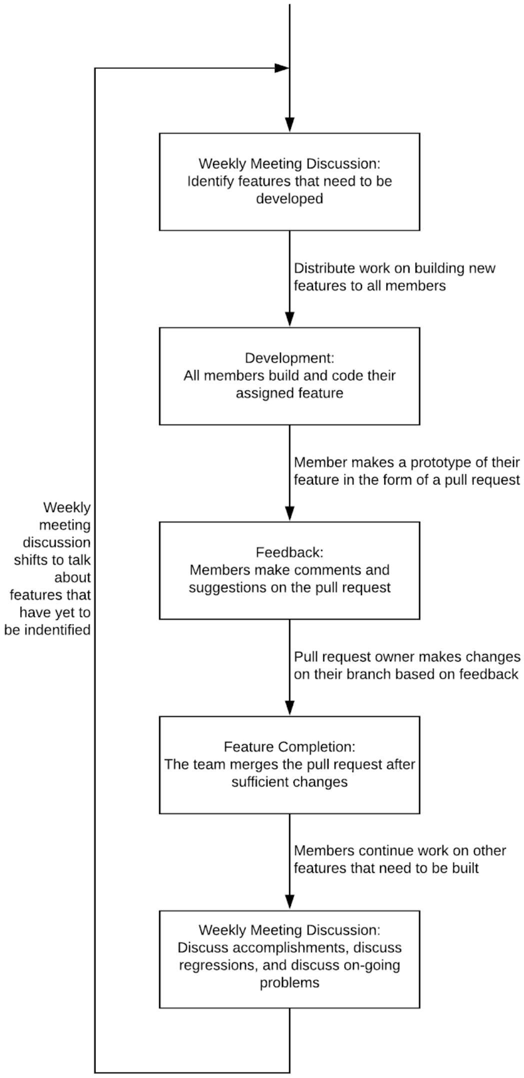
COM S 227:	Object-oriented Programming
COM S 228:	Introduction to Data Structures
COM S 311:	Introduction to the Design and Analysis of Algorithms
COM S 474:	Introduction to Machine Learning
MATH 207:	Matrices and Linear Algebra
S E 309:	Software Development Practices
S E 329:	Software Project Management
S E 339:	Software Architecture and Design

New Skills/Knowledge acquired that was not taught in courses

- Neural network design.
- Cost functions used in neural networks.
- Analyzing large data sets.
- Managing privacy of data used programs for analyzing datasets.

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)


Weekly Meeting Discussion:
Identify features that need to be developed
Distribute work on building new features to all members
Development:
All members build and code their assigned feature
Member makes a prototype of their feature in the form of a pull request
Feedback:
Members make comments and suggestions on the pull request
Pull request owner makes changes on their branch based on feedback
Feature Completion:
The team merges the pull request after sufficient changes
Members continue work on other features that need to be built
Weekly Meeting Discussion:
Discuss accomplishments, discuss regressions, and discuss on-going problems
Weekly meeting discussion shifts to talk about features that have yet to be indentified

1 Introduction

1.1 Acknowledgement

Thank you to Dr. Julie Dickerson for providing guidance and structure for this endeavor. Thank you to Inter-University Consortium for Political and Social Research (ICPSR) for providing the large data sets that were used in the training and creation of the program. Thank you to Iowa State University for allowing us to use their hardware to run and test our program.

1.2 Problem and Project Statement

Human aging is a topic that has been studied throughout history. Scientists, doctors, sociologists, and the general public all want to know what characteristics indicate a decline in health and what actions can be taken to slow down this decline. Knowing this information can allow people to increase their life expectancy and overall quality of life.

Using health data collected using sensors by ICPSR and research regarding machine-learning techniques, our product will find and return the various patterns found in the data. After training the program using large quantities of training data, it will be able to accept data and return information regarding actions in which the user can take that have proven effective for other patients with similar characteristics.

1.3 Operational Environment

Because our end-product requires mining a large set of given data to make the machine learn and output results, we recommend that our software is run on a high-end machine, preferably one with a high-end GPU. We will be developing our software expecting our users to be using a high-end machine. If they do not, the time for results to appear from our product will take longer.

1.4 Requirements

Functional Requirements:

- Program accurately assesses patterns in data related to aging.
- User can continue to input data to increase the accuracy of the program.
- Program outputs aspects of the input data that affects aging
- Projects completed by May 2020

Non-Functional Requirements:

- Written in Python
- Clear, well-documented code.
- Privacy of subjects included in test data is considered.
- Appropriate size of training data is used to properly train the program.
- The results of the running program are outputted in a user friendly format.

1.5 Intended Users and Uses

The users of our end product will mostly consist of sociology and psychology scientists. They will be using our product to analyze data in hopes of finding patterns related to aging.

1.6 Assumptions and Limitations

Assumptions

- The program will be used by scientists and researchers
- The program will be ran on high-end software
- The program will produce results that correlate body movement to age.

Limitations

- The software will only be used by scientists and researchers
- The budget is limited to hardware owned by the team members and hardware owned by Iowa State University.
- The data input into the program must fit a specific format.
- The program will only have English as the language.
- The program will rely heavily on GPU usage.
- The program needs to be completed by the beginning of May 2020

1.7 Expected End Product and Deliverables

Our first major deliverable would to get the data parser and data formatter set up. These are necessary parts to be completed before we can expect out program to learn data. Even though this is our first deliverable, it is expected that we can work on the machine learning module by using dummy or mock data until the data parser and formatter is set up. The expected delivery date for this is late January 2020.

Our second major deliverable would to get our database set up, linking it to our data formatter module and our machine learning module. The database would read in data from the formatter module, storing the formatting data into the database to reduce duplicate formatting. The machine learning module will then read in the data, but no learning will be implemented yet. The database will also need to support the future implementation of storing the results after the machine learning module is setup. The expected delivery date for this is mid-late February 2020.

Our third major deliverable is to create the machine learning module. This part is where all the machine learning takes place. This module reads in formatted data from the database and produces a result in the form of data from the learning process. This data is then sent and stored in our database, and it will be sent to a data visualizer when it gets developed. The expected delivery date for this is late March 2020 to early April 2020.

Our last major deliverable is to create the result data parser and visualizer modules. These are the final pieces of our program, and will allow scientists/researchers to visualize the data in a friendly format, as well as show the pure data results of learning. The expected delivery date for this is late April 2020.

In the end, we will have created a program that can receive and analyze large datasets concerning aging and its effects. The program would use machine learning to possibly view this data in a different light than conventional methods. Our clients will be provided with our findings, as well as a framework for using the trained program on new sets of data. The hope will be that our program can provide new insights into how aging affects the body and mind, and perhaps lead to new advances in how we confront the universal challenge of getting old.

2. Specifications and Analysis

2.1 Proposed Design

We will be using TensorFlow in Python to create an unsupervised machine learning algorithm. The program will take in data from the ACTIVE (Advanced Cognitive Training for Independent and Vital Elderly) study conducted at the University of Michigan which already anonymized data to protect the privacy of individuals. The program will determine outcomes of aging based on the data input into the program. The program will run on a GPU cluster provided by Iowa State University. The data will be produced in a readable format so that it can be presented to scientists interested in our findings.

Currently, we haven't created code or tested anything that is part of the project so far, we have only researched the building blocks required to understand how machine learning functions before diving in. This includes research on concepts like neural networks and back-propagation. We have watched simulations and videos discussing machine learning and have inspected and compiled machine learning code that has been shown to work.

2.2 Design Analysis

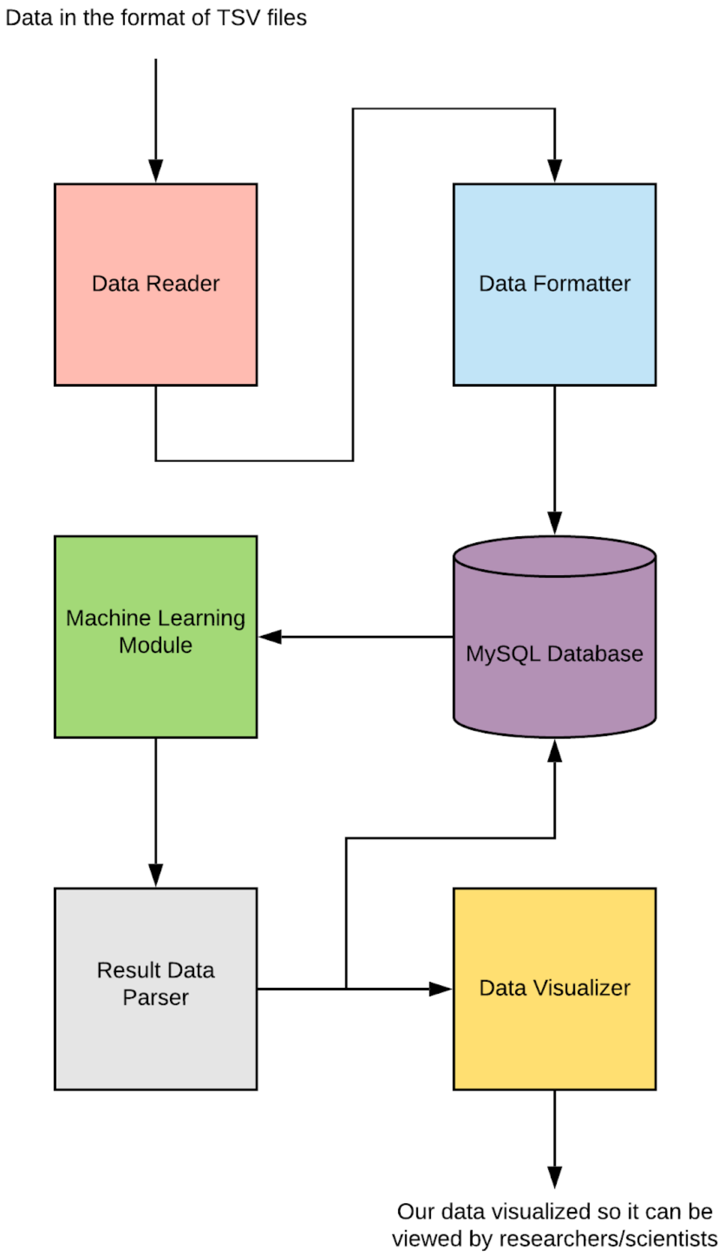
So far, we have done preliminary research into what methods we will be using to design our system. We have found that the TensorFlow library will best suit what we need. To continue with the project, we need to figure out a system to read in the data for cleaning. We will also need to do experiments with TensorFlow to become more familiar with the platform. We will also have to do more research into unsupervised learning because all of our research so far has been on supervised learning.

2.3 Development Process

For this project, we have decided to follow the Agile Development Process. The Agile Development Process is a very intuitive process and is well-known amongst everyone within our group making it great for our team to use. This process allows us to easily assign new tasks to members, monitor the tasks that are being worked on, and quickly fix any issues that may arise from any tasks. Along with those positives Agile allows us to quickly and efficiently make changes to our project as we may see fit.

2.4 Design Plan

Our design is created in a modular fashion where the data we get is put through a data reader, which then goes through a formatter so that we can parse through it much easier. This parsed and formatted data then gets stored in our database, in which then, our machine learning module will take and retrieve results from the input data. The result data will be parsed and stored in our database, as well as visualized in a friendly manner. The diagram below details the data flow and the components proposed.


Data in the format of TSV files
Data Reader
Data Formatter
Machine Learning Module
MySQL Database
Result Data Parser
Data Visualizer
Our data visualized so it can be viewed by researchers/scientists

3. Statement of Work

3.1 Previous Work And Literature

We will not be using previously created software, and we expect to use our own, with the exception of libraries and dependencies. There has been research on machine learning to study aging, but not using the data we will be using.

3.2 Technology Considerations

We plan on using TensorFlow which has many advantages and some disadvantages, especially when compared to its competitors:

Advantages of TensorFlow include its graph visualization, its library management, its tools for debugging, and scalability.

Disadvantages of TensorFlow include its lower computation speed and limited GPU support.

3.3 Task Decomposition

We expect to decompose each module into a series of tasks to complete, each with their own dependencies on other tasks.

3.4 Possible Risks And Risk Management

Our beginner's knowledge of machine learning and linear algebra affects some of our members. This is a risk that may impact the progress of our project. We're mitigating this risk by researching a lot on machine learning in our first semester.

3.5 Project Proposed Milestones and Evaluation Criteria

Each deliverable we have is considered a major milestone for our project. Our most important milestone will be finishing the machine learning module, as that is the main focus of our project.

Each module will be thoroughly unit tested to ensure it works.

3.6 Project Tracking Procedures

We will use both GitLab and Jira to keep track of our progress. We will use GitLab for the low-level details and code changes, and use Jira for the high-level details and card completion.

3.7 Expected Results and Validation

We expect we will have a fully-functioning program that can input data and output results that help understand the effects of aging. We will do extensive testing and communication with faculty members knowledgeable on the matter to confirm what we create is accurate.

4. Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

November 15, 2019: Inception

- Requirements are gathered for the project and research is done.

December 20, 2019: Design

- The project is fully conceptualized and tested.

January 25, 2020: Data Parser & Formatter

- We develop the data parser and formatter.

February 20, 2020: Database Storage

- We set up our database so that we can store formatted input and output data.

March 30, 2020: Machine Learning Module

- We create our machine learning module that reads in the formatted data from our database.

April 30, 2020: Data Visualizer & Completion

- We create the data visualizer that allows our results to be easily viewed by scientists and researchers.

4.2 Feasibility Assessment

We predict that it is highly likely that our project will be finished by the due date. The biggest challenge we must overcome include our beginner's knowledge on machine learning.

4.3 Personnel Effort Requirements

The duration of the task depends on the complexity of it. For a low-complexity task, we expect 1-2 hours to be spent. For a medium-complexity task, we expect 2-4 hours spent. For a high-complexity task, we expect 5+ hours and plenty of technical communication with other team members to finish the task.

4.4 Other Resource Requirements

The resources required will be our own personal machines, the data that we will use for our machine learning program, and the GPU clusters provided by Iowa State University.

4.5 Financial Requirements

The financial requirements for this project is small if any due to the free cost of the GPU clusters provided by Iowa State University and the use of our own personal machines.

5. Testing and Implementation

5.1 Interface Specifications

We expect to use Mockito and the unittest framework in Python for testing.

5.2 Hardware and software

The same hardware that we use to develop will also be used for testing.

5.3 Functional Testing

We expect to unit test each of our modules. We will then combine our modules and test the reliability and efficiency of the system as a whole. We will present it to our client to make sure it meets standards.

5.4 Non-Functional Testing

We will do this using the GPU clusters at Iowa State University to test performance. There will be assessments on algorithms used to ensure the lowest time-complexity. We will be using the program as we develop to ensure usability.

5.5 Process

We have not been able to test our process because we have not reached that part of the project yet.

5.6 Results

We have not obtained any results because we have not reached that part of the project yet.

6. Closing Material

6.1 Conclusion

We do not have any conclusion because we have not reached that part of the project yet.

6.2 References

There are no references yet at this point of the project.

6.3 Appendices

There is no additional information yet at this point of the project.