Requirements Document for HIVENet

Enrique Ferndandez, Alex Garcia, William Wodrich, Andrew Davis

CS 461 Oregon State University

Fall 2018

Abstract

Type here

CONTENTS

1 INTRODUCTION

This software requirement specification describes the scope of our project and an overview of the requirements this project will meet. A list of abbreviations and definitions is provided within this section.

1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the Hierarchical Information Variant Exchanging Network (HIVENet). This document will explain the purpose and development of the system as well as provide a high-level understanding of HIVENet's infrastructure, restraints, and interface.

1.2 Scope

The HIVENet facial recognition software application is designed to demonstrate a new method to optimize neural network training and illustrate methods to enable communication between neural networks. The tool will be built to interface with Google's Deep Learning API, TensorFlow™.

1.3 Glossary

- **HIVENet:** "Hierarchical Information Variant Exchanging Network." Used throughout the document to refer to the project application.
- **Edge-Device:** A computer wirelessly connected to the core-device. Equipped with a camera to receive video input.
- **Edge-Process:** The tandem structure of the artificial neural network and inference engine housed on an edge device.
- **Core-Device:** A central computer wirelessly connected to all edge-devices.
- **Core-Process:** The HIVENet application manager, housed on the core-device. It periodically receives updates from edge-proccesses and handles the variant exchange process.
- **Neural Network:** "NN." A computational framework that employs several learning algorithms to process complex data input. This framework is generally implemented as directed weighted graphs of artificial neurons.
- **Artificial Neurons:** The most basic component of a neural network. They mirror neurons in the brain, acting as aids in the overall decision process of NNs [1].
- **Coefficients/Weights:** The values inside of a NN that define and modifiy its behavior. They are associated with the artificial neurons and edges that connect them.
- **Training:** The act of providing a NN with a dataset to operate on.
- **Inference Engine:** TBD.
- **Fitness Level:** An enumeration of an edge-process' competency. As the NNs train themselves on a dataset, it will modify itself to increase accuracy, raising the fitness level.

1.4 References

[1] Eyal Reingold, "Artificial Neural Networks Technology," *University of Toronto Mississauga*, [Online]. Available: *http://www.psych.utoronto.ca/users/reingold/courses/ai/cache/neural_ToC.html*. [Accessed October 31 2018]

1.5 Overview

The following sections will provide an overview of HIVENet's infrastructure and system functionality. Chapter two will also mention user interaction and assumptions we have associated with our client. Finally, any constraints measure on the application will be covered.

2 OVERALL DESCRIPTION

This section will give a high-level HIVENet software infrastructure and front-end usability. This software will be explained in a matter of how the HIVENet infrastructure will be setup, how users will interact with HIVENet, and what data will be passed. By the end, assumptions and constraints for the system will be presented.

2.1 Product Perspective

This application will consist primarily of underlying code of hierarchical NN stack that will pass datasets, coefficients, and alter edge-devices inference engines that will allow for those same edge-devices to be able to identify the users that are using this produce across the entire network. The user will interact with an authentication GUI webpage.

2.2 Product Functions

HIVENet is designed to register and recognize any user who interacts with this application. It uses a hierarchical, convolutional NN that self-trains its edge-devices. Each edge-device will have an established communication channel to the centeral-device. This channel will be the primary communication to share NN coefficients and datasets to all other edge-devices. The central-device's functionality will be to take all end-devices and choose the one with the greatest fitness level. That node will then reconfigure the inference engines on edge-devices to further optimize the success rate of the facial recognition software.

2.3 User Characteristics

With HIVENet, the user will be able to register with an edge-device and create an account that will contain their first and last name. Users can register with physical facial characteristics as their form of identification. That same user will then be able to login into any other edge-devices.

2.4 Assumptions and Dependencies

Our facial recognition software will initially distinguish three facial features: eyebrows, eye color, and lips.

2.5 Constraints

2.6 Apportioning of Requirements

3 SPECIFIC REQUIREMENTS

3.1 Interfaces

3.1.1 User Interfaces

3.1.2 Software Interfaces

3.2 Functional Requirements

This section includes the requirements that specify the fundamental actions of the software system, and the necessary outcomes that are expected by the client.

3.3 Performance Requirements

ID: FR1

TITLE:

DESC:

DEPEND: