NORTHWESTERN UNIVERSITY

An Analysis of Human and Robot-Generated Information Streams to Inform Dynamic Autonomy Allocation

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

MASTER OF SCIENCE

Field of Mechanical Engineering

By

Christopher X. Miller

EVANSTON, ILLINOIS

July 2020

"Happy the man, and happy he alone,

he who can call today his own:

he who, secure within, can say,

Tomorrow do thy worst, for I have lived today."

– Horace, *Odes*

ABSTRACT

An Analysis of Human and Robot-Generated Information Streams to Inform Dynamic Autonomy Allocation

Christopher X. Miller

An ideal autonomy allocation framework shifts autonomous assistance based on the user's needs and desires. A framework seeks to classify its input signals – human control commands, environmental signals, or the interaction between the human-robot team – as 'shift' or 'do not shift' between levels of assistance. A human subject study was designed to collect ground truth data to determine when to shift between levels of autonomy in a 1-dimensional assistance framework. Information streams from the human, the interaction between the human and the robot, and the environment were analyzed. Machine learning methods – both classical and deep learning – were trained on this data. An analysis of information streams from the human-robot team indicate an aggregate classification accuracy of 59% and 85% for human-only and interaction-only data sets. Interaction specifically yielded a notable improvement from chance (46%), when using classical machine learning methods. Deep learning methods, as designed for this research,

were not very effective at classifying 'when to shift.' This work demonstrates the classification power of human-only and the fantastic value of interaction information streams for use in the design of shared control frameworks. This work further analyzes the usefulness of different levels of autonomy and user experiences. Possible future work to optimize the presented classifiers and to develop individualized control authority allocation paradigms is presented and encouraged.

Acknowledgements

I believe every effort is the amalgamation of the individual's efforts and those supporting them along the way. As this thesis is very likely the end of my academic career, I find it prudent to thank those who helped me on my journey so far.

To my advisor Brenna Argall and my labmates – Alex Broad, Tem Gebrekristos, Deepak Gopinath, Siddarth Jain, Vishwajeet Karmarkar, Mike Lee, Mahdieh Nejati, Kyra Rydy, Andrew Thompson, and Michael Young – thank you for your support, your debugging advice, your tolerance of my sense of humor and puns, and most importantly for your camaraderie. To Michael Young I owe extra thanks for not only setting a Texas-sized example of true technical skill and genuine character, but also for being a thoughtful and an incredibly supportive friend.

To my colleagues and mentors from my pre-Northwestern career at CMU-NREC (Dimi Apostolopoulos and Pat Weber) and Penn State (Tim Kane, Sean Brennan, Jeff Schiano, Julio Urbina, and Helen Hintz) – thank you for your support, your reference letters, your last hour schematic reviews, and for sculpting my world view.

To the crazy people I met in Chicago — my time in this beautiful city would not have been the same without you all. From the first day touring Northwestern to now, you helped me call this city home. To list everything we've done from *Mother's* to sailing to too-much-for-a-grad-student-budget-Michelin-Star'd-dining is impossible. Bit Meehan, Steve Bookbinder, Ira Rose, Michael Young, Seline Tan-Tores, Tom Collis, Amanda Hajj,

Ian Abraham, Kat Popovic, Rehan Ghani, and Folkers Rojas thank you for the adventures, the memories, and the love. Never be a stranger, my house is your house.

To all my crazy, brilliant friends who stood by me from the start – Vikas Aragam, Karthik Pillutla, Nick Petrunyak, Gage Walters, Ravender Virk, his grace Tengiz Gegelia, Kara Morgan, and Alisha Rege – I could write an entire thesis on our past adventures. Thank you for the love, the laughs, the food, the memes, and the adventures. Life would not be the same without you all. I doubly thank Alisha and Ravender for helping me keep my sanity while completing this thesis; I quite literally would not have been able to complete project without your support. Words cannot describe my gratitude.

To Abhiram "Dave Witson" Chivukula – without your guidance and mentorship, I'd not have even gotten into graduate school, let alone win two fellowships and figure out at least part of what makes me happy in life. You pushed me harder than anyone else in my life but, you pushed me to discover the true meaning of excellence and, most importantly, happiness. Thank you, Abhi, for being the older brother I never had.

To my late father, Chris (June 1963 – March 2009) – thank you inspiring me to become an engineer, challenging me to think smarter-and-not-harder, for giving me my obnoxious laugh, for setting egregiously high standards, and for giving me advice to last a lifetime. I finally believe that I've made you proud. I love you, dad.

To my mom, Mary – you had the impossible task of raising Michael and I after dad died. We both were already extremely strong-willed and stubborn kids. You faced your fears and demons and provided us with the resources necessary to grow into the men we are today. I know without a modicum of doubt none of this work would be possible

without you. Thank you, mom, so very much, for everything you've done and continue to do for Michael and I. I love you.

This work was supported by a grant from the United States Department of Defense through the National Defense Science Engineering Graduate (NDSEG) Fellowship Program under fellowship number F-6826820873. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the U.S. Department of Defense.

Dedication

To Mom, Dad, Michael, and all of my friends for believing in me.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Many persons with mobility-limiting disabilities rely on assistive devices — electric wheelchairs, canes, walkers, etc — to regain some, if not most, of their freedom-of-mobility. In the mid-1980s [**31**], roboticists started to bring robotics autonomy to the assistive technologies domain. However, many users of these early technologies felt that these devices *limited* their freedom-of-mobility rather than improved their freedom-of-mobility; many assistive device users felt the robotic autonomy was doing *too much* or completing tasks they felt capable of completing. It quickly became evident to researchers that assistive device end-users strongly prefer to collaborate with robotic devices rather than be 'moved' by their devices [**26**].

Outside of the assistive technologies domain, human-robot teams are rapidly becoming ubiquitous in society. Thus, the need for robots to recognize and adapt to their human part's changing abilities becomes critical. This is especially true when teams are operating in dynamic, complex, and uncertain environments where poor robotic understanding can lead to human injury or worse.

Shared control, a subdomain within robotics, seeks to effectively split a physical system's control authority between a human and an robotic agent. In one particular sub-area within shared control, a robot's autonomous assistance is broken into discrete, well-defined levels known as levels-of-autonomy (LOAs). Presently, to shift between these LOAs, a user typically directly requests an increase or decrease in autonomous assistance. This

thesis explores how the information streams generated from the human, the environment, and from within the human-robot team may be used to shift between these LOAs without the user's explicit request. Succinctly, this work seeks to present preliminary research towards solving the problem of 'when to shift' between LOAs autonomously.

The questions of 'when-to-shift?' between LOAs and 'what information useful?' is framed as a supervised classification problem. This work is *not* an exhaustive search of all possible solutions for classifying when-to-shift between LOAs. For tractability, this thesis focuses solely on the information streams from the human and as the result of interaction between the human and the robot. The specific contributions of this document are presented below.

1.1. Main Contributions

The main contributions of this document are as follows:

- A human subjects study to generate a labeled data set from which supervised machine learning algorithms may classify when to switch between LOAs
- An analysis of both classical and deep learning classification methods for use in possible autonomy allocation frameworks
- A evaluation of different information streams — namely those from the human and human-robot interaction — for their suitability in predicting 'when-to-shift'

1.2. Thesis Outline

The remainder of this thesis shall be presented as follows. Beginning in Chapter 2, a review of relevant literature in shared control and dynamic control authority allocation is presented to demonstrate both this research's exigency and to extend its place within

the field. Chapter 3 presents an experiment and a *post-hoc* analysis of collected data to inform the design of future autonomy allocation frameworks. Finally, in Chapter 4 key conclusions and future directions are presented.

CHAPTER 2

Background and Related Work

Here a presentation of relevant prior work within the domains of shared control and dynamic control authority allocation are presented. This chapter seeks to inform the reader of this thesis's place among the related literature and subsequently build this work's exigency.

2.1. A Review of Human-Robot Control Sharing

Robots are rapidly becoming more ubiquitous throughout society; as such, so is the human-robot team. As was demonstrated in [**26**], individuals working in human-robot teams seek to work along side their robotic partners rather than have fully autonomous robots that neglect their human counterpart's desires. The idea of *shared control* describes this desire to work together with one's robotic partner [**32**]. Most early shared control literature seeks to allocate control authority to optimize for human-robot team success in completing a desired task known *a priori* [**29**]. Other early strategies sought to optimize for the user's safety and provide early interventions to halt the robotic platform from performing unsafe activities [**44**] (e.g. blocking a wheelchair from driving off a ledge [**6**]). More modern trends in shared control (past 10 years) seek to provide the human with as much control as possible [**26**]. Earlier in this trend, linear blending schemes adapted a blending parameter to an individual's needs [**20**]. Similarly, others propose using POMDPs [**27**], probabilistic methods [**14**], [**30**], or control barrier functions [**10**].

Most recently, researchers have modeled a human's ability as a probability distribution that is continuously updated to incorporate the most recent information [**38**] (e.g. the robot state, environment state, and/or task information). Others have demonstrated success using data-driven, model-based shared controllers that seek to generate generalizable control policies that do not require knowledge of a user's long-term objectives [**11**]. While these state-of-the-art methods are achieving excellent performance in laboratory settings, it was shown in [**23**] that many shared control methods may not necessarily optimal for the end-user if they are optimal for the robot; [**23**] concluded that the future shared control frameworks may require individualized personalization.

Interestingly, there exists a possible framework for further individualized customization. In 1978, Sheridan and Verplank introduced the concept of Levels of Autonomy (LOAs) [**42**]. In this interaction paradigm, a fixed number of discretized LOAs are defined where, within each LOA, the control authority allocated to each the human and the robot is pre-defined. As the level increases from lowest to highest, so increases the amount of robotic assistance provided to the human. Ten LOA were initially defined in this one-dimensional framework. More recently, this idea has morphed into the concept of an *autonomy spectrum* [**16**] where each level rests on a two-dimensional plane with autonomous assistance on the vertical axis and function-to-be-executed on the horizontal. Such a 2D framework could be customized to an individual to follow the suggestions proposed in [**23**] such that each level is optimized for a person's needs and team's task at hand.

Where as some research has been conducted in defining functions to shift between individual LOAs on a linear scale [**15**], [**47**] not enough research has been conducted to

claim a definitive solution. Therefore, the proposed work seeks to explore one possible solution to inform a framework for shifting between linearly-defined, 1D LOAs. Future work may prove to define custom LOAs, a 2D shifting framework, and generalizable shifting methods to define 'when to shift' and 'to which level.'

2.2. A Review of the Information Streams used in Dynamic Control Allocation

Nearly all shared control frameworks rely on information from within the human-robot team or the team's environment to allocate control authority [**32**], [**44**]. Allocation frameworks based in safety tend to depend on *environmental* information streams [**6**]. Those streams used in assistive technologies rely more heavily on information derived from the *human*'s control signals [**23**], [**33**]. Some such information extracted from the *human* may include simple measures such as the smoothness of the human's input [**5**] or the frequency of issued commands [**33**]. More complex analyses of human-based information streams compute measures of comprehension that investigate the human's understanding of the robot's physical limitations via control theoretic stability metrics [**21**], [**49**].

Another interesting information stream to inform autonomy allocation monitors human-robot team *performance*. As was shown in [**44**], there are few real-time, measures of human-robot performance that generalize for all tasks. Most shared control literature review human-robot team performance *post-hoc* using measures such as task completion time or accumulated mistakes. Fortunately, some recent work has sought to close this gap by defining real-time computable performance measures in areas such as command following and trajectory following [**34**]. Further research needs to be conducted to better define task agnostic measures of human-robot team performance.

More recently, there has been an emphasis on leveraging the information encoded in the *interaction* between the human and the robot. Some such measures of interaction monitor the robot's trust in the human's capabilities (i.e. the divergence between the human's and the robot's control commands) [**12**]. Other measures investigate the quality of the information stream between the human and the robot [**51**]. Most recently, one measure compare an existing human-robot team with high-performing human-robot teams (e.g. Discrete N-Dimensional Entropy of Behavior — DNDEB) [**50**]. Given that human-robot team performance depends directly on the interaction between the human and the robot, it can also be stated that *performance* information streams are a subgroup within *interaction* information streams.

Some of measures of *interaction* are quite computationally intensive and may require computing platforms capable executing highly-parallelized code [**11**]. However, processing such information streams is still less computationally intensive than computing those from most environmental information streams whereas processing entire point clouds may be necessary. Finally, many measures of the human's state are typically the least computationally intensive [**33**]. However, some of the most informative information signals within human information streams may require specialized and/or costly hardware (e.g. EEG caps) [**7**].

Most, if not all, of the aforementioned information streams may rely on some form of manual feature engineering to extract information to inform an autonomy allocation framework. Some believe an ideal framework may require little-to-no feature extraction and process all raw information streams as to generalize to many platforms [**36**]. The need for an investigation into which of the information streams — or combinations thereof —

are most effective at informing an autonomy allocation framework is observed. Thus, a first analysis of the aforementioned is presented in this thesis.

CHAPTER 3

Classifying Human and Autonomy Signals to Inform Autonomy Allocation

3.1. Introduction

This chapter describes a human trials study to generate labeled data to classify 'when to shift' between discrete LOAs. It further presents classification attempts using both classical and modern (deep learning) machine learning methods. Finally, this chapter reviews the classifier's results and presents a discussion of other study-related observations.

3.2. Information Streams in the Human-Robot Team

Numerous information streams exist within the human-robot team. Information can be derived from the robot's *environment* via LIDAR or encoder measurements. Information about the *human's* desires and state may be derived from their joystick or other interface devices. The *robot's* planner provides information about its state through its sensor readings and suggested motions. Less commonly considered are information streams that describe the *interaction* between the human and the robot; for example, the difference between a planner's and a human's control commands. This smorgasbord of information flowing within the human-robot team can be leveraged to improve the human-robot team capabilities in the human's favor. To that end, four primary information streams are proposed in this thesis to exist within the human-robot team (Table 3.1).

Information Stream Name	**Description**
Human Control Signals ($\mathcal{H}$)	Information derived from the human's control inputs
Autonomy Control Signals ($\mathcal{C}$)	Information derived from the robot's planner's control commands
Interaction Signals ($\mathcal{I}$)	Any information derived as a result of the human-robot interaction (e.g. differences between the human's control commands and the robot's); includes the team's task performance
Environmental Signals ($\mathcal{E}$)	Any information derived from the environment in which the team is performing a task

Table 3.1. The four proposed signal groups within the human-robot team and their description

Which of these information streams — or combinations thereof — are best suited to inform an autonomy allocation framework most optimally is to be presented in a later part of this thesis. To refine the scope of this thesis, only information streams from $\mathcal{H}$, $\mathcal{I}$, and $\mathcal{E}$ are considered. Information from $\mathcal{C}$ is not analyzed as to retain a focus on shifting autonomy for the human's needs.

Some of these information streams are more computationally cost effective than others. For example, processing environmental data as derived from a large point cloud is more costly than computing features from a human's joystick commands. Therefore, an ideal allocation framework considers not only the richest information stream, but also the associated computational cost. The information streams presented in Table 3.1 are roughly ordered from least- to most-computationally costly. An in-depth discussion of computational cost is not provided, only considered in a general sense.

3.3. Autonomy Framework Description

The proposed autonomy allocation framework is designed around the 1-dimensional, discrete levels of autonomy framework proposed by [**42**]. In other words, shifting is only

allowed between adjacent LOAs in a linear framework. LOAs, A, are partitioned into n discrete levels such that $A_i, i \in [0, n]$ where, as $i \to n$, the control authority shifts from the human to the robot. Within this framework, transitions may only occur between adjacent autonomy levels following $A_i \to A_{i+1}$ or $A_i \to A_{i-1}$. In summary, the amount of robotic assistance increases with A_i and vice-versa.

Here, autonomously shifting between LOAs is framed as a supervised, multi-class classification problem where a learning algorithm will resolve a function such that $g : X \to Y$, where X are features and Y are the set of class labels. Here, Y includes three labels: shift up ($y_i = 2$), shift down ($y_i = 0$), a default label, no shift ($y_i = 1$). Features vectors x_i are labeled explicitly when a human or a naïve autonomy shifting framework requests a shift up or down, otherwise they are assigned the default. The set of features, X, are extracted from the information sets presented in Table 3.1. Classification models are trained over feature extracted from these information streams and combinations of these streams to reason about the information necessary to shift between LOAs.

3.4. Experimental Design

This section presents the experimental design and procedures used to build the data set used to construct a proposed autonomy allocation method. For this preliminary research, only "able-bodied" participants met the project's exclusion criteria.

3.4.1. Robotic Platform

A customized, powered wheelchair is used for this experiment. The wheelchair, shown in Figure 3.1, is a commercially available Permobil C300 (Permobil, Sweden) wheelchair

fitted with a Velodyne LiDAR (Velodyne, San Jose, CA), RGB-D cameras, and an on-board computer. The wheelchair can be autonomously driven, manually driven using a standard two-dimensional joystick, or the computer can blend the human's and the robot's commands.

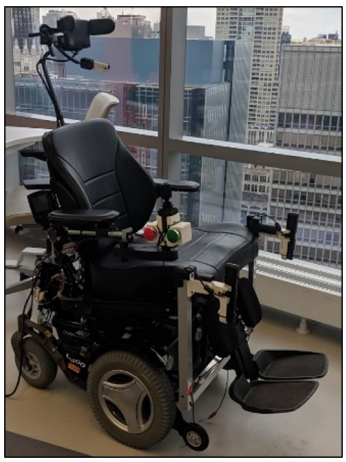

Figure 3.1. The argallab autonomous wheelchair

A Microsoft Surface 2 (Microsoft Corporation, Redmond, WA) is mounted on the wheelchair to present the user with secondary tasks to modulate a participant's cognitive load. Finally, headphones are installed on the wheelchair to allow the platform to provide the user with audible alerts to changes in system state.

3.4.2. Level of Autonomy Control Descriptions

Three LOAs are presented for this research. These levels were selected and refined via an in-house pilot study. These are not to be confused with $\mathcal{A}$ which are computations of human-robot agreement.

(1) A_0 **- Teleoperation**: In A_0, the executed command directly maps the user's input to the command output and no assistance is provided by the autonomy.

(2) A_1 **- Autonomous Stopping**: In A_1, the autonomy prevents the user from running into nearby obstacles by blocking commands that will lead to collisions and allowing direct command mapping otherwise.

(3) A_2 **- Blended Autonomy**: In A_2, the wheelchair blends the human's commands ($\mathbf{u}_h$) with the autonomy's obstacle avoidance planner ($\mathbf{u}_r$) using a linear blending paradigm (eq. 3.1) to produce an executed command, $\mathbf{u}_c$.

$$\mathbf{u}_c = (1 - \alpha) \cdot \mathbf{u}_r + \alpha \cdot \mathbf{u}_h \tag{3.1}$$

The blending parameter ($\alpha = 0.5$) is selected empirically. A_2 follows the control paradigm proposed in [**20**].

3.4.3. Autonomy Switching

A naïve autonomy shifting framework derived from measures of safety is implemented on the wheelchair. The robot shifts up if it is located less than 0.65 m to the nearest obstacle for more than 3 seconds. If the robot is located more than 0.75 m to the nearest obstacle for more than 3 seconds, the robot shifts down. At any time during a trial, the user can verbally request a shift up or down in assistance and the experiment's proctor will change the LOA. In this framework, the human's request to shift *always supersedes* the robot's request. Shift requests by the human or the robot beyond the minimum or maximum level-of-autonomy are discarded. In all instances, the user is alerted of a successful shift via an audible alert.

3.4.4. The Study

The following section describes the study conducted using the aforementioned wheelchair platform and autonomy allocation framework.

3.4.4.1. Primary Task. Six different wheelchair obstacle courses were hand-designed for participants to navigate. The obstacle courses contained different permutations of navigating a doorway, a shallow ramp, a tight straight-away, a tight curve, and avoiding a moving obstacle (a manually driven RC car). Figure 3.2 displays one such course. A participant completes a course after they return to the course's starting location in the correct orientation.

Figure 3.2. An example obstacle course

Each participant completes each course once. At the start of each course, a participant is assigned a starting LOA. The order in which users experience each course and the assigned starting LOAs are assigned using a counterbalanced, within-subjects design.

3.4.4.2. Secondary Task. A secondary 'distraction' task is presented to the participant on the wheelchair-mounted tablet to modulate the participant's cognitive load while they navigate the obstacle course. This distraction task (Figure 3.3) is based on the 'gauges' sub-task of the NASA Multi-Attribute Task Battery (MATB-II) [**41**]. This task is well-studied and widely used in human-factors literature for cognitive load modulation. Cognitive load is modulated to sufficiently increase task complexity such that the users may require autonomous assistance.

The participant is instructed to monitor and respond to fluctuating gauge and lights on a mock control panel 3.3. Participants must keep each of the four fluctuating gauges within two tick marks of a center marker and respond if a gauge exceeds this range by tapping the button below the gauge on the tablet's touchscreen. If one of the two large

lights changes colors, the participant must respond by tapping the button. The rate at which the participant experiences malfunctions and the gauge speeds are tunable.

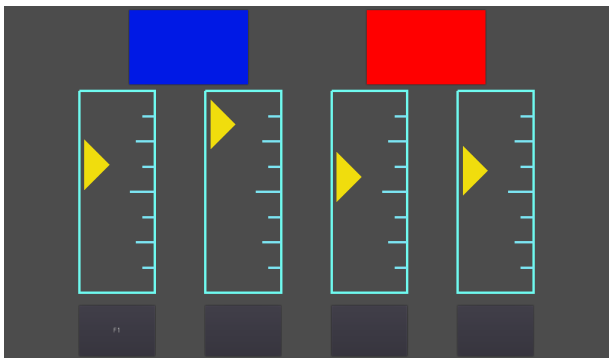

Figure 3.3. The secondary task. Here, the second gauge from the left is out-of-bounds

Our system uses the U.S. Air Force Research Laboratory's suggested settings [**19**]. According to [**15**], users need an incentive to interact with secondary tasks sufficiently. To that end, users must respond to at least 75% of malfunctions accurately else the experiment is paused until secondary task performance exceeds 75%; the performance threshold was selected empirically.

3.4.4.3. Study Protocol. To complete the experimental tasks, participants drive the robotic wheelchair platform through unique obstacle after undergoing training. The entire experimental protocol and consent form were approved by the Northwestern University Institutional Review Board.

At the start of each session, each participant is given a description of the hardware and the trial. They are then trained for each LOA and the distraction task. Training is complete after the participant completes one traversal of a demonstration course at each LOA and after they complete distraction task training.

At the start of each trial, the wheelchair is driven to the starting location by the experimenter and the obstacle course is set. The proctor walks through the course for the participant to demonstrate the path. At any time during the task, the participant may ask for directions. Before the trial's start, the participant is informed of their starting

LOA and reminded that they may verbally request an increase or decrease in LOA at anytime. The participant is then allowed to begin the course. This process is repeated for each of the six trials.

After completing a trial, the participant completes a survey and the trial time is recorded. After completing all trials, a final, custom survey is administered and the participant is provided with an honorarium for their time.

3.4.4.4. Participant Evaluation Metrics. The two metrics utilized for evaluating participant performance and experimental perceptions were:

(1) **Perceived Difficulty**: The raw, modified NASA-TLX score [**25**]. The TLX score is shown to be a sufficient measure of perceived difficulty [**24**] and the modified TLX is quicker-to-administer, comparable substitute [**30**]. The modified TLX is presented to each participant after each trial. The seven statements for the TLX are presented in Appendix A.

(2) **Subjective Measures**: A 5-point Likert survey consisting of six statements was presented to each study participant after completing all six courses. These questions assessed the subject's perception of the experiment and their perception of the individual LOAs.

3.5. Data Set Preparation and Description

The following section describes the specific signals extracted from the wheelchair navigation tasks, participant demographics, data labeling, and filtering methods.

3.5.1. Signal Selections and Descriptions

The raw signals from the robot that fall into each of the three information streams being reviewed are presented in Table 3.2. Since the focus of this work is shifting as a result of human and/or the human-robot interaction, $\mathcal{E}$ is reviewed only in combination with both $\mathcal{H}$ and $\mathcal{I}$. Features are extracted from these signals via either hand engineering or feature learning. All signals are sampled at a constant 25Hz.

Information Stream	**Raw Signal Description**
Human Control Signals ($\mathcal{H}$)	J_x, J_y: control commands from the operator's joystick from the joystick's x- and y-axes. ω_h, v_h: the human's rotational and linear velocity commands after translation from x-y joystick to linear-rotational values oftentimes compactly presented as $\mathbf{u}_H$
Interaction Signals ($\mathcal{I}$)	ω_r, v_r: the robot's rotational and linear velocity commands from the planner; oftentimes compactly presented as $\mathbf{u}_R$ ω_h, v_h: the human's rotational and linear velocity commands after translation from x-y joystick to linear-rotational values oftentimes compactly presented as $\mathbf{u}_H$
Environmental Signals ($\mathcal{E}$)	$\mathcal{R}$: the robot's distance to the nearest obstacle as extracted from the planner's costmap

Table 3.2. The raw signals grouped into each of the three information streams reviewed; features are extracted from these raw signals via hand engineering or feature learning methods

3.5.2. Data Set Description

Each subject (N=16) completed six different courses for a total of 96 total trials. The mean age of participants was 26.1 $\pm$ 5.4 years and an equal number of men and women participated in the study. A total of 1044 shifts were requested by both users and the naïve shifting algorithm. Table 3.4 presents the breakdown of shifts up and down for the

human and the robot. On average, each trial had 10.8 ± 4.38 shifts; Table 3.3 presents statistics for both the human-robot team and the human alone. Furthermore, each trial lasted 112.4 ± 20.3 seconds with a minimum trial time of 68 seconds and a maximum trial time of 160 seconds. No participants requested clarification regarding the course or their level of autonomy during their experiments.

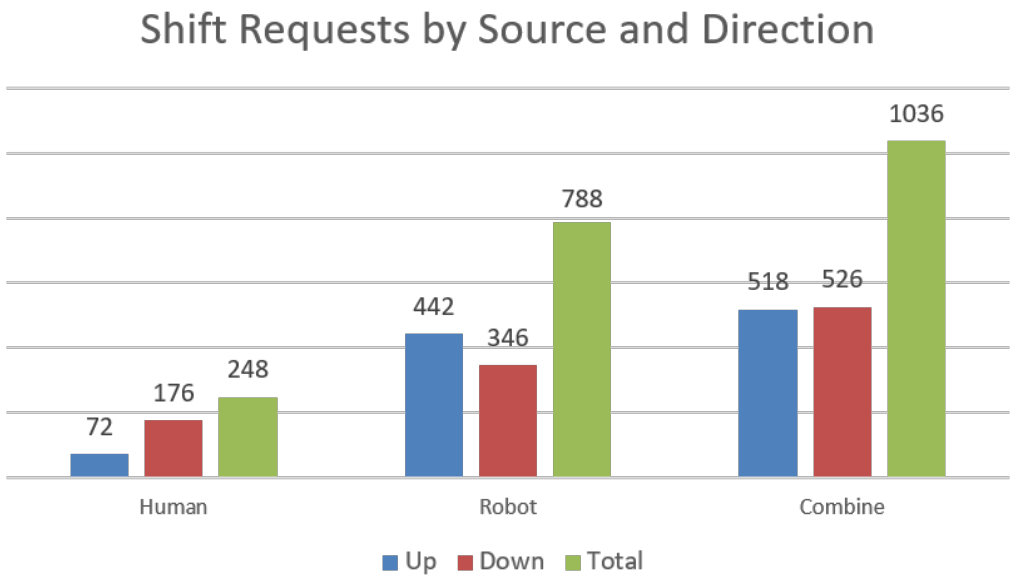

Figure 3.4. Shift requests grouped by requester and organized by direction

Statistic	**Human and Robot**	**Human Only**
Mean Requests per Trial	10.8	2.57
Standard Deviation	4.38	2.97
Median Requests per Trial	10	1
Max Requests	24	14
Min Requests	0	0

Table 3.3. Statistics describing the shift requests for both the human and the human-robot team

3.5.3. Shift Filtering

During data collection, it was anecdotally observed that some study participants disagreed with the robot's changes in assistance. In response to this disagreement, some participants would 'cancel' the robot's shift LOA shift by immediately requesting a shift that would return the participant the previous LOA. To train on these shifts would bias the learner to classify shifts that shouldn't have occurred. Since these cancellations occurred in rapid succession, these requests could be filtered by removing all opposing shift requests that occurred less than some constant time, τ_c, apart. This value, τ_c was selected by clustering the time between all shifts of opposing directions. It was found that all shifts in opposing directions that occurred less than 3.03s apart were to be set aside for later analysis and not included in the training data.

3.5.4. Data Labeling and Conditioning

A look-back window will be created from the shift request back some time N as to label the points prior to a shift which support the need for a shift. All other points shall be labeled 'no shift.' Since this duration, N, is unknown, it becomes an additional hyperparameter tuned during training. All values of N yielded very few shift labels and thus, create a highly-imbalanced data set. The accommodations to address class imbalance will be presented within sections 3.6 (classical) and 3.7 (deep learning).

Since the control commands change for different LOAs (e.g. a user may provide less frequent commands in a higher LOA), it is determined that the data should be conditioned on the LOA in which it were sampled. This conditioning also addresses the differences in available classes in within different LOAs. In A_0 (up, no shift) and A_2 (down, no shift),

a binary classification problem is presented. Unlike the boundary LOAs, a multi-class classification (up, down, no shift) problem is presented in A_1.

3.6. Classical Classifiers

Classical machine learning methods present significant capacity for learning complex classification models that can out-perform deep learning methods. As such, five different classical models were trained and their results analyzed. This section presents the data engineering, feature engineering, learner descriptions, and model performance for those classical machine learning methods.

3.6.1. Feature Engineering

From the raw information streams presented in Table 3.2, five hand-engineered features were extracted. This subsection presents these features — it must be noted that there are many other possible features and these were selected from trends in recent literature.

3.6.1.1. Frequency of Human Input — $\mathcal{H}_\Omega$. Researchers in automotive engineering [**8**] have used frequency of human input to classify the need for autonomous assistance. Researchers in rehabilitation robotics [**33**] have used the formulation in (eq. 3.2 - 3.3) to classify robotic arm task difficulty.

$$\Omega_t = \frac{1}{N} \sum_{k=t-N}^{t} \frac{1}{t_k - t_{k-1}} \tag{3.2}$$

$$\mathcal{H}_{\Omega_E} \leftarrow \alpha \cdot \Omega_t + (1 - \alpha) \cdot \Omega_{t-1} \tag{3.3}$$

where Ω_t is the present input rate, α is the weighting on the most-up-to-date measurement in an Exponential Moving Average (empirically set, $\alpha = 0.8$), t_k is the time at which a new joystick command from J_x or J_y is received (see Table 3.2, $\mathcal{H}$), N specifies the filter size, and $\mathcal{H}_\Omega$ is the human frequency feature.

3.6.1.2. Smoothness of Human Control Signals — $\mathcal{H}_\mathcal{S}$. The smoothness of a person's motions is often monitored to track an individual's recovery from injuries such as stroke or similar [**4**]. One such measure of smoothness is the dimensionless SPARC, or Spectral Arc Length (eq. 3.4 - 3.6). This feature is selected for both its effectiveness at measuring smoothness among healthy and differently-abled populations and for its high noise-immunity [**5**].

$$\mathcal{H}_\mathcal{S} \leftarrow -\int_0^{\omega_c} \left[\left(\frac{1}{\omega_c} \right)^2 + \left(\frac{d\hat{V}(\omega)}{d\omega} \right)^2 \right]^{\frac{1}{2}} d\omega \tag{3.4}$$

$$\hat{V}(\omega) = \frac{V(\omega)}{V(0)} \tag{3.5}$$

$$\omega_c = \min\{\omega_c^{max}, \min\{\omega, \hat{V}(r) < \bar{V} \; \forall \; r > \omega\}\} \tag{3.6}$$

where $V(\omega)$ is the Fourier magnitude spectrum of the user's joystick inputs (derived from from J_x or J_y; see Table 3.2, $\mathcal{H}$) over an N second sliding window of commands, $\hat{V}(\omega)$ is the normalized Fourier magnitude spectrum of the user's inputs, ω_c is a dynamic cutoff frequency tuned for noise sensitivity, ω_c^{max} is the joystick's bandwidth, and $\mathcal{H}_\mathcal{S}$ is the human smoothness feature.

3.6.1.3. Complex Human-Robot Agreement — $\mathcal{I}_C$. Prior work presented in [**12**] indicates that comparing the divergence between human's and the robot's projected paths is suitable for dynamic autonomy allocation. This difference, or measure of how much the robot and the human 'agree,' compares a projection of the robot's pose based upon human's current command to a projection of the robot's posed from the autonomy's planner with a fixed time horizon. The curve length is computed using a discretized version of the Fréchet distance [**1**] as shown in (eq. 3.7 - 3.8). A moving average of Fréchet distances is computed to filter possible noise.

$$\mathcal{F} = \inf_{\alpha,\beta} \max_{t\in[0,1]} \{d\left(A(\alpha(t)), B(\beta(t))\right)\} \tag{3.7}$$

$$\mathcal{I}_C \leftarrow \frac{1}{N} \sum_{i=t-N}^{t} \mathcal{F}(\mathbf{u}_{H_i}, \mathbf{u}_{R_i}) \tag{3.8}$$

where A, B are two curves, `inf` is the infinium (or greatest lower bound), d is a distance function, $\mathbf{u}_H$ and $\mathbf{u}_R$ the human's and robot's projections, respectively (see: Table 3.2, $\mathcal{I}$), N specifies the moving average filter's size, and $\mathcal{I}_C$ is the complex agreement feature.

3.6.1.4. Simple Human-Robot Agreement — $\mathcal{I}_s$. Inspired by $\mathcal{I}_C$, a simple feature is designed to compare the instantaneous commands provided by the human and the autonomy at a given time step by computing the L2-normed difference between the human and autonomy commands. To filter noise, an N-point moving average of these differences

is also computed as shown in (eq. 3.9).

$$\mathcal{I}_s \leftarrow \frac{1}{N} \sum_{i=t-N}^{t} ||\mathbf{u}_{H_i} - \mathbf{u}_{R_i}||_2 \tag{3.9}$$

where $\mathbf{u}_H$ and $\mathbf{u}_R$ the human's and robot's control command,s respectively (see: Table 3.2, $\mathcal{I}$), N specifies the filter size, and $\mathcal{I}_s$ is the simple agreement feature.

3.6.1.5. Distance to Nearest Obstacle — $\mathcal{E}_\mathcal{D}$. The nearest obstacle in the planner's costmap is identified and the obstacle's distance, following the rule in (eq. 3.10), is returned. This rule is derived from the perception system's range-accuracy specifications and the platform's minimum stopping distance.

$$\mathcal{E}_\mathcal{D} \leftarrow \begin{cases} 0.50 & \mathcal{R} \leq 0.50 \\ \mathcal{R} & 0.50 < \mathcal{R} < 3.00 \\ 3.00 & \mathcal{R} \geq 3.00 \end{cases} \tag{3.10}$$

where $\mathcal{R}$ is the distance to the nearest obstacle (see: Table 3.2, $\mathcal{E}$) and $\mathcal{E}_\mathcal{D}$ is the filtered distance.

3.6.2. Data Engineering

After feature engineering, the data were min-max normalized for each unique feature such that all values are bounded from 0 to 1 and that the "maximal" value is mapped to one. The minimum and maximum values used for min-max normalization for each feature were derived from the feature's physical parameters. After normalization, all features were labeled and conditioned upon their LOA following the description described in section 3.5.4.

The data set was split into train-validate-test data sets following a 60-20-20% ratio. To accommodate class imbalance — the majority class, 'no-shift" comprised 90-95% of the data — the training set's over represented class was randomly down-sampled such that a 90:10%, over-represented:under-represented class balance was achieved. Majority class down-sampling is historically a good starting point for class-imbalance and the ratios are selected empirically using domain-specific knowledge for the given data set [**3**]. Finally, the training, validation, and test data sets were shuffled.

To test the performance of each information stream in label prediction, models were trained using different data sets containing only certain features; Table 3.4 presents these groupings. More exhaustive information stream groupings are for future work.

Information Stream	**Features Selected**
$\mathcal{H}$	$(\mathcal{H}_\Omega, \mathcal{H}_\mathcal{S}) \in X$
$\mathcal{I}$	$(\mathcal{I}_C, \mathcal{I}_\mathcal{S}) \in X$
$\mathcal{H} + \mathcal{I}$	$(\mathcal{H}_\Omega, \mathcal{H}_\mathcal{S}, \mathcal{I}_C, \mathcal{I}_\mathcal{S}) \in X$
$\mathcal{H} + \mathcal{I} + \mathcal{E}$	$(\mathcal{H}_\Omega, \mathcal{H}_\mathcal{S}, \mathcal{I}_C, \mathcal{I}_\mathcal{S}, \mathcal{E}_\mathcal{D}) \in X$

Table 3.4. The features included in different data sets, X, to compare different information streams' classification ability

3.6.3. Classical Models

Five different classical machine learning classification algorithms were selected for training: support vector machines (SVM), random forests (RF), gradient boosting classifiers (GBC), logistic regression (LR), and Naïve Bayes (NB). These methods were selected as to encompass a breadth of different classical machine learning classification algorithms. The popular machine learning package `scikit-learn` was used to implement all classifiers and performance metrics [**37**]. Table 3.6 presents the models trained and their key

hyperparameters and Table 3.5 presents a brief discussion for each model. The parameters used for all models are presented in Table 3.7.

To select the best hyperparameters for each learner, a grid search optimizing over macro-F_1 scores was performed for each learner [**45**], at each LOA, at each (group of) information stream(s). All learners used `scikit-learn`'s default parameters unless otherwise specified in the sections that follow. The grid search was realized using the `hypopt` package which allows for the use of a specified validation data set and parallel learner training [**35**]. In total, 9072 models were trained spanning each LOA, feature set, and hyperameter.

3.6.4. Model Performance Measures

To compare information steam classification performance, aggregate balanced accuracy was computed. This score first computes the balanced accuracy, or the arithmetic mean of the class-wise accuracies for each LOA-specific model. Then, a weighted average of these balanced accuracies is computed where the accuracies are balanced on the number of points in a LOA-specific model's training set. This score treats each class with equal importance and for this preliminary work, all classes are considered equally important.

As a point of comparison an aggregate chance score was computed from the weighted average of each LOA's individual chance scores. The individual chance scores were the inverse of the number of classes per LOA and the weights were the number of points in the test set for each LOA.

Model Name	Model Discussion
`SVM`	Constructs an N-dimensional hyperplane to separate different features belonging to different labels [**18**]; selected for their popularity in multi-class classification
`RF`	An ensemble learner that fits multiple decision trees on multiple subsets of the data set and uses averaging to select the model's prediction [**9**]; selected to test ensemble methods and for its immunity to over fitting.
`GBC`	Also known as gradient boosted decision trees, combine many weakly predictive decision trees to create a stronger predictive model. More weakly predictive trees are fit during the learning process on the negative gradient of the learner's loss function [**22**]. Notable, this method is quite robust to over-fitting even with many boosting stages; selected to test boosting methods on the data set
`LR`	Seeks the relationship between features and classes using probabilities using a logistic (S-curve, sigmoid); estimates the parameters of a logistic function and subsequently computes class probabilities from this function [**17**]; selected for its simplicity and widely acceptance
`NB`	A learning algorithm that applies Bayes' theorem and "naïvely" assumes every feature, given its class variable's value, is conditionally independent [**52**]; here, the `ComplementNB` implementation of the Naïve Bayes classifier is selected because of its capability to handle class imbalance [**39**] and address the "severe assumptions" made by Multinomial Naive Bayes classifiers; selected to investigate the efficacy of Bayesian networks on this dataset

Table 3.5. A discussion for each classical model investigated

3.6.5. Classical Results

Model classification performance was tested using only non-augmented test data and neither validation nor training data were included in testing. Figure 3.5 presents the learner-wise aggregate balanced accuracy scores for all of the learners. The aggregate scores presented are computed over the best performing individual LOAs for each information stream within each individual learner.

The gradient boosted classifiers (GBC) and random forests (RF) out-performed all other models significantly. Specifically, the GBCs performed best for all information

Model Name	Model Description
`SVM`	**kernel functions**: (`rbf`, `poly`, `linear`) **degree** (`poly` only): (3, 8) **C**: (0.1, 1, 10, 100) **class_weight**: `auto`: (`n_samples/ (n_classes * np.bincount(y))` **gamma** (`rbf` and `poly` only): `auto` (kernel coefficient that follows: `1/n_features`)
`RF`	**n_ estimators**: (10, 100, 1000, 2000, 5000) **class_weight**: `auto`: (`n_samples/ (n_classes * np.bincount(y))`)
`GBC`	**n_ estimators**: (10, 100, 1000, 2000, 5000, 10000) **class_weight**: `auto`: (`n_samples/ (n_classes * np.bincount(y))`) **n_iter_no_change** : 1000
`LR`	**C**: (0.1, 1, 10, 100, 1e3, 1e4, 1e5)
`NB`	**alpha**: (1, 2, 10, 20, 100) **fir_prior**: (True, False) **norm**: (True, False)

Table 3.6. Each classical learner and its key hyperparameters.

Parameter	Values
Look-Back Window	$N \in [0.25, 3.50]$ in 0.25 steps

Table 3.7. Settings and hyperparameters used for all classical learners

streams except for the *human*-only information stream. There, the RF out-performed the GBC. Table 3.8 summarizes this performance. All other models barely surpassed chance and will not be reviewed further. To compare the predictive power of each the individual and combined information streams, the best performers for each stream were selected and are plotted in Figure 3.6.

Information Stream	Best Learner
$\mathcal{H}$	RF
$\mathcal{I}$	GBC
$\mathcal{H} + \mathcal{I}$	GBC
$\mathcal{H} + \mathcal{I} + \mathcal{E}$	GBC

Table 3.8. Best classical learners for each information stream

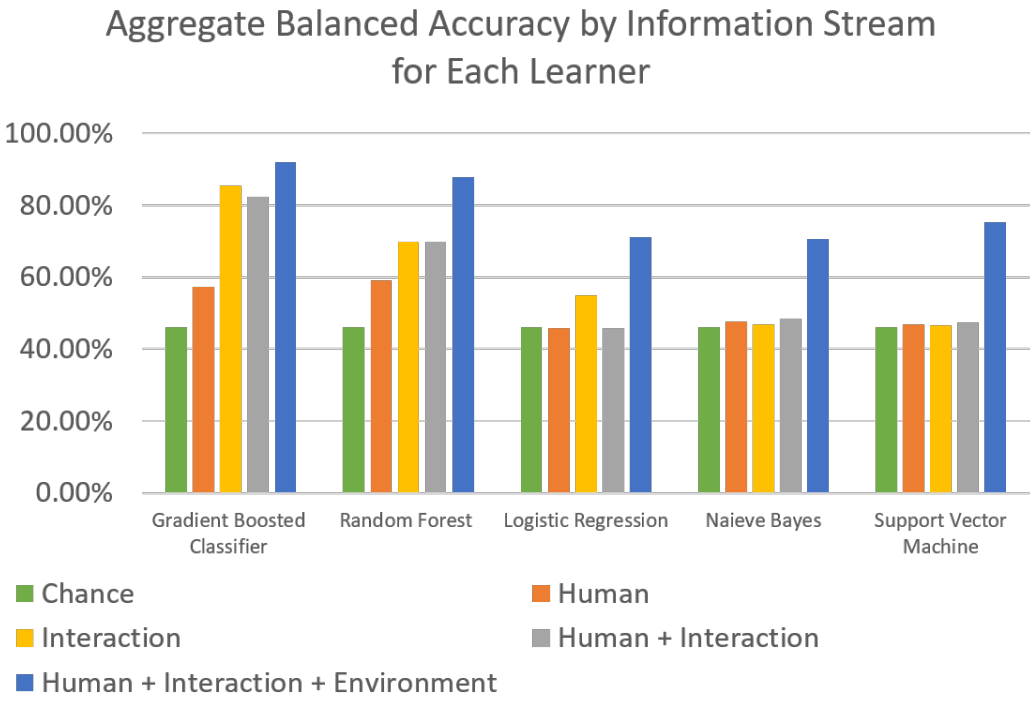

Figure 3.5. Aggregate balanced accuracies per information stream per learner

It's apparent that monitoring only $\mathcal{H}$ encodes sufficient information to perform better than chance (46% vs. 59%). Providing more predictive power still, $\mathcal{I}$ provided an improvement of 26% increase over $\mathcal{H}$ and 36% over chance for a score of 85%. Combining the information streams $\mathcal{H}$ and $\mathcal{I}$ performed slightly worse (85% vs. 82%) than $\mathcal{I}$. This may have occurred because of the similarity of information encoded in both $\mathcal{I}$ and $\mathcal{H}$. Furthermore, providing more information increases the dimensions of the data set which could create sparsity in the data set and thus impact performance. Finally, adding information stream $\mathcal{E}$ provided a modest improvement over $\mathcal{I}$ alone at 92%.

To assess the model's aggregate skill, the upper and lower bounds on the model's classification error are presented in Table 3.9. These error values are the balanced, aggregate error bounded by the balanced, aggregate confidence interval (Wilson score interval).

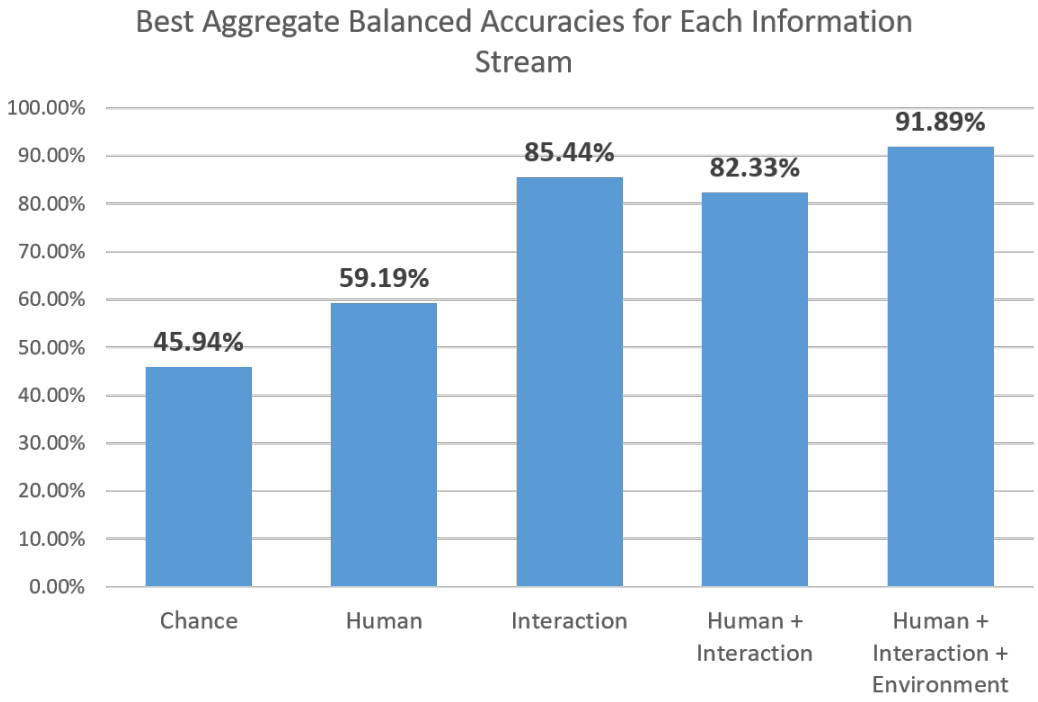

Figure 3.6. The best classical learner per information stream's aggregate balanced accuracy for test data

Information Stream	Aggregate Error	Confidence Interval
$\mathcal{H}$	40.81%	0.5862%
$\mathcal{I}$	14.56%	0.4207%
$\mathcal{H} + \mathcal{I}$	17.67%	0.4550%
$\mathcal{H} + \mathcal{I} + \mathcal{E}$	8.11%	0.3209%

Table 3.9. Aggregate balanced classical model error and the related confidence interval

3.7. Deep Learning Methods

Modern machine learning methods presently focuses heavily upon on the area of deep learning. One of the greatest advantages of deep learning over classical machine learning is the near elimination of manual feature engineering thus, expediting the model learning process.

3.7.1. Data Engineering

The raw data streams presented in Table 3.2 were min-max normalized about their minimum and maximum values as derived from their source's physical parameters. After normalization, the data were divided into overlapping windows (one point in, one point out) of data where each window is labeled according to when a shift was requested. Windows labeled as 'shift' were not overlapped with 'no-shift' windows and vice-versa. The data were once again conditioned on LOA. A class imbalance of roughly 99.5% for the majority class was observed for all window sizes and LOA conditionings. To manage this imbalance, two data sets were generated. One was randomly under-sampled following both a 90:10% majority:minority class balance. The other was constructed from random under-sampled data set where the minority classes were randomly up-sampled until all classes are uniformly represented [**28**]. The data sets were divided into train/validation/test sets following a 60-20-20 split. The test and validation sets were not augmented using any up- or down-sampling.

Since the human control commands are but a transformation of the joystick commands, generating models from a combination of $\mathcal{I} + \mathcal{H}$ was deemed redundant and precluded from training. Table 3.10 presents the signals used in each information stream for deep learning. The table also clearly demonstrates the overlap between $\mathcal{I} + \mathcal{H}$ and thus justifies its preclusion due to redundancy.

3.7.2. Deep Learning Models Trained

Six different deep learning models were selected for this task. Each model was inspired by either existing autonomy shifting literature, its popularity in the field of deep learning, or

Information Stream	Features Selected
$\mathcal{H}$	$(J_x, J_y, \omega_h, v_h) \in X$
$\mathcal{I}$	$(\omega_r, \omega_h, v_r, v_h) \in X$
$\mathcal{H} + \mathcal{I} + \mathcal{E}$	$(\omega_r, \omega_h, v_r, v_h, J_x, J_y, \mathcal{R}) \in X$

Table 3.10. The raw signals for each information stream in, X, for comparison using deep learning

a similar problem. Table 3.12 presents these model's components and Table 3.11 presents the inspiration behind each model. The parameters used for all models are presented in table 3.13. All models were trained on every LOA, for every value of N, and on both the 90:10%- and the uniformly- class balanced data sets. Models were validated and tested using the non-augmented data. The best performing model within each LOA were selected by comparing balanced accuracy scores. All models underwent early-stopping when the model's validation loss increased or remained constant for five epochs. A total of 120 deep learning models were trained.

3.7.3. Model Performance Measures

Aggregate balanced accuracy — the same model performance measure presented in section 3.6.4 for the classical machine learning models — was used to compare different deep learning models.

3.7.3.1. Deep Learning Results. All presented results are based on model performance for test data alone. The 90:10% randomly under-sampled data set yielded no improvement over baseline for all learners and will be precluded from all analysis. Only the uniformly-balanced results shall be presented. For each information stream, there existed a different best learner; these results are presented in Table 3.14. To compare the

Model Name	Model Discussion
`2DCNN`	Inspired by image processing [**48**] and EEG classification [**46**]; CNNs should extract many complex features
`LSTM Only`	Usage inspired by prior work [**51**] where similar LSTM network architectures were used to classify signal degradation in different control levels to inform autonomy allocation; LSTM units should capture time-varying features
`BLSTM Only`	Same literature reference [**51**] as `LSTM Only`; considers the bidirectional time-varying features instead of only uni-directional features
`1DCNN-LSTM`	Inspired by the UC Irvine's Human Activity Recognition data set [**2**] and [**40**], this model implements the model presented here [**13**] which achieves good accuracy on the UCI HAR set
`ConvLSTM2D`	Same inspiration as above [**13**],[**2**]; the ConvLSTM2D [**43**] unit is a variant of the LSTM that contains a convolution operation within the LSTM cell
`2DCNN-LSTM`	Inspired by `1DCNN-LSTM` and `LSTM Only`; it is believed that a 2DCNN could extract more features than a 1D CNN and more LSTM layers may improve classification performance

Table 3.11. A brief discussion for each deep learning model and its origins

predictive power of each information stream, the best performers' aggregate accuracies are presented in Figure 3.7.

The results of shown in Figure 3.7 demonstrate similar ($\mathcal{H}$) or poorer ($\mathcal{I}$: -27% and $\mathcal{H}+\mathcal{I}+\mathcal{E}$: -7%) aggregate performance across all information streams while using deep learning techniques when compared to the best classical learners presented in Figure 3.6. Similar performance for $\mathcal{H}$ reinforces the formerly mentioned result that human-only information streams encode enough information to out-perform chance and possibly inform an autonomy shifting framework.

To assess the deep learning models' aggregate skill, the upper and lower bounds on the model's classification error are presented in Table 3.15. These error values are the

Model Name	Model Description
`2DCNN`	**3x 2D CNNs**: *filters* (in order): 32, 64, and 128, *kernel*: (2,2) **MaxPool2D**: *size*: 2 **Dropout**: *rate*: 0.10 **3x Dense Layers**: *Number of Units* (in order): 500, 250, 50 **Output Layer**
`LSTM Only`	**3x LSTM Units**: *Number of Units*: 200 each **3x Dense Layers**: *Number of Units* (in order): 500, 250, 50 **Output Layer**
`BLSTM Only`	**3x Bi-Directional LSTM Units**: *Number of Units*: 200 each **3x Dense Layers**: *Number of Units* (in order): 500, 250, 50 **Output Layer**
`ConvLSTM2D`	**1x ConvLSTM2D**: *filters*: 64 each, *kernel*: (1,3) **Dropout**: *rate*: 0.10 **1x Dense Layers**: *Number of Units*: 100 **Output Layer**
`1DCNN-LSTM`	**2x 1D CNNs**: *filters*: 64 each, *kernel*: 5 **Dropout**: *rate*: 0.10 **MaxPool1D**: *size*: 4 **1x LSTM Unit**: *Number of Units*: 100 **1x Dense Layers**: *Number of Units*: 100 **Output Layer**
`2DCNN-LSTM`	**3x 2D CNNs**: *filters* (in order): 32, 64, and 128, *kernel*: (2,2) **MaxPool2D**: *size*: (3,3) **Dropout**: *rate*: 0.10 **3x LSTM Units**: *Number of Units*: 200 each **3x Dense Layers**: *Number of Units* (in order): 500, 250, 50 **Output Layer**

Table 3.12. Each deep learning model trained and its architecture.

Parameter	Values
`loss_function`	`categorical_crossentropy`
`batch_size`	128
Output Layer Activation	`softmax`
Activation Function Otherwise	`relu`
Look-Back Window	$N \in [1.25, 3.50]$ in 0.25 steps

Table 3.13. Settings used for all deep networks

Information Stream	Best Learner
$\mathcal{H}$	`ConvLSTM2D`
$\mathcal{I}$	`2DCNN`
$\mathcal{H} + \mathcal{I} + \mathcal{E}$	`LSTM Only`

Table 3.14. Best deep networks for each information stream

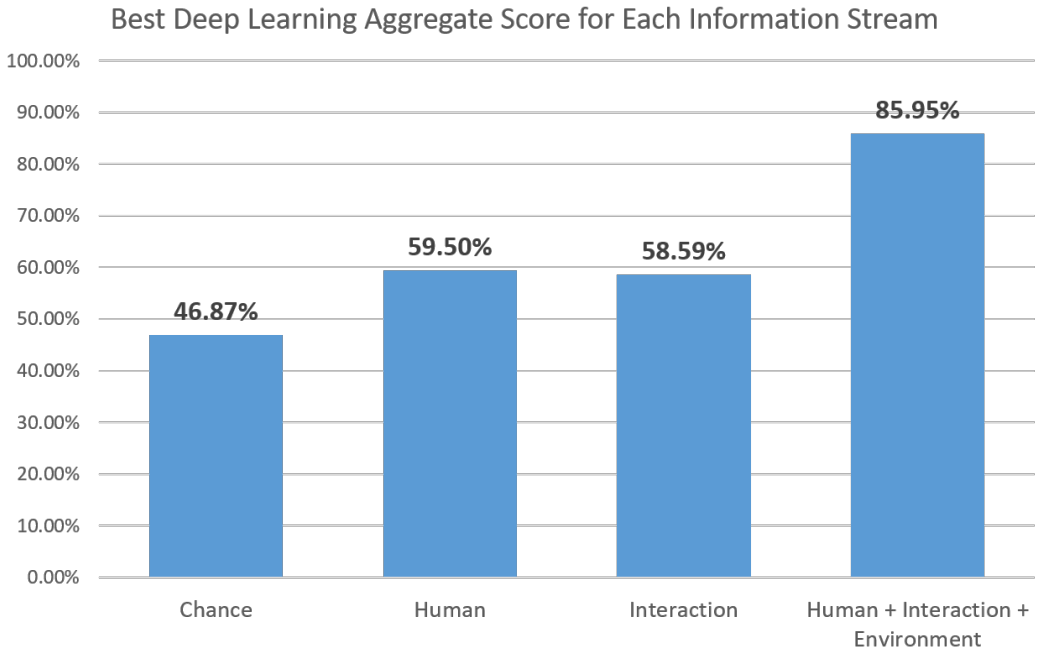

Figure 3.7. The best deep learner per information stream's aggregate balanced accuracy for test data

balanced, aggregate error bounded by the balanced, aggregate confidence interval (Wilson score interval).

Information Stream	Aggregate Error	Confidence Interval
$\mathcal{H}$	40.50%	0.4749%
$\mathcal{I}$	41.41%	0.4503%
$\mathcal{H} + \mathcal{I} + \mathcal{E}$	14.05%	0.3349%

Table 3.15. Aggregate balanced deep learning model error and the related confidence interval

The differences in performance for the other information streams may be the result of the extreme class imbalance. Many deep networks struggle to achieve good performance with highly imbalanced data sets, even after data augmentation [**28**]; however, since some models do respond well to random up- and down-sampling, this exercise was not in vain as it may inform future researchers of methods already dis-proven. Future work investigating more complex models (e.g. variational auto-encoders), simpler models (e.g. fewer units in each layer to prevent training set over-fitting) or conducting more exhaustive grid searches on the existing deep networks are likely necessary. Furthermore, some feature extraction layers may not be able to extract all needed features; some operations, such as the square root function (see computation for $\mathcal{H}_{\mathcal{S}}$ in eq. 3.4) cannot be computed with a CNN. Perhaps, a combination of manual and automatic feature extraction may be necessary to improve deep learning classification accuracy.

3.8. Analysis of Human Behavior and Preferences

The following section reviews results from the surveys administered to each individual following each trial or at the conclusion of all trials. Since these survey results only encompass results from sixteen people, computing meaningful statistical significance is not possible. These results should be interpreted as patterns worthy of further investigation.

3.8.1. LOA Preference Analysis

Following the conclusion of all trials, study participants were presented with the statement *I found* A_X *to be a useful mode of operation* three times where, A is a LOA and $_X \in [0, 2]$

on a five-point Likert scale. For this scale, a higher score indicates stronger agreement. The survey's results are presented in Figure 3.8.

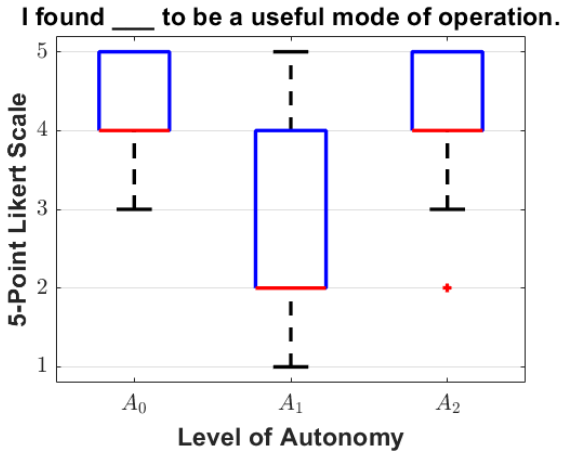

Figure 3.8. User preference for each realized LOA on a 5-point Likert scale

In future studies, a different implementation of A_1 may be necessary to increase participant satisfaction. However, these results do indicate that both A_0 and A_2 were well-received.

3.8.2. Participant Experience Survey

Following the conclusion of all trials, study participants were presented with three statements on a 5-point Likert scale regarding different aspects of their experiences completing the experiment. The survey's results are presented in Figure 3.9. For this scale, a higher score indicates strong agreement and vice-versa. These statements were:

- **Q1**: *It was easy for me to adapt to the new assistance.*
- **Q2**: *The changes in assistance helped me complete the course.*
- **Q3**: *I expected the changes in assistance.*

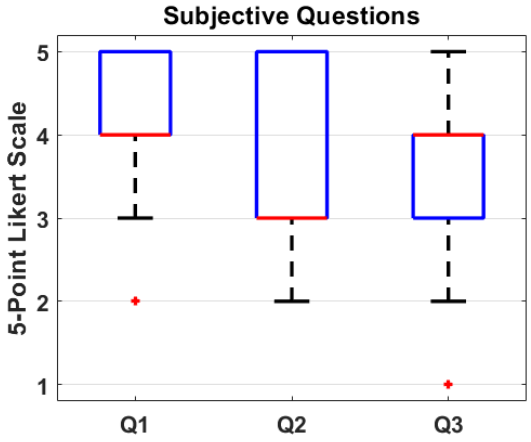

Figure 3.9. Responses to each experiment experience survey prompt

The results indicate that participants were generally amenable to changes in assistance throughout the study and they generally found the changes in assistance helpful. The results to **Q2** may improve if the implementation of A_1 is improved. Finally, participants mostly expected changes in assistance via the audible alerts — this result is supported by a past, in-house pilot study on alert paradigms.

3.8.3. Participant Perceived Difficulty Analysis

Following the completion of each trial, each user was presented with the modified NASA TLX survey [**25**], [**24**]. The modified TLX survey [**30**] removes the TLX's typically administered pair-wise comparisons to simplify the survey. The subject- and course-wise TLX results are presented in Figure 3.10. The modified TLX scores range from 0 to 42 and a higher score indicates higher perceived difficulty. The presentation of the TLX questionnaire is presented in Appendix A.

Little discrepancy is observed in perceived difficulty between each course, demonstrating the comparability of all trials. The perceived difficulty varied minimally between

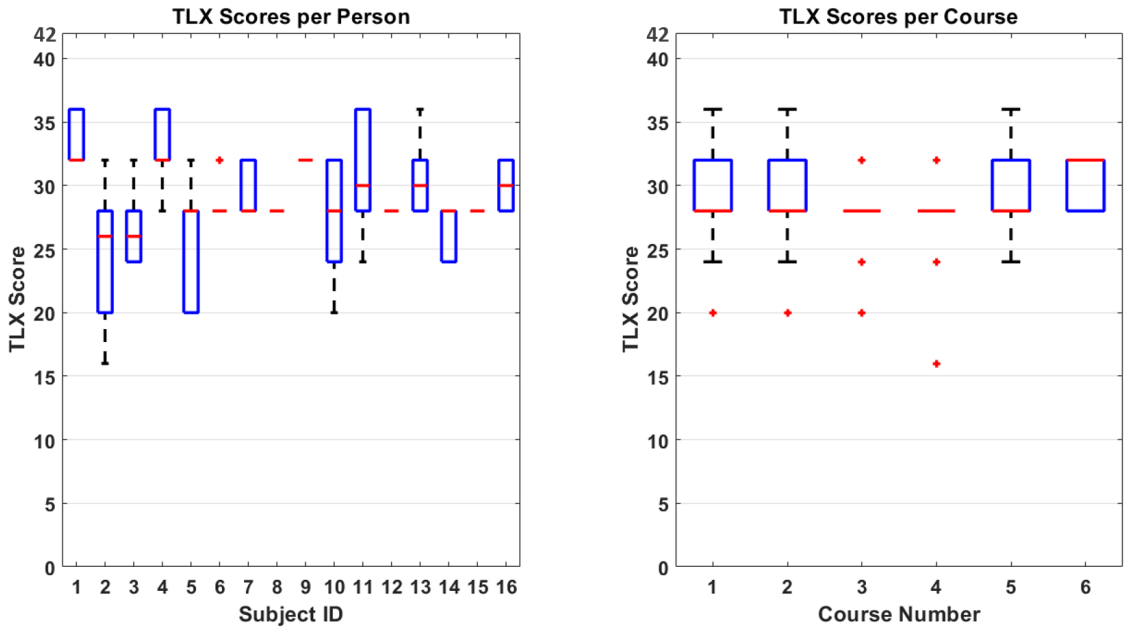

Figure 3.10. Subject-wise (**left**) and course-wise (**right**) TLX scores

participants as well, indicating the experiment was equally challenging for all participants. Thus, all subject-trial pair data sets are comparable in human experience. Finally, not enough data exists to draw conclusions about the perceived difficulty and the efficacy of the presented autonomy shifting paradigm.

3.8.4. Shift Cancellation Analysis

The number of cancellations to avoid shifting up or to avoid shifting down to any given LOA were nearly equal with 55 and 54 cancellations, respectively. The only observable pattern among the cancellation data is presented in Figure 3.11. Users almost exclusively 'canceled' shifts to avoid entering LOA $= A_1$. This further reinforces the need to redesign this LOA for future studies.

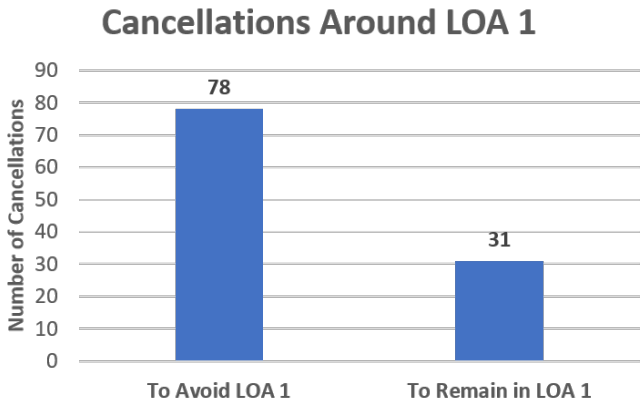

Figure 3.11. Analysis of the shifting cancellation data

CHAPTER 4

Conclusions

The aim of this research sought to inform the design of dynamic autonomy allocation frameworks by analyzing the classification power of different information streams within the human-robot team. This work successfully developed different machine learning models capable of classifying data from different information streams to inform an autonomous shifting framework 'when to shift' between discrete LOAs. Specific results and future directions — both near and far — are presented below.

4.1. Main Takeaways

This work proved that it is, indeed, possible to train an algorithm capable of answering the question 'when to shift' between discrete LOAs in a 1-dimensional assistance framework. It also showed that the most computationally intensive information streams — namely those from the environment — are not required in an autonomy shifting framework. Furthermore, and most interestingly, it was shown that the human's control inputs encode enough information to perform better than chance (13% improvement) alone. This demonstrates the possibility of autonomy shifting frameworks that review the human's inputs alone. However, future studies need to be conducted to present more concrete statements.

Monitoring the interaction between the human-robot team, however, poses significant potential for informing an autonomy shifting framework. The interaction information

streams alone beat chance by nearly 40%. At this time, it appears much more likely that future autonomy shifting frameworks could rely on the human-robot *interaction* information streams to determine 'when to shift.'

This work also demonstrated the power of hand-engineered features and the important of domain-specific specific knowledge within machine learning. Classical machine learning methods still pose significant benefits within robotics and shared control research. In short, bigger and deeper methods are not always better. By presenting results that *did not* perform, future researchers may not need to invest resources into testing or assessing these specific methods at a later date. *However,* there still exists potential to use different deep learning methods for feature extraction to minimize the need for manual feature extraction — this remains an area for future research. Another contribution of this work also provides future researchers with a benchmark to compare future autonomy shifting frameworks. The efficacy of two of three presented LOAs and highlights the need for a comprehensive study comparing the usefulness of different LOAs are also presented in this thesis.

This work may serve as an early attempt to address the latent space between the human and the robot when the robot is determining how to best help the human. In other words, this work attempts to estimate the human's desires and needs based on estimates of the human's desires for a population of able-bodied individuals. Approximating the human's needs and desires for a given environment using very little information about the individual the robot is assisting — humans address this via empathy, — remains an open problem and this work takes a small step towards a resolution. More advanced methods

will may initially estimate human needs from a population-wise data set and use transfer learning to update the initial model for an individual's unique needs and desires.

4.2. Future Directions

Immediate future directions of this work — for researchers wishing to expand on this work directly include:

- Testing deep networks with fewer units in the dense and LSTM layers as the models may be over fitting to the training data
- Test this data set using different autoencoders specifically designed to handle severely class-imbalanced data sets; this idea is inspired by [**36**].

When reviewing this work with a longer time horizon, the following are proposed for future study:

- The investigation of non-linear (2D) shifting paradigms and frameworks that seek to answer not only 'when to shift?' but also 'where to shift?' with a focus on *interaction* stream information
- The development of frameworks that adapt to an end-user's ever changing needs and capabilities possibly by using transfer learning to 'seed' an initial system for later adaptation

References

[1] ALT, H., AND GODAU, M. COMPUTING THE FRéCHET DISTANCE BETWEEN TWO POLYGONAL CURVES. *International Journal of Computational Geometry & Applications 05*, 01n02 (Mar. 1995), 75–91.

[2] ANGUITA, D., GHIO, A., ONETO, L., PARRA, X., AND REYES-ORTIZ, J. L. A public domain dataset for human activity recognition using smartphones. In *ESANN 2013* (2013), 21th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2013, i6doc – Ciaco cooperative, pp. 437–442.

[3] BAGNELL, D., AND SINGH, S. Statistical techniques in robotics (16-831, f11) lecture 23, December 2010.

[4] BALASUBRAMANIAN, S., MELENDEZ-CALDERON, A., AND BURDET, E. A robust and sensitive metric for quantifying movement smoothness. *IEEE Transactions on Biomedical Engineering 59*, 8 (Aug. 2012), 2126–2136.

[5] BALASUBRAMANIAN, S., MELENDEZ-CALDERON, A., ROBY-BRAMI, A., AND BURDET, E. On the analysis of movement smoothness. *Journal of NeuroEngineering and Rehabilitation 12*, 1 (Dec. 2015).

[6] BAUM, T. E., WOLKOWICZ, K. L., CHOBOT, J. P., AND BRENNAN, S. N. Negative obstacle detection using LiDAR sensors for a robotic wheelchair. In *The Proceedings of the Dynamics Systems and Controls Conference* (Sept. 2018), American Society of Mechanical Engineers.

[7] BI, L., FAN, X., AND LIU, Y. Eeg-based brain-controlled mobile robots: A survey. *IEEE Transactions on Human-Machine Systems 43*, 2 (2013), 161–176.

[8] BOER, E. R., AND RAKAUSKAS, M. E. Steering entropy revisited. In *Driving assessment 2005 : proceedings of the 3rd International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design* (2005), University of Iowa.

[9] Breiman, L. Random forests. *Machine Learning 45*, 1 (2001), 5–32.

[10] Broad, A., Murphey, T., and Argall, B. Operation and imitation under safety-aware shared control. In *Springer Proceedings in Advanced Robotics*. Springer International Publishing, 2020, pp. 905–920.

[11] Broad, A., Murphey, T., and Argall, B. D. Highly parallelized data-driven MPC for minimal intervention shared control. *CoRR abs/1906.02318* (2019).

[12] Broad, A., Schultz, J., Derry, M., Murphey, T., and Argall, B. Trust adaptation leads to lower control effort in shared control of crane automation. *IEEE Robotics and Automation Letters 2*, 1 (Jan. 2017), 239–246.

[13] Brownlee, J. How to develop rnn models for human activity recognition time series classification, Mar 2020.

[14] Chen, M., Nikolaidis, S., Soh, H., Hsu, D., and Srinivasa, S. Planning with trust for human-robot collaboration. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (Feb. 2018), ACM.

[15] Chiou, M., Stolkin, R., Bieksaite, G., Hawes, N., Shapiro, K. L., and Harrison, T. S. Experimental analysis of a variable autonomy framework for controlling a remotely operating mobile robot. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Oct. 2016), IEEE.

[16] Coppin, G., and Legras, F. Controlling swarms of unmanned vehicles through user-centered commands.

[17] Cramer, J. The Origins of Logistic Regression. Tinbergen Institute Discussion Papers 02-119/4, Tinbergen Institute, Dec. 2002.

[18] Crammer, K., and Singer, Y. On the algorithmic implementation of multiclass kernel-based vector machines. *J. Mach. Learn. Res. 2* (Mar. 2002), 265–292.

[19] D, M. The u.s. air force-developed adaptation of the multi-attribute task battery for the assessment of human operator workload and strategic behavior. 152.

[20] Erdogan, A., and Argall, B. D. The effect of robotic wheelchair control paradigm and interface on user performance, effort and preference: An experimental assessment. *Robotics and Autonomous Systems 94* (Aug. 2017), 282–297.

[21] Franklin, G. F., Powell, J. D., and Emami-Naeini, A. *Feedback Control of Dynamic Systems (7th Edition)*, 7th ed. Pearson, 2015.

[22] Friedman, J. H. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics 29*, 5 (2001), 1189–1232.

[23] Gopinath, D., Jain, S., and Argall, B. D. Human-in-the-loop optimization of shared autonomy in assistive robotics. *IEEE Robotics and Automation Letters 2*, 1 (Jan. 2017), 247–254.

[24] Hart, S. G. Nasa-task load index (NASA-TLX): 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting 50*, 9 (Oct. 2006), 904–908.

[25] Hart, S. G., and Staveland, L. E. Development of NASA-TLX (task load index): Results of empirical and theoretical research. In *Advances in Psychology*. Elsevier, 1988, pp. 139–183.

[26] Horn, O. Smart wheelchairs: Past and current trends. In *2012 1st International Conference on Systems and Computer Science (ICSCS)* (Aug. 2012), IEEE.

[27] Javdani, S., Admoni, H., Pellegrinelli, S., Srinivasa, S. S., and Bagnell, J. A. Shared autonomy via hindsight optimization for teleoperation and teaming. *The International Journal of Robotics Research 37*, 7 (June 2018), 717–742.

[28] Johnson, J. M., and Khoshgoftaar, T. M. Survey on deep learning with class imbalance. *Journal of Big Data 6*, 1 (Mar. 2019).

[29] Kim, H., Biggs, S., Schloerb, D., Carmena, J., Lebedev, M., Nicolelis, M., and Srinivasan, M. Continuous shared control for stabilizing reaching and grasping with brain-machine interfaces. *IEEE Transactions on Biomedical Engineering 53*, 6 (June 2006), 1164–1173.

[30] Kucukyilmaz, A., and Demiris, Y. Learning shared control by demonstration for personalized wheelchair assistance. *IEEE Transactions on Haptics 11*, 3 (July 2018), 431–442.

[31] Madarasz, R., Heiny, L., Cromp, R., and Mazur, N. The design of an autonomous vehicle for the disabled. *IEEE Journal on Robotics and Automation 2*, 3 (1986), 117–126.

[32] Musić, S., and Hirche, S. Control sharing in human-robot team interaction. *Annual Reviews in Control 44* (2017), 342–354.

[33] Nejati, M., and Argall, B. D. Characterization of assistive robot arm teleoperation: A preliminary study to inform shared control. *arXiv: Robotics [cs.RO] arXiv/3269586* (2020).

[34] Nejati, M., Young, M., and Argall, B. D. Interface operation and implications for shared-control assistive robots. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)* (June 2019), IEEE.

[35] Northcutt, C. G. hypopt: hyper-parameter optimization using a validation set library. `https://pypi.org/project/hypopt/`, 2018–2019.

[36] Park, D., Hoshi, Y., and Kemp, C. C. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. *IEEE Robotics and Automation Letters 3*, 3 (July 2018), 1544–1551.

[37] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research 12*, 85 (2011), 2825–2830.

[38] Reddy, S., Levine, S., and Dragan, A. D. Shared autonomy via deep reinforcement learning. *CoRR abs/1802.01744* (2018).

[39] Rennie, J. D. M., Shih, L., Teevan, J., and Karger, D. R. Tackling the poor assumptions of naive bayes text classifiers. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning* (2003), ICML'03, AAAI Press, p. 616–623.

[40] Sainath, T. N., Vinyals, O., Senior, A., and Sak, H. Convolutional, long short-term memory, fully connected deep neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (Apr. 2015), IEEE.

[41] Santiago-Espada, Y., Myer, R. R., Latorella, K. A., and Comstock, James R., J. The multi-attribute task battery II (MATB-II) software for human performance and workload research: A user's guide. Tech. Rep. NASA/TM-2011-217164, L-20031, NF1676L-12800, The National Aeronautics Space Administration: Langley Research Center, 2011.

[42] Sheridan, T. B., and Verplank, W. L. Human and computer control of undersea teleoperators. Tech. Rep. ADA057655, Cambridge, Massachusetts, 1976.

[43] Shi, X., Chen, Z., Wang, H., Yeung, D., Wong, W., and Woo, W. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *CoRR abs/1506.04214* (2015).

[44] Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A., and Goodrich, M. Common metrics for human-robot interaction. In *Proceeding of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction - HRI '06* (2006), ACM Press.

[45] Tharwat, A. Classification assessment methods. *Applied Computing and Informatics* (Aug. 2018).

[46] Thomas, J., Comoretto, L., Jin, J., Dauwels, J., Cash, S. S., and Westover, M. B. EEG CLassification via convolutional neural network-based interictal epileptiform event detection. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (July 2018), IEEE.

[47] Wang, J., and Lewis, M. Human control for cooperating robot teams. In *2007 2nd ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (2007), pp. 9–16.

[48] Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., and Xu, W. Cnn-rnn: A unified framework for multi-label image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016).

[49] Xu, X., Tabuada, P., Grizzle, J. W., and Ames, A. D. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine 48*, 27 (2015), 54–61.

[50] Young, M., Nejati, M., and Argall, B. Discrete N-dimensional entropy of behavior: DNDEB. In *Proceedings of the IEEE International Conference on Intelligent Robots (IROS)* (2019), IEEE.

[51] Young, M., Nejati, M., Erdogan, A., and Argall, B. D. An analysis of degraded communication channels in human-robot teaming and implications for dynamic autonomy allocation. In *Field and Service Robotics* (2017), M. Hutter and R. Siegwart, Eds., Springer Proceedings in Advanced Robotics, Springer International Publishing, pp. 665–679. Book DOI: 10.1007/978-3-319-67361-5.

[52] Zhang, H. The optimality of naïve bayes. In *In FLAIRS2004 conference* (2004).

APPENDIX A

Modified NASA TLX Questionnaire

The following presents the modified NASA-TLX survey questions presented to study participants. These are from [**30**] which are inspired by [**25**]. Each statement is presented with a 7-point Likert scale where a 7 indicates "strongly agree" and a 1 indicates "Strongly disagree." The scores are summed together without weighting – the score's range is from 0 to 42. A higher score indicates a perceived higher human cognitive load.

(1) **Mental Demand:** The task required a large amount of mental and perceptual activity (e.g., thinking, deciding, calculating, remembering, looking, searching, etc)

(2) **Physical Demand:** The task required a large amount of physical activity (e.g., pulling, pushing, turning, controlling, activating, etc.)

(3) **Temporal Demand:** I needed to be quick to perform the task.

(4) **Performance:** I was successful in accomplishing the goals of the task set by the experimenter (or myself).

(5) **Effort:** I had to work hard (mentally and physically) to accomplish the task.

(6) **Frustration Level:** I felt irritated / stressed / annoyed during the task.

Vita

Christopher Miller was born in Abington, Pennsylvania in 1993 to Christopher and Mary Miller. He was raised in Audubon, Pennsylvania. Before attending Northwestern University, Chris received his Bachelor of Science in Electrical Engineering with honors from The Pennsylvania State University in 2016. During college, Chris received multiple scholarships, interned at NASA's Jet Propulsion Laboratory, served as the student IEEE president, and submitted an undergraduate thesis under the guidance of Dr. Sean Brennan. From 2016 to 2017, he was a Staff Electrical Engineer at Carnegie Mellon University's National Robotics Engineering Center (NREC).

While at Northwestern, Chris researched shared autonomy for persons with disabilities in the Shirley Ryan AbilityLab as a member of the argallab and under the guidance of Dr. Brenna Argall. Chris received both the National Science Foundation's Graduate Research Fellowship and the United States Department of Defense's National Defense Science and Engineering Graduate (NDSEG) Research Fellowship in 2017. He accepted the NDSEG.

Shortly after graduation, Chris worked at an engineering services firm near Los Angeles, CA. As of 2021, Chris is a Product Manager at TuSimple in San Diego, California. He lives in San Diego, California.